國立陽明交通大學

電子研究所

博士論文

Institute of Electronics
National Yang Ming Chiao Tung University
Doctoral Dissertation

無線通訊系統中基於圖神經網路之波束成形與用戶關聯設計

Graph Neural Network-Enabled Beamforming and User Association in Wireless Communication Systems

研究生：詹坤霖（Kun-Lin Chan）

指導教授：簡鳳村（Feng-Tsun Chien）

中華民國一一五年八月

August 2026

無線通訊系統中基於圖神經網路之波束成形與用戶關聯設計

Graph Neural Network-Enabled Beamforming and User Association in Wireless Communication Systems

研究生：詹坤霖　　Student：Kun-Lin Chan
指導教授：簡鳳村　　Advisor：Feng-Tsun Chien

國立陽明交通大學
電子研究所

博士論文

A Dissertation
Submitted to Institute of Electronics
College of Electrical and Computer Engineering
National Yang Ming Chiao Tung University
in Partial Fulfillment of the Requirements
for the Degree of
Doctor of Philosophy
in
Electronics

August 2026
Taiwan, Republic of China

中華民國 一一五年八月

Acknowlegment

This dissertation could not have been completed without the guidance, collaboration, and support of many people throughout my doctoral journey. First and foremost, I would like to express my deepest gratitude to my advisor, Prof. Feng-Tsun Chien. From the very beginning, when I took the courses you offered, to becoming your Ph.D. student under your supervision, and finally to the completion of this dissertation, you have provided not only invaluable knowledge in the fields of communications and machine learning, but also careful and patient guidance in research. Your mentorship has helped me develop comprehensive insights and a clear vision in my research area. I would also like to extend my sincere appreciation to Prof. Ronald Y. Chang for providing me with clear directions in both research and paper writing, as well as for teaching me many valuable skills in academic writing and a deeper understanding of research trends. In addition, as a member of the commlab, I would like to thank Prof. Tzu-Hsien Sang, Prof. Carrson C. Fung, Prof. Geng-Shi Jeng, and Prof. David W. Lin. Through their courses and the weekly commlab seminars, I have gained extensive knowledge in the field of communications and learned many important presentation skills for research.

I would like to express my special thanks to my outstanding coworker and close friend, Chi-Kai. Back in my freshman year of college, if it were not for your invitation to transfer together to the Department of Electronics at National Yang Ming Chiao Tung University, I would not be where I am

today. From undergraduate and graduate coursework to doctoral research, your help, companionship, and encouragement have supported me through the challenging journey of pursuing a Ph.D. I would also like to thank all the members of commlab, including my peers and junior colleagues who have worked alongside me. My sincere appreciation also goes to the commlab assistant, Chu-Ming, for her strong emotional support and for teaching me how to take on the responsibilities of managing a laboratory. Finally, I am truly grateful to my family for their unconditional support and encouragement, which have enabled me to successfully complete my doctoral journey.

Kun-Lin
August 2026

無線通訊系統中基於圖神經網路之波束成形與用戶關聯設計

學生：詹坤霖　　指導教授：簡鳳村 博士

國立陽明交通大學

電子研究所

摘 要

本論文著眼於可重構智慧表面在無線通訊系統中擴展覆蓋範圍與提升頻譜效率的能力，探討在多基地台與多使用者系統中，可重構智慧表面能為波束成形與使用者關聯策略帶來多大程度的彈性。本論文主要分為三個部分。第一部分研究在具備負載平衡考量的可重構智慧表面輔助多基地台多使用者毫米波網路中，使用者關聯、基地台波束成形以及 可重構智慧表面相位調整的聯合最佳化問題。為了解決此一複雜問題，我們提出一種基於圖神經網路的方法，該方法對於不同數量的基地台與使用者具有良好的泛化能力。此外，我們引入自適應注意力式聚合機制，以緩解深層圖神經網路中的過度平滑問題。第二部分將研究延伸至超大規模多輸入多輸出系統。超大規模多輸入多輸出預期可大幅提升第六代無線通訊系統的頻譜效率，但同時也引入近場效應，其中遠場傳輸可能因能量擴散效應而對近場使用者造成干擾。本部分考慮一個可重構智慧表面輔助之多基地台超大規模多輸入

多輸出系統，系統中同時包含近場與遠場使用者。我們提出一種基於圖神經網路的方法，聯合最佳化使用者關聯、基地台波束成形與可重構智慧表面相位調整，以最大化總和速率，並結合基於相關性的注意力機制與近場陣列對齊強化訓練，以減輕能量擴散問題。第三部分探討無蜂巢系統中的波束成形設計。傳統基於集中式最佳化的波束成形方法會產生大量的通訊負擔。為了解決此問題，我們結合聯邦學習與圖神經網路，提出一種聯邦式圖神經網路架構用於無蜂巢系統的波束成形設計。此方法不僅能降低資料傳輸需求，亦能對不同系統參數（例如使用者數量）具備良好的泛化能力，進而提升未來無線通訊系統中波束成形設計的效率與可擴展性。

關鍵字：可重構智慧表面、波束成形、使用者關聯、超大規模多輸入多輸出、深度學習、圖神經網路、聯邦學習。

Graph Neural Network-Enabled Beamforming and User Association in Wireless Communication Systems

Student: Kun-Lin Chan Advisors: Dr. Feng-Tsun Chien

Institute of Electronics

National Yang Ming Chiao Tung University

Abstract

Focusing on the capability of Reconfigurable Intelligent Surfaces (RISs) to expand coverage and enhance spectral efficiency in wireless communication systems, this dissertation investigates the extent to which RIS can provide flexibility in beamforming and user association strategies for multi–base station (BS) and multi–user equipment (UE) systems. This dissertation consists of three main parts. In the first part, we investigate the joint optimization of user association, BS beamforming, and RIS phase adjustment in an RIS-aided multi-BS multi-UE millimeter-wave (mmWave) network with load-balancing considerations. To address this complex problem, we propose a graph neural network (GNN)-based approach with enhanced generalizability to varying numbers of BSs and UEs. An adaptive attention-based aggregation (AAA) mechanism is incorporated to mitigate the oversmoothing issue in deep GNNs. In the second part, the study is extended to Extremely Large-Scale Multiple-Input Multiple-Output (XL-MIMO) systems. XL-MIMO is expected to significantly improve spectral efficiency in sixth-generation (6G) wireless systems, but it also introduces near-field effects, where far-field

transmissions may interfere with near-field users due to energy spread effects (ESEs). This part considers a RIS-assisted multi-BS XL-MIMO system with both near-field and far-field users. We propose a GNN-based approach to jointly optimize user association, BS beamforming, and RIS phase shifts for sum-rate maximization, incorporating correlation-based attention and near-field array alignment enhanced training to mitigate energy spread. Finally, the third part explores beamforming design in cell-free systems. Conventional centralized optimization-based beamforming designs incur significant communication overhead. To address this issue, we integrate federated learning (FL) with GNNs and propose a federated GNN framework for beamforming design in cell-free systems. This approach not only reduces the required data transmission but also achieves generalization across different system parameters, such as the number of users, thereby enabling more efficient and scalable beamforming design for future wireless communication systems.

Keywords: Reconfigurable intelligent surface, beamforming, user association, XL-MIMO, deep learning, graph neural network, federated learning.

Contents

1 Introduction **1**

1.1 Significance of the Research 1

1.2 Related Works 5

1.2.1 Related Works for Part I: GNN for the Joint Design of RIS-aided User Association and Beamforming 5

1.2.2 Related Works for Part II: GNN for User Association and Beamforming in Mixed Near- and Far-Field RIS-aided Systems 7

1.2.3 Related Works for Part III: Federated Graph Neural Network for Beamforming in Cell-free MIMO Systems 7

1.3 Motivation 11

1.3.1 Motivation of Part I: GNN for the Joint Design of RIS-aided User Association and Beamforming 11

1.3.2 Motivation of Part II: GNN for User Association and Beamforming in Mixed Near- and Far-Field RIS-aided Systems 12

1.3.3 Motivation of Part III: Federated Graph Neural Network for Beamforming in Cell-free MIMO Systems 12

1.4 Contributions of the Research 14

1.5 Dissertation Organization 16

1.6 Notations 16

2 Preliminary **17**

2.1 Introduction to Graph Neural Network (GNN) 17

2.2 Theoretical Basis of GNN 19

2.3 GNN for Communications 21

3 GNN for the Joint Design of RIS-aided User Association and Beamforming 25

3.1 System Model 25

3.2 Problem Description 27

3.3 Methodology 29

3.3.1 Preliminary 29

3.3.1.1 Representation of the Considered System 29

3.3.1.2 Input Data 29

3.3.2 GNN Design 30

3.3.2.1 GNN Structure 30

3.3.2.2 Oversmoothing Issue and Solution 34

3.3.2.3 Sparse Attention for Aggregation 35

3.3.2.3.1 Computing Attention Weights 35

3.3.2.3.2 Sparsifying the Attention Matrix 35

3.3.2.3.3 Adaptively Adjusting the Sparsity Level . . . 36

3.3.2.4 Tuning Aggregation for Combination 37

3.3.2.5 Output of the GNN 37

3.3.2.6 Training of GNN 39

3.3.3 Complexity Analysis 40

3.3.3.1 Sparse Attention for Aggregation 42

3.3.3.1.1 Computing Attention Weights 42

3.3.3.1.2 Sparsifying the Attention Matrix 42

3.3.3.1.3 Adaptively Adjusting the Sparsity Level . . . 42

3.3.3.2 Tuning Aggregation for Combination 43

3.4 Simulations 44

3.4.1 Simulation Setups 44

3.4.2 Convergence 46

3.4.3 Sum-rate Performance 47

3.4.4 Effects of RIS on User Association 52

3.4.5 Effects of Oversmoothing . . . 53
3.4.6 Complexity . . . 55
3.4.7 Generalizability . . . 56

4 GNN for User Association and Beamforming in Mixed Near- and Far-Field RIS-aided Systems 62

4.1 System Model . . . 62
4.2 Energy Spread Effect (ESE) . . . 65
4.3 Problem Description . . . 65
4.4 Methodology . . . 67
4.4.1 GNN Design . . . 67
4.4.1.1 GNN Structure . . . 67
4.4.1.2 Output of the GNN . . . 67
4.4.1.3 Solution for ESE: Attention Mechanism Design . . . 70
4.4.1.4 Solution for ESE: Penalty Design for Enhancing Training 71
4.4.2 Complexity Analysis . . . 71
4.4.2.1 CBA Aggregation . . . 72
4.5 Simulations . . . 74
4.5.1 Simulation Setups . . . 74
4.5.2 Sum-rate Performance . . . 76
4.5.3 User Association Examination . . . 77
4.5.4 Variation of Mixed Near- and Far-Field Channel . . . 77
4.5.5 Complexity . . . 78
4.5.6 Generalizability . . . 78

5 Federated Graph Neural Network for Beamforming in Cell-free MIMO Systems 82

5.1 Background . . . 82
5.1.1 Introduction to Federated Graph Neural Network(FedGNN) . . . 82
5.1.2 Literature Review of Cell-free MIMO Beamforming . . . 85
5.2 System Model and Problem Description . . . 86

5.3 Methodology 87
5.3.1 Method 1: GNN-assisted FL for Beamforming in Cell-free MIMO Systems 87
5.3.1.1 Workflow of Method 1 88
5.3.1.2 GNN Structure of Method 1 90
5.3.1.3 The Advantages and Disadvantages of Method 1 91
5.3.2 Method 2: FL-assisted GNN for Beamforming in Cell-free MIMO Systems 91
5.3.2.1 Workflow of Method 2 91
5.3.2.2 GNN Structure of Method 2 93
5.3.2.3 The Advantages and Disadvantages of Method 2 93
5.4 Simulations 94
5.4.1 Simulation Setups 94
5.4.2 Simulation Results 95

6 Conclusion and Future Work **99**
6.1 Conclusion 99
6.2 Future Work 100

Bibliography **102**

List of Figures

1.1 An RIS-assisted multi-BS multi-UE system (an example of two BSs and 4 UEs is shown here). 3

1.2 A RIS-aided multiuser mmWave system. The blue and orange ellipses denote user associations with BS 1 and BS 2, respectively, while the green area highlights interference to nearby-angle near-field UE 1 if association BS 1–UE 2 were formed instead (energy spread effect). 4

2.1 The aggregation and combination in a GNN [1]. 18

2.2 Graph representation of the various communication issues. 22

3.1 An RIS-assisted multi-BS multi-UE system. 26

3.2 Graph representation of the considered RIS-aided multi-BS multi-UE system. 30

3.3 The multilayered structure of the proposed GNN-AAA. 31

3.4 Details of the initial layer in the proposed GNN-AAA. 32

3.5 Details of the τth updating layer for the BS–UE nodes in the proposed GNN-AAA. 33

3.6 Details of the τth updating layer for the RIS node in the proposed GNN-AAA. 34

3.7 Details of the output layer in the proposed GNN-AAA. 38

3.8 System topology for simulation. 46

3.9 Training loss vs. training epochs, where $B = 3$, $U = 6$, $P_{\max,b} = 25$ dBm $\forall b$, and $M = 100$. 47

3.10 Testing sum rate vs. BS maximum transmit power ($P_{\max,b}, \forall b$) for $B = 3$, $U = 6$, and $M = 100$. 48
3.11 Testing sum rate and load-balancing satisfaction rate vs. λ in (3.33) for $B = 3$, $U = 6$, $P_{\max,b} = 25$ dBm $\forall b$, and $M = 100$. 49
3.12 Testing sum rate vs. BS maximum transmit power ($P_{\max,b}, \forall b$) for $B = 3$, $U = 6$, and $M = 100$, under different pilot correlation scenarios using random and ZC pilot sequences of length $L = 31$. 50
3.13 Testing sum rate vs. RIS–UE center distance for $B = 3$, $U = 6$, $M = 100$, and $P_{\max,b} = 25$ dBm. 51
3.14 Testing sum rate vs. number of RIS passive elements (M) for $B = 3$, $U = 6$, and $P_{\max,b} = 25$ dBm. 52
3.15 Testing sum rate vs. BS maximum transmit power ($P_{\max,b}, \forall b$) for $B = 3$, $U = 6$, and $M = 100$, with 2-bit quantized RIS phase shifts. 53
3.16 Testing sum rate vs. BS maximum transmit power ($P_{\max,b}, \forall b$) for $B = 3$, $U = 6$, and $M = 100$, with 1-bit and 2-bit quantized RIS phase shifts. . . 54
3.17 Testing sum rate vs. BS maximum transmit power ($P_{\max,b}, \forall b$) for $B = 3$, $U = 6$, and $M = 100$, with the effect of imperfect CSI. 55
3.18 Example scenarios to illustrate the effects of RIS on UA in multi-BS multi-UE systems, with $B = 3$, $U = 6$, $M = 100$, and $P_{\max,b} = 25$ dBm. (a) UA for *GNN-AAA [no RIS]*. 56
3.19 Example scenarios to illustrate the effects of RIS on UA in multi-BS multi-UE systems, with $B = 3$, $U = 6$, $M = 100$, and $P_{\max,b} = 25$ dBm. (b) UA for *GNN-AAA [proposed]*. 57

4.1 A RIS-aided multiuser mmWave XL-MIMO system. 63
4.2 Beam patterns of a 128-element ULA with far-field beamforming, evaluated at both far-field and near-field observation points. The desired angle is $\theta' = 90°$, with the far-field observation taken at 100 m and near-field observations at $r = 5, 15, 30$ m. The angular offset is defined as $\Delta_\theta = \theta' - \theta$. 66
4.3 Details of the initial layer in the proposed GNN. 68

4.4 Details of the τth updating layer for the BS–UE nodes in the proposed GNN. 68
4.5 Details of the τth updating layer for the RIS node in the proposed GNN. . 69
4.6 Details of the output layer in the proposed GNN. 69
4.7 Simulation topologies. (a) Scenario 1 with randomly distributed UEs. (b) Scenario 2 with fixed UEs arranged to induce ESEs. 75
4.8 Testing sum rate vs. BS maximum transmit power $P_{\max}$ for Scenarios 1 and 2, with $B = 3$, $U = 10$, and $M = 100$. 76
4.9 User association snapshots in Scenario 2 for (a) the proposed GNN, (b) GNN (no CBA and NAA), and (c) both model-based and codebook-based methods, with $B = 3$, $U = 10$, and $M = 100$. Note that the transmission links reflected by the RIS are omitted in this figure for clarity. 80
4.10 Testing sum rate vs. radius of the near-field region for Scenario 1, with $B = 3$, $U = 10$, and $M = 100$. 81

5.1 The classification of FedGNNs. 83
5.2 GNN-assisted FL for the beamforming problem in cell-free MIMO systems. 88
5.3 The structure of the GNN at the CPU for the GNN-assisted FL method. . 90
5.4 FL-assisted GNN for the beamforming problem in cell-free MIMO systems. 92
5.5 The structure of the GNN at each BS for the FL-assisted GNN method. . 94
5.6 Simulation topologies. 95
5.7 The testing sum rate vs. the maximum BS transmit power. 96
5.8 The convergence results of each BS. 96
5.9 The testing sum rate vs. the number of UEs. 97

List of Tables

3.1 List of Symbols 58
3.2 Simulation Parameters 59
3.3 Squared norm of Node Differences and Sum Rate for $B = 3$, $U = 6$, $M = 100$, $Q = 8$, and $P_{\max,b} = 25$ 60
3.4 Runtime (Unit: Millisecond) 60
3.5 Sum Rate for *GNN-AAA [proposed]* over Various Numbers of BSs (B) in Training and Testing, with $M = 100$, $U = 6$, and $P_{\max,b} = 25$ 61
3.6 Sum Rate for *GNN-AAA [proposed]* over Various Numbers of UEs (U) in Training and Testing, with $M = 100$, $B = 3$, and $P_{\max,b} = 25$ 61

4.1 Runtime (Unit: Millisecond) 79
4.2 Sum Rate for Various UE Numbers Used in Testing and Training 79

5.1 Summary of Related Works 86
5.2 Sum rate for various U's used in testing and training 98

Chapter 1

Introduction

In this chapter, the research background, related works, motivations, and contributions of this dissertation, *Graph Neural Network-Enabled Beamforming and User Association in Wireless Communication Systems*, are introduced. The dissertation mainly consists of three research topics: the main research topic, a GNN-based method for the joint design of RIS-aided user association and beamforming; a GNN-based method for user association and beamforming in mixed near- and far-field RIS-aided systems; and a federated GNN framework for beamforming in cell-free MIMO systems. Each research topic is individually introduced in this chapter, while the detailed methodologies and results are presented in Chapters 3, 4, and 5, respectively.

1.1 Significance of the Research

Reconfigurable intelligent surfaces (RISs) are considered to be a promising technology for expanding coverage and enhancing spectral efficiency in wireless systems [2–4]. RISs enable the reconfiguration of the transmission environment to mitigate signal blockage and interference. Thus, RISs have the potential to improve the spectral efficiency of mmWave communications that suffer serious signal attenuation. Various methods have been proposed to jointly optimize transmit beamforming and RIS phase shifts, employing conventional optimization techniques (e.g., [2, 3]), data-driven deep learning approaches (e.g., [5–7]), or graph neural network (GNN) techniques (e.g., [8–11]). Among these

methods, GNNs have proven their viability in achieving superior performance and lower complexity compared to optimization-based approaches, as well as enhanced generalizability compared to deep neural network (DNN)-based approaches, as shown in [8, 9].

Another advantage of RIS-aided systems is the enhanced flexibility in *user association (UA)* due to coverage extension. For example, as illustrated in Fig. 1.1, user equipment (UE) 2, which is not within the coverage of base station (BS) 2, may now be associated with BS 2 with the assistance of an RIS. This proves valuable when the link between UE 2 and BS 1 is obstructed, even if UE 2 is closer to BS 1. Another interesting observation is that the UA strategy becomes less straightforward when *load balancing* is taken into consideration. For the example in Fig. 1.1, when considering load distribution among BSs, a more balanced UA strategy may link UE 2 to the less-loaded BS 2 instead of BS 1, even though linking UE 2 to BS 1 is also feasible with the assistance of an RIS. Therefore, RIS technology has the potential to facilitate a more flexible and load-aware UA strategy.

As described above, this research aims to investigate the potential of RIS-aided UA strategy. Therefore, the first part of this research, which focuses on jointly optimizing UA, transmit beamforming (BF), and RIS phase shifts in a RIS-assisted multi-BS multi-UE mmWave system with BS load balancing considerations, through a graph neural network (GNN)-based approach, is presented in Chapter 3. This joint design is challenging due to the non-convex nature of the problem and the strong coupling among the variables, especially in multi-BS settings. Furthermore, in mmWave systems, accurate channel state information (CSI) acquisition is difficult. GNNs offer a promising solution by learning design variables directly from pilot signals, bypassing explicit channel estimation and significantly reducing computational burden compared to traditional AO-based methods. The proposed GNN-based method introduces two unique features. First, while most existing GNN-based methods [12–16] demonstrate GNN's generalizability to the number of UEs, our model is designed with inherent generalization capabilities for *both* the number of UEs and BSs. Second, the proposed GNN includes a built-in mechanism to handle the *oversmoothing* issue [17] inherent in deep GNNs. Oversmoothing refers to the issue where graph node features become increasingly similar after passing through a deep updating structure, which could potentially degrade a GNN's learning capability. We intro-

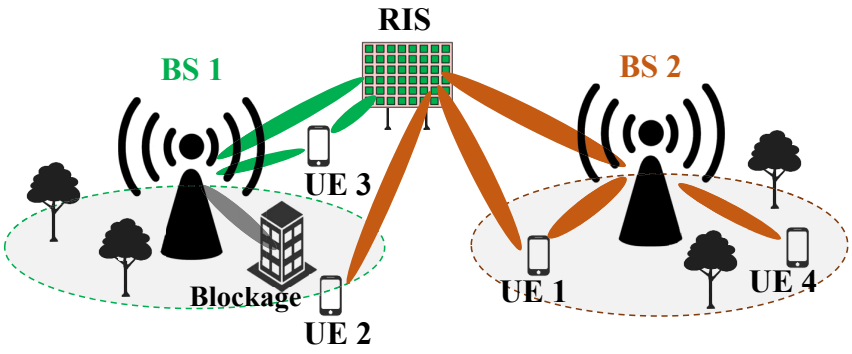


Figure 1.1: An RIS-assisted multi-BS multi-UE system (an example of two BSs and 4 UEs is shown here).

duce an adaptive attention-based aggregation (AAA) mechanism that integrates adaptivity into GNNs, enabling adaptive control of node information exchange to strike a balance between mitigating oversmoothing and maintaining learning effectiveness. This approach is novel to the considered problem and distinguishes it from previous applications such as urban traffic flow forecasting [18] and fraud detection [19]. The first part of the research has been published in [20] and [21].

In the second part of this research, we continue to investigate the RIS-aided UA strategy and turn our attention to mixed near- and far-field systems resulting from extremely large-scale multiple-input multiple-output (XL-MIMO) in Chapter 4. XL-MIMO, employing hundreds or thousands of antennas, is a key sixth-generation (6G) technology for improving spectral efficiency and spatial resolution. XL-MIMO changes the structure of the electromagnetic field and enlarges the near-field region, since the Rayleigh distance scales with the square of the antenna number, leading to the coexistence of near- and far-field channels [22, 23]. In this setting, far-field beamforming can cause energy to spill into adjacent near-field regions, creating the *energy spread effect* (*ESE*), i.e., interference to near-field users at nearby angles. As illustrated in Fig. 1.2, if BS 1 serves UE 2 with far-field beamforming, UE 1 in the adjacent near-field region suffers interference from the ESE.

To mitigate inter-user interference caused by the ESE in mixed near- and far-field sys-

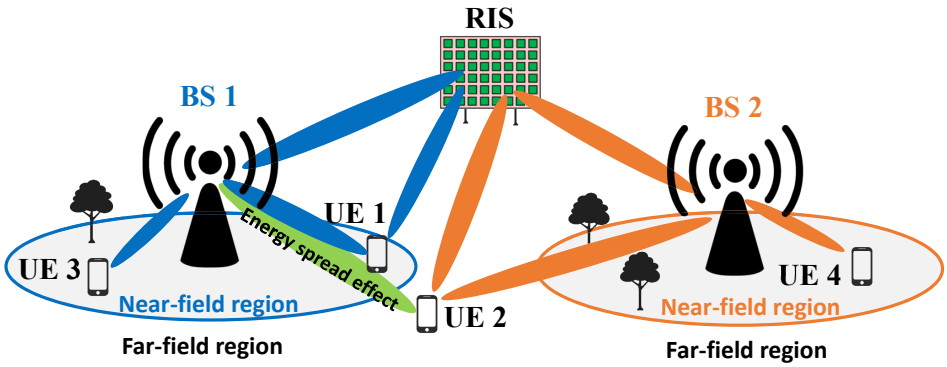


Figure 1.2: A RIS-aided multiuser mmWave system. The blue and orange ellipses denote user associations with BS 1 and BS 2, respectively, while the green area highlights interference to nearby-angle near-field UE 1 if association BS 1–UE 2 were formed instead (energy spread effect).

tems, it is useful to combine RIS technology with judiciously designed user association and beamforming strategies. Fig. 1.1 shows that associating UE 2 with BS 1 creates interference for UE 1 in BS 1's near-field region, which can be mitigated by reassigning UE 2 to BS 2 via RIS. For this problem, we propose to jointly optimize user association, BS beamforming, and RIS phase shifts in RIS-aided XL-MIMO multi-BS multi-UE systems using a GNN-based solution to mitigate energy spread-induced interference. With GNN's permutation invariance and equivariance properties, this GNN-based approach enables effective generalization to diverse system settings (such as numbers of UEs). The second part of the research has been published in [24].

In the final part of this research, we shift our focus to beamforming in cell-free systems. We investigate the integration of Federated Learning (FL) and GNNs, and examine whether such a combination can offer further advantages for beamforming in cell-free systems as presented in Chapter 5. GNNs possess the capability to extract features and learn from graph-structured data composed of nodes and edges. Owing to this capability, they have been widely applied in various networked systems and domains, such as neuroscience, social networks, transportation networks, and wireless communication networks. However, in recent years, the awareness of data privacy has significantly increased, and

users are often reluctant to upload their personal data directly to a centralized network processing unit. FL addresses this concern by distributing learning tasks to client devices and allowing them to upload only the learned feature embeddings or model parameters to the central server. Such an approach helps prevent the leakage of sensitive user data. Consequently, recent studies have proposed integrating GNNs with FL, leading to the development of federated GNNs (FedGNNs). This framework not only enables the processing of graph-structured networks but also preserves data privacy. Therefore, in this final part, we propose an approach that integrates FL and GNNs to address the beamforming problem in cell-free multi-user wireless communication systems. The proposed framework aims to reduce signaling overhead through a decentralized learning scheme while maintaining generalization capability across system-specific parameters, such as the number of users and base stations.

In summary, this research consists of three main parts: GNN for the Joint Design of RIS-aided User Association and Beamforming, GNN for Mixed Near- and Far-Field RIS-aided Systems, and Federated Graph Neural Network for Beamforming in Cell-free MIMO Systems, which are presented in Chapters 3, 4, and 5, respectively.

1.2 Related Works

1.2.1 Related Works for Part I: GNN for the Joint Design of RIS-aided User Association and Beamforming

RIS-assisted UA in multi-BS multi-UE systems have been studied in [25–27] using conventional optimization techniques and alternating optimization (AO) methods. Mei *et al.* [25] proposed a joint design of RIS–UE association and BS power control to balance signal-to-interference-plus-noise ratio (SINR) among UEs in multi-RIS-aided systems. Zhao *et al.* [26] aimed at jointly optimizing BS–UE association, BS power control, and RIS phase shifts in RIS-aided systems. Taghavi *et al.* [27] addressed the joint optimization of BS active beamforming, RIS passive beamforming, and BS–UE association to maximize the sum rate in multi-RIS-aided systems, and proposed an AO-based algorithm

which requires multiple iterations.

The Learning-based methods such as DNNs [28] and deep reinforcement learning (DRL) [29–31] have been proposed for RIS-assisted UA. Amiriara *et al.* [28] proposed a DNN-based approach to jointly design RIS–user association, RIS reflection matrix, and BS beamforming for a system aided by multiple RISs. Aung *et al.* [29] proposed a DRL-based approach to maximize energy efficiency in multi-RIS networks by jointly optimizing RIS–user association, reflective elements ON/OFF states, phase shift, and transmit power. Wang *et al.* [30] proposed a Deep Q-Network (DQN)-based algorithm to jointly optimize BS–user association, computation offloading, resource allocation, and RIS phase shifts for minimizing long-term energy consumption. Zhu *et al.* [31] proposed a soft actor-critic DRL-based approach to jointly optimize the BS–RIS–UE association and RIS passive beamforming, aiming to maximize the sum rate in a RIS-assisted wireless network.

GNN-based approaches have been proposed for designing UA strategies in multi-BS systems [14–16]. Randall *et al.* [14] proposed combining Double Deep Q Networks with GNNs to develop a UA algorithm aimed at maximizing network throughput while ensuring user fairness in small-cell networks. Zhang *et al.* [15] proposed a digital twin architecture integrated with a GNN to jointly optimize UA and power allocation to maximize the average user rate in terahertz multi-cell networks. Ranasinghe *et al.* [16] presented a GNN-based access point (AP) selection algorithm based on path loss and shadow fading for cell-free massive multiple-input multiple-output (MIMO) systems.

GNN-based approaches have been proposed for the joint design of transmit beamforming and RIS phase shifts [8–11]. Jiang *et al.* [8] developed a GNN architecture parameterizing the mapping from received pilots to BS and RIS beamforming without explicit channel estimation. Chen *et al.* [9] proposed a two-phase GNN for sum rate maximization in multi-UE MISO systems with a relay and RIS. Jin *et al.* [10] introduced a model-driven GNN to reduce the complexity of joint BS and RIS beamforming. Yeh *et al.* [11] incorporated angular CSI and graph attention to jointly design BS beamforming and RIS phase shifts for multi-UE MISO mmWave networks.

1.2.2 Related Works for Part II: GNN for User Association and Beamforming in Mixed Near- and Far-Field RIS-aided Systems

Beamforming techniques [32–36] have been investigated for near-field communications in XL-MIMO systems. Specifically, in [32], an angle- and distance-based beamforming for near-field communications was proposed. In [33], a distance-aware beamforming was developed by leveraging the extra degrees of freedom in the near-field region. In [34], hybrid beamforming architectures with subarray allocation strategies were explored. For beamforming with codebook designs, in [35], a spherical-domain codebook was introduced to replace the conventional far-field angle-domain counterpart. In [36], a hierarchical codebook for the passive beamforming was proposed to reduce beam alignment overhead in XL-MIMO systems.

Several studies have examined mixed near- and far-field systems [37–39]. In [37], the ESE was analyzed with respect to BS antenna number and UE locations, showing that precise beamforming can effectively suppress interference. In [38], the sum rate of reconfigurable intelligent surface (RIS)-aided XL-MIMO was evaluated under maximum-ratio transmission and zero-forcing precoding, considering different UE visibility regions. In [39], the supportable number of UEs by a BS was characterized as a function of BS antenna number and transmit power.

Prior studies on the integration of RISs and XL-MIMO [40–42] focused on optimizing BS beamformers and RIS phase shifts for objectives such as UE received signal power [40], fairness and energy efficiency [41], and weighted sum rate [41], using methods such as alternating optimization [40, 42] or manifold optimization [41].

1.2.3 Related Works for Part III: Federated Graph Neural Network for Beamforming in Cell-free MIMO Systems

For beamforming in cell-free MIMO systems, traditional approaches typically rely on a central processing unit (CPU) to determine the beamforming vectors for each base station using available channel state information and optimization algorithms, such as zero-forcing (ZF) [43], weighted minimum mean-square-error (WMMSE) [44], and Ma-

jorization–Minimization [45]. However, in practical systems, the timeliness of channel information must be considered. As the numbers of base stations and users increase, conventional beamforming techniques lead to significant signaling overhead and high computational complexity, resulting in longer processing times.

To reduce computational complexity, several studies have explored machine learning and deep learning approaches for beamforming in cell-free MIMO systems. In [46], M. Zaher et al. considered a power allocation problem based on the WMMSE algorithm and proposed a DNN composed of stacked fully connected layers to learn the optimal power allocation. On the other hand, DRL-based methods do not rely on traditional optimization algorithms for beamforming models. Instead, they learn beamforming strategies through interactions with the system environment. In [47], Y. Al-Eryani et al. employed DRL to jointly design base station clustering and beamforming to maximize the sum capacity. In [48], W. Li et al. focused on maximizing the energy efficiency of cell-free MIMO systems by training a DRL model at the central processing unit to interact with the environment and determine the optimal beamforming for all base stations. However, conventional DNN and DRL approaches generally suffer from limited generalization capability. In other words, when system-specific parameters change, such as the number of users, these models typically require retraining to adapt to the new system configuration. To address this limitation, graph neural networks (GNNs) have emerged as a promising deep learning framework. Owing to their unique model structure, GNN architectures can be designed to generalize across varying system parameters.

Several studies have therefore proposed GNN-based methods for beamforming in cell-free MIMO systems. In [49], L. Salaün et al. addressed the power allocation problem in cell-free MIMO systems. They noted that many existing power allocation algorithms rely on traditional second-order cone programming (SOCP), which often incurs high computational complexity. To mitigate this issue, the authors proposed a GNN-based method that not only reduces computational complexity but also achieves generalization with respect to the numbers of users and base stations. In their proposed GNN architecture, each base station–user pair is modeled as a node, with the channel between the base station and the user used as the node feature. The network learns the transmit power for each

pair to maximize the minimum SINR. In [50], X. Yan et al. considered a joint base station selection and beamforming design problem. They pointed out that centralized GNN approaches require substantial signaling overhead in multi-base-station systems. To address this issue, the authors divided all base stations into several clusters and proposed a decentralized optimization framework for each cluster. In the proposed GNN model, base stations within the same cluster share local channel information as node features, while statistical channel information from other clusters is incorporated as edge features. Simulation results demonstrate that the proposed approach achieves higher sum capacity compared with the traditional fractional programming algorithm. In [51], Y. Ma et al. proposed a GNN-based method for beamforming in cell-free MIMO systems that uses channel state information as input. The authors further introduced a sparsification mechanism for the input channel information, aiming to remove less important base station–user links and focus the learning process on the most relevant channel information. This strategy reduces computational complexity while maintaining performance; simulation results show that the proposed method significantly reduces complexity while achieving only a slight degradation in sum capacity compared with the WMMSE approach. In [52], Y. Zhao et al. transformed the beamforming problem in cell-free MIMO systems into a power allocation problem by adopting maximum ratio beamforming. They then proposed a GNN-based algorithm to solve the resulting power allocation problem. The proposed GNN model achieves generalization with respect to the number of users, and simulation results demonstrate strong generalization performance by evaluating the model under different user numbers during training and testing. However, the beamforming methods for cell-free MIMO systems mentioned above are all centralized approaches, where the generation of beamforming vectors relies on computations performed at a CPU. In a cell-free MIMO system, this requires the CPU to collect signals containing channel state information from all base stations. In systems with a large number of base stations and users, this process results in substantial signaling overhead.

To address this issue, several studies have explored decentralized beamforming methods for cell-free MIMO systems. In decentralized beamforming, each base station determines its own beamforming vector locally, with the goal of reducing the signaling

overhead required for beamforming generation. In [53], I. Atzeni et al. proposed a decentralized beamforming design for cell-free MIMO systems to minimize the system minimum mean-square error (MMSE). To enable cooperation among base stations, the authors introduced an over-the-air signaling mechanism, where signal exchanges between base stations and users allow base stations to share cross-term information, which is then used to compute the beamforming vectors. In [54], F. Fredj et al. proposed a multi-agent DRL algorithm for beamforming design in cell-free MIMO systems. In this framework, each base station is modeled as an agent that collects local information and generates its own beamforming strategy. During training, the policies of all agents are jointly updated at the central processing unit to maximize the sum capacity.

As one of the most representative decentralized learning paradigms, FL has also been applied to beamforming design in cell-free MIMO systems. In FL, models are deployed at local nodes and trained using local data, while global objectives are achieved through model or gradient aggregation combined with local training. In [55], D. Wang et al. studied beamforming in cell-free MIMO systems by training DNN models at individual base stations to generate their respective beamforming vectors. The authors proposed two federated learning strategies. In the first strategy, beamforming labels are obtained using the WMMSE algorithm, and aggregated channel information from all base stations is provided to assist local model training. In the second strategy, the central processing unit defines a negative sum-capacity loss function, which is used to guide the training of the local models at each base station. Similarly, in [56], D. Yu et al. deployed DNN models at individual base stations to generate beamforming vectors, with the objective of maximizing the sum capacity of the cell-free MIMO system. In their federated learning framework, the DNN models at each base station are combined with the WMMSE algorithm, and aggregated channel information from all base stations is shared via the central processing unit during training to assist local model updates.

1.3 Motivation

1.3.1 Motivation of Part I: GNN for the Joint Design of RIS-aided User Association and Beamforming

For the jointly optimization problem of UA, transmit BF, and RIS phase shifts in a RIS-assisted multi-BS multi-UE mmWave system, conventional optimization and AO-based algorithms have high complexity and long execution times.

To overcome these limitations, the Learning-based methods as proposed in [28–31] are proposed for RIS-assisted UA. Nevertheless, these approaches lack generalizability across different system configurations, such as varying numbers of UEs. Specifically, the number of parameters in the output layer of a DNN or the action network of a DRL depends on the number of UEs and/or BSs. As a result, any change in the number of UEs or BSs necessitates retraining the DNN or DRL model, limiting their adaptability to dynamic network settings.

GNN-based approaches for UA have been proposed in [14–16]. However, these works did not consider RIS-aided scenarios or incorporate load balancing into their UA strategy designs. While user association in RIS-aided systems was investigated in [57], the GNN-based RIS–UE association strategy proposed for single-BS multi-UE systems cannot be directly applied to multi-BS scenarios.

There are GNN-based approaches proposed for the joint design of transmit beamforming and RIS phase shifts [8–11]. However, these works focused on single-BS systems and cannot be directly extended to multi-BS scenarios or to problems involving UA which is specific to multi-BS settings.

Therefore, the first research part is to address this research gap by jointly optimizing UA, transmit BF, and RIS phase shifts in a RIS-assisted multi-BS multi-UE mmWave system with BS load balancing considerations, through a GNN-based approach.

1.3.2 Motivation of Part II: GNN for User Association and Beamforming in Mixed Near- and Far-Field RIS-aided Systems

The existing works [32–36] have investigated the near-field Beamforming techniques in XL-MIMO systems. However, these works focused primarily on near-field scenarios and did not explicitly address the mixed near- and far-field environment inherent to XL-MIMO systems. In the literature, investigation on the mixed near- and far-field systems has been presented in [37–39]. Nevertheless, theses works did not explicitly address ESE suppression.

Several works [40–42] have been proposed for the optimization of BS beamformers and RIS phase shifts for the RIS-aided XL-MIMO systems. However, these works neither explicitly considered RISs as an enabler for energy spread suppression nor developed beamforming and user association-native solutions to it. Besides, they relied on computationally intensive optimization approaches.

To bridge this research gap, this dissertation jointly optimizes user association, BS beamforming, and RIS phase shifts in RIS-aided XL-MIMO multi-BS multi-UE systems using a GNN-based solution tailored to mitigate energy spread-induced interference.

1.3.3 Motivation of Part III: Federated Graph Neural Network for Beamforming in Cell-free MIMO Systems

For the beamforming problem in cell-free multi-user wireless communication systems, early studies primarily relied on traditional optimization algorithms, such as the WMMSE and Majorization–Minimization methods [44,45]. However, these conventional optimization algorithms often suffer from high computational complexity. To address this issue, several works have explored deep learning-based approaches. For instance, [46] employed a conventional DNN, while [47, 48] adopted DRL algorithms. Nevertheless, conventional deep learning approaches often encounter generalization limitations. In multi-user systems, the number of users may frequently change. Once the number of users varies, these traditional deep learning models typically require retraining in order to adapt to the new system configuration.

To address this limitation, GNNs, as a branch of deep learning, have attracted significant attention. Owing to their unique architecture, GNNs can be designed to generalize with respect to certain system parameters, thereby reducing the time and computational cost required for retraining. Consequently, several works, such as [49–52], have adopted GNN-based approaches to address the beamforming problem in cell-free multi-user wireless communication systems. However, the aforementioned methods for beamforming in cell-free systems are still centralized approaches. Specifically, these methods require each base station to transmit the information necessary for optimization to a CPU, where the optimization problem is solved and the resulting beamforming vectors are then sent back to the base stations. Such a process incurs significant signaling overhead. Moreover, the CPU must handle the beamforming optimization for all base stations simultaneously, which results in a substantial computational burden.

To mitigate these issues, several studies have explored decentralized approaches [53–56], where the optimization tasks are distributed to individual base stations. The main objective of these methods is to reduce the signaling overhead associated with centralized processing. Among decentralized learning paradigms, FL is particularly attractive due to its emphasis on data privacy preservation. In an FL-based framework, base stations are treated as FL clients, and each base station only accesses locally available data, such as its own channel information. Channel information is not shared among base stations, which also reduces the signaling overhead associated with information exchange.

For wireless communication systems, reducing the time required to transmit information to the CPU is highly important. The key information for beamforming optimization is CSI or signals that implicitly contain CSI. In practice, beamforming algorithms must consider the timeliness of channel information, since wireless channels may vary rapidly and CSI therefore has a limited validity period. Consequently, reducing signaling delay allows more time for the optimization algorithm to compute beamforming solutions, thereby improving the practical feasibility of the algorithm.

In FL, the goal is to obtain a global model for beamforming optimization. Each base station trains a local deep learning model using its own data and only transmits model parameters or gradients to the CPU. The CPU then performs model aggregation to obtain

the global model. Compared with traditional centralized approaches, this framework can significantly reduce the amount of data transmitted to the CPU and alleviate the computational burden at the CPU. However, existing decentralized approaches for beamforming in cell-free systems generally do not address the issue of generalization. For example, when the number of users changes, retraining of the FL model is often unavoidable. The time cost associated with retraining may offset the advantage of reduced signaling overhead brought by decentralized approaches.

Therefore, in this part of the research, we aim to address the beamforming problem in cell-free multi-user wireless communication systems by proposing a method that integrates FL and GNNs. The proposed approach aims to retain the advantage of decentralized methods in reducing signaling overhead while simultaneously achieving generalization across system parameters, such as the number of users and the number of base stations.

1.4 Contributions of the Research

The main contributions of the first research part are summarized as follows:

- We propose a GNN-based approach to jointly optimize UA, BS transmit beamforming, and RIS phase shifts in RIS-aided multi-BS multi-UE systems, while satisfying load-balancing constraints. The proposed method avoids the need for explicit channel estimation by learning directly from pilot signals. It demonstrates generalizability to varying numbers of UEs *and* BSs, enabling low-complexity training with few UEs and BSs and allowing the direct use of shared, pre-trained GNN modules for newly added BS–UE links without requiring retraining. The proposed GNN model employs unsupervised training to maintain low complexity.

- We propose a built-in mechanism called AAA within the proposed GNN model, consisting of two components: 1) sparse attention for aggregation and 2) tuning aggregation for combination. This mechanism mitigates oversmoothing in deep GNNs by refining the node feature aggregation process, with adaptive control to

balance information exchange and optimize learning performance, while adding only modest complexity relative to the overall model.

- Simulation results show that the proposed scheme achieves superior sum-rate performance compared to both optimization-based and learning-based benchmarks, with favorable complexity, high generalizability, robustness to pilot correlation, and proven effectiveness in mitigating oversmoothing. A detailed complexity analysis of the proposed scheme is also presented.

The main contributions of the second research part are summarized as follows:

- A GNN-based approach is proposed to tackle the high-dimensional nonconvex design problem in mixed near- and far-field systems. The framework incorporates correlation-based attention (CBA) aggregation and near-field array alignment (NAA) enhanced training to explicitly handle the ESE, where CBA helps identify association pairs that may cause energy spread, and NAA-enhanced training strengthens near-field UE signals to mitigate this effect.

- Simulation results show that the proposed GNN outperforms model- and codebook-based benchmarks in sum rate, with increasing advantages under intensified energy spread conditions. The effectiveness of the CBA and NAA mechanisms is further demonstrated by visualizations showing more flexible and efficient user association.

The main contributions of the third research part are summarized as follows:

- A method that integrates FL and GNNs is proposed to address the beamforming problem in cell-free multi-user wireless communication systems. The proposed approach aims to reduce signaling overhead while achieving generalization on the number of UEs.

- Simulation results show that the proposed federated GNN can have a comparable performance with the centralized-based beamforming benchmark in sum rate. It is further shown that the proposed federated GNN framework achieves reduced signaling overhead relative to the centralized beamforming benchmark.

1.5 Dissertation Organization

The structure of this dissertation is as follows: Chapter 2 presents the background of GNN. Chapter 3 presents the main research, GNN-based method for the joint design of RIS-aided user association and beamforming. Chapter 4 introduces an extension of the research, GNN-based method for user association and beamforming in mixed near- and far-field RIS-aided systems. Chapters 5 demonstrates the third research part, the federated GNN for beamforming in cell-free MIMO systems. Finally, the conclusion of the dissertation and future works are provided in Chapters 6.

1.6 Notations

$\mathbf{a}^*$, $\mathbf{a}^T$, and $\mathbf{a}^H$ represent the conjugate, transpose, and conjugate transpose of vector $\mathbf{a}$, respectively. $[\mathbf{a}]_i$ represents the ith entry of vector $\mathbf{a}$, and $[\mathbf{a}]_{i:j}$ represents the ith entry to the jth ($j > i$) entry of vector $\mathbf{a}$. $\Re\{\cdot\}$ and $\Im\{\cdot\}$ denote the real and imaginary parts of the argument, respectively. $|\cdot|$, $\|\cdot\|_0$, $\|\cdot\|$, and $\|\cdot\|_F$ denote the absolute value, 0-norm, 2-norm, and Frobenius norm, respectively. $\text{diag}\{a_1, a_2, \ldots, a_N\}$ represents a diagonal matrix with terms $a_1, a_2, \ldots, a_N$ on the diagonal. $\lfloor\cdot\rfloor$ denotes the floor operation.

Chapter 2

Preliminary

2.1 Introduction to Graph Neural Network (GNN)

Graph Neural Networks (GNNs) are a branch of deep learning designed to learn from data that can be represented in the form of graphs. The term "graph" here does not refer to images or visual data in the conventional sense, but rather to structures composed of nodes and edges. Many real-world problems can be modeled as graphs, such as wireless communication networks, social networks, molecular structures, and transportation networks. GNNs are capable of learning the relationships among different entities in a graph, thereby enabling the derivation of effective problem-solving strategies [58]. This distinguishes GNNs from other deep learning approaches, such as convolutional neural networks (CNNs), which are designed for grid-structured data like images, and recurrent neural networks (RNNs), which are tailored for sequential data.

GNNs are a promising approach to address generalizability challenges [8,9,59]. GNNs operate on a graph structure through two key processes: aggregation, where nodes exchange information, and combination, where gathered information is integrated and updated via neural networks. Through aggregation and combination of information of nodes, GNN has the potential to achieve effective learning outcomes for communication problems. Moreover, GNN's *permutation invariance and equivariance properties* enable effective generalization to diverse system settings (such as numbers of UEs and BSs) without the need for retraining the embedded model. Permutation invariance means that the

overall output remains unchanged regardless of node ordering, while permutation equivariance means that permuting the input results in the corresponding permutation of the output.

The learning process of a GNN is carried out based on the graph of interest. It extracts features from node or edge information and enables information exchange among nodes. By leveraging both the intrinsic information of each node and the information received from other nodes, a GNN can learn the underlying complex relationships among nodes in the graph. The learning mechanism of a GNN mainly consists of two components: aggregation and combination [1], as illustrated in Fig. 2.1. Aggregation refers to the process of information exchange among nodes, where each node communicates with its neighboring nodes. Here, neighboring nodes are defined as those connected by edges. After aggregation, each node performs combination, where it integrates its own information with the aggregated information to obtain updated node representations. The combination step is typically implemented by a neural network that takes both the node's own features and the aggregated features as inputs, for example, a neural network composed of stacked fully connected layers.

GNN learning is generally node-centric. For instance, in a wireless communication system where one base station serves multiple users, users can be modeled as nodes, and a GNN can be employed to learn the required transmit power for each user. One key advantage of GNNs is that all nodes can share the same set of neural network parame-

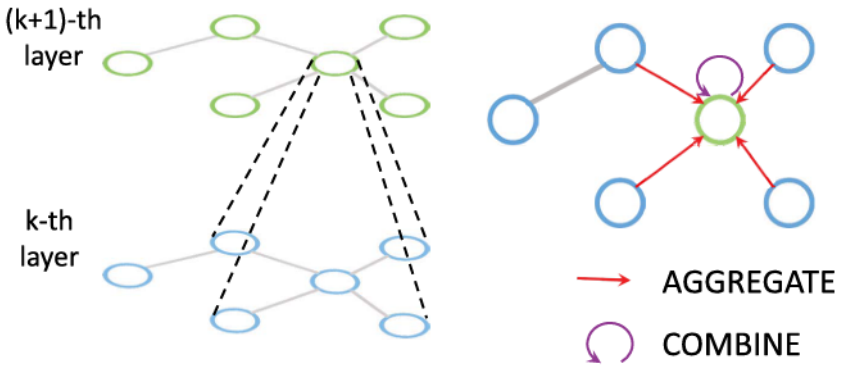


Figure 2.1: The aggregation and combination in a GNN [1].

ters. Specifically, neural networks are used in the aggregation step to process exchanged information and in the combination step to update node representations. Due to this node-centric learning structure, the neural networks in the architecture can be shared across all nodes. This not only reduces computational complexity but also enables GNNs to adapt to varying numbers of nodes. For example, when the number of users (treated as nodes) changes, the GNN does not require retraining, which can be regarded as achieving generalization with respect to the number of users.

2.2 Theoretical Basis of GNN

The theoretical foundation of GNNs is rooted in graph theory and representation learning. A graph is typically denoted as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ represents the set of nodes and $\mathcal{E}$ denotes the set of edges. Each node is associated with a feature vector, and the connectivity between nodes is described by an adjacency matrix. The key idea of GNNs is to learn node representations by iteratively aggregating information from neighboring nodes, a process commonly referred to as message passing. Through multiple layers of aggregation and transformation, GNNs are able to capture both local and global structural information of the graph.

A fundamental property of GNNs is their ability to handle the permutation invariance of graph data. Since there is no inherent ordering of nodes in a graph, GNNs employ symmetric aggregation functions, such as summation or averaging, to ensure that the learned representations are independent of node ordering. This property is essential for maintaining consistency and robustness across different graph representations. Two essential properties of the GNNs is *permutation equivariance* and *permutation invariance*. First, GNNs are inherently order-agnostic with respect to nodes. For a given graph, relabeling the nodes should not change the underlying problem. For example, in a wireless communication system where one base station serves multiple users and transmit power allocation is considered, users are modeled as nodes. Changing the user indices should not affect the overall transmit power allocation strategy obtained. This order-agnostic nature of GNNs is reflected in the properties of permutation equivariance and permutation

invariance.

Permutation equivariance means that if the order of input nodes is permuted, the output of the GNN is permuted in the same way. To achieve permutation equivariance, GNNs employ order-invariant functions, such as sum, mean, or max pooling, particularly in the aggregation step. As a real-world example, consider analyzing the molecular structure of a chemical compound: relabeling atoms does not change the properties of the molecule. If atom 1 is relabeled as atom 2, the corresponding GNN output for that atom should remain unchanged. Similarly, in a wireless communication system, consider the transmit power allocation problem where users are treated as nodes. If user 1 is assigned a transmit power of 1 dBm by the GNN, then after relabeling user 1 as user 2, the corresponding GNN output for that user should still be 1 dBm.

Permutation invariance, on the other hand, means that if the order of input nodes is permuted, the overall output of the GNN remains exactly the same. To ensure permutation invariance, the functions used in the GNN must also be order-independent. For example, in determining whether a chemical compound is toxic, relabeling atoms does not alter the molecular properties; thus, the GNN's output regarding toxicity should remain unchanged. In a wireless communication system, consider again the transmit power allocation problem. If the output represents the total transmit power of a base station and is, for instance, 20 dBm, then this result should remain 20 dBm regardless of how the users (nodes) are relabeled.

A GNN that satisfies both permutation equivariance and permutation invariance can handle arbitrary permutations of node indices without requiring retraining. Combined with the fact that all nodes in a GNN share the same set of neural network parameters, this enables strong generalization capability. This property is particularly important in wireless communication systems. For example, if a GNN can generalize over the number of users, it can adapt to dynamic changes in user population without retraining, thereby significantly reducing the computational cost and time required for retraining.

From a theoretical perspective, GNNs can be broadly categorized into spectral-based and spatial-based approaches. Spectral methods are grounded in graph signal processing and define convolution operations in the Fourier domain using the graph Laplacian. In

contrast, spatial methods directly perform convolution in the node domain by aggregating information from local neighborhoods. While spectral approaches provide strong mathematical foundations, spatial methods are generally more scalable and widely adopted in practical applications.

The expressive power of GNNs has been extensively studied and is closely related to the Weisfeiler–Lehman test for graph isomorphism. It has been shown that many standard GNN architectures are at most as powerful as the one-dimensional Weisfeiler–Lehman test in distinguishing graph structures. This limitation has motivated the development of more expressive models that can capture higher-order structural information.

Despite their success, GNNs face several theoretical challenges. One notable issue is oversmoothing, where node representations become indistinguishable as the number of layers increases. We have proposed a solution to oversmoothing incorporated into the GNNs, which is presented in Chapter 3. Another challenge is oversquashing, which refers to the difficulty of propagating information from distant nodes through a limited embedding space. These challenges highlight the need for careful architectural design and further theoretical advancements.

In recent years, GNNs have gained significant attention in wireless communications, where network entities such as users, base stations, and links can be naturally modeled as graphs. The message passing mechanism of GNNs aligns well with the modeling of interference, cooperation, and resource allocation problems. As a result, GNN-based approaches have shown great potential in addressing complex optimization problems in next-generation communication systems.

2.3 GNN for Communications

In wireless communication networks, base stations and users can be regarded as nodes. Alternatively, the links between base stations and users can also be treated as nodes [60]. In other words, the graph can be designed according to the specific problem under consideration, as illustrated in Fig. 2.2. In this figure, three communication scenarios are represented using graphs, including device-to-device (D2D) networks, cell-free networks, and

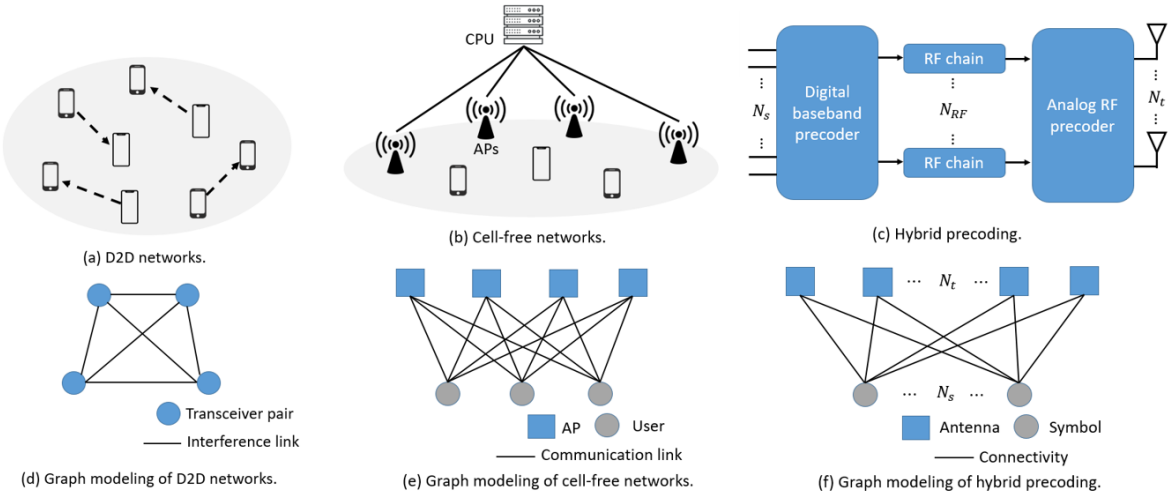


Figure 2.2: Graph representation of the various communication issues.

hybrid beamforming.

In D2D networks, the goal of generalization is typically at the level of transceiver pairs. Therefore, each transceiver pair is modeled as a node. Since transceiver pairs may cause interference to each other, the connections between nodes (i.e., edges) are defined as interference links. Under this formulation, the GNN can jointly learn information from both nodes and edges. For example, in learning power control for transceiver pairs, it is necessary not only to capture the channel information of each transceiver pair but also to learn the interference relationships among them to effectively coordinate interference.

In cell-free networks, access points (APs) or base stations (BSs) can be modeled as one type of node, while users are treated as another type of node, forming a heterogeneous graph. The distinction between homogeneous and heterogeneous graphs lies in whether the nodes belong to a single type or multiple types. A homogeneous graph contains only one type of node, whereas a heterogeneous graph consists of multiple types of nodes. For instance, the D2D network described above corresponds to a homogeneous graph, while the cell-free network considered here is modeled as a heterogeneous graph.

In general, nodes are defined based on the entities involved in the optimization problem or the target of generalization. Entities that carry essential information for the optimization task can be modeled as nodes or edges. Depending on these design choices,

the resulting graph may naturally be either homogeneous or heterogeneous. There is no fixed rule for choosing between the two; rather, it depends on how the designer utilizes the available system information and leverages the GNN to learn from it. Although the figure illustrates APs/BSs and users as different types of nodes, an alternative design is to treat each BS–user link as a node, as will be demonstrated in Chapter 3. This is motivated by the fact that the available information comes from uplink pilots transmitted from each user to each BS. In this design, the pilot information of each BS–user link is independently learned at each node while allowing information exchange among nodes. Meanwhile, the GNN architecture is designed to ensure generalization with respect to both the number of BSs and users. More details are provided in Chapter 3.

The final example shown in the Fig. 2.2 is hybrid beamforming. In this case, antennas and transmitted symbols are modeled as nodes. This formulation enables generalization over both the number of antennas and the number of transmitted symbols, while allowing the model to independently learn their respective information. For instance, antenna nodes encode channel information, whereas symbol nodes represent the transmitted symbols. Based on this design, the GNN can learn both the analog precoder and the digital baseband precoder for hybrid beamforming.

As shown above, In wireless communication systems, many problems naturally exhibit graph structures—for example, users, base stations, and devices can be modeled as nodes, while interference links, channel gains, or cooperation relationships form the edges. This makes GNNs particularly suitable for modeling complex and large-scale communication networks such as D2D networks, cell-free systems, and heterogeneous networks. Specifically, the key advantages of the GNNs are shown as follows to explain why GNNs are Suitable for Wireless Communications. 1) Interference-aware modeling: Wireless networks are fundamentally interference-limited. GNNs can explicitly capture interference relationships through edges, enabling more accurate system modeling. 2) Scalability and generalization: GNNs share parameters across nodes and edges, allowing them to generalize to networks of different sizes and topologies without retraining. 3) Distributed implementation: Message-passing mechanisms in GNNs align well with decentralized communication architectures, making them suitable for distributed resource

allocation and control. 4) Permutation invariance/equivariance: The output of a GNN does not depend on node ordering, which is important because communication networks do not have a fixed ordering of users or devices. 5) Low computation complexity: The conventional optimization methods typically rely on iteration processes and result in high computation complexity, while the GNNs as a machine learning-based method have a lower computation complexity once they are well-trained. That implies that the GNNs are more practical than the conventional optimization methods from a real-time implementation perspective.

Chapter 3

GNN for the Joint Design of RIS-aided User Association and Beamforming

In this chapter, the proposed GNN-based approach for the joint design of user association and beamforming in a RIS-aided multi-BS multi-UE system is introduced. We investigate the extent to which RIS can provide flexibility in user association through a GNN. The system model, problem formulation, methodology, and simulation results are presented in sequence.

3.1 System Model

In this section, we present the system model as shown in Fig. 3.1. The symbols used in this research part are summarized in Table 3.1.

We consider a RIS-aided downlink mmWave communication system with B BSs, each equipped with N_T transmit antennas, and U single-antenna UEs, as illustrated in Fig. 1.1. The sets of all BSs and UEs are denoted by $\mathcal{B} = \{1, 2, \ldots, B\}$ and $\mathcal{U} = \{1, 2, \ldots, U\}$, respectively. The RIS is composed of M passive reflecting elements with adjustable continuous phase terms $\{\phi_1, \ldots, \phi_M\} \triangleq \{e^{j\theta_1}, \ldots, e^{j\theta_M}\}$, which rotates the incident signals and can be formulated as the diagonal phase matrix

$$\mathbf{\Phi} = \operatorname{diag}\{\phi_1, \phi_2, \ldots, \phi_M\}. \tag{3.1}$$

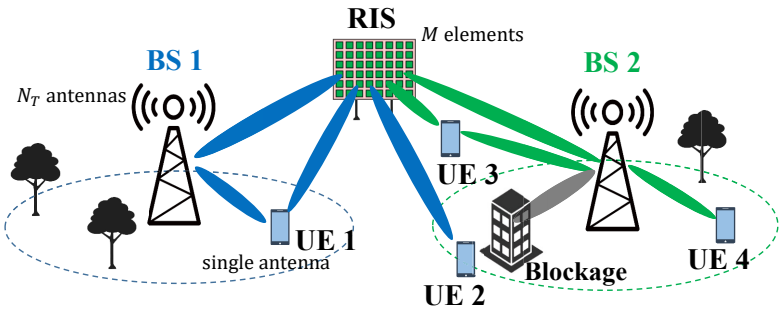


Figure 3.1: An RIS-assisted multi-BS multi-UE system.

The indicator of the BS–UE association is defined as $\delta_{b,u} \in \{0, 1\}$, for $u \in \mathcal{U}$ and $b \in \mathcal{B}$, where $\delta_{b,u} = 1$ represents that the uth UE is served by the bth BS, and $\delta_{b,u} = 0$ otherwise. $\mathcal{S}_u^{\mathrm{BS}} = \{b \mid \delta_{b,u} = 1, b \in \mathcal{B}\}$ is the set of BSs serving the uth UE, whereas the set of UEs served by the bth BS is denoted by $\mathcal{C}_b^{\mathrm{UE}} = \{i \mid \delta_{b,i} = 1, i \in \mathcal{U}\}$.

To model the rich scattering properties of the mmWave channel, we adopt the clustered channel model [61]. The channel between the bth BS and the RIS, denoted by $\mathbf{H}_{b,R} \in \mathbb{C}^{M \times N_T}$, the channel between the RIS and the uth UE, denoted by $\mathbf{h}_{R,u} \in \mathbb{C}^{M \times 1}$, and the channel between the bth BS and the uth UE, denoted by $\mathbf{h}_{b,u} \in \mathbb{C}^{N_T \times 1}$, are defined as follows:

$$\mathbf{H}_{b,R} = \sqrt{\frac{N_T M}{N_\gamma}} \rho_{b,R} \sum_{\ell=1}^{N_{cl}} \sum_{i=1}^{L_\ell} \alpha_{\ell,i} \mathbf{a}_{\mathrm{RIS}} \left(\psi_{\ell,i}^{R,r}, \theta_{\ell,i}^{R,r}\right) \times \mathbf{a}_{\mathrm{BS}}^H \left(\psi_{\ell,i}^{b,t}, \theta_{\ell,i}^{b,t}\right), \tag{3.2}$$

$$\mathbf{h}_{R,u} = \sqrt{\frac{M}{N_\gamma}} \rho_{R,u} \sum_{\ell=1}^{N_{cl}} \sum_{i=1}^{L_\ell} \alpha_{\ell,i} \mathbf{a}_{\mathrm{RIS}} \left(\psi_{\ell,i}^{R,t}, \theta_{\ell,i}^{R,t}\right), \tag{3.3}$$

$$\mathbf{h}_{b,u} = \sqrt{\frac{N_T}{N_\gamma}} \rho_{b,u} \sum_{\ell=1}^{N_{cl}} \sum_{i=1}^{L_\ell} \alpha_{\ell,i} \mathbf{a}_{\mathrm{BS}} \left(\psi_{\ell,i}^{b,t}, \theta_{\ell,i}^{b,t}\right), \tag{3.4}$$

where $N_\gamma = \sum_{\ell=1}^{N_{cl}} L_\ell$, and $\alpha_{\ell,i}$ is the complex gain of the ith ray in the ℓth cluster, which are assumed to be Gaussian distributed with zero mean and unit variance. $\rho_{b,R}$, $\rho_{R,u}$, and $\rho_{b,u}$ are the path loss of the channels $\mathbf{H}_{b,R}$, $\mathbf{h}_{R,u}$, and $\mathbf{h}_{b,u}$, respectively. The azimuth and elevation angles of departure (AoD) are denoted by $\psi_{\ell,i}^{b,t}, \theta_{\ell,i}^{b,t}$ for the bth BS and $\psi_{\ell,i}^{R,t}, \theta_{\ell,i}^{R,t}$

for the RIS. The azimuth and elevation angles of arrival (AoA) are denoted by $\psi_{\ell,i}^{R,r}, \theta_{\ell,i}^{R,r}$ for the RIS. The array response vector for each BS, configured with a uniform linear array aligned parallel to the x-axis, is expressed as $[\boldsymbol{a}_{\mathrm{BS}}]_{n_t}(\psi,\theta) = e^{j\frac{2\pi n_t d^{\mathrm{BS}}}{\lambda_c}\cos(\psi)\cos(\theta)}$ for $n_t = 1,\ldots,N_T$, where we set the antenna spacing $d^{\mathrm{BS}} = \lambda_c/2$. The array response vector for the RIS, configured with a uniform rectangular array placed on the yz-plane, is expressed as $[\boldsymbol{a}_{\mathrm{RIS}}(\psi,\theta)]_n = e^{j\frac{2\pi d^{\mathrm{RIS}}}{\lambda_c}(n_1\sin(\psi)\cos(\theta)+n_2\sin(\theta))}$, where $n_1 = \mathrm{mod}(n-1,\sqrt{M})$, $n_2 = \lfloor (n-1)/\sqrt{M} \rfloor$, and we set the RIS element spacing $d^{\mathrm{RIS}} = \lambda_c/2$.

The received signal of the uth UE is given by

$$\begin{aligned} y_u = &\sum_{b\in\mathcal{S}_u^{\mathrm{BS}}} \left(\mathbf{h}_{b,u}^H + \mathbf{h}_{R,u}^H \boldsymbol{\Phi} \mathbf{H}_{b,R}\right) \mathbf{f}_{b,u} s_u \\ &+ \sum_{b\in\mathcal{S}_u^{\mathrm{BS}}} \sum_{v\in\mathcal{C}_b^{\mathrm{UE}},\, v\neq u} \left(\mathbf{h}_{b,u}^H + \mathbf{h}_{R,u}^H \boldsymbol{\Phi} \mathbf{H}_{b,R}\right) \mathbf{f}_{b,v} s_v \\ &+ \sum_{\ell\in\mathcal{B}\setminus\mathcal{S}_u^{\mathrm{BS}}} \sum_{v\in\mathcal{C}_\ell^{\mathrm{UE}}} \left(\mathbf{h}_{\ell,u}^H + \mathbf{h}_{R,u}^H \boldsymbol{\Phi} \mathbf{H}_{\ell,R}\right) \mathbf{f}_{\ell,v} s_v + \mathrm{n}_u, \end{aligned} \tag{3.5}$$

where s_u is the uth UE's data symbol with $\mathbb{E}(s_u) = 0$, $\mathbb{E}(|s_u|^2) = 1$, and $\mathbb{E}(s_u^H s_v) = 0, \forall v \neq u$, $\mathbf{f}_{b,u} \in \mathbb{C}^{N_T\times 1}$ is the beamformer at the bth BS for the uth UE, and n_u is the additive white Gaussian noise (AWGN) with zero mean and variance σ^2. The second and third terms on the right-hand side of (3.5) correspond to the intra-cell and inter-cell interference, respectively. The SINR of the uth UE can be derived as

$$\gamma_u = \frac{\left|\sum_{b\in\mathcal{S}_u^{\mathrm{BS}}} \left(\mathbf{h}_{b,u}^H + \mathbf{h}_{R,u}^H \boldsymbol{\Phi} \mathbf{H}_{b,R}\right) \mathbf{f}_{b,u}\right|^2}{\sum_{i\in\mathcal{U},i\neq u} \left|\sum_{b\in\mathcal{S}_i^{\mathrm{BS}}} \left(\mathbf{h}_{b,u}^H + \mathbf{h}_{R,u}^H \boldsymbol{\Phi} \mathbf{H}_{b,R}\right) \mathbf{f}_{b,i}\right|^2 + \sigma^2}. \tag{3.6}$$

Then, the achievable rate of the uth UE is expressed as

$$R_u = \log_2(1+\gamma_u). \tag{3.7}$$

3.2 Problem Description

In the considered RIS-aided multi-BS multi-UE mmWave system, BS–UE association (user association), transmit beamforming, and RIS phase shifts all have significant impacts on the system performance and are highly interdependent. Therefore, their joint

design is essential. Moreover, ensuring BS load balancing is vital for improving the overall user communication experience and facilitating carbon-aware network operation for sustainable sixth-generation (6G) systems [62]. Our objective is to jointly optimize user association, transmit beamforming, and RIS phase shifts to maximize the sum rate while satisfying the BS load balancing constraint. The problem is mathematically formulated as follows:

$$\max_{\{\delta_{b,u}\},\{\mathbf{f}_{b,u}\},\mathbf{\Phi}} \quad \sum_{u\in\mathcal{U}} R_u \tag{3.8a}$$

$$\text{s.t.} \quad \delta_{b,u} \in \{0,1\}, \quad \forall b \in \mathcal{B},\, u \in \mathcal{U}, \tag{3.8b}$$

$$|\phi_m| = 1, \quad \forall m \in \{1,\ldots,M\}, \tag{3.8c}$$

$$\sum_{b\in\mathcal{B}} \delta_{b,u} = 1, \quad \forall u \in \mathcal{U}, \tag{3.8d}$$

$$\sum_{u\in\mathcal{C}_b^{\mathrm{UE}}} \delta_{b,u} \le Z_{\max,b}, \quad \forall b \in \mathcal{B}. \tag{3.8e}$$

$$\sum_{u\in\mathcal{U}} \delta_{b,u} \|\mathbf{f}_{b,u}\|^2 \le P_{\max,b}, \quad \forall b \in \mathcal{B}. \tag{3.8f}$$

Here, (3.8b) enforces a binary BS–UE association indicator, and (3.8c) imposes the unit-modulus constraint on RIS phase shifts. This work considers the case where each UE can be served by only one BS, as specified by (3.8d). The load-balancing constraint in (3.8e) limits the number of UEs served by the bth BS to $Z_{\max,b}$, and (3.8f) enforces the transmit power constraint, with $P_{\max,b}$ denoting the maximum transmit power allowed at the bth BS.

This complex and non-convex problem poses challenges for conventional optimization and AO-based algorithms, which are often computationally intensive and time-consuming. Similarly, traditional DNN and DRL methods face scalability limitations. To address these challenges, we develop a GNN model with inherent scalability and complexity advantages. The existing GNN-based methods for designing transmit beamformers and RIS phase shifts, such as those in [8–11], cannot be directly applied to joint optimization involving user association due to the additional tightly coupled variables. Besides, they did not incorporate a mechanism in their solution frameworks to address the oversmoothing issue inherent in multi-layer GNN architectures. We are therefore motivated to propose

a novel GNN model to solve Problem (3.8), with built-in mechanisms to mitigate over-smoothing, as presented in the next section.

3.3 Methodology

In this section, we present the proposed GNN with adaptive attention-based aggregation method, termed GNN-AAA, to jointly optimize user association, transmit beamforming, and RIS phase adjustments for solving Problem (3.8), as detailed below.

3.3.1 Preliminary

3.3.1.1 Representation of the Considered System

We consider a fully-connected graph consisting of one RIS node and $B \times U$ BS–UE nodes, which represent the RIS and the transmission links between B BSs and U UEs, respectively, as depicted in Fig. 3.2. Node features are exchanged and updated across multiple GNN layers, enabling the model to learn the system's channel and interference information. Specifically, RIS nodes learn the phase shifts, while BS–UE nodes learn the user association and beamformers.

3.3.1.2 Input Data

The inputs of the proposed GNN model are the decorrelated received uplink pilot signals at all the BSs. Let the pilot transmitted by the uth UE be $\mathbf{x}_u \in \mathbb{C}^{L\times 1}$, where L is the length of each pilot sequence and $||\mathbf{x}_u||^2 = 1$. Then, the $N_T \times L$ received pilot matrix observed at the bth BS is given by

$$\mathbf{Y}_b = \sum_{u\in\mathcal{U}} \left(\mathbf{h}_{b,u}^H + \mathbf{h}_{R,u}^H \mathbf{\Phi} \mathbf{H}_{b,R}\right)^T \mathbf{x}_u^H + \mathbf{N}_b, \tag{3.9}$$

where $\mathbf{N}_b$ is a noise matrix. Assuming that pilots transmitted by different UEs are mutually orthogonal, the decorrelated received pilot for the uth UE at the bth BS is

$$\begin{aligned} \widetilde{\mathbf{y}}_{b,u} &= \mathbf{Y}_b \mathbf{x}_u \\ &= \mathbf{h}_{b,u}^* + \mathbf{H}_{b,R}^T \mathbf{\Phi} \mathbf{h}_{R,u}^* + \mathbf{n}_{b,u}, \end{aligned} \tag{3.10}$$

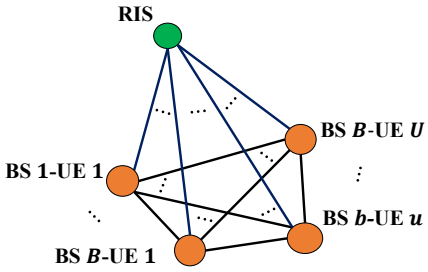


Figure 3.2: Graph representation of the considered RIS-aided multi-BS multi-UE system.

where $\mathbf{n}_{b,u} = \mathbf{N}_b \mathbf{x}_u$. All the channel information is contained in $\widetilde{\mathbf{y}}_{b,u}$ for $b \in \mathcal{B}, u \in \mathcal{U}$, which serves as the input of the GNN without requiring explicit CSI knowledge.

In the case where pilots are not perfectly orthogonal, inter-user correlation introduces additional interference in the decorrelated received pilots:

$$\begin{aligned}\widetilde{\mathbf{y}}_{b,u} &= \mathbf{h}_{b,u}^{*} + \mathbf{H}_{b,R}^{T} \boldsymbol{\Phi} \mathbf{h}_{R,u}^{*} \\ &+ \sum_{u' \in \mathcal{U}, u' \neq u} \left(\mathbf{h}_{b,u'}^{H} + \mathbf{h}_{R,u'}^{H} \boldsymbol{\Phi} \mathbf{H}_{b,R}\right)^{T} \mathbf{x}_{u'}^{H} \mathbf{x}_{u} + \mathbf{n}_{b,u},\end{aligned} \tag{3.11}$$

where $\mathbf{x}_{u'}^{H} \mathbf{x}_u$ captures the correlation between the uth UE and the other UEs. If a random sequence is used, where each element of $\mathbf{x}_u$ is either $-1/\sqrt{L}$ or $1/\sqrt{L}$ with equal probability, the correlation can be modeled as a Gaussian distribution with zero mean and variance $1/L$ for large L from the central limit theorem [63, 64]. If a deterministic pilot sequence is used, such as Zadoff–Chu (ZC) sequences, the correlation will be a deterministic value [65, 66].

3.3.2 GNN Design

3.3.2.1 GNN Structure

The structure of the proposed GNN model is shown in Fig. 3.3, which consists of three parts: the initial layer, Q updating layers (the first two parts are described in this subsection), and the output layer (described in Section 3.3.2.5). The graph can be used to facilitate understanding of the feature exchanges in each GNN layer, as will be elaborated below. Anatomical details of Fig. 3.3 are given in Fig. 3.4-3.7.

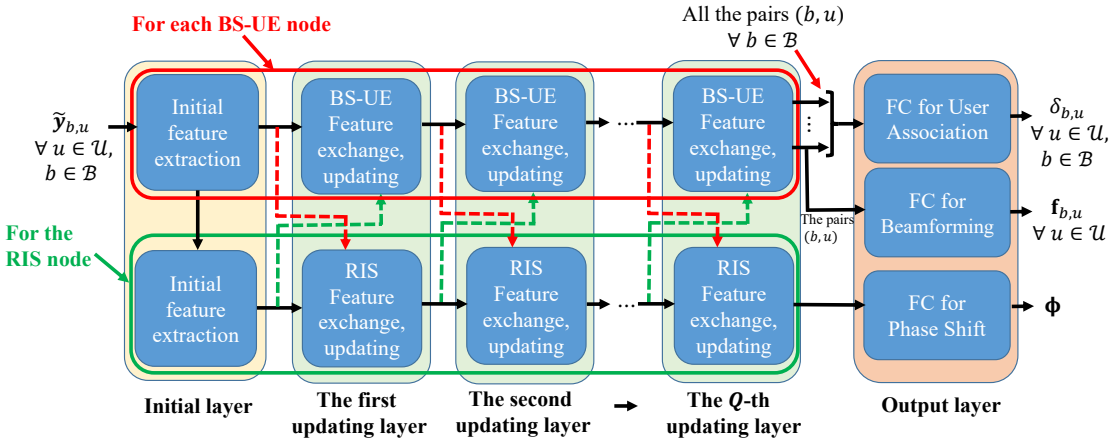


Figure 3.3: The multilayered structure of the proposed GNN-AAA.

Initial Layer: In the initial layer, as shown in Fig. 3.4, the decorrelated pilot signals $\widetilde{\mathbf{y}}_{b,u}$, $\forall u \in \mathcal{U}$, $b \in \mathcal{B}$ are fed into the feature extractor $\Psi_U^{(0)}(\cdot) : \mathbb{R}^{2N_T \times 1} \mapsto \mathbb{R}^{K \times 1}$ in each BS–UE node, where $\Psi_U^{(0)}(\cdot)$ is implemented by a single-hidden-layer DNN with $2N_T$ input neurons, $2K$ hidden neurons, and K output neurons (hereafter denoted as a DNN of size $2N_T \times 2K \times K$), and K is an adjustable parameter. For the (b,u)th BS–UE node, the output of the feature extractor, represented as

$$\mathbf{z}_{(b,u)}^{(0)} = \Psi_U^{(0)}\big(\mathbf{z}_{(b,u)}^{(0)}\big) \in \mathbb{R}^{K \times 1}, \tag{3.12}$$

is passed to the RIS node in the initial layer and all the RIS and BS–UE nodes in the next layer, where $\mathbf{z}_{(b,u)}^{(0)} = \big[\Re\{\widetilde{\mathbf{y}}_{b,u}^T\}, \Im\{\widetilde{\mathbf{y}}_{b,u}^T\}\big]^T \in \mathbb{R}^{2N_T \times 1}$. The RIS node aggregates the BS–UE node features $\mathbf{z}_{(b,u)}^{(0)}$, $\forall u \in \mathcal{U}$, $b \in \mathcal{B}$ and produces the RIS node feature

$$\mathbf{z}_R^{(0)} = \Psi_R^{(0)}\Big(f_0\big(\mathbf{z}_{(1,1)}^{(0)}, \ldots, \mathbf{z}_{(b,u)}^{(0)}, \ldots, \mathbf{z}_{(B,U)}^{(0)}\big)\Big) \in \mathbb{R}^{K \times 1} \tag{3.13}$$

using the feature extractor $\Psi_R^{(0)}(\cdot) : \mathbb{R}^{K \times 1} \mapsto \mathbb{R}^{K \times 1}$ implemented via a DNN of size $K \times 2K \times K$. To ensure the *permutation invariance* property of the GNN learning approach, the function $f_0(\cdot)$ is empirically chosen as an element-wise mean function, i.e., $f_0\big(\mathbf{z}_{(1,1)}^{(0)}, \ldots, \mathbf{z}_{(b,u)}^{(0)}, \ldots, \mathbf{z}_{(B,U)}^{(0)}\big) = \frac{1}{B \times U} \sum_{b \in \mathcal{B}} \sum_{u \in \mathcal{U}} \mathbf{z}_{(b,u)}^{(0)}$. Specifically, the phase shifts of the RIS generated by the GNN should be invariant to the ordering of the BS–UE pair of the GNN input.

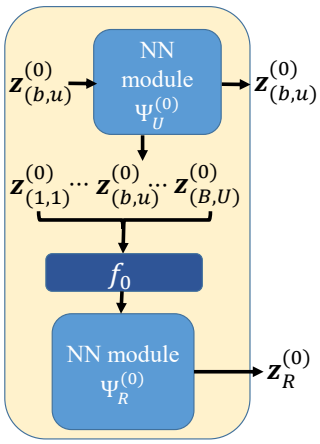


Figure 3.4: Details of the initial layer in the proposed GNN-AAA.

Updating Layers: In each updating layer τ ($\tau = 1, \ldots, Q$), all the BS–UE nodes and the RIS node exchange information (aggregation) and update their features (combination), which are then passed to the next updating layer.

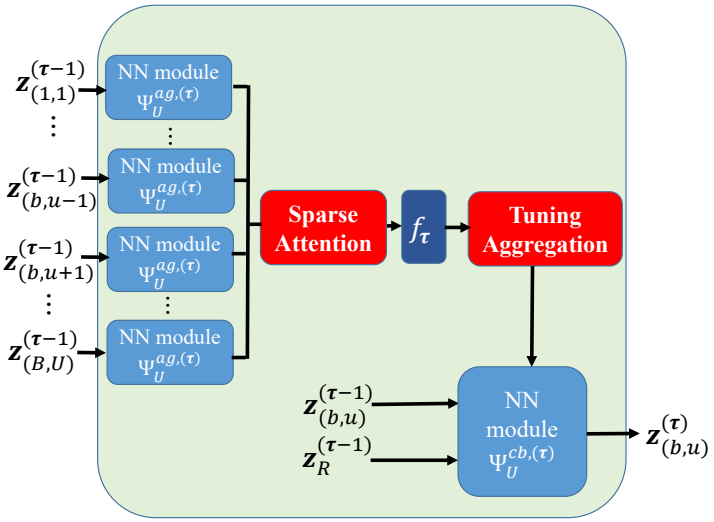
In the aggregation for the BS–UE nodes, each BS–UE node exchanges features with its neighboring BS–UE nodes. Specifically, in the τth layer, as shown in Fig. 3.5, the feature vectors of all neighboring BS–UE nodes from the previous $(\tau-1)$th layer are first aggregated by $\Psi_U^{\mathrm{ag},(\tau)}(\cdot) : \mathbb{R}^{K\times 1} \mapsto \mathbb{R}^{K\times 1}$, implemented by a DNN of size $K \times K \times K$, and subsequently transformed into

$$\mathbf{z}_{(b,u)}^{\mathrm{ag},(\tau)} = f_\tau \left(\left\{ \Psi_U^{\mathrm{ag},(\tau)} \left(\mathbf{z}_{(d,i)}^{(\tau-1)} \right) \right\}_{(d,i)\neq(b,u)} \right) \in \mathbb{R}^{K\times 1} \tag{3.14}$$

to preserve the *permutation equivariance* property of the GNN. In this work, $f_\tau(\cdot)$ is empirically chosen as a summation function.

In the combination for the BS–UE nodes, each BS–UE node combines its own features from the last updating layer, the aggregated features, and the features from the RIS node, to update the node feature in the current layer. That is, the node feature vector $\mathbf{z}_{(b,u)}^{(\tau)}$ of the (b,u)th BS–UE node at the τth layer is updated by

$$\mathbf{z}_{(b,u)}^{(\tau)} = \Psi_U^{\mathrm{cb},(\tau)}(\mathbf{z}_{(b,u)}^{(\tau-1)}, \mathbf{z}_{(b,u)}^{\mathrm{ag},(\tau)}, \mathbf{z}_R^{(\tau-1)}) \in \mathbb{R}^{K\times 1}, \tag{3.15}$$



The τ-th updating layer (BS-UE)

Figure 3.5: Details of the τth updating layer for the BS–UE nodes in the proposed GNN-AAA.

where the combiner $\Psi_U^{\mathrm{cb},(\tau)}(\cdot) : \mathbb{R}^{3K\times 1} \mapsto \mathbb{R}^{K\times 1}$ is implemented using a DNN of size $3K \times K \times K$.

In the aggregation for the RIS nodes, all the BS–UE nodes exchange their features with the RIS node, as shown in Fig. 3.6. The aggregated BS–UE node features are given by

$$\mathbf{z}_R^{\mathrm{ag},(\tau)} = f_{\tau,R}\left(\{\Psi_R^{\mathrm{ag},(\tau)}(\mathbf{z}_{(b,u)}^{(\tau-1)})\}_{b\in\mathcal{B},u\in\mathcal{U}}\right) \in \mathbb{R}^{K\times 1}, \tag{3.16}$$

where the aggregator $\Psi_R^{\mathrm{ag},(\tau)}(\cdot) : \mathbb{R}^{K\times 1} \mapsto \mathbb{R}^{K\times 1}$ is implemented using a DNN of size $K \times K \times K$, and, similar to $f_0(\cdot)$ in (3.13), $f_{\tau,R}(\cdot)$ is empirically chosen as an element-wise mean function to ensure permutation invariance.

In the combination for the RIS node, by combining the aggregated BS–UE node features and the RIS node features from the last layer, the feature vector of the RIS node at the τth layer is updated by

$$\mathbf{z}_R^{(\tau)} = \Psi_R^{\mathrm{cb},(\tau)}\left(\mathbf{z}_R^{(\tau-1)}, \mathbf{z}_R^{\mathrm{ag},(\tau)}\right) \in \mathbb{R}^{K\times 1}, \tag{3.17}$$

where the combiner $\Psi_R^{\mathrm{cb},(\tau)}(\cdot) : \mathbb{R}^{2K\times 1} \mapsto \mathbb{R}^{K\times 1}$ is implemented using a DNN of size $2K \times K \times K$.

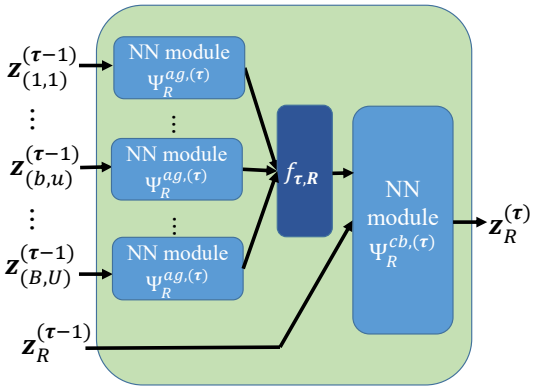


Figure 3.6: Details of the τth updating layer for the RIS node in the proposed GNN-AAA.

To address the oversmoothing issue caused by repeated aggregation in multi-layer GNN architectures, which can degrade the learning performance [17], we propose a mechanism for aggregation and combination, as described next.

3.3.2.2 Oversmoothing Issue and Solution

Oversmoothing refers to the issue where graph node features become increasingly similar after traversing a deep updating structure, which could potentially degrade a GNN's learning capability [17]. We propose a mechanism called adaptive attention-based aggregation (AAA) to mitigate the oversmoothing problem in GNNs, which consists of two complementary components: 1) sparse attention for aggregation, and 2) tuning aggregation for combination. Sparse attention focuses on the most important neighboring BS–UE nodes by restricting the set of neighbor BS–UE nodes with which each node exchanges information. Tuning aggregation controls whether aggregated neighbor features are incorporated into a node's updated features, especially when sparse attention alone is insufficient to suppress harmful neighbors. These harmful neighbors are those whose aggregated features make a node more similar to others, thereby exacerbating the oversmoothing effect. Adaptive and flexible adjustments are achieved through these two components to prevent

excessive information loss, thereby preserving the GNN's learning capability.

3.3.2.3 Sparse Attention for Aggregation

Sparse attention for aggregation comprises the following steps: a) computing attention weights, b) sparsifying the attention matrix, and c) adaptively adjusting the sparsity level.

3.3.2.3.1 Computing Attention Weights In sparse attention, the features from the neighbor nodes are first assigned attention weights. The aggregation for the BS–UE node in the τth layer with attention is given by

$$\mathbf{z}_{(b,u)}^{\mathrm{ag},(\tau)}=f_{\tau}\left(\left\{\widehat{a}_{(b,u),(d,i)}^{(\tau)}\Psi_{U}^{\mathrm{ag},(\tau)}\left(\mathbf{z}_{(d,i)}^{(\tau-1)}\right)\right\}_{(d,i)\neq(b,u)}\right)\in\mathbb{R}^{K\times 1}, \tag{3.18}$$

where $\widehat{a}_{(b,u),(d,i)}^{(\tau)}$ is the attention weight of the node pair between the (b,u)th BS–UE node and the (d,i)th BS–UE node in the τth layer. The attention weight is calculated using the Euclidean distance between the features of the (b,u)th BS–UE node and the (d,i)th BS–UE node, where a larger distance corresponds to a higher attention weight, i.e.,

$$\widehat{a}_{(b,u),(d,i)}^{(\tau)}=\eta\left(\left\|\mathbf{z}_{(b,u)}^{(\tau)}-\mathbf{z}_{(d,i)}^{(\tau)}\right\|\right), \tag{3.19}$$

where $\eta(\cdot)$ is the softmax function for normalization. The use of attention weights ensures that neighbor nodes with less similar features (and thus more informative) are assigned higher aggregation weights.

3.3.2.3.2 Sparsifying the Attention Matrix A sparsity mechanism is applied to the attention weights to mitigate the oversmoothing problem. Inspired by [67], we extract only the most significant neighboring node features for information exchange, discarding the less relevant ones. First, define $\mathbf{U}^{(\tau)}\in\mathbb{R}^{BU\times BU}$ with elements $\mathbf{U}_{(b,u),(d,i)}^{(\tau)}=\widehat{a}_{(b,u),(d,i)}^{(\tau)}$, $\forall b,d\in\mathcal{B},u,i\in\mathcal{U}$. Then, let $\mathbf{A}^{(\tau)}\in\mathbb{R}^{BU\times BU}$ be the sparse attention weight matrix at the τth updating layer, and let $\mathbf{A}_{(b,u)}^{(\tau)}=\left[a_{(b,u),(1,1)}^{(\tau)},\,a_{(b,u),(1,2)}^{(\tau)},\ldots,a_{(b,u),(B,U)}^{(\tau)}\right]\in\mathbb{R}^{1\times BU},\ \forall b\in\mathcal{B},u\in\mathcal{U}$ be the (b,u)th row of $\mathbf{A}^{(\tau)}$, which represents the attention weights for the (b,u)th BS–UE node at the τth updating layer. We find a sparse attention weight

matrix by solving

$$\widehat{\mathbf{A}}^{(\tau)} = \underset{\mathbf{A}^{(\tau)}}{\arg\min} \left\| \mathbf{A}^{(\tau)} - \mathbf{U}^{(\tau)} \right\|_F, \tag{3.20a}$$

$$\text{s.t. } \left\| \mathbf{A}^{(\tau)}_{(b,u)} \right\|_0 \leq \beta^{(\tau)}_{(b,u)} BU, \ \forall b \in \mathcal{B}, u \in \mathcal{U}. \tag{3.20b}$$

Here, (3.20b) represents the sparsity constraint, restricting the number of nonzero elements in $\mathbf{A}^{(\tau)}_{(b,u)}$ to $\beta^{(\tau)}_{(b,u)} BU$, where $\beta^{(\tau)}_{(b,u)} \in (0, 1]$ is a parameter that controls the degree of sparsity. By finding the optimal $\widehat{\mathbf{A}}^{(\tau)}$ in (3.20), we retain neighbor node features that differ from a node's dominant features, enabling effective information exchange with neighbors that contribute more unique information. This mitigates oversmoothing, where nodes become overly similar after multiple rounds of information exchange. To reduce complexity, we retain only the largest $\beta^{(\tau)}_{(b,u)} BU$ elements in $\mathbf{A}^{(\tau)}_{(b,u)}$ and set the remaining elements to zero.

3.3.2.3.3 Adaptively Adjusting the Sparsity Level An adaptive adjustment of $\beta^{(\tau)}_{(b,u)}$ is further applied to sparse attention. To guide this adjustment, the *squared norm of node differences*, which is a measure of oversmoothing, is computed based on the squared differences between the feature vectors of each node pair. Specifically, for the (b, u)th node in the τth updating layer, the squared norm of node differences is given by

$$p^{(\tau)}_{(b,u)} = \sum_{(d,i) \in \mathcal{N}_{(b,u)}} \left\| \mathbf{z}^{(\tau)}_{(b,u)} - \mathbf{z}^{(\tau)}_{(d,i)} \right\|^2, \tag{3.21}$$

where $\mathcal{N}_{(b,u)}$ is the set of neighbor nodes for the (b, u)th node. A smaller squared norm of node differences suggests a more severe oversmoothing issue for the node. Then, we compare each node's squared norm of node differences between the τth and $(\tau - 1)$th updating layers and identify the nodes where the squared norm of node differences in the τth layer is lower than that in the $(\tau - 1)$th layer. These nodes, which experience worsening oversmoothing across layers, are collected into the set $\mathcal{V}^{(\tau)}$. For each node in $\mathcal{V}^{(\tau)}$, we define $\beta^{(\tau)}_{(b,u)}$ based on the ratio of the decrease in each node's squared norm of node differences relative to all nodes in $\mathcal{V}^{(\tau)}$, using the softmax function, i.e.,

$$\beta^{(\tau)}_{(b,u)} = \frac{e^{\left| p^{(\tau-1)}_{(b,u)} - p^{(\tau)}_{(b,u)} \right|}}{\sum_{(d,i) \in \mathcal{V}^{(\tau)}} e^{\left| p^{(\tau-1)}_{(d,i)} - p^{(\tau)}_{(d,i)} \right|}}, \quad \forall (b, u) \in \mathcal{V}^{(\tau)}. \tag{3.22}$$

For nodes not in $\mathcal{V}^{(\tau)}$, we set $\beta_{(b,u)}^{(\tau)} = 1$, $\forall (b,u) \notin \mathcal{V}^{(\tau)}$, as they are deemed not experiencing significant oversmoothing in the τth updating layer.

3.3.2.4 Tuning Aggregation for Combination

In tuning aggregation, we adjust the impact of the aggregated feature on the updated feature. For each node, its oversmoothing status is evaluated. If a node exhibits worse oversmoothing in the current layer compared to the previous layer, the extent of utilizing aggregated information for that node is reduced. With the adaptive tuning aggregation, the node feature vector $\mathbf{z}_{(b,u)}^{(\tau)}$ of the (b,u)th BS–UE node at the τth layer is updated by

$$\mathbf{z}_{(b,u)}^{(\tau)} = \Psi_U^{\text{cb},(\tau)}\left(\mathbf{z}_{(b,u)}^{(\tau-1)}, \beta_{(b,u)}^{(\tau)}\mathbf{z}_{(b,u)}^{\text{ag},(\tau)}, \mathbf{z}_R^{(\tau-1)}\right) \in \mathbb{R}^{K\times 1}, \tag{3.23}$$

where the aggregated feature $\mathbf{z}_{(b,u)}^{\text{ag},(\tau)}$ is adaptively weighted through the parameter $\beta_{(b,u)}^{(\tau)}$ specified in (3.22).

3.3.2.5 Output of the GNN

The output layer consists of three blocks which produce the learned results of user association, transmit beamforming, and RIS phase shifts, as shown in Fig. 3.7.

UA Block: The UA block contains U modules that generate the UA results of the U UEs, where each module is implemented by a single fully-connected (FC) layer with softmax as the output function. Specifically, we collect the features associated with the uth UE after the Qth updating layer and transform them into the feature

$$\mathbf{z}_{(b,u)}^{\text{out}} = \left[\mathbf{z}_{(b,u)}^{(Q)}, \mathbf{z}_u^{(Q)}\right]^T \in \mathbb{R}^{2K\times 1}, \tag{3.24}$$

where $\mathbf{z}_u^{(Q)} = \sum_{b=1}^{B} \mathbf{z}_{(b,u)}^{(Q)}$ can be seen as the synthesized feature associated with the uth UE. The uth UA module first maps the features $\mathbf{z}_{(b,u)}^{\text{out}}$ to a scalar $\omega_{(b,u)}$ using an FC linear function $\Psi_{\text{UA}}(\cdot) : \mathbb{R}^{2K\times 1} \mapsto \mathbb{R}$, i.e., $\omega_{(b,u)} = \Psi_{\text{UA}}(\mathbf{z}_{(b,u)}^{\text{out}})$. Then, the softmax function is applied to $\omega_{(b,u)}$ to produce B softmax values

$$\widetilde{\delta}_{b,u} = \frac{e^{\omega_{(b,u)}}}{\sum_{b\in\mathcal{B}} e^{\omega_{(b,u)}}}, \quad \text{for } b = 1,\ldots,B. \tag{3.25}$$

Since the softmax output can be interpreted as the posterior probability of association between the bth BS and the uth UE under the current channel condition, the final UA

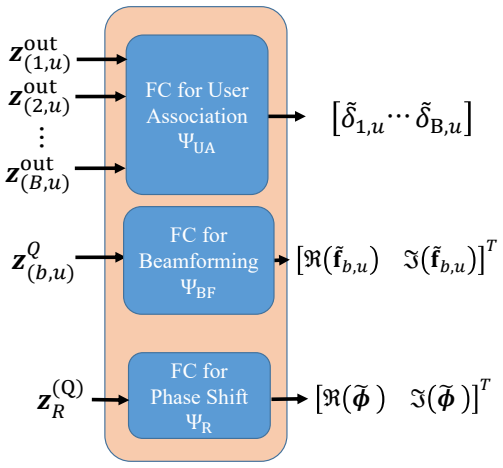


Figure 3.7: Details of the output layer in the proposed GNN-AAA.

result that satisfies constraint (3.8d) for the uth UE can be chosen as the BS that yields the maximum $\delta_{b,u}$, i.e.,

$$\begin{cases} \delta_{b_u^*,u} = 1, & \text{for } b_u^* = \arg\max_{b\in\mathcal{B}} \widetilde{\delta}_{b,u} \\ \delta_{b,u} = 0, & \text{for } b \neq b_u^*, b \in \mathcal{B}. \end{cases} \tag{3.26}$$

Beamformer Block: The beamformer block contains U modules that generate the transmit beamformers for all the U UEs, with the uth module of the bth BS being implemented by a single FC layer $\Psi_{\text{BF}}(\cdot) : \mathbb{R}^{K\times 1} \mapsto \mathbb{R}^{2N_T\times 1}$ to produce

$$\mathbf{z}^{\text{BF}}_{(b,u)} = \Psi_{\text{BF}}\big(\mathbf{z}^{(Q)}_{(b,u)}\big) \in \mathbb{R}^{2N_T\times 1}, \tag{3.27}$$

which corresponds to the beamformer for the uth UE transmitted by the bth BS determined by the UA block. The vector of the first N_T elements $\widetilde{\mathbf{f}}_{b,u,1} = [\mathbf{z}^{\text{BF}}_{(b,u)}]_{1:N_T}$ represents the real part of the beamformer, while the vector of the last N_T elements $\widetilde{\mathbf{f}}_{b,u,2} = [\mathbf{z}^{\text{BF}}_{(b,u)}]_{(N_T+1):2N_T}$ corresponds to the imaginary part of the beamformer. The output beamformer of the bth BS for the uth UE is then given by

$$\widetilde{\mathbf{f}}_{b,u} = \widetilde{\mathbf{f}}_{b,u,1} + j\widetilde{\mathbf{f}}_{b,u,2}. \tag{3.28}$$

To satisfy the constraint (3.8f), a soft beamforming matrix $\widehat{\mathbf{g}}_b$ for the bth BS is constructed during training by

$$\widetilde{\mathbf{f}}_b \triangleq \left[\widetilde{\delta}_{b,1}\widetilde{\mathbf{f}}_{b,1},\ \widetilde{\delta}_{b,2}\widetilde{\mathbf{f}}_{b,2},\ \ldots,\ \widetilde{\delta}_{b,U}\widetilde{\mathbf{f}}_{b,U}\right], \tag{3.29}$$

which can be regarded as the contribution of beamformers from the bth BS for each UE under the soft user association $\widetilde{\delta}_{b,u},\ \forall u \in \mathcal{U}$ in the training stage. The virtual composite beamformer complying with constraint (3.8f) for the uth UE is

$$\mathbf{f}_{b,u} = \frac{\widetilde{\mathbf{f}}_{b,u}}{\sum_{b\in\mathcal{B}} \widetilde{\delta}_{b,u}\|\widetilde{\mathbf{f}}_b\|}\sqrt{P_{\max}}. \tag{3.30}$$

This suggests that, during testing, the uth UE's beamformer transmitted by the b_u^*th BS satisfying the power constraint (3.8f) is given by

$$\mathbf{f}_{b_u^*,u} = \frac{\widetilde{\mathbf{f}}_{b_u^*,u}}{\sum_{i\in\mathcal{C}_{b_u^*}^{\mathrm{UE}}} \delta_{b_u^*,i}\|\widetilde{\mathbf{f}}_{b_u^*,i}\|}\sqrt{P_{\max}}, \tag{3.31}$$

where $\mathcal{C}_{b_u^*}^{\mathrm{UE}} = \{i \mid \delta_{b_u^*,i} = 1, i \in \mathcal{U}\}$ is the set of UEs being served by the BS b_u^* that is associated with the uth UE during testing, with b_u^* defined in (3.26).

RIS Phase Shift Block: Finally, the RIS phase shift block collects the RIS feature vector from the Qth updating layer and generates a $2M \times 1$ vector $\mathbf{z}^{\mathrm{RIS}} = \Psi_{\mathrm{RIS}}(\mathbf{z}_R^{(Q)}) \in \mathbb{R}^{2M\times 1}$, where $\Psi_{\mathrm{RIS}}(\cdot) : \mathbb{R}^{K\times 1} \mapsto \mathbb{R}^{2M\times 1}$ is implemented by an FC layer. The first M elements and the last M elements of $\mathbf{z}^{\mathrm{RIS}}$ are the real and imaginary parts of the RIS phase shifts, respectively, and normalization is applied to ensure unit length. That is,

$$\widetilde{\phi}_m \triangleq \left[\mathbf{z}^{\mathrm{RIS}}\right]_m + j\cdot\left[\mathbf{z}^{\mathrm{RIS}}\right]_{M+m} \text{ and } \phi_m = \frac{\widetilde{\phi}_m}{\left|\widetilde{\phi}_m\right|},$$
$$m = 1, \ldots, M. \tag{3.32}$$

3.3.2.6 Training of GNN

The proposed GNN model employs *unsupervised* offline training, utilizing training samples generated from different pilot signal realizations. The training loss function combines the sum-rate maximization objective with a penalty term that enforces the load balancing constraint in (3.8e). It is defined as

$$\mathcal{L} = -\sum_{u\in\mathcal{U}} R_u + \lambda\cdot L_p, \tag{3.33}$$

where

$$L_p \triangleq \sum_{b\in\mathcal{B}} \max\left(0, \sum_{u\in\mathcal{C}_b^{\mathrm{UE}}} \delta_{b,u} - Z_{\max,b}\right) \tag{3.34}$$

is the penalty for violating the load balancing constraint, and λ is a weight that balances the influence of the sum rate and the penalty for load-balancing violations.

The proposed method is summarized in Algorithm 1.

Remark 1: It is worthwhile to note that the proposed GNN model exhibits generalizability to varying numbers of UEs *and* BSs, i.e., the proposed model can be trained with fewer UEs and BSs while allowing testing with larger numbers of UEs and BSs. This is achieved because all learning modules within the feature extraction and update processes are shared across BS–UE links, and the architecture and the information exchange mechanism are designed to satisfy permutation invariance and permutation equivariance properties. For example, as shown in the red solid-outlined region in Fig. 3.3, all BS–UE nodes share the same learning module chain. Besides, the NN modules in Fig. 3.4 and Fig. 3.5 can be simply replicated multiple times when there are more UEs and/or BSs. Furthermore, the parameter settings of the learning modules in the output layer are independent of the numbers of UEs and BSs. This generalizability across varying numbers of UEs and BSs facilitates low-complexity training with fewer UEs and BSs, enables the direct application of shared, pre-trained GNN learning modules to newly added BS–UE links without retraining, and is a highly desirable property in practical dynamic network environments.

3.3.3 Complexity Analysis

In this section, we analyze the computational complexity of the proposed GNN-AAA method. The total complexity results from the sum of computations in the initial layer, updating layers, and output layer. The analysis can be expressed in terms of the number of UEs, U, the number of BSs, B, the length of a feature vector in each updating layer, K, the number of transmit antennas at each BS, N_T, the number of updating layers, Q, and the number of RIS elements, M.

Algorithm 1 The Proposed GNN-AAA Method

Require: The decorrelated received pilot, $\widetilde{\mathbf{y}}_{b,u},\ \forall b \in \mathcal{B}, u \in \mathcal{U}$.

Ensure: The indicator of user association, $\delta_{b,u}$, the transmit beamformer, $\mathbf{f}_{b,u}$, and the matrix of RIS phase shifts, $\mathbf{\Phi}$, for $b \in \mathcal{B}, u \in \mathcal{U}$

1: Extract the initial feature, $\mathbf{z}_{(b,u)}^{(0)}, \mathbf{z}_{R}^{(0)}$ according to (3.12) and (3.13), respectively
2: **for** each updating layer $\tau \in \{1, 2, \ldots, Q\}$ **do**
3: Each BS–UE node generates the aggregated feaures $\mathbf{z}_{(b,u)}^{\text{ag},(\tau)}$ according to (3.18), with the weights of sparse attention $\widehat{a}_{(b,u),(d,i)}^{(\tau)}$ given by (3.20)
4: Each BS–UE node updates its feature $\mathbf{z}_{(b,u)}^{(\tau)}$ according to (3.23), with the weights of sparse attention given by (3.22)
5: The RIS node generates the aggregated feature $\mathbf{z}_{R}^{\text{ag},(\tau)}$ according to (3.16)
6: The RIS node updates its feature $\mathbf{z}_{(b,u)}^{(\tau)}$ according to (3.17)
7: **end for**
8: Generate the final feature for user association, $\mathbf{z}_{(b,u)}^{\text{out}}$, according to (3.24)
9: Generate the indicator of user association, $\delta_{b,u}$, according to (3.26)
10: Generate the transmit beamformer, $\mathbf{f}_{b,u}$, according to (3.30)
11: Generate the matrix of RIS phase shifts, $\mathbf{\Phi}$, according to (3.32)

Initial Layer: The computations involve input feature extraction for each BS–UE node in (3.12) and feature aggregation and extraction for the RIS node in (3.13). The computations of input feature extraction by $\Psi_U^{(0)}$ of size $2N_T \times 2K \times K$ in (3.12) include multiplications with neuron weights and ReLU activation, which require $\mathcal{O}(2K^2 + 4N_T K)$ and $\mathcal{O}(3K)$ operations for each BS–UE node, respectively. The complexity for all BS–UE nodes is therefore $\mathcal{O}\big(BU(2K^2 + (4N_T + 3)K)\big)$. The computations in (3.13) include computing the element-wise mean of the BS–UE node input features, which requires $\mathcal{O}(BUK)$ operations, followed by feature extraction using $\Psi_R^{(0)}$ of size $K \times 2K \times K$ with ReLU activation, which requires $\mathcal{O}(4K^2 + 3K)$ operations. Summing these up, the overall complexity of the initial layer is approximated as $\mathcal{O}\Big(BU\big(2K^2 + (4N_T + 4)K\big)\Big)$.

Updating Layers: The computations for the BS–UE nodes and the RIS node are analyzed separately. The BS–UE node complexity arises from executing the AAA mechanism, as detailed below.

3.3.3.1 Sparse Attention for Aggregation

The complexity encompasses three main steps: a) computing attention weights, b) sparsifying the attention matrix, and c) adaptively adjusting the sparsity level. For analytical convenience, we consider the worst-case scenario in terms of complexity, where $\beta_{(b,u)}^{(\tau)} = 1, \forall b \in \mathcal{B},\ u \in \mathcal{U}$ and $\tau \in \{1, \ldots, Q\}$.

3.3.3.1.1 Computing Attention Weights The complexity involves aggregating features from neighboring nodes and multiplying the aggregated features by the attention weights as in (3.18), and computing the attention weights themselves as in (3.19). The computations in (3.18) include, for each BS–UE node, extracting features from $BU-1$ neighboring nodes using $\Psi_U^{\text{ag},(\tau)}$ of size $K \times K \times K$ with ReLU activation, which requires $\mathcal{O}(2K^2+2K)$ operations, and multiplying the aggregated features by the attention weights followed by aggregation, which requires $\mathcal{O}\big((BU-1)K\big)$ and $\mathcal{O}\big((BU-2)K\big)$ operations, respectively. In (3.19), computing pairwise Euclidean distances between all BU nodes' feature vectors involves $\mathcal{O}((BU-1)K)$ operations, and the softmax operation adds $\mathcal{O}(BU)$ operations. Overall, the total complexity for all BS–UE nodes is approximated as $\mathcal{O}\big(B^2U^2(3K+1) + BU(2K^2+2K)\big)$.

3.3.3.1.2 Sparsifying the Attention Matrix The complexity of solving (3.20) is determined by projecting each row vector of $\mathbf{A}^{(\tau)}$ onto an ℓ_0 unit ball [67], which can be implemented by partial sorting and simplex projection, requiring $\mathcal{O}\big(BU\log(BU-1)\big)$ operations. Therefore, the overall complexity for all BS–UE nodes is approximately $\mathcal{O}\big(B^2U^2\log(BU)\big)$.

3.3.3.1.3 Adaptively Adjusting the Sparsity Level The complexity of this step involves computing pairwise feature differences between each BS–UE node and its $BU-1$ neighboring nodes as in (3.21), which requires $\mathcal{O}\big(B^2U^2K\big)$ operations, and computing the power differences with a row-wise softmax over vectors of length BU as in (3.22), which require $\mathcal{O}(B^2U^2)$ operations, for all BS–UE nodes. Therefore, the total complexity for all BS–UE nodes is approximated as $\mathcal{O}\big(B^2U^2(K+1)\big)$.

3.3.3.2 Tuning Aggregation for Combination

The complexity of this step includes that for first performing element-wise multiplication between each BS–UE node's aggregated feature vector and the adaptive tuning adjustment as in (3.23), which requires $\mathcal{O}(K)$ operations, followed by feature combination using $\Psi_U^{\text{cb},(\tau)}$ of size $3K \times K \times K$ with ReLU activation, which requires $\mathcal{O}(4K^2 + 2K)$ operations. Thus, the total complexity for all BS–UE nodes is approximated as $\mathcal{O}\big(BU(4K^2 + 3K)\big)$.

Summing up these complexity orders, the total complexity for the BS–UE nodes across the Q updating layers is $\mathcal{O}\Big(QB^2U^2\big(\log(BU) + 4K + 2\big) + QBU(6K^2 + 5K)\Big)$.

The RIS node complexity involves aggregating features from all BS–UE nodes as in (3.16) and combining the aggregated feature with the RIS node's feature from the previous layer as in (3.17). In (3.16), feature aggregation using $\Psi_R^{\text{ag},(\tau)}$ of size $K \times K \times K$ with ReLU activation requires $\mathcal{O}\big(BU(2K^2 + 2K)\big)$ operations, while the element-wise mean operation requires $\mathcal{O}(BUK)$ operations. In (3.17), the combination using $\Psi_R^{\text{cb},(\tau)}$ of size $2K \times K \times K$ with ReLU activation requires $\mathcal{O}(3K^2 + 2K)$ operations. Summing these up, the total complexity for the RIS node across the Q updating layers is $\mathcal{O}\big(QBU(2K^2 + 3K) + Q(3K^2 + 2K)\big)$.

Output Layer: The computations include UA, beamforming, and RIS phase shift calculations for the output. The UA complexity comprises computing the FC linear function $\Psi_{\text{UA}}(\cdot)$, which requires $\mathcal{O}(2BK)$ operations, and the softmax in (3.25), which requires $\mathcal{O}(B)$ operations, for one UE. This sums up to $\mathcal{O}\big(BU(2K + 1)\big)$ for all UEs. The beamforming complexity comprises computing $\Psi_{\text{BF}}(\cdot)$ in (3.27), which requires $\mathcal{O}(2N_TK)$ operations, followed by normalization to satisfy the power constraint as in (3.30), which requires $\mathcal{O}(N_T)$ operations, for each BS–UE node. This sums up to $\mathcal{O}\big(BUN_T(2K + 1)\big)$ for all BS–UE nodes. The RIS phase shift complexity involves computing $\Psi_{\text{RIS}}(\cdot)$, which requires $\mathcal{O}(2MK)$ operations, followed by normalization to enforce the unit-modulus constraint as in (3.32), which requires $\mathcal{O}(M)$ operations. Adding up these numbers, the total complexity for the output layer is $\mathcal{O}\Big(BU\big((2N_T + 2)K + (N_T + 1)\big) + M(2K + 1)\Big)$.

In summary, the overall computational complexity of the proposed GNN-AAA method

is given by

$$\begin{aligned}
&\underbrace{\mathcal{O}\Big(BU\big(2K^2+4N_TK+4K\big)\Big)}_{\text{initial layer}}\\
&+\underbrace{\mathcal{O}\Big(QB^2U^2\big(\log(BU)+4K+2\big)+QBU\big(8K^2+8K\big)\Big)}_{\text{updating layers}}\\
&+\underbrace{\mathcal{O}\Big(BU\big((2N_T+2)K+(N_T+1)\big)+M(2K+1)\Big)}_{\text{output layer}}. \qquad (3.35)
\end{aligned}$$

It is worthwhile to examine the complexity of the proposed AAA mechanism relative to the overall model, excluding the computations of aggregators and combiners since they are not part of the AAA mechanism, as summarized below:

$$\underbrace{\mathcal{O}\Big(QB^2U^2\big(\log(BU)+3K\big)}_{\text{sparse attention}}+\underbrace{\mathcal{O}\big(QBUK\big)}_{\text{tuning aggregation}}. \qquad (3.36)$$

Using the simulation parameters $N_T = 8$, $K = 512$, and $Q = 8$, the proportion of AAA's computational complexity order relative to the proposed GNN-AAA model is calculated as 5.82% for $B = 3, U = 6, M = 100$, 6.85% for $B = 3, U = 6, M = 200$, 10.76% for $B = 5, U = 8, M = 100$, and 11.69% for $B = 5, U = 8, M = 200$. These results indicate that the AAA mechanism adds only a modest increase to the overall computational complexity order, which is further corroborated by runtime measurements presented in the simulation section.

3.4 Simulations

3.4.1 Simulation Setups

We simulate a RIS-aided multi-UE multi-BS system where each BS has $N_T = 8$ antennas and operates in the $f_c = 28$ GHz mmWave band. For training, the default parameters are set as $B = 3$ BSs, $U = 6$ UEs, and $M = 100$ passive RIS elements. These parameters are adjusted for testing as specified in the subsequent simulation results. The BSs are randomly distributed along a semicircular arc centered at $(0, 0, 0)$ (in meters) in Cartesian

coordinate with a radius of 100 m, and the RIS is at $(-15, -20, 0)$, as shown in Fig. 3.8. Unless otherwise noted, UEs are uniformly distributed in a circular region with a radius of 20 m, centered at $(0, 0, -20)$. For the mmWave channel, there are $N_{cl} = 5$ clusters in each channel, with $L_\ell = 10$ rays per cluster. The azimuth/elevation AoAs and AoDs, $\psi_{\ell,i}^{b,t}, \theta_{\ell,i}^{b,t}, \psi_{\ell,i}^{R,t}, \theta_{\ell,i}^{R,t}, \psi_{\ell,i}^{R,r}, \theta_{\ell,i}^{R,r}$, $\forall b \in \mathcal{B}$, follow the uniform distribution over $[0, 2\pi]$. The path loss of the channel associated with the RIS is modeled as $30 + 22\log(d_{\star,R})$, where $d_{\star,R}$ is the distance between the communication node $\star$ (which may be a BS or a UE) and the RIS. The path loss between the bth BS and the uth UE with distance $d_{b,u}$ is modeled as $32.6 + 36.7\log(d_{b,u})$. The pilot length is $L = 31$. The uplink pilot transmission power and the noise power are set as 15 dBm and -85 dBm, respectively. The default BS maximum transmit power is set as $P_{\max,b} = 25$ dBm for all b. For the GNN, the sizes of the neural network (NN) modules in each layer are set as follows: $2N_T \times 1024 \times 512$ for $\Psi_U^{(0)}$, $512 \times 1024 \times 512$ for $\Psi_R^{(0)}$, $512 \times 512 \times 512$ for $\Psi_U^{\text{ag},(\tau)}$, $\Psi_U^{\text{cb},(\tau)}$, $\Psi_R^{\text{ag},(\tau)}$, and $\Psi_R^{\text{cb},(\tau)}$ for $\tau = 1, 2, \ldots, Q$, where $Q = 8$ is the number of updating layers in the GNN. The number of training samples in each epoch is 10^5, with a total of 200 epochs, and the batch size is 1024. An initial learning rate is set as 10^{-3} which decreases after 3 epochs by a factor of 0.98. The number of testing samples is 10^5. The weight λ is set to 2 and $Z_{\max,b}$ is set to U/B for all b. All simulation parameters are summarized in Table 3.2.

We denote the comparison schemes using the format: 'method [remark],' where the 'remark' is optional. The schemes in comparison are described as follows:

- *GNN-AAA [proposed]:* The proposed method.
- *GNN-AAA [no LB]:* Same as the proposed method but without load-balancing (LB) considerations, i.e., (3.34) is not included in the loss function.
- *GNN-AAA [for BF] + RSP [for UA]:* The proposed method is applied for BS and RIS beamforming (BF) only, while UA is determined by selecting the BS that provides the highest received signal power (RSP) at the UE.
- *GNN-AAA [no RIS]:* The proposed method is applied to the scenario without RIS.
- *GNN*: The GNN model without incorporating the AAA mechanism.

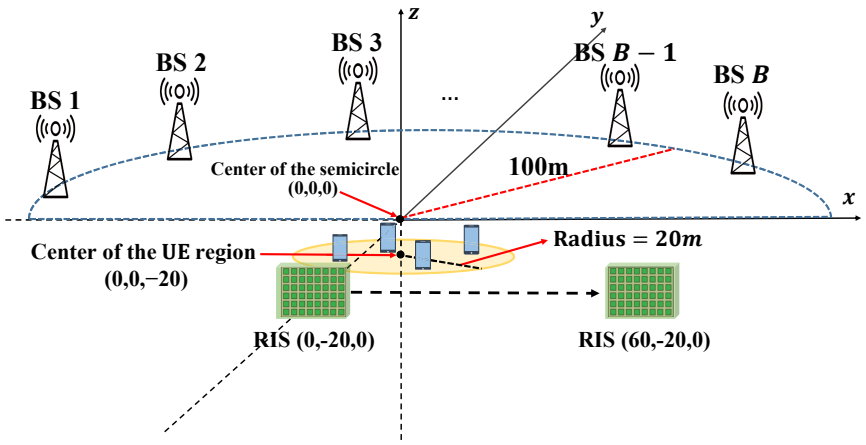


Figure 3.8: System topology for simulation.

- *HGAT:* The heterogeneous graph attention network (HGAT) based on the GAT framework in [68] is applied to the considered problem using the same loss function as the proposed GNN, with its attention mechanism and attention weight design adapted accordingly.
- *DNN:* A conventional DNN with fully connected layers, with the same model size, input, and loss function as the proposed GNN.
- *Model-based:* The model-based method from [27], which jointly optimizes BS and RIS beamforming as well as UA.

3.4.2 Convergence

Fig. 3.9 plots the training loss given in (3.33) against the training epochs to analyze the convergence behavior of *GNN-AAA [proposed]*. The results indicate that the inclusion of the penalty term (3.34) for load balancing (*GNN-AAA [proposed]* vs. *GNN-AAA [no LB]*) and the incorporation of the AAA mechanism (*GNN-AAA [proposed]* vs. *GNN*) have minimal impact on the convergence rate. All three schemes demonstrate consistent convergence at approximately 120 training epochs. However, since the loss function is inversely related to the sum rate, they converge to different final loss values, which roughly correspond to their respective sum-rate performance, as discussed next.

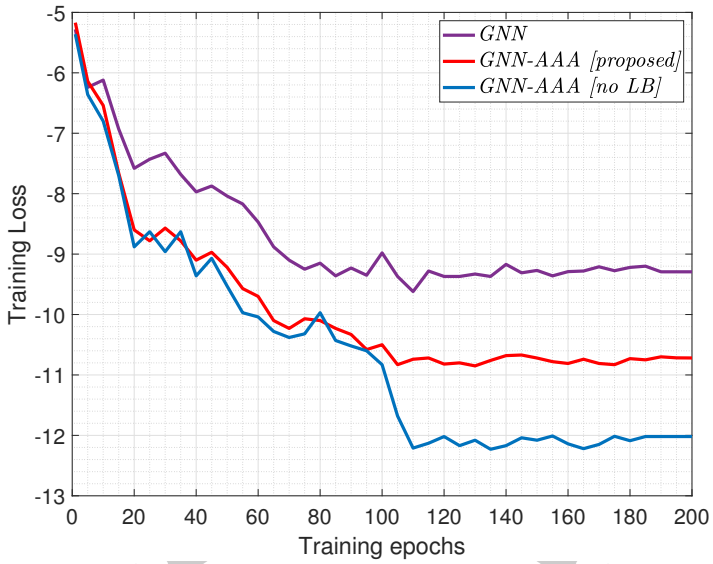


Figure 3.9: Training loss vs. training epochs, where $B = 3$, $U = 6$, $P_{\max,b} = 25$ dBm $\forall b$, and $M = 100$.

3.4.3 Sum-rate Performance

Effects of BS Transmit Power: Fig. 3.10 presents the testing sum-rate performance with respect to BS maximum transmit power. *GNN-AAA [no LB]* achieves the highest sum rate, although it results in a high violation of the LB constraint (3.8e), compared to nearly 0% for all other schemes (less than 0.1%). This highlights the effectiveness of incorporating the penalty term (3.34) in the loss function to meet (3.8e). While unbalanced UA might provide higher sum rates in certain scenarios, load-balanced UA ensures equitable resource allocation, consistent user experiences, and robust network operations. *GNN-AAA [proposed]* outperforms all benchmarks under the same LB constraints. Specifically, it achieves a higher sum rate than *Model-based* and *DNN* methods due to its superior architecture design and enhanced feature extraction capabilities. Moreover, it outperforms *HGAT*, highlighting the effectiveness of our oversmoothing-aware attention design and adaptive adjustment strategy in improving learning performance. While *HGAT* leverages an attention mechanism that enhances generalization to heterogeneous graph structures and varying neighborhood sizes, and thus yielding better performance than *GNN*, it is not specifically tailored to mitigate oversmoothing.

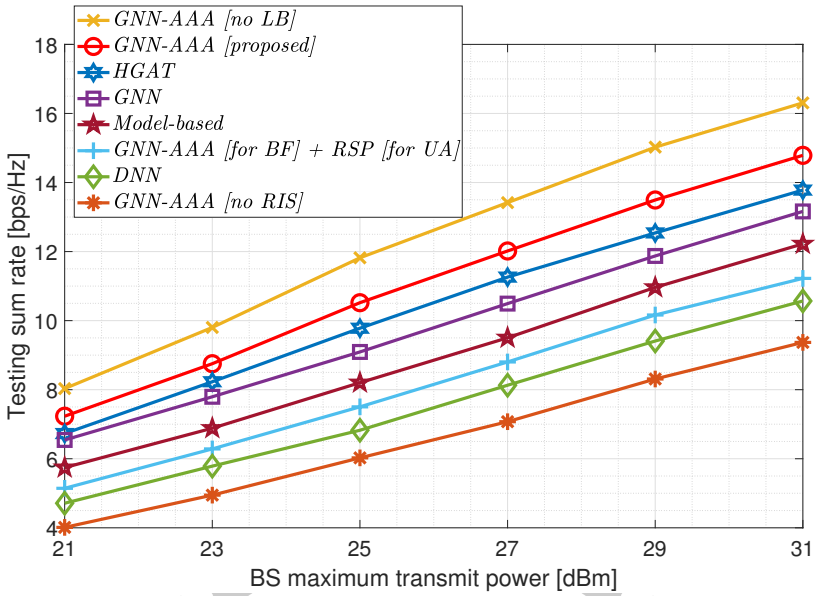


Figure 3.10: Testing sum rate vs. BS maximum transmit power ($P_{\max,b}, \forall b$) for $B = 3$, $U = 6$, and $M = 100$.

Among GNN-based schemes, *GNN-AAA [proposed]* outperforms *GNN*, demonstrating that the AAA mechanism improves learning performance by effectively mitigating oversmoothing. Compared to *GNN-AAA [for BF] + RSP [for UA]*, *GNN-AAA [proposed]* achieves a higher sum rate since RSP-based UA does not consider multiuser interference during UA. This demonstrates the effectiveness of *GNN-AAA [proposed]* in learning channel and interference relationships among all BS–RIS–UE links from pilot signals. Moreover, a comparison between *GNN-AAA [proposed]* and *GNN-AAA [no RIS]* shows the benefits of deploying RIS in the system.

Effects of the Weight λ in (3.33)*:* Fig. 3.11 shows how the testing sum rate and load-balancing satisfaction rate vary with λ. As λ increases, the satisfaction rate increases while the sum rate decreases for all learning-based schemes, reflecting the loss function in (3.33), where higher λ prioritizes load-balancing over sum rate. At $\lambda = 2$, the satisfaction rate reaches approximately 99.9%, demonstrating the effectiveness of the proposed loss function. Thus, $\lambda = 2$ is selected for all simulations to achieve good sum rate performance while satisfying the load-balancing constraint.

Pilot Correlation Effects: Fig. 3.12 examines the pilot correlation effects, as modeled

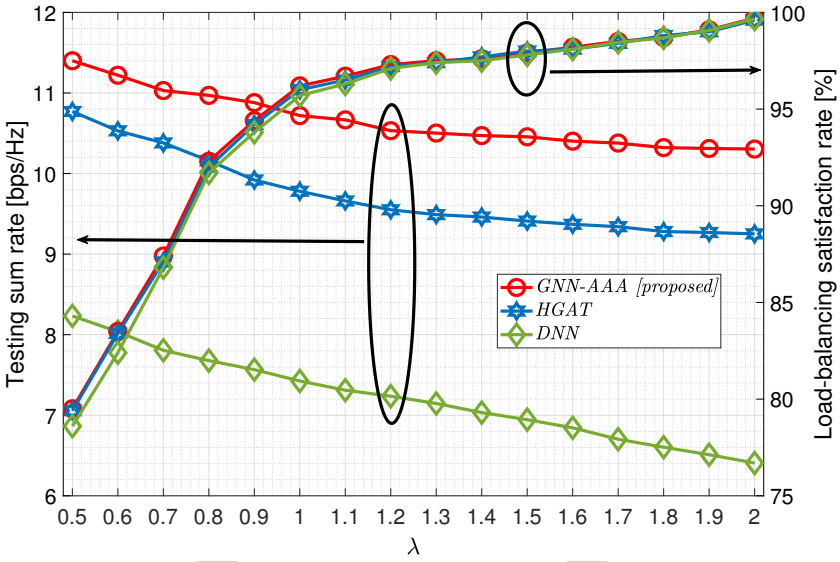


Figure 3.11: Testing sum rate and load-balancing satisfaction rate vs. λ in (3.33) for $B = 3$, $U = 6$, $P_{\max,b} = 25$ dBm $\forall b$, and $M = 100$.

in (3.11), where UE pilots are not perfectly orthogonal. Two pilot types are considered: random and ZC sequences, both of length $L = 31$. The results show that while all methods experience sum-rate degradation under the presence of pilot correlation, *GNN-AAA [proposed]* and *HGAT* are less affected, maintaining performance close to the ideal (uncorrelated) condition. In contrast, *DNN* suffers a more noticeable performance drop, especially with the ZC sequence. This highlights the superior robustness of GNNs against pilot correlations. Furthermore, it is observed that the random sequence outperforms the ZC sequence across all three schemes. This difference primarily arises from the distinct types of additional correlation introduced in each case. In the random sequence case, the correlation follows a Gaussian distribution with zero mean and variance $1/L$, whereas in the ZC sequence case, it is a deterministic value of $1/\sqrt{L}$. Since the random sequence has a higher probability of producing smaller correlation values than the ZC sequence, it generally imposes less negative impact on the learning process of a data-driven model, leading to smaller degradation in sum-rate performance.

Effects of RIS–UE Distances: Fig. 3.13 presents the testing sum-rate performance vs. the distance between the RIS and the UE center. The UE center is fixed at $(0, 0, 0)$, while the RIS position is moved from $(0, -20, 0)$ to $(60, -20, 0)$, as shown in Fig. 3.8. The

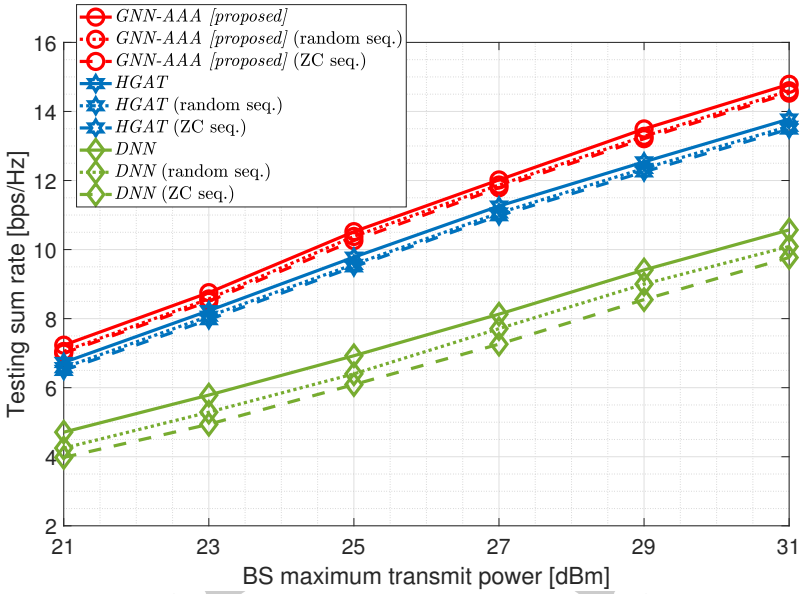


Figure 3.12: Testing sum rate vs. BS maximum transmit power ($P_{\max,b}, \forall b$) for $B = 3$, $U = 6$, and $M = 100$, under different pilot correlation scenarios using random and ZC pilot sequences of length $L = 31$.

performance declines as the distance between the UE center and the RIS increases due to longer reflection paths and higher path loss, which reduce the received signal strength. The maximum sum rate is achieved when the RIS–UE center distance is 20 m, representing the optimal RIS position to assist all BSs along the semicircle in serving the UEs effectively. Furthermore, *GNN-AAA [proposed]* demonstrates adaptability to varying RIS positions, maintaining superior performance compared to other schemes. The comparative trends across all schemes remain consistent with those observed in Fig. 3.10.

Effects of the Number of RIS Elements: Fig. 3.14 presents the testing sum-rate performance vs. the number of RIS passive elements. An increasing number of RIS elements leads to higher sum rates due to enhanced received signal strength and greater flexibility to reshape the wireless channel. Notably, *GNN-AAA [proposed]* with $M = 50$ delivers performance comparable to *Model-based* and *DNN* schemes with $M = 150$, demonstrating its efficiency, as a higher number of RIS elements increases the complexity of passive beamforming design and channel estimation. The proposed GNN-AAA even avoids the need for explicit channel estimation altogether by directly using received pilots from UEs

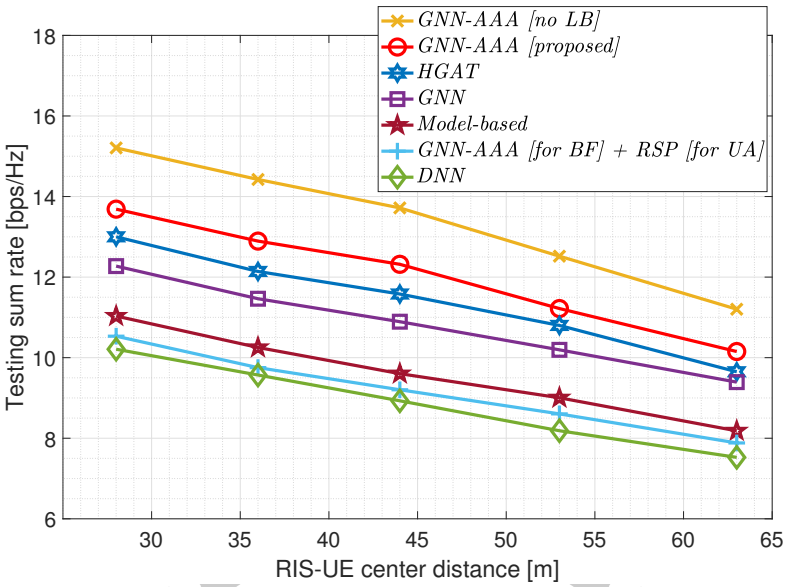


Figure 3.13: Testing sum rate vs. RIS–UE center distance for $B = 3$, $U = 6$, $M = 100$, and $P_{\max,b} = 25$ dBm.

as input, which is a favorable property when adopting a higher number of RIS elements.

Discrete RIS Phases: Fig. 3.15 shows the testing sum rate vs. BS transmit power with 2-bit quantized RIS phase shifts, where continuous phases from each method are discretized to reflect practical RIS implementation. Compared to Fig. 3.10, all methods experience a performance loss, about 1.5 bps/Hz. Despite this, *GNN-AAA [proposed]* retains a consistent advantage over other benchmarks in both continuous and discrete RIS phase conditions. Fig. 3.16 shows the sum-rate performance comparisons between 1-bit quantized RIS phase shifts and 2-bit quantized RIS phase shifts. The performances of the *GNN-AAA [proposed]*, *Model-based* and *DNN* all degrade when using 1-bit quantized RIS phase shifts compared to 2-bit quantized RIS phase shifts, with a performance loss of approximately $15\% \sim 20\%$. Despite this degradation, *GNN-AAA [proposed]* consistently outperforms *Model-based* and *DNN*.

Effects of the imperfect CSI: Fig. 3.17 shows the testing sum rate vs. BS transmit power with the effect of imperfect CSI. Particularly, the imperfect RIS-user channels $\hat{\mathbf{h}}_{R,u} = \mathbf{h}_{R,u} + \delta\mathbf{h}_{R,u}$, where $\delta\mathbf{h}_{R,u} \sim \mathcal{CN}(0, \sigma_e^2)$ are used to calculate the loss function during training. Both *GNN-AAA [proposed]* and *DNN* demonstrate robustness against imperfect CSI because they are trained using datasets composed of various channel re-

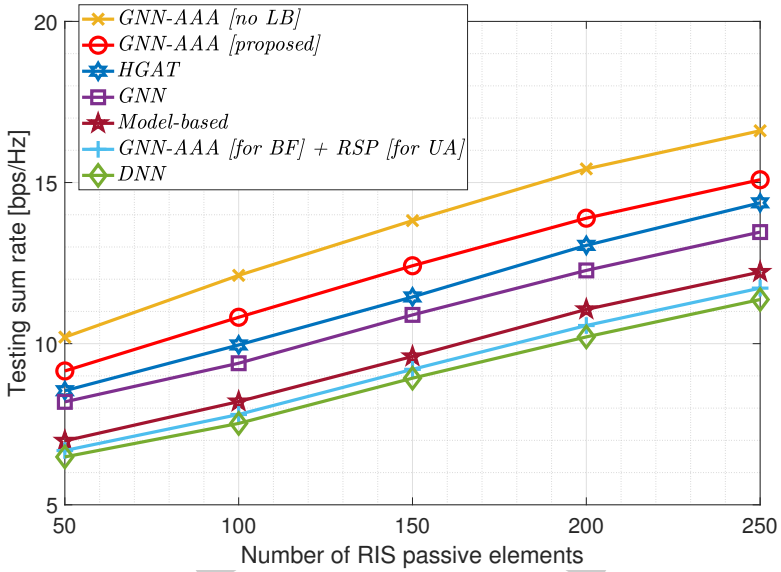


Figure 3.14: Testing sum rate vs. number of RIS passive elements (M) for $B = 3$, $U = 6$, and $P_{\max,b} = 25$ dBm.

alizations. In contrast, the performance of *Model-based* degrades significantly due to its strong reliance on accurate CSI for optimization.

3.4.4 Effects of RIS on User Association

In Fig. 3.10, it was shown that *GNN-AAA [proposed]* achieves a significantly higher sum rate compared to *GNN-AAA [no RIS]*. This improvement stems not only from enhanced received signal strength and reduced interference, as previously mentioned, but also from the added flexibility of RIS in providing additional communication paths that help extend BS coverage and improve UA decisions. An illustrative example, comparing UA decisions by *GNN-AAA [no RIS]* and *GNN-AAA [proposed]*, is shown in Fig. 3.18 and 3.19. Different colors represent the UEs served by each BS, while gray beams indicate blocked paths. For *GNN-AAA [no RIS]*, UE 1 is associated with BS 2, and UE 3 is associated with BS 1 to satisfy the LB constraint. In contrast, for *GNN-AAA [proposed]*, UE 3 selects BS 2, its nearest BS, while UE 1 connects to BS 3 through the RIS which compensates for the path loss caused by the greater distance. This configuration leverages the RIS's flexibility to make more efficient UA decisions, resulting in higher average sum rates. Such

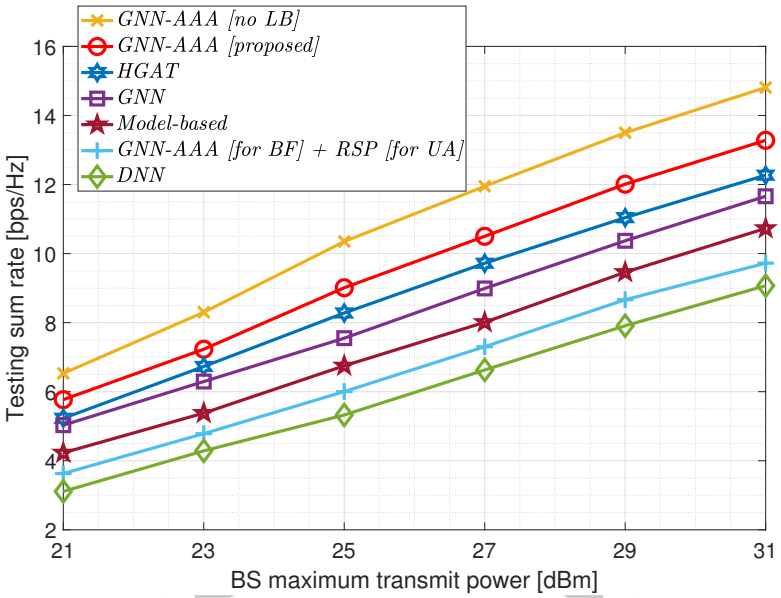


Figure 3.15: Testing sum rate vs. BS maximum transmit power ($P_{\max,b}, \forall b$) for $B = 3$, $U = 6$, and $M = 100$, with 2-bit quantized RIS phase shifts.

improved UA decisions are consistently observed across testing samples, demonstrating the effectiveness of the proposed GNN-AAA in optimizing UA strategies for higher sum rates while maintaining LB considerations in RIS-aided multi-BS multi-UE systems.

3.4.5 Effects of Oversmoothing

The squared norm of node differences, defined in (3.21), is used to quantify the level of oversmoothing in GNN-based methods (higher values indicate less oversmoothing). Table 3.3 presents the performance of *GNN-AAA [proposed]* and other benchmarks in terms of squared norm of node differences and sum rate. Comparing *GNN-AAA [proposed]* and *GNN*, it is seen that the improved sum rate of *GNN-AAA [proposed]* is due to reduced oversmoothing. However, focusing solely on mitigating oversmoothing does not necessarily enhance sum-rate performance. To illustrate this, a variant of *GNN-AAA [proposed]* using non-adaptive aggregation, termed *GNN-AAA w/ non-adaptive aggregation*, is tested, where sparse attention and tuning aggregation are enforced for all nodes in all layers without the mechanism in (3.22), i.e., $\beta_{(b,u)}^{(\tau)}$ is computed for all (b, u) rather than just those in $\mathcal{V}^{(\tau)}$. While *GNN-AAA w/ non-adaptive aggregation* achieves the highest

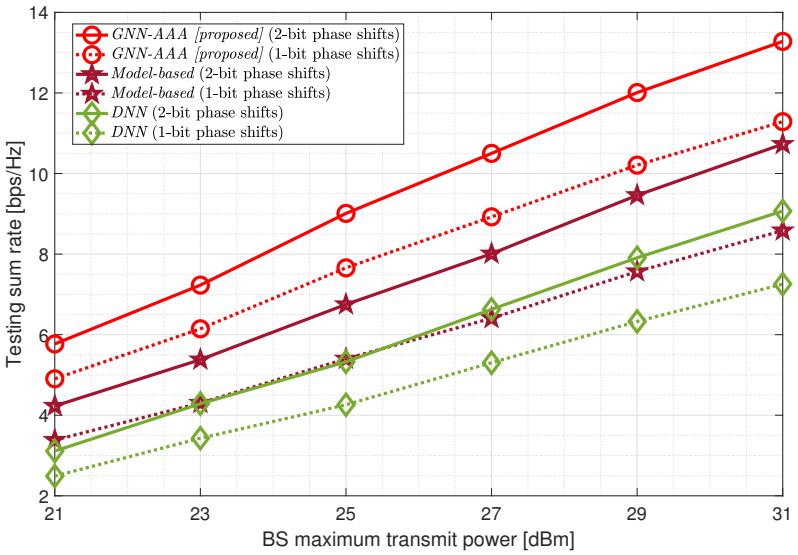


Figure 3.16: Testing sum rate vs. BS maximum transmit power ($P_{\max,b}, \forall b$) for $B = 3$, $U = 6$, and $M = 100$, with 1-bit and 2-bit quantized RIS phase shifts.

squared norm of node differences, meaning least oversmoothing, its sum rate is lower than that of *GNN-AAA [proposed]*. This suggests that for deep GNNs, it is crucial to make judicious decisions on whether and to what extent to aggregate information among neighboring nodes. By monitoring the oversmoothing status of each BS–UE node in every updating layer and addressing it adaptively at the node and layer levels, *GNN-AAA [proposed]* avoids unnecessary sacrifices in information exchange and achieves superior overall performance.

An existing method that tackles oversmoothing by incorporating residual connections into GNNs, termed *GNN w/ residual connections* [13], is also compared. *GNN-AAA [proposed]* outperforms *GNN w/ residual connections* in both squared norm of node differences and sum rate. This is because the residual connections force each node at every layer to combine with initial input features without controlling information exchange, making it less adaptable to individual node states and thus less effective at mitigating oversmoothing.

HGAT mitigates oversmoothing by assigning varying importance to neighboring nodes during aggregation, which helps preserve node feature uniqueness. As a result, it achieves

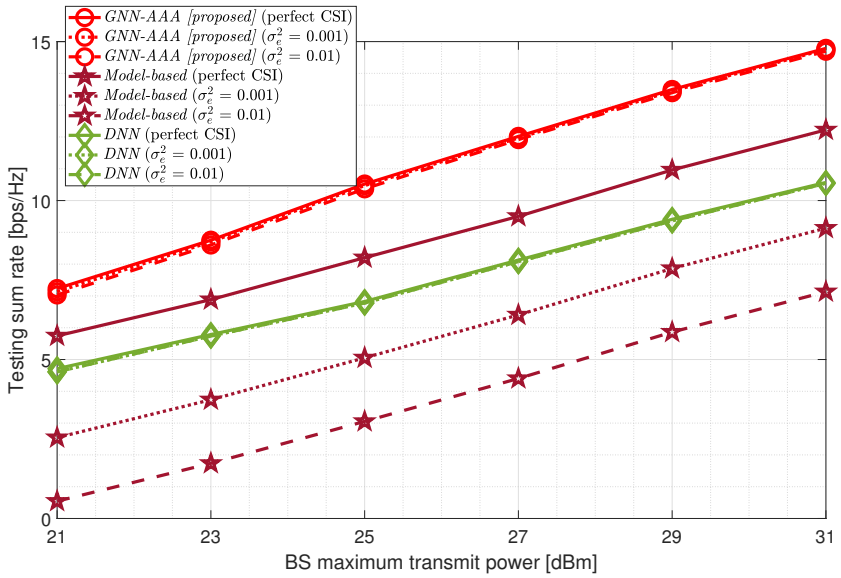


Figure 3.17: Testing sum rate vs. BS maximum transmit power ($P_{\max,b}, \forall b$) for $B = 3$, $U = 6$, and $M = 100$, with the effect of imperfect CSI.

higher squared norm of node differences and sum rate compared to *GNN*. However, it still falls short of *GNN-AAA [proposed]*, which more effectively addresses oversmoothing through its tailored attention weights and adaptive adjustment mechanisms.

3.4.6 Complexity

Table 3.4 reports the average runtime required to obtain a feasible solution for one testing sample under different settings of BSs (B), UEs (U), and RIS elements (M). While incorporating AAA in *GNN-AAA [proposed]* results in a slight increase in runtime compared to *GNN*, it significantly enhances sum-rate performance, as demonstrated previously. The runtime increase from incorporating the AAA mechanism in *GNN-AAA [proposed]* generally agrees with the complexity analysis in Section 4.4.2, where the AAA mechanism contributes roughly 10% of the total computational complexity. This confirms that the AAA mechanism adds minimal overhead to the overall complexity. Furthermore, although *DNN* runs slightly faster than *GNN-AAA [proposed]*, it delivers inferior sum-rate performance. The *Model-based* method has the highest runtime due to its iterative optimization process. *GNN-AAA [no RIS]* achieves the shortest runtime as it bypasses RIS

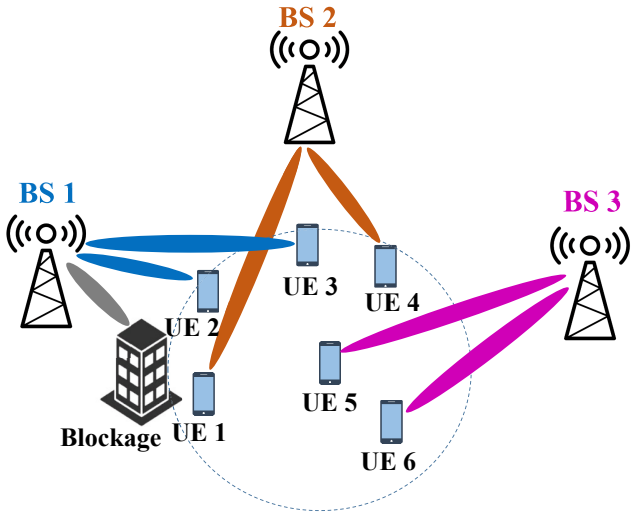


Figure 3.18: Example scenarios to illustrate the effects of RIS on UA in multi-BS multi-UE systems, with $B = 3$, $U = 6$, $M = 100$, and $P_{\max,b} = 25$ dBm. (a) UA for *GNN-AAA [no RIS]*.

phase shift calculations but suffers from the lowest sum-rate performance due to the absence of RIS. As B, U, or M increases, runtime increases for all schemes.

3.4.7 Generalizability

Table 3.5 and Table 3.6 evaluate the generalizability of the proposed GNN-AAA across different numbers of BSs and UEs, respectively, by analyzing cases with mismatched training and testing scenarios. In Table 3.5, the highest sum rate is achieved when the numbers of BSs used for training and testing match (bottom row). However, even with a mismatch, the proposed method maintains comparable performance with minimal degradation, highlighting its ability to be trained on fewer BSs for reduced complexity or when the exact number of BSs in testing is unknown. Table 3.6 shows similar results across different numbers of UEs, demonstrating the practicality of *GNN-AAA [proposed]* since the number of UEs often varies over time and is typically unknown in advance.

Note that *Model-based* and *DNN* approaches do not possess generalization properties. The *Model-based* method use optimization and AO-based algorithms to simplify or decompose the objective function, with their structure inherently tied to the number of BSs

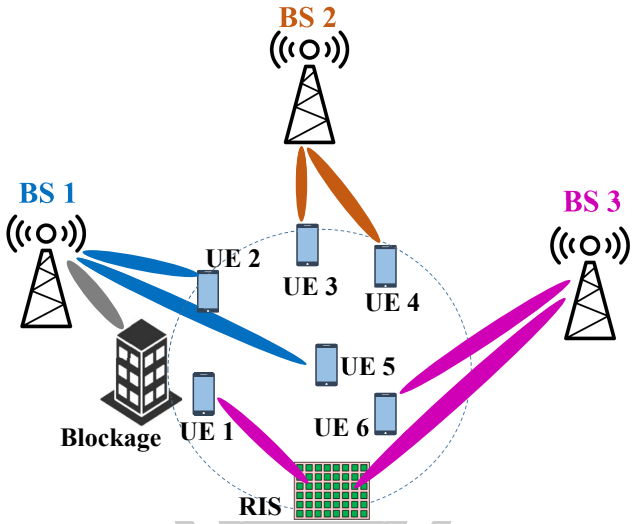


Figure 3.19: Example scenarios to illustrate the effects of RIS on UA in multi-BS multi-UE systems, with $B = 3$, $U = 6$, $M = 100$, and $P_{\max,b} = 25$ dBm. (b) UA for *GNN-AAA [proposed]*.

and UEs. Any change in BS or UE numbers requires rerunning these methods to find a solution. DNN-based methods rely on fully connected layers to map decorrelated pilots to system variables, with input and output structures tied to the number of BSs and UEs. Any change in BS or UE numbers requires redesigning the DNN model architecture and retraining the model.

Table 3.1: List of Symbols

Number of BSs	B
Number of transmit antennas at each BS	N_T
Number of UEs	U
Number of RIS elements	M
The RIS phase shift matrix	$\mathbf{\Phi}$
The mth diagonal element of $\mathbf{\Phi}$	ϕ_m
The BS–UE association indicator	$\delta_{b,u}$
The set of BSs serving the uth UE	$\mathcal{S}_u^{\mathrm{BS}}$
The set of UEs served by the bth BS	$\mathcal{C}_b^{\mathrm{UE}}$
The channel between the bth BS and the RIS	$\mathbf{H}_{b,R}$
The channel between the RIS and the uth UE	$\mathbf{h}_{R,u}$
The channel between the bth BS and the uth UE	$\mathbf{h}_{b,u}$
The beamforming vector of the (b,u)th BS–UE link	$\mathbf{f}_{b,u}$
The achievable rate of the uth UE	R_u
The maximum number of UEs served by the bth BS	$Z_{\max,b}$
The maximum transmit power at the bth BS	$P_{\max,b}$
Length of pilot signals	L
Number of updating layers	Q
The feature extractor for BS–UE nodes in initial layer	$\Psi_U^{(0)}$
The feature extractor for the RIS node in initial layer	$\Psi_R^{(0)}$
The aggregator for BS–UE nodes in τth updating layer	$\Psi_U^{\mathrm{ag},(\tau)}$
The aggregator for the RIS node in τth updating layer	$\Psi_R^{\mathrm{ag},(\tau)}$
The combiner for BS–UE nodes in τth updating layer	$\Psi_U^{\mathrm{cb},(\tau)}$
The combiner for the RIS node in τth updating layer	$\Psi_R^{\mathrm{cb},(\tau)}$
Parameter of feature extractors, aggregators, and combiners	K
The fully-connected linear layer for UA	$\Psi_{\mathrm{UA}}(\cdot)$
The fully-connected linear layer for BF	$\Psi_{\mathrm{BF}}(\cdot)$
The fully-connected linear layer for RIS phase shifts	$\Psi_{\mathrm{RIS}}(\cdot)$
The initial feature of the (b,u)th BS–UE node	$\mathbf{z}_{(b,u)}^{(1)}$
The initial feature of the RIS node	$\mathbf{z}_R^{(1)}$
The (b,u)th BS–UE node feature in τth updating layer	$\mathbf{z}_{(b,u)}^{(\tau)}$
The (b,u)th BS–UE aggregated feature in τth updating layer	$\mathbf{z}_{(b,u)}^{\mathrm{ag},(\tau)}$
The RIS aggregated feature in τth updating layer	$\mathbf{z}_R^{\mathrm{ag},(\tau)}$
The RIS node feature in τth updating layer	$\mathbf{z}_R^{(\tau)}$
The output feature for UA	$\mathbf{z}_{(b,u)}^{\mathrm{out}}$
The output feature for BF	$\mathbf{z}_{(b,u)}^{\mathrm{BF}}$
The output feature for RIS phase shifts	$\mathbf{z}^{\mathrm{RIS}}$

Table 3.2: Simulation Parameters

Number of BSs in training	$B = 3$
Number of UEs in training	$U = 6$
Number of transmit antenna per BS	$N_T = 8$
Carrier frequency	28 GHz
Number of RIS elements (default)	$M = 100$
Number of clusters per channel	$N_{cl} = 5$
Number of rays per cluster	$L_\ell = 10$
Length of each pilot sequence	$L = 31$
Uplink pilot transmission power	15 dBm
Noise power	-85 dBm
BS maximum transmit power (default)	$P_{\max,b} = 25$ dBm, $\forall b \in \mathcal{B}$
Size of the NN modules in the initial layer	$\Psi_U^{(0)} : 2N_T \times 1024 \times 512$ $\Psi_R^{(0)} : 512 \times 1024 \times 512$,
Size of the NN modules in the updating layer	$\Psi_U^{\text{ag},(\tau)}, \Psi_U^{\text{cb},(\tau)}, \Psi_R^{\text{ag},(\tau)}$, $\Psi_R^{\text{cb},(\tau)} : 512 \times 512 \times 512$ for $\tau = 1, 2, \ldots, Q$
Number of updating layers	$Q = 8$
Number of training samples per epoch	10^5
Number of epochs	200
Batch size	1024
Learning rate	10^{-3}, decreasing after 3 epochs by a factor of 0.98
Number of testing samples	10^5
Tunable weight in the loss function	$\lambda = 2$
Maximum number of UEs served by a BS	$Z_{\max,b} = U/B$, $\forall b \in \mathcal{B}$

Table 3.3: Squared norm of Node Differences and Sum Rate for $B = 3$, $U = 6$, $M = 100$, $Q = 8$, and $P_{\max,b} = 25$

Scheme	Squared norm of node differences	Sum rate (bps/Hz)
GNN-AAA [proposed]	9.702	10.519
HGAT	8.421	9.780
GNN	4.622	8.870
GNN w/ residual connections	9.203	9.987
GNN-AAA w/ non-adaptive aggregation	12.052	9.816

Table 3.4: Runtime (Unit: Millisecond)

Scheme	$M = 100$	
	$B = 3, U = 6$	$B = 5, U = 8$
GNN-AAA [proposed]	3.28	3.72
HGAT	3.11	3.30
GNN	3.03	3.21
GNN-AAA [for BF] + RSP [for UA]	3.28	3.43
Model-based	8.66×10^3	12.76×10^3
DNN	3.11	3.36
GNN-AAA [no RIS]	2.89	3.13

Scheme	$M = 200$	
	$B = 3, U = 6$	$B = 5, U = 8$
GNN-AAA [proposed]	3.52	3.93
HGAT	3.35	3.57
GNN	3.25	3.45
GNN-AAA [for BF] + RSP [for UA]	3.47	3.63
Model-based	9.46×10^3	13.83×10^3
DNN	3.42	3.68
GNN-AAA [no RIS]	3.17	3.36

Table 3.5: Sum Rate for *GNN-AAA [proposed]* over Various Numbers of BSs (B) in Training and Testing, with $M = 100$, $U = 6$, and $P_{\max,b} = 25$

B (testing)	2	3	4	5
Sum rate (bps/Hz) B (training) $= 3$	10.116	10.519	10.977	11.445
Sum rate (bps/Hz) B (training) $= B$ (testing)	10.259	10.519	11.202	11.782

Table 3.6: Sum Rate for *GNN-AAA [proposed]* over Various Numbers of UEs (U) in Training and Testing, with $M = 100$, $B = 3$, and $P_{\max,b} = 25$

U (testing)	2	3	4	5	6	7	8	9
Sum rate (bps/Hz) U (training) $= 6$	9.265	9.877	10.242	10.416	10.519	10.674	10.741	11.167
Sum rate (bps/Hz) U (training) $= U$ (testing)	9.838	10.234	10.376	10.490	10.519	11.128	11.448	11.976

Chapter 4

GNN for User Association and Beamforming in Mixed Near- and Far-Field RIS-aided Systems

In this chapter, the proposed GNN-based approach for the ESE mitigation in a RIS-aided XL-MIMO multi-BS multi-UE system is introduced. We try to develop a solution for suppressing the ESE with the joint design of RIS-aided user association and beamforming through a GNN. The system model, problem formulation, methodology, and simulation results are presented in sequence.

4.1 System Model

We consider a RIS-aided downlink mmWave system with B BSs, each equipped with N_T uniform linear arrays (ULA) antennas, a RIS with M passive elements, and U single-antenna UEs, as illustrated in Fig. 4.1. The size of the transmit antenna array at each BS is $D_{\mathrm{BS}} = (N_T - 1)d_{\mathrm{BS}}$, where $d_{\mathrm{BS}} = \lambda/2$ is the antenna spacing, with λ denoting the wavelength. UEs may be located in the near-field or far-field region with respect to their serving BSs, with the boundary determined by the Rayleigh distance $d_{\mathrm{Rayleigh}} = \frac{2D_{\mathrm{BS}}^2}{\lambda}$, whereas both UEs and BSs are assumed to be in the far-field region with respect to the RIS. Let the sets of all BSs and UEs be $\mathcal{B} = \{1, 2, \ldots, B\}$ and $\mathcal{U} = \{1, 2, \ldots, U\}$,

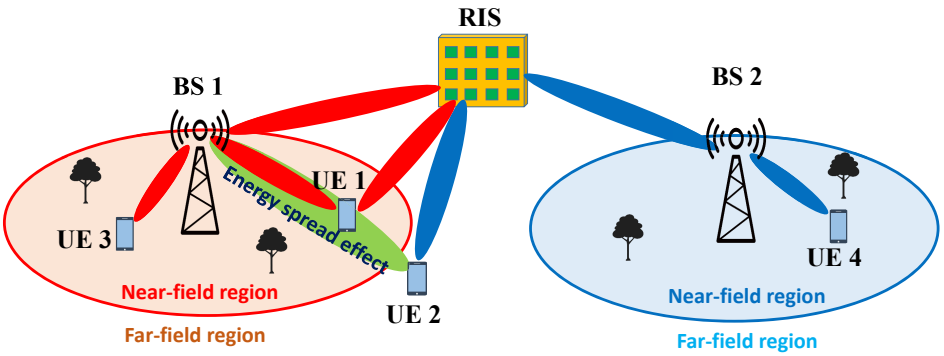


Figure 4.1: A RIS-aided multiuser mmWave XL-MIMO system.

respectively.

The near-field channel between the bth BS and the uth UE located in the near-field region is given by [69]

$$\mathbf{h}_{b,u}^{\mathrm{NF}} = \rho_{b,u}\sqrt{\frac{N_T}{L^{\mathrm{NF}}}}\sum_{\ell=1}^{L^{\mathrm{NF}}}\kappa_\ell \mathbf{a}_{\mathrm{NF}}(r_{b,u,\ell},\xi_{b,\ell}). \tag{4.1}$$

Here, κ_ℓ is the complex gain of the ℓth ($\ell = 1,\ldots,L^{\mathrm{NF}}$) path of the near-field channel, modeled as a Gaussian random variable with zero mean and unit variance; $\rho_{b,u}$ is the path loss of the channel between the bth BS and the uth UE; $r_{b,u,\ell}$ is the distance along the ℓth path between the center of the antenna array at the bth BS and the uth UE; $\xi_{b,\ell}$ is the AoD of the ℓth path at the bth BS, defined as the angle between the line connecting the antenna array center at the bth BS to the observation point and the pointing direction of the antenna array; and $\left[\mathbf{a}_{\mathrm{NF}}(r_{b,u,\ell},\xi_{b,\ell})\right]_n = \frac{1}{\sqrt{N_T}}e^{j\frac{2\pi}{\lambda}(r_{b,u,\ell,n}-r_{b,u,\ell})}$ is the array response vector between the bth BS and the uth UE following the spherical wave assumption, where $r_{b,u,\ell,n} = \sqrt{r_{b,u,\ell}^2+\eta_n^2 d_{\mathrm{BS}}^2 - 2\eta_n d_{\mathrm{BS}} r_{b,u,\ell}\cos(\xi_{b,\ell})}$ is the distance along the ℓth path between the nth antenna of the bth BS and the uth UE, with $\eta_n = (2n - N_T - 1)/2$ for $n = 1,\ldots,N_T$.

The far-field channels between the bth BS and the RIS, the RIS and the uth UE, and the bth BS and the uth UE are given respectively by [70]

$$\mathbf{H}_{b,R} = \rho_{b,R}\sqrt{\frac{N_T M}{N_\gamma}}\sum_{\ell=1}^{N_{cl}}\sum_{i=1}^{L_\ell}\alpha_{\ell,i}\mathbf{a}_{\mathrm{RIS}}\left(\psi_{\ell,i}^{\mathrm{AoA}},\theta_{\ell,i}^{\mathrm{AoA}}\right)\times\mathbf{a}_{\mathrm{FF}}^H(\xi_{b,\ell,i}),$$

$$\mathbf{h}_{R,u} = \rho_{R,u}\sqrt{\frac{M}{N_\gamma}}\sum_{\ell=1}^{N_{cl}}\sum_{i=1}^{L_\ell}\alpha_{\ell,i}\mathbf{a}_{\mathrm{RIS}}\left(\psi_{\ell,i}^{\mathrm{AoD}},\theta_{\ell,i}^{\mathrm{AoD}}\right),$$

and

$$\mathbf{h}_{b,u}^{\mathrm{FF}} = \rho_{b,u}\sqrt{\frac{N_T}{N_\gamma}}\sum_{\ell=1}^{N_{cl}}\sum_{i=1}^{L_\ell}\alpha_{\ell,i}\mathbf{a}_{\mathrm{FF}}(\xi_{b,\ell,i}).$$

Here, N_{cl} is the number of clusters and L_ℓ is the number of rays for the ℓth cluster; $N_\gamma = \sum_{\ell=1}^{N_{cl}} L_\ell$; $\alpha_{\ell,i}$ is the complex gain of the ith ray in the ℓth cluster, modeled as Gaussian with zero mean and unit variance; $\rho_{b,R}$ and $\rho_{R,u}$ are the path loss of the channels $\mathbf{H}_{b,R}$ and $\mathbf{h}_{R,u}$, respectively; $\psi_{\ell,i}^{\mathrm{AoD}}, \theta_{\ell,i}^{\mathrm{AoD}}$ and $\psi_{\ell,i}^{\mathrm{AoA}}, \theta_{\ell,i}^{\mathrm{AoA}}$ denote the azimuth and elevation angles of departure and arrival at the RIS for the ith ray in the ℓth cluster; $\xi_{b,\ell,i}$ is the AoD at the bth BS for the ith ray in the ℓth cluster; $[\mathbf{a}_{\mathrm{FF}}(\xi_{b,\ell})]_{n_t} = e^{j\frac{2\pi n_t d_{\mathrm{BS}}}{\lambda}\cos(\xi_{b,\ell})}$ for $n_t = 1,\ldots,N_T$ is the array response vector for the far-field channel of the bth BS; and $[\mathbf{a}_{\mathrm{RIS}}(\psi,\theta)]_n = e^{j\frac{2\pi d_{\mathrm{RIS}}}{\lambda}(n_1\sin(\psi)\cos(\theta)+n_2\sin(\theta))}$ is the array response vector for the RIS, where $n_1 = \mathrm{mod}(n-1,\sqrt{M})$, $n_2 = \lfloor (n-1)/\sqrt{M}\rfloor$, and $d_{\mathrm{RIS}} = \lambda/2$ is the RIS element spacing.

The RIS is modeled by the diagonal phase-shift matrix $\mathbf{\Phi} = \mathrm{diag}\{\phi_1,\phi_2,\ldots,\phi_M\} = \mathrm{diag}\left\{e^{j\theta_1},\ldots,e^{j\theta_M}\right\}$. The BS–UE association is defined by $\delta_{b,u} \in \{0,1\}$ for $u \in \mathcal{U}$ and $b \in \mathcal{B}$, where $\delta_{b,u} = 1$ if the uth UE is served by the bth BS and $\delta_{b,u} = 0$ otherwise. The set of BSs serving the uth UE is $\mathcal{S}_u^{\mathrm{BS}} = \{b \mid \delta_{b,u} = 1, b \in \mathcal{B}\}$, while the set of UEs served by the bth BS is $\mathcal{C}_b^{\mathrm{UE}} = \{u \mid \delta_{b,u} = 1, u \in \mathcal{U}\}$.

The received signal at the uth UE is given by

$$\begin{aligned} y_u = &\sum_{b\in\mathcal{S}_u^{\mathrm{BS}}}\left(\mathbf{h}_{b,u}^H + \mathbf{h}_{R,u}^H\mathbf{\Phi}\mathbf{H}_{b,R}\right)\mathbf{f}_{b,u}s_u \\ &+ \sum_{b\in\mathcal{S}_u^{\mathrm{BS}}}\sum_{u'\in\mathcal{C}_b^{\mathrm{UE}},u'\neq u}\left(\mathbf{h}_{b,u}^H + \mathbf{h}_{R,u}^H\mathbf{\Phi}\mathbf{H}_{b,R}\right)\mathbf{f}_{b,u'}s_{u'} \\ &+ \sum_{b'\in\mathcal{B}\setminus\mathcal{S}_u^{\mathrm{BS}}}\sum_{u'\in\mathcal{C}_{b'}^{\mathrm{UE}}}\left(\mathbf{h}_{b',u}^H + \mathbf{h}_{R,u}^H\mathbf{\Phi}\mathbf{H}_{b',R}\right)\mathbf{f}_{b',u'}s_{u'} + \mathrm{n}_u, \end{aligned} \tag{4.2}$$

where $\mathbf{h}_{b,u}$ equals $\mathbf{h}_{b,u}^{\mathrm{NF}}$ or $\mathbf{h}_{b,u}^{\mathrm{FF}}$ depending on whether the uth UE is in the near-field or far-field region, s_u is the uth UE's data symbol with $\mathbb{E}(s_u) = 0$, $\mathbb{E}(|s_u|^2) = 1$, and $\mathbb{E}(s_u^H s_{u'}) = 0, \forall u \neq u'$, $\mathbf{f}_{b,u} \in \mathbb{C}^{N_T\times 1}$ is the beamformer at the bth BS for the uth UE, and

n_u is the AWGN with zero mean and variance σ^2. The second and third terms of (4.2) correspond to the intra-cell and inter-cell interference, respectively. The SINR of the uth UE, denoted by γ_u, can be derived based on (4.2), and the achievable rate of the uth UE is $R_u = \log_2(1+\gamma_u)$.

4.2 Energy Spread Effect (ESE)

With mixed near- and far-field UEs, the mismatch between near-field channels and far-field beamforming leads to the ESE, which increases intra-cell interference as expressed in the second term of (4.2). For a far-field beamforming vector $\mathbf{f}$ targeting at the desired far-field direction θ', the beam focusing patterns at the far-field observation direction θ and at the near-field observation location (r,θ) are given respectively by $G_{\mathrm{FF,FF}}(\theta;\mathbf{f}) = \frac{1}{N_T}\left|\mathbf{f}^H\mathbf{a}_{\mathrm{FF}}(\theta)\right|$ and $G_{\mathrm{FF,NF}}(r,\theta;\mathbf{f}) = \frac{1}{N_T}\left|\mathbf{f}^H\mathbf{a}_{\mathrm{NF}}(r,\theta)\right|$. Fig. 4.2 illustrates that while the far-field beam pattern $G_{\mathrm{FF,FF}}(\theta;\mathbf{f})$ focuses on the target far-field direction, the near-field response $G_{\mathrm{FF,NF}}(r,\theta;\mathbf{f})$ spreads around it (i.e., energy spread), causing interference to nearby near-field UEs. Note that the mismatch between far-field channels and near-field beamforming can also cause intra-cell interference to far-field UEs. However, due to higher path loss of far-field channels, this interference is relatively minor and can be considered negligible compared to the impact of far-field beamforming on near-field UEs, i.e., ESE. Therefore, to maximize the sum rate of mixed near- and far-field systems, it is imperative to mitigate the ESE. We propose a GNN-based solution with two key features to address this, i.e., *CBA aggregation* and *NAA enhanced training*.

4.3 Problem Description

We aim to maximize the system sum rate in the presence of both near- and far-field users by jointly optimizing the BS–UE association, BS transmit beamformers, and RIS phase

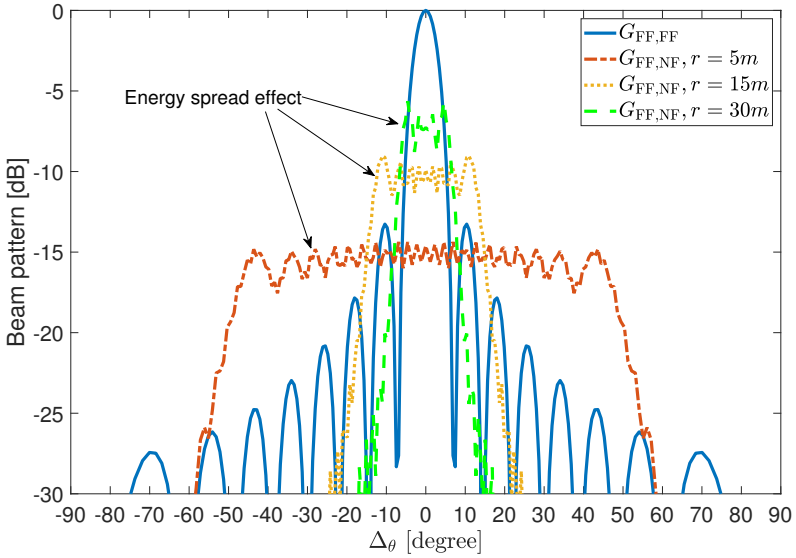


Figure 4.2: Beam patterns of a 128-element ULA with far-field beamforming, evaluated at both far-field and near-field observation points. The desired angle is $\theta' = 90°$, with the far-field observation taken at 100 m and near-field observations at $r = 5, 15, 30$ m. The angular offset is defined as $\Delta_\theta = \theta' - \theta$.

shifts, subject to load balancing constraints. The problem is formulated as:

$$\max_{\{\delta_{b,u}\},\{\mathbf{f}_{b,u}\},\mathbf{\Phi}} \quad \sum_{u\in\mathcal{U}} R_u \tag{4.3a}$$

$$\text{s.t.} \quad \delta_{b,u} \in \{0,1\}, \quad \forall b \in \mathcal{B},\ u \in \mathcal{U}, \tag{4.3b}$$

$$|\phi_m| = 1, \quad \forall m \in \{1,\ldots,M\}, \tag{4.3c}$$

$$\sum_{b\in\mathcal{B}} \delta_{b,u} = 1, \quad \forall u \in \mathcal{U} \tag{4.3d}$$

$$\sum_{u\in\mathcal{C}_b^{\mathrm{UE}}} \delta_{b,u} \le Z_{\max}, \quad \forall b \in \mathcal{B} \tag{4.3e}$$

$$\sum_{u\in\mathcal{U}} \delta_{b,u} ||\mathbf{f}_{b,u}||^2 \le P_{\max}, \quad \forall b \in \mathcal{B} \tag{4.3f}$$

where (4.3b)–(4.3f) denote the BS–UE association, RIS phase, single-BS service, load balancing (limited to $Z_{\max}$ UEs per BS), and BS power ($P_{\max}$) constraints, respectively.

4.4 Methodology

In this section, we present the proposed GNN with CBA aggregation and NAA-enhanced training, to jointly optimize user association, transmit beamforming, and RIS phase adjustments for solving Problem (4.3) and explicitly handling the ESE, as detailed below.

4.4.1 GNN Design

4.4.1.1 GNN Structure

We model the considered system as an undirected complete graph with one RIS node and $B \times U$ BS–UE nodes. The GNN input is the set of decorrelated received pilots at all BSs, obtained without explicit CSI, using mutually orthogonal unit-norm UE pilots. The decorrelated pilot corresponding to the uth UE at the bth BS can be derived as $\widetilde{\mathbf{y}}_{b,u} = \mathbf{h}_{b,u}^{*} + \mathbf{H}_{b,R}^{T}\mathbf{\Phi}\mathbf{h}_{R,u}^{*} + \mathbf{n}_{b,u}$, with $\mathbf{n}_{b,u}$ denoting noise. The overall structure of the proposed GNN is shown in Fig. 4.3-4.6, comprising initial layer, update layers, and output layer.

Initial Layer: As depicted in Fig. 4.3, the initial feature of the (b,u)th BS–UE node is extracted from the decorrelated pilot as

$$\mathbf{z}_{(b,u)}^{(0)} = \Psi_{U}^{(0)}\left(\left[\Re\{\widetilde{\mathbf{y}}_{b,u}^{T}\}, \Im\{\widetilde{\mathbf{y}}_{b,u}^{T}\}\right]^{T}\right),\ \forall b \in \mathcal{B},\ u \in \mathcal{U}, \tag{4.4}$$

where $\Psi_{U}^{(0)}$ is a feature extractor implemented by neural networks. The initial feature of the RIS node contains the aggregated initial features of BS–UE nodes through an element-wise mean function $f^{(0)}(\cdot)$ to preserve permutation invariance, followed by a feature extractor $\Psi_{R}^{(0)}(\cdot)$, i.e.,

$$\mathbf{z}_{R}^{(0)} = \Psi_{R}^{(0)}\left(f^{(0)}\left(\mathbf{z}_{(1,1)}^{(0)}, \ldots, \mathbf{z}_{(b,u)}^{(0)}, \ldots, \mathbf{z}_{(B,U)}^{(0)}\right)\right). \tag{4.5}$$

4.4.1.2 Output of the GNN

Output Layer: The output layer comprises three blocks for user association, transmit beamforming, and RIS phase shifts, respectively, as shown in Fig. 4.6. User association is obtained via a fully-connected (FC) layer with softmax output. Specifically, after Q update layers, the feature of the uth UE is collected as $\mathbf{z}_{(b,u)}^{\text{out}} = \left[\mathbf{z}_{(b,u)}^{(Q)}, \mathbf{z}_{u}^{(Q)}\right]^{T}$, where

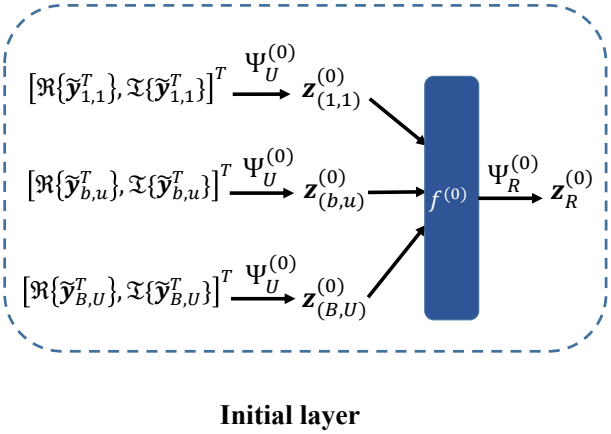


Figure 4.3: Details of the initial layer in the proposed GNN.

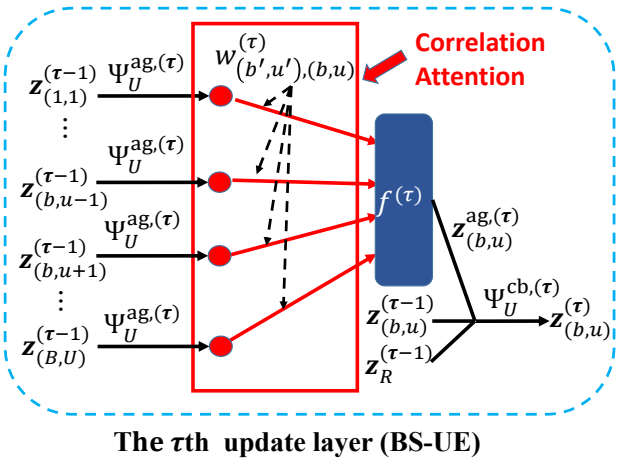


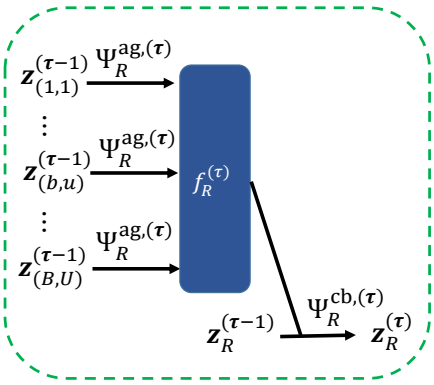
Figure 4.4: Details of the τth updating layer for the BS–UE nodes in the proposed GNN.

$\mathbf{z}_u^{(Q)} = \sum_{b=1}^{B} \mathbf{z}_{(b,u)}^{(Q)}$ represents its synthesized feature. Then, $\mathbf{z}_{(b,u)}^{\text{out}}$ is mapped to a scalar via function $\Psi_{\text{UA}}(\cdot)$ followed by softmax, producing $\widetilde{\delta}_{b,u} = \text{Softmax}\big(\Psi_{\text{UA}}(\mathbf{z}_{(b,u)}^{\text{out}})\big)$ for $b = 1, \ldots, B$. The final user association result for the uth UE is chosen as the BS that maximizes $\widetilde{\delta}_{b,u}$, i.e.,

$$\delta_{b_u^*,u} = 1, \quad \text{for } b_u^* = \arg\max_{b \in \mathcal{B}} \widetilde{\delta}_{b,u}, \tag{4.6}$$

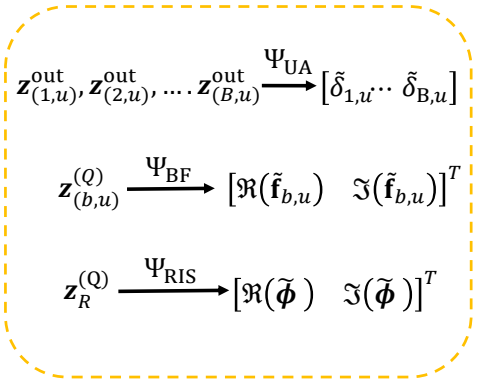
and $\delta_{b,u} = 0$, for other $b \neq b_u^*$, $b \in \mathcal{B}$.

The transmit beamformer for the (b_u^*, u)th BS–UE link is generated by mapping the feature of the (b_u^*, u)th BS–UE node after Q update layers through function $\Psi_{\text{BF}}(\cdot)$ implemented by a single FC layer, i.e., $\mathbf{z}_{(b_u^*,u)}^{\text{BF}} = \Psi_{\text{BF}}(\mathbf{z}_{(b_u^*,u)}^{(Q)})$. The beamformer of the bth BS for



The τth update layer (RIS)

Figure 4.5: Details of the τth updating layer for the RIS node in the proposed GNN.



Output layer

Figure 4.6: Details of the output layer in the proposed GNN.

the uth UE, prior to power control, is given by $\widetilde{\mathbf{f}}_{b_u^*,u} = [\mathbf{z}_{(b_u^*,u)}^{\mathrm{BF}}]_{1:N_T} + j \cdot [\mathbf{z}_{(b_u^*,u)}^{\mathrm{BF}}]_{(N_T+1):2N_T}$. Then, it is normalized to meet the power constraint (4.3f):

$$\mathbf{f}_{b_u^*,u} = \eta_{b_u^*} \widetilde{\mathbf{f}}_{b_u^*,u}, \tag{4.7}$$

where $\eta_{b_u^*} = \sqrt{P_{\max}} / \sum_{u \in \mathcal{C}_{b_u^*}^{\mathrm{UE}}} ||\widetilde{\mathbf{f}}_{b_u^*,u}||$ is the normalization factor.

The RIS phase shifts are obtained by first mapping the RIS node feature after Q update layers through function $\Psi_{\mathrm{RIS}}(\cdot)$ implemented by a single FC layer, i.e., $\mathbf{z}_R^{\mathrm{RIS}} = \Psi_{\mathrm{RIS}}(\mathbf{z}_R^{(Q)})$. Then, the RIS phase shifts, prior to length normalization, are given by $\left[\widetilde{\phi}_1, \ldots, \widetilde{\phi}_M\right]^T = [\mathbf{z}_R^{\mathrm{RIS}}]_{1:M} + j \cdot [\mathbf{z}_R^{\mathrm{RIS}}]_{(M+1):2M}$. These are normalized to unit length, i.e., $\phi_m = \widetilde{\phi}_m / \left|\widetilde{\phi}_m\right|$ for $m = 1, \ldots, M$, to meet the constraint (4.3c).

4.4.1.3 Solution for ESE: Attention Mechanism Design

Update Layers: The initial features are processed through Q update layers that perform feature aggregation and combination. *CBA* is incorporated to mitigate the ESE. Based on the beam pattern analysis in Fig. 4.2, it is observed that correlations between channel response vectors indicate potential interference, with higher correlation implying stronger interference. To exploit this, we encode correlations as attention weights to allow the GNN to prioritize information from interfering nodes. Since explicit CSI is unavailable, these weights are computed from the correlations of decorrelated pilot signals, which still preserve critical channel information.

The proposed CBA weight for the (directed) edge from the (b', u')th BS–UE node to the (b, u)th BS–UE node at the τth ($\tau = 1, \ldots, Q$) update layer is given by

$$w^{(\tau)}_{(b',u'),(b,u)} = \frac{|\widetilde{\mathbf{y}}^H_{b',u'} \widetilde{\mathbf{y}}_{b,u}|}{\sum_{\beta \in \mathcal{B}, \mu \in \mathcal{U}, (\beta,\mu) \notin \{(b',u'),(b,u)\}} |\widetilde{\mathbf{y}}^H_{\beta,\mu} \widetilde{\mathbf{y}}_{b,u}|}, \tag{4.8}$$

which signifies the importance of the edge from the (b', u')th to the (b, u)th BS–UE node relative to the aggregated importance of the remaining edges toward the (b, u)th node. Then, the aggregation at the (b, u)th BS–UE node is expressed as

$$\mathbf{z}^{\text{ag},(\tau)}_{(b,u)} = f^{(\tau)} \left(\left\{ w^{(\tau)}_{(b',u'),(b,u)} \Psi^{\text{ag},(\tau)}_U \left(\mathbf{z}^{(\tau-1)}_{(b',u')} \right) \right\}_{(b',u') \neq (b,u)} \right), \tag{4.9}$$

where $f^{(\tau)}(\cdot)$ is a summation function and $\Psi^{\text{ag},(\tau)}_U(\cdot)$ is an aggregation operator implemented by neural networks. The features in (4.9) aggregated from the other graph nodes are then combined with the features at the (b, u)th BS–UE node from the previous update layer, $\mathbf{z}^{(\tau-1)}_{(b,u)}$, as well as the RIS node feature from the previous update layer, $\mathbf{z}^{(\tau-1)}_R$, i.e.,

$$\mathbf{z}^{(\tau)}_{(b,u)} = \Psi^{\text{cb},(\tau)}_U \left(\mathbf{z}^{(\tau-1)}_{(b,u)}, \mathbf{z}^{\text{ag},(\tau)}_{(b,u)}, \mathbf{z}^{(\tau-1)}_R \right), \tag{4.10}$$

where $\Psi^{\text{cb},(\tau)}_U(\cdot)$ is a combination operator implemented by neural networks. This process is illustrated in Fig. 4.4.

The RIS node features are updated in a similar fashion, by combining the aggregated BS–UE node features and the RIS node feature from the previous update layer. As shown in Fig. 4.5, the RIS node feature at the τth update layer is given by

$$\mathbf{z}^{(\tau)}_R = \Psi^{\text{cb},(\tau)}_R \left(\mathbf{z}^{(\tau-1)}_R, f^{(\tau)}_R \left(\left\{ \Psi^{\text{ag},(\tau)}_R \left(\mathbf{z}^{(\tau-1)}_{(b,u)} \right) \right\}_{b \in \mathcal{B}, u \in \mathcal{U}} \right) \right), \tag{4.11}$$

where $\Psi_R^{\mathrm{cb},(\tau)}(\cdot)$ and $\Psi_R^{\mathrm{ag},(\tau)}(\cdot)$ are combinator and aggregator, respectively, and $f_R^{(\tau)}(\cdot)$ is a permutation-invariant element-wise mean function.

4.4.1.4 Solution for ESE: Penalty Design for Enhancing Training

Model Training: The proposed GNN is trained in an unsupervised manner using pilot-signal realizations. Its loss function integrates the objective in (4.3a), a penalty to ensure the load-balancing constraint (4.3e) is met, and a penalty to enhance near-field beamforming, given by

$$\mathcal{L} = -\sum_{u\in\mathcal{U}} R_u + \lambda_1 \cdot \sum_{b\in\mathcal{B}} \max\Big(0, \sum_{u\in\mathcal{C}_b^{\mathrm{UE}}} \delta_{b,u} - Z_{\max}\Big) - \lambda_2 \cdot \sum_{u\in\mathcal{U}_{\mathrm{NF}}, b\in\mathcal{S}_u^{\mathrm{BS}}} \left|\mathbf{a}_{b,u}^H \mathbf{f}_{b,u}\right|, \tag{4.12}$$

where $\mathbf{a}_{b,u} = \sum_{\ell=1}^{L^{\mathrm{NF}}} \mathbf{a}_{\mathrm{NF}}(r_{b,u,\ell}, \xi_{b,\ell})$, λ_1 and λ_2 are weighting factors for the two penalties, and $\mathcal{U}_{\mathrm{NF}}$ denotes the set of near-field UEs. The second penalty exploits angle- and distance-dependent near-field beam patterns to sharpen beam focus at target points, enhancing near-field received signal strength and mitigating the ESE. This NAA-enhanced training is not applied to far-field UEs, as sharpening far-field beams still causes interference to nearby-angle near-field UEs (Fig. 4.2). Note that the computation of the second penalty requires UE location knowledge during training, but neither location nor explicit CSI is needed at inference.

4.4.2 Complexity Analysis

In this section, we analyze the computational complexity of the proposed GNN method. The total complexity results from the sum of computations in the initial layer, updating layers, and output layer. The analysis can be expressed in terms of the number of UEs, U, the number of BSs, B, the length of a feature vector in each updating layer, K, the number of transmit antennas at each BS, N_T, the number of updating layers, Q, and the number of RIS elements, M.

Initial Layer: The computations involve input feature extraction for each BS–UE node in (4.4) and feature aggregation and extraction for the RIS node in (4.5). The computa-

tions of input feature extraction by $\Psi_U^{(0)}$ of size $2N_T \times 2K \times K$ in (4.4) include multiplications with neuron weights and ReLU activation, which require $\mathcal{O}(2K^2 + 4N_T K)$ and $\mathcal{O}(3K)$ operations for each BS–UE node, respectively. The complexity for all BS–UE nodes is therefore $\mathcal{O}\big(BU(2K^2 + (4N_T + 3)K)\big)$. The computations in (4.5) include computing the element-wise mean of the BS–UE node input features, which requires $\mathcal{O}(BUK)$ operations, followed by feature extraction using $\Psi_R^{(0)}$ of size $K \times 2K \times K$ with ReLU activation, which requires $\mathcal{O}(4K^2 + 3K)$ operations. Summing these up, the overall complexity of the initial layer is approximated as $\mathcal{O}\Big(BU\big(2K^2 + (4N_T + 4)K\big)\Big)$.

Updating Layers: The computations for the BS–UE nodes and the RIS node are analyzed separately. The BS–UE node complexity arises from executing the AAA mechanism, as detailed below.

4.4.2.1 CBA Aggregation

The complexity involves aggregating features from neighboring nodes and multiplying the aggregated features by the attention weights as in (4.9), and computing the attention weights themselves as in (4.8). The computations in (4.9) include, for each BS–UE node, extracting features from $BU - 1$ neighboring nodes using $\Psi_U^{\mathrm{ag},(\tau)}$ of size $K \times K \times K$ with ReLU activation, which requires $\mathcal{O}(2K^2 + 2K)$ operations, and multiplying the aggregated features by the attention weights followed by aggregation, which requires $\mathcal{O}\big((BU - 1)K\big)$ and $\mathcal{O}\big((BU - 2)K\big)$ operations, respectively. In (4.8), computing pairwise inner product between all BU nodes' feature vectors involves $\mathcal{O}(2BUN_T)$ operations, and the softmax operation adds $\mathcal{O}(BU)$ operations. Overall, the total complexity for all BS–UE nodes is approximated as $\mathcal{O}\big(B^2U^2(2N_T + 2K + 1) + BU(2K^2 + 2K)\big)$.

The complexity of the feature combination for each BS–UE node using $\Psi_U^{\mathrm{cb},(\tau)}$ of size $3K \times K \times K$ with ReLU activation, which requires $\mathcal{O}(4K^2 + 2K)$ operations. Thus, the total complexity for all BS–UE nodes is approximated as $\mathcal{O}\big(BU(4K^2 + 2K)\big)$.

Summing up these complexity orders, the total complexity for the BS–UE nodes across the Q updating layers is $\mathcal{O}\Big(QB^2U^2\big(2N_T + 2K + 1\big) + QBU(6K^2 + 4K)\Big)$.

The RIS node complexity involves aggregating features from all BS–UE nodes and combining the aggregated feature with the RIS node's feature from the previous layer

as in (4.11). In (4.11), the feature aggregation using $\Psi_R^{\text{ag},(\tau)}$ of size $K \times K \times K$ with ReLU activation requires $\mathcal{O}\big(BU(2K^2+2K)\big)$ operations, while the element-wise mean operation requires $\mathcal{O}(BUK)$ operations. The combination using $\Psi_R^{\text{cb},(\tau)}$ of size $2K \times K \times K$ with ReLU activation requires $\mathcal{O}(3K^2+2K)$ operations. Summing these up, the total complexity for the RIS node across the Q updating layers is $\mathcal{O}\big(QBU(2K^2+3K)+Q(3K^2+2K)\big)$.

Output Layer: The computations include UA, beamforming, and RIS phase shift calculations for the output. The UA complexity comprises computing the FC linear function $\Psi_{\text{UA}}(\cdot)$, which requires $\mathcal{O}(2BK)$ operations, and the softmax function, which requires $\mathcal{O}(B)$ operations, for one UE. This sums up to $\mathcal{O}\big(BU(2K+1)\big)$ for all UEs. The beamforming complexity comprises computing $\Psi_{\text{BF}}(\cdot)$, which requires $\mathcal{O}(2N_TK)$ operations, followed by normalization to satisfy the power constraint as in (4.7), which requires $\mathcal{O}(N_T)$ operations, for each BS–UE node. This sums up to $\mathcal{O}\big(BUN_T(2K+1)\big)$ for all BS–UE nodes. The RIS phase shift complexity involves computing $\Psi_{\text{RIS}}(\cdot)$, which requires $\mathcal{O}(2MK)$ operations, followed by normalization to enforce the unit-modulus constraint, which requires $\mathcal{O}(M)$ operations. Adding up these numbers, the total complexity for the output layer is $\mathcal{O}\Big(BU\big((2N_T+2)K+(N_T+1)\big)+M(2K+1)\Big)$.

In summary, the overall computational complexity of the proposed GNN-AAA method is given by

$$\begin{aligned}
&\underbrace{\mathcal{O}\Big(BU\big(2K^2+4N_TK+4K\big)\Big)}_{\text{initial layer}} \\
&+\underbrace{\mathcal{O}\Big(QB^2U^2\big(2N_T+2K+1\big)+QBU\big(8K^2+7K\big)\Big)}_{\text{updating layers}} \\
&+\underbrace{\mathcal{O}\Big(BU\big((2N_T+2)K+(N_T+1)\big)+M(2K+1)\Big)}_{\text{output layer}}.
\end{aligned} \tag{4.13}$$

It is worthwhile to examine the complexity of the proposed CBA aggregation relative to the overall model, excluding the computations of aggregators and combiners, as summarized below:

$$\mathcal{O}\Big(QB^2U^2\big(2N_T+1\big)\Big). \tag{4.14}$$

Using the simulation parameters $N_T = 128$, $K = 512$, and $Q = 8$, the proportion of CBA's computational complexity order relative to the proposed GNN model is calculated as 0.21% for $B = 3, U = 6, M = 100$, 0.21% for $B = 3, U = 6, M = 200$, 0.45% for $B = 5, U = 8, M = 100$, and 0.45% for $B = 5, U = 8, M = 200$. These results indicate that the CBA aggregation adds only a modest increase to the overall computational complexity order, which is further corroborated by runtime measurements presented in the simulation section.

4.5 Simulations

4.5.1 Simulation Setups

We consider a simulation setup with multiple BSs operating at $f_c = 28$ GHz, placed along a semicircular arc of radius 120 m centered at $(0, 0, 0)$, where the RIS with $M = 100$ elements is also located. Each BS is equipped with $N_T = 128$ antennas. The Rayleigh distance is $d_{\text{Rayleigh}} = 86.4$ m. $B = 3$ BSs and $U = 10$ UEs are considered, with an equal split between near- and far-field UEs. Two UE distribution scenarios are considered. In Scenario 1 (Fig. 4.7 (a)), near-field UEs are randomly placed within a circle of radius equal to the Rayleigh distance around each BS, while far-field UEs are randomly placed within a semicircle of radius 20 m centered at $(0, 0, -20)$. In Scenario 2 (Fig. 4.7 (b)), UEs are fixed at specific locations designed to induce ESEs. For instance, if UE 1 and UE 6 are both served by BS 1, the far-field beam intended for UE 1 may spread energy toward UE 6. A similar case arises for other UEs sharing similar angular relationships with their serving BS. The mmWave channel comprises 5 clusters with 10 rays per cluster, and the path-loss model follows [71]: $-32.4 - 21\log_{10}(d) - 20\log_{10}(f_c)$ [dB]. Other parameters are: noise power -85 dBm, maximum transmit power $P_{\max} = 25$ dBm, maximum load $Z_{\max} = \lfloor U/B \rfloor$, and weighting factors $\lambda_1 = 2$ and $\lambda_2 = 3.5$.

For the GNN, the module dimensions are configured as $2N_T \times 1024 \times 512$ for $\Psi_U^{(0)}$, $512 \times 1024 \times 512$ for $\Psi_R^{(0)}$, $512 \times 512 \times 512$ for $\Psi_U^{\text{ag},(\tau)}$, $\Psi_U^{\text{cb},(\tau)}$, $\Psi_R^{\text{ag},(\tau)}$, and $\Psi_R^{\text{cb},(\tau)}$. The network includes $Q = 8$ update layers. Training uses 10^5 samples per epoch over 200 epochs with a batch size of 1024. The initial learning rate is 10^{-3}, decayed by a factor of

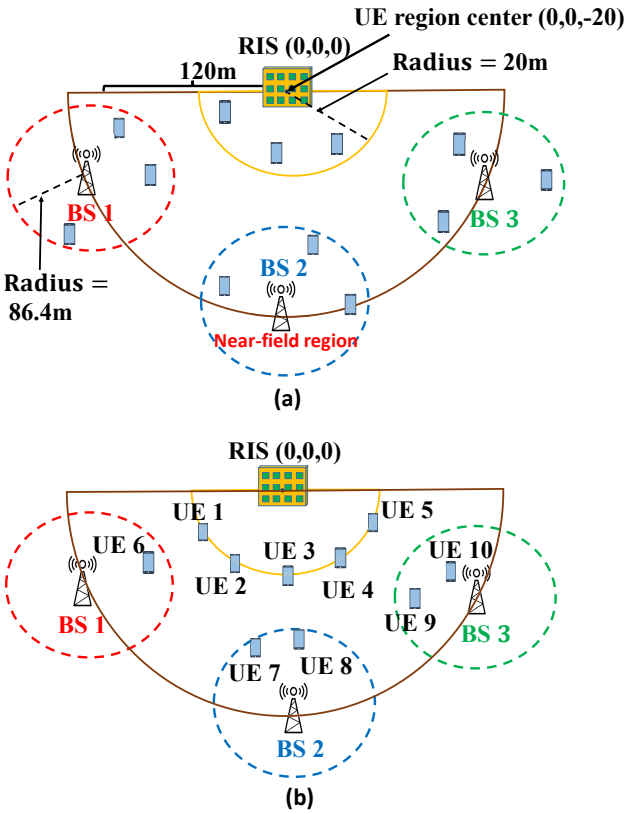


Figure 4.7: Simulation topologies. (a) Scenario 1 with randomly distributed UEs. (b) Scenario 2 with fixed UEs arranged to induce ESEs.

0.98 after 3 epochs. For evaluation, 10^4 testing samples are employed.

We evaluate the proposed GNN against the following benchmarks: 1) **GNN (no NAA):** The proposed GNN without NAA enhanced training, i.e., excluding the second penalty term in (4.12), 2) **GNN (no CBA and NAA):** The proposed GNN without both CBA aggregation and NAA-enhanced training, 3) **Model-based:** A joint optimization of user association, transmit beamformers, and RIS phase shifts following the model-based method in [70], and 4) **Codebook-based:** Transmit beamformers and RIS phase shifts are selected using a codebook-based method [36], while user association is determined via exhaustive search with the sum-rate criterion.

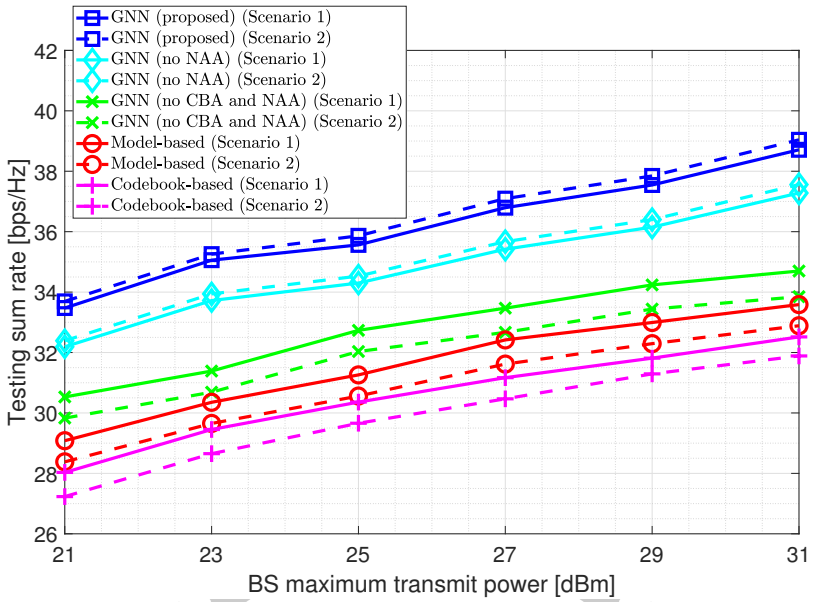


Figure 4.8: Testing sum rate vs. BS maximum transmit power $P_{\max}$ for Scenarios 1 and 2, with $B = 3$, $U = 10$, and $M = 100$.

4.5.2 Sum-rate Performance

Fig. 4.8 shows the testing sum rate vs. $P_{\max}$. For a fair comparison, all results are obtained under conditions where the load-balancing constraint in (4.3e) is satisfied almost 100%. The solid and dashed lines in Fig. 4.8 correspond to Scenario 1 and Scenario 2, respectively. In both scenarios, the proposed GNN outperforms all benchmarks, achieving the highest sum rate by effectively extracting channel and interference information directly from pilots. The model-based method yields suboptimal solutions due to relaxation, while the codebook-based approach is limited by predefined codewords and received signal power-based selection. Comparing the proposed GNN and GNN (no NAA) schemes, the inclusion of the second penalty in (4.12) sharpens near-field beamforming, boosting near-field UE power and the overall sum rate. Further removing CBA, as in the GNN (no CBA and NAA) scheme, leads to further degraded performance, confirming the effectiveness of the proposed attention mechanism.

4.5.3 User Association Examination

Comparing Scenario 1 and Scenario 2, the proposed GNN proves effective in mitigating the ESE, which is more pronounced in Scenario 2 due to the close proximity of near- and far-field UEs. By learning judicious user association decisions, the proposed GNN identifies and avoids associations that could trigger interference. As shown in a realistic snapshot for Scenario 2 in Fig. 4.9, the proposed GNN prevents UE 1 and UE 6 from being served by the same BS (Fig. 4.9(a)), unlike GNN (no CBA and NAA) (Fig. 4.9(b)), where multiple UE pairs (e.g., UEs 1 and 6, UEs 3 and 8, UEs 5 and 10) suffer from energy spread, and model-based and codebook-based methods (Fig. 4.9(c)), where the user association decisions are primarily dominated by the BS–UE distances. The enhanced ability to handle ESEs contributes to the higher average sum rates of the proposed GNN shown in Fig. 4.8.

Furthermore, in Fig. 4.8, all schemes except the proposed GNN achieve higher sum rates in Scenario 1 than in Scenario 2, due to stronger ESEs in Scenario 2. The proposed GNN, by effectively mitigating the ESE through improved user association decisions, achieves higher sum rates in Scenario 2 than in Scenario 1, demonstrating its effectiveness in mixed near- and far-field systems.

4.5.4 Variation of Mixed Near- and Far-Field Channel

Fig. 4.10 shows the testing sum rate vs. the near-field radius (10 m to the Rayleigh distance of 86.4 m) for Scenario 1 (Scenario 2 exhibits a similar trend). A smaller radius shortens the distances between near-field UEs and the BS, increasing near-field received power and reducing energy spread from far-field UEs, thus improving sum rates for all schemes. As the radius grows, the ESEs intensify, widening the performance gap between the proposed GNN and benchmarks, with the proposed GNN consistently outperforming benchmarks under varying degrees of energy spread conditions.

4.5.5 Complexity

Table 4.1 reports the average runtime required to obtain a feasible solution for one testing sample under different settings of BSs (B), UEs (U), and RIS elements (M). While incorporating CBA in *GNN (proposed)* results in a slight increase in runtime compared to *GNN*, it significantly enhances sum-rate performance, as demonstrated previously. The *Codebook-based* method has the highest runtime due to exhaustive search process, and the *Model-based* method has the second highest runtime due to its iterative optimization process. As B, U, or M increases, runtime increases for all schemes.

4.5.6 Generalizability

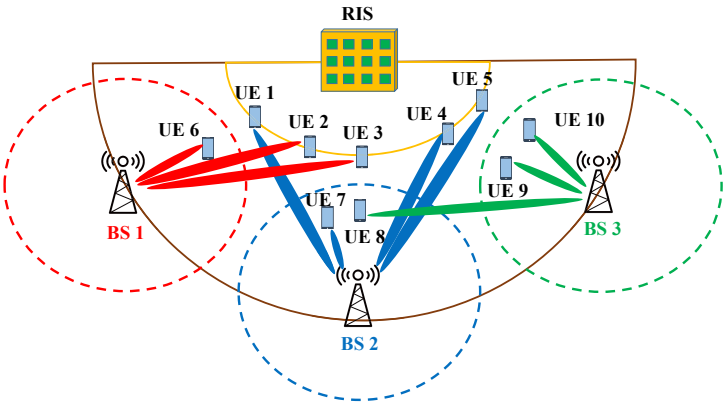
Table 4.2 demonstrates the generalizability of the proposed GNN, showing its effectiveness under a mismatch between training and testing UE numbers, with only minor sum-rate degradation. This shows a favorable property and adaptability of the proposed GNN to practical variations in UE numbers.

Table 4.1: Runtime (Unit: Millisecond)

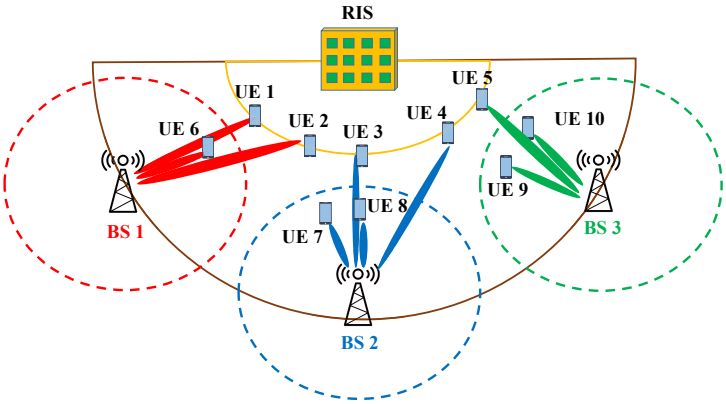
Scheme	$M = 100$	
	$B = 3, U = 6$	$B = 5, U = 8$
GNN (proposed)	3.03	3.42
GNN (no CBA and NAA)	3.01	3.40
Model-based	9.45×10^3	14.33×10^3
Codebook-based	10.11×10^3	15.02×10^3
Scheme	$M = 200$	
	$B = 3, U = 6$	$B = 5, U = 8$
GNN (proposed)	3.22	3.63
GNN (no CBA and NAA)	3.20	3.60
Model-based	9.88×10^3	14.83×10^3
Codebook-based	10.26×10^3	15.36×10^3

Table 4.2: Sum Rate for Various UE Numbers Used in Testing and Training

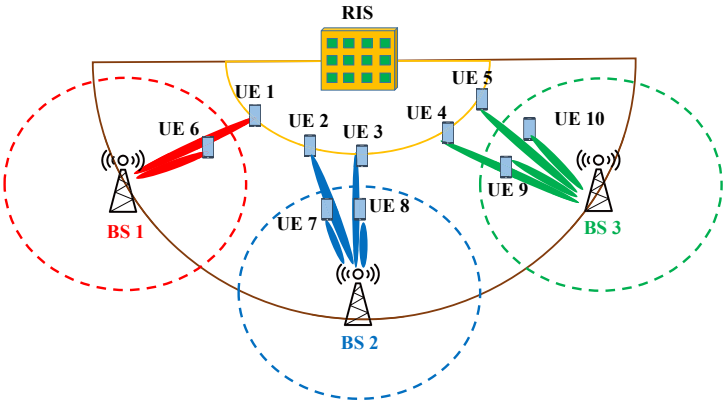
U (testing)	6	8	10	12	14
Sum rate (bps/Hz) U (training) $= 10$	29.54	32.05	35.36	37.43	38.96
Sum rate (bps/Hz) U (training) $= U$ (testing)	30.65	33.29	35.36	38.21	39.68



(a)



(b)



(c)

Figure 4.9: User association snapshots in Scenario 2 for (a) the proposed GNN, (b) GNN (no CBA and NAA), and (c) both model-based and codebook-based methods, with $B = 3$, $U = 10$, and $M = 100$. Note that the transmission links reflected by the RIS are omitted in this figure for clarity.

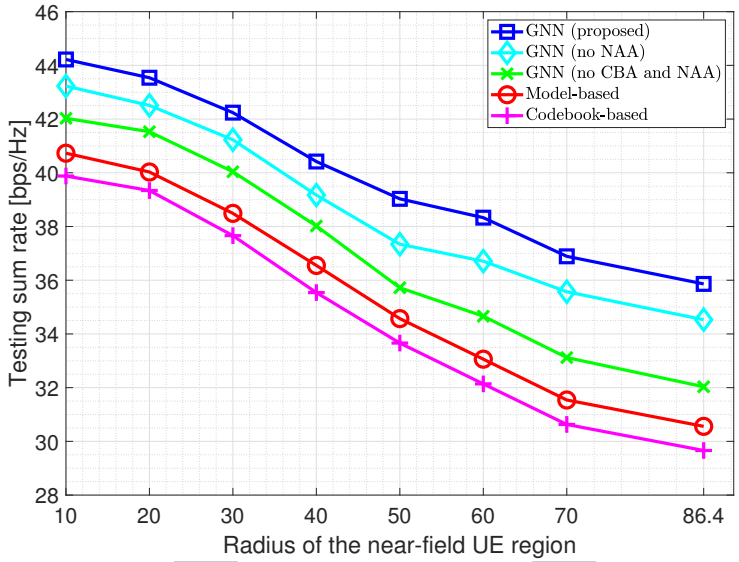


Figure 4.10: Testing sum rate vs. radius of the near-field region for Scenario 1, with $B = 3$, $U = 10$, and $M = 100$.

Chapter 5

Federated Graph Neural Network for Beamforming in Cell-free MIMO Systems

In this chapter, the proposed federated GNN-based approach for the joint design of beamforming in cell-free MIMO Systems is introduced. We investigate the combination of FL and GNN to improve efficiency on the required communication overhead for the beamforming in cell-free MIMO Systems. The background, system model, problem formulation, methodology, and simulation results are presented in sequence.

5.1 Background

5.1.1 Introduction to Federated Graph Neural Network(FedGNN)

GNNs, with their ability to extract features and learn from graph-structured data composed of nodes and edges, have been widely applied in various types of networks and systems, such as neuroscience, social networks, transportation networks, and wireless communication networks.

However, with the increasing awareness of data privacy in recent years, users are reluctant to upload their personal data directly to a centralized processing unit. Federated

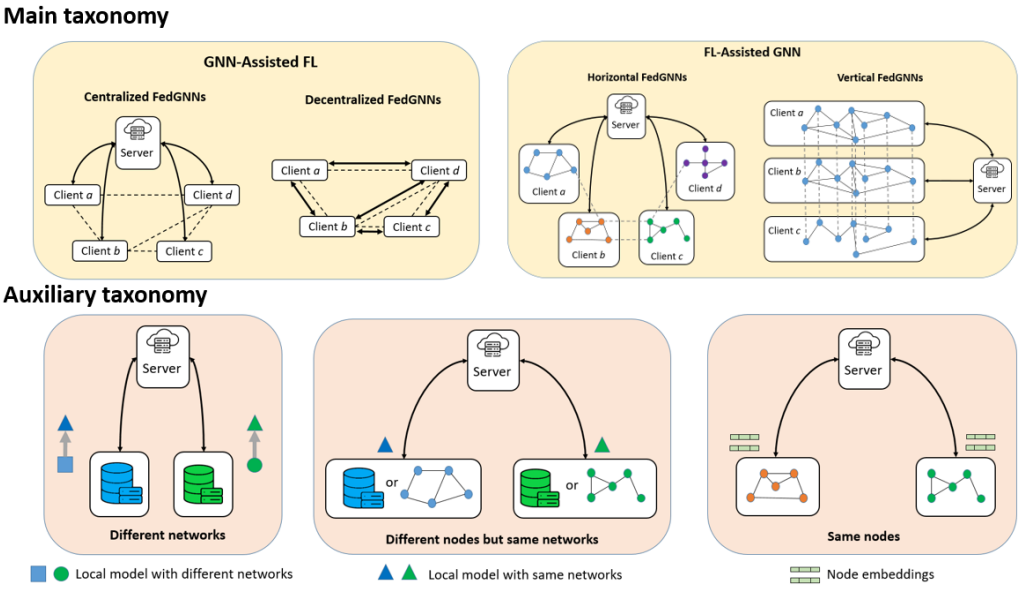


Figure 5.1: The classification of FedGNNs.

learning (FL) addresses this issue by distributing tasks to client devices, allowing them to upload only the learned feature embeddings or model parameters to the central server. This approach helps prevent data privacy leakage.

As a result, federated GNNs (FedGNNs), which integrate GNNs with FL, have emerged. These methods can handle graph-structured data while preserving data privacy. In [72], R. Liu et al. provide a comprehensive survey and analysis of existing FedGNN approaches. Based on how GNN and FL are integrated, FedGNNs can be broadly categorized into two types: GNN-assisted FL and FL-assisted GNNs, as illustrated in Fig. 5.1. Furthermore, according to the heterogeneity of FL clients, they can be divided into three categories: 1) Clients process graphs with the same set of nodes; 2) Clients process graphs with different nodes but share the same learning model architecture; 3) Clients adopt different learning model architectures.

Among the two main categories of FedGNNs, the first category, GNN-assisted FL, refers to scenarios where an existing network already has an FL task, and GNN is employed to learn from a graph constructed by treating clients as nodes in order to assist the FL task. Depending on whether a central processing unit is present, this category can be further divided into centralized FedGNNs and decentralized FedGNNs.

Correlations may exist among clients in the network. For example, traffic sensors located in nearby areas may observe similar traffic conditions. Therefore, clients can be represented as nodes to form a graph, and GNN can be used to learn the relationships among clients, thereby enhancing the FL task.

In centralized FedGNNs, the training of the GNN can be conducted either at the central server or at the clients. In the original FL framework, clients perform local training and upload their model parameters to the central server for aggregation, which produces a global model that is then sent back to the clients. This process is repeated iteratively.

If the GNN is trained at the central server, in addition to collecting local model parameters from clients, the server uses them to train the GNN, which then serves as the aggregation mechanism, replacing conventional weighted averaging. On the other hand, if the GNN is trained at the clients, it mainly addresses client heterogeneity. Since clients may have different data distributions or model architectures, they can share a global graph to obtain relational information among clients. The GNN then combines this relational information with local learning to produce feature embeddings, which are uploaded as part of the FL process.

The second category, FL-assisted GNNs, refers to scenarios where clients collaboratively train a GNN model while preserving data privacy. Each client performs local GNN training using its own data, and a central server aggregates these models via FL to obtain a global GNN model.

Based on how the data graphs are constructed across clients, FL-assisted GNNs can be further divided into horizontal FedGNNs and vertical FedGNNs. In horizontal FedGNNs, clients construct graphs with the same type of data but different nodes. In contrast, in vertical FedGNNs, clients construct graphs with different types of data but share the same nodes. For example, in horizontal FedGNNs, clients may handle users from different regions but all process height data, whereas in vertical FedGNNs, clients handle users from the same region, but some process height data while others process weight data.

5.1.2 Literature Review of Cell-free MIMO Beamforming

We summarize the current development of beamforming techniques for cell-free MIMO systems in the table below. A brief description of the comparison criteria listed in the table is provided as follows:

1. Centralized or Decentralized: This indicates whether the beamforming technique relies on a central processor. Centralized approaches require a central processor to generate the beamformers, whereas decentralized approaches do not.

2. Generalization Target: This refers to whether the beamforming technique can generalize with respect to specific system parameters. For example, GNN-based methods can generalize to different numbers of users, meaning that the model can adapt to varying numbers of UEs without retraining.

3. CSI Assumption: This distinguishes whether the beamforming technique requires channel state information (CSI) to be known in order to operate.

4. Optimization Objective Function: The objective functions generally include minimum mean-square error (MMSE), sum capacity (sum rate), energy efficiency, etc.

5. Proposed Method: The solutions can be broadly categorized into conventional optimization algorithms (e.g., zero-forcing, WMMSE) and machine learning/deep learning approaches, including DNN, DRL, and FL.

Regarding the references listed in the table, more detailed descriptions can be found in Chapter 1 under Related Works of the Third Research Part.

However, in the literature on beamforming for cell-free MIMO systems, there has been no discussion of approaches that simultaneously achieve decentralization and generalization capability. Therefore, in this part of the research, we aim to combine federated learning with GNNs to develop a decentralized beamforming method with generalization capability, with the objective of maximizing the sum capacity of the cell-free MIMO system.

Table 5.1: Summary of Related Works

Reference	Centralized or Decentralized	Generalization target	CSI assumption	Objective function Oversmoothing	Proposed Method
[43]	Centralized	×	known	Inter-user Interference Minimization	Zero-forcing
[44]	Centralized	×	known	Minimum Mean Square Error	WMMSE
[45]	Centralized	×	known	Sum Capacity	Majorization Minimization
[46]	Centralized	×	known	Minimum Mean Square Error	DNN
[47]	Centralized	×	known	Sum Capacity	DRL
[48]	Centralized	×	known	Energy efficiency	DRL
[49]	Centralized	Number of BSs and users	known	Minimum SINR Maximization	GNN
[50]	Decentralized	Number of users	known	Sum Capacity	GNN
[51]	Centralized	Number of users	known	Sum Capacity	GNN
[52]	Centralized	Number of users	known	Sum Capacity	GNN
[53]	Decentralized	×	known	Minimum Mean Square Error	bisection method and over-the-air signaling
[54]	Decentralized	×	known	Sum Capacity	DRL
[55]	Decentralized	×	known	Sum Capacity	DNN and FL
[56]	Decentralized	×	known	Sum Capacity	WMMSE, DNN and FL

5.2 System Model and Problem Description

Consider a cell-free multi-user wireless communication system, where B base stations (BSs) serve U single-antenna users. Each BS is equipped with N_T antennas. Let $\mathcal{B} = \{1, 2, \ldots, B\}$ denote the set of BSs, and $\mathcal{U} = \{1, 2, \ldots, U\}$ denote the set of users. The received signal at user u can be expressed as

$$\mathbf{y_u} = \sum_{b \in \mathcal{B}} \sum_{u' \in \mathcal{U}} \mathbf{h}_{b,u}^H \mathbf{f}_{b,u} \mathbf{s}_{u'} + n_u, \tag{5.1}$$

where $\mathbf{h}_{b,u}$ is the channel between BS b and user u, $\mathbf{s}_u$ is the transmitted symbol for user u with unit-power, $\mathbf{f}_{b,u}$ is the beamformer between BS b and user u, n_u is the additive white Gaussian noise (AWGN) with zero mean and variance σ_u^2.

The SINR of user u can be written as

$$\mathrm{SINR}_u = \frac{\left| \sum_{b \in \mathcal{B}} \mathbf{h}_{b,u}^H \mathbf{f}_{b,u} \right|^2}{\sum_{u' \in \mathcal{U} \setminus \{u\}} \left| \sum_{b \in \mathcal{B}} \mathbf{h}_{b,u'}^H \mathbf{f}_{b,u'} \right|^2 + \sigma_u^2}. \tag{5.2}$$

The sum rate of this system can be written as

$$R = \sum_{u \in \mathcal{U}} \log\left(1 + \mathrm{SINR}_u\right). \tag{5.3}$$

The problem of optimizing the beamforming vectors to maximize the system sum rate can be formulated as

$$\begin{aligned} &\max_{\mathbf{f}_{b,u}, b\in\mathcal{B}, u\in\mathcal{U}} R \\ &\text{s. t. } \sum_{u\in\mathcal{U}} \mathbf{f}_{b,u}^H \mathbf{f}_{b,u} \leq P_{b,\max}, \forall b \in \mathcal{B}, \end{aligned} \tag{5.4}$$

where $P_{b,\max}$ is the maximum transmit power of BS b.

Assume that each user transmits pilot signals to all BSs, which serve as local data for each BS. The received signal at BS b can be expressed as

$$\widehat{\boldsymbol{Y}}_b = \sum_{u\in\mathcal{U}} \mathbf{h}_{b,u}^* \mathbf{x}_u^H + \boldsymbol{N}_b \in \mathbb{C}^{N_T \times L}, \tag{5.5}$$

where $\mathbf{x}_u \in \mathbb{C}^{L\times 1}$ is the pilots transmitted by user u, L is the length of a pilot sequence, and $\boldsymbol{N}_b$ is the AWGN.

After decorrelation, the received signal at BS b can be obtained as

$$\widehat{\boldsymbol{y}}_{b,u} = \widehat{\boldsymbol{Y}}_b \mathbf{x}_u = \mathbf{h}_{b,u}^* + \boldsymbol{n}_{b,u} \in \mathbb{C}^{N_T \times 1}, \tag{5.6}$$

where $\widehat{\boldsymbol{y}}_{b,u}$ is taken as the local data of user u at BS b.

5.3 Methodology

As described in the previous subsection Introduction to FedGNN, we aim to design two schemes, GNN-assisted FL and FL-assisted GNN, to address the beamforming problem in cell-free MIMO systems.

5.3.1 Method 1: GNN-assisted FL for Beamforming in Cell-free MIMO Systems

In this approach, the objective of FL is to train a global model that is distributed to each base station (BS) acting as a client. Each BS utilizes the global model to generate the required beamforming vectors based on the received pilot signals.

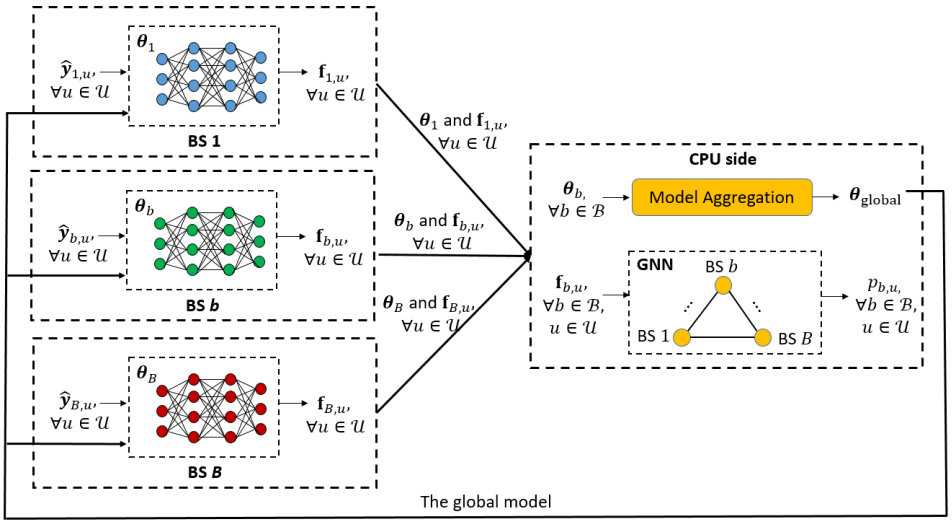


Figure 5.2: GNN-assisted FL for the beamforming problem in cell-free MIMO systems.

To achieve this goal, each BS performs local training independently, and model aggregation is conducted through a central processing unit. The learning model adopted for local training at each BS is a DNN composed of stacked fully connected layers.

In addition, to coordinate the beamforming strategies learned by different BSs and maximize the system sum rate, a GNN model is further trained at the central processing unit (CPU). This GNN learns the transmit power allocation for each BS, thereby mitigating inter-BS interference. The training procedure of this approach is illustrated in Fig. 5.2.

5.3.1.1 Workflow of Method 1

The workflow of the proposed method 1 is given as follows:

I. The CPU initialize the global model $\boldsymbol{\theta}_{\text{global}}$. $\boldsymbol{\theta}_{\text{global}}$ is sent to each BS to be the local DNN model θ_b, $\forall b \in \mathcal{B}$.

- $\boldsymbol{\theta}_{\text{global}}$: the set of the trainable parameters of the global model.
- $\boldsymbol{\theta}_b$: the set of the trainable parameters of the local DNN model at BS b.

II. Each BS uses the received pilot signals $\widehat{\boldsymbol{y}}_{b,u}$, $\forall b \in \mathcal{B}$, $u \in \mathcal{U}$ as local data and trains a local DNN model with the sum rate considering only intra-cell interference as

the training objective since the local data of each BS cannot be shared among all the BSs. After training, each BS transmits its local DNN model and the learned beamforming vectors $\mathbf{f}_{b,u},\ \forall b \in \mathcal{B},\ u \in \mathcal{U}$ to the central processing unit.

- The SINR and sum rate considering only intra-cell interference:

$$\mathrm{SINR}_{b,u} = \frac{\left|\mathbf{h}_{b,u}^H \mathbf{f}_{b,u}\right|^2}{\sum_{u' \in \mathcal{U} \setminus \{u\}} \left|\mathbf{h}_{b,u'}^H \mathbf{f}_{b,u'}\right|^2 + \sigma_u^2}. \tag{5.7}$$

$$R_b = \sum_{u \in \mathcal{U}} \log\left(1 + \mathrm{SINR}_{b,u}\right),\ \forall b \in \mathcal{B}. \tag{5.8}$$

- The loss function of each BS:

$$\mathcal{L}_b = -R_b,\ \forall b \in \mathcal{B}. \tag{5.9}$$

III. The CPU aggregates the local models from all BSs to obtain the global model. Meanwhile, it uses the beamforming vectors learned by each BS as inputs to train the GNN, which learns the transmit power allocation for each BS, with the objective of maximizing the sum rate.

- Model aggregation:

$$\boldsymbol{\theta}_{\text{global}} = \sum_{b \in \mathcal{B}} = \frac{D_b}{D_{\text{total}}} \boldsymbol{\theta}_b, \tag{5.10}$$

  where D_b is the amount of the local data at BS b, and D_{total} is the total amount of the local data at all the BSs.

- Consider the SINR and sum rate under transmit power control at each base station, $p_{b,u},\ \forall b \in \mathcal{B},\ u \in \mathcal{U}$.

$$\mathrm{SINR}_u^{'} = \frac{\left|\sum_{b \in \mathcal{B}} p_{b,u} \mathbf{h}_{b,u}^H \mathbf{f}_{b,u}\right|^2}{\sum_{u' \in \mathcal{U} \setminus \{u\}} \left|\mathbf{h}_{b,u'}^H \mathbf{f}_{b,u'}\right|^2 + \sigma_u^2}. \tag{5.11}$$

$$R^{'} = \sum_{u \in \mathcal{U}} \log\left(1 + \mathrm{SINR}_u^{'}\right), \tag{5.12}$$

  where the transmit power must satisfy $\sum_{u \in \mathcal{U}} p_{b,u} \leq P_{b,\max},\ \forall b \in \mathcal{B}$.

- The loss function of the GNN at the CPU:

$$\mathcal{L}_{\text{CPU}} = -R'. \tag{5.13}$$

IV. The CPU sends the global model back to all BSs.

Steps II–IV are iterated until convergence.

5.3.1.2 GNN Structure of Method 1

The GNN architecture at the CPU is shown in Fig.5.3, consisting of an initial layer, several updating layers, and an output layer. In this GNN, node b corresponds to BS b. The input to node b is formed by concatenating the beamforming vectors of BS b, $\widehat{\boldsymbol{z}}_b = [\mathbf{f}_{b,1}, \mathbf{f}_{b,2}, \ldots, \mathbf{f}_{b,U}]$, while the output of node b, $\boldsymbol{p}_b = [p_{b,1}, p_{b,2}, \ldots, p_{b,U}]$, represents the transmit power from BS b to all users.

In the updating layers, the information of node b is updated as follows: the information from all other BSs is first processed by a neural network (NN) module, and then aggregated using an element-wise max operation. The aggregated information is subsequently combined with the information from node b in the previous layer through another NN module to produce the updated representation of node b at the current layer. After passing through several updating layers, the updated representation of node b is fed into a linear layer in the output layer to obtain the transmit power from BS b to all users.

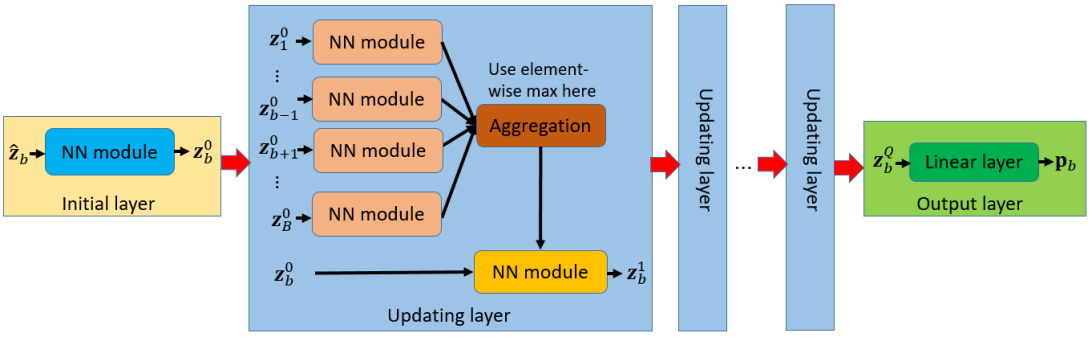


Figure 5.3: The structure of the GNN at the CPU for the GNN-assisted FL method.

5.3.1.3 The Advantages and Disadvantages of Method 1

The advantages and disadvantages of the proposed method 1 are as follows:

Advantages:

- It can achieve generalization with respect to the number of BSs.
- Centralized coordination helps mitigate inter-BS interference, eliminating the need for information sharing among BSs.
- Compared with centralized beamforming design, it significantly reduces communication overhead.

Disadvantages: In addition to local training at the BSs, the convergence of the GNN training must also be considered, which may result in a longer time to complete the FL process.

5.3.2 Method 2: FL-assisted GNN for Beamforming in Cell-free MIMO Systems

The objective of this method is to train a global GNN model, which each BS uses to generate the required beamforming vectors based on the received pilot signals.

In this cell-free multi-user system, each BS serves all users. To enable generalization with respect to the number of users, users are treated as nodes to construct a graph. Based on this graph, each BS locally trains a GNN model, and the CPU collects the training results from all BSs to perform FL aggregation.

To maximize the system sum rate, the CPU gathers the beamforming vectors generated by each BS to compute the sum rate, and then feeds this result back to each BS as the loss function for local training. The training procedure of this method is illustrated in Fig. 5.4.

5.3.2.1 Workflow of Method 2

The workflow of the proposed method 2 is given as follows:

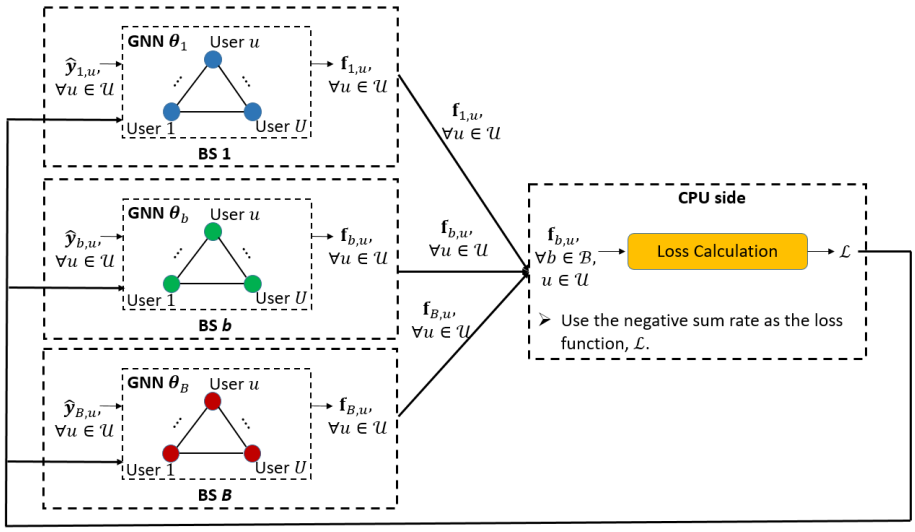


Figure 5.4: FL-assisted GNN for the beamforming problem in cell-free MIMO systems.

I. Each BS initialize the local GNN model θ_b, $\forall b \in \mathcal{B}$.

- $\boldsymbol{\theta}_b$: the set of the trainable parameters of the local GNN model at BS b.

II. Each BS uses the received pilot signals $\widehat{\boldsymbol{y}}_{b,u}$, $\forall b \in \mathcal{B}$, $u \in \mathcal{U}$ as local data and generate the initial beamforming vectors $\mathbf{f}_{b,u}$, $\forall b \in \mathcal{B}$, $u \in \mathcal{U}$ based on the initial local GNN model.

III. Each BS transmits its initial beamforming vectors to the CPU. CPU calculate the initial sum rate based on the initial beamforming vectors from all the BSs, and then the initial sum rate is sent back to each BS.

- The SINR of user u and the sum rate:

$$\mathrm{SINR}_u = \frac{\left|\sum_{b\in\mathcal{B}} \mathbf{h}_{b,u}^H \mathbf{f}_{b,u}\right|^2}{\sum_{u'\in\mathcal{U}\setminus\{u\}} \left|\sum_{b\in\mathcal{B}} \mathbf{h}_{b,u'}^H \mathbf{f}_{b,u'}\right|^2 + \sigma_u^2}. \tag{5.14}$$

$$R = \sum_{u\in\mathcal{U}} \log\left(1 + \mathrm{SINR}_u\right). \tag{5.15}$$

IV. Each BS uses the sum rate computed by the CPU as the loss function for training its local GNN model. After training, each BS transmits the beamforming vectors learned by its local GNN model to the CPU.

- The negative of the received sum rate is used as the local loss function for each base station, i.e., $\mathcal{L} = -R$.

V. The CPU computes the sum rate based on the beamforming vectors learned by all BSs and feeds the result back to each BS.

Steps IV and V are iterated until convergence.

5.3.2.2 GNN Structure of Method 2

The local GNN architecture at each BS is illustrated in Fig. 5.5, which shows the GNN at BS b. All BSs are assumed to adopt the same GNN architecture, consisting of an initial layer, several updating layers, and an output layer. In the GNN at BS b, node u corresponds to user u. The input to node u is $\widehat{\boldsymbol{y}}_{b,u}$, which represents the decorrelated vector of the pilot signal transmitted from user u and received at BS b. The output of node u is $\mathbf{f}_{b,u}$.

In the updating layers of this GNN, the information of node u is updated as follows: the information from all other users is first processed by a NN module, and then aggregated using an element-wise max operation. The aggregated information is combined with the information of node u from the previous layer through another NN module to produce the updated representation of node u at the current layer. After passing through several updating layers, the updated representation of node u is fed into a linear layer in the output layer to obtain the beamforming vector $\mathbf{f}_{b,u}$ from BS b to user u.

5.3.2.3 The Advantages and Disadvantages of Method 2

The advantages and disadvantages of the proposed method 2 are as follows:

Advantages:

- It can achieve generalization with respect to the number of users.
- The computational burden on the central processing unit is reduced.
- Compared with centralized beamforming design, it significantly reduces communication overhead.

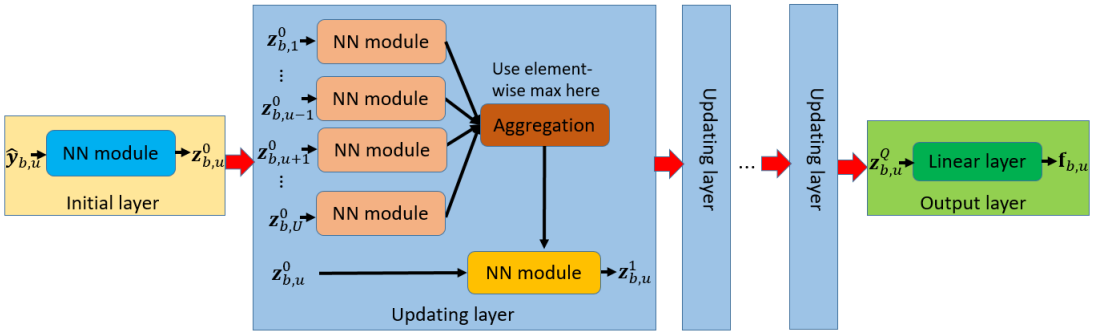


Figure 5.5: The structure of the GNN at each BS for the FL-assisted GNN method.

Disadvantages: Due to the lack of information sharing among base stations, the mitigation of inter-BS interference relies on multiple iterative steps.

5.4 Simulations

5.4.1 Simulation Setups

The following presents the preliminary simulation results based on the aforementioned FL-assisted GNN approach (as illustrated in Fig. 5.2). In the simulation figures, the proposed method is denoted as FL–GNN. The benchmark used for comparison is the centralized WMMSE method [44].

In the simulation setup, multiple BSs are placed along a semicircular arc of radius 100 m centered at $(0, 0, 0)$, and the UEs are randomly placed within a semicircle of radius 20 m centered at $(0, 0, -20)$ as shown in Fig. 5.6. The number of BSs is $B = 4$, the number of users is $U = 4$, and each BS is equipped with $N_T = 16$ antennas. The channels between the BSs and the UEs are modeled by Rayleigh fading channels. The noise power is set to -90 dBm, and the maximum transmit power of each BS ranges from -10 dBm to 10 dBm. The path loss model is given by $\text{Path loss (dB)} = -30.5 - 36.7 \log_{10}(d_{b,u})$ where $d_{b,u}$ denotes the distance between BS b and user u.

For the FL-assisted GNN configuration, all BSs adopt local GNN models with identical architectures and dataset sizes. The number of updating layers in each local GNN

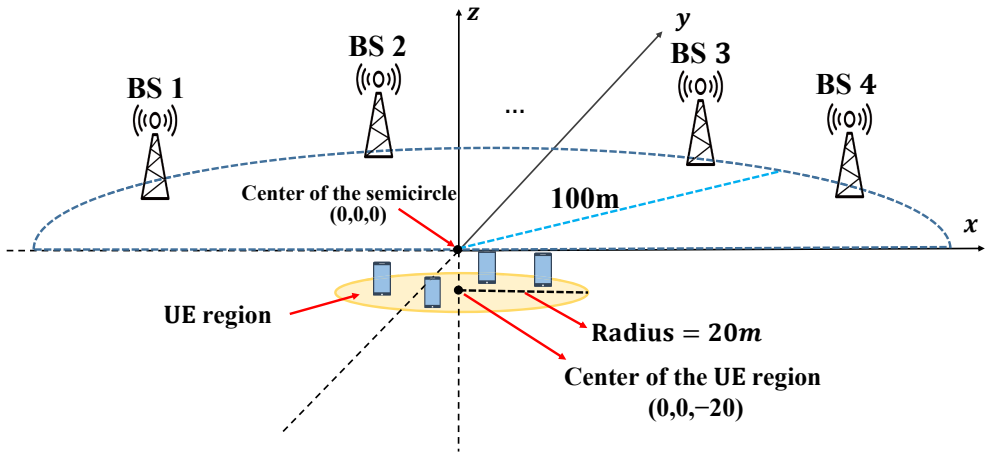


Figure 5.6: Simulation topologies.

model is set as $Q = 4$. Each local GNN model consists of two updating layers, and the neural network module is configured as $2M \times 128 \times 128$ (initial layer) and $128 \times 128 \times 128$ (updating layer, aggregation, and combination). Each BS performs local training for 200 epochs, with 100,000 training samples and 10,000 testing samples. The learning rate is set to 0.001, and the number of FL communication rounds is 20.

We denote the proposed FL-assisted GNN as **FL–GNN [proposed]** and evaluate it against the following benchmarks: 1) **WMMSE**: A centralized method with WMMSE [44] conducts the optimization of all the beamformers, 2) **GNN**: A centralized method with a GNN model optimizes all the beamformers with the same model size as the proposed FL-assisted GNN. The loss function is set by the system sum-rate objective, 3) **DNN:** A centralized method with a conventional DNN model optimizes all the beamformers with the same model size as the proposed FL-assisted GNN. The loss function is set by the system sum-rate objective.

5.4.2 Simulation Results

Fig. 5.7 shows the test sum rate versus the maximum transmit power of the base stations, while Fig. 5.8 illustrates the convergence behavior of the four base stations during the training process in this simulation. From Fig. 5.7, it can be observed that FL–GNN

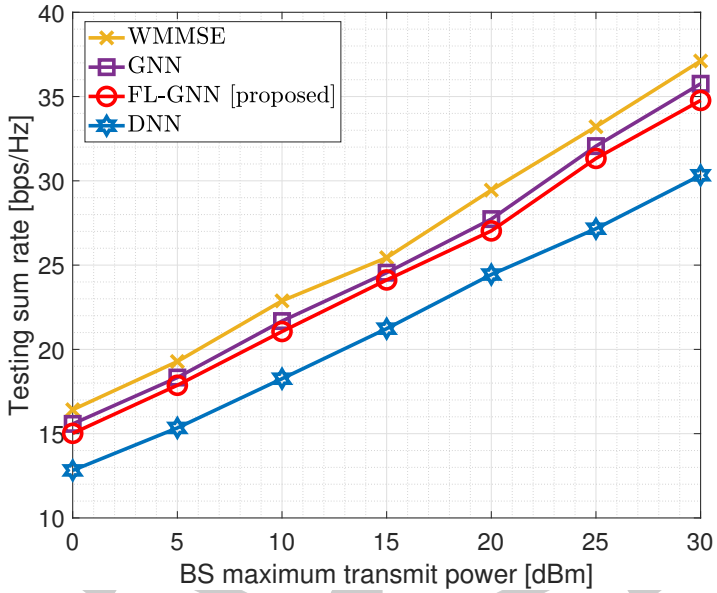


Figure 5.7: The testing sum rate vs. the maximum BS transmit power.

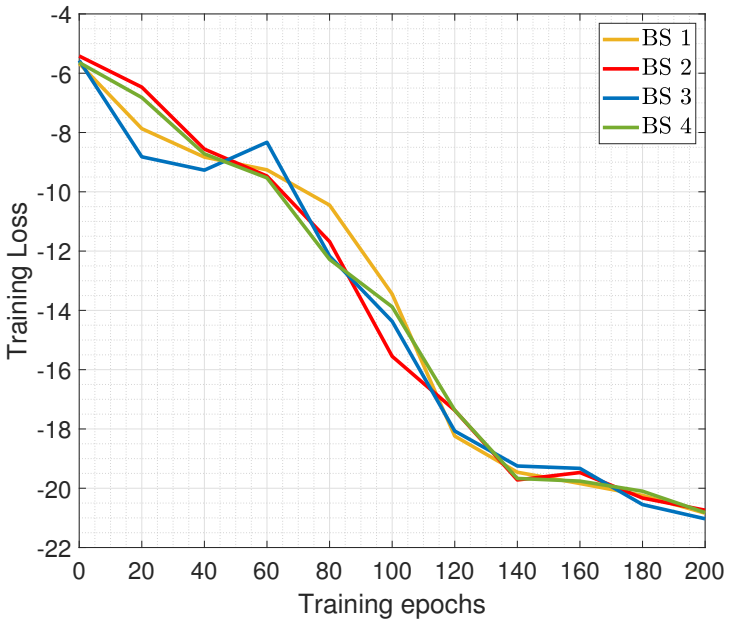


Figure 5.8: The convergence results of each BS.

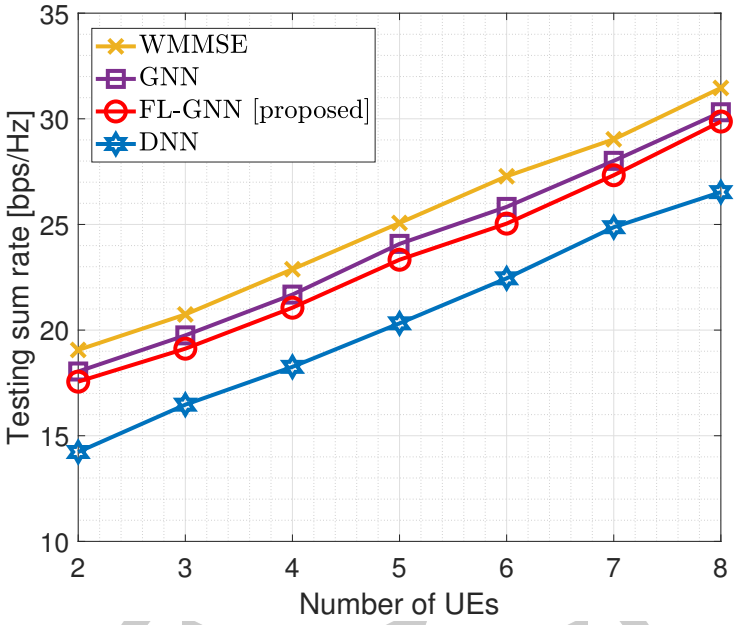


Figure 5.9: The testing sum rate vs. the number of UEs.

[proposed] achieves sum-rate performance comparable to that of the two centralized approaches, GNN and WMMSE. Moreover, FL–GNN [proposed] exhibits a clear performance advantage over DNN. This performance gain stems from employing a GNN as the local model at each BS, enabling the model to effectively capture the complex interference relationships among multiple UEs and BSs and thereby generate beamformers that better mitigate interference. In addition, the FL framework allows all local models at the BSs to be collaboratively optimized toward the system-wide sum-rate objective without requiring frequent information exchange among the BSs.

Fig. 5.9 illustrates the test sum rate versus the number of UEs. A similar performance trend can be observed in Fig. 5.9. FL–GNN [proposed] consistently achieves sum-rate performance comparable to that of the centralized GNN and WMMSE approaches under different numbers of UEs. These results demonstrate that FL–GNN [proposed] can effectively adapt to environments with varying numbers of users while maintaining competitive sum-rate performance.

Moreover, the proposed FL–GNN can significantly reduce the communication cost required for generating beamforming vectors during practical operation, since it elimi-

nates the need to transmit all information to the central processing unit. Instead, each base station can directly generate the beamforming vectors using only the received pilot signals from users. Therefore, approximately 40% of the data transmission overhead can be reduced during practical deployment when generating beamformers for a system with 4 BSs, 4 users, and 8 antennas at each BS.

Table 5.2: Sum rate for various U's used in testing and training

U (testing)	2	3	4	5	6	7	8
Sum rate (bps/Hz) U(training) $= 4$	16.898	18.476	21.060	22.917	24.533	26.876	29.242
Sum rate (bps/Hz) U (training) $= U$ (testing)	17.562	19.123	21.060	23.335	25.045	27.334	29.875

Finally, to examine the generalizability of the proposed FL–GNN, Table 5.2 shows the sum rate differences between the case when the numbers of users, respectively used in training and testing are different and the case when they are identical. The performance degradation due to mismatches in parameters for training and testing is not significant. These results demonstrate that the proposed FL–GNN, despite being optimized in a decentralized manner, is able to maintain strong generalization capability.

Chapter 6

Conclusion and Future Work

In this chapter, we provide conclusions for the three research parts introduced in this work, and we further discuss future work based on the current research findings.

6.1 Conclusion

In the first research part, we have proposed a GNN-AAA approach for the joint optimization of UA, BS transmit beamforming, and RIS phase shifts in RIS-aided multi-BS multi-UE mmWave networks. The proposed GNN-AAA demonstrated enhanced generalization capabilities across both numbers of BSs and UEs compared to traditional GNNs, exhibiting only minor performance degradation in mismatched training and testing scenarios. This generalizability highlights the practicality of the proposed method for dynamic and scalable network environments. The AAA mechanism integrated into GNN-AAA showed quantifiable effectiveness in mitigating oversmoothing in deep GNNs, as reflected by improvements in the squared norm of node differences, while maintaining adequate selective information exchange to ensure superior sum-rate performance, with only a modest increase in computational complexity. Furthermore, the proposed GNN-AAA effectively bypassed the need for explicit channel estimation, addressing a critical bottleneck in large-RIS-aided systems. Simulation results validated that GNN-AAA achieved higher sum-rate performance compared to model-based method, HGAT, DNN, and conventional UA strategies from various perspectives, demonstrated GNN-AAA's robustness

against pilot correlation, and highlighted the load-balancing benefits of deploying RISs.

In the second research part, a GNN-based approach has been proposed to jointly optimize user association, BS beamforming, and RIS phase shifts in RIS-aided multi-BS multi-UE XL-MIMO systems. Mechanisms specifically designed to handle ESEs in near- and far-field coexisting systems, namely, CBA aggregation and NAA-enhanced training, were incorporated into the framework. Simulations showed that the proposed method outperforms model- and codebook-based schemes in sum rate with a lower computational complexity, effectively addresses ESEs, and generalizes well to varying numbers of UEs during inference.

In the third research part, we have proposed a method that combines FL and GNN to design beamforming for cell-free MIMO systems. This approach integrated the advantages of both FL and GNN: FL reduces the communication overhead required for optimization, while GNN enables generalization across different numbers of users. For future wireless communication systems, this can effectively reduce communication latency and accommodate variations in the number of users in the environment. Preliminary simulation results showed that, compared with centralized beamforming optimization methods, the proposed FL-assisted GNN approach achieved comparable sum-rate performance with significantly reduced communication overhead.

6.2 Future Work

For the GNN-based method for RIS-assisted systems, future research directions include exploring alternative graph representations to model complex interactions and extending the proposed framework to scenarios involving multiple RISs. These advancements could further enhance the flexibility, scalability, and performance of GNN-AAA in more general and complex network topologies.

On the other hand, for FedGNN-based beamforming design in cell-free systems, future research directions include attempting to reduce the reliance of the central processing unit (CPU) on CSI during the training of the proposed two methods, namely FL-assisted GNN and GNN-assisted FL. This requires further investigation into how each client ob-

tains its data within the FL framework and how information can be shared without significantly increasing communication overhead, in order to facilitate overall system optimization. Moreover, future work may extend FedGNN beyond beamforming tasks to include additional functionalities, such as user association and RIS beamforming design.

Bibliography

[1] Y. Shen, Y. Shi, J. Zhang, and K. B. Letaief, "Graph neural networks for scalable radio resource management: Architecture design and theoretical analysis," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 1, pp. 101–115, Jan. 2021.

[2] Q. Wu and R. Zhang, "Intelligent reflecting surface enhanced wireless network via joint active and passive beamforming," *IEEE Trans. Wireless Commun.*, vol. 18, no. 11, pp. 5394–5409, Nov. 2019.

[3] C. Huang, A. Zappone, G. C. Alexandropoulos, M. Debbah, and C. Yuen, "Reconfigurable intelligent surfaces for energy efficiency in wireless communication," *IEEE Trans. Wireless Commun.*, vol. 18, no. 8, pp. 4157–4170, Aug. 2019.

[4] C. Pan *et al.*, "Multicell MIMO communications relying on intelligent reflecting surfaces," *IEEE Trans. Wireless Commun.*, vol. 19, no. 8, pp. 5218–5233, Aug. 2020.

[5] C. Huang, G. C. Alexandropoulos, C. Yuen, and M. Debbah, "Indoor signal focusing with deep learning designed reconfigurable intelligent surfaces," in *Proc. IEEE 20th Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, Cannes, France, Jul. 2019, pp. 1–5.

[6] B. Sheen, J. Yang, X. Feng, and M. M. U. Chowdhury, "A deep learning based modeling of reconfigurable intelligent surface assisted wireless communications for phase shift configuration," *IEEE Open J. Commun. Soc.*, vol. 2, pp. 262–272, Jan. 2021.

[7] R. Zhang *et al.*, "Energy efficiency maximization in RIS-assisted SWIPT networks with RSMA: A PPO-based approach," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 5, pp. 1413–1430, May 2023.

[8] T. Jiang, H. V. Cheng, and W. Yu, "Learning to reflect and to beamform for intelligent reflecting surface with implicit channel estimation," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 7, pp. 1931–1945, Jul. 2021.

[9] B.-J. Chen, R. Y. Chang, F.-T. Chien, and H. V. Poor, "Graph neural network-based joint beamforming for hybrid relay and reconfigurable intelligent surface aided multiuser systems," *IEEE Wireless Commun. Lett.*, vol. 12, no. 10, pp. 1811–1815, Oct. 2023.

[10] W. Jin, J. Zhang, C.-K. Wen, S. Jin, X. Li, and S. Han, "Low-complexity joint beamforming for RIS-assisted MU-MISO systems based on model-driven deep learning," *IEEE Trans. Wireless Commun.*, vol. 23, no. 7, pp. 6968–6982, July 2024.

[11] T.-J. Yeh, W.-C. Tsai, C.-W. Chen, and A.-Y. Wu, "Enhanced-GNN with angular CSI for beamforming design in IRS-assisted mmwave communication systems," *IEEE Commun. Lett.*, vol. 28, no. 4, pp. 827–831, Apr. 2024.

[12] Y. Li, Y. Lu, R. Zhang, B. Ai, and Z. Zhong, "Deep learning for energy efficient beamforming in MU-MISO networks: A GAT-based approach," *IEEE Wireless Commun. Lett.*, vol. 12, no. 7, pp. 1264–1268, July 2023.

[13] Y. Li, Y. Lu, B. Ai, O. A. Dobre, Z. Ding, and D. Niyato, "GNN-based beamforming for sum-rate maximization in MU-MISO networks," *IEEE Trans. Wireless Commun.*, vol. 23, no. 8, pp. 9251–9264, Aug. 2024.

[14] M. Randall, P. Belzarena, F. Larroca, and P. Casas, "Deep reinforcement learning and graph neural networks for efficient resource allocation in 5G networks," in *Proc. IEEE Latin-American Conf. Commun. (LATINCOM)*, Rio de Janeiro, Brazil, Nov. 2022, pp. 1–6.

[15] H. Zhang, X. Ma, X. Liu, L. Li, and K. Sun, "GNN-based power allocation and user association in digital twin network for the terahertz band," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 10, pp. 3111–3121, Oct. 2023.

[16] V. Ranasinghe, N. Rajatheva, and M. Latva-aho, "Graph neural network based access point selection for cell-free massive MIMO systems," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Madrid, Spain, Dec. 2021, pp. 1–6.

[17] Q. Li, Z. Han, and X.-M. Wu, "Deeper insights into graph convolutional networks for semi-supervised learning," in *Proc. AAAI*, vol. 32, pp. 3539–3544. IEEE, Apr. 2018.

[18] B. Lu, X. Gan, H. Jin, L. Fu, X. Wang, and H. Zhang, "Make more connections: Urban traffic flow forecasting with spatiotemporal adaptive gated graph convolution network," *ACM Trans. Intell. Syst. Technol.*, vol. 13, no. 2, pp. 1–25, Apr. 2022.

[19] Y. Tian, G. Liu, J. Wang, and M. Zhou, "ASA-GNN: Adaptive sampling and aggregation-based graph neural network for transaction fraud detection," *IEEE Trans. Comput. Soc. Syst.*, vol. 11, no. 3, pp. 3536–3549, June 2024.

[20] K.-L. Chan, F.-T. Chien, and R. Y. Chang, "Beamforming and load-balanced user association in RIS-aided systems via graph neural networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Denver, CO, USA, Jun. 2024, pp. 4997–5002, IEEE.

[21] K. L. Chan, R. Y. Chang, F. T. Chien, and H. V. Poor, "Beamforming and load-balanced user association in RIS-aided mmwave systems via adaptive attention graph neural networks," *IEEE Trans. Wireless Commun.*, vol. 25, pp. 5781–5796, 2026.

[22] H. Lu and Y. Zeng, "Communicating with extremely large-scale array/surface: Unified modelling and performance analysis," *IEEE Trans. Wireless Commun.*, vol. 21, no. 6, pp. 4039–4053, Jan. 2022.

[23] H. Lu *et al.*, "A tutorial on near-field XL-MIMO communications toward 6G," *IEEE Commun. Surv. Tutor.*, vol. 26, no. 4, pp. 2213–2257, 2024.

[24] K.-L. Chan, R. Y. Chang, and F.-T. Chien, "Graph neural network-based beamforming and user association for mixed near- and far-field RIS-aided mmwave systems," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Glasgow, Scotland, UK, May 2026, IEEE.

[25] W. Mei and R. Zhang, "Performance analysis and user association optimization for wireless network aided by multiple intelligent reflecting surfaces," *IEEE Trans. Wireless Commun.*, vol. 69, no. 9, pp. 6296–6312, Sep. 2021.

[26] D. Zhao, H. Lu, Y. Wang, H. Sun, and Y. Gui, "Joint power allocation and user association optimization for IRS-assisted mmWave systems," *IEEE Trans. Wireless Commun.*, vol. 21, no. 1, pp. 577–590, Jan. 2022.

[27] E. M. Taghavi, R. Hashemi, A. Alizadeh, N. Rajatheva, M. Vu, and M. Latva-aho, "Joint active-passive beamforming and user association in IRS-assisted mmWave cellular networks," *IEEE Trans. Veh. Technol.*, vol. 72, no. 8, pp. 10448–10461, Aug. 2023.

[28] H. Amiriara, F. Ashtiani, M. Mirmohseni, and M. Nasiri-Kenari, "IRS-user association in IRS-aided MISO wireless networks: Convex optimization and machine learning approaches," *IEEE Trans. Veh. Technol.*, vol. 72, no. 11, pp. 14305–14316, Nov. 2023.

[29] P. S. Aung, Y. Kyaw Tun, Z. Han, and C. S. Hong, "Energy-efficiency maximization of multiple RISs-enabled communication networks by deep reinforcement learning," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Seoul, Korea, Republic of, May 2022, pp. 2181–2186.

[30] Z. Wang, Y. Wei, Z. Feng, F. R. Yu, and Z. Han, "Resource management and reflection optimization for intelligent reflecting surface assisted multi-access edge computing using deep reinforcement learning," *IEEE Trans. Wireless Commun.*, vol. 22, no. 2, pp. 1175–1186, Feb. 2023.

[31] Y. Zhu, M. Li, Y. Liu, Q. Liu, Z. Chang, and Y. Hu, "DRL-based joint beamforming and BS-RIS-UE association design for RIS-assisted mmWave networks," in *Proc.*

IEEE Wireless Commun. and Net. Conf. (WCNC), Austin, TX, USA, Apr. 2022, pp. 345–350.

[32] H. Zhang, N. Shlezinger, F. Guidi, D. Dardari, M. F. Imani, and Y. C. Eldar, "Beam focusing for near-field multiuser MIMO communications," *IEEE Trans. Wireless Commun.*, vol. 21, no. 9, pp. 7476–7490, Sep. 2022.

[33] X. Cao, M. Mohammadi, H. Q. Ngo, and M. Matthaiou, "Distance-aware precoding for near-field capacity improvement in XL-MIMO," in *Proc. IEEE 95th Veh. Technol. Conf. (VTC)*, Helsinki, Finland, 2022, pp. 1–5.

[34] S. Shan and Y. Li, "Distance-aware precoding for near-field capacity improvement in XL-MIMO," in *Proc. IEEE Wireless Commun. and Net. Conf. (WCNC)*, Milan, Italy, 2025, pp. 1–6.

[35] Z. Wu and L. Dai, "Multiple access for near-field communications: SDMA or LDMA?," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 6, pp. 1918–1935, Jun. 2023.

[36] X. Wei, L. Dai, Y. Zhao, G. Yu, and X. Duan, "Codebook design and beam training for extremely large-scale RIS: Far-field or near-field?," *China Commun.*, vol. 19, no. 6, pp. 193–204, Jun. 2022.

[37] Y. Zhang, C. You, L. Chen, and B. Zheng, "Mixed near- and far-field communications for extremely large-scale array: An interference perspective," *IEEE Commun. Letters*, vol. 27, no. 9, pp. 2496–2500, Sep. 2023.

[38] X. Cao, M. Mohammadi, H. Q. Ngo, and M. Matthaiou, "RIS-assisted XL-MIMO for coexistence of near-field and far-field communications," in *Proc. IEEE Wireless Commun. and Net. Conf. (WCNC)*, Dubai, United Arab Emirates, 2024, pp. 1–6.

[39] Y. Zhang, Z. Yang, S. Yue, L. Liu, and B. Di, "Hybrid near-field and far-field XL-MIMO: How many users can be supported?," *IEEE Commun. Letters*, vol. 28, no. 10, pp. 2402–2406, Oct. 2024.

[40] T. Wang, C. You, F. Zhou, and C. Yin, "Base station beamforming design in near-field XL-IRS beam training," *IEEE Commun. Letters*, vol. 28, no. 4, pp. 932–936, Apr. 2024.

[41] W. D. S. Junior, D. W. M. Guerra, J. C. M. Filho, T. Abrão, and E. Hossain, "Manifold-based optimizations for RIS-aided massive MIMO systems," *IEEE Open J. Commun. Soc.*, vol. 5, pp. 7913–7940, 2024.

[42] Y. Zhang *et al.*, "Joint optimization design for near-field IRS-assisted modular XL-MIMO systems," *IEEE Trans. Veh. Technol.*, Jun. 2025.

[43] M. Joham, W. Utschick, and J. A. Nossek, "Linear transmit processing in MIMO communications systems," *IEEE Trans. on Signal Processing*, vol. 53, no. 8, pp. 2700–2712, Aug. 2005.

[44] S. S. Christensen *et al.*, "Weighted sum-rate maximization using weighted MMSE for mimo-bc beamforming design," *IEEE Trans. Wireless Commun.*, vol. 7, no. 12, pp. 4792–4799, Dec. 2008.

[45] S. H. Park *et al.*, "Joint precoding and multivariate backhaul compression for the downlink of cloud radio access networks," *IEEE Trans. on Signal Processing*, vol. 61, no. 22, pp. 5646–5658, Nov. 2013.

[46] M. Zaher *et al.*, "Learning-based downlink power allocation in cell-free massive MIMO systems," *IEEE Trans. Wireless Commun.*, vol. 22, no. 1, pp. 174–188, Jan. 2023.

[47] Y. Al-Eryani, M. Akrout, and E. Hossain, "Multiple access in cell-free networks: Outage performance, dynamic clustering, and deep reinforcement learning-based design," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 4, pp. 1028–1042, Apr. 2021.

[48] W. Li *et al.*, "Deep reinforcement learning for energy-efficient beamforming design in cell-free networks," in *Proc. IEEE Wireless Commun. and Net. Conf. (WCNC)*, Nanjing, China, 2021, pp. 1–6.

[49] L. Salaün *et al.*, "A GNN approach for cell-free massive MIMO," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Rio de Janeiro, Brazil, 2022, pp. 3053–3058.

[50] X. Yan *et al.*, "Access point selection and beamforming design for cell-free network: From fractional programming to GNN," *IEEE Trans. Wireless Commun.*, vol. 23, no. 8, pp. 9345–9360, Aug. 2024.

[51] Y. Ma *et al.*, "Joint power control and precoding for cell-free massive MIMO systems with sparse multi-dimensional graph neural networks," *IEEE Trans. Veh. Technol.*, 2025.

[52] Y. Zhao *et al.*, "A dynamic power allocation approach for downlink cell-free massive MIMO with graph neural network," *IEEE Trans. Veh. Technol.*, vol. 74, no. 4, pp. 6518–6532, Apr. 2025.

[53] I. Atzeni, B. Gouda, and A. Tölli, "Distributed precoding design via over-the-air signaling for cell-free massive MIMO," *IEEE Trans. Wireless Commun.*, vol. 20, no. 2, pp. 1201–1216, Feb. 2021.

[54] F. Fredj *et al.*, "Distributed beamforming techniques for cell-free wireless networks using deep reinforcement learning," *IEEE Trans. Cogn. Commun. Netw.*, vol. 8, no. 2, pp. 1186–1201, June 2022.

[55] D. Wang *et al.*, "Federated learning for precoding design in cell-free massive MIMO systems," *IEEE Open J. Commun. Soc.*, vol. 4, pp. 1567–1582, 2023.

[56] D. Yu, M. Boloursaz Mashhadi, and R. Tafazolli, "Model-based neural collaboration framework for resource-efficient downlink beamforming in cell-free massive MIMO systems," *IEEE Trans. Commun.*, vol. 73, no. 10, pp. 8980–8995, Oct. 2025.

[57] M. Liu, C. Huang, M. Di Renzo, M. Debbah, and C. Yuen, "Cooperative beamforming and RISs association for multi-RISs aided multi-users mmWave MIMO systems through graph neural networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Rome, Italy, May 2023, pp. 4286–4291.

[58] B. Khemani, S. Patil, K. Kotecha, and S. Tanwar, "A review of graph neural networks: concepts, architectures, techniques, challenges, datasets, applications, and future directions," *Journal of Big Data, 11(1), 18*, 2024.

[59] H. Han *et al.*, "SWIPTNet: a unified deep learning framework for SWIPT based on GNN and transfer learning," *IEEE Trans. Mob. Comput.*, pp. 1–11, Apr. 2025 (early access).

[60] Y. Shen, J. Zhang, S. H. Song, and K. B. Letaief, "Graph neural networks for wireless communications: From theory to practice," *IEEE Trans. Wireless Commun.*, vol. 22, no. 5, pp. 3554–3569, May 2023.

[61] A. A. M. Saleh and R. Valenzuela, "A statistical model for indoor multipath propagation," *IEEE J. Sel. Areas Commun.*, vol. SAC-5, no. 2, pp. 128–137, Feb. 1987.

[62] C.-S. Yang *et al.*, "Toward 6G sustainable mobile communications," *IEEE Wireless Commun.*, vol. 32, no. 1, pp. 44–50, Feb. 2025.

[63] H. Cramér, *Random Variables and Probability Distributions*, Cambridge University Press, 1970.

[64] H. Q. Ngo, E. G. Larsson, and T. L. Marzetta, "Energy and spectral efficiency of very large multiuser MIMO systems," *IEEE Trans. Commun.*, vol. 61, no. 4, pp. 1436–1449, Apr. 2013.

[65] T. L. Marzetta, "Noncooperative cellular wireless with unlimited numbers of base station antennas," *IEEE Trans. Wireless Commun.*, vol. 9, no. 11, pp. 3590–3600, Nov. 2010.

[66] D. Chu, "Polyphase codes with good periodic correlation properties (corresp.)," *IEEE Trans. Inf. Theory*, vol. 18, no. 4, pp. 531–532, July 1972.

[67] H. Cheng, J. T. Zhou, W. P. Tay, and B. Wen, "Attentive graph neural networks for few-shot learning," 2020, arXiv preprint arXiv:2007.06878.

[68] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," *arXiv preprint arXiv:1710.10903*, 2017.

[69] Z. Dong, X. Li, and Y. Zeng, "Characterizing and utilizing near-field spatial correlation for XL-MIMO communication," *IEEE Trans. Commun.*, vol. 72, no. 12, pp. 7922–7937, Dec. 2024.

[70] E. M. Taghavi, R. Hashemi, A. Alizadeh, N. Rajatheva, M. Vu, and M. Latva-aho, "Joint active-passive beamforming and user association in IRS-assisted mmwave cellular networks," *IEEE Trans. Veh. Technol.*, vol. 72, no. 8, pp. 10448–10461, Aug. 2023.

[71] "3rd Generation Partnership Project (3GPP). (2020). TR38.901 Specification: Study on Channel Model for Frequencies From 0.5 to 100 GHz (Release 14)," .

[72] R. Liu, P. Xing, Z. Deng, A. Li, C. Guan, and H. Yu, "Federated graph neural networks: Overview, techniques, and challenges," *IEEE Trans. Neural Networks and Learning Systems*, vol. 36, no. 3, pp. 4279–4295, 2025.