Applying Graph Neural Network in Deep Reinforcement Learning to Optimize Wireless Network Routing

Xiao Xu
School of Electronics and Information Engineering
Suzhou University of Science and Technology
Suzhou, China
xuxiao@post.usts.edu.cn

You Lu
School of Electronics and Information Engineering
Suzhou University of Science and Technology
Suzhou, China
luyou@usts.edu.cn

Qiming Fu
School of Electronics and Information Engineering
Suzhou University of Science and Technology
Suzhou, China
413944186@qq.com

***Abstract*—At present, the traffic in wireless sensor networks (WSN) is growing at an extremely fast speed, consuming more and more network resources. This undoubtedly affects the transmission performance of WSN. Good and efficient routing technology is one of the key technologies to solve this problem. Limited by the dynamic network state, traditional routing technology faces some problems such as performance degradation and lack of learning ability. In contrast, Deep Reinforcement Learning (DRL), which has the ability of decision-making and online learning, has a better effect in facing the routing optimization problem. DRL can learn routing strategy online or offline through reinforcement learning mechanism and deep neural network. However, the existing routing models based on DRL use fully connected neural networks or convolutional neural networks, and cannot learn the network topology information. This will lead to the failure of the previously trained routing model in the face of a new network. Therefore, under the background that WSN nodes may fail, resulting in topology changes, this paper combines Graph Neural Network (GNN) with DRL, and proposes GRL-NET intelligent routing algorithm. The algorithm uses GNN instead of conventional neural network to construct DRL Agent. With the help of GNN, GRL-NET can not only learn the complex relationship among network topology, traffic and routing from the perspective of network topology, but also run in a network topology that has never appeared before. In order to evaluate the effect of GRL-NET, several groups of experiments were conducted under different traffic intensity. Experimental results show that GRL-NET can not only learn the best routing strategy, but also keep good results in the never-seen network topology.**

***Keywords*-graph neural network; deep reinforcement learning; wireless sensor networks; routing; energy conservation**

I. Introduction

Wireless sensor network (WSN) is composed of a large number of sensor nodes with limited energy deployed in a specific area, which is an autonomous multi-hop self-organizing network system[1]. At present, WSN has good application prospects, such as Internet of Things, environmental monitoring and beyond 5G(B5G). An important feature of WSN is that each node has limited energy. And limited by the arrangement position, once the nodes are arranged, it will become difficult to supplement energy by secondary human intervention[2]. Under the increasing traffic pressure, WSN will face the problem of intense energy consumption of nodes. In order to overcome these limitations in WSN, good and efficient routing technology is very important[3]. Through route optimization, the energy consumption of nodes and the network burden can be reduced.

At present, routing algorithms in wireless networks have been widely studied. In traditional wireless network routing research, algorithms are usually designed based on candidate clusters[4]. For example, the algorithm SOAR[5] uses the Expected Transmission Count(ETX) as routing metrics, and selects candidate node sets between source nodes and sink nodes and sets priority for routing. The algorithm GeRaf[6] selects the candidate forwarding node set based on the geographical location information and takes the distance between the node and the sink node as the routing metric. For a certain node, when the distance between a neighbor node and a sink node is smaller than the distance between the neighbor node and the sink node, the neighbor node is selected as a candidate node. Although this kind of routing algorithm can effectively improve the transmission performance of wireless networks, it faces the threat of routing holes. Considering the complexity and dynamic changes of the network state, these algorithms lack the corresponding self-learning and dynamic adjustment capabilities. Moreover, some traditional self-heuristic intelligent algorithms (such as ant colony algorithm and particle swarm optimization algorithm) have been tried to solve the routing optimization problem of wireless networks. Although these algorithms

978-1-6654-0745-8/21/$31.00 ©2021 IEEE
DOI 10.1109/CBD54617.2021.00045

can learn some knowledge from the environment, they have some problems such as slow convergence and local optimal solution[7]. The problems existing in the traditional wireless network routing algorithms will have a certain impact on the network performance. It is of great significance to find practical solutions to these problems.

In recent years, machine learning technology has developed rapidly and is widely used to solve practical problems. Among them, Deep Reinforcement Learning (DRL) performs well in the control decision-making problem with markov decision processes (MDP)[8]. By analyzing the routing optimization problem, it can be found that its future state only depends on the current state. This is a typical markov decision processes. Under this background, researchers regard DRL as the key to solve the routing optimization problem and do a lot of research. Sendra first proposed an intelligent routing algorithm based on Deep Q Network(DQN)[9]. It can learn the best routing strategy by means of neural network. In the follow-up research, researchers optimized the action representation and DRL Agent algorithm, and realized the global and real-time intelligent routing control[10–12].

However, it is well known that the energy of nodes in WSN is limited. Once one or more nodes fail due to energy depletion, the network topology will change. Some literatures show that when the network topology changes, the existing routing algorithm based on DRL can not work well[13]. The cause of this phenomenon can be considered as the graph structure (e.g., network topology) in the routing optimization scenario. At present, the routing algorithm based on DRL uses standard neural networks (e.g., convolutional neural networks, fully connected neural networks) to build Agent[14]. Although these traditional deep learning models have achieved good results in Euclidean spatial data (language, image, video, etc.), they have some limitations in processing non-Euclidean spatial data (e.g., chemical structure, computer network). This means that DRL cannot learn topology information when dealing with routing optimization problems.

Fortunately, with the introduction of GNN, the problems related to graph structure have been well solved[15]. By using GNN to model the graph structure, we can learn the relationships among graph elements and the potential rules in them. Therefore, considering the possible failure of WSN nodes leading to the change of topology, this paper combines GNN with DRL and proposes GRL-NET intelligent routing algorithm. Compared with the traditional wireless network routing algorithm, GRL-NET has the following two advantages:

1) Under the framework of DRL, GRL-NET obtains reward update Agent by interacting with network environment in a "trial and error" way. This means that GRL-NET can not only learn routing strategy from historical experience, but also realize online update through interaction with environment. In this case, GRL-NET can learn the best energy-saving routing strategy aiming at energy consumption.

2) GRL-NET uses GNN instead of conventional neural network. With the help of GNN, GRL-NET can not only learn the complex relationship among network topology, traffic and routing from the perspective of graph. In addition, it can run in a network topology that has never appeared before, without repeated training.

The rest of this paper is arranged as follows: the second part describes the principle behind GRL-NET; The third part introduces how to build GRL-NET. The fourth part introduces the comparative experiment and result analysis. The fifth part summarizes the full text and looks forward to the future research direction.

II. Background and Problem Description

In this paper, we propose GRL-NET, which is a routing algorithm based on DRL and GNN. GRL-NET takes DRL as its running framework. However, in order to break through the limitations of conventional DRL algorithm, GRL-NET uses GNN to build Agent in DRL. Therefore, this section is divided into two parts. First, we talk about the working principle of DRL, how to use DRL to solve the routing optimization problem and its defects. Secondly, we will describe the working principle of GNN in the routing scenario.

A. DRL for routing

As an important branch of machine learning, DRL is different from supervised learning and unsupervised learning, and it interacts with the environment to learn[16]. Different from traditional reinforcement learning methods, DRL uses deep neural network (DNN) instead of tables, and expresses strategies in a functional approximation way. In this way, DRL can avoid the dimension disaster problem faced by traditional table-based reinforcement learning in the face of complex practical problems.

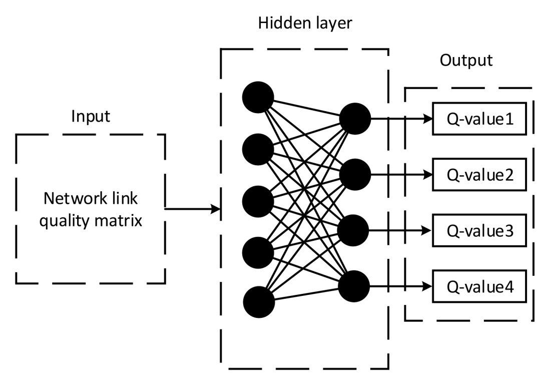

Figure 1: Structure of DQN Agent

Although trained DRL Agent can significantly improve the routing optimization problem. However, the DRL Agent can't work effectively in an unprecedented network topology. By analyzing the structure of DRL Agent, we can find the cause of this phenomenon. Taking the classic DQN as an example, the neural network model structure of its Agent in the end-to-end transmission routing problem is shown in Fig. 1. The model takes the network link quality matrix as the network state and inputs it into the neural network. The hidden layer of neural network consists of two layers of fully connected network and activation function. Finally, the model outputs Q-values corresponding to multiple candidate paths of end-to-end transmission. According to Q-value, a candidate path can be selected as the end-to-end transmission path by -greedy strategy. Generally speaking, this approach is effective. However, it should be noted that the input and dimension of the model are determined by the actual network scale (such as the size of network topology). Once the training is completed, the dimensions of model input and output become fixed. Therefore, when faced with a network of different scales, the model is likely to receive input with abnormal dimension size. Although we can change the input dimension by cutting or filling, it will break the potential topological information in the matrix. Even if the new network scale is the same, the dimensions of the input data need not be changed. However, the essence of computer network is graph. The existing method of using neural network to process state matrix can not learn the information in graph structure. This will limit the performance of DRL in the new network. In a word, although DRL performs well in routing optimization. However, it lacks the ability of relational reasoning and generalization of graph structure, and can't run effectively in the new network environment.

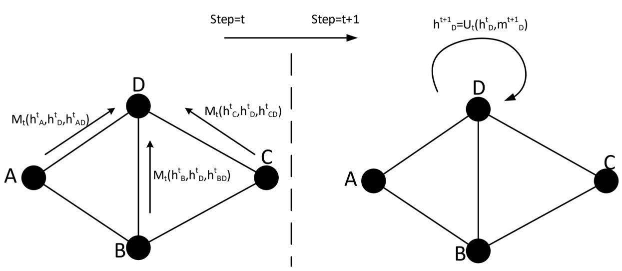

Figure 2: Operation process of MPNN

B. Graph Neural Networks in network scenario

GNN is a kind of method based on deep learning to deal with map domain information. Because GNN can learn the relationship between nodes, it has generalization ability in different graph structures. When learning with GNN, we can not only deal with the problems related to graph nodes, but also predict the relationship between nodes. At present, there are many different GNN models. The most commonly used one is Message Passing Network(MPNN)[17]. In the network routing scenario, the structure of MPNN is shown in Fig. 2. The running process of MPNN includes aggregation and update. In the aggregation stage, for node v at time t, MPNN executes formula 1 and formula 2. In which h_v^t represents the characteristics of node v, $N(v)$ represents the neighbor node of node v, and e_{vw} represents the characteristics of the link between node v and neighbor node w. The message m_v^{t+1} obtained by node v at $t+1$ can be extracted by aggregation. In the update stage, the node feature h_v^{t+1} at $t+1$ is obtained by calculating the node v feature h_v^t and the message m_v^{t+1} through the update function U_t. In the total time step t, the above two steps are repeated continuously for each node. At last, according to Banach's Fixed Point Theorem, we can get the characteristics of each node. On this basis, many different types of tasks (such as node classification and traffic prediction) can be realized by using node characteristics.

$$m_v^{t+1} = \sum_{w \in N(v)} M_t\left(h_v^t, h_w^t, e_{vw}\right) \tag{1}$$

$$h_v^{t+1} = U_t\left(h_v^t, m_v^{t+1}\right) \tag{2}$$

ComparingComparing GNN with traditional neural network, we can find the difference. Traditional neural networks are good at dealing with well-structured data such as pictures and languages. In this connection, GNN can deal with irregular data such as graphs well and make full use of graph structure information. That's why we chose GNN to build DRL Agent to improve its generalization.

III. Design GRL-NET with GNN

A. network model

Generally, when DRL is used for route optimization, the network can be modeled as graph $G = (V, E)$. V represents the node set in the network, and E represents the edge set in the network. However, there is no clear topology in wireless networks. Therefore, it is necessary to build a topology between the source node and the sink node. At present, there are many ways to build topology in wireless networks. In this paper, we choose the method based on forward region to construct a multipath tree topology in wireless network. The result is shown in Fig. 3. The network topology can be obtained by this method.

In the traditional calculation process of DRL, the network state is expressed in the form of traffic matrix or link capacity matrix. However, this representation method can't let Agent learn topology information, and it doesn't fit the design concept of MPNN. Therefore, in this paper, the network state is expressed as $s = (edge\_index, x)$. $edge\_index$ is a matrix with dimension $2 * E$. E is the number of links in the network. Two elements in each column in $edge\_index$ indicate that there is a link composed of these two nodes in the network. X is a matrix with dimension $N * L$. N is the number of nodes in the network.

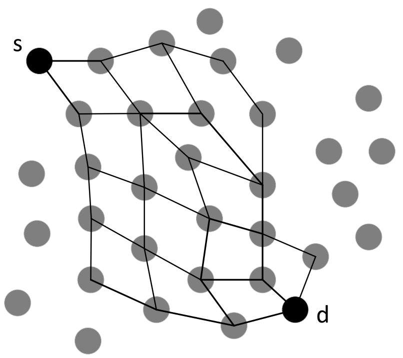

Figure 3: Create tree topology in WSN

L is the feature vector length of each node. X stores the set of feature vectors of network nodes. The learning goal of DRL Agent is to give the transmission path from source node to sink node according to $edge\_index$ and X . The following description of GRL-NET will be based on $s = (edge\_index, x)$.

B. DRL Agent in GRL-NET

In this part, we describe how GRL-NET constructs DRL Agent based on GNN. Generally speaking, the path selection methods of DRL Agent in route optimization can be divided into explicit and implicit ones[18]. Explicit path selection will lead to convergence difficulties with the growth of network scale. Therefore, this paper chooses the implicit path selection method. We use a set of link weight to represent the actions of Agent. Then we can find the path between the source node and the sink node by Dijkstra algorithm. Obviously, under the implicit path selection method, the action space of DRL Agent is continuous. Therefore, DRL-NET chooses deep deterministic policy gradient (DDPG) algorithm to build Agent.

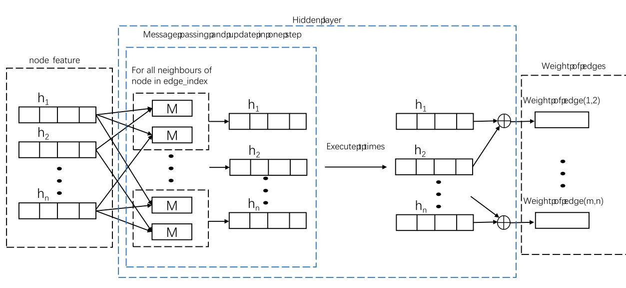

Figure 4: Structure of Actor network

DDPG is a model-free strategy algorithm under continuous action. It is based on actor-critic framework. In concrete implementation, DDPG Agent consists of Actor network and Critic network. The role of the Actor network is to give actions (link weight) according to the network status. Its structure is shown in Fig. 4. The network input is the network state $s = (edge_index, x)$. In Actor network, node feature vector is composed of background flow, residual energy, node degree and flow size. GNN based on MPNN is used as the hidden layer. Aggregate and update the node status in step t. Finally, the weight of each link can be obtained as the output of Actor network by aggregating the characteristics of nodes. The function of Critic network is to evaluate the action given by Actor network in network state S. Its network structure is shown in Fig. 5. The input of Ctitic network is almost the same as that of Actor network. However, in order to represent the network state S and the routing result of the Actor network at the same time, the node feature vector consists of background traffic, residual energy, node degree, traffic size, and whether it is a transmission node. The hidden layer runs in the same way as the Actor network. Finally, the features of all nodes are gathered to get Q-value. In addition, in the process of Agent learning, DDPG adopts experience playback mechanism and delayed update. By this method, Agent can learn the routing strategy in continuous action space better.

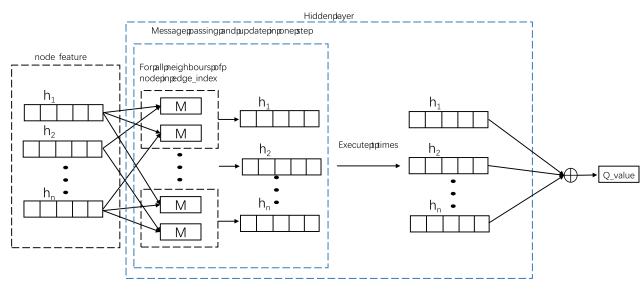

Figure 5: Structure of Critic network

C. GRL-NET operation process

Based on the above Actor and Critic networks, we can build DDPG Agent. Under the optimization framework of DRL, the optimization process of DRL-NET is as follows. In the initial stage, a multi-path tree topology needs to be established according to the source node and the convergence building. In the training stage, the Agent will go through several times of epic. At the beginning of each scenario, the environment gives an initial state s . The Actor network gives the corresponding action a according to the state. According to the action, and sends it to the network environment for execution. The environment gives the corresponding feedback reward r and shifts to the next hop state s'. In this way, samples (s,a,r,s') can be obtained and stored in the experience pool. Then, the Agent needs to be updated. When the size of the experience pool is sufficient, we update the Actor and Critic networks by batch sampling. After iteration, the Agent can finally learn the optimal routing strategy.

IV. Evaluation experiment

In this section, we trained GRL-NET in the simulation environment and evaluated its optimization effect and generalization. The secion is divided into two parts. Firstly, the paper describes the detailed parameters and network environment of GRL-NET. Secondly, we evaluate GRL-NET and analyze the results.

A. Environment setup

With the application of GNN, we use PyTorch Geometric to realize GRL-NET. The hidden layer of Actor network consists of two layers of GNN. To prevent over-fitting, a layer of Dropout is added to the hidden layer. The activation function selects Selu and Sigmoid. The hidden layer structure and activation function of Critic network are the same as those of Actor network.

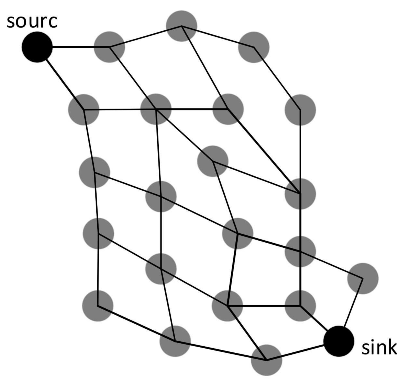

Figure 6: WSN topology

In terms of network environment setting, we build the topology shown in Fig. 6 in the wireless network. Each node has limited energy and randomly generated forwarding delay. The flow from the source node to the sink node is randomly generated according to Poisson distribution. At the beginning of training, we set an exploration rate of 0.3 for Agent to generate random actions. With the training, the exploration rate decreases gradually. However, it should be noted that the exploration rate will not be completely zero. In this way, the Agent's exploration of the environment can be preserved.

B. Convergence of GRL-NET

This experiment evaluates the performance of GRL-NET in the training process with energy consumption as the optimization goal. The experimental results are shown in Fig. 7. In Fig. 7, the abscissa is the training episode, and the ordinate is the average energy consumption in episode. By observing Fig. 7, it can be found that with the training, the energy consumed in episode is gradually decreasing. Finally, the experimental results converge to a lower point.

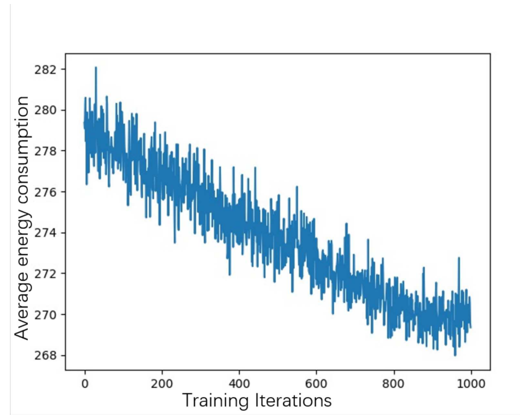

Figure 7: Energy consumption optimization progress

This shows that DRL-NET can effectively reduce the transmission energy consumption in wireless networks and has good convergence.

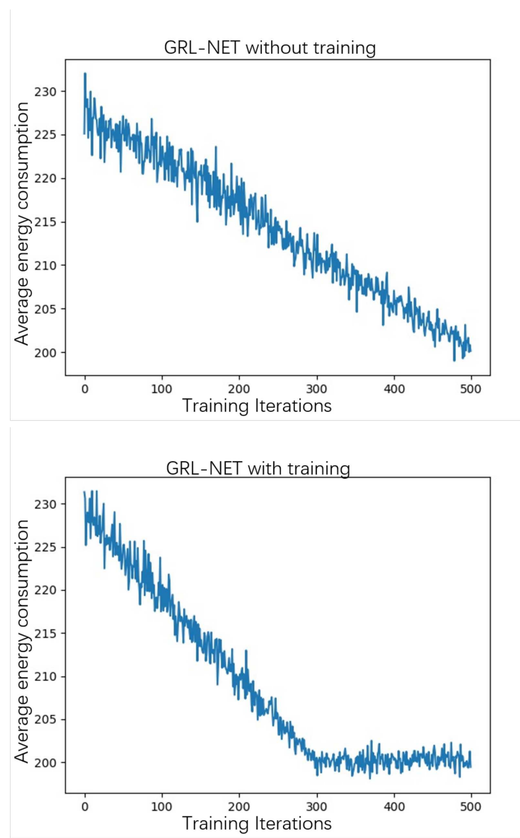

Figure 8: Generalization evaluation of GRL-NET

C. Generalization of GRL-NET

This experiment evaluates the generalization ability of GRL-NET under the new network topology. We put GRL-NET which has been trained into a new network topology and observe its convergence. For comparison, another untrained GRL-NET can be placed in this topology. In this way, we can compare the performance of GRL-NET under the new network topology.

The result is shown in Fig. 8. It can be found in Fig. 8 that the trained GRL-NET can obtain lower transmission energy consumption at the beginning. Moreover, it can converge faster. This shows that GRL-NET has good convergence. Even in the brand-new network topology, GRL-NET can still run effectively, without a lot of training.

V. Conclusion

In this paper, in order to better realize routing in WSN, we propose GRL-NET routing optimization algorithm based on GNN and DRL. Experiments show that GRL-NET can not only realize learning strategies according to optimization objectives, but also have good generalization ability.

In the follow-up study, we will further explore the potential of GRL-NET. At present, GRL-NET can realize single objective optimization. On this basis, we hope that GRL-NET can optimize multiple objectives. In this way, it can better meet the requirements of future wireless networks for routing algorithm performance.

References

[1] Kumar D. Kaur T. Maco-qcr: Multi-objective aco based qos-aware cross-layer routing protocols in wsn. In *IEEE Sensors Journal*, 2020.

[2] Hou X et al. Sheraz M, Ahmed M. Artificial intelligence for wireless caching: Schemes, performance, and challenges. 2020.

[3] Barhumi I . Al-Tous H. Reinforcement learning framework for delay sensitive energy harvesting wireless sensor networks. *IEEE Sensors Journal*, 2020.

[4] Huang Y . Liu X, Chen Z. Research on reliability-guaranteed opportunity routing algorithm for smart distribution power grid wsns. *Chinese Journal of Sensors and Actuators*, 2019.

[5] E. Rozner, J. Seshadri, Y. Mehta, and L. Qiu. Soar: Simple opportunistic adaptive routing protocol for wireless mesh networks. *IEEE Transactions on Mobile Computing*, 8:1622–1635, 2009.

[6] Rao et al. Zorzi, M. Geographic random forwarding (geraf) for ad hoc and sensor networks: multihop performance. *IEEE Transactions on Mobile Computing*, 2(4):337–348, 2003.

[7] Zirui Zhuang, Jingyu Wang, Qi Qi, Haifeng Sun, and Jianxin Liao. Toward greater intelligence in route planning: A graph-aware deep learning approach. *IEEE Systems Journal*, 14(2):1658–1669, 2020.

[8] Penghao Sun, Julong Lan, Zehua Guo, Di Zhang, Xianfu Chen, Yuxiang Hu, and Zhi Liu. Deepmigration: Flow migration for nfv with graph-based deep reinforcement learning. *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, pages 1–6, 2020.

[9] Sandra Sendra, Albert Rego, Jaime Lloret, Jose Miguel Jimenez, and Oscar Romero. Including artificial intelligence in a routing protocol using software defined networks. In *2017 IEEE International Conference on Communications Workshops (ICC Workshops)*, pages 670–674, 2017.

[10] Nguyen Cong Luong, Dinh Thai Hoang, Shimin Gong, Dusit Niyato, and Dong In Kim. Applications of deep reinforcement learning in communications and networking: A survey. In *IEEE Communications Surveys Tutorials*, volume PP, pages 1–1, 2019.

[11] Z. Xu, J. Tang, J. Meng, W. Zhang, Y. Wang, C. H. Liu, and D. Yang. Experience-driven networking: A deep reinforcement learning based approach. In *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications*, pages 1871–1879, 2018.

[12] Q. Xu, Y. Zhang, K. Wu, J. Wang, and K. Lu. Evaluating and boosting reinforcement learning for intra-domain routing. In *2019 IEEE 16th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*, pages 265–273, 2019.

[13] P. Almasan, J Suárez-Varela, A. Badia-Sampera, K. Rusek, P. Barlet-Ros, and A. Cabellos-Aparicio. Deep reinforcement learning meets graph neural networks: exploring a routing optimization use case. *arXiv*, 2019.

[14] Peter W Battaglia, Jessica B Hamrick, V. Bapst, A. Sanchez-Gonzalez, V. Zambaldi, M. Malinowski, A. Tacchetti, D. Raposo, A. Santoro, and R. Faulkner. Relational inductive biases, deep learning, and graph networks. 2018.

[15] Krzysztof Rusek, José Suárez-Varela, Paul Almasan, Pere Barlet-Ros, and Albert Cabellos-Aparicio. Routenet: Leveraging graph neural networks for network modeling and optimization in sdn. *IEEE Journal on Selected Areas in Communications*, 38(10):2260–2270, 2020.

[16] Chenyi Liu, Mingwei Xu, Nan Geng, and Xiang Zhang. A survey on machine learning based routing algorithms. In *Journal of Computer Research and Development(chinese with english abstract)*, volume 57, pages 671–687, 4 2020.

[17] J. Gilmer, Samuel S Schoenholz, Patrick F Riley, O. Vinyals, and George E Dahl. Neural message passing for quantum chemistry. 2017.

[18] A. Valadarsky, M. Schapira, D. Shahaf, and A. Tamar. Learning to route. In *the 16th ACM Workshop*, 2017.