Title	Deep Reinforcement Learning for Wireless Multihop Networks: Factor Graph and Nested Lattice Approaches
Author(s)	崔，志瀚
Citation	
Issue Date	2023-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/18313
Rights	
Description	Supervisor: リム 勇仁，先端科学技術研究科，修士(情報科学)

Master's Thesis

Deep Reinforcement Learning for Wireless Multihop Networks: Factor Graph and Nested Lattice Approaches

CUI Zhihan

Supervisor Associate Professor Yuto LIM
Second Supervisor Professor Yasuo TAN

Graduate School of Advanced Science and Technology
Japan Advanced Institute of Science and Technology
Master of Science (Information Science)

March 2023

DEEP REINFORCEMENT LEARNING FOR WIRELESS MULTIHOP NETWORKS: FACTOR GRAPH AND NESTED LATTICE APPROACHES

By CUI Zhihan (2110078)

A thesis submitted to
School of Information Science,
Japan Advanced Institute of Science and Technology,
in partial fulfillment of the requirements
for the degree of
Master of Science

Supervisor :	Associate Professor LIM, Yuto
Main Examiner :	Associate Professor LIM, Yuto
Examiners :	Professor TAN, Yasuo
	Professor KURKOSKI, Brian Michael
	Associate Professor BEURAN, Razvan Florin

Graduate School of Advanced Science and Technology
Japan Advanced Institute of Science and Technology
(Information Science)

February 2023

Abstract

Wireless Multihop Network (WMN) has been considered as the possible technology of Device-to-Device (D2D) to provide the services for increasing the traffic in beyond 5th Generation mobile network (5G). In WMN, these devices are all wirelessly connected to each other in a mesh and forward data to target device over the network using other devices as relay nodes. Data hops from device to device until it reaches its destination. Due to the advantages of carrying many users and having high throughput, WMN can meet the high demand of next-generation wireless communication and has been actively studied.

But there are still some issues that need to be studied and resolved in WMN, like due to the uncertainty of source node choosing a path to send the message and the nature of multihop fashion, the performance of network capacity can degrade drastically. To solve this problem, a appropriate path selection algorithm is needed for the device to determine the path composed of relay nodes when sending data.

There are many achievements in Machine Learning (ML) for path selection problems. With the development of hardware equipment, the high computational complexity of ML has also been solved. Among them, Deep Reinforcement Learning (DRL) can solve the path selection problem most effectively and has been applied to wireless networks. However, the delay caused by the computation time of DRL still cannot meet the low-latency requirements of the future communication, and other methods are needed to solve it. The high latency problem is solved from the transport layer using network coding.

To solve these two problems in WMN, the purpose of this research is to propose a novel Factor Graph-based Deep Reinforcement Learning (fDRL) scheme with two learning path selection algorithms for the Q-Learning called SNR-based Learning Path Selection (NLPS) algorithm and SINR-based Learning Path Selection (INLPS) algorithm. These two algorithms train the model using DRL to select best multihop path from source device to target device with highest transmission rate to increase total network capacity in WMN. The difference is that INLPS algorithm considers interference by using SINR as reward while NLPS algorithm using SNR as reward. In NLPS and INLPS, Factor Graph (FG) representation is used to reduce the heavy iteration of training phase in DRL. For latency problem, Nested Lattice Code (NLC) is used in Compute-and-Forward (CoF) strategy to reduce the time slots when data is transmitting.

According to the theoretical and numerical studies with the assumption of the system model, simulation results reveal that FG can reduce computation time up to 99% for DRL. As for two proposed algorithms, NLPS and INLPS increase the network capacity. When there are 50 nodes, the average network capacity of INLPS (2.52 Mbps) is about 3.19 times higher than NLPS (0.79 Mbps) and 6.46 times higher than use FG and NLC (0.39 Mbps). When the number of nodes is 100, the values become 1.83 Mbps, 0.43 Mbps and 0.36 Mbps respectively and INLPS algorithm is about 4.26 times higher than NLPS and about 5.08 times higher than use FG and NLC. For number of iterations, NLPS uses 101 iterations to reach 98% of highest throughput and INLPS uses 185 iterations when the number of nodes is 50. These two algorithms use 156 iterations and 263 iterations respectively when the number of nodes is 100.

Based on the result, the thesis concludes that FG can reduce the computation time for DRL before training. With the ad of FG, NLPS and INLPS can find best path for each node in a shorter time. Both NLPS and INLPS can increase network capacity. The difference is that INLPS increase more but NLPS cost a small number of iterations. With the aid of NLC, the network capacity is increased further.

Keywords: Deep Reinforcement Learning; Factor Graph; Nested Lattice Code; Learning Path Selection Algorithm; Wireless Multihop Networks

Acknowledgement

Foremost, I would like to express my deepest appreciation to my supervisor, Associate Professor Yuto LIM for his patient guidance and support for this study. His sincerity and motivation have deeply inspired me a lot and his generosity helped my time in JAIST enjoyable. As my second supervisor, I would also like to extend my deepest gratitude to Professor Yasuo TAN for his support and constant encouragement to continue my study.

I also sincerely thank Professor KURKOSKI, Brian Michael for his patient instruction in my minor research. With his guidance and sharing, I have acquired the concepts and knowledge about the new research area, Information Theory.

I would like to express my deep appreciation to researchers from TAN Laboratory and WiSE Laboratory (LIM Laboratory) for their help and sharing in collaboration meetings. With their friendliness, I enjoyed my student very much during these two years.

What is more, I would like to extend my sincere thanks to Dr KHUN Aung Thura Phyo, seniors in my lab, for his knowledge sharing and guidance for this thesis through my Master's program.

Finally, I would like to thank unknown significant others (USO) who help me directly or indirectly to complete my masters' degree. Last but not least, I am always thankful to my family for their love and letting me decorate my brighter future by myself.

Contents

List of Figures

List of Tables

List of Symbols

The following list describes several symbols that are used within the body of this document:

α	Learning rate	
β	Attenuation constant	
ϵ	Threshold	[bps]
η	Noise level	[dBm/Hz]
γ	Discount factor	
τ	Transmission time of node i	
$a_m^{'}$	The best action in mth layer	
a_m	Action of mth layer	
B	Channel Bandwidth	[Hz]
B_t	Number of bits in test frame	[bits]
C_m	The mth layer	
D	Destination node	
d_0	Decorrelation distance	[m]
d_{ij}	Distance between node i and node j	[m]
e_f	Frame Error Rate	
$F(A)$	Global function of root node A	

G_{ij}	Power ratio between node i and node j	
k	Number of nodes in the corresponding layer	
K_m	Total number of nodes in mth layer	
L	Size of data packet	[bytes]
l	Airtime cost	[μs]
M	Total number of layers	
N	Total number of wireless nodes	
$n_m^{k_m}$	The kth node in mth layer	
O	Channel access overhead	[μs]
P_i	Transmit power of node i	[dBm]
PL_0	Pathloss under Friis free space model	
PL_{ij}	Channel gain between node i and node j	[dB]
$Q_m(i,j)$	Q-value of mth layer from node i to node j	
R_0	Basic rate of test frame	[bps]
r_{ij}	Reward between node i and node j	
S	Source node	
s_m	State of mth layer	
$SINR_{ij}$	Signal to interference plus noise ratio at node j from node i	
SNR_{ij}	Signal to noise ratio at node j from node i	
t	Number of iterations	
T_v	Transmission time of vth time slot	[s]
U	End-to-end throughput	[bps]
V	Total number of time slots	

W_{ij}	Wall attenuation from node i to node j	[dB]
X_σ	Gaussian random variable with zero mean, shadowing attenuation caused by flat fading	[dB]

List of Abbreviations

5G 5th Generation mobile network

AI Artificial Intelligence

ALM Airtime Link Metric

APs Access Points

AWGN Additive White Gaussian Noise

BS Base Station

CoF Compute-and-Forward

D2D Device-to-Device

DRL Deep Reinforcement Learning

fDRL Factor Graph-based Deep Reinforcement Learning

FER Frame Error Rate

FG Factor Graph

INLPS SINR-based Learning Path Selection

LPS Learning Path Selection

ML Machine Learning

NLC Nested Lattice Code

NLPS SNR-based Learning Path Selection

RL Reinforcement Learning

SINR Signal-to-Interference-plus-Noise Ratio

SNR Signal-to-Noise Ratio

SP Sum-Product

SPST Shortest Path Spanning Tree

WMN Wireless Multihop Network

Chapter 1

Introduction

Wireless communication is a system of communication that supports the transmission of information (voice, video, data, etc) over large distances using free space as the communication medium. As the latest step in how wireless communications is connecting to the Internet, 5th Generation mobile network (5G) is well known to the computer networking era. As a promising technology for 5G and future wireless networks, it has been attractive as an active research field for decades. To cope with the growth of mobile data traffic and devices, the later generation of the wireless system such as 5G or beyond 5G (B5G/6G) is expected to be developed with the standard for the dense environment. Therefore, it is crucial to take into consideration how to improve the technology that brings advanced wireless systems like 5G.

This chapter will introduce the background environment of the research, the research problem that degrades the performance of the wireless network systems. And them, the research motivation with objectives and how the research is going to conduct for solving the research problem will be discussed.

1.1 Research Background

According to Mohammed H. Alshari [3], by 2026, it is expected that approximately 65% of the world's population will be on 5G networks. It means the number of people using mobiles devices such as smartphones, smartwatches and so on, will have great growth. As next generation, 6G communication is expected to provide better services for users than 5G, such as a larger network coverage, high throughput and to accommodate large numbers of users and low latency communication at the same time. Some novel technologies will be applied to 6G, including extremely large bandwidth (more than 1,000MHz waves) and high Artificial Intelligence (AI), including network services, business and network environment. Figure 1.1 [1] presents the timeline of 6G communication.

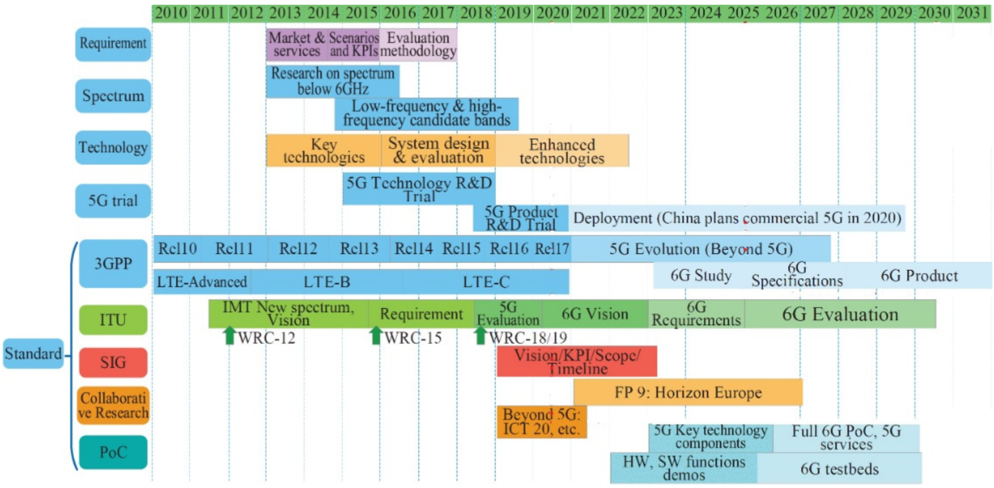


Figure 1.1: Timeline of 6G wireless networks [1]

6G system is expected to attend a remarkable revolution of internet. This revolution will completely differentiate 6G from previous networks and evolve wireless communication from the “Internet of Things” to “Internet of Intelligence”. Particularly, 6G need to support ubiquitous AI services from the cloud network to edge devices and surpass the mobile networks. AI will promote the development of 6G in designing and optimizing architectures, protocols, and operations. 6G communication will support the following three expected scenarios of applications in future wireless network systems:

- Enhanced Mobile Broadband-Plus (eMBB-Plus): In 6G, eMBB-Plus will replace eMBB in 5G. eMBB-Plus can provide users with high data rates, large network coverage area and high-quality of experience (QoE) in data utilization and standards;

- Secure ultra-reliable low-latency communications (SURLLC): Vehicular communications in 6G could also largely benefit from SURLLC [4].

SURLLC in 6G is an advancement of the URLLC and the mMTC in 5G It has more stringent demands on latency and reliability;

- Unconventional data communications (UCDC): Currently, the actual meaning and composition of UCDC [5] lack proper definition. But some of the following facets should be addressedi n 6G communication: holographic, tactile, and human-bond communications.

6G is a research field for serving data transmitting services with applying all new techniques. These various techniques are expected to be apply in 6G in the future like Distributed Massive Multi-Input Multi-Output (MIMO) system and Reconfigurable Intelligent Surface (RIS). Also, G. Trivedi [6] explained that Wireless Multihop Network (WMN) is also one of the main technologies in 6G, which can decentralize the calculations, organize network automatically,reach high network capacity and be deployed in a short time.

However, there are still many problems that degrade the overall network performance. Although WMN brings the attractive feature of increasing network capacity, there are still many problems that degrade the overall network performance. Without careful planning each relay nodes for every data packet, it is difficult to efficiently utilize all nodes and increase network capacity. Some of the challenges include high interference, overhead problem, and highly susceptible to link failure.

We will briefly introduce and identify some of the problems of the WMN wireless communication in this chapter by following the motivation, objectives and approach through the research.

1.2 Problem Statement

With the development of society and the progress of technology of 6G, people demand more higher requirements for wireless data transmission, especially in terms of user experienced capacity, power consumption, ultra-low latency, and so on. 6G wireless networks should increase the network spatial spectrum and energy efficiency as much as possible while increasing the number of users. WMN meets requirements by using other nodes as the relay nodes and transmit messages from a source node to a destination node over longer distance. Thus, WMN only can extend the network coverage, but it can degrade the network capacity drastically due to the uncertainty of source node choosing a path to send the message and also the nature of multihop fashion. In particular, the first problem is happened when the number of nodes in the network is increasing, there are many data transmission paths to be selected from a source node to the corresponding destination node by an efficient path selection algorithm. Since the criteria and conditions of each path are different, the result of the path selection has a great influence on the entire network capacity. Second problem is occurred when a node has to send not only its own message, but also other nodes' messages in the network, in which it can lead to high latency for the sent message to reach its destination node. Besides that, the queuing time and processing time will increase drastically when when the number of nodes is large.

Therefore, the problems addressed in this thesis are to use AI to select the path for each source node to destination node and form the network topology with the highest throughput in WMN. Besides that, it is also necessary to solve the high latency problem caused by traffic congestion and the long training time brought by AI.

1.3 Related Works and Motivation

This section elaborates the existing related works of this thesis into two main parts. First, we investigate the network capacity analysis in the WMN. Second, we summarize the studies of Factor Graph (FG), Nested Lattice Code (NLC) and Deep Reinforcement Learning (DRL) in the WMN.

A plethora of research works on the network capacity in the wireless networks. These research works can be mainly divided into three categories, i.e., analytical modelling, network routing, transmit power control. In the analytical modelling, C. Fujimura et al. [7] proposed an analytical expressions for maximum end-to-end throughput varying number of hops and payload length in the string-topology network. Besides that, S. Rezaei et al. [8] considered a routing policy and nodes' distribution as well as medium access control (MAC) layer together into analytical modeling of end-to-end throughput. In the viewpoint of network routing, W. Lee et al. [9] and J. Gui et al. [10] are jointly considered the node fairness and the energy saving when the routing policy of wireless networks is designed, respectively. Some researchers also look into the routing problem in the tree-based structure topology. For example, D. Eliiyi et al. [11] proposed a parallel algorithm to find all root nodes of a network in which the root node can considerably reduce the overall energy consumption and increase the network lifetime. Through this algorithm, the root node can greatly reduce the computation time as well. Meanwhile, Y. Yu et al. [12] studied the consensus transmit power control (CTPC) algorithm to maximizing end-to-end throughput in the highly dense WMN environment. The proposed CTPC algorithm is extended into the investigation of full-duplex system for WMN environment. [13]. More specifically, A.T.P. Khun et al. [14] presented an optimal achievable transmission capacity (OATC) scheme that enables the transmission modes of mixture of concurrent and sequential transmissions (MCST) scheme or concurrent transmission (CT) only in the full-duplex WMN environment. This novel OATC scheme reveals high transmission capacity with low transmit power and low interference power regardless of the number of nodes is increasing.

For FG, Y. Mao et al. [15] studied the low complexity algorithmic framework of FG for link loss monitoring in the centralized manner of wireless sensor networks. The proposed algorithm iteratively updates the estimates of link losses upon receiving or detecting the loss of recently sent packets by the sensors. Similarly, W. Li et al. [16] focused on the nonparametric variant of sum-product algorithm, called sequential particle-based SPA (SPSPA), for FG to infer the multi-sensor target states over time in the distributed manner of wireless sensor networks. Both studies show the great achievement of FG in wireless networks. In recent year, C. Jiang et al. [17] applied the FG into the real smartphone navigation system, i.e., Pedestrian dead reckoning exploring human walking gaits, in which the FG can effectively solve practical application problems in the wireless communication.

A few of research works on the network capacity analysis using both coding theory and factor graph representation. In the coding theory, X. Bu et al. [18] applied the network coding in the WMN to solve maximum-minimum optimization problem of cooperative communication. Their research work can significantly increase the capacity of wireless networks. Besides that, they also considered to jointly optimize relay node selecting, scheduling and flow routing for the cooperative communication in the WMN environment. As for the lattice coding theory, there are no studies that apply to the WMN. But according to J. Xue et al. [19], the lattice decoder can achieve low word error rate (WER) for power-constrained wireless communications. On the other hand, Y. Mao et al. [15] studied the low complexity algorithmic framework of FG for link loss monitoring in the centralized manner of wireless sensor networks. The proposed algorithm iteratively updates the estimates of link losses upon receiving or detecting the loss of recently sent packets by the sensors.

In 1959, the concept of Machine Learning (ML) was first proposed by Arthur Samuel in [20]. However, limited by the technology at the time, hardware equipment could not meet ML's demand for high computation. In 1970s, 'AI winter' happened caused by pessimism about machine learning effectiveness. In recent years, with the development of hardware technology, the computing power of processors has become stronger and stronger, and ML has gradually become popular. With the popularity of ML, there have been many studies combining it with wireless networks in recent years. J. Rosenberger et al. [21] apply DRL Multi-Agent System (MAS) in different devices to decentralize the calculation to locate the resources in Industrial Internet of Things (IIoT). Even the systems and resources are keep changing, there method runs very well and time is very low, which inspired us to use DRL to reduce computation time.

Among all kinds of ML, Reinforcement Learning (RL) is the most suitable for solving path selection problems and some researchers have applied it to the WMN. D. A. Dugaev et al. [22] presented an application of RL-based algorithms to the routing task in wireless multihop topologies and a flexible, reliable, adaptive packet forwarding scheme has been developed, which showed significantly better results in packet loss ratio and route recovery time values, compared to the classical routing approach, widely used in the current ad hoc multihop networks.

RL has many derivative algorithms. Among them, Q-learning, as a model-free RL algorithm, processes data without environment model and adaptation, is widely used in the path selection problem of wireless networks. Researchers have applied Q-learning to various situation. Some of them focus on the interference channel like T. Wongphatcharatham et al. [23], who propose the multi-agent Q-learning to optimize the transmit power of transmitters within interference channel by maximizing Signal-to-Interference-plus-Noise

Ratio (SINR). There results shows that transmitters are able to allocate own transmit power and get better sum-rate than the traditional methods such as the maximum power allocation and the random power allocation. Some of them focus on the end-to-end transmission rate like X. Wang et al. [24], who propose a Q-learning-based relay selection algorithm to decentralized the computation of nodes in the multihop clustered networks based on the Q-learning and resulting a near-optimal E2E rate and better performance than traditional decentralized solutions from one source node to one destination node. But they ignore the interference and the situation where multiple users transmit data at the same time. With these research results, we decided to use Q-learning to solve the path selection problem proposed in this thesis.

Through these related research works, we realize that the study of using NLC design, FG approach and AI, especially Q-Learning in Reinforcement Learning (RL) on the influence on the network capacity of a WMN is not enough and needs more investigate. Thus, our motivation of this thesis is to further investigate the breakthroughs of network capacity improvement by using Q-Learning, NLC design and FG approach in the WMN environment.

1.4 Research Objectives

The objectives of this thesis are:

- Examine the performances of network capacity and computation time in WMN with FG and NLC;
- Propose novel fDRL scheme with two learning path selection algorithms for the Q-Learning to achieve better network capacity while reducing the computation time with the aid of FG.

1.5 Research Approach

To mitigate the problems addressed in this thesis, we consider DRL with both FG and NLC approaches in our research study. For first aforementioned problem, we apply FG-based DRL (fDRL) scheme to select a relay node for each hop and find best path to the destination node. With these paths, a resultant of network topology that can achieve a better network capacity can be established. For the second problem, NLC that is used to reduce the link error probability of a channel can correct the errors through the integration of Compute-and-Forward (CoF) strategy. By the way, the entire network capacity can be improved more by reducing the required number of time slots. Besides that, We also propose two novel learning path selection algorithms, i.e., SNR-based Learning Path Selection (NLPS) algorithm focuses on increasing the end-to-end throughput from source node to destination node considering SNR as reward of DRL; and SINR-based Learning Path Selection (INLPS) algorithm uses the results obtained from NLPS and considers SINR as reward during the training phase of DRL to find the appropriate path for any source node.

After reviewing the basic study of WMN, NLC and FG, we first evaluate the performance of the network capacity and computation time using FG and NLC to select the best network topology and compare it with not using these two tools as the first objective to define the research methodology of the research. Then, two novel path selection algorithms based on Q-Learning in WMN called NLPS algorithm and INLPS algorithm is proposed to obtain the maximum achievable capacity of the network. NLPS algorithm tries to find the best path with highest SNR for each node to the root node considering there are no interference in the channel. INLPS consider the interference of channel as SINR and mitigate the influence from interference by selecting the path with highest SINR for each node to the root node. Therefore, our main methodologies for this research are applying the NLPS algorithm and INLPS algorithm to find the best network topology in the WMN to optimize network capacity. And then, in the transmission layer, we apply NLC and Compute-and-Forward (CoF) to encode and decode for information and send data packet to increase the network capacity further by reduce the number of time slots.

1.6 Research Methodology

The research methodology is defined to revise the performance of FG, NLPS, INLPS adn NLC in WMN as the following workflow shown in the following Figure 1.2.

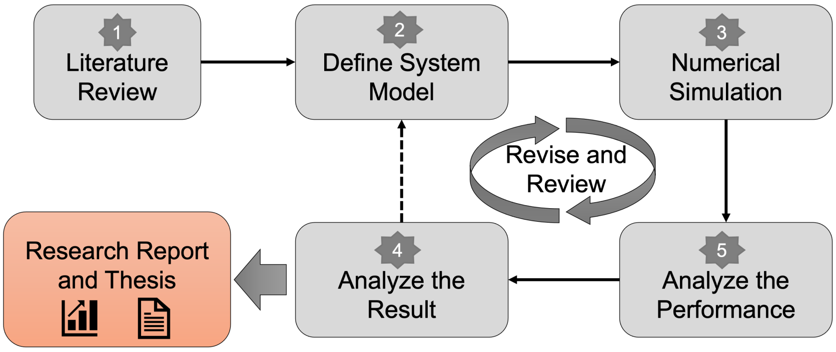


Figure 1.2: 5-step research methodology

After reviewing the related research in the literature reviewing section as the very first step of the methodology, in the system model step, the WMN model with D2D communication is considered as network model. Also, other models like interference model are defended in this step.

After defining the system model, the theoretical and numerical simulation is conducted to evaluate the netwprk capacity, computation time and number of iterations in WMN model with several scenarios: applying FG, applying FG and NLPS and applying INLPS. NLC is also used in transmission phase of each scenario. The capacity means the total time required to send all packets that each node needs to transmit to the root node divided by the total size of data packets that have been transmitted. The computation time is defined as the time spent to run the algorithm programs in each scenario. Depending on the hardware on which the program is running, the calculation time will change. The evaluation of number of iterations is only considered in the scenarios of NLPS and INLPS algorithms, including the performance of changes in parameters of RL (learning rate, discount factor and threshold).

The simulation results got from numerical simulation is analyzed and made into figures to analyze the performances of proposed algorithms. By comparing it to other results of related works, the system models are revised and reviewed. After simulating again, the results and performances is analyzed until get optimal one. The research report and program are written after that as a thesis.

1.7 Thesis Organization

The thesis of the research is organized with three main sections with a literature review to in-deep understand the background of the research, defining the research methodology to study the trade-off between NLPS algorithm and INLPS algorithm in the WMN, and formulating the research problem. The detail of the thesis is organized by the following:

In chapter 1: as the brief introduction section, the background introduction of the research, some of the challenge problems of WMN and the focus research problem of the research followed by the research motivation and objectives of the study are described. Besides that, the methods or approaches to investigate for solving the problem the research are briefly explained.

In chapter 2: the literature review of the fundamental knowledge related to wireless networks and basic theory of WMN, FG, NLC and DRL, including some methods used in this thesis to apply FG, NLC and DRL to WMN to achieve the objectives.

In chapter 3: this chapter details of fDRL scheme, system models, how to apply FG and NLC to the WMN. Besides that, two novel path selection algorithms, NLPS and INLPS, are proposed. The formulas and algorithms of NLPS and INLPS which improved by Q-Learning and applying to the WMN for path selection are also described in detail to achieve the two objectives of this thesis and propose a possible solution to the research problem.

In chapter 4: this chapter shows the parameters and settings of the numerical simulations. Also, the results of simulations are included in this chapter. The simulation results are discussed in detail to show the performance gain of the research in terms of network capacity, computation time and some other performance matrices.

In chapter 5: this chapter is the conclusion of the thesis to summarize the research and is concluded with the advantages of the proposed algorithms. And then, the contributions and further works for additional investigation of future wireless communication are discussed in this chapter.

Chapter 2

Background

The reviews of the prerequisites fundamental knowledge for this research are evaluated in this chapter. First, a brief introduction about D2D network and WMN is explored and then the related works and principles of FG and NLC are illustrated. Next part is the basic knowledge of DRL. Application of each technology in WMN are also reviewed and discussed.

2.1 Device-to-Device Communication

D2D communication is a type of wireless communication technology that enable direct communication between the nearest wireless devices rather than through the infrastructure. With D2D communication, the data between a device pair can be transmit without going through the main network such as Access Points (APs) or Base Station (BS) as long as they are close. D2D communication is a concept for improving the device performance by allowing direct transmission between very close pairs of users. However, with the development of hardware technology, longer-distance D2D communication has also been gradually realized. As 5G promises more devices to be connected faster in a small cell, direct communication with the infrastructure mode of D2D communication become one of the essential technologies to support 5G wireless networks [25]. Therefore, current research trends have shown that D2D will be one of the technologies of the new next-generation mobile network.

Although D2D communication offers many benefits over LTE systems, there are several problems in terms of interference mitigation, device discovery and synchronization, mode selection, security, and Quality of Serves (QoS). To realize the potential of D2D communication, intensive research has been carried out by both academia and industry to address these issues. In the survey paper, [26], the authors categorize D2D communication based on spectrum reuse and provide the-state-of-art based on the classification in terms of performance metrics studied and conclude with the advantages and disadvantages of the spectrum sharing schemes, common assumptions and the maturity of D2D communication in the real world.

2.2 Wireless Multihop Network

A wireless network is a network that consists of several nodes that communicate via wireless channels. Depending on the architecture, wireless networks can be divided into two categories. Before the use cases with ad hoc paradigm, the traditional wireless system in the cellular paradigm is with the static infrastructure with Access Points (APs) and Base Station (BS). Two users require to go through the BS for communications in the infrastructure network. However, centralizing at the APs or BS in infrastructure mode cannot fulfil and have some demand to serve the increasing number of devices because of the long-distance communication.

A Wireless Multihop Network (WMN) is set of wirelessly connected nodes without an aid of centralized infrastructure that all the nodes process cooperatively and forward any packet via relaying nodes by multihop fashion. With the direct communications of the multihop fashion, D2D communication can extend the range of the transmissions. Using other nodes as the relay nodes, the packets can be transmitted from a source node to a destination node over longer distance. Thus, WMN not only can extend the network coverage, but also can achieve better network capacity by using the relay nodes strategy. Compared with traditional wireless networks, the WMN can use each user in the network as an AP to transmit data without a BS. This will greatly reduce the cost of building BS for the operating company. Figure 2.1 shows an example of a wireless multihop network.

Although the WMN has advantages such as decentralization, high network capacity, and long transmission distance, it also has many problems.

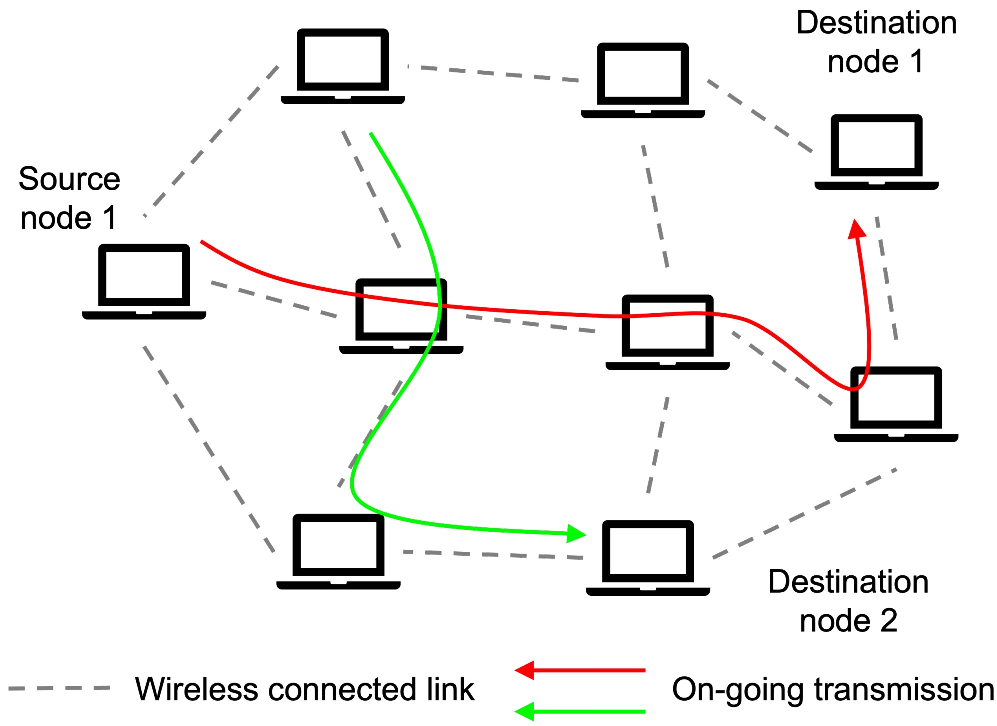


Figure 2.1: Example of a multihop wireless network

The first problem is that, because the increasing number of nodes in the network, there are many data transmission paths to be selected from the source nodes to the corresponding destination nodes by an efficient path selection algorithm. Since the criteria and conditions of each path are different, the results of path selection have great influence on the entire network capacity problem. Therefore, it is necessary to design a path selection algorithm for WMN that can effectively select path for each nodes and optimize network capacity, which is one of objectives of this research. Second problem is occurred when too many nodes send messages to a single node, in which it can lead to high latency for the sent message to reach its destination node. Besides that, the queuing time and processing time will increase drastically when when the number of nodes is large.

2.3 Airtime Link Metric

Airtime reflects the amount of channel resources consumed by transmitting the frame over a particular link and the extensively framework allows this metric to be overridden by any path selection metric as specified in the mesh profile. The IEEE802.11s mesh WLAN specification of Airtime Link Metric (ALM) [27] can capture the link quality as a function of the estimated frame loss probability as follows:

$$l = \left(O + \frac{B_t}{R_0}\right)\frac{1}{1 - e_f} \tag{2.1}$$

where l is airtime cost, O is channel access overhead, which includes frame headers, training sequences, access protocol frames, and so on. Here O equals to the sum of physical (PHY) header, MAC header, acknowledgement (ACK), distributed coordination function inter-frame space (DIFS), short inter-frame space (SIFS), slot time and minimum of contention window (CW$_{min}$). B_t is number of bits in test frame, R_0 is basic rate for test frame and e_f is Frame Error Rate (FER). ALM can use a numerical value to express the quality of the link. l is small means the quality of the link is better. For example, a node send test frames (8192 bits) through a link with data rate of 1 Mb/s. the channel access overhead is 65 μs, PHY header is 192 μs, ACK is 304 μs and slot time is 9 μs. The total time is 570 μs. This airtime and overhead value is converted to units of 0.01 TU (10.24 μs), i.e., 855.66 (rounded to 856). If the frame error rate is 80%, the airtime cost (l) is 4280. In this thesis, airtime cost is used by FG to identify an efficient radio-aware path.

2.4 Factor Graph and Sum-Product Algorithm

This section introduce the basic knowledge of factor graph and sum-product algorithm. The application of factor graph in the WMN also included in this section.

2.4.1 Factor Graph

A Factor Graph (FG) is a bipartite graph representing the factorization structure of a global function into a product of smaller local functions, each local function contains the product from other factor. FGs have two types of nodes:

- Variables, which can be either evidence variables when their value is known, or query variables when their value should be predicted;
- Factors, which define the relationships between variables in the graph and represent functions on subsets of the variables.

Figure 2.2 is an example of FG. From X_1 to X_4 is variables and from f_1 to f_3 is factors. Each factor can be connected to many variables and comes with a factor function to define the relationship between these variables. Each factor function has a weight associated with it, which describes how much influence the factor has on its variables in relative terms. In other words, the weight encodes the confidence we have in the relationship expressed by the factor function. If the weight is high and positive, we are very confident in the function that the factor encodes; if the weight is high and negative, we are confident that the function is incorrect. The weight can be learned from training data, or assigned manually.

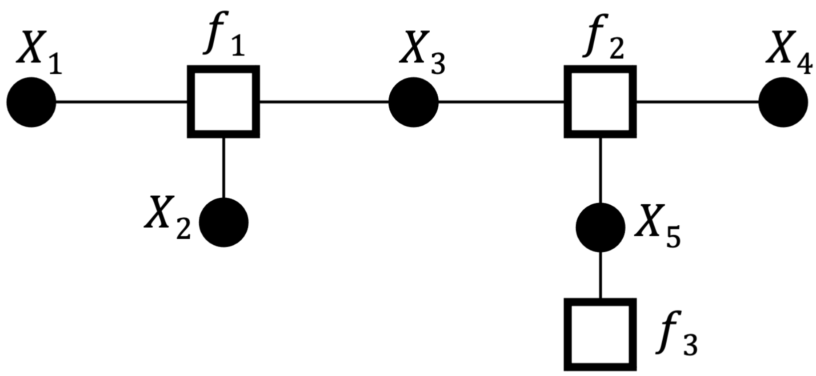


Figure 2.2: Example of a factor graph

There are three main advantages to using factor graphs when designing algorithms:

- FG can represent a wide variety of problems. By laying bare the compositional structure of the problem, they expose opportunities to improve computational performance.

- FG is beneficial in designing and thinking about modelling your problem, even aside from performance considerations.

Because many optimization problems in robotics have the locality property, FGs can model a wide variety of problems across AI and robotics.

2.4.2 Sum-Product Algorithm

Sum-Product (SP) is an algorithm to compute the global function of FG. According to the structure of FG, it calculates the local function at each factor and variable as the products of them, and then combines the local functions by multiplication and addition them together, Finally at the root factor/variable, the global function is obtained.

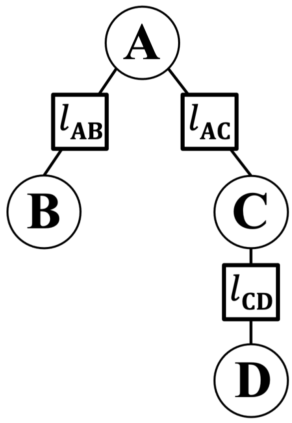


Figure 2.3: A modified FG for the product $l_{AB} + (l_{AC} \cdot l_{CD})$

In this thesis, to apply FG to WMN, SP is modified. Figure 2.3 is an example of modified FG. In this FG, each factor is represented as a node. All the nodes are equal to each other in WMN environments and the factor value of a node can be neglected. Each variable is only connected to two nodes, and its value is denoted as l, which is airtime cost and is calculated by ALM. For root node A, the relation between node C and node D is parent node and child node, so the product of node E is multipling them together, which is $l_{AC} \cdot l_{CD}$. Similarly, node B and node C are sibling nodes, so the product from them is sum them together, which is

$$F(A) = l_{AB} + (l_{AC} \cdot l_{CD}) \tag{2.2}$$

where $F(A)$ is the global function of root node A. The global function is small means good performance of this network structure. Compared with other algorithms, the advantage of the SP algorithm is that it considers the structure of the tree-based graphs to calculate the weights, which makes the global function better reflect the state of the network topology and is more suitable for the WMN environment.

2.5 Lattice Coding Theory

This section introduces the basic formulation of lattice coding theory [28] for an additive white Gaussian noise (AWGN) channel of the wireless communications.

2.5.1 Nested Lattice Code

A lattice Λ is an infinite structure with no power constraint [28]. A lattice code (LC) is a finite codebook designed to satisfy a power constraint. In other words, LC is an error-correcting code. The messages are represented as real numbers and transmit through lattice points. With the help of these points, the messages can be encoded and decoded correctly even some bits of codewords are changed by noise. The encode function of LC is given by

$$x = G \cdot b \tag{2.3}$$

where x is codeword, G is generator matrix, and b is messages (information integers). With different generator matrix, the codeword is different. A vector x is a lattice point if it can be formed as a linear combination of the basis vectors scaled by positive integers b. For decoding, the function is written as

$$\widehat{x} = \underset{\lambda \in \Lambda}{\operatorname{argmin}} \parallel y - \lambda \parallel^2 \tag{2.4}$$

and this equation is frequently expressed using quantizer function, $\widehat{x} = \mathcal{Q}_\Lambda(y)$. Lattice decoding tries to find the lattice point $\widehat{x} \in \Lambda$, which is closest to an arbitrary y in the Euclidean distance. The y is the received messages through the channel model.

Nested Lattice Code (NLC) is defined using two lattices: the coding lattice and the shaping lattice. The coding lattice gives the lattice code its error-correcting properties. Whereas, the shaping lattice enforces the power constrain for the lattice code. In this thesis, we use the NLC with 8-dimensional LC (E8 lattice code) and generator matrix G is

$$G = \begin{bmatrix} 1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1/2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1/2 & -1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1/2 & 0 & -1 & 1 & 0 & 0 & 0 & 0 \\ 1/2 & 0 & 0 & -1 & 1 & 0 & 0 & 0 \\ 1/2 & 0 & 0 & 0 & -1 & 1 & 0 & 0 \\ 1/2 & 0 & 0 & 0 & 0 & -1 & 1 & 0 \\ 1/2 & 0 & 0 & 0 & 0 & 0 & -1 & 2 \end{bmatrix}$$

NLC can control the transmit power level by mapping the original power at the lattice point of the shaping lattice to the constrained power at the lattice point of the coding lattice.

As shown in Figure 2.4, in this lattice example, Λ_c is the coding lattice and Λ_s is the shaping lattice. If a codeword is transmitted with the original power, the distance between this point and zero point is very far. It is highly recommended to be shaped back into the area of the shaping lattice Λ_s. With this, the transmit power is constrained and saved even though the lattice decoding will result the same. In NLC, the decoding function is different with normal LC, which is given by

$$x = G \cdot b - K \cdot \mathcal{Q}_{\Lambda_s}(\frac{G \cdot b}{K}) \tag{2.5}$$

Since we consider the NLC with 8-dimensional LC (E8 lattice code), i.e., $A_8/4A_8$, so the positive integer, $K = 4$ and b is chosen from 0 to $K - 1$ randomly.

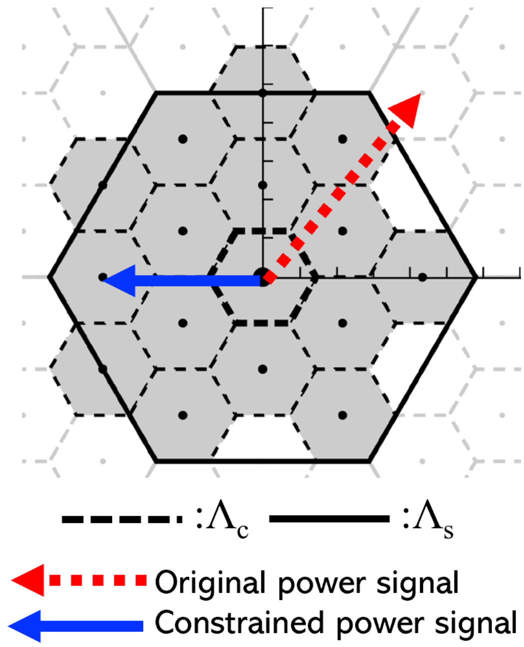


Figure 2.4: Illustration of nested lattice code

2.5.2 Compute-and-Forward Strategy

A compute-and-forward (CoF) strategy that enables relay node to decode linear equations of the transmitted message using the noisy linear combinations provided by the channel. In other words, CoF is used to reduce the number of time slots when two nodes send messages to one node at the same time. The codewords are combined linearly and sent though the channel. At receiver, those codewords can be decoded by lattices correctly even with the presence of noise. Figure 2.5 shows the system model of CoF strategy.

In the figure, $\mathbf{w}$ is messages and can be encoded to the corresponding codewords $\mathbf{x}$. Upon transmitted over an AWGN channel $\mathbf{H}$ with thermal noise $\mathbf{z}$, they become the receiving codewords $\mathbf{y}$, which are decoded to the receiving messages $\mathbf{u}$. Because CoF strategy relies on codes with a linear structure, NLC can be used in it [29]. In this thesis, we combine two $\mathbf{x}$ to one $\mathbf{y}$, which is shown in Figure 2.6. In particular, the NLC is used to reduce the error on the messages, whereas the CoF is used to reduce the number of time slots when transmitting messages.

In Figure 2.6, two codewords $\mathbf{x}_1$ and $\mathbf{x}_2$ are sent through channel and become $\mathbf{h}_1\mathbf{x}_1$ and $\mathbf{h}_2\mathbf{x}_2$ scaled by channel coefficients vectors $\mathbf{h}_1$, $\mathbf{h}_2$. Since

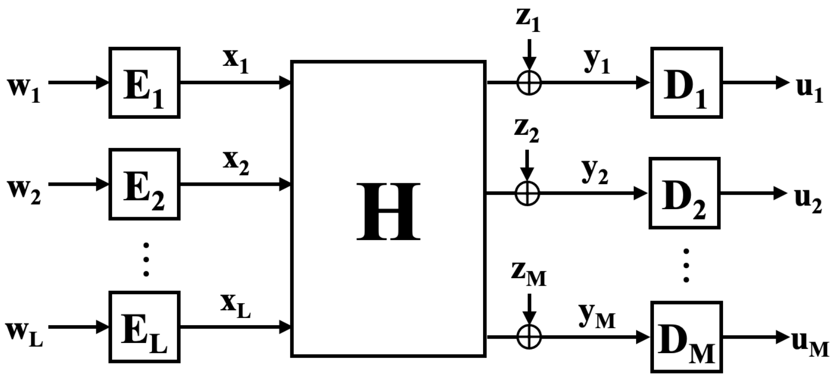


Figure 2.5: System model of L transmitters reliably communicate to M relay nodes over an AWGN channel

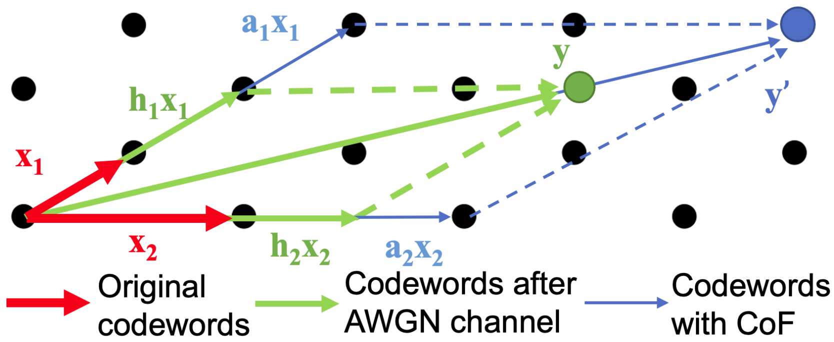


Figure 2.6: Linear combination of codewords on the lattice points

the linear combination $\mathbf{y}$ is not on a lattice point, to decode it correctly, $\mathbf{y}$ needs to be scaled by $\mathbf{a}_1$, $\mathbf{a}_2$ to a proper lattice point, which is $\mathbf{y}'$. Because the thermal noise is also scaled at the same time, it is important to find a suitable set of $\mathbf{a}_1$, $\mathbf{a}_2$. It can be calculated by

$$\mathbf{a} = \frac{P\mathbf{h}^{\mathbf{T}}\mathbf{a}}{1 + P \parallel \mathbf{h} \parallel^2}, \quad \parallel \mathbf{a} \parallel^2 < 1 + P \parallel \mathbf{h} \parallel^2 \tag{2.6}$$

where P is the transmit power. The decoder will decode correctly knowing $\mathbf{a}_1$, $\mathbf{a}_2$.

2.6 Deep Reinforcement Learning

This section introduces the basic knowledge of deep reinforcement learning, including principles, methods and practical applications in WMN.

2.6.1 Reinforcement Learning

Reinforcement Learning (RL) is a research field of machine learning in which an agent learns via trial and error. This problem is often modeled mathematically as a Markov decision process (MDP).

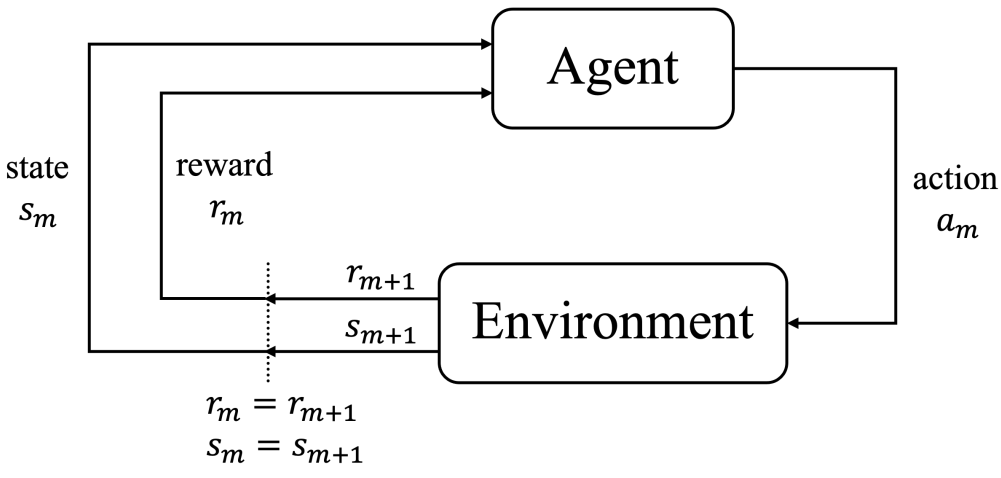


Figure 2.7: Diagram of the loop recurring in reinforcement learning algorithm

In Figure 2.7, RL algorithm contents several elements: agent, environment, state s, action a and reward r. The learning process of agent start by receiving the current state s_m from the environment. Agent is the main part of RL, which is AI and decides to do which action depending on the current state s_m and reward r_m . After the agent makes a decision, the action a_m he makes are sent to the environment. The environment calculate reward r_{m+1} this action a_m earned and sent it to the agent with next state s_{m+1}. Then agent do decision again based on this state and reward. With looping this algorithm continuously, the agent tries to find a near-optimal policy, the best action at state, by maximizing the expected cumulative reward.

2.6.2 Q-Learning

Q-learning that is known as an independent model of the RL algorithm is able to learn the action value for a particular state in the defined environment or system. The independent model of Q-learning does not rely on the model of the environment or system. Besides that, the model can not only deal with the problem in the stochastic conditions, but also its reward does not depend on the dynamically change of adaptations or adjustments. F. S. Melo [30] explained that Q-learning is used to obtain an optimal state by maximizing the expected value of the total reward over any and all successive iteration from the origin of the current state for any Finite Markov Decision Process (FMDP). In other words, Q-learning can search an optimal action as the best selection policy for any given FMDP with the given infinite exploration time and in the case of a fully-random policy or a partly-random policy. In the Q-learning, "Q" is a function that is also called Q-function. This Q-function computes the expected value of the instantaneous reward for an action to be taken in the given state. In general, the Q-function can be written as:

$$Q^{new}(s_m, a_m) = (1 - \alpha)Q(s_m, a_m) + \alpha(r_m + \gamma Q_{max}(s_{m+1}, a)) \tag{2.7}$$

where s_m and s_{m+1} are current state and next state, a_m and r_m are action and reward, α and γ are learning rate and discount factor. The learning rate determines to what extent newly acquired information overrides old information. A factor of 0 makes the agent learn nothing (exclusively exploiting prior knowledge), while a factor of 1 makes the agent consider only the most recent information (ignoring prior knowledge to explore possibilities). In fully deterministic environments, a learning rate $\alpha = 1$ is optimal. When the problem is stochastic, the algorithm converges under some technical conditions on the learning rate that require it to decrease to zero. The discount factor determines the importance of future rewards. A factor of 0 will make the agent shortsighted by only considering current rewards r_m, while a factor approaching 1 will make it strive for a long-term high reward. If the discount factor meets or exceeds 1, the action values may diverge. Starting with a lower discount factor and increasing it towards its final value accelerates learning [31].

In Q-function, updated Q-value $Q^{new}(s_m, a_m)$ is the sum of three factors:

- $(1-\alpha)Q(s_m, a_m)$: the current value (weighted by one minus the learning rate)
- αr_m: the reward $r_m = r(s_m, a_m)$ to obtain if action a_m is taken when in state s_m (weighted by learning rate)
- $\alpha\gamma Q_{max}(s_{m+1}, a)$: the maximum reward that can be obtained from the next state s_{m+1} (weighted by learning rate and discount factor)

The Q-values form the Q-table and are updated with each training. Agent decided the next action according to the Q-values in the Q-table.

There are generally two ways to break out of the loop of Q-learning and terminate training. One is to preset a target reward value. When the agent's reward reaches this value, it means that the training is completed and the training is terminated. But it is difficult to set an appropriate reward value when the training results are completely unpredictable. Another way is to set a threshold. When the reward obtained by the agent changes less than the threshold within several iterations, the training is terminated, and the policy at this time is regarded as a near-optimal one. This approach maximizes reward and leads to better results and it is used in this thesis.

Q-learning is widely used in path selection problems. In the path selection problem, users need to choose the next path according to their own situation until they reach the destination, which is very consistent with the operation method of Q-learning. In WMN, the selection of multihop relay nodes from the source node to the destination node is a path selection problem. Here each node acts as an agent and action refer to the next selected multihop relay node. In each selection of multihop, nodes need to make choices based on their own network environment and get reward according to the choices they make to learn and optimize the policy, which is to maximize the objective parameters of WMN. The Q-learning algorithm used in this thesis will be discussed in detail in Chapter 3.

2.6.3 Deep Learning

Deep learning is a machine learning technique that teaches computers to do what comes naturally to humans: learn by example. Deep learning is a key technology behind driverless cars, enabling them to recognize a stop sign, or to distinguish a pedestrian from a lamppost. It is the key to voice control in consumer devices like phones, tablets, TVs, and hands-free speakers. Deep learning is getting lots of attention lately and for good reason. It's achieving results that were not possible before.

In deep learning, a computer model learns to perform classification tasks directly from images, text, or sound. Deep learning models can achieve state-of-the-art accuracy, sometimes exceeding human-level performance. Models are trained by using a large set of labeled data and neural network architectures that contain many layers.

Most deep learning methods use neural network architectures, which is why deep learning models are often referred to as deep neural networks. Figure 2.8 shows a neural network. The term "deep" usually refers to the number of hidden layers in the neural network. Traditional neural networks only contain 2-3 hidden layers, while deep networks can have as many as 150. Deep learning models are trained by using large sets of labeled data and neural network architectures that learn features directly from the data without the need for manual feature extraction.

2.7 Summary

This chapter has described the fundamental research background of wireless communications, including Device-to-Device (D2D) and Wireless Multihop Network (WMN). Besides that, the explanation of ALM and its calculation are elaborated. Furthermore, the basic knowledge of three used techniques in this thesis, i.e., Factor Graph (FG), Nested Lattice Code (NLC) and rein-

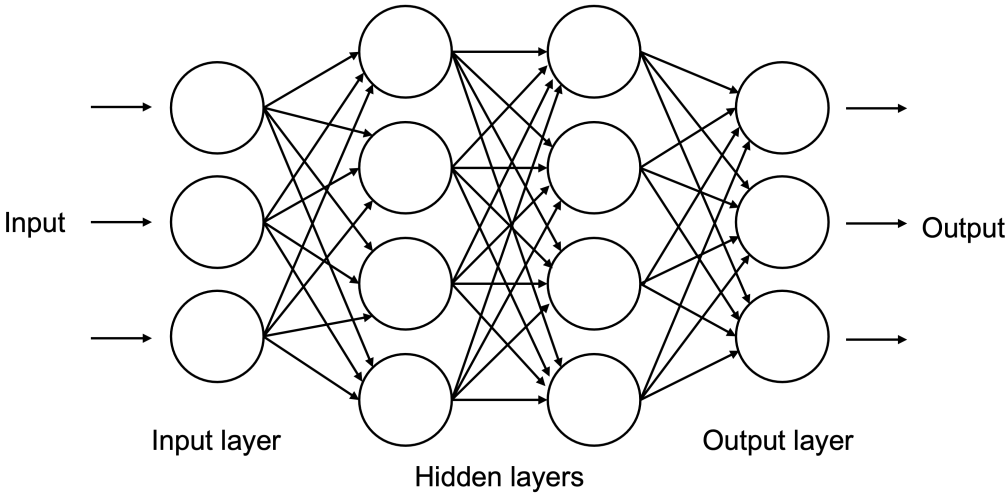


Figure 2.8: Neural network

forcement learning to the WMN environment in order to solve the problem statements, which has been presented in the previous chapter.

Chapter 3

Proposed Path Selection Schemes

This chapter mainly introduces Factor Graph-based Deep Reinforcement Learning (fDRL) scheme with two novel path selection algorithms proposed in this thesis: SNR-based Learning Path Selection (NLPS) algorithm and SINR-based Learning Path Selection (INLPS) algorithm, including Principles, formulas, algorithms and other details. Besides that, the system model used in this thesis is also described.

3.1 System Model

The system model of the research methodology is described in this section, including network model, channel model, interference model and link capacity model.

3.1.1 Network Model

In this thesis, the Wireless Multihop Network (WMN) is considered as network model. Device-to-Device (D2D) communication is applied, which means user in wmn works as an Access Points (APs) to connect to each other with wireless links and transmits data to other users directly without using Base Station (BS). Here we assume that the network topology of a WMN can be modelled as a graph network model. In this model, all nodes have same transmit power, bandwidth, antenna gain and received signal strength indicator

(RSSI). Also, nodes are stationary and each node must have a connection to other node(s). For transmitting data, a node can send to or receive from only one other node in a single time slot. An example of WMN model used in this thesis is shown in Figure 3.1 of 6 nodes with full-duplex node and half-duplex node. In this thesis, we assume that all nodes are working in full-duplex mode. Among these nodes, one node will be selected as the root node and connected to the Ethernet. Other nodes link with the root node using multihops and access the Ethernet.

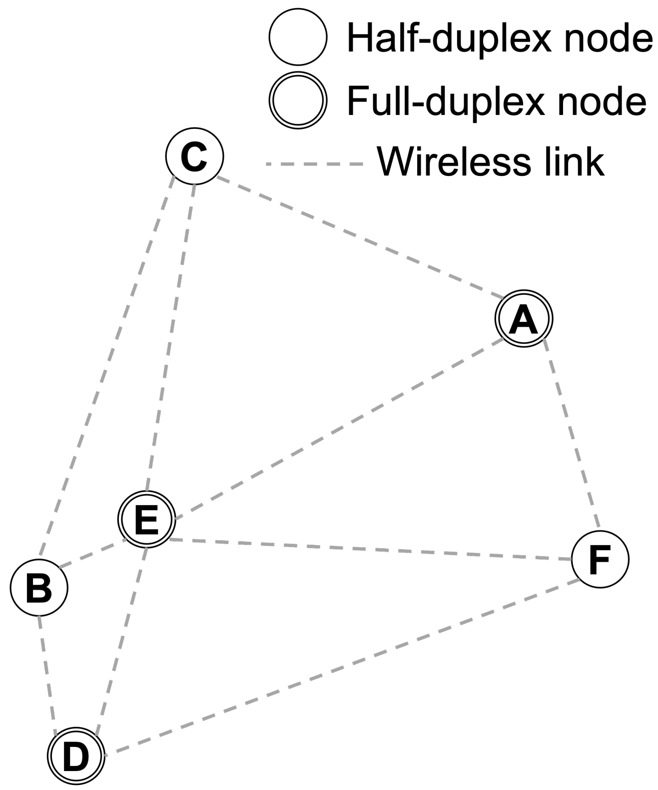


Figure 3.1: An example of WMN model

3.1.2 Channel Model

The network topology influences the network capacity. In this thesis, the wireless nodes are random uniformly distributed in the coverage area and the distance between two wireless nodes $i(x_i, y_i)$ and $j(x_j, y_j)$ is computed for the use of Physical Layer (PHY) model as introduced below.

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \tag{3.1}$$

The transmit power of each node is fixed in the network model of this thesis, therefore the received power at the receiving node j based on the distance between two nodes and the shadowing result from objections. By consideration, the wall attenuation among two nodes, W_{ij}, the shadowing attenuation from objections, X_σ, and the channel gain between two nodes is considered on the Log-distance Fading model with the following pathloss:

$$PL_{ij} = PL_0 + 10 \cdot \beta \cdot \log_{10}(\frac{d_{ij}}{d_0}) - W_{ij} + X_\sigma \tag{3.2}$$

where PL_0 is assumed as Friis free space model, $PL_0 = 20 \cdot \log_{10}(d_0)$, β is attenuation constant or pathloss exponent and d_0 is decorrelation distance by considering instantaneous fading as a Gaussian random variable (X_σ) with zero mean and standard deviation of σ and W_{ij} is the wall attenuation.

The power ratio at the receiving node j with the signal attenuation between wireless node i and node j according to the pathloss PL_{ij} is

$$G_{ij} = \frac{1}{10^{\left(\frac{PL_{ij}}{10}\right)}} \tag{3.3}$$

$G_{ij} \cdot P_i$ is the receive power at receiving nodes

3.1.3 Interference Model

After computing the channel gain, the Signal-to-Interference-plus-Noise Ratio (SINR) of link can be obtained. SINR is calculated based on the receive power, interference during transmission and noise in the channel.

The interference model in this thesis is defined as the ratio between the receiving signal power of the on-going transmission to the total interference power of all other active transmitting nodes and thermal noise of the receiving system, which manifest itself in the calculation of SINR. Figure 3.2 depicts the illustration of signal to interference and noise ratio.

Node i is sending messages to node j. At the same time, node 1 to k are also sending messages to other nodes besides node i and node j, where K will be the total number of active interfering nodes in the network of WMN.

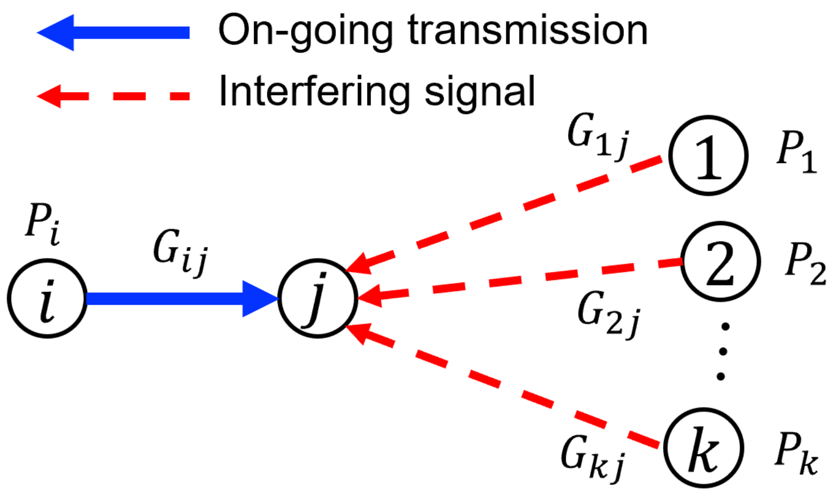


Figure 3.2: Illustration of signal to interference and noise ratio

The SINR at the receiving node j with the transmit power P_i is

$$SINR_{ij} = \frac{G_{ij} \cdot P_i}{Noise + Interference} \tag{3.4}$$

and

$$Noise = \eta_j \cdot B \tag{3.5}$$

where η_j is noise level at receiving node j. In this thesis, all nodes are in the same channel and the bandwidth and noise level is the same. The noise is defined as Additive White Gaussian Noise (AWGN). B is the bandwidth of the channel.

The interference power is

$$Interference = \sum_{k \in K, k \neq i} G_{kj} \cdot P_k \tag{3.6}$$

where k denotes the interfering nodes and P is the transmit power.

In addition, the Signal-to-Noise Ratio (SNR) in this thesis is defined as the ratio of receive power and noise, which is

$$SNR_{ij} = \frac{G_{ij} \cdot P_i}{\eta_j \cdot B} \tag{3.7}$$

3.1.4 Network Capacity Model

The link rate (bps) between sending node i and receiving node j is computed under the level of SINR with AWGN channel model by applying Shannon's capacity theorem.

$$R_{ij} = B \cdot \log_2 \left(1 + SINR_{ij}\right) \tag{3.8}$$

The network capacity is computed based on the link rate of on-going transmission and time slot in the networks. For each link, the transmission time is the size of data packet L divided by link rate, which is

$$\tau_i = \frac{L}{R_i} \tag{3.9}$$

A required time slot (T_v) is defined as the maximum time for data transmission of the number of active transmitting nodes at vth time slot, which is

$$T_v = \max_{i \in I} \{\tau_i\} \tag{3.10}$$

where v is the number of time slots and I is the set of nodes sending data packets in this time slot.

Network capacity (Ca) is defined as the total number of sent packets from all the nodes to a root node is divided by the total of required time slots, which is

$$Ca = \frac{(N-1) \cdot L}{\sum\limits_{v=1}^{V} T_v} \tag{3.11}$$

where N is total number of nodes in WMN and V is total of required time slots. We assume that each node has only one data packet to send to the destination node in this thesis.

The E2E throughput U from the source node S to the destination node D is defined as the sum of the link rates of all links on the path, which is

$$U = \sum_{m=0}^{M} R_{n^m \; n^{m+1}}, n = \{1, 2, \cdots, N, n \in S, n \in D\} \tag{3.12}$$

3.2 Proposed FG-based DRL Scheme

Factor Graph-based Deep Reinforcement Learning (fDRL) scheme proposed in this thesis is shown in Figure 3.3. It consists of five parts. DRL core combines with Factor Graph (FG), Learning Path Selection (LPS) algorithm, and Nested Lattice Code (NLC). In FG, Shortest Path Spanning Tree (SPST) is used to transfer WMN to a tree-based network topology and sum-product algorithm calculates and selects best root node. Next, two LPS algorithm based on Q-learning, which are NLPS and INLPS algorithm are proposed. NLPS is modified from [24] for multiple sources (including relay nodes) to single destination and INLPS is proposed by me. At last, Compute-and-Forward (CoF) with NLC is applied to reduce time slots in transmitting phase. The constraint of fDRL scheme is that it can be only applied to tree network topology.

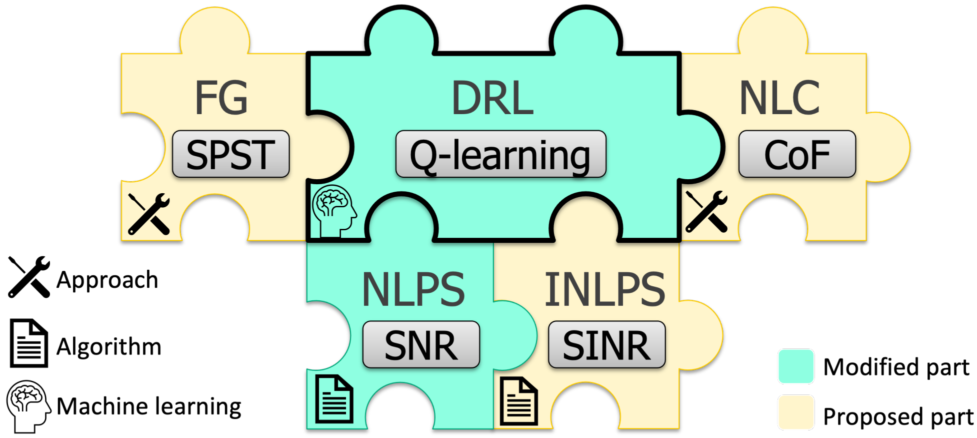


Figure 3.3: Proposed fDRL scheme

3.3 Factor Graph Approach

A Factor Graph (FG) is a bipartite graph representing the factorization structure of a global function into a product of smaller local functions, each local function contains the product from other factor. The global function can represent the whole FG. Using the sum-product algorithm, the global function can represent the whole FG, which can determine the best network capacity from a tree-based structure topology.

In the NLPS algorithm, the role of FG is to prepossess the nodes in the WMN, and filter out the root nodes which is most likely to generate high network capacity for the NLPS algorithm, thereby reducing the number of iterations amount of the NLPS algorithm.

FG prepossesses the nodes through the following three steps:

1) Use Dijkstra's algorithm to find the SPST;

2) Calculate total link weight of SPSTs;

3) Compare the total link weight and Select the best-metric path.

In step 1, the nodes and links in WMN can be represented into the undirected graph. As shown in Figure 3.4, for each node in WMN as the root node, Dijkstra's algorithm is used to find the path from root node to all other nodes. Dijkstra's algorithm is an algorithm for finding the shortest paths between nodes in a graph. If the link is selected by the Dijkstra's algorithm, it becomes logical connected link with a weight calculated by ALM. Whit this algorithm, the shortest path from the root node to all other nodes is selected to form a tree-based topology. This topology is called Shortest Path Spanning Tree (SPST), which has minimum weight paths from root node to all the other network nodes. Each node in WMN can be used as the root node and generate its own SPST, so the number of nodes is consistent with the number of SPSTs.

Next, in step 2, The total link weight of each SPST is calculated. The weight of each link is represented by airtime cost, which is calculated by ALM and the lower the value, the better the quality of the link.

In general, the total link weight is obtained by adding the weight of each link and this is also done in the Dijkstra's algorithm. But this way is difficult to distinguish which SPST is better. Besides that, in the actual network environment, the structure of the network will also have a great impact on the performance of the entire network. Therefore, simple summation method cannot accurately and completely represent the quality of the network. Instead of this method, in this papaer we regard the SPST as a FG, and use the sum-product algorithm to calculate its total link weight. The sum-product algorithm performs addition and multiplication according to the relationship between nodes, therefore the total link weight is also affected by the network topology. Compared to simple summation method, sum-product algorithm can better distinguish SPSTs.

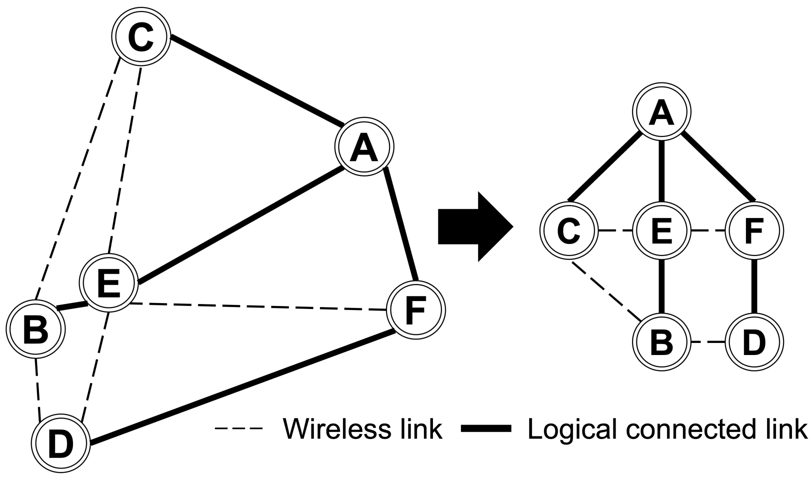


Figure 3.4: SPST computation for node A by Dijkstra's algorithm

Figure 3.5 is applying FG and the sum-product algorithm on SPST of node A is shown. l is ALM cost of the link and can be calculated by ALM. In FG, we assume all the nodes are equal in the WMN environment and the factor value of a node can be neglected. For root node A, the relation between node E and node B is parent node and child node, so the product of node E is multiplied them together, which is $l_{AE} \cdot l_{EB}$. Similarly, the product of node F is $l_{AF} \cdot l_{FD}$. Node C, node E, and node F are sibling nodes, so the product from them is summing them together, which is

$$F(A) = l_{AC} + (l_{AE} \cdot l_{EB}) + (l_{AF} \cdot l_{FD}) \tag{3.13}$$

where $F(A)$ is the global function of root node A. This global fucntion represents to the totak link weight. The total link weight is small means good performance of this network topology.

At last, in step 3, After the link weights of each node are calculated by sum-product algorithm, these SPSTs are compared to each other with their own total link weights and the smallest one is selected as the best one. As shown in Figure 3.6, by comparing the total link weight of SPST for different root nodes, the best-metric path is selected.

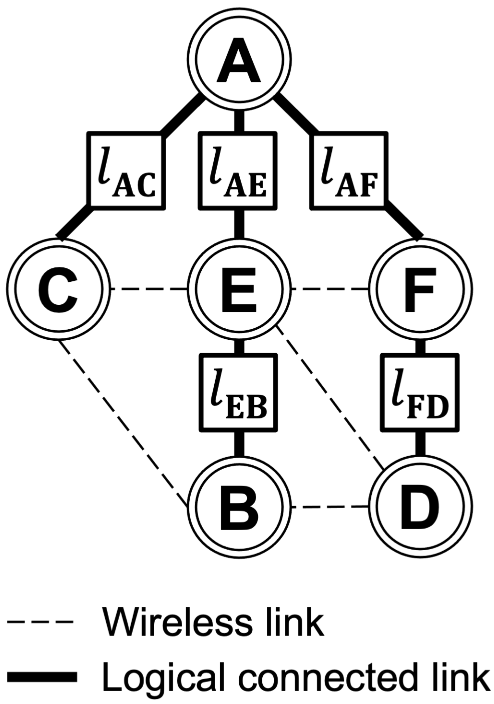


Figure 3.5: Applying FG and sum-product algorithm on SPST of node A

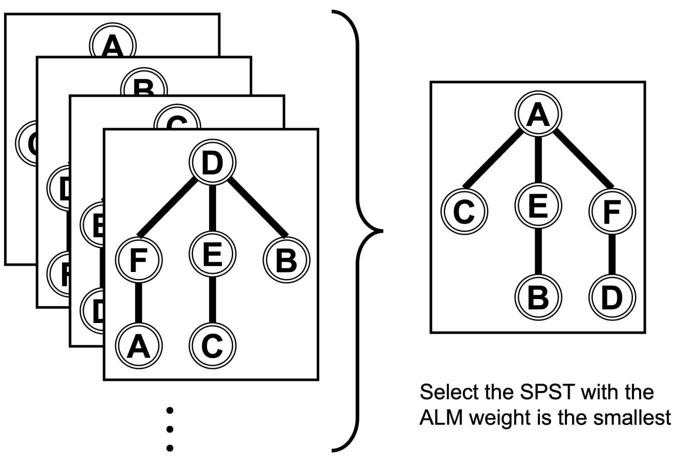


Figure 3.6: Best-metric path selection of a SPST

With FG, the best network topology is selected more efficiently and this kind of routing structure can optimize the overall network capacity. The root node of this best network topology is passed to the NLPS algorithm as the preprocessing result of FG and is used as the root node in the NLPS algorithm for learning phase. The algorithm and flowchart of FG is shown in Algorithm 1 and Figure 3.7

Algorithm 1 Factor Graph Algorithm

Definition: l is airtime cost, F is global function, ALM is airtime link metric
Input: Position of nodes
Output: Root node D

1: Calculate l of each link using ALM (2.1)
2: **for** each node n **do**
3: Set $D = n$
4: Find the SPST of n by applying Dijkstra's algorithm with l as weight
5: Calculate F of SPST using sum-product algorithm
6: **end for**
7: Select the best F as output root node D

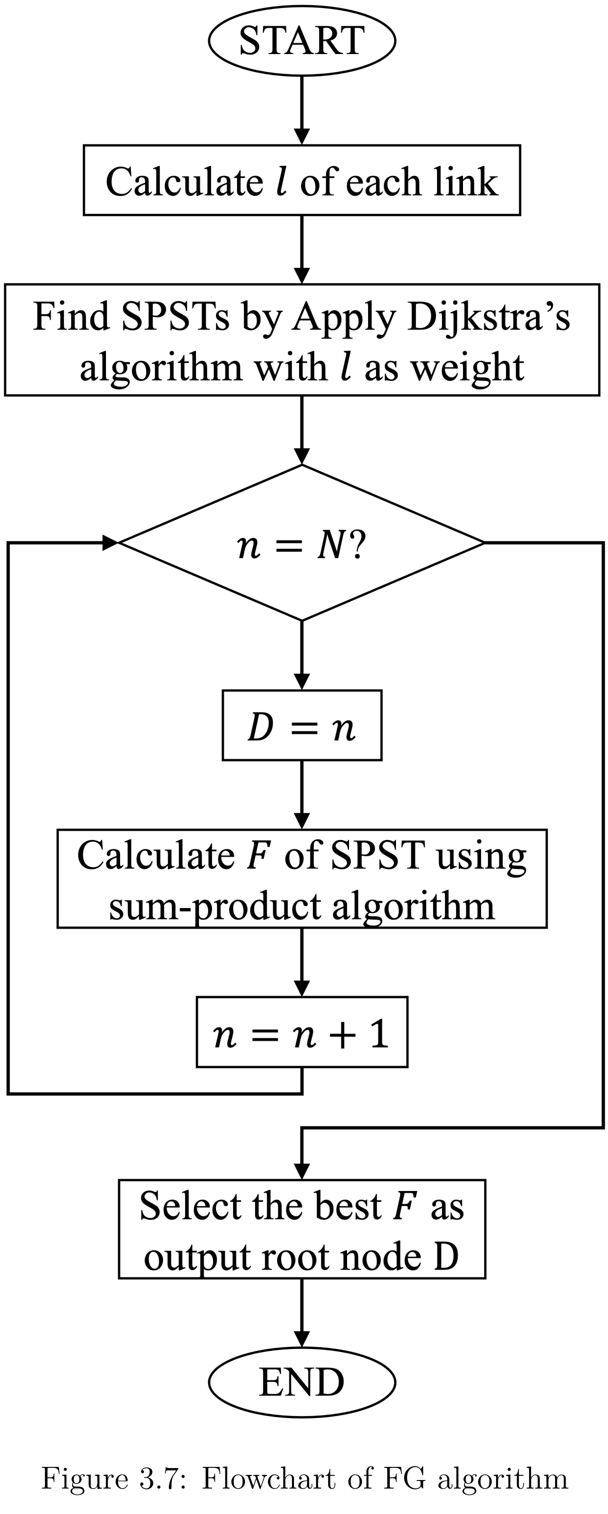


Figure 3.7: Flowchart of FG algorithm

3.4 SNR-Based Learning Path Selection Algorithm

SNR-based Learning Path Selection (NLPS) algorithm is a derivative algorithm based on Q-learning. Same as Q-learning, the objective of NLPS algorithm is to maximize the reward. In NLPS, the reward r for using the link between node i and node j is equal to the SNR of this link, which is

$$r_{ij} = SNR_{ij} \tag{3.14}$$

Since SNR is used, not SINR, as reward, NLPS algorithm will look for the path with the highest SNR and ignore the effect of interference between links. Having the highest SNR on this path means having the highest E2E rate. When the network topology is initialized, it is not yet possible to determine the links for transmission, the interference and SINR cannot be calculated. Therefore, the NLPS algorithm has practical significance in establishing network topology.

To apply NLPS algorithm, the nodes in WMN need to be layered according to the root node provided by FG and transmission range calculated by RSSI. As shown in Figure 3.8, all nodes are divided into M layers. m is the number of layers and k_m is the number of nodes in the mth layer. C_m means the mth layer and $n_{k_m}^{m}$ means the kth node in mth layer. In layer m, the total number of nodes is K_m. Nodes are layered by transmission range. According to the system model of this thesis, all nodes send messages to the root node D, forming to a tree-based topology. Nodes can only send data to nodes in adjacent layers. They cannot send across layers or to nodes in the same layer.

Next, NLPS algorithm is applied to select path. The algorithm is composed of three phases: initialization, training, and selection.

3.4.1 Initialization

During initialization phase, reward tables is generated by $K_{m-1} \times K_m$ first. Each pair of adjacent layers has one reward table containing the rewards of all possible pairs of state and action, which is sending nodes and receiving nodes. Because in NLPS algorithm, the reward is equal to SNR, these reward tables

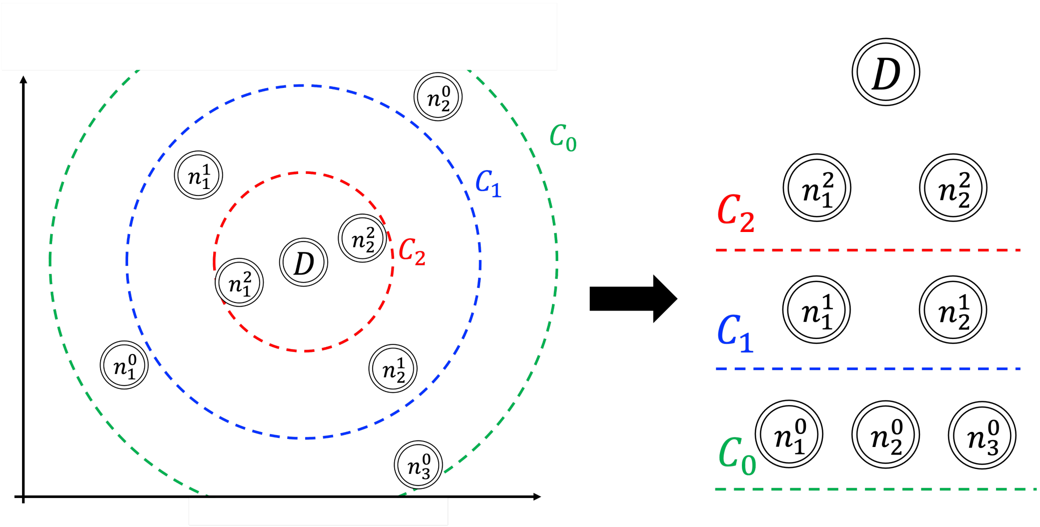


Figure 3.8: Nodes layered by transmission range in WMN

can be obtained before the training phase. The reward table is described by Table 3.1. In this table, m is the layer of receiving nodes and s_m and a_m means states and actions, respectively. The position of the nodes has been determined, so the SNR will not change again, and the reward table does not need to be updated with training.

Table 3.1: Reward table for mth layer

$s_m \backslash a_m$	n_1^m	n_2^m	...	$n_{K_m}^m$
n_1^{m-1}	r_{11}^m	r_{12}^m	...	$r_{1K_m}^m$
n_2^{m-1}	r_{21}^m	r_{22}^m	...	$r_{2K_m}^m$
...	...	...	...	...
$n_{K_{m-1}}^{m-1}$	$r_{K_{m-1}1}^m$	$r_{K_{m-1}2}^m$	...	$r_{K_{m-1}K_m}^m$

Same as reward table, each pair of adjacent layers has one Q-table. Q-table stores the Q-value of all possible pairs of state and action and will be updated after each iteration in training phase. As for the initialization of Q-table, we set every Q-values in every Q-tables is 0 to let agent select its first action randomly, which is fair to every node. The Q-table is shown in Table 3.2. Here $Q_m(i, j)$ is Q-value of node i and node j in the mth layer(C_m). For example, $Q_m(n_1^{m-1}, n_2^m)$ means the Q-value of the first node in C_{m-1} and the second node in the mth layer, and this Q-value is stored in the Q-table of the mth layer.

Table 3.2: Q-table for mth layer

$s_m \backslash a_m$	n_1^m	n_2^m	...	$n_{K_m}^m$
n_1^{m-1}	$Q_m(n_1^{m-1}, n_1^m)$	$Q_m(n_1^{m-1}, n_2^m)$	...	$Q_m(n_1^{m-1}, n_{K_m}^m)$
n_2^{m-1}	$Q_m(n_2^{m-1}, n_1^m)$	$Q_m(n_2^{m-1}, n_2^m)$	...	$Q_m(n_2^{m-1}, n_{K_m}^m)$
...	...	...	...	...
$n_{K_{m-1}}^{m-1}$	$Q_m(n_{K_{m-1}}^{m-1}, n_1^m)$	$Q_m(n_{K_{m-1}}^{m-1}, n_2^m)$	...	$Q_m(n_{K_{m-1}}^{m-1}, n_{K_m}^m)$

3.4.2 Training

The main task of the training phase is to update the Q-value according to the Q-function until the agents jump out from the training loop, which means the best path is find out. Q-function is the key part of training phase, which calculates the Q-value depend on the current values and future values. The standard Q-function (2.7) only considers the current reward, i.e. SNR of current hop. This will cause agents to only focus on the reward of current hop, while ignoring the status of the entire path. This is similar to the greedy search, which is inefficient. If the SNR of current hop is very large but after this hop, the SNRs of all multihops are small, the agent will still choose this hop, resulting a very bad E2E rate. Therefore, the Q-function should consider all hops. However, it leads to the heavy computation, even it has been reduced by FG. To get better result, we make a compromise choice. In this thesis, we make the Q-function to select the larger one between the reward of current hop and next hop. This makes the agents to consider the reward it will get further. The modelled Q-function is given by

$$Q_m(s_m, a_m) = \begin{cases} (1-\alpha)Q_m(s_m, a_m) & \\ \quad +\alpha(r_m(s_m, a_m) + \gamma Q_{max}(m+1)), & m \leq M, r_m(s_m, a_m) > r_{m+1} \\ (1-\alpha)Q_m(s_m, a_m) & \\ \quad +\alpha(r_{m+1} + \gamma Q_{max}(m+1)), & m \leq M, r_m(s_m, a_m) < r_{m+1} \\ r_m(s_m, a_m), & m = M+1 \end{cases} \tag{3.15}$$

where

s_m: state represents the relay node in C_{m-1} that want to send data packet to a node in C_m.

a_m: action represents the relay node in C_m selected by agent to send data packet. Possible actions for s_m include all possible relay nodes in C_m.

$r_m(s_m, a_m)$: reward represents the evaluation of choosing a link to send data packet. Each link has its own reward, which reflects it is good or bad to do this action in this state. In NLPS algorithm, reward is equal to the SNR.

α: The learning rate determines the proportion of current hop's Q-value and next hop's Q-value. α is large means that agent pays more attention to the Q-value of the next hop and reward. In this thesis, $0 \leq \alpha \leq 1$

γ: discount factor determines the weight of the next hop's Q-value. γ is large mean that agent pays more attention to the Q-value of the next hop.

r_{m+1}: reward of next hop in C_{m+1}. r_{m+1} is calculated by

$$r_{m+1} = r_{m+1}(a_m, a'_{m+1}) \tag{3.16}$$

where

$$a'_{m+1} = \underset{a_{m+1}}{\operatorname{argmax}}\ Q_{m+1}(a_m, a_{m+1}) \tag{3.17}$$

As in equation (3.17), a'_{m+1} is the best action at the state a_m in C_{m+1} with maximum Q-value. As the reward of best action in next hop, r_{m+1} is compared with the reward of current hop $r_m(s_m, a_m)$ and larger one is used in the update of Q-value. Through this method, users can not only be limited to the reward of current hop, but also take a long-term view and consider the next step to choose the path.

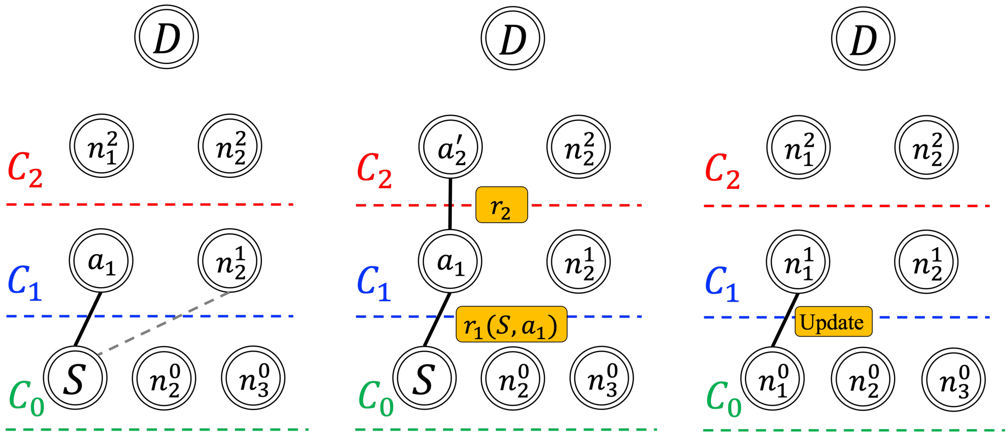
$Q_{max}(m+1)$: the maximum Q-value in C_{m+1} and is calculated by

$$Q_{max}(m+1) = Q_{m+1}(a_m, a'_{m+1}) \tag{3.18}$$

$Q_{max}(m+1)$ represents the Q-value of the best action in C_{m+1} and participate in the updating of Q-value as a new information.

Here we use the example in Figure 3.8 to explain how the Q-function works in training phase. As shown in Figure 3.9, n_1^0 is set as a source node S. S needs to send data packets to destination node (root node) D. $s_1 = S$, $m = 1$. First, a_1 is randomly selected from C_1. Here we assume that node n_1^1 is selected. $a_1 = n_1^1$. Next, SNR of this link is calculated by equation (3.7) as the reward $r1(S, a_1)$ of doing the action a_1. From the Q-table of C_1 and

C_2, we can find the best action at a_1 by equation (3.17) and assume that $a'_2 = n^2_1$. The SNR of this link is calculated as r_2. Also, $Q_{max}(2)$ is calculated by equation (3.18). the rewards of current hop $r_1(S, a_1)$ and next hop r_2 is compared to each other to decide which equation will be used to update in equation (3.15). At last, the Q-value of n^0_1 and n^1_1 is updated and stored in the Q-table. Then, for next layer C_2, action node is set as state, $s_2 = n^1_1$,



(a) Random selection (b) Reward comparison (c) Update Q-value

Figure 3.9: Steps of updating $Q_1(n^0_1, n^1_1)$

$m = 2$, and repeat the same step as C_1, and Q-value in Q-table of C_2 is updated. This training phase is repeat for each layer until $m = M+1$, which means it have reached to the destination node D and one iteration is finished. If the changes of E2E rate from S to D is lower than threshold ϵ, the training is stopped and completed. Otherwise, another iteration of training begins.

As the number of iterations increases, the Q-table is gradually updated. When the training is completed, each layer has its own complete Q-table, including the updated Q-value between all nodes in the adjacent two layers.

3.4.3 Selection

Agent needs to find the best path from S to D through Q-tables after the training of all layers in WMN is completed. First, set $s_1 = S$, C_1 searches all Q-values in the row of s_1 in the Q-table of C_1 and finds the maximum Q-value and select it as the best relay node a'_2 from s_1. Next, set $s_2 = a'_2$,

C_2 repeat the step above to find best relay node. After all layers find their own best relay node, the best multihop path from S to D is built.

3.4.4 Network Topology Formation

After completing the three phases of initialization, training and selection, the best multihop path of one node is selected. Except the root node, all other nodes have data packets that need to be sent to the root node according to the system model. Start with the furthest layer, each node applies these three phases to find its path as source node. Combining all best path of all nodes, a tree-based network topology with highest network capacity in WMN is established, which is shown in Figure 3.9.

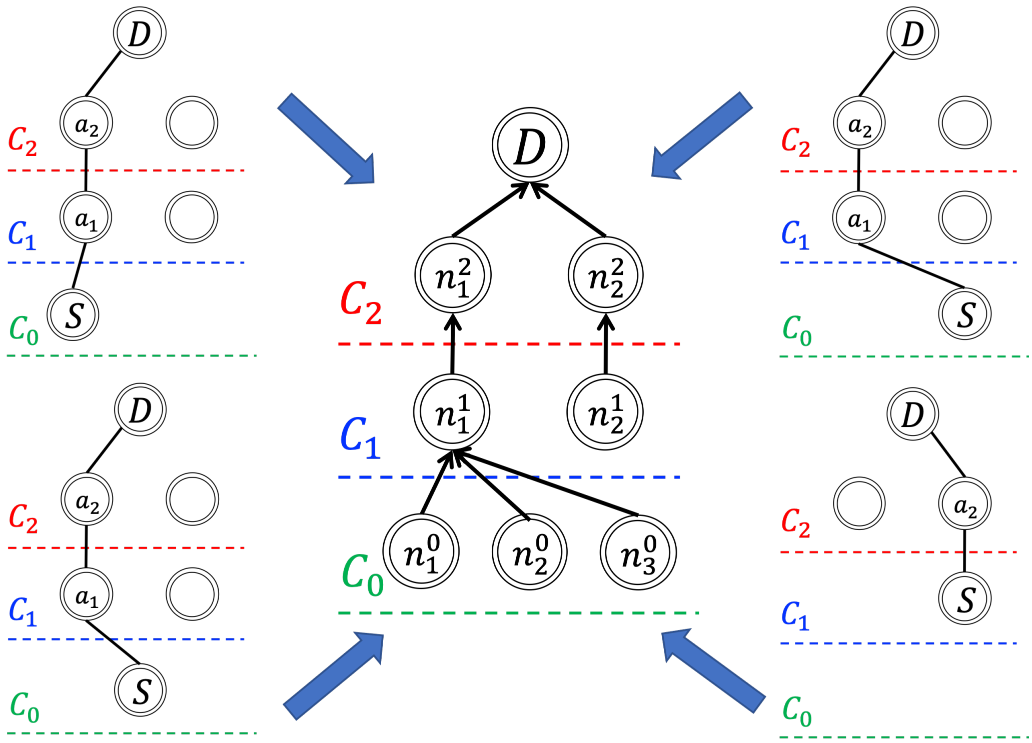


Figure 3.10: Forming the best network topology

The algorithm and flowchart of NLPS is shown in Algorithm 2 and Figure 3.11.

Algorithm 2 SNR-Based Learning Path Selection Algorithm

Definition: N is total number of nodes, t is number of iterations, m is the number of layers, k is the number of nodes in the corresponding layer, U is E2E throughput

Input: Position of nodes, root node D

Output: Topology information of nodes

Determine m of nodes based on the transmission range

Phase 1. Initialization

Set all Q-values in Q-tables to zero
Calculate SNR_{ij} of each link and store to reward table

Phase 2. Training

Set S is marked as visited node
while all the relay nodes are not visited **do**
  Set $n_{k_{m-1}}^{m-1} = S$, $U_0 = 0$, $t = 1$
  while true **do**
    for $m = m : M$ **do**
      $s_m = n_{k_{m-1}}^{m-1}$
      s_m randomly selects a relay node as a_m from C_m
      Calculate r_{m+1} from equation (3.16)
      Calculate $Q_{max}(m+1)$ equation (3.18)
      Update $Q_m(s_m, a_m)$ equation (3.15)
      $s_{m+1} = a'_m$ as the best relay node
    end for
    Calculate U_t
    if $|U_t - U_{t-1}| \leq \epsilon$ **then**
      break
    else
      $t = t + 1$
    end if
  end while
  if S has visited all the relay nodes **then**
    break
  else

Select another relay node as S

end if

end while

Phase 3. Selection

for $m = m : M$ **do**

for $k_{m-1} = 1 : K_{m-1}$ **do**

Select a'_m in the Q-table of C_m

end for

end for

Phase 4. Network Topology Formation

Combine all paths into a tree-based topology

Broadcast topology information to all nodes

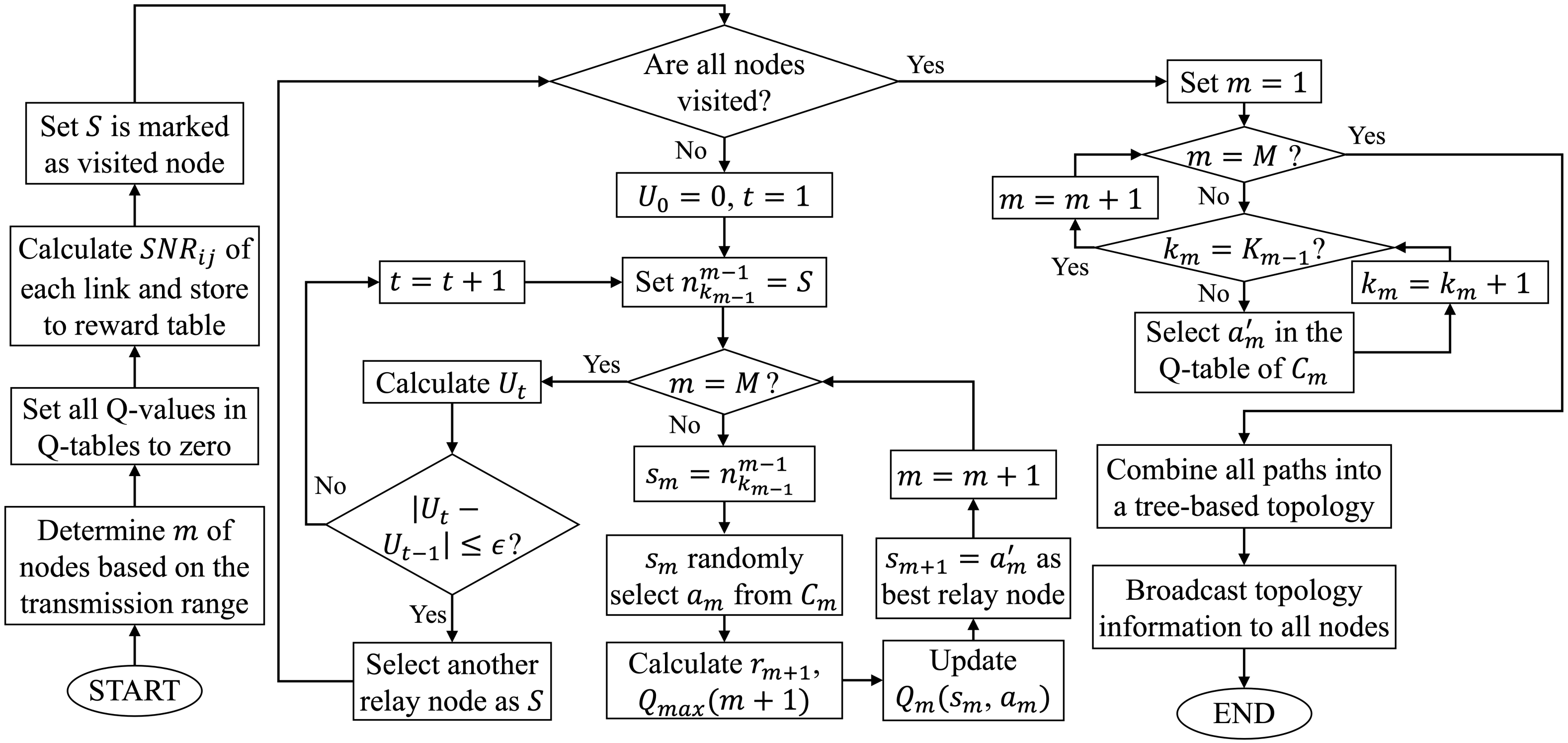


Figure 3.11: Flowchart of NLPS algorithm

3.5 SINR-Based Learning Path Selection Algorithm

In NLPS algorithm, interference between links is ignored. This is because it is impossible to determine which links will be used to transmit data packets without planning the transmission paths of each node first. According to the interference model in this thesis, the specific value of the interference cannot be obtained. Besides that, when planning the transmission path for the first time, ignoring interference can allow nodes to find their own best path more efficiently.

However, in the real world, interference has a great influence on the transmission rate of the link and it has to be taken into account when calculating the transfer rate. Therefore, it is necessary and fit the reality to incorporate interference into the path selection algorithm.

To find a high throughput path and increase the network capacity, we propose SINR-based Learning Path Selection (INLPS) algorithm. The INLPS algorithm is improved based on the NLPS algorithm. The goal of INLPS algorithm is to find the best path with high E2E throughput for each node. INLPS algorithm is applied after NLPS algorithm. A network topology combining multiple pest path of nodes is established by NLPS algorithm. With interference model and network capacity model defined in this thesis, the SINR of each link in each time slot is obtained. The SINR of a link is the average of the SINR of this link for each time slot. All SINRs are stored in a table.

In INLPS algorithm, the policy is to find the path with high SINR, reward r need to be changed, which decides the action and lead the agent to reach the goal. Agent will tend to choose the path with higher rewards. Therefore, setting SINR as reward can find the path with high throughput, which is

$$r_{ij} = SINR_{ij} \tag{3.19}$$

Also, SINR table is equal to the reward table in INLPS algorithm.

The INLPS algorithm is as follows: First, apply the NLPS algorithm. After it finished, save the root node and all Q-tables. Use the network topology obtained by the NLPS algorithm to calculate the SINR table and use it as a reward table and use the root node of the NLPS algorithm as D

for INLPS algorithm. After that, enter initialization phase, training phase and selection phase. In the initialization phase, the Q-table after completely updated in NLPS algorithm is used instead of setting all Q-values to 0. The training phase and selection phase are the same as the NLPS algorithm. With INLPS algorithm, the resulting network topology has high capacity.

The INLPS algorithm is similar to applying the NLPS algorithm twice, and the computation time is doubled. In order to reduce the computation time, in the INLPS algorithm, instead of using FG approach to preprocess nodes in the WMN again, the previous result is used as the root node to skip the preprocess phase. Besides that, the Q-tables of the NLPS algorithm is used to instead of the initialization of the Q-tables, therefore the actions made by the agent are more directional, and the best path can be found faster, thereby reducing the number of iterations, and the data of the NLPS algorithm will not be wasted, saving storage space.

The algorithm and flowchart of INLPS is shown in Algorithm 3 and Figure 3.12.

Algorithm 3 SINR-Based Learning Path Selection Algorithm

Definition: N is total number of nodes, t is number of iterations, m is the number of layers, k is the number of nodes in the corresponding layer, U is E2E throughput

Input: Position of nodes, root node D, topology information of nodes by NLPS algorithm

Output: Topology information of nodes

Determine m of nodes based on the transmission range

Phase 1. Initialization

Store all Q-values of NLPS algorithm to Q-tables

Calculate $SINR_{ij}$ of each link and store to reward table

Phase 2. Training

Set S is marked as visited node

while all the relay nodes are not visited **do**

  Set $n^{m-1}_{k_{m-1}} = S$, $U_0 = 0$, $t = 1$

  while true **do**

    for $m = m : M$ **do**

      $s_m = n^{m-1}_{k_{m-1}}$

      s_m randomly selects a relay node as a_m from C_m

      Calculate r_{m+1} from equation (3.16)

      Calculate $Q_{max}(m+1)$ equation (3.18)

      Update $Q_m(s_m, a_m)$ equation (3.15)

      $s_{m+1} = a'_m$ as the best relay node

    end for

    Calculate U_t

    if $|U_t - U_{t-1}| \leq \epsilon$ **then**

      break

    else

      $t = t + 1$

    end if

  end while

  if S has visited all the relay nodes **then**

    break

    else
        Select another relay node as S
    end if
end while

Phase 3. Selection

for $m = m : M$ **do**
    for $k_m = 1 : K_{m-1}$ **do**
        Select a'_m in the Q-table of C_m
    end for
end for

Phase 4. Network Topology Formation

Combine all paths into a tree-based topology
Broadcast topology information to all nodes

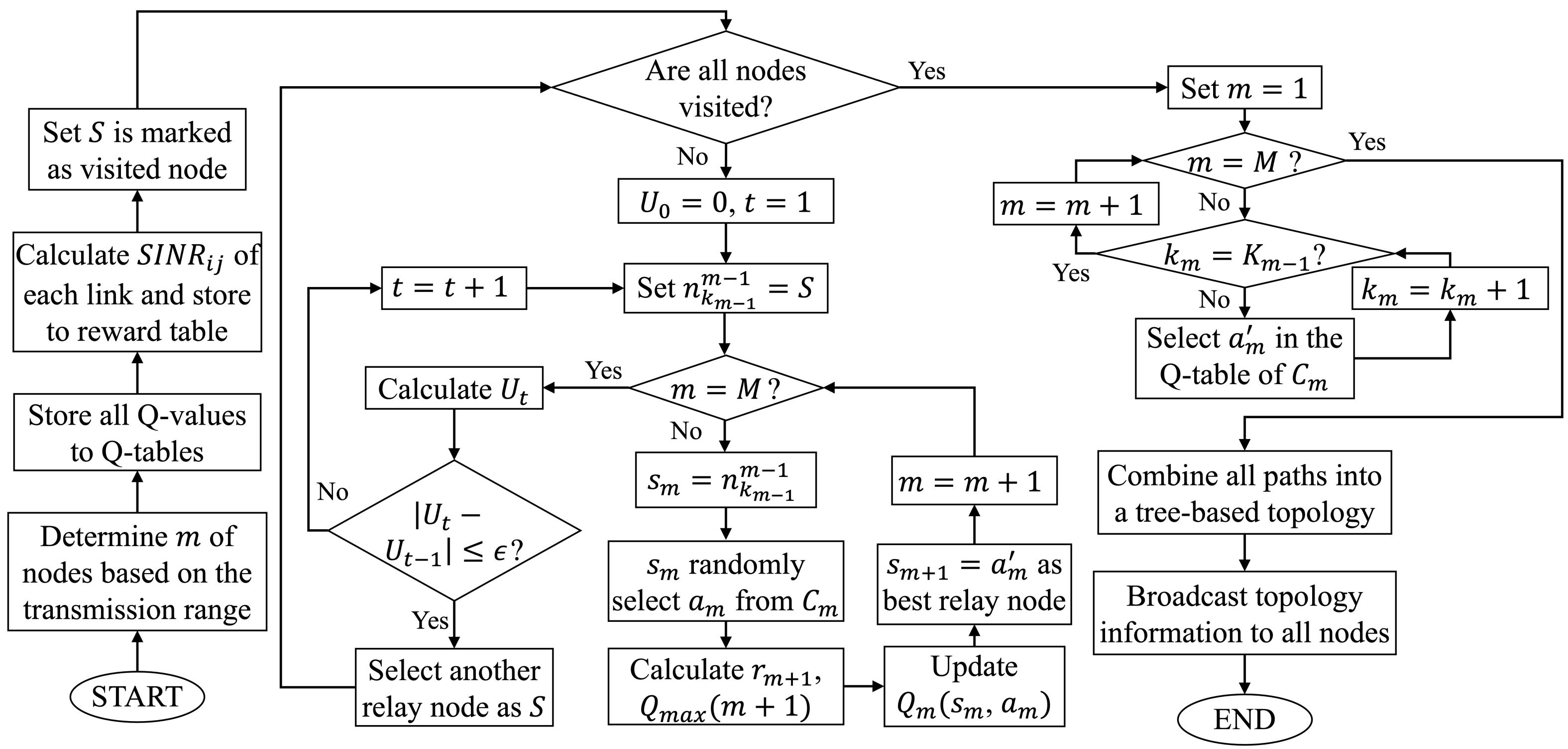


Figure 3.12: Flowchart of INLPS algorithm

3.6 Compute-and-Forward

Although LPS algorithm increase the E2E throughput in path selection, in the transmission stage, due to the high node density, it is necessary to increase network capacity by arranging information transfer in network coding. Nested Lattice Code (NLC) and Compute-and-Forward (CoF) strategy with NLC is used to reduce time slots when two nodes send data packet to a same node.

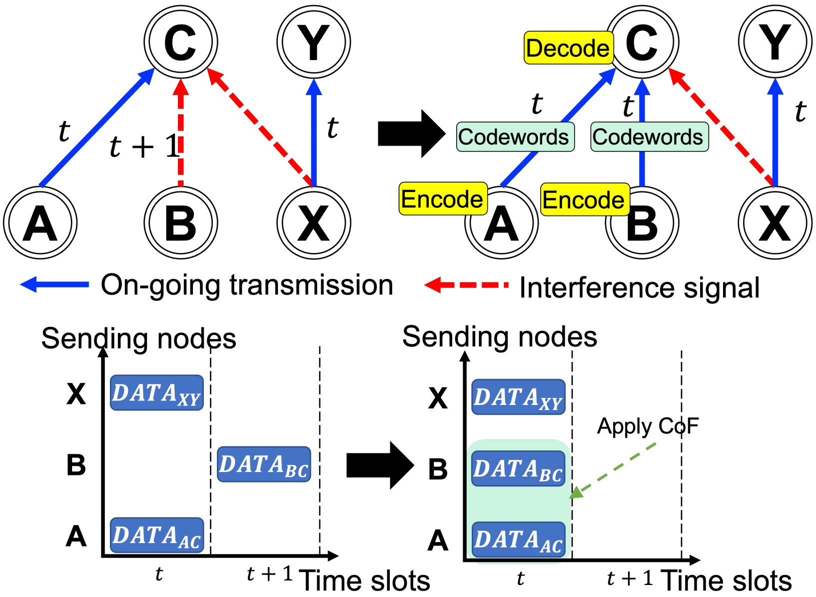


Figure 3.13: Apply CoF strategy in transmission phase

As shown in Figure 3.13, in a single time slot, nodes A, B try to send data packets to node C and X tries to send data packets to node Y. Before applying CoF, the latency can be very large if they try to send at same time, which means it needs more time slots to finish all data transmissions. In this situation, the network capacity will be very low based on system models. Applying CoF strategy, data packets sent from node A and node B at the same time can be encoded at the sending node separately and can be decoded at the receiving node C. With the help of NLC, these two encoded packets including codewords are decoded together at the receiving node C, therefore the resultant of SINR becomes better. As a result, the transmission rate is increased. Furthermore, the total number of time slots is reduced.

In Figure 3.13, the SINR at node C applying CoF strategy is calculated by

$$SINR_{AC\&BC} = \frac{G_{AC} \cdot P_A + G_{BC} \cdot P_B}{\eta_j \cdot B + G_{XY} \cdot P_X} \tag{3.20}$$

If more than two nodes try to send data packets to the same node in a time slot, the receiving node will randomly select two of the sending nodes to establish a connection and transmit data packets, and the remaining nodes need to wait until the next time slot to try to send again.

The algorithm and flowchart of pairing node is shown in Algorithm 4 and Figure 3.14

Algorithm 4 Pairing Node Algorithm

Definition: x is number of time slots, m is number of layers, k is the number of nodes in the corresponding layer

Input: Position of nodes, topology information of all nodes

Output: Total number of time slots, information of pairing nodes

$x = 0$
while D does not receive all data packets **do**
    for $m = 1 : M + 1$ **do**
        for each node in layer C_m **do**
            y = number of transmitting nodes to the same receiving node
            if $y \geq 2$ **then**
                Select two nodes randomly
                Set the remaining nodes to wait
            end if
        end for
    end for
    $x = x + 1$
end while

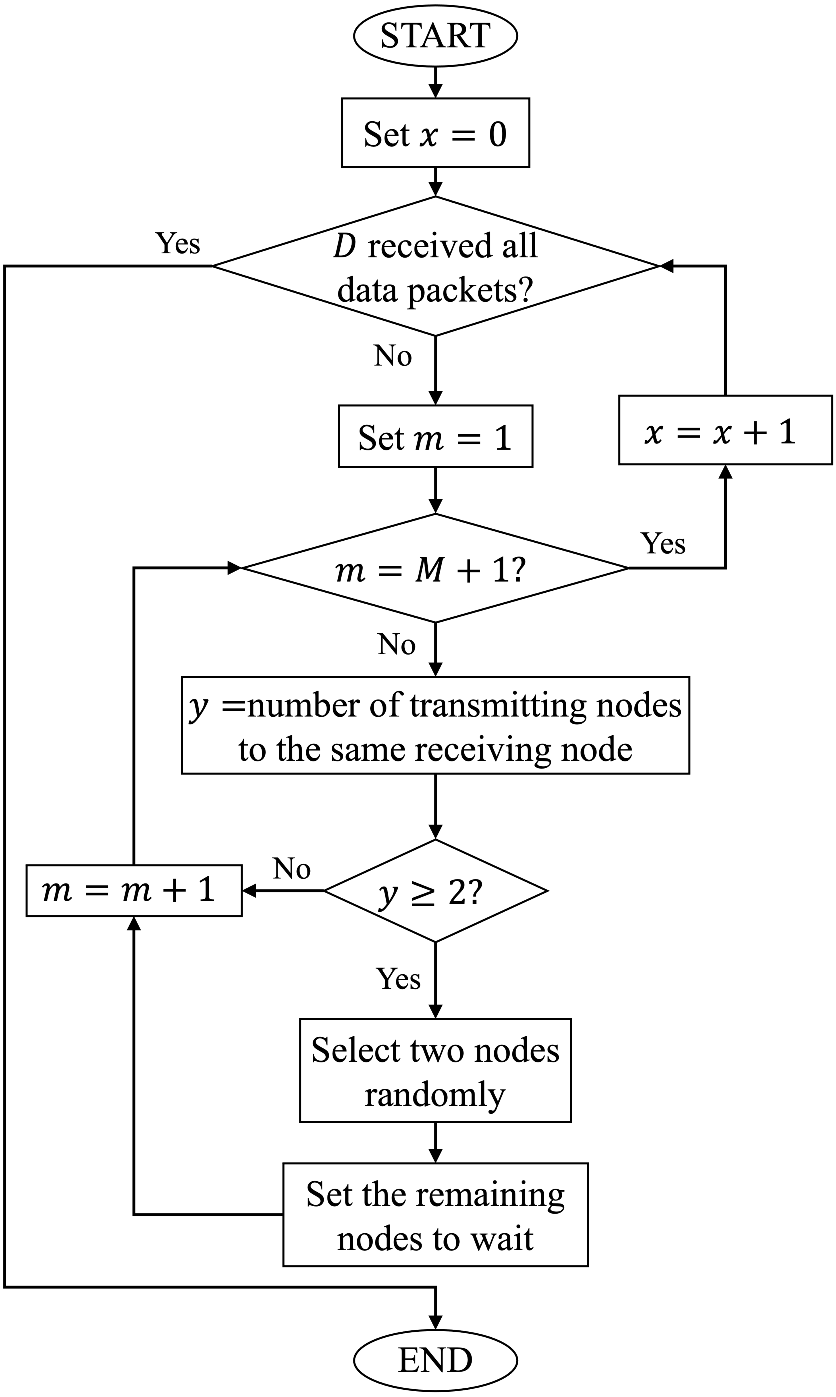


Figure 3.14: Flowchart of CoF

3.7 Summary

This chapter has introduced the system models, including network model, channel model, interference model and network capacity model. Besides that, fDRL scheme with NLPS algorithm and INLPS algorithm is proposed. NLPS algorithm uses FG approach to preprocess the nodes to find root node and use Q-learning to find the best network topology with optimal network capacity ignoring the interference. This algorithm is very suitable for use when establishing a network topology for the first time. Based on the results of FG approach and NLPS algorithm, the INLPS algorithm selects the path with the highest E2E throughput for each node and establishes the network topology, which can further improve the network capacity and reduce the computation time. CoF strategy is introduced into the transmission stage to further increase network capacity through network coding.

Chapter 4

Simulation Studies and Results

4.1 Introduction

The objective of this thesis is to explore the influence of Factor Graph (FG), SNR-based Learning Path Selection (NLPS), SINR-based Learning Path Selection (INLPS) and Nested Lattice Code (NLC) on Wireless Multihop Network (WMN). Specifically, it is manifested in two aspects: network capacity and computation time. The numerical simulation in this thesis is divided into four parts as shown in Figure 4.1: FG, NLPS, INLPS and NLC.

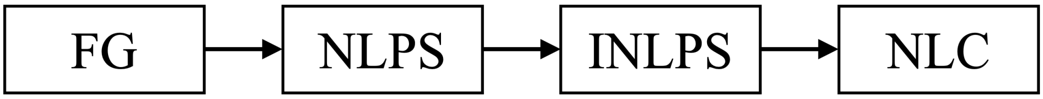


Figure 4.1: Block diagram of simulation program

Several years ago, P. Alevizos [32] has proved that FG can be applied to coding theory, cooperative localization and time scheduling in multihop networks. FG has also been used to solve path selection problems in recent years. H. Alghafari et al. [33] propose a algorithm based on FG and solve the problem of resource allocation and path selection in WMN. Their result shows that the algorithm can give better partitioning results than other load-based methods. Therefore, it is feasible to apply FG to the path selection algorithm and can get good results. In this thesis, FG is used to preprocesses the nodes for the algorithms and select the best root nodes. One of the possible network topologies formed by this root nodes are expected to have the largest network capacity. Second, the computation time of FG needs to be as small as possible. These two points are proved in this chapter.

As for NLPS algorithm and INLPS algorithm, the network capacity of network topology generated by these two algorithms may not be able to reach the maximum value. Instead of that, these two algorithms are expected to achieve optimum network capacity while greatly reducing computation time to meet the low latency requirements in real network environments.

There is currently no research on the use of NLC for network capacity increasing in WMN. This thesis attempts to combine NLC and CoF strategy, to encode and decode messages in the transmission phase, expecting to reduce the number of time slots and further increase network capacity.

So far, there is no research on the combination of FG, Q-learning, and NLC applied to the path selection problem of WMN. Therefore, the numerical simulation in this thesis is completely new and uncharted and can provide certain reference value for society and researchers.

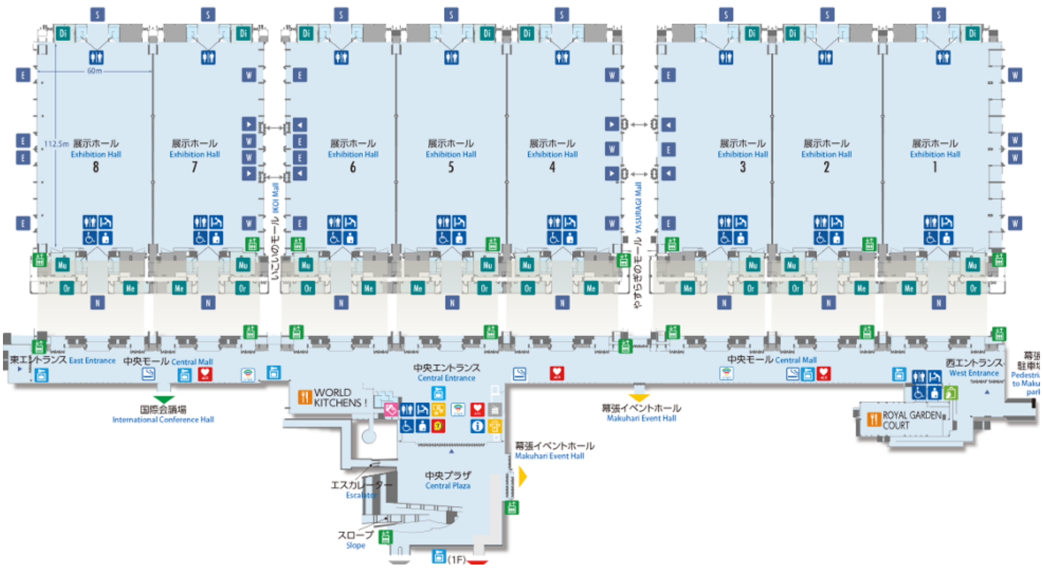
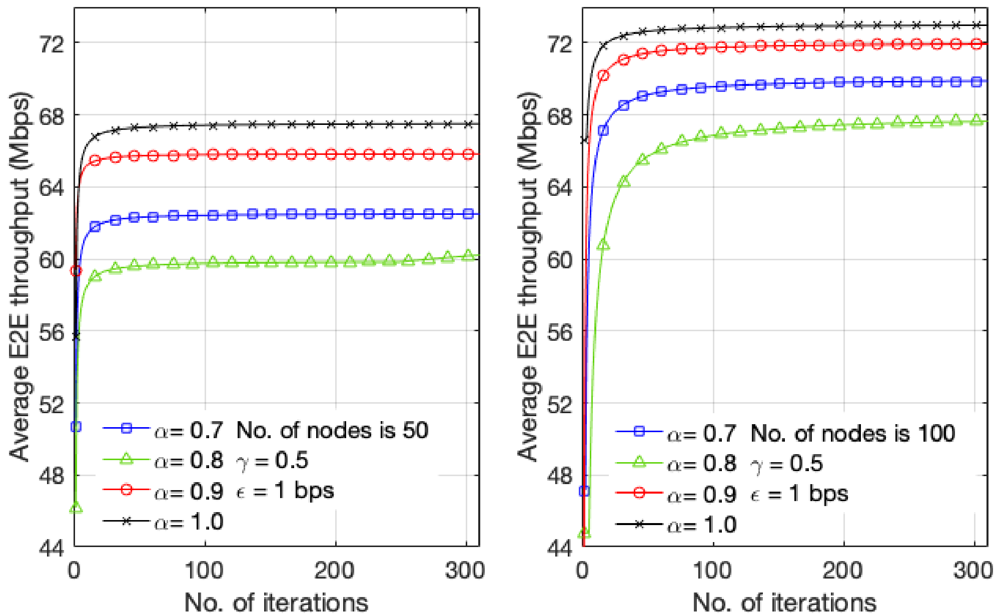


Figure 4.2: Floor map of Makuhari Messe [2]

The application of the scheme in this thesis is expected to the network system of the exhibition hall and stadium with roof, such as the WiFi system of Makuhari Messe [2] in Japan, which is shown in Figure 4.2. If the area of the exhibition hall is very large, it is expected to use 4 to 6 network systems proposed in this thesis to cover the entire area. The simulation parameters are set according to the application in the real situation, including user density, occlusions, size of the exhibition hall and stadium, etc.

4.2 Simulation Parameters and Settings

This numerical evaluation is to study the performance of network capacity and computation time with/without FG, NLPS, INLPS and NLC under the IEEE802.11ax standard specification [34]. In this thesis, we assume that each relay node has only one packet to send to the destination node and nodes can send one signal or receive up to two signals with CoF at the same time. The simulation program is MATLAB R2022a and the desktop environment is Apple Mac mini (2018) with Intel Core i7 3.2 GHz, 64GB DDR4 RAM.



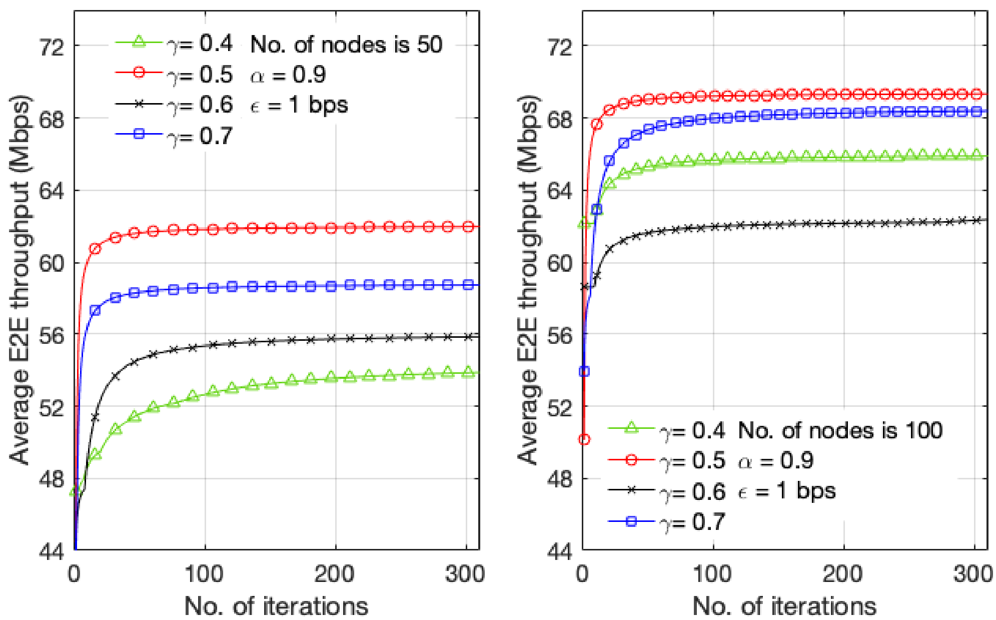
(a) 50 nodes (b) 100 nodes

Figure 4.3: Learning rate comparison

For Q-learning, the default value for learning rate is 1. However, the Q-function used in this thesis has been modified, so it is necessary to re-investigate the appropriate value for simulation. Figure 4.3 shows the comparison of different learning rate at 50 nodes and 100 nodes with same discount factor and threshold.

The result shows that the best learning rate for our Q-function is 1, however, according to equation (3.15), $\alpha = 1$ means that only new information is considered when updating the Q-value, and the influence of current information is completely ignored. In future work, the movement of nodes will be considered, which is included in the current information. It is necessary to give some weight to current information. Also, there is not much difference between $\alpha = 1$ and $\alpha = 0.9$ according to the result. Thus, in this thesis we set $\alpha = 0.9$.

According to equation (3.15), the discount factor in Q-learning decided the importance of the Q-value of the best action. The discount factor is larger means that agent's vision is more long-term, considering the next action more important. We investigate the most suitable value of discount factor for our simulation. The result is shown in Figure 4.4.



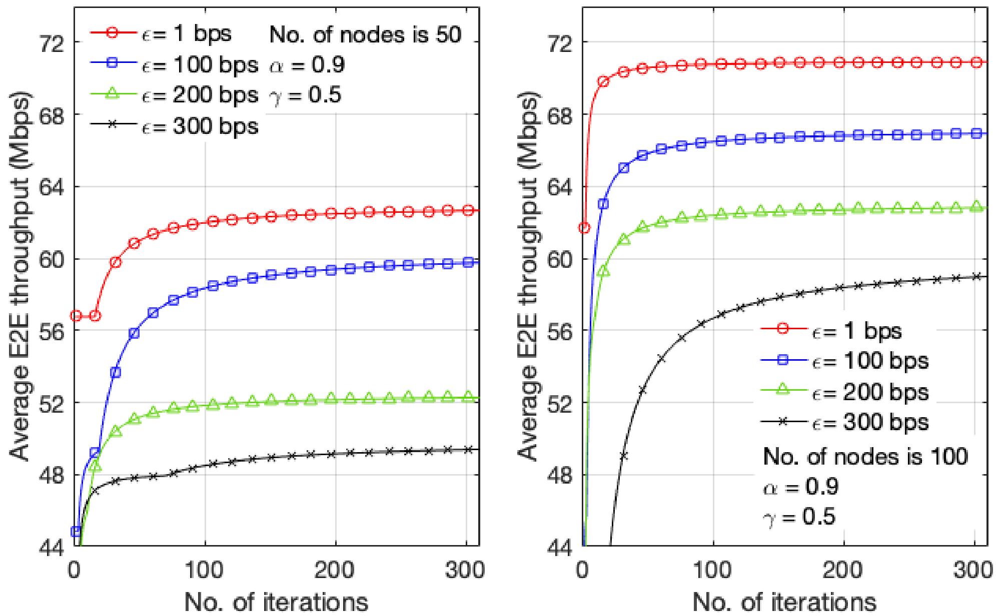
(a) 50 nodes

(b) 100 nodes

Figure 4.4: Discount factor comparison

Through the simulation result, when $\gamma = 0.5$, the performance of average E2E throughput is better, which means setting the importance of $Q_max(m+1)$ as 0.5 is the best for our training. Thus, the discount factor of Q-learning in this thesis is set to 0.5.

The threshold of Q-learning determines when training stops and finishes. In general, the smaller the threshold, the better the results. But at the same time the number of iterations will also increase. In some cases, if the threshold is too small, it may lead to endless training. To avoid this, it is necessary to investigate an appropriate threshold for our simulation. Figure 4.5 is the result of comparison of different thresholds with 50 nodes and 100 nodes.



(a) 50 nodes (b) 100 nodes

Figure 4.5: Threshold comparison

Through this result, we can conclude that the lower the threshold, the higher the average E2E throughput. When the threshold is set to minimum (1bps), the training works fine and almost reaches the maximum value before 300 iterations. In subsequent iterations, the throughput is almost unchanged. Thus, if endless training occurred, we could stop the training after 300 iterations and use this value as the training result. For other threshold, because the results is worse, we set $\epsilon = 1$ bps in this thesis.

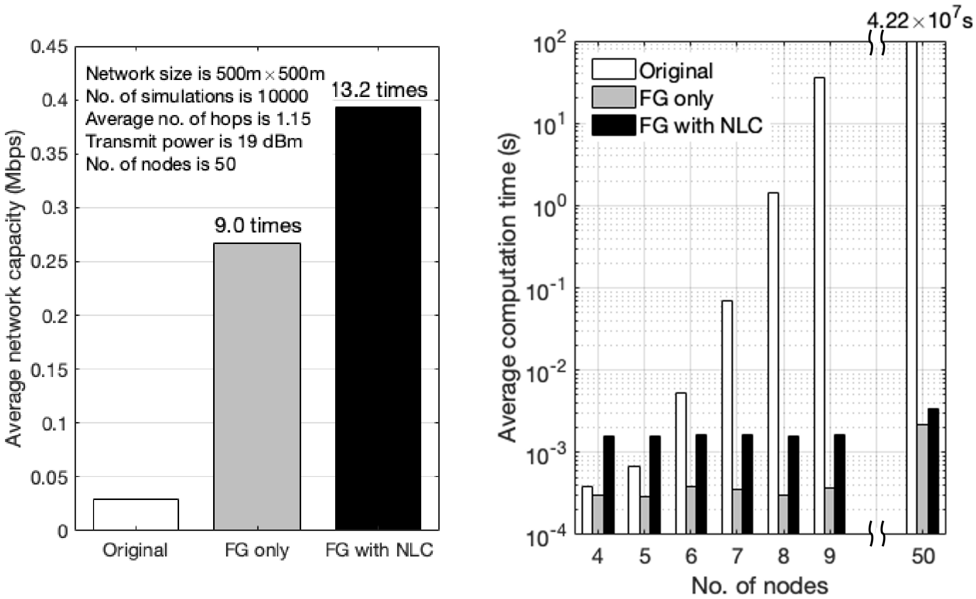
The parameters and values are listed in Table 4.1. The attention constant and shadowing attenuation of log-distance pathloss model is set as 3.5 and 4.0 dB to fit the real situation of the application set by this thesis. The values of other parameters are set with reference to other papers using IEEE802.11ax standard specification and similar network models as this thesis.

Table 4.1: Simulation Parameters and Settings

Parameter	**Value**
Network coverage size	500 m $\times$ 500 m
Number of nodes	50$\sim$100 nodes
Number of simulations	10,000 times
Transmit power	19 dBm
Attenuation constant	3.5
Wall attenuation	0 dB
Shadowing attenuation	4 dB
Decorrelation distance	10 m
PHY header length	39.2 μs
MAC header size	320 bits
Channel bandwidth	20 MHz
Test payload size	8192 bits
Noise level	-174 dBm/Hz
FER	10^{-4}
DIFS	34 μs
SIFS	16 μs
CW_{min}	15 μs
ACK size	112 bits
Slot time	9 μs
Basic rate	6 Mbps
Learning rate	0.9
Discount factor	0.5
Threshold	1 bps
Data packet size	1000 bytes

4.3 Simulation Scenarios, Results and Discussion

The numerical simulation mainly investigates the performance of network capacity and computation time by comparing different scenarios and consists of two parts. In the first part, we investigate the performance of network capacity and computation time with/without FG and NLC to prove that when FG preprocesses data, it can find the root node to improve network capacity and reduce computation time. There are three kinds of scenarios. The network capacity of 'Original' is obtained by calculate the average network capacity of all kinds of possible routing in WMN. For scenarios of 'FG only' and 'FG with NLC' are using FG to select path and using FG and NLC to select path a at the same time, respectively. In the second part, NLPS algorithm and INLPS algorithm is applied to find best network topology in WMN. The scenarios 'NLPS only' and 'INLPS only' are using NLPS algorithm and INLPS algorithm to select best path respectively. Based on this, 'NLPS with NLC' and 'INLPS with NLC' means NLC with CoF strategy is also used in transmission phase to increase network capacity further.



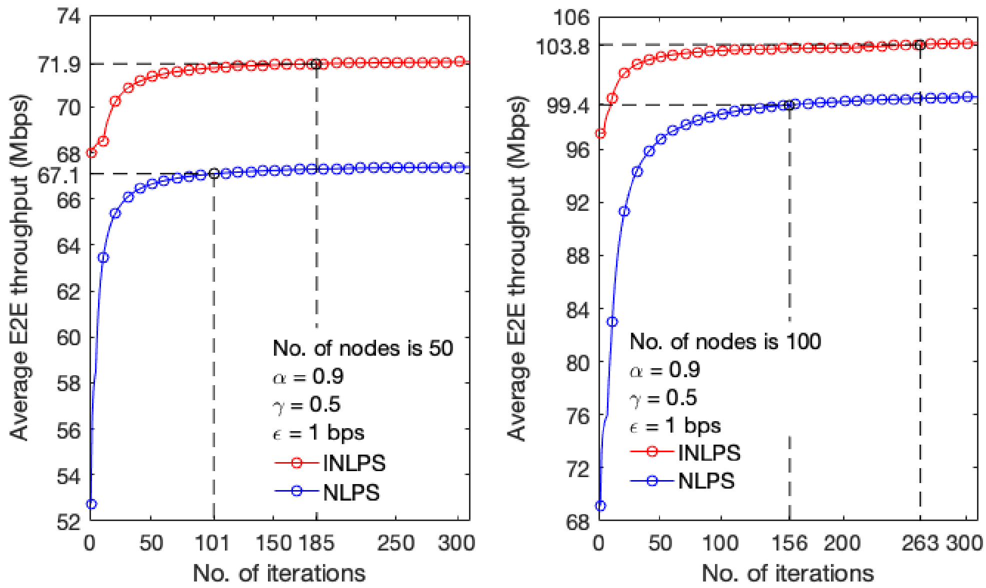
(a) Network capacity comparison (b) Computation time comparison

Figure 4.6: Comparison of network capacity and computation time of FG

Figure 4.6 shows the results of first part. In Figure 4.6 (a), the average network capacity using FG is 0.267 Mbps, which is about 9.0 times higher than without using FG (0.030 Mbps) when the number of nodes is 50. If NLC is also applied, the average network capacity reach to about 13.2 times higher, which is 0.397 Mbps. As for computation time in Figure 4.6 (b), when the number of nodes is smaller than 6, applying FG and NLC will cost more computation time. If there are more than 6 nodes, the computation time of 'Original' becomes higher. When the number of nodes is 50, it costs about 4.22×10^7 s (16 months), while FG with NLC only costs 3.44×10^{-3} s. When there are more than 50 nodes, the simulation is only run to 50 nodes due to the excessive time it takes to calculate all possible paths.

Through these simulation results, we can conclude that the network topology selected by FG can increase the average network capacity and greatly reduce the calculation time. Also, NLC can further increase network capacity by combining the two signals to reducing time slots. Although the computation time will increase slightly, it is also within an acceptable latency in the real network environment. The network capacity is still very low though it is increased by FG and NLC. It is because most network topologies are star-type, which means most nodes choose to send data packets directly to the root node. Also, nodes select the path through the weight of the path, which does not mean that the entire network capacity will reach the maximum. In this situation, it needs large number of time slots to finish the transmission. According to the system model in this thesis, the network capacity becomes very low. It is necessary to use algorithms to plan paths for each nodes.

In the second part of numerical simulation, we investigate the performance of number of iterations, network capacity and computation time. Figure 4.7 shows the results of comparison of number of iterations between NLPS and INLPS when number of nodes is 50 and 100. These results depict that INLPS algorithm can reach higher average E2E throughput than NLPS algorithm. Here number of settling iterations is defined as the number of iterations for the response to reach, and stay within, 2% of its final value. In Figure 4.7 (a), When there are 50 nodes, the number of settling iterations of NLPS (101) is smaller than INLPS (185), which means NLPS cost less iterations to complete the training than INLPS. In Figure 4.7 (b), we get the same result with 100 nodes, which is 156 iterations and 263 iterations



(a) 50 nodes

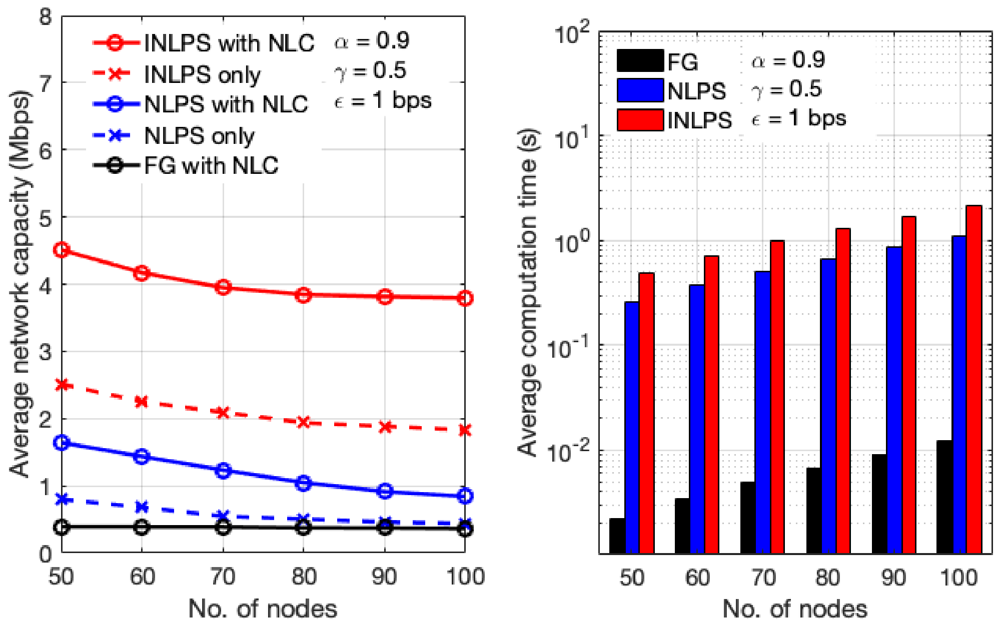
(b) 100 nodes

Figure 4.7: Comparison of no. of iterations between NLPS and INLPS

respectively. Therefore, we can conclude that from 50 nodes to 100 nodes, the average E2E throughput of the path selected by the INLPS algorithm is higher than that of the NLPS algorithm, but it takes more iterations.

This is because the INLPS algorithm uses the Q-values obtained by the NLPS algorithm to initialize its Q-tables and uses them to train based on the results of the NLPS algorithm, so that the average E2E throughput of INLPS algorithm is higher. Since the initialized Q-table of the INLPS algorithm has been updated many times by the NLPS algorithm, it is difficult to find a better path based on it, so the number of iterations spent by INLPS algorithm is more than that of the NLPS algorithm.

The results of the network capacity are shown as Figure 4.8 (a). The number of nodes is from 50 nodes to 100 nodes and five scenarios is compared in this result. This simulation result show that with the increase of nodes, the network capacity decreases non-linearly. This is because the density of nodes is too high and the interference between links causes the SINR to drop. Also, when there are 50 nodes, the average network capacity of INLPS (2.52 Mbps) is about 3.19 times higher than NLPS (0.79 Mbps) and 6.46 times higher than use FG and NLC (0.39 Mbps). When the number of nodes is



(a) Network capacity comparison (b) Computation time comparison

Figure 4.8: Comparison of network capacity and computation time of NLPS and INLPS

100, the values become 1.83 Mbps, 0.43 Mbps and 0.36 Mbps respectively and INLPS algorithm is about 4.26 times higher than NLPS and about 5.08 times higher than use FG and NLC. The reason why the INLPS algorithm performs better is that the INLPS algorithm is trained based on the results of the NLPS algorithm and tries to find a better path. Also, the INLPS algorithm uses SINR as a reward to avoid using links with low SINR when selecting paths. Therefore, the resulting network topology has a higher capacity than NLPS algorithm.

Moreover, if NLC is applied, the When apply NLC, the average network capacity is further increased, which is about 1.93 times higher. This is because NLC and CoF encode and decode two signals together, reducing the number of time slots in half. According to equation (3.11), the network capacity becomes higher.

About the computation time in Figure 4.8 (b), the INLPS algorithm is about 2 times higher than NLPS algorithm because INLPS needs to run NLPS first to get results as its input. Also, FG takes very little time to find root node. If FG is not applied, NLPS needs to be run on all nodes. While it is possible to find a network topology that maximizes network capacity, the computation time is N times higher, which is unacceptable in a real network environment. To sum up, the NLPS algorithm and the INLPS algorithm can find the network topology that can make the network capacity close to the maximum under the premise of greatly reducing the computation time.

4.4 Summary

This chapter first introduces the parameters and settings of the numerical simulation and its application in the real world. Second, the simulations results are evaluated, discussed and compared in this chapter. All these results can show that:

- FG can find the root node that increases the network capacity and greatly reduces the computation time;
- NLPS algorithm and INLPS algorithm can achieve optimum network capacity by selecting path for each node. With the aid of FG, the training time is greatly reduced;
- NLC and CoF can reduce the number of time slots to increase network capacity further.

These simulation results can prove that we have achieved the research objectives set in this thesis and provide a feasible solution to the proposed problems in chapter 1.

Chapter 5

Conclusion

5.1 Concluding Remarks

This research has focused on optimizing network capacity through Factor Graph-based Deep Reinforcement Learning (fDRL) learning path selection scheme in Wireless Multihop Network (WMN). Moreover, reducing the computation time is also one of the research objectives of this thesis. Two Q-learning-based path selection algorithms are proposed: SNR-based Learning Path Selection (NLPS) algorithm and SINR-based Learning Path Selection (INLPS) algorithm. These two algorithms use the root node selected by Factor Graph (FG), which uses sum-product algorithm with airtime cost calculated by Airtime Link Metric (ALM) as weight, to select best path for each node using modified Q-function and form a network topology. This network topology has optimized network capacity. Since the nodes are preprocessed with FG. The training times of the algorithms are greatly reduced, which meets the latency requirements of the real network world. In addition, this thesis uses NLC to encode and decode messages and uses CoF strategy to send them, which reduces the number of time slots and increase the network capacity further. The simulation results prove that the scheme of this thesis has achieved the preset research objectives.

5.2 Contributions

First, the method of converting WMN to FG and using sum-product algorithm to calculate its weight to select the best tree-based network topology proposed in this thesis proves that it is feasible to apply FG to solve the network capacity problem of WMN. Secondly, this thesis innovatively applies NLC to WMN and obtains experimental results. This is of certain reference value to other studies.

The two algorithms in this thesis, NLPS and INLPS, are proposed according to the real network environment. In the WMN environment, the initial network topology can be established in a short period of time by applying the NLPS algorithm. Because in the real network environment, the SINR of the link can be directly obtained, so the INLPS algorithm can be implemented without NLPS. This will greatly shorten the computation time of the INLPS algorithm, and greatly reduce the network delay caused by it. For studies applying Reinforcement Learning (RL) to wireless network in 6G, this thesis presents a method and demonstrates its feasibility by numerical simulation.

5.3 Future Works

The future directions of the research can be extended as the following:

- Node pairing: This thesis randomly select nodes pair to apply CoF. One of the future work is to apply a pair selection algorithm for CoF to choose nodes to increase network capacity

- Fairness, Overhead and Energy saving: In NLPS and INLPS, algorithms focus on optimizing the network capacity. In future work, algorithms will be designed to take into account multiple aspects like consider fairness to reduce nodes' waiting time, consider to reduce the overhead to implement into real networks easier and consider to adjust transmit power depend on the distance to the receiving node to save energy.

- Moving nodes: The nodes in this thesis are set as static nodes. But in reality, nodes are constantly moving, such as mobile phones and vehicles. Learning rate α is set to consider the information brought by the moving node when updating the Q-value. Therefore, optimizing network capacity in the presence of node mobility will be a research goal in future studies.

- Deep learning (DL): To apply the scheme in this thesis to the complex network environment in reality, in future works, DL will be applied to select path in the neural network with hidden layers to solve more complex path selection problems in WMN.

- Least Absolute Shrinkage and Selection Operator (LASSO): LASSO is a regression analysis method that performs both variable selection and regularization to enhance the prediction accuracy and interpretability of the resulting statistical model. LASSO has been used to find stars in space by their light. According to this method, LASSO is expected to be applied in path selection problems in this thesis.

Bibliography

[1] Khaled B. Letaief, Wei Chen, Yuanming Shi, Jun Zhang, and Ying-Jun Angela Zhang. The roadmap to 6g: Ai empowered wireless networks. *IEEE Commun. Mag.*, 57(8):84–90, 2019.

[2] Makuhari messe facility guide. `https://www.m-messe.co.jp/docs/facility/messe_facilityguide.pdf`.

[3] Mohammed H. Alsharif, Anabi Hilary Kelechi, Mahmoud A. Albreem, Shehzad Ashraf Chaudhry, M. Sultan Zia, and Sunghwan Kim. Sixth Generation (6G) Wireless Networks: Vision, Research Activities, Challenges and Potential Solutions. *Symmetry*, 12(4):1–5, 2020.

[4] Ying-Chang Liang, Dusit Niyato, Erik G. Larsson, and Petar Popovski. Guest editorial: 6g mobile networks: Emerging technologies and applications. *China Commun.*, 17(9):90–91, 2020.

[5] Shuping Dang, Osama Amin, Basem Shihada, and Mohamed-Slim Alouini. What should 6g be? *Nature Electron.*, 3(1):20–29, 2020.

[6] Gautam Trivedi and Bijan Jabbari. On wireless link connectivity for resilient multi-hop networks. In *IEEE Mil. Commun. Conf. (MILCOM)*, pages 285–290, 2021.

[7] Chikara Fujimura, Kosuke Sanada, and Kazuo Mori. Throughput analysis for string-topology full-duplex multi-hop network. In *Int. Symp. on Wireless Pers. Multimedia Commun. (WPMC)*, pages 535–541, 2017.

[8] Shahbaz Rezaei, Mohammed Gharib, and Ali Movaghar. Throughput analysis of IEEE 802.11 multi-hop wireless networks with routing consid-

eration: A general framework. *IEEE Trans. on Commun.*, 66(11):5430–5443, 2018.

[9] Won Jae Lee, Jin Ki Kim, Kyeong Rok Kim, and Jae Hyun Kim. Second receiver selection algorithm for fairness in full duplex communications. In *IEEE VTS Asia Pacific Wireless Commun. Symp. (APWCS)*, pages 1–5, 2019.

[10] Jinsong Gui and Kai Zhou. Flexible adjustments between energy and capacity for topology control in heterogeneous wireless multi-hop networks. *J. Netw. Syst. Manage.*, 24(4):789–812, 2016.

[11] Deniz Türsel Eliiyi, Hilal Arslan, Vahid Khalilpour Akram, and Onur Uğurlu. Parallel identification of central nodes in wireless multi-hop networks. In *Signal Process. and Commun. Appl. Conf. (SIU)*, pages 1–4, 2020.

[12] Yu Yu, Shashi Shah, Yasuo Tan, and Yuto Lim. End-to-end throughput evaluation of consensus TPC algorithm in multihop wireless networks. In *Int. Wireless Commun. and Mobile Comput. Conf. (IWCMC)*, pages 941–946, 2015.

[13] Aung Thura Phyo Khun, Yuto Lim, and Yasuo Tan. Optimal achievable transmission capacity scheme with transmit power control in full-duplex wireless multihop networks. In *Int. Conf. on Mobile Comput. and Ubiquitous Netw. (ICMU)*, pages 1–6, 2021.

[14] Aung Thura Phyo Khun, Yuto Lim, and Yasuo Tan. MAC protocol design and analysis for full-duplex wireless networks using mcst scheme. In *Int. Wireless Commun. and Mobile Comput. (IWCMC)*, pages 1394–1399, 2022.

[15] Yongyi Mao, Frank Kschischang, Baochun Li, and Subbarayan Pasupathy. A factor graph approach to link loss monitoring in wireless sensor networks. *IEEE J. on Sel. Areas in Commun.*, 23(4):820–829, 2005.

[16] Wei Li, Zhen Yang, and Haifeng Hu. Distributed multi-sensor tracking in wireless networks using nonparametric variant of sum-product

algorithm. In *Asia-Pacific Conf.on Commun. (APCC)*, pages 132–137, 2013.

[17] Changhui Jiang, Yuwei Chen, Chen Chen, Jianxin Jia, Haibin Sun, Tinghuai Wang, and Juha Hyyppä. Smartphone PDR/GNSS integration via factor graph optimization for pedestrian navigation. *IEEE Trans. on Instrum. and Meas.*, 71:1–12, 2022.

[18] Xiande Bu, Chuan Liu, Qiang Yu, Lei Yin, and Feng Tian. Optimization on cooperative communications based on network coding in multihop wireless networks. In *Int. Wireless Commun. and Mobile Comput. (IWCMC)*, pages 384–387, 2020.

[19] Jiajie Xue and Brian Michael Kurkoski. Lower bound on the error rate of genie-aided lattice decoding. In *IEEE Int. Symp. on Inf. Theory (ISIT)*, pages 3232–3237, 2022.

[20] A. L. Samuel. Some studies in machine learning using the game of checkers. *IBM J. of Res. and Develop.*, 3(3):210–229, 1959.

[21] Julia Rosenberger, Michael Urlaub, Felix Rauterberg, Tina Lutz, Andreas Selig, Michael Bühren, and Dieter Schramm. Deep reinforcement learning multi-agent system for resource allocation in industrial internet of things. *Sensors*, 22(11), 2022.

[22] Dmitrii A. Dugaev, Ivan G. Matveev, Eduard Siemens, and Viatcheslav P. Shuvalov. Adaptive reinforcement learning-based routing protocol for wireless multihop networks. In *Int. Scientific-Tech. Conf. on Actual Problems of Electron. Instrum. Eng. (APEIE)*, pages 209–218, 2018.

[23] Tanutsorn Wongphatcharatham, Watid Phakphisut, Thongchai Wijitpornchai, Poonlarp Areeprayoonkij, Tanun Jaruvitayakovit, and Pimkhuan Hannanta-anan. Multi-agent q-learning for power allocation in interference channel. In *Int. Tech. Conf. on Circuits/Syst., Comput. and Commun. (ITC-CSCC)*, pages 876–879, 2022.

[24] Xiaowei Wang and Xin Wang. Reinforcement learning-based multihop relaying: A decentralized q-learning approach. *Entropy (Basel)*, 23(10), 2021.

[25] Mahda Noura and Rosdiadee Nordin. A survey on interference management for device-to-device (D2D) communication and its challenges in 5G networks. *J. of Netw. and Comput. Appl.*, 71:130–150, 2016.

[26] Arash Asadi, Qing Wang, and Vincenzo Mancuso. A survey on device-to-device communication in cellular networks. *IEEE Commun. Surv. & Tut.*, 16(4):1801–1819, 2014.

[27] ANSI/IEEE Standard 802.11. Part II: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications mesh networking. *IEEE Standard Specification 802.11s Mesh WLAN*, 10 September 2011.

[28] Brian Michael Kurkoski. Encoding and indexing of lattice codes. *IEEE Trans. on Inf. Theory (ISIT)*, 64(9):6320–6332, 2018.

[29] Bobak Nazer and Michael Gastpar. Compute-and-forward: Harnessing interference through structured codes. *IEEE Trans. on Inf. Theory (ISIT)*, 57(10):6463–6486, 2011.

[30] Francisco S Melo. Convergence of q-learning: A simple proof. *Inst. Of Syst. and Robot., Tech. Rep*, pages 1–4, 2001.

[31] Vincent François-Lavet, Raphaël Fonteneau, and Damien Ernst. How to discount deep reinforcement learning: Towards new dynamic strategies. *CoRR*, abs/1512.02011, 2015.

[32] Panagiotis Alevizos. *Factor Graphs: Theory and Applications*. PhD thesis, TECHNICAL UNIVERSITY OF CRETE, 2012.

[33] Hadeel Alghafari and Mohammad Sayad Haghighi. Decentralized joint resource allocation and path selection in multi-hop integrated access backhaul 5g networks. *Comput. Netw.*, 207:108837, 2022.

[34] ANSI/IEEE Standard 802.11. Part II: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications enhancements

for high-efficiency WLAN. *IEEE Standard Specification 802.11ax*, 19 May 2021.

List of Publications

1. CUI Zhihan, Khun Aung Thura Phyo, LIM Yuto, and TAN Yasuo. Study of Network Capacity in Wireless Multihop Networks with Factor Graph. In *Joint Conf. of Hokuriku Chapters of Elect. and Inf. Soc. (JHES)*, 2022.

2. CUI Zhihan, Khun Aung Thura Phyo, LIM Yuto, and TAN Yasuo. Study of Network Capacity in Wireless Multihop Networks with Nested Lattice and Factor Graph. In *IEICE Tech. Committee on Inf. Netw. (IEICE-IN)*, 2022.

3. CUI Zhihan, Khun Aung Thura Phyo, LIM Yuto, and TAN Yasuo. Study of Deep Reinforcement Learning for Wireless Multihop Networks. In *IEICE Tech. Committee on Sensor Netw. and Mobile Intell. (IEICE-SeMI)*, 2023 (To be submitted).