

Selective Learning for Sample-Efficient Training in Multi-Agent Sparse Reward Tasks (Extended Abstract)*

Xinning Chen¹, Xuan Liu^{1†}, Yanwen Ba¹, Shigeng Zhang², Bo Ding³, Kenli Li¹

¹College of Computer Science and Electronic Engineering, Hunan University, China

²School of Computer Science and Engineering, Central South University, China

³School of Computer Science, National University of Defense Technology, China

{chenxinning,xuan_liu,yanwenba,lkl}@hnu.edu.cn, sgzhang@csu.edu.cn, dingbo@nudt.edu.cn

Abstract

Learning effective strategies in sparse reward tasks is one of the fundamental challenges in reinforcement learning. This becomes extremely difficult in multi-agent environments, as the concurrent learning of multiple agents induces the non-stationarity problem and a sharply increased joint state space. Existing works have attempted to promote multi-agent cooperation through experience sharing. However, learning from a large collection of shared experiences is inefficient as there are only a few high-value states in sparse reward tasks, which may instead lead to the curse of dimensionality in large-scale multi-agent systems. This paper focuses on sparse-reward multi-agent cooperative tasks and proposes an effective experience-sharing method, **Multi-Agent Selective Learning (MASL)**, to boost sample-efficient training by reusing valuable experiences from other agents. MASL adopts a retrogression-based selection method to identify high-value traces of agents from the team rewards, based on which some recall traces are generated and shared among agents to motivate effective exploration. Moreover, MASL selectively considers information from other agents to cope with the non-stationarity issue while enabling efficient training for large-scale agents. Experimental results show that MASL significantly improves sample efficiency compared with state-of-the-art MARL algorithms in cooperative tasks with sparse rewards.

1 Introduction

Deep reinforcement learning (DRL) has shown its advantage in sequential decision-making control tasks, such as Atari games, game theory and robot control [Hosu and Rebedea, 2016; Li *et al.*, 2020; Chu *et al.*, 2020]. However, recent success depends heavily on a well-formed reward function that provides explicit feedback to each agent at each step.

*This is an extended abstract of the paper [Chen *et al.*, 2023] that won the Outstanding Paper Award at ECAI 2023.

†Corresponding Author.

For some complex tasks with sparse rewards, such as autonomous driving and robotic control, learning an optimal policy becomes extremely difficult for agents due to the lack of feedback signals. Sparse rewards are delayed, which provide feedback to agents only in a few states (e.g., when the agent reaches a goal), while in most cases, agents are not rewarded. Discovering high-value states in sparse reward tasks is a hard exploration problem for agents, which has not been well studied in the RL domain.

For single-agent environments, previous studies on sparse reward have improved exploration efficiency by using additional supervised signals, such as expert demonstrations [Stadie *et al.*, 2017; Ziebart *et al.*, 2008], intrinsic rewards [Bellemare *et al.*, 2016; Pathak *et al.*, ; Tang *et al.*, 2017]. However, these methods may struggle to learn effective strategies when directly applied to multi-agent environments, since the large joint state space induced by the co-evolution of multiple agents poses a great challenge to policy learning. Moreover, the sparse reward challenge in multi-agent tasks is aggravated by the need for policy coupling and the non-stationarity of multi-agent environments.

To promote cooperation in large state space, existing multi-agent reinforcement learning (MARL) algorithms take advantage of experience sharing. The centralized training with decentralized execution (CTDE) paradigm is widely used to enable agents to share their experience during centralized training [Foerster *et al.*, 2018; Li *et al.*, 2019; Lowe *et al.*, 2017]. However, fully sharing experience among agents may lead to the curse of dimensionality as the joint state-action space grows exponentially with the number of agents [Nguyen *et al.*, 2020]. Recent works have tried to simplify the learning process by adopting techniques of inverse kinematic [Kubus *et al.*, 2018; Perrusquía *et al.*, 2021], attention mechanism [Iqbal and Sha, 2019; Jiang and Lu, 2018; Liu *et al.*, 2020], mean field theory [Yang *et al.*, 2018] and dropout [Kim *et al.*, 2019]. However, useful information may be overlooked in the process of reducing interaction. How to maximize the valuable experiences while simplifying the interaction remains a question, which is particularly important in the tasks with sparse reward settings.

In this paper, we focus on sparse reward in cooperative multi-agent scenarios, where both the challenges brought by sparse reward and inherent non-stationarity greatly reduce learning efficiency. We propose a method called multi-agent

selective learning (MASL) to achieve sample-efficient training. The key idea of MASL is to select only valuable experiences of other agents instead of the whole huge trajectory space to accelerate the learning process. First, inspired by the advanced idea of using a backtracking model to improve sample efficiency [Goyal *et al.*, 2019], we introduce a centralized backtracking model for multiple agents, which generates recall traces from high-value samples. More importantly, each agent not only speeds up learning by imitating its own recall traces, but also shares the traces with other agents for aiding effective exploration. Second, we propose to selectively use information of the related agents but not all, which effectively mitigates the non-stationarity of the multi-agent environments and enhances the scalability of our approach in scenarios with more agents. Last but not least, considering that it is difficult to identify high-value experiences when all agents obtain a shared team reward, we specifically consider fully cooperative tasks with shared team reward and design a retrogression-based selection method to overcome the difficulty of recognizing contributors from the shared reward.

The contribution of this paper is summarized as fourfold:

1) We propose multi-agent selective learning (MASL), an effective method to boost learning efficiency in multi-agent cooperative tasks with sparse rewards.

2) We introduce a centralized backtracking model to guide cooperative exploration and exploit a retrogression-based selection method to extract high-value agent trajectories from the team’s success.

3) We propose a selector to selectively use information from other agents based on their relevancy to improve the training speed of centralized learning.

4) Experiment results in several cooperative multi-agent tasks with sparse rewards show that MASL achieves higher sample efficiency than the baselines, especially in tasks with large-scale agents.

This paper is only an abbreviated version of the ECAI 2023 conference paper [Chen *et al.*, 2023]. We refer the reader to the full paper for all the details omitted below.

2 Methodology

To address the challenge of sparse reward and non-stationarity, MASL utilizes two techniques: 1) it backtracks the trajectories of high-value states to help agents avoid the unguided exploration process to speed up training. 2) it selects the information of other K agents based on the relevancy, rather than all agents, balancing the stability and learning efficiency of policy learning.

MASL is built on the framework of centralized training and decentralized execution. As shown in Fig.1, each agent i learns an independent deterministic policy $\mu_i(a_i|o_i; \theta_i)$ parameterized by θ_i and a Q-network $Q_i(o_1, \dots, o_N, a_1, \dots, a_N; \omega_i)$ parameterized by ω_i , where o_i and a_i represent observation and action, respectively. More importantly, the selectors for agents and a centralized backtracking model B_ϕ are used for selective learning in the training phase. During the execution, the backtracking model, selectors and Q-value networks will be removed, allowing each agent to act based on its local policy.

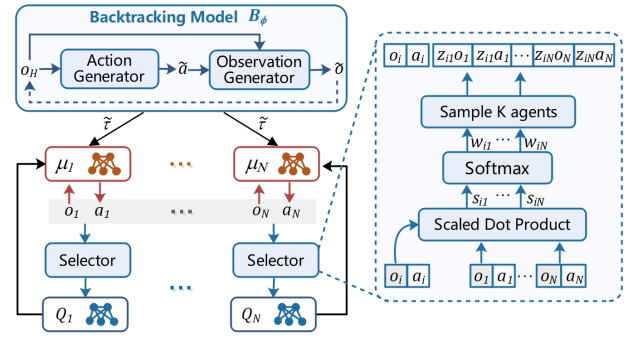


Figure 1: The architecture of MASL.

2.1 Backtracking Based on Optimal Trajectory Selection

To improve sample efficiency in sparse reward tasks, we introduce the backtracking model in RL to multi-agent tasks. However, it is inefficient to directly apply the model to MARL because of the need to identify high-value states and the neglect of collaboration. Therefore, we mainly deal with two key problem: 1) How to improve multi-agent cooperation using the backtracking model? 2) How to identify high-value trajectory when all agents share a reward?

First, we design a global backtracking model that is shared among all agents to facilitate cooperative exploration. When an agent discovers a high-value state within the vast state space, we utilize the high-value states to generate recall traces, which are then shared among agents to expedite the exploration process. An example is shown in Fig.2, where two agents are exploring the environment to reach the targets. When agent A2 discovers a target (e.g., a landmark) and successfully identifies a high-value state (i.e., observation in partially observable environments), A2 shares the recall traces generated from this high-value state to guide the other agent to reach the known high-value states from a new path. In this way, agent A1 can quickly recognize the target and acquire the individual ability to reach a certain target in the early stage. Second, we propose a retrogression-based selection method to select the optimal trajectories.

Backtracking Model for Multi-agent Scenarios. In the training phase, we maintain a centralized backtracking model B_ϕ shared by all agents. The backtracking model B_ϕ utilizes agents’ good private states to generate traces for N agents to train their policies. As shown in Fig.1, B_ϕ is composed of an observation generator and an action generator. Given a high-value observation o_i^H of agent i , the action generator $q(\tilde{a}_i|o_i^H)$ generates an action \tilde{a}_i that may cause the high-value state. The observation generator $q(\Delta o_i|o_i^H, \tilde{a}_i)$ outputs the observation variation $\Delta o_i = \tilde{o}_i - o_i^H$ according to o_i^H and \tilde{a}_i , by which we get a previous observation \tilde{o}_i indirectly and then further obtain a sequence of $(\tilde{o}_i, \tilde{a}_i)$ -tuples as recall trace $\tilde{\tau}_i$ for policy training.

To achieve cooperation exploration, a natural idea is knowledge sharing. Therefore, each agent shares their recall traces with the other agents. We omit the the subscript of recall traces in the following content. Given observation \tilde{o} , the

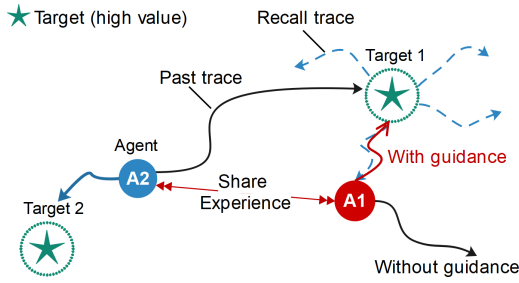


Figure 2: An example of exploiting optimal experience to provide guidance.

policy μ_i parameterized by θ_i is updated to guide agents to take action \tilde{a} as:

$$L(\theta_i) = \log \mu_i(\tilde{\tau}) = \sum_{t=0}^M \log \mu_i(\tilde{a}|\tilde{o}) \quad (1)$$

The policy for each agent is updated by two terms, one over the trajectories collected by interaction and another over the recall traces generated by the backtracking model as Eq.1.

Since the backtracking model B_ϕ predicts previous actions and observations based on agents' high-value observations, the distribution of recall traces should match the distribution of the high-value trajectory as closely as possible. Given a high-value trajectory $\tau = (o^1, a^1, r^1, \dots, o^T, a^T, r^T)$, B_ϕ is trained by:

$$\begin{aligned} \mathcal{L}(\phi) &= \log \prod_{t=0}^T q(\Delta o_t, a_t | o_{t+1}) \\ &= \sum_{t=0}^T \log q(a_t | o_{t+1}) + \log q(\Delta o_t | a_t, o_{t+1}). \end{aligned} \quad (2)$$

Note that agents only share the recall traces with other agents during training, and act independently during execution.

Retgression-based Trajectory Selection. To build an effective backtracking model, a key problem is to identify high value trajectories, which could be difficult when all agents share a team reward. We propose a retrogression-based selection method to recognize contributors from the shared reward, which infers all the contributors by letting all the agents to take a step backward in turn and observe how reward changes. The selected high-value trajectories are deposited to a high-value buffer \tilde{B} .

2.2 Training Based on Key Information Selection

To balance stability and scalability, we allow agents to obtain information from the other agents for stable training, and we introduce a selector for each agent to select key information for efficient learning and scalability. Driven by the intuition that neighboring agents are more likely to interact, the selector computes the relevancy weight w_{ij} for each agent j by matching their observations. As shown in Fig.1, the weight w_{ij} compares o_i with o_j by using the computationally efficient scale dot score function to get $s_{ij}(o_j, o_i) = \frac{o_j^T o_i}{\sqrt{d}}$, where d is the dimensionality of the observation. The matching values are passed into a Softmax function to obtain relevancy weights:

$$w_{ij} = \frac{\exp(s_{ij}(o_j, o_i))}{\sum_{j \neq i} \exp(s(o_k, o_i))}. \quad (3)$$

Then, we get a normalized weight vector $W_i \triangleq (w_{i1}, \dots, w_{ij}, \dots, w_{iN})$, $j \neq i$, which satisfies $\sum_{j \neq i} w_{ij} \equiv 1$. Then, we select K agents in a sample way and obtain an vector $z_i = \{(z_{i1}, \dots, z_{ij}, \dots, z_{iN}), z_{ij} \in [0, 1], \}$, where $z_{ij} = 1$ indicate the agent j is selected. The filtered information is replaced with a zero vector.

During off-line learning, the networks are updated using mini-batch samples. The selector chooses K agents based on the average weight of the mini-batch samples. Then, it can be seen as the information of the unselected agents is dropped out in current training round. This helps narrow down the critic networks and leads to faster training.

3 Experiments

In this section, we perform experiments to investigate the effectiveness of our method on two continuous control tasks of resource collection and rover exploration, which are modified from the widely used multiple-particle environment (MPE) benchmark [Lowe *et al.*, 2017]. we compare MASL with several state-of-the-art solutions, including MADDPG [Lowe *et al.*, 2017], DDPG [Lillicrap *et al.*, 2016], MADDPG-MD [Kim *et al.*, 2019], MF [Yang *et al.*, 2018], MAAC [Iqbal and Sha, 2019] and IGASIL [Hao *et al.*, 2019].

3.1 Resource Collection Tasks

The resource collection task requires N agents to collect L resources in multiple resource pools while avoiding collision, as shown in Fig.3(a). In Fig. 3, we plot the episode reward of different algorithms as the training progresses. MASL outperforms the other methods in all the cases. It is clear that MASL learns more rapidly and reaches higher episode rewards earlier than the baselines. In contrast, the other methods converge to suboptimal performance due to their limited exploration. Moreover, it is noteworthy that as the number of agents increases, MASL maintains its superior performance in terms of learning speed and final reward attainment.

3.2 Rover Exploration Tasks

The rover exploration task involves N agents learning to cooperatively explore L distinct target areas separately, as in Fig. 4(a). The number of agents needed to explore a target area is referred to as the coupling requirement, which remains unknown to the agents. The rover exploration tasks bring more challenges to agents due to the requirement of deep cooperation.

As shown in Fig. 4(b), the reward curve for MASL demonstrates notably faster growth compared to the other methods. MASL agents learn quickly to form a team of two to explore an area in RE-6 scenarios. Similar results can be found in RE-8 scenario ($N = 8, L = 4$). As the number of agents increases, the joint state space expands quickly, which slows down the training process. However, MASL still maintains an advanced performance (see Fig. 4(c)). We observe that once an agent reaches a target occasionally, MASL quickly learns to form teams to explore different targets.

To evaluate the performance of MASL in long-horizon tasks, we conduct experiment on the 100-step rover exploration task with 12 agents and 3 areas ($N = 12, L = 3$),

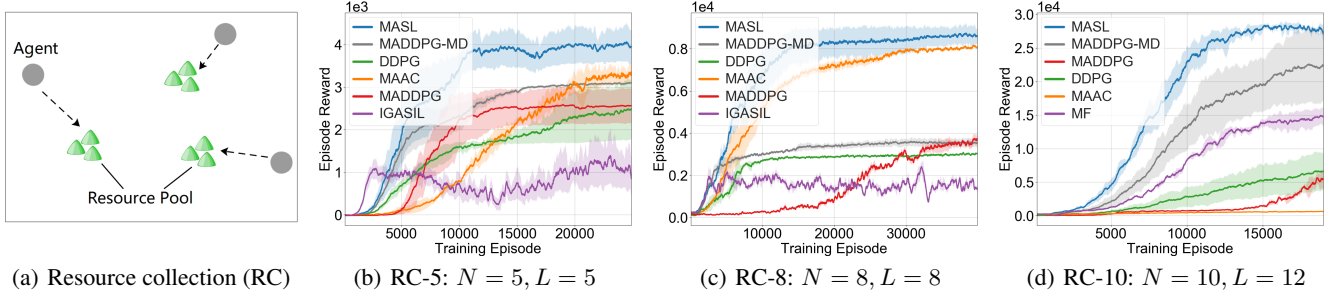


Figure 3: Resource collection tasks: (a) Task description; (b-d) Episode rewards in three settings.

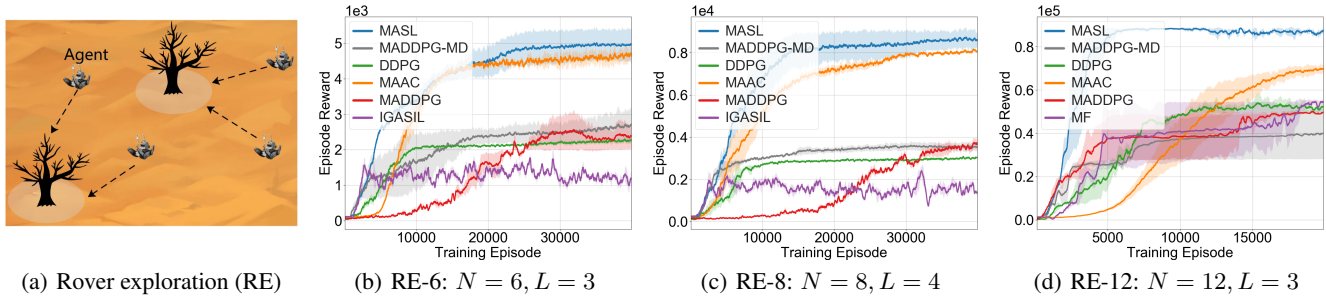


Figure 4: Rover exploration tasks: (a) Task description; (b-d) Episode rewards in three settings.

which requires agents to explore 3 areas with a high coupling requirement of 4. As shown in Fig.4(d), MASL still achieves higher sample efficiency in the long-term games compared to the baselines of MADDPG-MD and MADDPG.

3.3 Discussion

In this section, we study the scalability of MASL to show its advantage in sparse reward tasks. We train agents in the resource collection scenarios with $N = [5, 8, 10, 20]$ agents respectively, and then report the final target completion rates of MASL compared with MADDPG-MD and mean field reinforcement learning (MF), which are designed to address large-scale agents. MF is added as a baseline method, which approximates the interaction within agents using an average effect to enable coordination between large-scale agents. As shown in the Fig. 5(a), MASL achieves higher target completion rates than the other baselines, even in the large-scale scenario with 20 agents. We observe that MASL only sacrifices 0.93% of the computation time to improve the sample efficiency by 8.8% compared with MADDPG-MD when $N = 5$. The sample efficiency of MASL is improved by 30.2% in the case of $N = 20$. We also study the scalability of the trained MASL agents by transferring the trained policies of 5 agents to 30 agents, and testing MASL in the environment of 30 agents ($N = 30, L = 30$). We observed that the trained decentralized policies of MASL in easy tasks can be directly scale to complex tasks with large-scale agents, while achieving a higher target completion rate of 36%, surpassing MADDPG-MD (24.7%). MASL enhances sample efficiency for sparse reward tasks, especially in large-scale scenarios. We also perform ablation studies, which show that

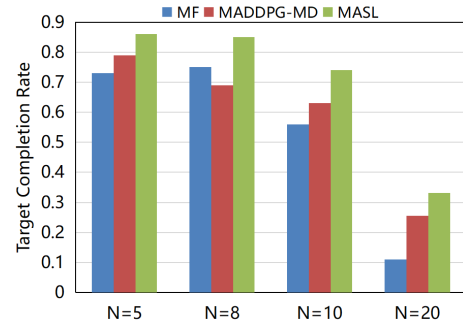


Figure 5: Scalability results: The target completion rates in the resource collection tasks with different number of agents.

learning from recall traces and focusing on relevant agents jointly accelerate training and aid in exploration. For more results, please refer to the full paper [Chen *et al.*, 2023].

4 Conclusions

In this paper, we propose an efficient training method called multi-agent selective learning (MASL), to improve sample efficiency for multi-agent sparse-reward tasks. By using a centralized backtracking model, MASL learns not only from traces obtained by interacting with the environment but also from recall traces generated by the backtracking model. Moreover, we design a selector to improve learning efficiency while balancing stability. Experiments show that MASL significantly speeds up learning in several multi-agent sparse-reward tasks, especially in tasks with large-scale agents.

Acknowledgments

This work was supported by the National Key Research and Development Program of China (2022YFC3400404), the National Science Foundation of China (62172154), the Hunan Provincial Natural Science Foundation of China under grant No. 2023JJ30702. Prof. Xuan Liu is the corresponding author of the paper.

References

- [Bellemare *et al.*, 2016] Marc G. Bellemare, Sriram Srinivasan, Georg Ostrovski, Tom Schaul, David Saxton, and Rémi Munos. Unifying count-based exploration and intrinsic motivation. In *Proc. Adv. Neural Inf. Process. Syst.*, pages 1471–1479, 2016.
- [Chen *et al.*, 2023] Xinning Chen, Xuan Liu, Yanwen Ba, Shigeng Zhang, Bo Ding, and Kenli Li. Selective learning for sample-efficient training in multi-agent sparse reward tasks. In *26th European Conference on Artificial Intelligence*, volume 372 of *Frontiers in Artificial Intelligence and Applications*, pages 413–420. IOS Press, 2023.
- [Chu *et al.*, 2020] Tianshu Chu, Sandeep Chinchali, and Sachin Katti. Multi-agent reinforcement learning for networked system control. In *ICLR*. OpenReview.net, 2020.
- [Foerster *et al.*, 2018] Jakob N. Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. Counterfactual multi-agent policy gradients. In *Proc. AAAI Conf. Artif. Intell.*, pages 2974–2982, 2018.
- [Goyal *et al.*, 2019] Anirudh Goyal, Philemon Brakel, William Fedus, Soumye Singhal, Timothy P. Lillicrap, Sergey Levine, Hugo Larochelle, and Yoshua Bengio. Recall traces: Backtracking models for efficient reinforcement learning. In *Proc. ICLR*, 2019.
- [Hao *et al.*, 2019] Xiaotian Hao, Weixun Wang, Jianye Hao, and Yaodong Yang. Independent generative adversarial self-imitation learning in cooperative multiagent systems. In *Proc. AAMAS*, pages 1315–1323, 2019.
- [Hosu and Rebedea, 2016] Ionel-Alexandru Hosu and Traian Rebedea. Playing atari games with deep reinforcement learning and human checkpoint replay. *CoRR*, abs/1607.05077, 2016.
- [Iqbal and Sha, 2019] Shariq Iqbal and Fei Sha. Actor-attention-critic for multi-agent reinforcement learning. In *Proc. Int. Conf. Mach. Learn.*, 2019.
- [Jiang and Lu, 2018] Jiechuan Jiang and Zongqing Lu. Learning attentional communication for multi-agent cooperation. In *Proceedings of Conference on Neural Information Processing Systems*, pages 7265–7275, 2018.
- [Kim *et al.*, 2019] Woojun Kim, Myungsik Cho, and Youngchul Sung. Message-dropout: An efficient training method for multi-agent deep reinforcement learning. In *Proc. AAAI Conf. Artif. Intell.*, pages 6079–6086, 2019.
- [Kubus *et al.*, 2018] Daniel Kubus, Rania Rayyes, and Jochen J. Steil. Learning forward and inverse kinematics maps efficiently. In *International Conference on Intelligent Robots and Systems*, pages 5133–5140. IEEE, 2018.
- [Li *et al.*, 2019] Shihui Li, Yi Wu, Xinyue Cui, Honghua Dong, Fei Fang, and Stuart J. Russell. Robust multi-agent reinforcement learning via minimax deep deterministic policy gradient. In *Proc. AAAI Conf. Artif. Intell.*, pages 4213–4220, 2019.
- [Li *et al.*, 2020] Haoran Li, Qichao Zhang, and Dongbin Zhao. Deep reinforcement learning-based automatic exploration for navigation in unknown environment. *IEEE Trans. Neural Networks Learn. Syst.*, 31(6):2064–2076, 2020.
- [Lillicrap *et al.*, 2016] Timothy P. Lillicrap, Jonathan J. Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. In *Proc. ICLR*, 2016.
- [Liu *et al.*, 2020] Yong Liu, Weixun Wang, Yujing Hu, Jianye Hao, Xingguo Chen, and Yang Gao. Multi-agent game abstraction via graph attention neural network. In *Proc. AAAI Conf. Artif. Intell.*, pages 7211–7218, 2020.
- [Lowe *et al.*, 2017] Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, Pieter Abbeel, and Igor Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. In *Proc. Adv. Neural Inf. Process. Syst.*, pages 6379–6390, 2017.
- [Nguyen *et al.*, 2020] Thanh Thi Nguyen, Ngoc Duy Nguyen, and Saeid Nahavandi. Deep reinforcement learning for multiagent systems: A review of challenges, solutions, and applications. *IEEE Trans. Cybern.*, 50(9):3826–3839, 2020.
- [Pathak *et al.*,] Deepak Pathak, Pulkit Agrawal, Alexei A. Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In *Proc. Int. Conf. Mach. Learn.*, pages 2778–2787.
- [Perrusquía *et al.*, 2021] Adolfo Perrusquía, Wen Yu, and Xiaoou Li. Multi-agent reinforcement learning for redundant robot control in task-space. *Int. J. Mach. Learn. Cybern.*, 12(1):231–241, 2021.
- [Stadie *et al.*, 2017] Bradley C. Stadie, Pieter Abbeel, and Ilya Sutskever. Third person imitation learning. In *Proc. ICLR*, 2017.
- [Tang *et al.*, 2017] Haoran Tang, Rein Houthooft, Davis Foote, Adam Stooke, Xi Chen, Yan Duan, John Schulman, Filip De Turck, and Pieter Abbeel. #exploration: A study of count-based exploration for deep reinforcement learning. In *Proc. Adv. Neural Inf. Process. Syst.*, pages 2753–2762, 2017.
- [Yang *et al.*, 2018] Yaodong Yang, Rui Luo, Minne Li, Ming Zhou, Weinan Zhang, and Jun Wang. Mean field multi-agent reinforcement learning. In *Proc. Int. Conf. Mach. Learn.*, pages 5567–5576, 2018.
- [Ziebart *et al.*, 2008] Brian D. Ziebart, Andrew L. Maas, J. Andrew Bagnell, and Anind K. Dey. Maximum entropy inverse reinforcement learning. In *Proc. AAAI Conf. Artif. Intell.*, pages 1433–1438, 2008.