Learning Individual Difference Rewards in Multi-Agent Reinforcement Learning

Extended Abstract

Chen Yang
School of Artificial Intelligence, University of Chinese Academy of Sciences
Institute of Automation, Chinese Academy of Sciences
Beijing, China
yangchen2021@ia.ac.cn

Guangkai Yang
School of Artificial Intelligence, University of Chinese Academy of Sciences
Institute of Automation, Chinese Academy of Sciences
Beijing, China
yangguangkai2019@ia.ac.cn

Junge Zhang*
School of Artificial Intelligence, University of Chinese Academy of Sciences
Institute of Automation, Chinese Academy of Sciences
Beijing, China
jgzhang@nlpr.ia.ac.cn

ABSTRACT

We investigate explicit solutions to multi-agent credit assignment problem. Specifically, we assign each agent individual difference rewards in addition to the team reward as to distinguish the contribution of different agents to the team. We present a novel reward decomposition network to estimate the influence of each agent's action on the team reward, and distribute difference rewards accordingly. Furthermore, we combine difference rewards with actor-critic framework and propose a new approach called *learning individual difference rewards* (LIDR). We evaluate LIDR on a set of StarCraft II micromanagement problems. Results show that LIDR significantly outperforms previous state-of-the-art methods.

KEYWORDS

Multi-Agent Systems; Credit Assignment; Reward Shaping

ACM Reference Format:
Chen Yang, Guangkai Yang, and Junge Zhang. 2023. Learning Individual Difference Rewards in Multi-Agent Reinforcement Learning: Extended Abstract. In *Proc. of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2023), London, United Kingdom, May 29 – June 2, 2023*, IFAAMAS, 3 pages.

1 INTRODUCTION

A great challenge for multi-agent reinforcement learning (MARL) is credit assignment [2], which focuses on attributing each agent's contribution to the team according to its behavior. If the credit assignment is not well handled, it can cause the lazy agent [9] issue and lead to low sample efficiency in practice. Unfortunately, in most MARL scenarios, all agents share a team reward, from which it is difficult to deduce each agent's contribution to the team. A common solution to credit assignment is reward shaping [6, 7], which differentiates each agent's credit by introducing extra rewards to agents. However, it generally requires prior knowledge on the environment and human labor to assign precise reward to individual agent, which is impractical in many MARL problems.

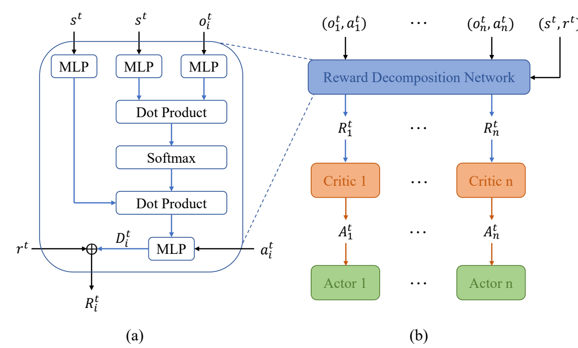

Figure 1: (a) The reward decomposition network structure. (b) The overall framework of LIDR.

In this paper, we propose a new MARL method called *learning individual difference rewards* (LIDR) to address the above issues. LIDR takes an approach that combines actor-critic [5] framework with difference rewards [1], and explicitly assigns credits by distributing individual difference rewards to each agent. Specifically, the critic is trained with the individual reward consisting of a global team reward and a local difference reward. The team reward is shared among agents, while the difference rewards vary among agents as to differentiate each agent's contribution to the team. To estimate difference rewards, we present a reward decomposition network to capture the influence of each agent's actions on the team reward and distribute individual rewards accordingly. As a result, LIDR is able to efficiently compute difference rewards without prior knowledge on the environment model, and the whole training procedure is conducted in a model-free manner.

2 METHOD

we propose the reward decomposition network to decompose the team reward into individual rewards, which are further assigned to agents. The individual reward for each agent i is formulated as $R_i = G + L_i$, where global reward G represents the feedback to the achievement under agents' cooperation, encouraging each agent to work with others, while local reward L_i represents the feedback to individual performance, adjusting credits to each agent according

*corresponding author

Proc. of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2023), A. Ricci, W. Yeoh, N. Agmon, B. An (eds.), May 29 – June 2, 2023, London, United Kingdom.

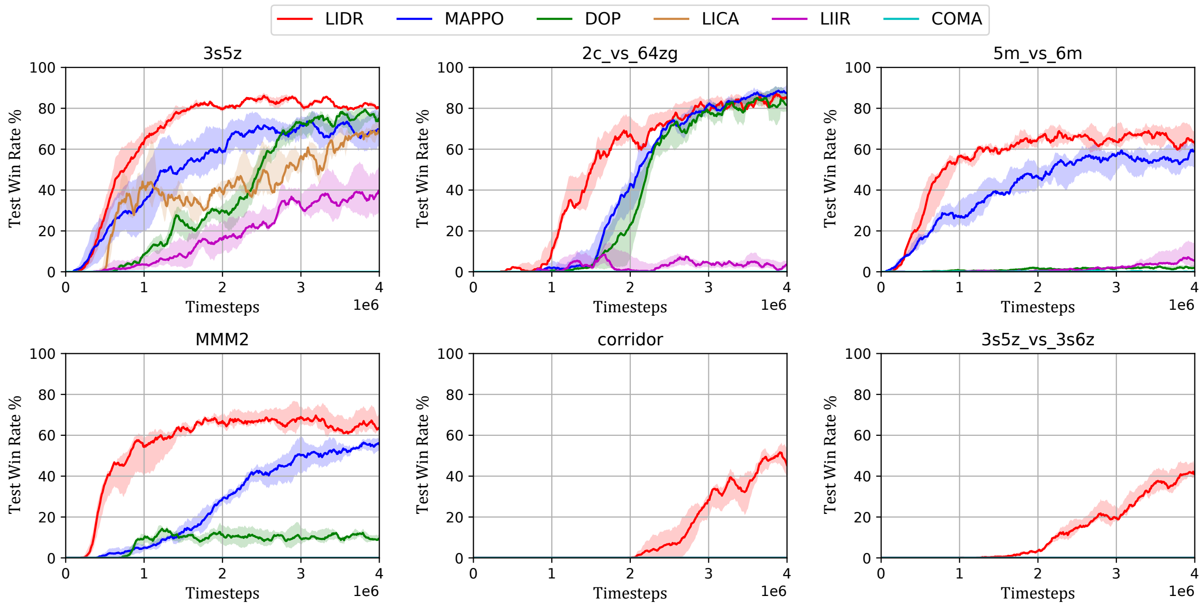

Figure 2: Test win rates of LIDR and baseline methods on SMAC.

to its behavior in the team. In practice, it is natural to choose the team reward r as G, and we adopt the difference reward D_i as L_i.

The structure of reward decomposition network is illustrated in Fig. 1 (a). Specifically, the network utilizes attention mechanism [10] for information integration. It takes global state s and local observation o_i as input, and outputs the distribution of estimated team reward over available actions of agent i: $\bar{R}(s, o_i, a)$.

To train the reward decomposition network, we minimize the following mean squared error (MSE) loss:

$$L(\eta) = \sum_{t=1}^{T}\sum_{i=1}^{n}(\bar{R}(s^t, o_i^t, a_i^t; \eta) - r^t)^2, \tag{1}$$

where η are the network parameters, r^t is the environmental reward at timestep t, and a_i^t is the action taken by agent i at timestep t. Ideally, the outputs of the network converge to

$$\bar{R}(s, o_i, a) = \mathbb{E}_{a_i=a, \boldsymbol{a}_{-i}\sim\boldsymbol{\pi}_{-i}}[r(s, \boldsymbol{a})]. \tag{2}$$

With the reward decomposition network, we can further compute individual difference rewards for agent i as

$$D_i = \bar{R}(s, o_i, a_i) - max_a\bar{R}(s, o_i, a), \tag{3}$$

where the former term is the estimated reward under action a_i, which is actually taken by agent i, and the latter term is the maximum estimated reward that could have been reached by changing agent i's policy. Then we utilize R_i to supervise the actor-critic learning process for each agent, as presented in Fig. 1 (b).

3 RESULTS

We evaluate LIDR on several micromanagement tasks from the SMAC [8] benchmark, where a group of decentralized agents controlled by MARL algorithms need to defeat another group of agents controlled by StarCraft II built-in AI. We elaborately select 5 baseline methods, which are: COMA [4], LIIR [3], LICA [13], MAPPO [12], and DOP [11]. The training configurations of these methods are set to the same for fair comparison.

The results in 6 different maps from SMAC are shown in Fig. 2. We observe that LIDR outperforms all baseline methods in hard maps (3s5z, 2c_vs_64zg, 5m_vs_6m), and the advantage of our method becomes more significant in super-hard maps (MMM2, corridor, 3s5z_vs_3s6z), especially in corridor and 3s5z_vs_3s6z, where all baseline methods fail to solve the tasks, while LIDR can achieve 40% win rate at the end of training. These results indicate that LIDR has more capacity in addressing credit assignment problem. In complex environments, it is very important for agents to have sufficient exploration to find the solution, and MARL algorithms that cannot well handle the credit assignment would fail. Specifically, if the MARL methods cannot distinguish between agents that conduct potential rewarding actions and agents that conduct uncooperative actions, and assign different credits to them, it could prevent agents from efficient exploration and eventually stuck in a local optimum. We attribute the performance of LIDR to individual difference rewards, which help differentiate credits among agents and diversify agents' behaviors for better exploration.

4 CONCLUSION

We present LIDR, a MARL method that aims to explicitly address credit assignment with difference rewards. Different from previous model-based approaches, LIDR utilizes a novel reward decomposition network to efficiently estimate difference rewards in a model-free way. Experiment results on SMAC benchmark empirically demonstrate the high sample efficiency and improved robustness of our proposed method.

ACKNOWLEDGMENTS

This work is supported in part by Basic Cultivation Fund project, CAS (JCPYJJ-22017), the National Natural Science Foundation of China (No.61876181),the Projects of Chinese Academy of Science (Grant No. QYZDB-SSW-JSC006), and the Youth Innovation Promotion Association CAS.

REFERENCES

[1] Adrian K Agogino and Kagan Tumer. 2004. Unifying temporal and structural credit assignment problems. In *Autonomous Agents and Multi-Agent Systems Conference*.

[2] Yu-Han Chang, Tracey Ho, and Leslie Kaelbling. 2003. All learning is local: Multi-agent learning in global reward games. *Advances in neural information processing systems* 16 (2003).

[3] Yali Du, Lei Han, Meng Fang, Ji Liu, Tianhong Dai, and Dacheng Tao. 2019. Liir: Learning individual intrinsic reward in multi-agent reinforcement learning. *Advances in Neural Information Processing Systems* 32 (2019).

[4] Jakob Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. 2018. Counterfactual multi-agent policy gradients. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 32.

[5] Vijay Konda and John Tsitsiklis. 1999. Actor-critic algorithms. *Advances in neural information processing systems* 12 (1999).

[6] Andrew Y Ng, Daishi Harada, and Stuart Russell. 1999. Policy invariance under reward transformations: Theory and application to reward shaping. In *Icml*, Vol. 99. 278–287.

[7] Jette Randløv and Preben Alstrøm. 1998. Learning to Drive a Bicycle Using Reinforcement Learning and Shaping.. In *ICML*, Vol. 98. Citeseer, 463–471.

[8] 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).

[9] Peter Sunehag, Guy Lever, Audrunas Gruslys, Wojciech Marian Czarnecki, Vinicius Zambaldi, Max Jaderberg, Marc Lanctot, Nicolas Sonnerat, Joel Z Leibo, Karl Tuyls, et al. 2017. Value-decomposition networks for cooperative multi-agent learning. *arXiv preprint arXiv:1706.05296* (2017).

[10] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).

[11] Yihan Wang, Beining Han, Tonghan Wang, Heng Dong, and Chongjie Zhang. 2020. Dop: Off-policy multi-agent decomposed policy gradients. In *International Conference on Learning Representations*.

[12] Chao Yu, Akash Velu, Eugene Vinitsky, Yu Wang, Alexandre Bayen, and Yi Wu. 2021. The surprising effectiveness of ppo in cooperative, multi-agent games. *arXiv preprint arXiv:2103.01955* (2021).

[13] Meng Zhou, Ziyu Liu, Pengwei Sui, Yixuan Li, and Yuk Ying Chung. 2020. Learning implicit credit assignment for cooperative multi-agent reinforcement learning. *Advances in Neural Information Processing Systems* 33 (2020), 11853–11864.