Joint User Scheduling and Beamforming Design for Multiuser MISO Downlink Systems

Shiwen He, *Member, IEEE*, Jun Yuan, Zhenyu An, *Student Member, IEEE*, Wei Huang, *Member, IEEE*,Yongming Huang, *Senior Member, IEEE*, and Yaoxue Zhang, *Senior Member, IEEE*

arXiv:2112.01738v1 [cs.IT] 3 Dec 2021

Abstract

In multiuser communication systems, user scheduling and beamforming design are two fundamental problems, which are usually investigated separately in the existing literature. In this work, we focus on the joint optimization of user scheduling and beamforming design with the goal of maximizing the set cardinality of scheduled users. Observing that this problem is computationally challenging due to the non-convex objective function and coupled constraints in continuous and binary variables. To tackle these difficulties, we first propose an iterative optimization algorithm (IOA) relying on the successive convex approximation and uplink-downlink duality theory. Then, motivated by IOA and graph neural networks, a joint user scheduling and power allocation network (JEEPON) is developed to address the investigated problem in an unsupervised manner. The effectiveness of IOA and JEEPON is verified by various numerical results, and the latter achieves a close performance but lower complexity compared with IOA and the greedy-based algorithm. Remarkably, the proposed JEEPON is also competitive in terms of the generalization ability in dynamic wireless network scenarios.

Index Terms

User scheduling, beamforming design, non-convex optimization, graph neural networks.

S. He and J. Yuan are with the School of Computer Science and Engineering, Central South University, Changsha 410083, China. S. He is also with the National Mobile Communications Research Laboratory, Southeast University, and the Purple Mountain Laboratories, Nanjing 210096, China. (email: {shiwen.he.hn, yuanjun}@csu.edu.cn).

Z. An is with the Purple Mountain Laboratories, Nanjing 210096, China. (email: anzhenyu@pmlabs.com.cn).

W. Huang is with the School of Computer Science and Information Engineering, Hefei University of Technology, Hefei 230601, China. (email: huangwei@hfut.edu.cn)

Y. Huang is with the National Mobile Communications Research Laboratory, School of Informatiecience and Engineering, Southeast University, Nanjing 210096, China. He is also with the Purple Mountain Laboratories, Nanjing 210096, China. (email: huangym@seu.edu.cn).

Y. Zhang is with the Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China. (email: zhangyx@tsinghua.edu.cn)

I. Introduction

With the explosive growth of Internet of Things (IoT) devices, wireless communication networks (WCNs) are increasingly facing the challenge of allocating finite transmit power and bandwidth for system utility maximization [1]. Accordingly, one needs to design advanced radio resource management schemes to serve numerous wireless access devices. Massive multiple-input multiple-output (MIMO) and multiuser transmission are two key enablers for supporting larger-scale connection in future WCNs [2]. Therefore, some works have been carried on researching the beamforming design [3], power allocation [4], and user scheduling [5], etc. Significantly, user scheduling and beamforming design are two fundamental problems in multiuser MIMO systems, while they are generally investigated separately in the existing literature, such as optimizing beamforming vectors with a given user set [6] or performing user scheduling combined with power allocation [7].

Meanwhile, the existing researches on coordinated multiuser communication are mainly based on the conventional Shannon theory [8], which assumes that the communication capacity has extremely low decoding error probability with enough long blocklength transmission. However, in the ultra-reliable low latency communication (URLLC) senarios, such as factory automation and remote surgery, this condition with the long blocklength transmission may not be satisfied [9]. To take the impact of finite blocklength transmission into account, the achievable rate has been expressed as a complicated function composed of the received signal-to-noise (SNR), the blocklength, and the decoding error probability, which is smaller than the Shannon rate [10]. Consequently, the optimization problem in scenarios with finite blocklength transmission is more challenging [11]. In order to solve the problem of interest, the algorithms designed in the aforementioned references are mainly based on the convex optimization theory [12]. However, such model-driven optimization algorithms usually suffer from a high computational complexity, which may restrict their practical application ability in WCNs.

Recently, deep neural networks (DNNs) have emerged as an effective tool to solve such challenging radio resource management problems in WCNs [13]. Different from the model-driven optimization algorithms running independently for each instance, DNNs are trained with numerous data to learn the mapping between radio resource optimization policies and wireless network environments. Hence, the main computational cost of DNNs are shifted into the offline training stage, and only simple mathematical operations are needed in the online optimization

stage. The works in [14], [15] show that DNNs could achieve competitive performance with lower computational complexity than existing model-driven optimization algorithms. A similar conclusion has been demonstrated in [16], where DNNs are used for beamforming optimization of multiuser multiple-input single-output (MISO) downlink systems, but the size of the considered problem is rather small. In addition, these DNN-based architectures [14]–[16] are mainly inherited from image processing tasks and not tailored to radio resource management problems, especially the fact that they fail to exploit the prior topology knowledge in WCNs. As the numerical results obtained in [17], the performance of DNNs degrades dramatically with increasing wireless network size.

To achieve a better scalability of learning-based radio resource management, a potential approach is to incorporate the network topology into the learning of neural networks, namely graph neural networks (GNNs) [18]. For instance, the authors of [19] combined DNNs with the geographic location of transceivers, and thereby proposed a spatial convolution model for wireless link scheduling problems with hundreds of nodes. In [20], a random edge graph neural network (REGNN) was developed for power allocation optimization on graphs formed by the interference links within WCNs. The work in [21] demonstrates that GNNs are insensitive to the permutation of data, such as channel state information, and it reflects another advantage of GNNs in terms of permutation invariance property [22]. Further, this work was extended in [23] to solve both power allocation and beamforming design problems via message passing graph neural networks (MPGNNs), which have the ability to generalize to large-scale problems while enjoying a high computational efficiency. However, their proposed designs in [21], [23] only investigated the continuous optimization problems with simple constraints. The discrete optimization problems with complicated constraints are still an opening issue and need to be further considered. Fortunately, the application of primal-dual learning in [24] provides an effective way to solve the complicated constrained radio resource management problems.

In this paper, we consider the joint optimization problem of user scheduling and beamforming design in the multiuser MISO downlink system with finite blocklength transmission. The goal of this work is to maximize the set cardinality of scheduled users subject to the constraints of per-user minimum rate demand and base station (BS) maximum allowable transmit power budget, which is a challenging discrete optimization problem. In brief, the main contributions and advantages of this work are summarized as follows:

- We first derive an iterative optimization algorithm (IOA) to address the investigated problem

using some basic mathematical operations, successive convex approximation (SCA) method, and uplink-downlink duality theory. It is also suitable for problem scenarios based on the Shannon capacity formula.

- We then formulate the investigated problem as a graph optimization problem by modeling the K-user downlink network as a fully-connected graph, and design a novel GNN-based learning algorithm, called JEEPON, to learn the radio resource management on graphs. Meanwhile, a primal-dual learning framework is developed to train JEEPON in an unsupervised manner.
- Finally, we conduct a large number of experiments and the computational complexity comparison for JEEPON, IOA, and the greedy-based algorithm. The obtained results demonstrate that JEEPON enjoys higher computational efficiency while IOA achieves closer performance compared with the greedy-based algorithm. Moreover, JEEPON has the ability to generalize to dynamic wireless network scenarios.

The remainder of this paper is organized as follows. Section II introduces a challenging radio resource management problem in the multiuser MISO downlink system. Section III proposes the IOA for solving the investigated problem. Section IV designs the JEEPON and provides a primal-dual learning framework to train it in an unsupervised manner. Numerical results are presented in Section V. Finally, conclusions are drawn in Section VI.

Notations: Throughout this paper, matrices and vectors are denoted by bold uppercase and lowercase letters, respectively. $|\cdot|$ indicates the absolute value of a complex scalar or the cardinality of a set. $\|\cdot\|_0$, $\|\cdot\|_1$, and $\|\cdot\|_2$ denote the ℓ_0-norm, ℓ_1-norm, and ℓ_2-norm, respectively. The superscripts $(\cdot)^T$, $(\cdot)^H$, and $(\cdot)^{-1}$ denote the transpose, conjugate transpose, and inverse of a matrix, respectively. $\mathbb{R}$, $\mathbb{R}^+$, and $\mathbb{C}$ are the sets of real, non-negative real, and complex numbers, respectively. $\mathbb{R}^{M\times 1}$ and $\mathbb{C}^{M\times 1}$ represent M-dimensional real and complex column vectors, respectively.

II. System Model and Problem Formulation

In this work, we consider a multiuser MISO downlink system with taking the reliable and delivery latency into account, where a BS with N antennas serves K single-antenna users. For simplicity, let $\mathcal{K} = \{1, 2, \cdots, K\}$ and $\mathcal{S} = \{1, 2, \cdots, K^*\} \subseteq \mathcal{K}$ be the set of total users and scheduled users, respectively, where $K^* \leq K$. The channel between BS and the k-th user is denoted as $\mathbf{h}_k \in \mathbb{C}^{N\times 1}$. Let $p_k \geq 0$ and $\mathbf{w}_k \in \mathbb{C}^{N\times 1}$ represent the transmit power and normalized

beamforming vector used by the BS for the k-th user, respectively. Thus, the received signal at the k-th user is given by

$$y_k = \sum_{l\in\mathcal{S}} \sqrt{p_l}\mathbf{h}_k^H \mathbf{w}_l s_l + n_k, \tag{1}$$

where s_l is the normalized data symbol intended for the l-th user, and n_k is the additive white Gaussian noise with $\mathcal{CN}\left(0, \sigma_k^2\right)$ at the k-th user. For notational convenience, we define $\overline{\mathbf{h}}_k = \frac{\mathbf{h}_k}{\sigma_k}$. Then, the signal-to-interference-plus-noise ratio (SINR) of the k-th user is expressed as

$$\gamma_k = \frac{p_k \left|\overline{\mathbf{h}}_k^H \mathbf{w}_k\right|^2}{\sum\limits_{l\neq k, l\in\mathcal{S}} p_l \left|\overline{\mathbf{h}}_k^H \mathbf{w}_l\right|^2 + 1}. \tag{2}$$

To satisfy the extreme requirements of delay, finite blocklength transmission regime is adopted in this paper. The results in [10] show that the achievable rate is not only a function of the received SNR (or SINR), but also the decoding error probability ϵ and the transmission finite blocklength n. Accordingly, the achievable rate of the k-th user in the finite blocklength case is given by[1]

$$R\left(\gamma_k\right) = C\left(\gamma_k\right) - \vartheta\sqrt{V\left(\gamma_k\right)}, \tag{3}$$

where $C\left(\gamma\right) = \ln\left(1+\gamma\right)$ denotes the Shannon capacity, $\vartheta = \frac{Q^{-1}(\epsilon)}{\sqrt{n}}$, $Q^{-1}\left(\cdot\right)$ is the inverse of Gaussian Q-function $Q\left(x\right) = \frac{1}{\sqrt{2\pi}}\int_x^\infty \exp\left(-\frac{t^2}{2}\right)dt$, and $V\left(\gamma\right)$ denotes the channel dispersion, which is defined as

$$V\left(\gamma\right) = 1 - \frac{1}{\left(1+\gamma\right)^2}. \tag{4}$$

The target of this work is to maximize the number of users belonging to the scheduled user set $\mathcal{S}\subseteq\mathcal{K}$ subject to the constraints of per-user minimum achievable rate demand and BS maximum allowable transmit power budget. Specifically, one needs to carefully select the scheduled user set $\mathcal{S}$, and design the transmit beams with reasonable power allocation. To this end, the joint optimization problem of user scheduling and beamforming design is formulated as follows[2]

$$\max_{\{p_k, \mathbf{w}_k\}} \left|\mathcal{S}\right|, \tag{5a}$$

$$\text{s.t. } r_k \leq R\left(\gamma_k\right), \left\|\mathbf{w}_k\right\|_2 = 1, \forall k\in\mathcal{S}, \tag{5b}$$

[1]The proposed algorithms is also suitable for solving similar optimization problems, where the user rate is based on Shannon capacity formula.

[2]For the ultra-dense or large-scale connective URLLC scenario, it may be a better choice to schedule as many users as possible while satisfying reliability and latency requirements. Accordingly, we aim to maximize the set cardinality of scheduled users in this work.

$$\sum_{k\in\mathcal{S}} p_k \leq P, \tag{5c}$$

where $|\mathcal{S}|$ is the cardinality of set $\mathcal{S}$, and P is the maximum allowable transmit power of the BS. Note that constraint (5b) represents the requirements of minimum achievable rate $r_k > 0$ and unit-norm beamforming vector of the k-th user, constraint (5c) indicates the maximum allowable transmit power limit of the BS. Hence, problem (5) is a mixed-integer nonlinear programming problem, which is non-convex and NP-hard. Generally speaking, it is difficult to obtain the global optimal solution, even the near-optimal solution. Although the greedy heuristic search algorithm could be considered as a possible effective solution, it brings extremely high computational complexity, especially in large-scale wireless network scenarios. In the following, to address the investigated problem, we first focus on developing a SCA-based iterative optimization algorithm, and then exploring another scheme with deep learning techniques.

Remark 1. *In general, the heuristic search can be used to determine the solution of problem* (5)*. For this scheme, inspired by the near-far effect of wireless communication, we exploit the difference in channel quality to select the users to be scheduled. Specifically, we develop the greedy heuristic search algorithm (GHSA), which first ranks the user channels from good to bad, and then filters the scheduled user set by a limited number of exhaustive and backtracking searches [25]. Since this algorithm has close performance and lower computational complexity compared with the exhaustive search algorithm, we shall use it as the baseline.*

III. Design of Optimization Algorithm

In this section, we pay our attention on designing an effective optimization algorithm for problem (5) from the perspective of conventional optimization theory. Since problem (5) is non-convex, the first thing is to transform it into a tractable form via some basic mathematical operations. One idea is to use the downlink-uplink duality [26] to transform the downlink problem (5) into a virtual uplink dual problem, i.e.,

$$\max_{\{q_k,\mathbf{w}_k\}} |\mathcal{S}|, \tag{6a}$$

$$\text{s.t. } r_k \leq R\left(\overleftarrow{\gamma}_k\right), \|\mathbf{w}_k\|_2 = 1, \forall k \in \mathcal{S}, \tag{6b}$$

$$\sum_{k\in\mathcal{S}} q_k \leq P,\ q_k \geq 0, \forall k \in \mathcal{S}, \tag{6c}$$

where q_k represents the virtual uplink transmit power of the k-th user, and $\overleftarrow{\gamma}_k$ indicates the virtual uplink SINR of the k-th user, which is defined as

$$\overleftarrow{\gamma}_k = \frac{q_k \left|\overline{\mathbf{h}}_k^H \mathbf{w}_k\right|^2}{\sum\limits_{l\neq k, l\in\mathcal{S}} q_l \left|\overline{\mathbf{h}}_l^H \mathbf{w}_k\right|^2 + 1}. \tag{7}$$

To clearly indicate whether a user is scheduled or not, we further introduce κ_k as a binary variable indicator of the user state, with $\kappa_k = 1$ if user k is scheduled and $\kappa_k = 0$ otherwise, $k \in \mathcal{K}$. Let $\boldsymbol{\kappa} = [\kappa_1, \kappa_2, \cdots, \kappa_k, \cdots, \kappa_K]^T$, thus, problem (6) can be reformulated as

$$\max_{\{\kappa_k, q_k, \mathbf{w}_k\}} \|\boldsymbol{\kappa}\|_0, \tag{8a}$$

$$\text{s.t. } \kappa_k r_k \leq R\left(\overleftarrow{\gamma}_k\right), \|\mathbf{w}_k\|_2 = 1, \forall k \in \mathcal{K}, \tag{8b}$$

$$\kappa_k \in \{0, 1\}, \forall k \in \mathcal{K}, \tag{8c}$$

$$\sum_{k\in\mathcal{K}} q_k \leq P,\ q_k \geq 0, \forall k \in \mathcal{K}. \tag{8d}$$

In problem (8), without incurring any confusion, the SINR $\overleftarrow{\gamma}_k$ expression in (7) is redefined as

$$\overleftarrow{\gamma}_k = \frac{q_k \left|\overline{\mathbf{h}}_k^H \mathbf{w}_k\right|^2}{\sum\limits_{l\neq k, l\in\mathcal{K}} q_l \left|\overline{\mathbf{h}}_l^H \mathbf{w}_k\right|^2 + 1}. \tag{9}$$

Remark 2. *In the designed algorithm, $\kappa_k = 1$ means that the k-th user is scheduled and sufficient transmission power will be allocated for the user, namely, $R(\gamma_k) \geq r_k$. On the other hand, $\kappa_k = 0$ means that the k-th user will not be scheduled. In such circumstance, if no power is allocated for the k-th user, then expressions $\kappa_k = 0, R(\gamma_k) = 0, ||\mathbf{w}_k||_2 = 0$ hold. However, the situation $\kappa_k = 0, 0 < R(\gamma_k) < r_k$ may also occur. For instance, for a candidate user set $\mathbf{O}$ and scheduled user set $\mathbf{P}, \mathbf{P} \subset \mathbf{O}$. Transmission power of the BS is not always precisely exhausted for scheduling user set $\mathbf{P}$. For any user k from the rest user set $\mathbf{O} \setminus \mathbf{P}$, if the residual power could not meet the minimum transmission power requirement, then we have $\kappa_k = 0, 0 < R(\gamma_k) < r_k, ||\mathbf{w}_k||_2 > 0$. This special case does not affect the whole solving process, however, we still filter the scheduled users with minimum transmission power requirement to ensure the feasibility of scheduled user set $\mathbf{P}$.*

The goal of function (8) is to assure that the users as many as possible are scheduled under the given constraints. According to the conclusion obtained in [11, Corollary 1], constraint (8b)

can be transformed equivalently into a new constraint expressed with SINR form. Let $\widetilde{\gamma}_k > 0$ be the minimum SINR for achieving minimum achievable rate r_k. Thus, problem (8) can be transformed into the following form

$$\max_{\{\kappa_k, q_k, \mathbf{w}_k\}} \|\boldsymbol{\kappa}\|_0, \tag{10a}$$

$$\text{s.t. } \kappa_k \widetilde{\gamma}_k \leq \overleftarrow{\gamma}_k, \|\mathbf{w}_k\|_2 = 1, \forall k \in \mathcal{K}, \tag{10b}$$

$$\kappa_k \in \{0, 1\}, \forall k \in \mathcal{K}, \tag{10c}$$

$$\sum_{k \in \mathcal{K}} q_k \leq P, \ q_k \geq 0, \forall k \in \mathcal{K}. \tag{10d}$$

To release the constraint (10c), we rewrite problem (10) into the following form [27]

$$\max_{\{\kappa_k, q_k, \mathbf{w}_k\}} \sum_{k \in \mathcal{K}} \kappa_k, \tag{11a}$$

$$\text{s.t. } 0 \leq \kappa_k \leq 1, \forall k \in \mathcal{K}, \tag{11b}$$

$$\sum_{k \in \mathcal{K}} \left(\kappa_k - \kappa_k^2\right) \leq 0, \tag{11c}$$

$$\text{(10b), (10d).} \tag{11d}$$

Constraints (11b) and (11c) assure that the value of κ_k equals to either one or zero, i.e., $\kappa_k \in \{0, 1\}$, $\forall k \in \mathcal{K}$. According to [28, Proposition 2], the strong Lagrangian duality holds for problem (11). We introduce similar mathematical tricks on handling constraint (11c), and rewrite the problem (11) as follows

$$\min_{\{\kappa_k, q_k, \mathbf{w}_k\}} -\sum_{k \in \mathcal{K}} \kappa_k + g(\boldsymbol{\kappa}) - h(\boldsymbol{\kappa}), \tag{12a}$$

$$\text{s.t. (10b), (10d), (11b),} \tag{12b}$$

where λ is a proper non-negative constant, and $g(\boldsymbol{\kappa})$ and $h(\boldsymbol{\kappa})$ are defined respectively as

$$g(\boldsymbol{\kappa}) \triangleq \lambda \sum_{k \in \mathcal{K}} \kappa_k + \lambda \left(\sum_{k \in \mathcal{K}} \kappa_k\right)^2, \tag{13a}$$

$$h(\boldsymbol{\kappa}) \triangleq \lambda \sum_{k \in \mathcal{K}} \kappa_k^2 + \lambda \left(\sum_{k \in \mathcal{K}} \kappa_k\right)^2. \tag{13b}$$

Note that the beamforming vectors $\{\mathbf{w}_k\}$ is only related with SINR $\overleftarrow{\gamma}_k$. Therefore, to obtain the optimal solution of problem (11), we need to maximize SINR $\overleftarrow{\gamma}_k$ with respect to $\{\mathbf{w}_k\}$.

Further, when the uplink transmit power $\{q_k\}$ is fixed, the optimal $\{\mathbf{w}_k\}$ for maximizing SINR $\overleftarrow{\gamma}_k$ is the minimum mean square error receiver, i.e.,

$$\mathbf{w}_k^* = \frac{\left(\mathbf{I}_N + \sum\limits_{k\in\mathcal{K}} q_k \overline{\mathbf{h}}_k \overline{\mathbf{h}}_k^H\right)^{-1} \overline{\mathbf{h}}_k}{\left\| \left(\mathbf{I}_N + \sum\limits_{k\in\mathcal{K}} q_k \overline{\mathbf{h}}_k \overline{\mathbf{h}}_k^H\right)^{-1} \overline{\mathbf{h}}_k \right\|_2}, \tag{14}$$

where $\mathbf{I}_N$ denotes N-by-N identity matrix. For the fixed $\{\mathbf{w}_k\}$, the inner optimization of problem (12) is reformulated as

$$\min_{\{\kappa_k, q_k\}} -\sum_{k\in\mathcal{K}} \kappa_k + g(\boldsymbol{\kappa}) - h(\boldsymbol{\kappa}), \tag{15a}$$

$$\text{s.t. } \widetilde{\gamma}_k \kappa_k - q_k \left|\overline{\mathbf{h}}_k^H \mathbf{w}_k\right|^2 + \varphi_k(\boldsymbol{\kappa}, \mathbf{q}) - \phi_k(\boldsymbol{\kappa}, \mathbf{q}) \le 0, \forall k \in \mathcal{K}, \tag{15b}$$

$$(10d),\ (11b), \tag{15c}$$

where $\varphi_k(\boldsymbol{\kappa}, \mathbf{q})$ and $\phi_k(\boldsymbol{\kappa}, \mathbf{q})$ are defined as

$$\varphi_k(\boldsymbol{\kappa}, \mathbf{q}) \triangleq \frac{1}{2}\left(\widetilde{\gamma}_k \kappa_k + \sum_{l\in\mathcal{K}, l\neq k} q_l \left|\overline{\mathbf{h}}_l^H \mathbf{w}_k\right|^2\right)^2, \tag{16a}$$

$$\phi_k(\boldsymbol{\kappa}, \mathbf{q}) \triangleq \frac{1}{2}\widetilde{\gamma}_k^2 \kappa_k^2 + \frac{1}{2}\left(\sum_{l\in\mathcal{K}, l\neq k} q_l \left|\overline{\mathbf{h}}_l^H \mathbf{w}_k\right|^2\right)^2. \tag{16b}$$

Problem (15) belongs to the class of difference of convex programming problem, since the objective function (15a) and constraint (15b) are the difference of two convex functions. In the sequel, we resort to the classic SCA-based methods [29]. Using respectively the convexity of functions $h(\boldsymbol{\kappa})$ and $\phi(\boldsymbol{\kappa}, \mathbf{q})$, we have

$$h(\boldsymbol{\kappa}) \ge \psi(\boldsymbol{\kappa}) \triangleq h\left(\kappa^{(\tau)}\right) + \sum_{k\in\mathcal{K}} h'\left(\kappa_k^{(\tau)}\right)\left(\kappa_k - \kappa_k^{(\tau)}\right), \tag{17}$$

where $h'(\kappa_k) \triangleq 2\lambda\left(\kappa_k + \sum\limits_{l\in\mathcal{K}} \kappa_l\right)$ and superscript τ denotes the τ-th iteration of the iterative algorithm presented shortly.

$$\begin{aligned} &\phi_k(\boldsymbol{\kappa}, \mathbf{q}) \ge \varrho_k(\boldsymbol{\kappa}, \mathbf{q}) \triangleq \phi_k\left(\boldsymbol{\kappa}^{(\tau)}, \mathbf{q}^{(\tau)}\right) \\ &+ \widetilde{\gamma}_k^2 \kappa_k^{(\tau)}\left(\kappa_k - \kappa_k^{(\tau)}\right) + \sum_{l\in\mathcal{K}, l\neq k} \rho_{k,l}\left(\mathbf{q}^{(\tau)}\right)\left(q_l - q_l^{(\tau)}\right), \end{aligned} \tag{18}$$

where $\rho_{k,m}(\mathbf{q}) \triangleq \left|\overline{\mathbf{h}}_m^H \mathbf{w}_k\right|^2 \sum\limits_{n\in\mathcal{K}, n\neq k} q_n \left|\overline{\mathbf{h}}_n^H \mathbf{w}_k\right|^2$. From the aforementioned discussions, the convex approximation problem solved at the $(\tau+1)$-th iteration of the proposed algorithm is given by

$$\min_{\{\kappa_k, q_k\}} -\sum_{k\in\mathcal{K}} \kappa_k + g(\boldsymbol{\kappa}) - \psi(\boldsymbol{\kappa}), \tag{19a}$$

$$\text{s.t. } \widetilde{\gamma}_k \kappa_k - q_k \left| \overline{\mathbf{h}}_k^H \mathbf{w}_k \right|^2 + \varphi_k(\boldsymbol{\kappa}, \mathbf{q}) - \varrho_k(\boldsymbol{\kappa}, \mathbf{q}) \leq 0, \forall k \in \mathcal{K}, \tag{19b}$$

$$(10d),\ (11b). \tag{19c}$$

Based on the above mathematical transformation, the IOA is summarized in Algorithm 1. In the description of Algorithm 1, δ denotes the maximum permissible error, and $\upsilon^{(\tau)}$ and $\zeta^{(t)}$ denote the objective value of problem (12) at the τ-th iteration and problem (19) at the τ-th iteration and the t-th iteration, respectively. The convergence of IOA can be guaranteed by the monotonic boundary theory. To speed up the convergence of IOA, we can first filter out the users who meets constraints (5b) and (5c) by adopting a single user communication with maximum ratio transmission and full power transmission.

Algorithm 1 Solution for constrained problem (12)

1: Let $\tau = 0$, $t = 0$, $\lambda = 10^{-2}$, and $\delta = 10^{-5}$. Initialize beamforming vector $\mathbf{w}_k^{(0)}$ and power allocation $p_k^{(0)}$, $\forall k \in \mathcal{K}$, such that constraint (5b) and (5c) are satisfied.

2: Initialize $\upsilon^{(0)}$ and $\zeta^{(0)}$, compute γ_k with $\mathbf{w}_k^{(0)}$ and $p_k^{(0)}$ to obtain $\overline{\gamma}_k$, and compute $q_k^{(0)}$ that is the k-th element of vector $\mathbf{q} = [q_1, \cdots, q_K]^T$ with

$$\mathbf{q} = \boldsymbol{\Psi}^{-1}\mathbf{1}, \tag{20}$$

where matrix $\boldsymbol{\Psi}$ is given by

$$[\boldsymbol{\Psi}]_{k,l} = \begin{cases} \dfrac{\left| \overline{\mathbf{h}}_k^H \mathbf{w}_k^{(0)} \right|^2}{\overline{\gamma}_k}, k = l, \\ -\left| \overline{\mathbf{h}}_l^H \mathbf{w}_k^{(0)} \right|^2, k \neq l. \end{cases} \tag{21}$$

3: Let $t \leftarrow t + 1$. Solve problem (19) to obtain $\kappa_k^{(t)}$, $q_k^{(t)}$, and $\zeta^{(t)}$, $\forall k \in \mathcal{K}$.

4: If $\left| \frac{\zeta^{(t)} - \zeta^{(t-1)}}{\zeta^{(t-1)}} \right| \leq \delta$, go to Step 5. Otherwise, go to Step 3.

5: Let $\tau = \tau + 1$, update $\mathbf{w}_k^{(\tau)}$ with $q_k^{(t)}$ and (14), and calculate the objective value $\upsilon^{(\tau)}$. If $\left| \frac{\upsilon^{(\tau)} - \upsilon^{(\tau-1)}}{\upsilon^{(\tau-1)}} \right| \leq \delta$, stop iteration and output $\kappa_k^{(t)}$, $q_k^{(t)}$, and $\mathbf{w}_k^{(\tau)}$. Otherwise, go to Step 3.

Nevertheless, with the expansion of wireless network scale, the convergence speed of IOA will slow down under predictable circumstances. In the sequel, to alleviate the computation burden while achieving near-optimal performance, we attempt to design a GNN-based learning model to solve problem (11).

IV. Graph Neural Network For Optimization Problem

In this section, we inherit the idea of IOA and propose an novel GNN-based learning algorithm to solve the problem (11). Specifically, we propose a *joint user scheduling and power allocation network*, termed as JEEPON, to learn the radio resource management on graphs. In addition, we also develop a primal-dual learning framework (PDLF) to train JEEPON in an unsupervised manner. Different from the proposed IOA that alternately updates the beamforming vectors, PDLF regards them as intermediate variables relevant to the uplink power. Accordingly, substituting the expression in (14) into (9), we have

$$\widehat{\gamma}_k = \frac{q_k \left|\overline{\mathbf{h}}_k^H \mathbf{\Lambda}^{-1} \overline{\mathbf{h}}_k\right|^2}{\sum\limits_{l\neq k, l\in\mathcal{K}} q_l \left|\overline{\mathbf{h}}_l^H \mathbf{\Lambda}^{-1} \overline{\mathbf{h}}_k\right|^2 + \left\|\mathbf{\Lambda}^{-1} \overline{\mathbf{h}}_k\right\|_2^2}, \tag{22}$$

where $\mathbf{\Lambda} = \mathbf{I}_N + \sum\limits_{k\in\mathcal{K}} q_k \overline{\mathbf{h}}_k \overline{\mathbf{h}}_k^H$. Thus, we rewrite problem (11) as follows

$$\max_{\{\kappa_k, q_k\}} \sum_{k\in\mathcal{K}} \kappa_k, \tag{23a}$$

$$\text{s.t. } 0 \leq \kappa_k \leq 1, \forall k \in \mathcal{K}, \tag{23b}$$

$$\sum_{k\in\mathcal{K}} \left(\kappa_k - \kappa_k^2\right) \leq 0, \tag{23c}$$

$$\kappa_k \widetilde{\gamma}_k \leq \widehat{\gamma}_k, \forall k \in \mathcal{K}, \tag{23d}$$

$$\sum_{k\in\mathcal{K}} q_k \leq P,\ q_k \geq 0, \forall k \in \mathcal{K}. \tag{23e}$$

To include those constraints into the training process of PDLF, we adopt the violation-based Lagrangian relaxation method [30] to formulate problem (23) as an unconstrained optimization problem. Observe that constraints (23b) and (23e) are easily satisfied by projecting them into the feasible region. The specific projection method will be discussed later. For constraints (23c) and (23d), we introduce the non-negative Lagrangian multipliers $\mu, \nu \in \mathbb{R}^+$ to capture how much the constraints are violated. Thus, the partial Lagrangian relaxation function of problem (23) is given by

$$\mathcal{L}(\boldsymbol{\kappa}, \mathbf{q}, \mu, \nu) = -\sum_{k\in\mathcal{K}} \kappa_k + \mu \sum_{k\in\mathcal{K}} \chi_c^{\geq}\left(\kappa_k - \kappa_k^2\right) + \nu \sum_{k\in\mathcal{K}} \chi_c^{\geq}\left(\kappa_k \widetilde{\gamma}_k - \widehat{\gamma}_k\right), \tag{24}$$

where $\chi_c^{\geq}(\cdot)$ represents the violation degree of an inequality constraint. Intuitively, for an inequality constraint $c(x) : ax-b \leq 0$, the violation degrees can be expressed as $\chi_c^{\geq}(x) = \max(0, \sigma_c(x))$, where $\sigma_c(x) = ax - b$. Further, the Lagrangian dual problem of (23) is formulated as

$$\max_{\mu,\nu} \min_{\boldsymbol{\kappa},\mathbf{q}} \mathcal{L}(\boldsymbol{\kappa}, \mathbf{q}, \mu, \nu). \tag{25}$$

In the sequel, we mainly discuss the entire architecture and implement process of PDLF. The key properties will also be discussed.

A. Primal-Dual Learning Framework

With regard to the aforementioned aspects, PDLF is developed for solving the Lagrangian dual problem (25), which is composed of two parts, the primal update part and the dual update part, as shown in Fig. 1. Specifically, in the primal update part, we propose a graph design module for constructing wireless network topology, a JEEPON module for optimizing vectors $\boldsymbol{\kappa}, \mathbf{q}$ (primal variables), a matrix inversion conversion (MIC) module for SINR computation, and a loss function module for updating the network parameters of JEEPON. In the dual part, we design a Lagrangian multipliers μ, ν (dual variables) update module by means of the subgradient optimization method. PDLF uses a limited number of epochs to alternately update the optimization vectors $\boldsymbol{\kappa}, \mathbf{q}$ and Lagrangian multipliers μ, ν. In particular, PDLF seeks to maximize the partial Lagrangian relaxation function (24) over vectors $\boldsymbol{\kappa}, \mathbf{q}$ with fixed μ, ν, and then minimize function (24) over μ, ν with fixed vectors $\boldsymbol{\kappa}, \mathbf{q}$. As a result, for the outer optimization of problem (25), with the given vectors $\boldsymbol{\kappa}^{(\tau)}, \mathbf{q}^{(\tau)}$ at the τ-th epoch, the Lagrangian multipliers are updated as

$$\begin{aligned} \mu^{(\tau+1)} &= \mu^{(\tau)} + \varepsilon_\mu \sum_{k\in\mathcal{K}} \chi_c^{\geq} \left(\kappa_k^{(\tau)} - (\kappa_k^{(\tau)})^2 \right), \\ \nu^{(\tau+1)} &= \nu^{(\tau)} + \varepsilon_\nu \sum_{k\in\mathcal{K}} \chi_c^{\geq} \left(\kappa_k^{(\tau)} \widetilde{\gamma}_k - \widehat{\gamma}_k \right), \end{aligned} \tag{26}$$

where ε_μ and ε_ν denote the update step-size of Lagrangian multipliers μ and ν, respectively. In addition, the Lagrangian multipliers are updated every epoch based on the violation degree of the training datasets. For the inner optimization of problem (25), JEEPON is designed to transform it into a statistical learning problem, which aims to obtain appropriate optimization vectors $\boldsymbol{\kappa}, \mathbf{q}$ by updating the network parameters of JEEPON.

The pseudocode of PDLF is summarized in Algorithm 2. Unlike the penalty-based training method in [16], the proposed PDLF jointly updates the JEEPON network parameters and the

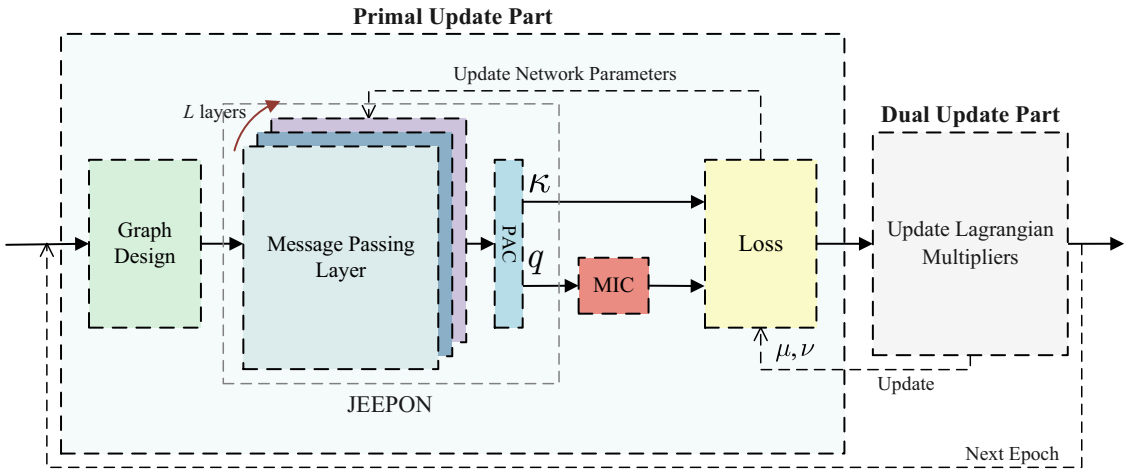


Fig. 1: The primal-dual learning framework.

dual variables in a single training stage. Specifically, we generate a training dataset $\mathcal{D} \triangleq \{\mathcal{D}_i\}_{i=1}^{N_\mathrm{s}}$ with N_s samples, and each sample with the same size. The training stage of PDLF lasts for N_e epochs in total. For notational convenience, let Θ and $\Phi(\mathcal{D}_i, \Theta) \triangleq \{\boldsymbol{\kappa}^{(i)}, \mathbf{q}^{(i)}\}$ represent trainable network parameter set of JEEPON and the radio resource optimization policy corresponding to sample $\mathcal{D}_i \in \mathcal{D}$, respectively. In the primal update part, PDLF first constructs the graph $\mathcal{G}_i(\mathcal{V}, \mathcal{E})$ for sample $\mathcal{D}_i$ (Setp 6), and takes it as the input of JEEPON. Then, JEEPON outputs the optimization policy $\Phi(\mathcal{D}_i, \Theta)$ associated with sample $\mathcal{D}_i$ (Step 7), and adopt the gradient optimizer with an appropriate learning rate η to update network parameters of JEEPON (Step 8). We also store the sub-gradient values of dual variables in this part to avoid repeated traversal of the training datasets (Steps 10,11). Therefore, in the dual update part, we can directly update the dual variables by adding the corresponding dual gradient variables in (26) (Step 13). The details of JEEPON are discussed further in the following section.

Remark 3. *PDLF is designed for training JEEPON, which alternately updates the network parameters of JEEPON and the Lagrangian multipliers to minimize the loss function* (24). *Once the JEEPON is trained, it can directly perform predictions for similarly configured wireless network scenarios without retraining via PDLF.*

B. Graph Representation and Model Design

The communication and interference links within wireless networks can be naturally modeled as a fully-connected graph with node and edge features [31]. In this work, we regard the user

Algorithm 2 PDLF For Training JEEPON

Input: $\mathcal{D} \triangleq \{\mathcal{D}_i\}_{i=1}^{N_\mathrm{s}}$: Training dataset with N_s samples;

N_e: Number of epochs;

η: The learning rate of gradient optimizer;

Θ: The trainable network parameter set of JEEPON;

$\varepsilon_\mu, \varepsilon_\nu$: Step size of Lagrangian multipliers.

Output: The optimization vectors $\boldsymbol{\kappa}, \mathbf{q}$.

Initialize Lagrangian multipliers: $\mu^{(0)}, \nu^{(0)}$.
Initialize the network parameter set Θ of JEEPON.
for epoch $\tau \leftarrow 1, 2, \cdots, N_\mathrm{e}$ **do**
  Initialize dual gradient variables: $\nabla_\mu^{(0)}, \nabla_\nu^{(0)}$.
  for each sample $\mathcal{D}_i : i \leftarrow 1, 2, \cdots, N_\mathrm{s}$ **do**
    Construct the graph $\mathcal{G}_i(\mathcal{V}, \mathcal{E})$ for sample $\mathcal{D}_i$.
    Obtain the policy $\Phi(\mathcal{D}_i, \Theta) \triangleq \{\boldsymbol{\kappa}^{(i)}, \mathbf{q}^{(i)}\}$ via JEEPON.
    Update Θ to minimize (24) through the gradient optimizer with learning rate η.
    Update dual gradient variables:
      $\nabla_\mu^{(i)} \leftarrow \nabla_\mu^{(i-1)} + \sum_{k \in \mathcal{D}_i} \chi_c^{\geq}(\kappa_k^{(i)} - (\kappa_k^{(i)})^2)$,
      $\nabla_\nu^{(i)} \leftarrow \nabla_\nu^{(i-1)} + \sum_{k \in \mathcal{D}_i} \chi_c^{\geq}(\kappa_k^{(i)} \widetilde{\gamma}_k - \widehat{\gamma}_k)$.
  end for
  Update Lagrangian multipliers: $\mu^{(\tau)} \leftarrow \mu^{(\tau-1)} + \varepsilon_\mu \nabla_\mu^{(N_\mathrm{s})}$, $\nu^{(\tau)} \leftarrow \nu^{(\tau-1)} + \varepsilon_\nu \nabla_\nu^{(N_\mathrm{s})}$.
end for

devices (UE) as nodes and interference links between UEs as edges. Thus, a wireless channel graph can be represented as $\mathcal{G}(\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ denotes the node set consisting of all UEs, and $\mathcal{E}$ denotes the edge set that includes all interference links. In this way, a toy of the four-user wireless channel graph is illustrated in Fig. 2, where UE_v denotes the v-th node, $v \in \mathcal{V}$, and gray dashed line denotes the interference edge between two different nodes. The nodes and edges in graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ have different features, e.g, a directed edge $\mathcal{E}(u, v)$ connecting nodes u (with features $\mathbf{x}_u$) and v (with features $\mathbf{x}_v$), has edge feature $\mathbf{e}_{u,v}$, $u, v \in \mathcal{V}$. Specifically, we define the feature vector of node v as $\mathbf{x}_v = |\overline{\mathbf{h}}_v^H \mathbf{h}_v|$ and the feature vector of edge $\mathcal{E}(u, v)$ as $\mathbf{e}_{u,v} = |\overline{\mathbf{h}}_u^H \mathbf{h}_v|$, which reflect the properties of the nodes and edges in graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$.

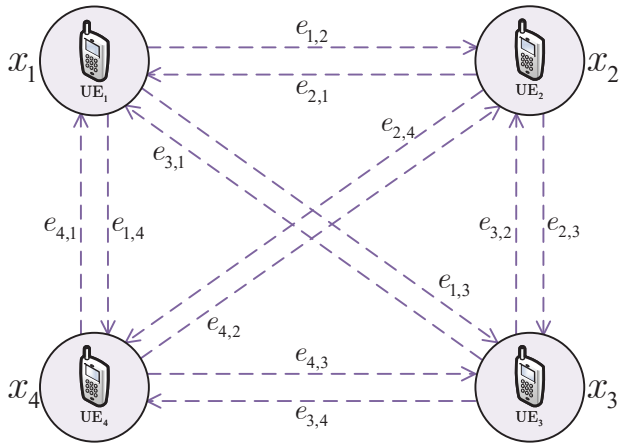


Fig. 2: Graph representation of a four-user wireless channel network.

After constructing graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$, we focus on the design of JEEPON to learn the radio resource management, where the optimization vectors $\boldsymbol{\kappa}, \mathbf{q}$ are carefully defined in the representation vector of nodes. Specifically, JEEPON applies a message passing mechanism [32] to iteratively update the representation vector of node v by aggregating features from its neighbor nodes and edges, $v \in \mathcal{V}$. This message passing process contains two-step. First, the model generates and gathers messages for each node from its first-order neighboring nodes and edges. Then, the model uses the aggregated messages to update the representation vector of nodes. After ℓ message passing, the representation vector of node v captures the messages within its ℓ-hop neighborhood nodes and edges. To be specific, the update rule of the ℓ-th message passing layer at node v is formulated as follows

$$\begin{aligned}
\mathbf{m}_{u,v}^{(\ell)} &= M_\theta^{(\ell)}\left(\boldsymbol{\beta}_u^{(\ell-1)}, \mathbf{x}_u, \mathbf{e}_{u,v}\right), u \in \mathcal{N}(v), \\
\mathbf{g}_v^{(\ell)} &= AGG\left(F_{\max}(\{\mathbf{m}_{u,v}^{(\ell)}\}), F_{\text{mean}}(\{\mathbf{m}_{u,v}^{(\ell)}\})\right), u \in \mathcal{N}(v), \\
\boldsymbol{\beta}_v^{(\ell)} &= U_\theta^{(\ell)}\left(\boldsymbol{\beta}_v^{(\ell-1)}, \mathbf{x}_v, \mathbf{g}_v^{(\ell)}\right), v \in \mathcal{V},
\end{aligned} \tag{27}$$

where $\mathcal{N}(v)$ is the first-order neighborhood set of node v. $\boldsymbol{\beta}_v^{(\ell)} = [\kappa_v, q_v] \in \mathbb{R}^2$ represents the pairwise optimization vector associated with node v at the ℓ-th layer, and $\boldsymbol{\beta}_v^{(0)}$ is initialized with an all-zero vector. Therefore, when the update of the ℓ-th message passing layer is completed, the representation vector of node v could be formulated as $[\boldsymbol{\beta}_v^{(\ell)}, \mathbf{x}_v]$. Fig. 3 illustrates the state update process of node v at the ℓ-th message passing layer. Here, $M_\theta^{(\ell)}(\cdot)$ is a message construction function defined on each edge to generate edge message $\mathbf{m}_{u,v}^{(\ell)} \in \mathbb{R}^m$ by combining the edge feature with the features of its incident nodes, where m is the output dimension size.

$AGG(\cdot)$ is a message aggregation function[3] that uses the concatenation of two set functions $F_{\max}(\cdot)$ and $F_{\text{mean}}(\cdot)$ to gather all relevant edge messages $\{\mathbf{m}_{u,v}^{(\ell)}|u \in \mathcal{N}(v)\}$ and generate the aggregated message $\mathbf{g}_v^{(\ell)} \in \mathbb{R}^{2m}$, where these two functions represent the maximum and average operations, respectively. $U_\theta^{(\ell)}(\cdot)$ is a state update function defined on each node to update the node representation by aggregating its incoming message $\mathbf{g}_v^{(\ell)}$. In JEEPON, function $M_\theta^{(\ell)}(\cdot)$ and function $U_\theta^{(\ell)}(\cdot)$ are parameterized by two different neural network modules.

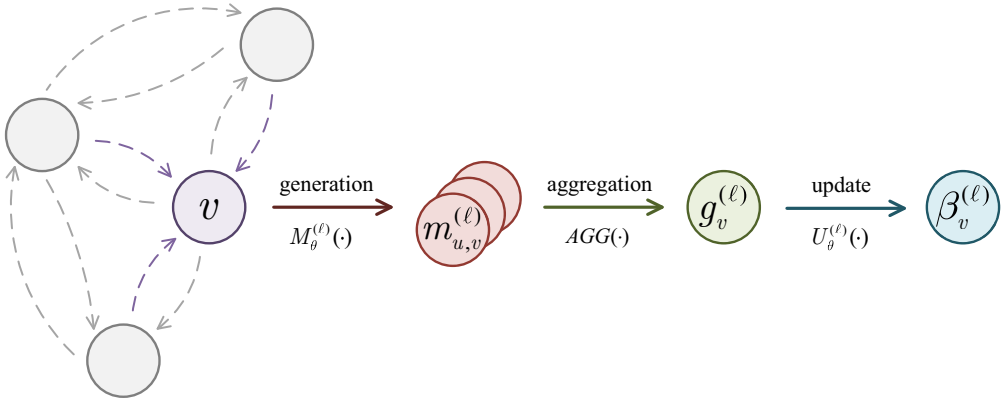


Fig. 3: The state update process of node v at the ℓ-th message passing layer.

Through the combination of several message passing layers, JEEPON can gather multi-hop node and edge features. An illustration of JEEPON is given in Fig. 4, which consists of L message passing layers and one projection activation (PAC) layer. Each message passing layer includes an input layer, an output layer, and two different MLPs which are composed of linear (LN) layers, batch normalization (BN) layers and activation (AC) layers. At the ℓ-th message passing layer, we update the optimization vector $\boldsymbol{\beta}_v^{(\ell)}$ of node $v \in \mathcal{V}$ by taking the neighbor node feature set $\mathbf{X}_u \triangleq \{\mathbf{x}_u|u \in \mathcal{N}(v)\}$, edge feature set $\mathbf{E}_{u,v} \triangleq \{\mathbf{e}_{u,v}|u \in \mathcal{N}(v)\}$ and previous layer node optimization vector $\boldsymbol{\beta}_v^{(\ell-1)}$ as input. Finally, the optimization vectors $\kappa_v^{(L)}$ and $q_v^{(L)}$ are obtained from the node optimization vector $\boldsymbol{\beta}_v^{(L)}$ at the last layer. In the final part of JEEPON, we utilize the PAC layer to put the optimization vectors $\boldsymbol{\kappa}, \mathbf{q}$ into the feasible region $\boldsymbol{\Omega} \triangleq \{\boldsymbol{\kappa}, \mathbf{q} : 0 \leq \kappa_k \leq 1, q_k \geq 0, \|\mathbf{q}\|_1 \leq P, \forall k \in \mathcal{K}\}$. To this end, we first use the rectified linear unit (ReLU) activation function $F_{\text{relu}}(\mathbf{z}) = \max\{\mathbf{0}, \mathbf{z}\}$ to make $\kappa_v^{(L)}, q_v^{(L)} \geq 0$, and then apply two projection

[3]To enable the permutation invariance property of JEEPON, $AGG(\cdot)$ tends to use set functions, such as sum, mean, or max/min.

functions to meet the constraints (23b) and (23e), which are defined as

$$\begin{aligned} \boldsymbol{\kappa}^* &= \min\{\boldsymbol{\kappa}, \mathbf{1}\}, \\ \mathbf{q}^* &= \frac{P}{\max\{P, \|\mathbf{q}\|_1\}}\mathbf{q}, \end{aligned} \tag{28}$$

where $\mathbf{0}, \mathbf{1} \in \mathbb{R}^K$ represent all-zero and all-one column vector with length K, respectively. The maximum and the minimum operations in (28) are performed in an element-wise manner.

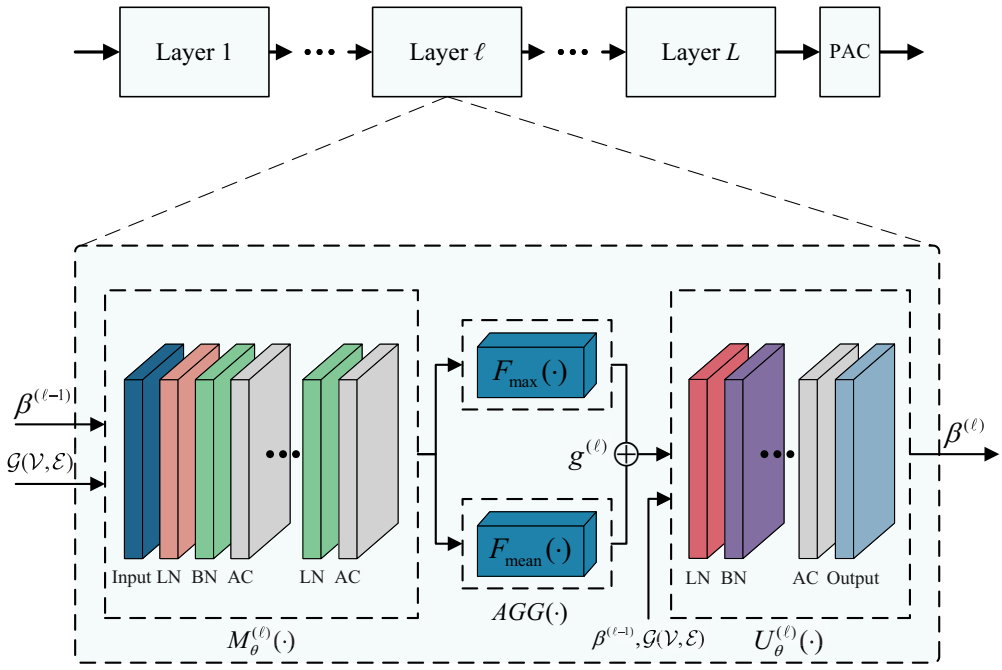


Fig. 4: The architecture of JEEPON with L message passing layers and one PAC layer.

In PDLF, we use the loss function module to numerically update the network parameters of JEEPON, which is implemented by *Pytorch*, a state-of-art deep learning library. However, due to the existence of matrix inversion and complex number operations in the uplink SINR equation (22), the subsequent calculation process of the loss function is time-consuming. To speed up the computation, we apply the following *Lemma 1* to construct a MIC module that replaces the direct matrix inversion by K matrix iterations. Specifically, it reduces the computational complexity to $\mathcal{O}(KN^2)$ instead of matrix inversion complexity $\mathcal{O}(KN^2 + N^3)$, where $\mathcal{O}(\cdot)$ represents the big-$\mathcal{O}$ notation for describing the computational complexity of the function.

Lemma 1. *According to the Sherman-Morrison formula [33], for an invertible square matrix* $\mathbf{A} \in \mathbb{C}^{N\times N}$*, if there exists two column vectors* $\mathbf{u}, \mathbf{v} \in \mathbb{C}^{N\times 1}$*,* $1 + \mathbf{v}^H\mathbf{A}^{-1}\mathbf{u} \neq 0$*, then the inverse*

of $\mathbf{A}$ *is given by*

$$(\mathbf{A}+\mathbf{u}\mathbf{v}^H)^{-1}=\mathbf{A}^{-1}-\frac{\mathbf{A}^{-1}\mathbf{u}\mathbf{v}^H\mathbf{A}^{-1}}{1+\mathbf{v}^H\mathbf{A}^{-1}\mathbf{u}}. \tag{29}$$

Based on this formula, let $\mathbf{T}_n=\mathbf{\Lambda}^{-1}, n\in\{0,1,\cdots,K\}$*, then* $\mathbf{T}_n$ *can be converted into an iterative matrix product form, which is formulated as follows*

$$\mathbf{T}_n=\begin{cases}\mathbf{I}_N, n=0,\\ \mathbf{T}_{n-1}-\dfrac{\mathbf{T}_{n-1}q_n\overline{\mathbf{h}}_n\overline{\mathbf{h}}_n^H\mathbf{T}_{n-1}}{1+q_n\overline{\mathbf{h}}_n^H\mathbf{T}_{n-1}\overline{\mathbf{h}}_n}, n>0.\end{cases} \tag{30}$$

The pseudocode of JEEPON is summarized in Algorithm 3, which does not require labels for the input datasets, i.e., the prior knowledge about user scheduling and beamforming design solutions. When the training of JEEPON is completed, the PDLF will not be needed and the optimization policy $\Phi(\mathcal{D}_i,\Theta)$ for smaple $\mathcal{D}_i\in\mathcal{D}$ can be obtained through L message passing layers and one PAC module (Steps 2-5). Note that JEEPON is trained offline with the aids of numerous training samples, and only simple mathematical operations are needed in the test stage. Therefore, its computational complexity is much lower than that of IOA.

Algorithm 3 JEEPON learning algorithm

Input: $\mathcal{G}_i(\mathcal{V},\mathcal{E})$: Wireless channel graph for sample $\mathcal{D}_i$;

L: Number of message passing layers;

Θ: The trainable network parameter set of JEEPON.

Output: An optimization policy $\Phi(\mathcal{D}_i,\Theta)\triangleq\{\boldsymbol{\kappa}^{(i)},\mathbf{q}^{(i)}\}$ for sample $\mathcal{D}_i$.

1: Initialize the node optimization vector: $\boldsymbol{\beta}_v^{(0)}, v\in\mathcal{V}$.

2: **for** layer $\ell\leftarrow 1,2,\cdots,L$ **do**

3: Update the node optimization vector $\boldsymbol{\beta}_v^{(\ell)}, v\in\mathcal{V}$ by (27).

4: **end for**

5: Obtain the policy $\Phi(\mathcal{D}_i,\Theta)\triangleq\{\boldsymbol{\kappa}^{(i)},\mathbf{q}^{(i)}\}$ from $\boldsymbol{\beta}_v^{(\ell)}, v\in\mathcal{V}$, and project it into the feasible region (28).

V. Numerical Results

In this section, numerical results are given for the joint optimization problem of user scheduling and beamforming design in the multiuser MISO downlink system. We first introduce the simulation environment and then evaluate the effectiveness of the proposed PDLF for training

JEEPON. The performance comparison of JEEPON, IOA and GHSA, and the generalizability test of JEEPON are also implemented. Finally, the computational complexities of these three algorithms are provided and discussed, where the speed advantage of JEEPON is clearly verified.

A. Description of Datasets and Parameters

In the experiment, we assume that the channel of the k-th user is modeled as $\mathbf{h}_k = \sqrt{\rho_k}\widetilde{\mathbf{h}}_k \in \mathbb{C}^{N\times 1}$ where $\widetilde{\mathbf{h}}_k \sim \mathcal{CN}(\mathbf{0}, \mathbf{1})$, and $\rho_k = 1/(1+(d_k/d_0)^{\varrho})$ is the long-term path-loss between user k and the BS. Here, d_k is the distance between the BS and k-th user, $d_0 = 50\mathrm{m}$ is the reference distance and $\varrho = 3$ indicates the path-loss exponent. As illustrated in Fig. 5, the K single-antenna users are randomly distributed in a range of $(d_{\min}, d_{\max})$ from the BS, here $d_{\min}, d_{\max} \in (d_0, d_r)$, and $d_r = 200\mathrm{m}$ is the cell radius. For simplicity, we assume that all users have the same additive noise variance, i.e., $\sigma_k^2 = \sigma^2, \forall k \in \mathcal{K}$, thus, the signal-to-noise ratio (SNR) is defined as $\mathrm{SNR} = 10\log_{10}\left(\frac{P}{\sigma^2}\right)$ in dB.

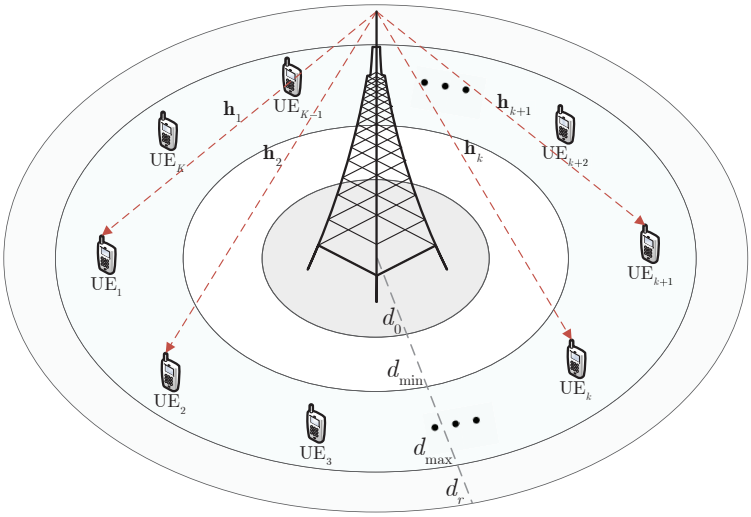


Fig. 5: User distribution of the system.

In the neural network module, two message passing layers and one PAC layer are used to form JEEPON and implemented by *Pytorch*. Significantly, the message construction function $M_\theta(\cdot)$ and state update function $U_\theta(\cdot)$ of each message passing layer are parameterized by MLPs with dimension sizes $\mathcal{M}_1$ and $\mathcal{M}_2$, respectively. Training and test stages for JEEPON are sequentially implemented. JEEPON is initialized by the Xavier initializer [34] before training and optimized by the adaptive moment estimation (Adam) optimizer [35] with the learning rate $\eta = 5\times 10^{-5}$, and

the update step-sizes ε_μ and ε_ν of Lagrangian multipliers are set to be 10^{-5}, respectively. In the training stage, PDLF is deployed to train JEEPON and the whole process lasts $N_{\mathrm{e}} = 400$ epochs. In the test stage, the trained JEEPON could be directly applied to the test dataset, which takes a wireless channel graph as input and outputs the related radio resource optimization policies. For each configuration, we report the average performance of JEEPON, IOA and GHSA on 500 test samples. In addition, the transmission data bits D, the transmission finite blocklength n and the decoding error probability ϵ for the communication system are also set to obtain SINR $\widetilde{\gamma}$ corresponding to the user's minimum achievable rate with following the conclusion in [11, Corollary 1]. Note that unless otherwise specified, the main parameters of the system adopt the default values in Table I.

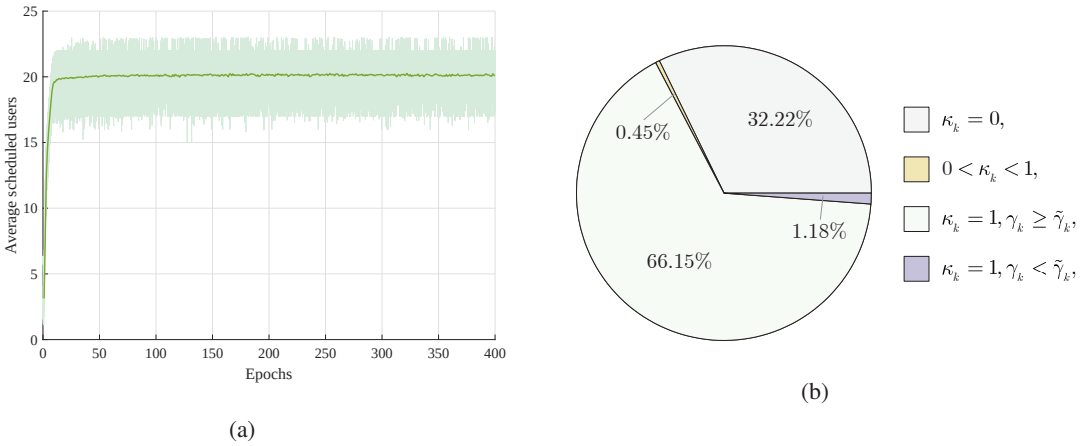
TABLE I: SYSTEM PARAMETERS.

Parameters	Values
Blocklength n	128
Decoding error probability ϵ	10^{-6}
Transmission data bits D	256 bits
BS antenna number N	32
Maximum permissible error δ	10^{-5}
Range of SNR	$0 \sim 30$ dB
Number of user K	$\{10, 20, 30, 40, 50\}$
Sizes of MLP in $M_\theta(\cdot)$	$\mathcal{M}_1 = \{7, 256, 128, 64, 32, 16, 4\}$
Sizes of MLP in $U_\theta(\cdot)$	$\mathcal{M}_2 = \{14, 256, 128, 64, 32, 16, 3\}$

B. Feasibility Analysis of PDLF

Two metrics are adopted to evaluate the feasibility of PDLF for training JEEPON. One is the convergence of the target value (23a) and the other is the proportion of users who satisfy the constraints (23c) and (23d). The former one reflects whether JEEPON could reach a steady state, and the latter is used to determine whether effective results are obtained by JEEPON. Fig. 6 illustrates the results with the system parameters $N_{\mathrm{s}} = 5000$, $K = 30$, $\mathrm{SNR} = 15\mathrm{dB}$, and $(d_{\min}, d_{\max}) = (50\mathrm{m}, 100\mathrm{m})$. In Fig. 6(a), the target value of different training samples varies in a certain range (light green line), while the average target value (solid green line) converges as the epoch increases to 20. Fig. 6(b) shows the proportion of users with various constraints on

5000 test samples. It is observed that the proportion of illegal results is 1.63%, which is much lower than the results satisfying the constraints. Note that in the test stage, the learned radio resource optimization policies may not fully satisfy the constraints for all samples due to the existence of neural network error. In the experiment, the value of $\kappa_k, k \in \mathcal{K}$ will be set to 1 if $0 < \kappa_k < 1$ is obtained, and all the scheduled users are filtered again with per-user minimum SINR requirement as stated in *Remark 2*.



(a) (b)

Fig. 6: The convergence curve of JEEPON, and user ratios in satisfying different constraints.

C. Performance Comparison and Generalizability Test

In this subsection, the performance of JEEPON, IOA and GHSA with different system parameters are evaluated and compared. For intuitive comparison, the obtained results of IOA and JEEPON are normalized by the results of GHSA and defined as $R_1 = \frac{N_{\mathrm{I}}}{N_{\mathrm{G}}} \times 100\%$ and $R_2 = \frac{N_{\mathrm{J}}}{N_{\mathrm{G}}} \times 100\%$, where N_{I}, N_{J} and N_{G} are the average number of scheduled users using IOA, JEEPON and GHSA, respectively.

1) Impact of Training Dataset Size: This experiment investigates how many training dataset samples are needed to learn a feasible JEEPON model. The training of JEEPON is implemented with the system parameters $N_{\mathrm{s}} \in \{1000, 2000, 3000, 4000, 5000\}$ (number of training sample), $\mathrm{SNR} = 10\mathrm{dB}$, $(d_{\min}, d_{\max}) = (60\mathrm{m}, 100\mathrm{m})$, and $K \in \{10, 20, 30, 40, 50\}$, and the results are summarized in Table II. From the table, the average performance of JEEPON improves as the training sample size increases and eventually reaches a steady state. It is conceivable that more training samples will not always bring a significant performance improvement and 5000 training samples are sufficient.

TABLE II: Average scheduled users with various training samples.

K	N_{G}	N_{J} with varying N_{s}				
		1000	2000	3000	4000	5000
10	8.922	8.106($\pm$0.12)	8.213($\pm$0.13)	8.317($\pm$0.09)	8.331($\pm$0.09)	8.334($\pm$0.06)
20	10.974	9.664($\pm$0.18)	9.815($\pm$0.12)	9.932($\pm$0.15)	10.083($\pm$0.12)	10.131($\pm$0.05)
30	12.068	10.383($\pm$0.19)	10.596($\pm$0.14)	10.782($\pm$0.12)	10.898($\pm$0.13)	11.012($\pm$0.11)
40	12.722	10.898($\pm$0.15)	11.083($\pm$0.16)	11.215($\pm$0.14)	11.385($\pm$0.11)	11.587($\pm$0.09)
50	13.228	11.109($\pm$0.20)	11.274($\pm$0.13)	11.498($\pm$0.15)	11.742($\pm$0.12)	11.925($\pm$0.10)

2) Performance with Various K and $(d_{\min}, d_{\max})$: This experiment investigates the influences of K and $(d_{\min}, d_{\max})$, where training and test datasets are generated with $\mathrm{SNR} = 10\mathrm{dB}$, and the obtained results are reported in Table III. From the table, it suggests that when K is small, the performance of JEEPON is closer to that of GHSA, because sufficient system resources are conducive to model optimization. JEEPON remains stable when K changes from 20 to 50, and there exist only 2.56% performance degradation at most. Besides, the performance gain of JEEPON improves with the distance interval changes from $20\mathrm{m}$ to $40\mathrm{m}$, since the smaller distance interval leads to the lack of diversity for each user, which brings more difficulties to the learning of JEEPON. In Fig. 7, we show the average performance of these three algorithms versus different $(d_{\min}, d_{\max})$, where $K = 30$ and $\mathrm{SNR} = 10\mathrm{dB}$. It suggests that JEEPON could achieve a more stable and closer performance compared with GHSA as $(d_{\min}, d_{\max})$ increases. Owing to the fact that the number of scheduled users is reduced with the increase of $(d_{\min}, d_{\max})$, and the obtained results are more homogeneous, which is beneficial to the learning of JEEPON.

TABLE III: Average performance normalized by GHSA with various K.

K	R_1 and R_2 with varying $(d_{\min}, d_{\max})$							
	(50m, 70m)		(60m, 80m)		(50m, 90m)		(60m, 100m)	
	R_1	R_2	R_1	R_2	R_1	R_2	R_1	R_2
10	100%	95.68%	99.98%	94.70%	100%	93.94%	99.89%	93.41%
20	99.67%	90.04%	99.64%	91.35%	99.52%	92.02%	99.22%	92.32%
30	99.94%	89.68%	99.63%	90.33%	98.80%	90.57%	98.77%	91.25%
40	99.86%	88.91%	99.54%	89.86%	98.52%	90.27%	98.24%	91.08%
50	99.84%	88.15%	98.73%	88.79%	97.48%	89.46%	97.10%	90.15%

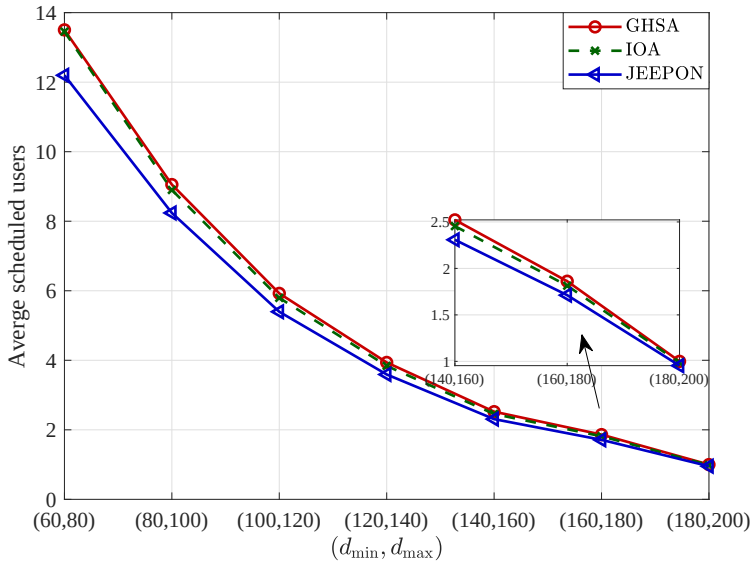


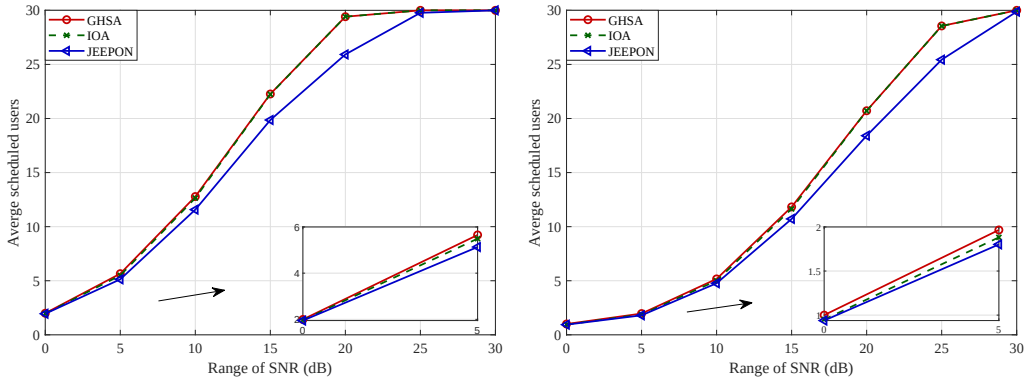
Fig. 7: Average number of scheduled users with various $(d_{\min}, d_{\max})$.

3) Performance with Various SNR Settings: This experiment compares the performance of JEEPON, IOA and GHSA with different SNR settings, and the obtained results are summarized in Table IV. It is observed that JEEPON achieves competitive performance (larger than 90.77%) with $\mathrm{SNR} = 5\mathrm{dB}$, while IOA maintains over 95.73% near-optimal performance compared with GHSA. Although the performance gap of JEEPON is enlarged with K increase, the trend of degradation is rather slow. For the configuration $\mathrm{SNR} = 15\mathrm{dB}$ and $(d_{\min}, d_{\max}) = (50\mathrm{m}, 100\mathrm{m})$, JEEPON obtains only a 1.78% relative performance gap with GHSA when K changes from 20 to 50. Even when $(d_{\min}$,$d_{\max}) = (100\mathrm{m}, 150\mathrm{m})$, JEEPON still maintains a stable performance. Moreover, Fig. 8 illustrates the average performance of these three algorithms in scenarios with different SNR settings. From Fig. 8(a), with the increase of SNR, the performance of JEEPON is worse than GHSA, but the result is not less than 88.13% ($\mathrm{SNR} = 20\mathrm{dB}$) and it finally achieves near-optimal performance due to abundant transmitting resources. Fig. 8(b) also shows similar results, but the number of scheduled users is reduced when $\mathrm{SNR} \leq 25\mathrm{dB}$ due to the higher resource requirements (farther distance) for a single user.

4) Performance with Various SINR Requirements: The ultimate scheduling results of the investigated problem are significantly affected by the per-user minimum SINR requirement, where value $\widetilde{\gamma}$ is obtained through the system parameters D, n, and ϵ. To this end, we test the performance of the proposed algorithms under the different (D, n, ϵ) with $K = 30$ and

TABLE IV: Average performance normalized by GHSA with various SNR settings.

SNR(dB)	K	R_1 and R_2 with varying $(d_{\min},d_{\max})$							
		(60m, 90m)		(90m, 120m)		(50m, 100m)		(100m, 150m)	
		R_1	R_2	R_1	R_2	R_1	R_2	R_1	R_2
5	10	98.96%	92.13%	98.43%	94.50%	99.37%	92.56%	97.74%	93.17%
	20	98.22%	91.08%	97.82%	92.43%	98.82%	91.15%	96.59%	92.03%
	30	97.17%	90.77%	96.58%	91.04%	97.75%	90.83%	95.73%	91.66%
15	20	99.99%	90.15%	99.60%	90.69%	100%	90.48%	99.17%	91.15%
	30	99.87%	89.20%	99.61%	89.78%	99.97%	89.52%	98.46%	90.60%
	50	99.64%	88.34%	99.55%	89.06%	99.80%	88.70%	97.36%	89.94%



(a) $K = 30$, $(d_{\min},d_{\max}) = (60\text{m}, 90\text{m})$ (b) $K = 30$, $(d_{\min},d_{\max}) = (100\text{m}, 150\text{m})$

Fig. 8: Average performance of JEEPON, IOA and GHSA with various SNR settings.

$\text{SNR} = 10\text{dB}$, and the obtained results are summarized in Table V. From the table, It is observed that the average performance of JEEPON remains above 88.97% compared with GHSA under different SINR requirements and user distribution distances. However, one needs to point out that with the reduction of SINR requirements, the performance improvement of JEEPON is lower than GHSA, especially when $(d_{\min},d_{\max}) = (60\text{m}, 80\text{m})$. Therefore, JEEPON shows a slight performance degradation compared with GHSA when the SINR requirement is reduced, while the performance improvement of IOA increases at the same time.

5) Generalizability with Various User Distributions: Generalizability is another critical evaluation property for JEEPON, and it focuses on investigating whether the trained network model has the ability to perform well in unknown wireless network scenarios. To test the generalizability, the proposed JEEPON is trained from a certain scenario whose system parameters

TABLE V: Average performance normalized by GHSA with various SINR requirements.

(D, n, ϵ)	$\widetilde{\gamma}$	R_1 and R_2 with varying $(d_{\min}$,$d_{\max})$							
		$(60\text{m}, 80\text{m})$		$(80\text{m}, 100\text{m})$		$(60\text{m}, 100\text{m})$		$(80\text{m}, 120\text{m})$	
		R_1	R_2	R_1	R_2	R_1	R_2	R_1	R_2
$(256, 256, 10^{-6})$	1.633	99.92%	88.97%	99.96%	90.36%	99.97%	89.82%	99.91%	91.03%
$(256, 128, 10^{-6})$	5.054	99.63%	90.33%	98.30%	91.87%	98.77%	91.25%	98.36%	92.78%
$(256, 96, 10^{-6})$	9.291	96.41%	90.79%	94.22%	92.94%	95.84%	91.84%	95.38%	93.62%
$(256, 64, 10^{-6})$	27.97	95.58%	91.05%	93.95%	93.05%	94.55%	92.76%	94.19%	94.08%

are different from the test ones. Specifically, JEEPON is trained with $\text{SNR} = 10\text{dB}$ and $(d_{\min}, d_{\max}) = (100\text{m}, 120\text{m})$, then the trained model is applied to the test scenarios with different $(d_{\min}$,$d_{\max})$, without any further training. Table VI shows comparison results of GHSA and JEEPON, where $R_3 = \frac{N_{\text{J},(100,120)}}{N_{\text{G}}} \times 100\%$ represents the average performance of JEEPON normalized by GHSA and $N_{\text{J},(100,120)}$ is the average number of scheduled users using JEEPON. Form the table, it is observed that JEEPON performs well over the neighboring user distribution distances when the test distance interval is 40m, and even up to 60m is feasible. Moreover, when the user distribution distance is $(80\text{m}, 100\text{m})$ which has no intersection with $(100\text{m}, 120\text{m})$, the performance of JEEPON is still acceptable at $K = 10$. Based on the aforementioned analysis, our proposed JEEPON can be well generalized to scenarios with neighboring user distribution distances if the number of users is fixed.

TABLE VI: Generalizability with various user distributions.

K	N_{G} and R_3 with varying $(d_{\min}$,$d_{\max})$											
	$(100\text{m}, 120\text{m})$		$(80\text{m}, 100\text{m})$		$(80\text{m}, 120\text{m})$		$(80\text{m}, 140\text{m})$		$(100\text{m}, 140\text{m})$		$(100\text{m}, 160\text{m})$	
	N_{G}	R_3	N_{G}	R_3	N_{G}	R_3	N_{G}	R_3	N_{G}	R_3	N_{G}	R_3
10	5.068	94.97%	7.504	86.06%	6.36	91.86%	5.578	92.58%	4.394	92.13%	3.89	90.64%
20	5.624	93.92%	8.548	84.63%	7.496	89.58%	6.852	90.42%	5.068	91.46%	4.664	89.02%
30	5.924	92.69%	9.054	83.34%	8.16	88.31%	7.562	89.74%	5.372	88.16%	5.036	87.83%
40	6.038	91.24%	9.304	82.75%	8.508	87.92%	8.058	90.85%	5.608	87.73%	5.301	85.13%
50	6.15	90.80%	9.538	83.04%	8.818	85.87%	8.29	88.73%	5.782	86.09%	5.444	82.62%

D. Computational Complexity Analysis

In this subsection, the computational complexities of JEEPON, IOA and GHSA are analyzed and compared. For GHSA, it adds a suitable user to the scheduled user set $\mathcal{S}$ each time when the system radio resources are not exhausted. We assume that the current number of scheduled users is $\hat{k}$, then the computational complexity of the next iteration is $\mathcal{O}((K-\hat{k}+1)It_1(N\hat{k}^3+N\hat{k}^2))$, where It_1 represents the number of iterations until the algorithm converges. Therefore, the computational complexity of GHSA is $\mathcal{O}(\sum_{\hat{k}=2}^{K}(K-\hat{k}+1)It_1(N\hat{k}^3+N\hat{k}^2))$. For IOA, we assume that the number of iterations in the two convergence processes are It_2 and It_3, respectively. Specifically, the computational complexity of the inner iterative process is $\mathcal{O}(It_2(NK^2))$ and the outer iterative process is $\mathcal{O}(It_3(It_2(NK^2)+KN^3))$, which is treated as the complexity of IOA. Since the complexity of GHSA is proportional to the cube of K, the convergence speed of IOA will be significantly faster than GHSA as K increases.

As for JEEPON, only the complexity of test stage is considered, which mainly includes three modules: the feature design module, the neural network module and the MIC module. Specifically, the computational complexity of the feature design module is $\mathcal{O}(NK^2)$, and the computational complexity of the MIC module is $\mathcal{O}(KN^2)$. For simplicity, we assume that each message passing layer of the neural network module is composed of the same two MLP models with $|\mathcal{M}|$ hidden layers, and denote the dimension size of the j-th hidden layer as $m_j \in \mathcal{M}$. Therefore, the computational complexity of the neural network module can be expressed as $\mathcal{O}(2K\sum_{\ell=1}^{L}\sum_{j=1}^{|\mathcal{M}|}(2+m_{\ell,j-1})m_{\ell,j})$. Based on the above analysis, the computational complexity of JEEPON is $\mathcal{O}(NK^2+KN^2+2K\sum_{\ell=1}^{L}\sum_{j=1}^{|\mathcal{M}|}(2+m_{\ell,j-1})m_{\ell,j})$, and the lower bound of complexity is $\mathcal{O}(NK^2+KN^2)$. Obviously, the overall computational complexity of JEEPON is even significantly lower than the partial computational complexity of IOA and GHSA, which reflects the computational speed advantage of JEEPON.

VI. Conclusions

In this paper, we studied the joint optimization problem of user scheduling and beamforming design in the multiuser MISO downlink system. Specifically, we proposed two optimization algorithms, one is the IOA by means of gradient descent and successive convex approximation, and the other is the JEEPON, a novel GNN-based learning model for fast performing user scheduling and beamforming design. Furthermore, in order to alleviate the computation burden,

both two algorithms perform power allocation instead of directly beamforming design through the uplink-downlink duality theory. In JEEPON, we model the K-user downlink network as a fully-connected graph, and learn the radio resource management on graphs in an unsupervised manner. Numerical results suggests that the performance of JEEPON is close to IOA and GHSA, while enjoying a lower computational complexity. Additionally, the proposed JEEPON shows a potential generalization ability in dynamic wireless network scenarios.

REFERENCES

[1] Y. Xu, G. Gui, H. Gacanin, and F. Adachi, "A survey on resource allocation for 5G heterogeneous networks: Current research, future trends, and challenges," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 2, pp. 668–695, Apr. 2021.

[2] S. He, Y. Zhang, J. Wang, J. Zhang, J. Ren, Y. Zhang, W. Zhuang, and X. Shen, "A survey of millimeter-wave communication: Physical-layer technology specifications and enabling transmission technologies," *Proc. IEEE*, vol. 109, no. 10, pp. 1666–1705, Oct. 2021.

[3] S. He, Y. Huang, L. Yang, B. Ottersten, and W. Hong, "Energy efficient coordinated beamforming for multicell system: Duality-based algorithm design and massive MIMO transition," *IEEE Trans. Commun.*, vol. 63, no. 12, pp. 4920–4935, Dec. 2015.

[4] X. Yu, Y. Du, X. y. Dang, S. H. Leung, and H. Wang, "Power allocation schemes for uplink massive MIMO system in the presence of imperfect CSI," *IEEE Trans. Signal Process.*, vol. 68, pp. 5968–5982, Oct. 2020.

[5] H. A. Ammar, R. Adve, S. Shahbazpanahi, G. Boudreau, and K. V. Srinivas, "Distributed resource allocation optimization for user-centric cell-free MIMO networks," *to be published in IEEE Trans. Wireless Commun.*, 2021, DOI:10.1109/TWC.2021.3118303.

[6] Q. Shi, M. Razaviyayn, Z. Q. Luo, and C. He, "An iteratively weighted MMSE approach to distributed sum-utility maximization for a MIMO interfering broadcast channel," *IEEE Trans. Signal Process.*, vol. 59, no. 9, pp. 4331–4340, Sept. 2011.

[7] G. Dong, H. Zhang, S. Jin, and D. Yuan, "Energy-efficiency-oriented joint user association and power allocation in distributed massive MIMO systems," *IEEE Trans. Veh. Technol.*, vol. 68, no. 6, pp. 5794–5808, June. 2019.

[8] C. E. Shannon, "A mathematical theory of communication," *Bell Syst. Technol. J.*, vol. 27, no. 3, pp. 379–423, Jul. 1948.

[9] A. A. Nasir, H. D. Tuan, H. H. Nguyen, M. Debbah, and H. V. Poor, "Resource allocation and beamforming design in the short blocklength regime for URLLC," *IEEE Trans. Wireless Commun.*, vol. 20, no. 2, pp. 1321–1335, Feb. 2020.

[10] Y. Polyanskiy, H. V. Poor, and S. Verdú, "Channel coding rate in the finite blocklength regime," *IEEE Trans. Inf. Theory*, vol. 56, no. 5, pp. 2307–2359, May 2010.

[11] S. He, Z. An, J. Zhu, J. Zhang, Y. Huang, and Y. Zhang, "Beamforming design for multiuser uRLLC with finite blocklength transmission," *to be published in IEEE Trans. Wireless Commun.*, 2021, DOI:10.1109/TWC.2021.3090197.

[12] D. Bertsekas, A. Nedic, and A. Ozdaglar, *Convex analysis and optimization*. Athena Scientific, 2003.

[13] C. She, C. Sun, Z. Gu, Y. Li, C. Yang, H. V. Poor, and B. Vucetic, "A tutorial on ultrareliable and low-latency communications in 6G: Integrating domain knowledge into deep learning," *Proc. IEEE*, vol. 109, no. 3, pp. 204–246, Mar. 2021.

[14] F. Liang, C. Shen, W. Yu, and F. Wu, "Towards optimal power control via ensembling deep neural networks," *IEEE Trans. Commun.*, vol. 68, no. 3, pp. 1760–1776, Mar. 2020.

[15] Y. Li, S. Han, and C. Yang, "Multicell power control under rate constraints with deep learning," *to be published in IEEE Trans. Wireless Commun.*, 2021, DOI:10.1109/TWC.2021.3088224.

[16] W. Xia, G. Zheng, Y. Zhu, J. Zhang, J. Wang, and A. P. Petropulu, "A deep learning framework for optimization of MISO downlink beamforming," *IEEE Trans. Commun.*, vol. 68, no. 3, pp. 1866–1880, Mar. 2020.

[17] T. Chen, X. Zhang, M. You, G. Zheng, and S. Lambotharan, "A GNN based supervised learning framework for resource allocation in wireless IoT networks," *to be published in IEEE Internet Things J.*, 2021, DOI:10.1109/JIOT.2021.3091551.

[18] S. He, S. Xiong, Y. Ou, J. Zhang, J. Wang, Y. Huang, and Y. Zhang, "An overview on the application of graph neural networks in wireless networks," *IEEE Open J. Commun. Soc.*, vol. 2, no. 2, pp. 2547–2565, Dec. 2021.

[19] W. Cui, K. Shen, and W. Yu, "Spatial deep learning for wireless scheduling," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 6, pp. 1248–1261, Mar. 2019.

[20] M. Eisen and A. Ribeiro, "Optimal wireless resource allocation with random edge graph neural networks," *IEEE Trans. Signal Process.*, vol. 68, pp. 2977–2991, Apr. 2020.

[21] Y. Shen, Y. Shi, J. Zhang, and K. B. Letaief, "A graph neural network approach for scalable wireless power control," in *2019 IEEE Globecom Workshops (GC Wkshps)*. IEEE, Dec. 2019, pp. 1–6.

[22] N. Keriven and G. Peyré, "Universal invariant and equivariant graph neural networks," in *33rd Conf. Neural Inf. Process. Syst.*, Dec. 2019, pp. 7092–7101.

[23] 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. 2020.

[24] S. He, S. Xiong, W. Zhang, Y. Yang, J. Ren, and Y. Huang, "GBLinks: GNN-based beam selection and link activation for ultra-dense D2D mmWave networks," *arXiv preprint arXiv:2107.02412*, Jul. 2021.

[25] T. Zhang, "Adaptive forward-backward greedy algorithm for learning sparse representations," *IEEE Trans. Inf. Theory*, vol. 57, no. 7, pp. 4689–4708, Jul. 2011.

[26] M. Schubert and H. Boche, "Solution of the multiuser downlink beamforming problem with individual SINR constraints," *IEEE Trans. Veh. Technol.*, vol. 53, no. 1, pp. 18–28, Jan. 2004.

[27] S. He, Y. Wu, D. W. K. Ng, and Y. Huang, "Joint optimization of analog beam and user scheduling for millimeter wave communications," *IEEE Commun. Lett.*, vol. 21, no. 12, pp. 2638–2641, Dec. 2017.

[28] E. Che, H. D. Tuan, and H. H. Nguyen, "Joint optimization of cooperative beamforming and relay assignment in multi-user wireless relay networks," *IEEE Trans. Wireless Commun.*, vol. 13, no. 10, pp. 5481–5495, 2014.

[29] K. Nguyen, L. Tran, O. Tervo, Q. Vu, and M. Juntti, "Achieving energy efficiency fairness in multicell MISO downlink," *IEEE Commun. Lett.*, vol. 19, no. 8, pp. 1426–1429, Aug. 2015.

[30] F. Fioretto, T. W. Mak, and P. Van Hentenryck, "Predicting AC optimal power flows: Combining deep learning and lagrangian dual methods," in *Proc. AAAI Conf. Artif. Intell.* AAAI, Apr. 2020, pp. 630–637.

[31] M. Lee, G. Yu, and G. Y. Li, "Graph embedding-based wireless link scheduling with few training samples," *IEEE Trans. Wireless Commun.*, vol. 20, no. 4, pp. 2282–2294, Apr. 2021.

[32] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl, "Neural message passing for quantum chemistry," in *Int. Conf. Mach. Learn.(ICML)*. PMLR, Aug. 2017, pp. 1263–1272.

[33] J. Sherman and W. J. Morrison, "Adjustment of an inverse matrix corresponding to a change in one element of a given matrix," *The Ann. Math. Statist.*, vol. 21, no. 1, pp. 124–127, Mar. 1950.

[34] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. 13th Int. Conf. Artif. Intell. Statist.* PMLR, May 2010, pp. 249–256.

[35] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Int. Conf. Learn. Represent.(ICLR)*, May 2015.