Learning to Advise and Learning from Advice in Cooperative Multiagent Reinforcement Learning

Extended Abstract

Yue Jin
Tsinghua University
jiny23@126.com

Shuangqing Wei
Louisiana State University
swei@lsu.edu

Jian Yuan
Tsinghua University
jyuan@tsinghua.edu.cn

Xudong Zhang
Tsinghua University
zhangxd@tsinghua.edu.cn

ABSTRACT

We propose a novel policy-level generative adversarial learning framework to enhance cooperative multiagent reinforcement learning (MARL), which consists of a centralized advisor, MARL agents and discriminators. The advisor is realized through a dual graph convolutional network (DualGCN) to give advice to agents from a global perspective via fusing decision information, resolving spatial conflicts, and maintaining temporal continuity. Each discriminator trained can distinguish between the policies of the advisor and an agent. Leveraging the discriminator's judgment, each agent learns to match with the advised policy in addition to learning by its own exploration, which accelerates learning and enhances policy performance. Additionally, an advisor boosting method which incorporates the relevant suggestion made by the discriminators into the training of DualGCN is proposed to further help improve MARL agents. We validate our methods in cooperative navigation tasks. Results demonstrate that our method outperforms baseline methods in terms of both learning efficiency and policy efficacy.

KEYWORDS

Multiagent Reinforcement Learning; Coordination; Graph Neural Network; Generative Adversarial Network

ACM Reference Format:
Yue Jin, Shuangqing Wei, Jian Yuan, and Xudong Zhang. 2022. Learning to Advise and Learning from Advice in Cooperative Multiagent Reinforcement Learning: Extended Abstract. In *Proc. of the 21st International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2022), Online, May 9–13, 2022*, IFAAMAS, 3 pages.

1 INTRODUCTION

Conventional multiagent reinforcement learning (MARL) methods suffer from low sample efficiency and poor local optimum in complex tasks. For example, in cooperative navigation [13, 14], a cooperative target selection policy that minimizes the overall arrival time is hard to be learned. Previous studies resort to exogenous expert demonstrations [5, 11, 17, 20, 22] or agents' mutual advising [2–4, 7, 19]. However, the expert demonstrations are usually unavailable in complex multiagent problems. Mutual advising between agents lacks global coordination at agent level.

Proc. of the 21st International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2022), P. Faliszewski, V. Mascardi, C. Pelachaud, M.E. Taylor (eds.), May 9–13, 2022, Online.

In this work, we propose learning to advise and learning from advice (LALA) to improve MARL. We focus on the cooperative tasks where agents need to avoid decision conflicts, such as target selection conflicts in cooperative navigation [12–14, 16, 18]. Each agent is deemed an advisee. A centralized advisor is learned to give advice to agents. We construct a space-time decision graph to characterize the relations between agents' decisions and propose a dual graph convolutional network (DualGCN) to learn smoothed and reconciled decisions as advice. To guide each agent to learn from the advice, we propose a policy-level generative adversarial network (PLGAN), where a discriminator distinguishes between the state-action sets characterizing the policies of the advisor and an agent. Each agent learns to follow the advice by optimizing a compound objective function consisting of an MARL objective and a regularization term reflecting closeness to the advised policy. Since the advised policy is reconciled and smoothed, it can improve MARL. Leveraging the discerning power of the discriminator, we propose a method to boost the advisor and further improve MARL.

2 METHOD

2.1 Learning to Advise with DualGCN

The advisor module is illustrated in the bottom right part of Figure 1. A space-time graph is used to characterize the relations between agents' decisions. Each agent at each time step corresponds to a vertex. Vertices of the same agent are connected (green edges) along the time dimension. Vertices of different agents at the same time step are connected with each other (red edges).

We propose DualGCN to fuse decision information and generate advice. The input features of a vertex is the decision made by an agent, denoted as π_v^{agent}. The output is the advised decision denoted as π_v. To cope with constraints imposed on agents' decisions for maintaining temporal continuity and resolving spatial conflicts, we divide the neighbors of each vertex by spatial domain and temporal domain as $\mathcal{N}_s(v)$ and $\mathcal{N}_t(v)$. For each neighbor set, a mean aggregator function similar to that used in GraphSAGE [9] is employed to aggregate features. The cost function for DualGCN is given as

$$\mathcal{L}_{Advisor} = \sum_{v\in\mathcal{V}} \Bigg[\sum_{u\in\mathcal{N}_t(v)} -\log(\sigma(\pi_v^T \pi_u)) - \sum_{w\in\mathcal{N}_s(v)} \log(\sigma(-\pi_v^T \pi_w)) + I_{label}(v)\cdot JSD(\pi_v || \pi_v^{\text{agent}}) \Bigg]. \tag{1}$$

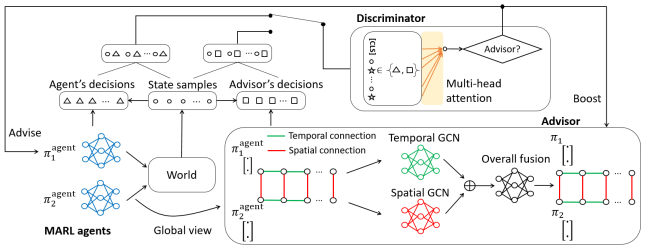
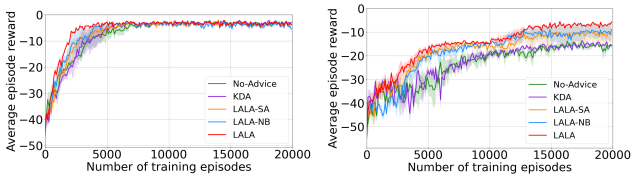


Figure 1: Illustration of LALA framework.



(a) N=7 (b) N=10

Figure 2: Convergence curves of average episode reward.

where σ is a sigmoid function. The first two terms are temporal discontinuity cost and spatial conflict cost, respectively. The last term is used to generate advice that can improve agents in a gradual manner, where $I_{label}(v)$ is an indicator function to express whether a decision made by an agent is the most confident among all agents.

2.2 Learning from Advice with PLGAN

To guide each agent to learn from advice, we propose PLGAN illustrated as the upper part of Figure 1. PLGAN includes an advisor-advisee discriminator that consists of a Transformer encoder [21] and a classifier. Similar to a previous work [15], the Transformer encoder is employed to learn representations of a policy with a set of state-action pairs as raw feature input. A class token (CLS) [6] is concatenated with the raw input. The learned embedding of the CLS token is deemed the representation of the policy and then mapped by the classifier to a judging probability.

Formally, let $\mathcal{B}_i^s$, $\mathcal{A}_i^\pi$, and $\mathcal{A}_i^G$ denote a set of states, actions taken by the agent's policy on these states, and the advised actions given byDualGCN, respectively. $\mathcal{D}^{\psi_i}$ denotes the discriminator parameterized with ψ_i. The loss function for the discriminator is

$$\mathcal{L}_{Disc,i} = -\mathbb{E}_{\mathcal{B}_i^s}\left[\log D^{\psi_i}(\mathcal{A}_i^G(\Omega), \mathcal{B}_i^s) + \log(1 - D^{\psi_i}(\mathcal{A}_i^\pi(\theta_i), \mathcal{B}_i^s))\right], \tag{2}$$

where θ_i denotes parameters of the policy of agent i, and Ω denotes parameters of DualGCN shared by all agents. Unlike typical GANs [1, 8] where there is ground truth data, the advice in LALA is learned as well in accordance with a global objective.

Each agent learns to match with the advised policy by maximizing the discriminator's loss, in addition to reward-based learning. Thus, the loss function for each agent is given by

$$\mathcal{L}_{Agent,i} = \mathcal{L}_\pi^{\theta_i} + \lambda \mathbb{E}_{\mathcal{B}_i^s} \log(1 - D^{\psi_i}(\mathcal{A}_i^\pi(\theta_i), \mathcal{B}_i^s)), \tag{3}$$

where $\mathcal{L}_\pi^{\theta_i}$ is a loss function for policy learning of an MARL algorithm. λ is a positive weight that balances agent's active learning and learning from advice.

2.3 Advisor Boosting via Advisor-Advisee Discriminator

By leveraging the discerning power of the discriminator, we propose a method to boost advisor's capability. The discriminator trained with (2) can give a higher probability score to the advisor than an agent. This probability score can thus be deemed a metric measuring the extent of advantage that the advisor's policy has over that of an agent. Therefore, we introduce an additional term into the original loss (1) for DualGCN, which reflects the advantage of the advisor's policy. Thus, the loss function for the advisor is given by

$$\mathcal{L}_{Advisor} = \sum_{v\in\mathcal{V}}\Bigg[\sum_{u\in N_t(v)} -\log(\sigma(\pi_v^T\pi_u)) - \sum_{w\in N_s(v)}\log(\sigma(-\pi_v^T\pi_w)) + I_{label}(v)\cdot JSD(\pi_v||\pi_v^{\text{agent}})\Bigg] - \mu\sum_{i=1}^{N}\mathbb{E}_{\mathcal{B}_i^v,\mathcal{B}_i^s}\left[\log D^{\psi_i}(\{\pi_v\}_{v\in\mathcal{B}_i^v}, \mathcal{B}_i^s)\right], \tag{4}$$

where μ is a positive weight; $\mathcal{B}_i^s$ denotes a set of states obeying the same sampling distribution as that used to train the discriminator; $\mathcal{B}_i^v$ denotes the vertex index set corresponding to $\mathcal{B}_i^s$.

Table 1: Cooperation success rate of different methods

	LALA	LALA-NB	LALA-SA	KDA	No-Advice
N=7	**0.930**	0.890	0.888	0.913	0.874
N=10	**0.757**	0.520	0.449	0.000	0.015

3 EXPERIMENTS

We evaluate LALA in a multi-agent cooperative navigation task with the same settings used in [13, 14]. In this task, N agents need to cooperate to reach the same number of targets using the minimum time. At each timestep, each agent selects a target and moves a step toward it. Potential decision conflicts exist in their target selections.

We implement LALA based on an MARL baseline [13] (No-Advice). We compare LALA with it and three advising approaches, i.e. knowledge distillation [10] (KDA), LALA with single state-action pair based discriminators [20] (LALA-SA), and LALA with no advisor boost (LALA-NB). Figure 2 shows the convergence curves of average episode reward. As can be seen from the results, LALA outperforms other methods consistently in terms of convergence speed and average reward. Table 1 shows the cooperation success rate. LALA achieves the highest success rate and performs the most stably among all methods. The result corresponding to N=10 indicates the significant superiority of LALA over other methods.

4 CONCLUSION AND FUTURE WORK

We propose LALA approach to improve MARL, which involves three-part learning, i.e. DualGCN advisor, MARL agents, and advisor-agent discriminators. Experimental results demonstrate the superiority of LALA in terms of both convergence speed and cooperation performance. For future works, introducing a more powerful GCN and GAN into LALA would be worthy of investigating.

ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China under Grant U20B2060.

REFERENCES

[1] Antonia Creswell, Tom White, Vincent Dumoulin, Kai Arulkumaran, Biswa Sengupta, and Anil A Bharath. 2018. Generative adversarial networks: An overview. *IEEE Signal Processing Magazine* 35, 1 (2018), 53–65.

[2] Felipe Leno Da Silva and Anna Helena Reali Costa. 2019. A survey on transfer learning for multiagent reinforcement learning systems. *Journal of Artificial Intelligence Research* 64 (2019), 645–703.

[3] Felipe Leno Da Silva, Ruben Glatt, and Anna Helena Reali Costa. 2017. Simultaneously learning and advising in multiagent reinforcement learning. In *Proceedings of the 16th International Conference on Autonomous Agents and MultiAgent Systems*. 1100–1108.

[4] Felipe Leno Da Silva, Matthew E Taylor, and Anna Helena Reali Costa. 2018. Autonomously Reusing Knowledge in Multiagent Reinforcement Learning. In *IJCAI*. 5487–5493.

[5] Mehul Damani, Zhiyao Luo, Emerson Wenzel, and Guillaume Sartoretti. 2021. PRIMAL _2: Pathfinding Via Reinforcement and Imitation Multi-Agent Learning-Lifelong. *IEEE Robotics and Automation Letters* 6, 2 (2021), 2666–2673.

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. 4171–4186.

[7] Zijian Gao, Kele Xu, Bo Ding, Huaimin Wang, Yiying Li, and Hongda Jia. 2021. KnowSR: Knowledge Sharing among Homogeneous Agents in Multi-agent Reinforcement Learning. *arXiv preprint arXiv:2105.11611* (2021).

[8] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2020. Generative adversarial networks. *Commun. ACM* 63, 11 (2020), 139–144.

[9] William L Hamilton, Rex Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*. 1025–1035.

[10] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531* (2015).

[11] Wonseok Jeon, Paul Barde, Derek Nowrouzezahrai, and Joelle Pineau. 2020. Scalable and sample-efficient multi-agent imitation learning. In *Proceedings of the Workshop on Artificial Intelligence Safety, co-located with 34th AAAI Conference on Artificial Intelligence, SafeAI@ AAAI*.

[12] Yue Jin, Shuangqing Wei, Jian Yuan, and Xudong Zhang. 2021. Hierarchical and Stable Multiagent Reinforcement Learning for Cooperative Navigation Control. *IEEE Transactions on Neural Networks and Learning Systems* (2021).

[13] Yue Jin, Shuangqing Wei, Jian Yuan, and Xudong Zhang. 2021. Information-Bottleneck-Based Behavior Representation Learning for Multi-agent Reinforcement Learning. In *Proceedings of IEEE International Conference on Autonomous Systems*.

[14] Yue Jin, Shuangqing Wei, Jian Yuan, Xudong Zhang, and Chao Wang. 2020. Stabilizing Multi-Agent Deep Reinforcement Learning by Implicitly Estimating Other Agents' Behaviors. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*. 3547–3551.

[15] Yue Jin, Yue Zhang, Tao Qin, Xudong Zhang, Jian Yuan, Houqiang Li, and Tie-Yan Liu. 2021. Supervised Off-Policy Ranking. *arXiv preprint arXiv:2107.01360* (2021).

[16] Yue Jin, Yaodong Zhang, Jian Yuan, and Xudong Zhang. 2019. Efficient Multi-agent Cooperative Navigation in Unknown Environments with Interlaced Deep Reinforcement Learning. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*. 2897–2901.

[17] Hyun-Rok Lee and Taesik Lee. 2019. Improved cooperative multi-agent reinforcement learning algorithm augmented by mixing demonstrations from centralized policy. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*. 1089–1098.

[18] Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, OpenAI Pieter Abbeel, and Igor Mordatch. 2017. Multi-agent actor-critic for mixed cooperative-competitive environments. In *Advances in Neural Information Processing Systems*. 6379–6390.

[19] Shayegan Omidshafiei, Dong-Ki Kim, Miao Liu, Gerald Tesauro, Matthew Riemer, Christopher Amato, Murray Campbell, and Jonathan P How. 2019. Learning to teach in cooperative multiagent reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 6128–6136.

[20] Jiaming Song, Hongyu Ren, Dorsa Sadigh, and Stefano Ermon. 2018. Multi-agent generative adversarial imitation learning. *arXiv preprint arXiv:1807.09936* (2018).

[21] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *arXiv preprint arXiv:1706.03762* (2017).

[22] Lantao Yu, Jiaming Song, and Stefano Ermon. 2019. Multi-agent adversarial inverse reinforcement learning. In *International Conference on Machine Learning*. PMLR, 7194–7201.