Multi-Agent Cooperation via Graph Convolutional Reinforcement Learning

Chaemoon Im, Jaeyoung Choe and Joongheon Kim
Department of Electrical and Computer Engineering, Korea University, Seoul, Republic of Korea
E-mails: {anscodla0314, cielblue0522, joongheon}@korea.ac.kr

***Abstract*—Among many factors in group-level decision making, communication between agents are significant to learn cooperative behavior, because it determines the tendency of actions of the agents in the group. Unlike conventional multi-agent reinforcement learning (MARL) algorithms which assumes ideal, static communication environment, this paper proposes Graph Convolutional Reinforcement Learning (GCRL), which utilizes Graph Convolutional Neural Network to analyze the communication status between the agents. Experiment results shows the potential of the proposed GCRL, in terms of maximized reward.**

***Index Terms*—Multi-agent Reinforcement Learning, Graph Convolutional Neural Network,**

I. Introduction

Group-level decision making has been a topic of continuous interests, but still remains as a challenging task in many field such as platooning in autonomous driving, multi-robot cooperation and so on [1], [2]. This is because normally in group, there is a limitation for obtaining information of the others, as described in Fig. 1. To handle this problem, multi-agent reinforcement learning (MARL) is proposed [3]. In particular, centralized-training-decentralized-execution structure (CTDE structure) enables efficient gathering of information and training, significantly increasing the performance of the group-level decision making. However, many MARL algorithms often rely on optimistic viewpoint such that communication and sharing information between the agents are always possible [4]. Considering that communication between the agents is affected by many factors such as distance, channel status and delay, studies on realistic multi-agent control should be proceeded without such strong assumption. Specifically, realistic MARL algorithm should consider the fact that information sharing between the agents can only be done partially, within the communication range.

Based on the fact that communication between the agents can be mathematically expressed as graph, graph neural network (GNN) has been integrated with classical reinforcement learning (RL) algorithms to increase RL's applicability in real-world settings with communication constraints. In particular, graph convolutional reinforcement learning (GCRL) is the first algorithm that adopts GNN to utilize communication information of the agents [5]. This paper examines the performance of the GCRL algorithm with other baselines in autonomous driving environment. By experiments in the situation where information sharing between the objects are limited, this paper analyzes GCRL algorithm's potential and characteristics.

Corresponding Author: Joongheon Kim
This work was supported by the IITP(Institute of Information & Communications Technology Planning & Evaluation)-ITRC(Information Technology Research Center) grant funded by the Korea government (Ministry of Science and ICT) (IITP-2025-RS-2024-00436887) and also by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT)(RS-2025-00561377).

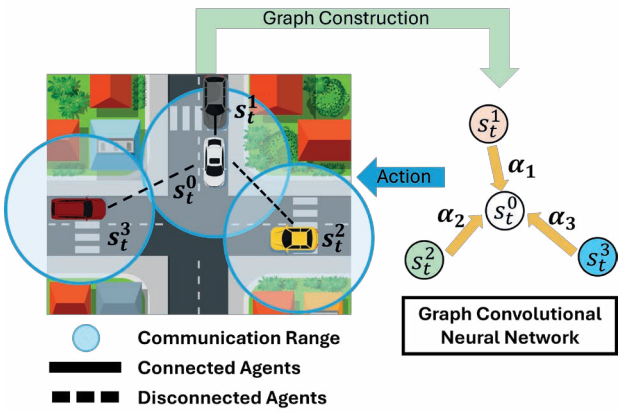

Fig. 1: A schematic illustration GCRL. In realistic scenario, agents can share information with only the ones within the communication range. GCRL captures this characteristic using GNNs.

II. Preliminaries

A. Graph and Graph Convolutional Neural Network (GCN)

A graph $\mathcal{G} = \langle V, E \rangle$ is composed of vertices V and edges E. Each i-th node in the graph has its node value n_i. The main difference between graph and other type of data is that it describes the relationship between the nodes using edge e_{ij}. An adjoint matrix $A_{ij} = e_{ij}$, is another expression for the graph.

GCN is a special type of neural network, which is designed to process graph-type data [6]. GCN generalizes the notion of the convolution, which has been utilized in many machine learning fields after the advent of convolutional neural network (CNN). Consider a matrix H whose row vectors are the node values of the graph G. Graph convoltution over a graph $\mathcal{G}$ with its adjoint matrix A is defined as follows,

$$H' = \sigma(AWH), \tag{1}$$

where σ and W are activation function and matrix of trainable parameters, respectively. Graph attention network (GAT) is

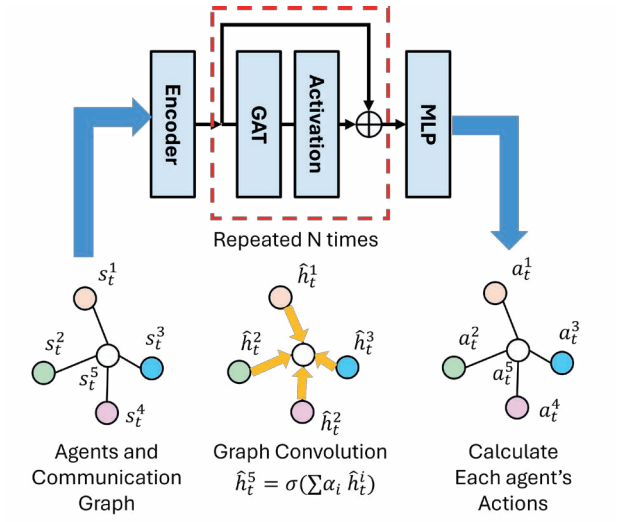

Fig. 2: An schematic illustration of GCRL.

another type of GCN, which combines information of the neighbor nodes with weights [7]. Graph convolution in GAT is defined as follows,

$$H' = \sigma(\sum \alpha_i W H_i). \tag{2}$$

where H_i denotes the i-th column vector of the H. By adopting another trainable parameters α_i, GAT allows having different importance to the neighboring nodes. This paper selects GAT to enable GCRL, which can effectively model the dynamics between the agents, where the importance of the information can vary across the agents.

B. Related Work

Before learning-based control, decision-making systems among groups mainly used rule-based and optimization-based approaches [8]. Rule-based approaches, such as finite state machines and behavior trees, provided efficient methods to interpret agent behavior [9]. On the other hand, optimization-based approaches, such as Distributed Constraint Optimization Problems (DCOP), offer the advantage of formalizing interactions between agents into mathematical models, providing mathematically guaranteed optimal solutions [10]. Although both of these methods provide an efficient interpretation of agents in structured environments, they also suffer from instability in dynamic and uncertain environments [11].

To address this issue, learning-based methods emerged, which allow agents to make decisions independently through interaction with the environment. In this way, learning-based methods have the advantage of demonstrating superior performance even in complex environments [12]. In particular, MARL is a prominent example of these learning-based control. Unlike existing methods, it demonstrates superior performance even in environments with an increasing number of agents or increased complexity.

However, these methods normally utilize interpretable, crafted data, such as images or grid-based states [13]. In many cases, real-world data is graph-based, making it difficult to process using conventional neural network architecture [14]. Special types of neural networks such as GCNs and GAT are utilized to process graph-structured data. Graph-based learning methods not only have the advantage of being able to learn graph-shaped data similar to real-world data, but also structurally respond to increases or decreases in the number of agents [15].

III. Algorithm Details

A. System Modeling

This paper utilizes MAgym's `TrafficJunction10-v0` environment, where 10 cars are cooperating with each others to reach their destination without collisions. Each car can only observe 3×3 tiles surrounding it. When car is noticed in the range of the observation, each car's id, current location in 2D and the destination are obtained.

B. Graph Construction

This paper constructs a graph of the agents, using the state information. k-th node represents k-th car and node value of that node is state of the k-th car. Because each k-th car can only observe 3×3 tiles surrounding it, the cars within the range are connected to the k-th car. Its edge value is set to be 1. Otherwise, for cars that not in the observation range, edge value is set to be 0, meaning they are disconnected to the k-th car. Node value of k-th car is then a vector contains the information about each car's id, current location in 2D and the destination.

C. Neural Network Architecture

As illustrated in Fig. 2, this paper utilizes structure similar to CommNet [4]. Firstly, each agent's state informations $s_t^1, \cdots, s_t^N$ are encoded via encoder, changing them to latent variables $h_t^1, \cdots, h_t^N$. After that, GCN performs graph convolution to those latent variables, yielding $\hat{h}_t^1, \cdots, \hat{h}_t^N$. By this procedure, adjacent agents are sharing their information. Finally, multi-layer perceptron and softmax activation function are applied to the latent variables, resulting actions $a_t^1, \cdots, a_t^N$.

D. Training

Training is done by similar manner with DQN. The overall network is trained by the information gathered from the clients. Here, current graph structure $\mathcal{G}_t$ and graph structure of next timestep $\mathcal{G}_{t+1}$ are also gathered. Then, Q-value considering the graph structure $Q(s_t, a_t|\mathcal{G}_t)$ is updated using Bellman equation. Using this Q-value, each agent decides which action to perform.

IV. Performance Evaluation

A. Experiment Setup

For the comparison, this paper selects representative MARL algorithms MADDPG and CommNet [16]. For both MADDPG and CommNet, this paper selects actor-critic structure.

TABLE I: A comparison between the graph-based MARL algorithms with conventional RL algorithms.

	MADDPG	CommNet	GCRL
Results	-130.7	-83.1	-22.3

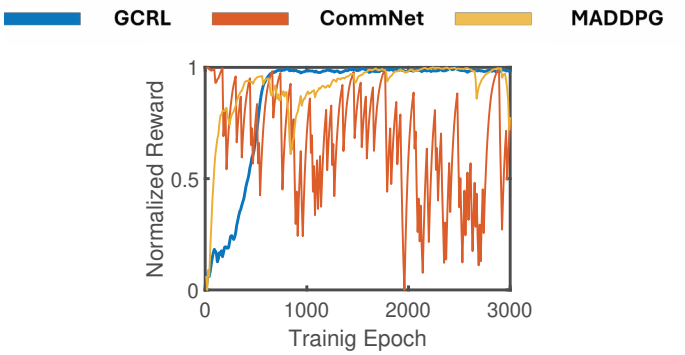

Fig. 3: Normalized rewards of the graph-based RL and conventional MARL algorithms.

B. Experiment Results

Performance Analysis. As described in Fig. 3 and Table I, the graph-based RL algorithms shows higher performance and convergence compared to other MARL algorithms, even when the information sharing between the agents is limited. In particular, the proposed GCRL achieves 82.9% and 72.3% higher performance compared to MADDPG and CommNet, respectively. This shows that the proposed graph-based RL algorithm well overcomes the limitations of partial observability. In addition, failure of the CommNet algorithm implies that overall information of the agent can be redundant in decision-making.

Effect of the GNN layer. In addition, as described in Fig. 4, as the number of the GNN layer increases, the performance of the proposed GCRL decreases. This is because due to the characteristic of GCN, N-layer GNN implies that the agent can utilizes other agent's information using N-hop transmission. This result also implies that other agent's information can be redundant in decision-making.

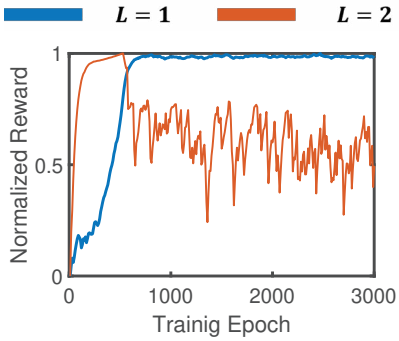

Fig. 4: Normalized rewards of the proposed algorithm varying the number of GNN layers.

V. Concluding Remarks

To enable MARL in real-world settings, limitations in communication between the agents must be considered. This paper examines GCRL, which adopts GCN to model the communication status and information sharing between the agents. Experiments verifies that the proposed GCRL can achieve multi-agent cooperation under the communication constraint, realizing MARL in realistic communication settings.

References

[1] Y. Khazaeni and C. G. Cassandras, "Event-driven trajectory optimization for data harvesting in multiagent systems," *IEEE Transactions on Control of Network Systems*, vol. 5, no. 3, pp. 1335–1348, Sept. 2018.

[2] L. An, G.-H. Yang, and S. Wasly, "Obstacle avoidance in distributed optimal coordination of multirobot systems: A trajectory planning and tracking strategy," *IEEE Transactions on Control of Network Systems*, vol. 11, no. 3, pp. 1335–1344, Sept. 2024.

[3] X. Xing, Z. Zhou, Y. Li, B. Xiao, and Y. Xun, "Multi-UAV adaptive cooperative formation trajectory planning based on an improved MATD3 algorithm of deep reinforcement learning," *IEEE Transactions on Vehicular Technology*, vol. 73, no. 9, pp. 12 484–12 499, Sept. 2024.

[4] S. Sukhbaatar, A. Szlam, and R. Fergus, "Learning multiagent communication with backpropagation," in *Proc. Advances in Neural Information Processing Systems*, D. D. Lee, M. Sugiyama, U. von Luxburg, I. Guyon, and R. Garnett, Eds., Barcelona, Spain, Dec. 2016, pp. 2244–2252.

[5] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proc. International Conference on Learning Representations (ICLR)*, Toulon, France, Apr. 2017.

[6] ——, "Semi-supervised classification with graph convolutional networks," in *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. [Online]. Available: https://openreview.net/forum?id=SJU4ayYgl

[7] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," in *Proc. International Conference on Learning Representations (ICLR)*, Vancouver, Canada, Apr. 2018.

[8] J. Broersen, M. Dastani, J. Hulstijn, Z. Huang, and L. van der Torre, "The boid architecture: Conflicts between beliefs, obligations, intentions and desires," in *Proc. International Conference on Autonomous Agents (AGENTS)*, Montreal, Canada, May. 2001, pp. 9–16.

[9] H. Wang, S. Kwong, Y. Jin, W. Wei, and K. Man, "Agent-based evolutionary approach for interpretable rule-based knowledge extraction," *IEEE Trans. Syst. Man Cybern. Part C*, vol. 35, no. 2, pp. 143–155, May. 2005.

[10] M. Yokoo, E. H. Durfee, T. Ishida, and K. Kuwabara, "The distributed constraint satisfaction problem: Formalization and algorithms," *IEEE Transactions on Knowledge and Data Engineering*, vol. 10, no. 5, pp. 673–685, Sep. 1998.

[11] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proc. IEEE*, vol. 95, no. 1, pp. 215–233, Jan. 2007.

[12] T. Rashid, M. Samvelyan, C. Schroeder de Witt, G. Farquhar, J. Foerster, and S. Whiteson, "Qmix: Monotonic value function factorisation for deep multi-agent reinforcement learning," in *Proc. International Conference on Machine Learning (ICML)*, Stockholm, Sweden, Jul. 2018, pp. 4292–4301.

[13] G. Papadopoulos, A. Kontogiannis, F. Papadopoulou, C. Poulianou, I. Koumentis, and G. A. Vouros, "An extended benchmarking of multi-agent reinforcement learning algorithms in complex fully cooperative tasks," in *Proc. International Conference on Autonomous Agents and Multiagent Systems*, Detroit, USA, Feb. 2025, pp. 1613–1622.

[14] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu, "A comprehensive survey on graph neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 1, pp. 4–24, Jan. 2021.

[15] M. Goarin and G. Loianno, "Graph neural network for decentralized multi-robot goal assignment," *IEEE Robotics Autom. Lett.*, vol. 9, no. 5, pp. 4051–4058, May. 2024.

[16] R. Lowe, Y. Wu, A. Tamar, J. Harb, P. Abbeel, and I. Mordatch, "Multi-agent actor-critic for mixed cooperative-competitive environments," in *Proc. International Conference on Neural Information Processing Systems (NeurIPS)*, Long Beach, CA, USA, Dec. 2017, pp. 6379–6390.