Graph Neural Networks: Learning Representations of Robot Team Coordination Problems

Matthew Gombolay
Georgia Institute of Technology
Atlanta, GA, USA
matthew.gombolay@cc.gatech.edu

Zheyuan Wang
Georgia Institute of Technology
Atlanta, GA, USA
pjohnwang@gatech.edu

KEYWORDS

Graph Neural Networks, Multi-Agent Systems, Multi-Robot Coordination, Dynamic Scheduling

ACM Reference Format:

Matthew Gombolay and Zheyuan Wang. 2022. Graph Neural Networks: Learning Representations of Robot Team Coordination Problems. In *Proc. of the 21st International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2022), Auckland, New Zealand, May 9–13, 2022*, IFAAMAS, 3 pages.

1 TUTORIAL OVERVIEW

Robot teams are increasingly being deployed in environments, such as manufacturing facilities and warehouses, to save cost and improve productivity. To efficiently coordinate multi-robot teams, fast, high-quality scheduling algorithms are essential to satisfy the temporal and spatial constraints imposed by dynamic task specification and part and robot availability. Traditional solutions include exact methods, which are intractable for large-scale problems, or application-specific heuristics, which require expert domain knowledge. What is critically needed is a new, automated approach that automatically learn lightweight, application-specific coordination policies without the need for hand-engineered features.

This tutorial is an introduction to graph neural networks and a showcase of the power of graph neural networks solving multi-robot coordination problems. We survey various frameworks of graph neural networks in recent literature, with a focus on their application in modeling multi-agent systems. We will introduce the multi-robot coordination (MRC) problems and review the most relevant methods available to solving MRC problems. We will discuss several successful applications of graph neural networks in MRC problems, with hands-on tutorials in the form of Python example code. With this tutorial we aim at providing an experience of employing graph neural networks in modeling multi-robot systems, from algorithm development to code implementation, thus opening up future opportunities in designing graph-based learning algorithms in broader multi-agent research.

2 TUTORIAL OUTLINE

In this tutorial, we will discover the power of graph neural networks for learning effective representations of multi-robot team coordination problems. The tutorial will feature two 90-minute sessions.

The first session will address the following: (a) How graph neural networks work – we will present a comprehensive overview of various graph neural networks proposed in prior literature, including both homogeneous and heterogeneous graphs and attention mechanisms; (b) How to model team coordination problems with graph neural networks – we will discuss what are the suitable applications that can be modeled in graph neural networks, with a focus on MRC problems; (c) How to optimize the parameters of graph neural networks for team coordination problems – we will discuss what learning methods can be used for training a graph neural network-based solver. We conclude this session with the most recurrent challenges and open questions.

The second session will provide a hands-on tutorial for how to get up and running with graph neural networks for coordination problems, with coding examples in Python Jupyter notebook. In particular, we will look into the ScheduleNet architecture [15], a heterogenous graph neural network-based solver for MRC problems under temporal and spatial constraints. The Jupyter notebook will work through the model implementation, training and evaluation of ScheduleNet models on synthetic dataset.

Suggested Duration. Half day (3 hours).

Tutorial Format. Entirely virtual. Both sessions will be streamed via Zoom or Bluejeans.

3 TARGET AUDIENCE

The target audience is intended to be students and researchers who have a strong grasp of machine learning but who may as of yet be unfamiliar with graph neural networks. Prior knowledge of Multi-agent Reinforcement Learning (MARL) or Planning & Scheduling would be helpful but is not necessary. While the tutorial showcases the application of graph neural networks in solving multi-robot coordination problems, the methodology introduced can be utilized by the audience to broader research problems in learning representations of multi-agent systems.

4 CONTENT

The tutorial will cover the following aspects.

4.1 Graph Neural Networks

Graph neural networks (GNNs), extending DNNs to learn from graph-structured data, were introduced by Scarselli et al. [9]. Research in GNNs either work with a spectral representation of the graphs directly [1], or define convolutions on the graph and operate on groups of spatially close neighbors [5]. Graph Attention Networks (GATs) [11] were proposed to learn the importance between nodes and its neighbors and fuse the neighbors by normalized attention coefficients. Recently, researchers have proposed heterogeneous GNNs, allowing for learning with different types of nodes

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, Auckland, New Zealand.

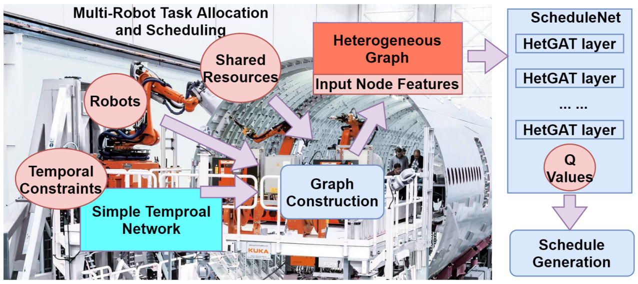


Figure 1: Overview of the ScheduleNet framework, which operates on the heterogeneous graph constructed by augmenting the STN of the problem, and predicts Q-values for scheduling. Courtesy: KUKA Robotics.

and edges, showing superior performance and model expressiveness [12]. GNNs have been widely applied in graph-based problems such as node classification, link prediction and clustering, and show convincing performance [16]. Specifically, GNNs have shown success in addressing resource optimization and scheduling problems, such as scheduling data processing clusters [7], wireless networks [3], and more [6].

4.2 Multi-Robot Coordination Problems

Given the recent developments in robotic technologies and the increasing availability of collaborative robots (cobots), multi-robot systems are increasingly being adopted in various manufacturing and industrial environments [17]. Research in related areas (e.g., multi-robot communication, team formation and control, path planning, task scheduling and routing) has also received significant attention [4]. In this tutorial, we focus on the problem of multi-robot task allocation and scheduling, with various real-world applications, such as manufacturing, warehouse automation and delivery systems [8]. To achieve an optimal schedule for a user-specified objective, the robots must be allocated with the proper number of tasks and process these tasks with optimal order, while satisfying temporal constraints such as task deadlines and wait constraints. The addition of spatial constraints (i.e., a specific work area can only be occupied by one robot at a time) makes scheduling difficult because one must reason through inter-coupled, disjunctive sequencing constraints impacting shared resources.

4.3 ScheduleNet

To overcome the limitations of conventional method in solving MRC problems, ScheduleNet is developed to learn high-performing scheduling policies for multi-robot teams under upper- and lower-bound temporal and spatial constraints [15]. Figure 1 shows the overall framework of ScheduleNet. We extend the simple temporal network (STN) [2], e.g. Fig. 2(a), that encodes the temporal constraints into a heterogeneous graph by adding nodes denoting various components, such as workers (human or robot) and physical locations or other shared resources. By doing so, ScheduleNet directly operates on the heterogeneous graph in a fully-convolutional manner and can estimate the Q-function of state-action pairs to be used for schedule generation. ScheduleNet is end-to-end trainable via imitation learning on small-scale problems and generalizes to large, unseen problems with an affordable increase in computation cost. This flexibility allows ScheduleNet to set a new state of the art for multi-robot coordination and in autonomously learning domain-specific heuristics for multi-agent applications.

Key to the success of ScheduleNet are the augmentations we make to the model representation, including attention mechanisms. For example, The original graph attention network [11] is only able to incorporate undirected, unweighted graphs, yielding that model insufficient for scheduling problems in which temporal constraints are represented by the direction and weight of the edge between the two corresponding event nodes. As such, we make two adaptations for the message passing and feature update phases as shown in Fig. 2(b): 1) The message passing follows the same direction of the edge (i.e., only the incoming neighbors of a node are considered); 2) Edge information is also aggregated when updating the node feature, which is done by adding a fully-connected layer inside each GAT layer that transforms the edge weight *edge* into the same dimension as the node feature using W_e. The output node feature $\vec{h}_i'$ is updated by (Eq. 1), where $N(i)$ is the set of neighbors of node i, W is the weight matrix applied to every node, $\vec{h}_j$ is the node feature from the previous layer, and α_{ij} are the attention coefficients. To stabilize the learning process, we utilize multi-head attention [11], consisting of K independent GAT layers computing nodes features in parallel and concatenating those features as the output.

$$\vec{h}_i' = \text{ReLU}\Big(\sum_{j \in N(i)} \alpha_{ij} (W\vec{h}_j + W_e edge_{ji}) \Big) \tag{1}$$

The GAT layer computes the feature embedding for each node by weighting neighbor features from the previous layer with feature-dependent and structure-free normalization, which makes the network non-parametric in the number of tasks. The pair-wise normalized attention coefficients are computed as shown in Fig. 2(b) using (Eq. 2), where $\vec{a}$ is the learnable weight, || represents concatenation, and $\sigma()$ is the LeakyReLU nonlinearity (with a negative input slope of 0.2). Softmax function is used to normalize the coefficients across all choices of j.

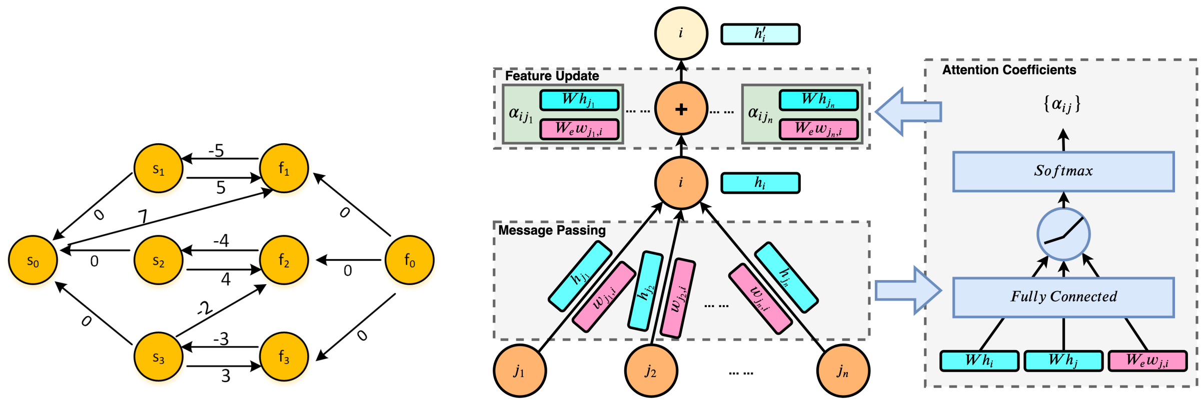
$$\alpha_{ij} = \text{softmax}_j \left(\sigma \left(\vec{a}^T \left[W\vec{h}_i W\vec{h}_j W_e edge_{ji} \right] \right) \right) \tag{2}$$

Given an STN and a set of robot-specific node features, the graph attention network, constructed by stacking several GAT layers, outputs the embeddings of each node. Then the robot embedding is obtained by averaging over all node embeddings.

5 PRESENTERS INFORMATION

The team has extensive research experience on this topic. Matthew Gombolay and Zheyuan Wang have co-authored several papers on applying graph neural networks in multi-robot research [10, 13–15]. Zheyuan Wang has written a introductory blog post on the subject (see https://core-robotics.gatech.edu/2020/09/17/scheduling-robots-gnn/). Resumes are attached at the end of proposal.

Dr. Matthew Gombolay (corresponding presenter) is an Assistant Professor of Interactive Computing at the Georgia Institute of Technology. He received a B.S. in Mechanical Engineering from the Johns Hopkins University in 2011, an S.M. in Aeronautics and



(a)

(b)

Figure 2: Fig. 2(a) depicts an STN with start and finish nodes for three tasks, as well as placeholder start and finish nodes, s_0 and f_0. Task 1 has a deadline constraint and there is a wait constraint between task 3 and task 2. Fig. 2(b) depicts the forward pass of the adapted graph attention layer (left-hand side), which consists of two phases: 1) Message passing: each node receives features of its neighbor nodes and the corresponding edge weights; 2) Feature update: neighbor features are aggregated using attention coefficients; the right-hand side illustrates how attention coefficients are calculated.

Astronautics from MIT in 2013, and a Ph.D. in Autonomous Systems from MIT in 2017. Gombolay's research interests span robotics, AI/ML, human–robot interaction, and operations research. Between defending his dissertation and joining the faculty at Georgia Tech, Dr. Gombolay served as a technical staff member at MIT Lincoln Laboratory, transitioning his research to the U.S. Navy, earning him an R&D 100 Award. His publication record includes a best paper award from American Institute for Aeronautics and Astronautics, a finalist for best student paper at the American Controls Conference, and a finalist for best paper at the Conference on Robot Learning. Dr Gombolay was selected as a DARPA Riser in 2018, received 1st place for the Early Career Award from the National Fire Control Symposium, and was awarded a NASA Early Career Fellowship for increasing science autonomy in space.

Zheyuan Wang is a Ph.D. candidate in the School of Electrical and Computer Engineering (ECE) at the Georgia Institute of Technology. He received the B.S. degree and the M. E. degree, both in Electrical Engineering, from Shanghai Jiao Tong University. He also received the M.S. degree from ECE, Georgia Tech. He is currently a Graduate Research Assistant in the Cognitive Optimization and Relational (CORE) Robotics lab, led by Prof. Matthew Gombolay. His current research interests focus on graph-based policy learning that utilizes graph neural networks for representation learning and reinforcement learning for decision-making, with applications to human-robot team collaboration, multi-agent reinforcement learning and stochastic resource optimization.

REFERENCES

[1] Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann Lecun. 2014. Spectral networks and locally connected networks on graphs. In *International Conference on Learning Representations (ICLR2014), CBLS, April 2014*.

[2] Rina Dechter, Itay Meiri, and Judea Pearl. 1991. Temporal constraint networks. *Artificial intelligence* 49, 1-3 (1991), 61–95.

[3] Mark Eisen and Alejandro Ribeiro. 2020. Optimal wireless resource allocation with random edge graph neural networks. *IEEE Transactions on Signal Processing* 68 (2020), 2977–2991.

[4] Eduardo Feo Flushing, Luca M Gambardella, and Gianni A Di Caro. 2017. Simultaneous task allocation, data routing, and transmission scheduling in mobile multi-robot teams. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 1861–1868.

[5] Will Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. In *Advances in neural information processing systems*. 1024–1034.

[6] Tengfei Ma, Patrick Ferber, Siyu Huo, Jie Chen, and Michael Katz. 2020. Online Planner Selection with Graph Neural Networks and Adaptive Scheduling. *Proceedings of the AAAI Conference on Artificial Intelligence* 34, 04 (Apr. 2020), 5077–5084. https://doi.org/10.1609/aaai.v34i04.5949

[7] Hongzi Mao, Malte Schwarzkopf, Shaileshh Bojja Venkatakrishnan, Zili Meng, and Mohammad Alizadeh. 2019. Learning Scheduling Algorithms for Data Processing Clusters. In *Proceedings of the ACM Special Interest Group on Data Communication (SIGCOMM '19)*. 270–288.

[8] Ernesto Nunes, Marie Manner, Hakim Mitiche, and Maria Gini. 2017. A taxonomy for task allocation problems with temporal and ordering constraints. *Robotics and Autonomous Systems* 90 (2017), 55–70.

[9] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. 2008. The graph neural network model. *IEEE Transactions on Neural Networks* 20, 1 (2008), 61–80.

[10] Esmaeil Seraj, Zheyuan Wang, Rohan Paleja, Matthew Sklar, Anirudh Patel, and Matthew Gombolay. 2021. Heterogeneous graph attention networks for learning diverse communication. *arXiv preprint arXiv:2108.09568* (2021).

[11] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph Attention Networks. *International Conference on Learning Representations* (2018).

[12] Xiao Wang, Houye Ji, Chuan Shi, Bai Wang, Yanfang Ye, Peng Cui, and Philip S Yu. 2019. Heterogeneous Graph Attention Network. In *The World Wide Web Conference*. ACM, 2022–2032.

[13] Zheyuan Wang and Matthew Gombolay. 2020. Heterogeneous Graph Attention Networks for Scalable Multi-Robot Scheduling with Temporospatial Constraints. In *Robotics: Science and Systems*.

[14] Zheyuan Wang and Matthew Gombolay. 2020. Learning scheduling policies for multi-robot coordination with graph attention networks. *IEEE Robotics and Automation Letters* 5, 3 (2020), 4509–4516.

[15] Zheyuan Wang, Chen Liu, and Matthew Gombolay. 2021. Heterogeneous graph attention networks for scalable multi-robot scheduling with temporospatial constraints. *Autonomous Robots* (2021), 1–20.

[16] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. 2019. How Powerful are Graph Neural Networks?. In *International Conference on Learning Representations*.

[17] Zhi Yan, Nicolas Jouandeau, and Arab Ali Cherif. 2013. A survey and analysis of multi-robot coordination. *International Journal of Advanced Robotic Systems* 10, 12 (2013), 399.