Comparison of Multi-Agent Reinforcement Learning Algorithms in Cooperative Grid Environments

Hyun-Woo Park° and Sang-Ki Ko

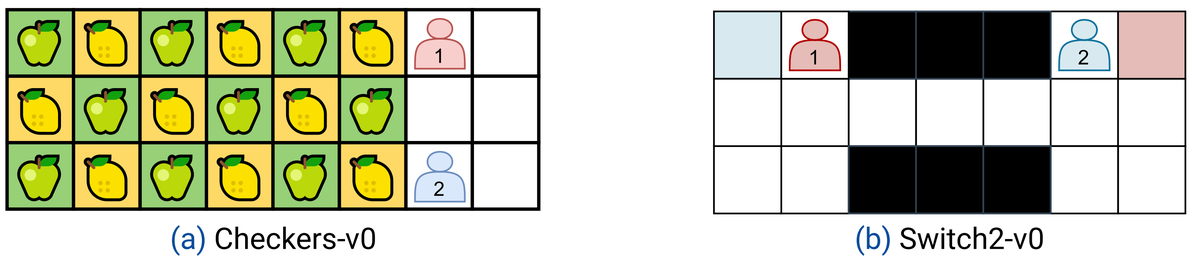
Computational Intelligence & Data Analytics Laboratory, Department of AI, University of Seoul

UNIVERSITY OF SEOUL

Abstract

MAPPO and IPPO could achieve competitive performance to existing off-policy MARL algorithms [1]. However, it remains to be studied whether on-policy algorithms can deal with their intrinsic limitations, such as the lack of exploration. So, we compare the MARL algorithms in highly cooperative grid environments to see whether they can learn how to cooperate effectively to achieve the optimal reward compared to VDN [2] and QMIX [3]

Environments



(a) Checkers-v0 (b) Switch2-v0

- Checkers-v0 is a game where an agent's reward depends on the fruit it eats, requiring some agents to accept negative rewards to maximize cumulative rewards for team (Figure-1a).
- The goal in Switch is for each agent to reach a location of the same color. The strategy changes depending on which agent enters the narrow tunnel first, in order to maximize the reward (Figure-1b).

Experiment Result

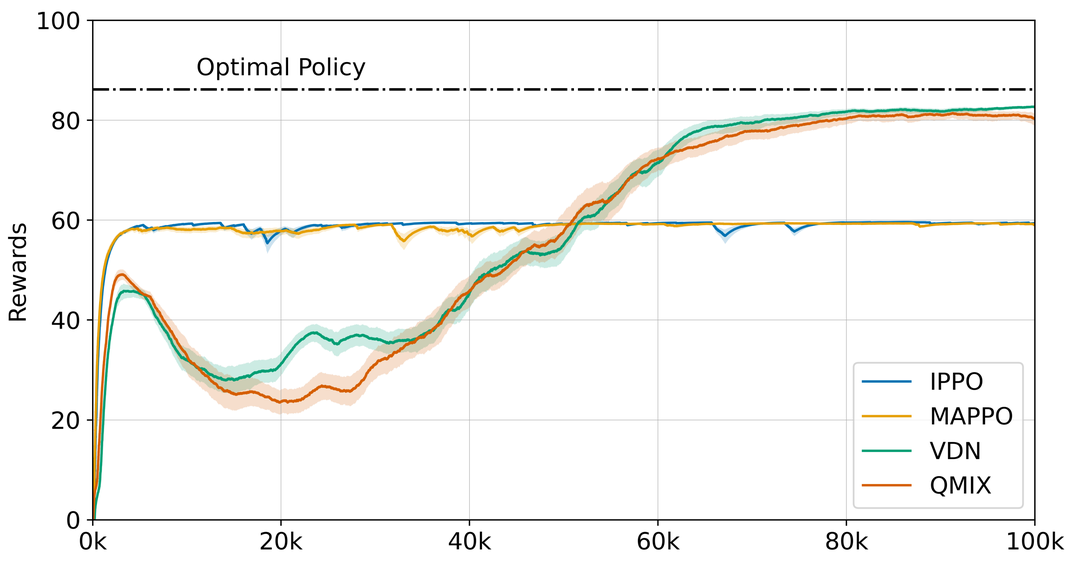


Figure 2. Comparison of on-policy (MAPPO and IPPO) and off-policy (QMIX and VDN) algorithms on Checkers with standard error.

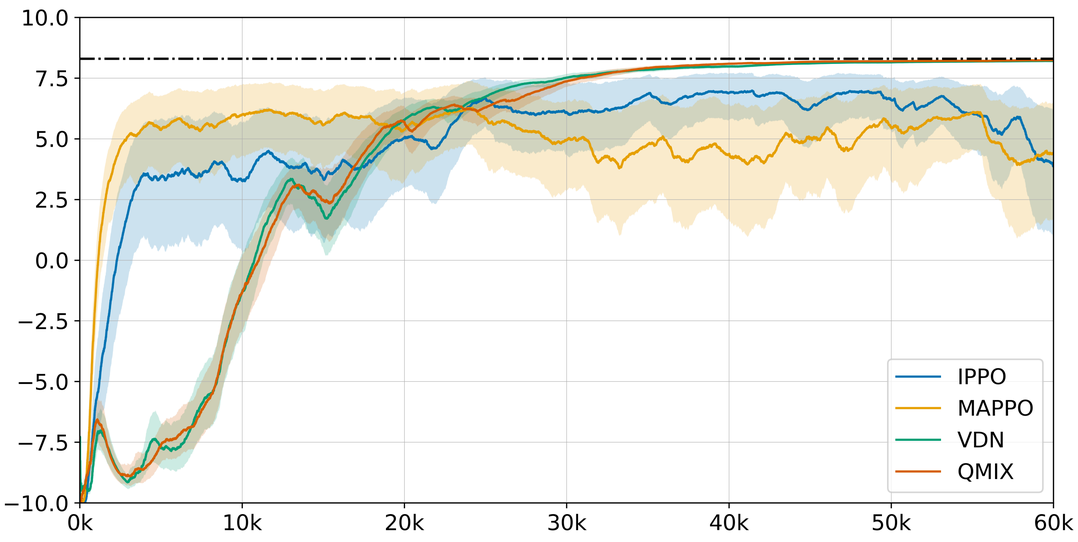


Figure 3. Comparison of on-policy (MAPPO and IPPO) and off-policy (QMIX and VDN) algorithms on Switch2 with standard error.

Analysis

Lazy Agent Problem in PPO

The results in Figure-2 show IPPO and MAPPO converge rapidly without performing enough exploration and do not find a way to exploit insensitive agent. On the other hand, VDN and QMIX find ways to exploit insensitive agents through sufficient exploration and attain the optimal policy.

Policy Oscillation in Switch2

In Figure-3two optimal policies make learning unstable for IPPO and MAPPO, which choose actions stochastically, as the optimal policy they learn constantly becomes different. On the other hand, VDN and QMIX learn deterministic policies as the ϵ decreases, so they learn a single optimal policy stably in Switch2.

Improvement

In order for PPOs to be able to utilize insensitive agents, the model must be able to design the rewards that agents receive through exploration. To do this, the authors propose storing the maximum and minimum reward received in each state and then shaping the reward delivered.

Table 1. Hyperparameter settings of MAPPO and Reward-Map MAPPO

Hyperparameters	MAPPO	Reward-Map MAPPO
Use Reward-Map	✗	✓
PPO epoch	1	1

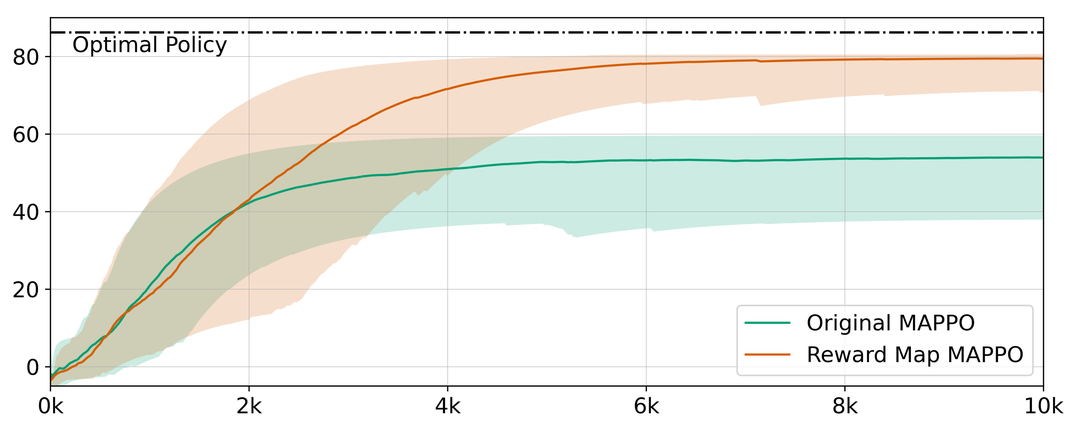


Figure 4. Comparison of MAPPO and Reward-Map MAPPO algorithms on Checkers with max/min values.

Reward Shaping with Reward Maps

Figure-4 shows that Reward-Map MAPPO, unlike traditional MAPPO, can utilize insensitive agents to learn optimal policies. This means that the model performs reward shaping correctly by utilizing Reward- Map.

Conclusion

IPPO and MAPPO struggle with high-level cooperative policies due to weak exploration and policy oscillation. However, Reward-Map MAPPO showed that using reward maps to shape rewards can enhance MAPPO's exploration and policy learning. Our study will focus on reward shaping using the state's predicted reward through clustering to generate reward maps in complex environments.

References

[1] C. Yu, A. Velu, E. Vinitsky, *et al.*, "The surprising effectiveness of PPO in cooperative multi-agent games," in *NeurIPS 2022*, 2022.

[2] P. Sunehag, G. Lever, A. Gruslys, *et al.*, "Value-decomposition networks for cooperative multi-agent learning," *CoRR*, vol. abs/1706.05296, 2017.

[3] T. Rashid, M. Samvelyan, C. S. de Witt, G. Farquhar, J. N. Foerster, and S. Whiteson, "QMIX: monotonic value function factorisation for deep multi-agent reinforcement learning," in *ICML 2018*, 2018.