Received 25 February 2024; accepted 22 March 2024. Date of publication 1 April 2024; date of current version 19 April 2024.

Digital Object Identifier 10.1109/OJCOMS.2024.3384110

Unsupervised Learning for Resource Allocation and User Scheduling in Wideband MU-MIMO Systems

CHIH-HO HSU AND ZHI DING (Fellow, IEEE)

Department of Electrical and Computer Engineering, University of California, Davis, CA 95616, USA

CORRESPONDING AUTHOR: Z. DING (e-mail: zding@ucdavis.edu)

This work was supported by the National Science Foundation under Grant 2029027 and Grant 2332760.

ABSTRACT By leveraging spatial diversity, MultiUser MIMO (MU-MIMO) can serve multiple users over shared time-frequency Resource Blocks (RBs) and substantially improve spectral efficiency. However, performances of wideband MU-MIMO systems are severely limited by both frequency-selective channels and Co-Channel Interference (CCI) among users. To reach the full potential of MU-MIMO, users should be scheduled at RBs with decent channel gains and minimal CCIs. Since such scheduling problem is NP-hard and the transmission time interval of modern wireless systems is ultra-short, it is critical to design efficient algorithms that can make satisfactory sub-optimal user scheduling decisions in real-time. Nonetheless, existing works either rely on heuristics or may not readily be applied to wideband system. To tackle these challenges, we propose a novel Unsupervised Learning-Aided Wideband Scheduling (ULAWS) framework. Specifically, ULAWS first utilizes Multi-Dimensional Scaling (MDS) based graph embedding and clustering to obtain intrinsic user groups with low CCI among co-channel users. Based on clustering results, we adopt Gale-Sharpley algorithm to find a stable matching between users and RBs. Next, a graph-based post-processing procedure stacked with three efficient steps is applied as refinement. Simulation results demonstrate performance gain over benchmark methods in terms of sum rate, fairness and outage rate, under various system parameters and scenarios.

INDEX TERMS Multiuser MIMO (MU-MIMO), co-channel interference (CCI), graph embedding, user grouping, wideband user scheduling.

I. INTRODUCTION

MULTIPLE-INPUT-MULTIPLE-OUTPUT (MIMO) technologies, which serve multiple users with a large number of antennas, have served as an effective solution to meet the increasing demand for emerging high-data-rate applications in both WiFi and cellular wireless systems [1]. By exploiting spatial diversity gain through multiple antennas, MIMO improves system capacity and Spectral Efficiency (SE) [2]. Based on MIMO, MultiUser MIMO (MU-MIMO) enables an Access Point (AP) or base station (BS) to simultaneously serve multiple users on the same time-frequency Resource Block (RB) [3] and hence can further improve system performance.

However, Co-Channel Interference (CCI) among users scheduled to the same Resource-Sharing Group (RSG) becomes a major obstacle to reaching the promised performance of MU-MIMO systems. Specifically, the efficacy of MU-MIMO depends on the level of CCI among the multiple RSGs, each occupying an RB. Because high spatial channel correlation among co-channel users [4] implies strong CCI, the capacity of MU-MIMO systems can be significantly improved by scheduling users with low mutual correlation to the same RSG. Therefore, MU-MIMO performance critically relies on user scheduling [4] and resource allocation [5].

To reduce CCI among users, MU-MIMO user scheduling algorithms should group users into RSGs across accessible RBs such that within each RSG, users enjoy low channel correlation [6]. However, the number of possible RSGs in such combinatorial problems increases exponentially with the number of users, thereby making exhaustive search and evaluation prohibitively costly even for a moderate number

of users [7]. In addition, due to channel variation and the ultra-short Transmission Time Interval (TTI) in current and future generation wireless systems [1], it is vital to design efficient algorithms that find and form RSGs with low CCI in real-time.

To mitigate the computation complexity for obtaining such RSGs, previous works have used channel correlation among users in a greedy [8], [9], [10], [11] or heuristic [12], [13] manner, where users are scheduled one by one to existing RSGs according to pre-defined criteria. However, these and other existing heuristic methods tend to favor locally optimum solutions. To overcome such shortcomings, several studies examined two-step unsupervised learning approaches to extract common features of user channel correlations [14], [15], [16], [17], [18]. These strategies first cluster users with similar channels and then separate users in the same cluster into different RSGs based on the insight that it is easier to cluster users with similar channels than to cluster users with dissimilar channels, (e.g., users with low channel correlation). Although these unsupervised strategies do not directly minimize CCI among users within each RSG, they lead to good performance for MU-MIMO systems by preventing high CCI users from being scheduled to the same RSG.

Nonetheless, to our best knowledge, no unsupervised learning-based scheduling algorithms have considered broader scheduling problems involving wideband (or multi-channel) systems. In fact, it is highly challenging to extend most existing works to wideband systems, such as the multi-subcarrier (multi-subband) physical layers of 5G-NR cellular networks or WiFi6/WiFi7, where channels can be frequency-selective due to multi-path effects. Namely, not only channel gains of users but also pair-wise channel correlations among users are different in each subband, which further exacerbates the design complexity of user scheduling algorithms. Therefore, it remains an open but critical problem on how to improve the sum rate and SE of wideband MU-MIMO systems by forming RSGs in consideration of both users' channel gain and channel correlation across a swath of subbands.

Motivated by the aforementioned challenges, our previous work [19] aims to (1) form RSGs by clustering users with dissimilar channels into the same RSG such that group users enjoy low CCI, and (2) form RSGs in a wideband (i.e., frequency-selective multi-channel) system by considering users' channel correlation over each subband. These considerations lead to an effective scheduling algorithm named SC-MS [19]. Specifically, SC-MS achieves (1) by applying spectral clustering and post-processing on the channel dissimilarity graph of users and achieves (2) with a heuristic procedure.

Yet, several shortcomings remain in SC-MS. First, it relies on heuristics to form RSGs in different subbands without a global optimization perspective. Second, SC-MS forms RSGs merely based on channel dissimilarity among users (i.e., focusing on CCI) without considering the effect of channel gain on the sum rate. Third, spectral clustering could incorrectly cluster users with high channel correlation into the same group due to *imperfect graph embedding* in Euclidean space, thereby lowering the sum rate.

In this paper, we improve three mentioned drawbacks of SC-MS and propose an Unsupervised Learning Aided Wideband Scheduling (ULAWS) algorithm that aims to schedule users and allocate RBs based on user channel characteristics such that the sum rate of MU-MIMO systems is maximized. Specifically, our main contributions are as follows:

1) To the best of our knowledge, this work is the first to propose a unifying and feasible user scheduling solution for wideband MU-MIMO systems with frequency-selective channel gains and channel correlations.

2) To address the first two shortcomings of SC-MS, we formulate the scheduling of users to RBs as a many-to-one matching game where each RB may host multiple MU-MIMO users whereas user preference over an RB is defined as its SINR estimated based on its channel gain and CCI from other users. In this way, we jointly consider users' global channel characteristics, including channel gain and CCI, across different subbands in our scheduling mechanism instead of relying on the heuristic as in [19].

3) We developed an unsupervised learning framework to estimate user CCI over each RB. Specifically, we first capture user channel correlation in each subband with a graph structure. We then perform graph embedding and clustering to determine intrinsic user groups in the constructed graphs to keep channel correlation among users in the same group low. We treat users within the same intrinsic group as potential co-channel users to estimate their CCI, which is then used to estimate their SINR.

4) Since users are matched with RBs based on estimated SINR instead of actual SINR, our matching result may still fall victim to local optimum and should be treated as a preliminary scheduling decision. To mitigate performance loss, we propose a graph-based post-processing procedure stacked with three efficient steps, including finding cliques, node reassignment and iterative refinement, to further improve the sum rate.

5) To tackle the third drawback of SC-MS, we improve graph embedding quality by formulating it as a Multidimensional Scaling (MDS) problem aiming to preserve the exact structure of graphs in Euclidean space. To overcome MDS's weakness in embedding dense and random graphs in our problem, we design a universal transformation function that maps arbitrary graphs' weights to a negatively skewed target distribution.

II. RELATED WORKS

In this section, we review existing works on user scheduling problems in MU-MIMO systems.

A. CHANNEL CORRELATION BASED GREEDY/HEURISTIC

As the performance of MU-MIMO is limited by CCI among users, pair-wise channel correlation is utilized greedily [8], [9], [10], [11] or heuristically [12], [13] to mitigate CCI.

For example, the authors of [8] designed a scheduling algorithm that first separates users into different RSGs when their channel correlation is greater than a predefined threshold and then iteratively assigns remaining users to RSGs with the minimum sum of channel correlation with existing users. In [10], they iteratively select a user with the highest SNR and assign it to the RSG that has the minimum of the maximum channel correlation with existing users. A similar approach is adopted in [9] yet they instead adopt increased throughput as the metric.

On the other hand, the authors of [12] modeled interference levels between users as edge weights in a graph and adopted a heuristic procedure that sequentially assign users to existing RSGs with the minimum increased sum of graph weights. In [13], they create a RSG by selecting and removing users (nodes) that belong to the maximum clique from the channel dissimilarity graph. The process is repeated until the graph becomes empty, creating a varying number of RSGs.

Although these approaches can efficiently reduce CCI by utilizing users' channel correlation, they in general tend to favor local optimum.

B. MANY-TO-ONE MATCHING WITH EXTERNALITY

In [21], [22], [23], user scheduling problems are formulated as many-to-one or many-to-many matching games. Since CCI exists among users in the same RSGs, an ideal algorithm should decide not only which RB the users are matched with, but also which users are matched in the same RB, and thus externality exists. Hence, previous works often design two-step algorithms to address such matching games. The first step creates an initial matching between users and RBs randomly [22], [23] or based on the users' preference list [21] that does not consider the effect of CCI from other users. Then, the second step iteratively searches for a blocking pair and swaps the matching of the corresponding user pair. The algorithm terminates when there exists no swap-blocking pair in the matching.

However, it may take numerous iterations for these methods to converge when the initial matching step is not well designed. For instance, the worst-case complexity of [21] is $\mathcal{O}(\frac{\Phi_{final}-\Phi_0}{\Delta})$ where Φ_0 and Φ_{final} denote the sum utilities of the initial matching and the final matching, and Δ is the minimum increase of each swap operation [21].

C. UNSUPERVISED LEARNING-AIDED USER SCHEDULING

Instead of assigning users to RBs based on greedy criteria or swapping the blocking user pairs in the matching, the authors of [14], [16], [17], [18] proposed two-step strategies to group users into RSGs based on their channel correlations using unsupervised clustering. For instance, the first step in [17] clusters users with highly correlated channels (i.e., users with high CCI) with K-means. The second step iteratively selects a RSG and schedules the user with the maximum rate increment and does not belong to any clusters that users of the selected RSG belong to. Similarly, the authors of [16] adopt Grassmannian K-means [16] to better cluster users with similar channels before iteratively assigning users from the same cluster to different RSGs with a rule-based approach to obtain the final scheduling decision. In this way, users with high CCI can be separated into different RSGs.

However, the path to extend these methods to a practical wideband scheduling problem is unclear. First, the number of generated RSGs would vary depending on the number of users with highly correlated channels while in practice we usually have a fixed number of RBs to accommodate users. Secondly, they focus only on channel correlation among users in the narrowband system while in wideband systems we have multiple subbands/channels with distinct channel characteristics.

A comparison of related work is listed in Table 1. We classify them according to the methodology used, whether they can schedule users to a given set of RBs, whether they jointly consider the effect of channel gain and channel correlation in their approach, and whether their approach can be extended to wideband/multi-channel systems. We observe that the limitation of matching-based methods is that they did not fully utilize problem-specific information including users' channel gain and channel correlation in their mechanism. On the other hand, existing clustering learning-based methods can not be readily applied to MU-MIMO systems with a fixed number of RBs and multiple frequency-selective channels.

Notations: Throughout this paper, we use lowercase bold letters for vectors, uppercase bold letters for matrices, calligraphic capital letters to denote sets and graphs, and non-bold font for scalars and functions. We use $(\cdot)^{\mathbf{T}}$, $\overline{(\cdot)}$ and $(\cdot)^{\mathbf{H}}$ to denote transpose, conjugate and conjugate transpose, respectively. We follow general conventions and summarize some special notations used in this paper in Table 2.

III. SYSTEM MODEL AND PROBLEM FORMULATION

A. MU-MIMO SYSTEM AND SUM RATE

Without loss of generality, we consider a single-cell OFDMA communication system where a BS (e.g., gNB) equipped with L antennas serves U single-antenna users. MU-MIMO in downlink transmission can accommodate multiple users in the same time-frequency RB[1] by leveraging spatial diversity brought by MIMO. Specifically, the operational spectrum is divided into F equal subbands. Across a schedule duration of T timeslots, there are FT RBs in total. We further denote $\mathrm{RB}(f,t)$ as the RB indexed by subband f and timeslot t,

[1]A time-frequency RB comprises one timeslot and one subband. Note that our RB is broadly defined as a resource unit occupied by users in an RSG and does not have to exactly match the PRB in 3GPP LTE or 5G.

TABLE 1. Literature review.

Ref #	Methodology	Fixed RBs	Joint CH gain & CCI	Multi-channels	Year
[20]	Meta heuristic (GA)	Yes	No	Yes	2018
[21]	Swap-matching	Yes	No	Yes	2017
[22]	Swap-matching	Yes	No	Yes	2017
[23]	Swap-matching	Yes	No	Yes	2019
[8]	Greedy (sum of CCI)	No	No	No	2016
[10]	Greedy (throughput)	Yes	No	Yes	2019
[9]	Greedy (CH gain, min-max CCI)	Yes	Yes	Yes	2017
[11]	Greedy (threshold)	No	No	No	2023
[12]	Graph heuristic (sum of weights)	Yes	Yes	Yes	2018
[13]	Graph heuristic (max clique)	No	No	No	2023
[15]	Clustering (K-means) + greedy	Yes	No	No	2018
[16]	Clustering (K-means) + greedy	No	No	No	2023
[17]	Clustering (K-means) + greedy	No	No	No	2019
[18]	Clustering (K-means) + greedy	No	No	No	2019
[19]	Clustering + graph heuristic	Yes	No	Yes	2023
ULAWS	Clustering + matching + graph heuristic	Yes	Yes	Yes	2023

Note: **CH** = Channel, **CCI** = Co-Channel Interference, **GA** = Genetic Algorithm,

TABLE 2. List of main notations.

Notation	Definition
$\boldsymbol{H}, \boldsymbol{X}$	3D CSI array / 3D scheduling decision array
$\boldsymbol{h}_i^f$	CSI vector of user i at subband f
$x_i^{f,t}$	Indicator denoting if user i is scheduled to RB (f,t)
$\mathcal{B}, \mathcal{F}, \mathcal{T}$	The set of resource blocks, subbands and timeslots
U, F, T, L	Number of users, subbands, timeslots and antennas
$\gamma_i^{f,t}$	SINR experienced by user i on RB(f,t)
$\rho(\boldsymbol{h}_i^f, \boldsymbol{h}_j^f)$	CSI correlation between $\boldsymbol{h}_i^f$ and $\boldsymbol{h}_j^f$ at subband f
$R_i^{f,t}, \widetilde{R}_i^{f,t}$	Achievable/effective data rate of user i at RB(f,t)
$\boldsymbol{z}_i^f$	Embedding vector of user i at subband f
$d_{i,j}^f$	Euclidean distance between z_i^f and z_j^f
$w_{i,j}^f / w_{i,j}^{f,t}$	Weight among users i and j in CSI *correlation* graph of subband f. / Weight among users i and j in CSI *dissimilarity* graph of RB (f,t)
$\boldsymbol{E}_{f,t}^{dis}$	Degree matrix in the CSI *dissimilarity* graph at RB(f,t)
$\Gamma_i^{f,t}$	Estimated CCI that user i may experience in RB(f,t)
$\Upsilon_i^{US}(f,t)$	The preference of a user i over RB (f,t)
$\Upsilon_{f,t}^{RB}(i)$	The preference of a RB (f,t) over user i
η	Matching function between users and RBs
ψ	Weights' transformation function for graph embedding

$\forall f \in \mathcal{F} = \{1, \ldots, F\}$ and $\forall t \in \mathcal{T} = \{1, \ldots, T\}$. The set of RBs is presented as $\mathcal{B} = \{(f,t) | f \in \mathcal{F}, t \in \mathcal{T}\}$.

We let $\boldsymbol{H} = \{h_{i,\ell}^f\}_{i,\ell,f} \in \mathbb{C}^{U \times L \times F}$ denote the channel state information (CSI) 3D-array. Each element $h_{i,\ell}^f \in \mathbb{C}$, $\forall i \in \mathcal{U} = \{1, \ldots, U\}$, $\forall \ell \in \mathcal{L} = \{1, \ldots, L\}$, $\forall f \in \mathcal{F}$ denotes the CSI between single antenna user i and BS's antenna ℓ on subband f. Without limiting user CSI to any channel model, here we consider a general case for which CSIs are random and independent among users, subbands and BS antennas. We further assume quasi-static block fading channels over the T transmission timeslots, below channel coherence time.

Without loss of generality, we assume that the BS has acquired user CSI knowledge from measurement report or based channel reciprocity in TDD systems. Therefore, the BS is responsible for scheduling users as groups over available RBs for every transmission period (i.e., T timeslots). To represent the scheduling between users (in groups) and RBs, we introduce an indicator 3D-array $\boldsymbol{X} = \{x_i^{f,t}\}_{i,f,t} \in \mathbb{C}^{U \times F \times T}$, where $x_i^{f,t}$ is defined as

$$x_i^{f,t} = \begin{cases} 1, & \text{if user } i \text{ is scheduled at RB}(f,t) \\ 0, & \text{Otherwise.} \end{cases} \quad (1)$$

Next, we let s_i be user i's data symbol and p_i be the transmission power allocated to user i. The downlink signal received by user i scheduled in RB(f,t) can be derived by:

$$y_i = \underbrace{\left(\boldsymbol{h}_i^f\right)^{\mathbf{H}} \boldsymbol{v}_i^f \sqrt{p_i} s_i}_{\text{desired signal}} + \underbrace{\left(\boldsymbol{h}_i^f\right)^{\mathbf{H}} \sum_{j \in \mathcal{U}, j \neq i} \boldsymbol{v}_j^f \sqrt{p_j} s_j x_j^{f,t}}_{\text{co-channel interference}} + \omega_i \quad (2)$$

where $\boldsymbol{h}_i^f = [h_{i,1}^f, \ldots, h_{i,L}^f]^{\mathbf{T}} \in \mathbb{C}^L$ is the CSI vector of user i at subband f, $\omega_i \sim \mathcal{CN}(0, \sigma^2)$ represents the complex i.i.d. Additive White Gaussian Noise (AWGN), σ^2 is the variance of AWGN (i.e., the noise power), and $\boldsymbol{v}_i^f \in \mathbb{C}^L$ is the unitary beamforming precoder used by the BS for user i at subband f, which can be selected as the Zero-Forcing (ZF) [24], weighted Minimum Mean Squared Error (MMSE) [25] or Maximum Ratio Transmitter (MRT) [26].

Based on the above formulation, Signal-to-Interference-plus-Noise Ratio (SINR) experienced by user i on RB(f,t) is

$$\gamma_i^{f,t} = \frac{\mathbb{E}\left[|p_i s_i|^2\right] \cdot \left|\left(\boldsymbol{h}_i^f\right)^{\mathbf{H}} \boldsymbol{v}_i^f\right|^2}{\sigma^2 + \sum_{j\in\mathcal{U}, j\neq i} x_j^{f,t} \cdot p_j \mathbb{E}\left[|s_j|^2\right] \cdot \left|\left(\boldsymbol{h}_i^f\right)^{\mathbf{H}} \boldsymbol{v}_j^f\right|^2}, \quad (3)$$

Without loss of generality and for simplicity, we apply the MRT precoders such that

$$\boldsymbol{v}_i^f = \frac{\overline{\boldsymbol{h}_i^f}}{\left\|\boldsymbol{h}_i^f\right\|}, \forall i \in \mathcal{U}, f \in \mathcal{F}. \quad (4)$$

Note that $\|\boldsymbol{h}_i^f\|$ is user i's channel gain at subband f. After applying (4) into (3), the SINR can be reformulated as

$$\gamma_i^{f,t} = \frac{\mathbb{E}\left[|p_i s_i|^2\right]}{\sigma^2 \|\boldsymbol{h}_i\|^{-2} + \sum_{j\in\mathcal{U}, j\neq i} x_j^{f,t} \cdot \mathbb{E}\left[|p_i s_i|^2\right] \cdot \rho^2\left(\boldsymbol{h}_i^f, \boldsymbol{h}_j^f\right)}, \quad (5)$$

where the spatial CSI correlation coefficient between $\boldsymbol{h}_i^f$ and $\boldsymbol{h}_j^f$ is defined by the magnitude of correlation coefficient

$$\rho\left(\boldsymbol{h}_i^f, \boldsymbol{h}_j^f\right) = \frac{\left|\left(\boldsymbol{h}_i^f\right)^{\mathbf{H}} \boldsymbol{h}_j^f\right|}{\left\|\boldsymbol{h}_i^f\right\| \left\|\boldsymbol{h}_j^f\right\|}, \quad \forall i, j \in \mathcal{U}, f \in \mathcal{F}. \quad (6)$$

Clearly, the CCI experienced by user i increases with the sum of squares of CSI correlations between user i and other users in the same RSG scaled by the expected value of transmitted symbol energy. Since we are interested in exploiting channel correlations of users in user scheduling problems and desire simple solutions, we let the BS normalize transmit power per user such that $\mathbb{E}[|p_i s_i|^2] = \mathbb{E}[|p_j s_j|^2] = 1, \forall i, j \in \mathcal{U}$.

Consequently, given total bandwidth β, the achievable data rate of user i scheduled at RB(f,t) can be obtained as

$$R_i^{f,t} = \frac{\beta}{F} \log_2\left(1 + \gamma_i^{f,t}\right). \quad (7)$$

B. MU-MIMO USER GROUPING AND SCHEDULING

We consider a practical system where each user can use a distinct Modulation Coding Scheme (MCS) for transmission [28]. A higher MCS index corresponds to a higher data rate at sufficiently high SINR [3]. The mapping between MCS index, minimum required SINR and spectral efficiency is depicted in Table 3 [27]. In this paper, we assume that BS assigns the highest achievable MCS index to users based on their SINR in the scheduled RB, which can be formulated as

$$c_i^* = \underset{\forall c_i, \phi(c_i) \leq \gamma_i^{f,t}}{\arg\max} \ \phi(c_i) \quad (8)$$

where c_i^* is the MCS index assigned to user i and $\phi(c_i)$ denotes the mapping function that returns the minimum requisite SINR value for a selected MCS index c_i.

TABLE 3. Table listing the mapping between MCS index, the minimum required SINR and spectral efficiency [27].

MCS index	SINR (dB)	Modulation	Code rate	SE (b/s/Hz)
1	-7.58	QPSK	0.076	0.1523
2	-5.78	QPSK	0.120	0.2344
3	-3.76	QPSK	0.190	0.3770
4	-1.77	QPSK	0.300	0.6016
5	0.17	QPSK	0.440	0.8770
6	2.07	QPSK	0.590	1.1758
7	3.99	16QAM	0.370	1.4766
8	5.83	16QAM	0.480	1.9141
9	7.91	16QAM	0.600	2.4063
10	9.76	64QAM	0.450	2.7305
11	11.64	64QAM	0.550	3.3223
12	13.65	64QAM	0.650	3.9023
13	15.58	64QAM	0.750	4.5234
14	17.48	64QAM	0.850	5.1152
15	19.46	64QAM	0.930	5.5547

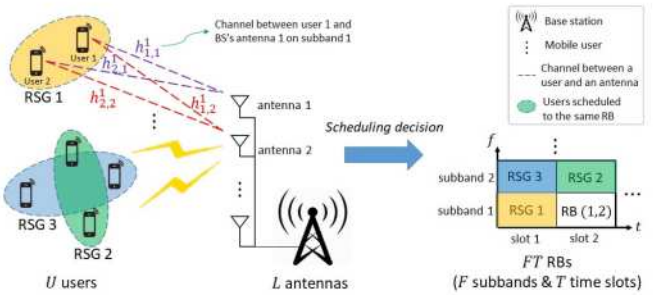


FIGURE 1. Illustration of the system model and objective. Note that user grouping is decided based on users' CSIs rather than geographical locations.

Practically, each user i has a minimum rate requirement $R_i^{\min}$. We use the step function $u(.)$ to define its effective data rate as

$$\widetilde{R}_i^{f,t} = \frac{\beta}{F} \zeta\left(c_i^*\right) \cdot u\left(\frac{\beta}{F} \cdot \zeta\left(c_i^*\right) - R_i^{\min}\right) \quad (9)$$

which means that the effective rate of a user is 0 unless its minimum rate is satisfied. Note that $\zeta(c_i^*)$ is the SE when using MCS of index c_i^*.

Finally, as shown in Fig. 1, our goal in this work is to design user grouping and scheduling algorithms that allocate a given set of RBs to U users to maximize the effective sum rate. This constrained optimization problem is formulated as:

$$\begin{aligned} \max_{\boldsymbol{X}} \quad & \sum_{f\in\mathcal{F}} \sum_{t\in\mathcal{T}} \sum_{i\in\mathcal{U}} x_i^{f,t} \widetilde{R}_i^{f,t} \\ \text{s.t.} \quad & \text{C1}: x_i^{f,t} \in \{0,1\}, \quad \forall i, f, t, \\ & \text{C2}: \sum_{f\in\mathcal{F}} \sum_{t\in\mathcal{T}} x_i^{f,t} = 1, \quad \forall i, \\ & \text{C3}: \sum_{i\in\mathcal{U}} x_i^{f,t} \leq L, \quad \forall f, t, \end{aligned} \quad (10)$$

where, among the three constraints {C1, C2, C3}, constraint C2 requires each user to be assigned to exactly one RB and constraint C3 states the maximum number of users scheduled in an RB may not exceed the number of antennas to ensure sufficient diversity within each RSG.

Note that (10) is a nonlinear integer programming problem and is NP-hard [29]. Therefore, the present challenge is to develop a low-complexity and effective algorithm that is relatively scalable for different system parameters such as total number of users, number of antennas, and channel realizations, to name a few.

IV. PROPOSED FRAMEWORK: ULAWS

A. BASIC APPROACH AND ASSUMPTIONS

As shown in (5), user SINR depends on both channel gain and CCI from users scheduled in the same RSG. To maximize the system's sum rate, users should be scheduled at RBs with sufficient channel gain and acceptably low cumulative CCI. However, finding such a scheduling decision is challenging, especially for the wideband scenario. Across a sufficiently wide bandwidth, each user CSI tends to vary from subband to subband. Thus, not only channel gains but also pair-wise CSI correlations of users are different over the many subbands. Even for dozens or hundreds of subbands, the user scheduling problem includes numerous possible scheduling options and local sum rate maxima for the problem of (10). To the best of our knowledge, there exists no known method that can jointly consider users' channel gain and CSI correlation of users across a wideband to optimize MU-MIMO scheduling decisions.

To solve the difficult MU-MIMO problem (10), we formulate the association between users and RBs as a *many-to-one matching game* where each RB can be matched with multiple users in an MU-MIMO user group. The matching game aims to associate users with RBs based on estimated user SINR. Users corresponding to the same RB belong to the same RSG in MU-MIMO. Ideally, we can take a global view of the problem since the matching process considers the estimated SINR of all RSG users when assigning and scheduling users to RBs.

It should be noted that in such matching game requires both user channel gains and CCIs on each RB to estimate SINR. Based on available user CSIs, user channel gains are known at the BS. However, CCIs depend on the selection of co-channel users and can not be determined *a priori*. In other words, user scheduling and assignment depend on user CCIs whereas user CCIs depend on user assignment. To overcome this circular problem, we propose to *estimate* the CCI a user may experience on each RB before the scheduling decision. We can then use the estimated CCI in the matching game.

It is not straightforward to accurately estimate CCI owing to numerous possible combinations of co-channel users. To achieve a good estimation, we restrict our options to allow a user to be scheduled to RBs only if the CSI correlation between this user and every other user scheduled to the same RB is relatively low in comparison with that between the user and users allocated with other RBs.

We refer to groups of users whose pairwise CSI correlation is relatively low as *intrinsic groups* because they can provide insights into the intrinsic structure of users' CSI correlation. This restriction implies that if a user is scheduled to subband f, its co-channel users must be from the intrinsic group it belongs to in subband f. Thus, instead of treating all users as potential co-channel users, this restrictive solution only considers users within the same intrinsic group to estimate a user's CCI if it is scheduled to that RB.

We now outline our basic approaches:

- To determine intrinsic groups for each subband, we treat the problem as a graph clustering problem. Unsupervised learning techniques based on graph embedding and clustering can efficiently solve the clustering problem.
- From the clustering outcomes, the estimated SINR can be obtained according to intrinsic groups.
- We then utilize the Gale-Sharpley algorithm [30] to find the stable matching between users and RBs.
- Graph-based post-processing refinement.

Notice that, since the matching game solution depends on estimated CSI and SINR, one cannot guarantee actual user SINRs at the assigned RB is as high as expected. We should treat the first matching result as a preliminary scheduling decision and regard users matched with the same RB only as a Preliminary RSG (P-RSG). Thereafter, a graph-based post-processing procedure involving three efficient steps is designed to refine the sum rate performance. Note that it is possible for some users without any RB assignment when the number of users exceeds the quota of every RB. In such cases, the unmatched users should be reassigned to RBs in the post-processing phase without exceeding constraints {C1, C2, C3}.

Applying the above steps, the overall procedure of our proposed "ULAWS" algorithm is illustrated in Fig. 2. The details of each step will be discussed in the following subsections.

B. CONSTRUCTION OF CSI CORRELATION GRAPH

The proposed ULAWS first determines intrinsic groups in each subband such that CSI correlation among users within the same group is relatively low. We start by modeling the pairwise CSI correlations between users in subband f with an undirected weighted graph $\mathcal{G}_f^{\text{sim}} = (\mathcal{U}, \mathcal{W}_f^{\text{sim}})$, where $\mathcal{U}$ is set of nodes representing users and $\mathcal{W}_f^{\text{sim}} = \{w_{i,j}^f\}_{i,1j\in\mathcal{U}}$ denotes the set of weighted edges describing CSI correlation of users. Here we define the weight of the edge connecting nodes (users) i and j as

$$w_{i,j}^f = \psi\left(\rho\left(\boldsymbol{h}_i^f, \boldsymbol{h}_j^f\right)\right) \tag{11}$$

where ψ is a monotonically increasing transformation function that maps the value of CSI correlation between user i and user j to an appropriate weight value. A higher weight

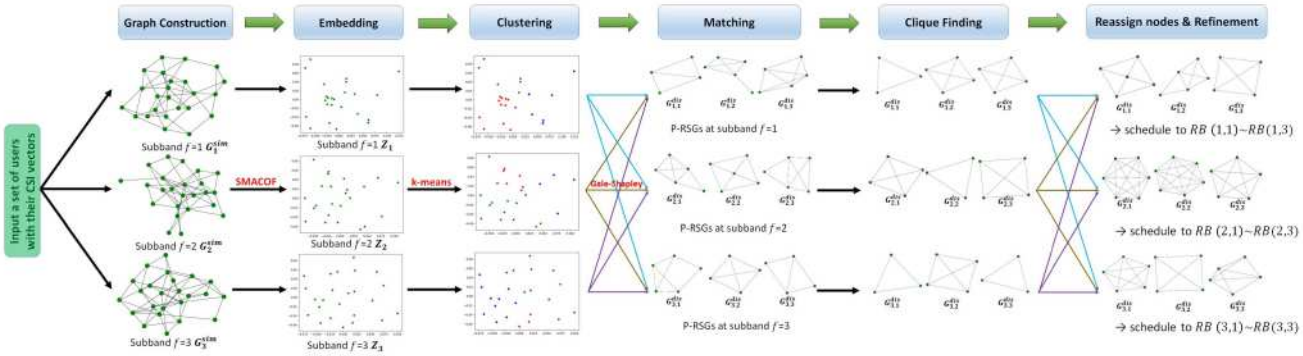


FIGURE 2. Illustration of a sample run of our proposed ULAWS with $U = 40, L = 8, F = 3, T = 3$. For the sake of exposition, all graphs are depicted using definition (22) and embeddings are visualized in 2D Euclidean space using Principal Component Analysis (PCA).

value implies a higher CSI correlation. We will provide the design of ψ in a later subsection.

Based on definition (11), we construct CSI correlation graphs for every subband $\mathcal{G}_f^{\text{sim}}, \forall f \in \mathcal{F}$ and *independently* find intrinsic groups in each graph. Note that finding intrinsic groups in a graph $\mathcal{G}_f^{\text{sim}}$ is equivalent to clustering $\mathcal{G}_f^{\text{sim}}$ into multiple subgraphs such that the weight among nodes in each subgraph is relatively low. Next, we discuss our graph clustering approach.

C. INTRINSIC USER GROUPING VIA GRAPH CLUSTERING

We note that clustering or partitioning on graphs is itself an NP-hard problem [31], [32]. Therefore, we first embed the graph into Euclidean space and apply K-means clustering next. In this way, cluster centers in K-means can act as anchor points to evaluate the relative distance of multiple nodes at the same time in Euclidean space. Although the acquired clusters are not necessarily optimal, this step can reduce the complexity from exponential to polynomial when compared to directly performing clustering on graphs.

1) Embedding: Given $\mathcal{G}_f^{\text{sim}}$ in subband f, let K be the dimension of the target embedding space. We aim to find a matrix $\mathbf{Z}_f = \{z_i^f\}_{i\in\mathcal{U}} \in \mathbb{R}^{U\times K}$, whose $i-$th row $z_i^f = [z_{i,1}^f, z_{i,2}^f, \ldots, z_{i,K}^f]$ is the coordinate of user i in K-dimensional Euclidean space, such that the Euclidean distance between user pairs approximates the corresponding weight value in $\mathcal{G}_f^{\text{sim}}$. i.e.,

$$d_{i,j}^f = \left\| z_i^f - z_j^f \right\| \approx w_{i,j}^f, \quad \forall i \in \mathcal{U}, j \in \mathcal{U} \tag{12}$$

We use $\boldsymbol{D}_f = \{d_{i,j}^f\}_{i,j\in\mathcal{U}}$ to denote the distance matrix.

We treat this as a metric MDS problem [33]. MDS and its extensions aim to learn embeddings of the original data in Euclidean spaces with a different dimension while preserving its local and global structure [34]. Since it only requires pairwise similarities as input, it is a suitable tool for our tasks.

The metric MDS can be solved by the SMACOF algorithm [35]. Given weights of $\mathcal{G}_f^{\text{sim}}$, the embedding matrix $\mathbf{Z}_f$ can be obtained by minimizing the stress function defined as

$$\begin{aligned}\tau(\mathbf{Z}_f) &= \sum_{i\in\mathcal{U}}\sum_{j\in\mathcal{U}}\left(d_{i,j}^f\left(z_i^f, z_j^f\right) - w_{i,j}^f\right)^2 \\ &= \sum_{i\in\mathcal{U}}\sum_{j\in\mathcal{U}}\left(\left(w_{i,j}^f\right)^2 + \left(d_{i,j}^f\right)^2 - 2w_{i,j}^f d_{i,j}^f\right) \\ &\le \kappa + \text{Tr}\left(\mathbf{Z}_f^T \mathbf{\Pi} \mathbf{Z}_f\right) - 2\cdot\text{Tr}\left(\mathbf{Z}_f^T\left(\boldsymbol{Q}_f(\mathbf{Z}_f)\right)\mathbf{Z}_f\right)\end{aligned} \tag{13}$$

where $Tr(\boldsymbol{B})$ denotes the trace of matrix $\boldsymbol{B}$, κ is a constant determined by weights of $\mathcal{G}_f^{\text{sim}}$, and $\boldsymbol{Q}_f = \{q_{i,j}^f\}_{i,j} \in \mathbb{C}^{U\times U}$ is a matrix whose elements are defined by

$$q_{i,j}^f = \begin{cases} \dfrac{w_{i,j}^f}{d_{i,j}^f\left(z_i^f, z_j^f\right)} & \text{if } i \neq j \\ 0 & \text{otherwise,} \end{cases} \tag{14}$$

is the auxiliary matrix and $\mathbf{\Pi} = U\cdot\boldsymbol{I} - \mathbf{1}\mathbf{1}^T$. The intuition of (13) is that when the squared difference between the weight of two adjacent nodes and the distance of two nodes in the Euclidean space is minimized, the embedding can preserve the graph structure accurately.

We can see from (13) that the last two terms of the right-hand side set the upper bound for the stress function and depend on $\mathbf{Z}_f$. By setting the gradient of τ with respect to $\mathbf{Z}_f$ to zero, we can obtain:

$$\frac{\partial\tau(\mathbf{Z}_f)}{\partial\mathbf{Z}_f} = 2\mathbf{\Pi}\mathbf{Z}_f - 2\left(\boldsymbol{Q}_f(\mathbf{Z}_f)\right)\mathbf{Z}_f = 0 \tag{15}$$

which imply the optimal $\mathbf{Z}_f$ that minimizes τ given matrix $\boldsymbol{Q}_f$ must satify the condition $\mathbf{Z}_f = U^{-1}\boldsymbol{Q}_f\mathbf{Z}_f$. This result is used in the iterative process of the SMACOF algorithm, whose overall procedure is summarized in Alg. 1.

Alg. 1 starts by randomly initializing the embedding matrix $\mathbf{Z}_f^0$. In each iteration $n > 0$, we compute the distance matrix $\boldsymbol{D}_f^n$ based on current embedding $\mathbf{Z}_f^n$. Next, matrix $\boldsymbol{Q}_f^n$ can be obtained using $\boldsymbol{D}_f^n$. Finally, we update the embedding matrix $\mathbf{Z}_f^n$ by using the following rule

$$\mathbf{Z}_f^{n+1} = U^{-1}\boldsymbol{Q}_f^n\mathbf{Z}_f^n \tag{16}$$

Algorithm 1: SMACOF Embedding Algorithm

Input: CSI correlation graph of subband f, $\mathcal{G}_f^{sim}$, target embedding dimension K
Output: Embedding matrix $\boldsymbol{Z}_f \in \mathbb{R}^{U\times K}$

Randomly initialize $\boldsymbol{Z}_f^0$
while *not converge* **do**
  $n \leftarrow$ iteration index
  Compute distance matrix $\boldsymbol{D}_f^n$ based on $\boldsymbol{Z}_f^n$
  Compute matrix $\boldsymbol{Q}_f^n$ based on $\boldsymbol{D}_f^n$
  Update $\boldsymbol{Z}_f^n$ based on $\boldsymbol{Q}_f^n$ using (16)
end

Algorithm 2: K-means Clustering Algorithm

Input: Embedding matrix $\boldsymbol{Z}_f$ for subband f
Output: Cluster centers $\{\boldsymbol{\mu}_t^f\}_{t=1}^T$, indicators δ

Randomly initialize cluster centers $\{\boldsymbol{\mu}_t^f\}_{t=1}^T$
while *not converge* **do**
  $\delta_{i,t}^f = \begin{cases} 1, \ t = \arg\min t \in \mathcal{T} \left\| \boldsymbol{z}_i^f - \boldsymbol{\mu}_t^f \right\| \ \forall i \in \mathcal{U}, t \in \mathcal{T} \\ 0, \ \text{Otherwise}, \end{cases}$
  $\boldsymbol{\mu}_t^f = \frac{\sum_{i\in\mathcal{U}} \delta_{i,t}^f \boldsymbol{z}_i^f}{\sum_{i\in\mathcal{U}} \delta_{i,t}^f}, \forall t \in \mathcal{T}$
end

known as the Guttman Transform [35]. We repeat this procedure until convergence. Here, we view SMACOF as converged either when it reaches maximum allowed iteration or when the following condition is satisfied

$$\left\| \boldsymbol{Z}_f^{n+1} - \boldsymbol{Z}_f^n \right\| < \lambda \tag{17}$$

where λ is a preset small constant.

2) Clustering: We next cluster the embedding result $\boldsymbol{Z}_f$ to identify T intrinsic groups, corresponding to T transmission timeslots, by using K-means. K-means aims to cluster data points into a specified number of clusters, represented by their centroid, such that the sum of variations within clusters is minimized [15]. This is given by

$$\begin{aligned} \min_{\delta^f, \mu^f} \ & \sum_{i\in\mathcal{U}} \sum_{t\in\mathcal{T}} \delta_{i,t}^f \left\| \boldsymbol{z}_i^f - \boldsymbol{\mu}_t^f \right\|^2 \\ \text{s.t.} \quad & \text{C1}: \delta_{i,t}^f \in \{0,1\}, \quad \forall i,j, \\ & \text{C2}: \sum_{t\in\mathcal{T}} \delta_{i,t}^f = 1, \quad \forall i \end{aligned} \tag{18}$$

where $\delta_{i,t}^f = 1$ if user i is assigned to cluster t and $\delta_{i,t}^f = 0$ otherwise. Recall that $\boldsymbol{\mu}_t^f \in \mathbb{R}^K$ is the center of cluster t.

We summarize our use of K-means as Alg. 2. We start with a set of randomly initialized cluster centers. In each iteration, we assign every user i to the cluster t whose center is the nearest. After that, we update cluster centers by averaging the coordinates of cluster users.

With a proper embedding for which (12) holds, pairwise Euclidean distance between two users can define their weight value $w_{i,j}^f$ in $\mathcal{G}_f^{sim}$, which monotonically increases with the magnitude of their CSI correlation coefficient. Thus, when objective (18) is minimized, the obtained clusters have the property that the CSI correlation between user pairs in the same cluster is relatively low. Hence, they can be viewed as intrinsic groups. We should note that these clusters are merely based on CSI correlation in a single subband without considering users' channel gain and CSI correlations in other subbands. Yet, they can provide insights into the intrinsic structure of CSI correlation in each subband that are helpful when estimating CCI.

We repeat the clustering process to find intrinsic groups over all subbands. We assume the t-th cluster in subband f represents the intrinsic group of RB (f,t). Thereafter, based on the assumption that only users belonging to the same intrinsic group can be potential co-channel users, we estimate the CCI that a user may experience in RB(f,t) according to:

$$\Gamma_i^{f,t} = \frac{U}{FT} \sum_{j\in\mathcal{U} j\neq i} \delta_{j,t}^f \cdot \rho^2\left(\boldsymbol{h}_i^f, \boldsymbol{h}_j^f\right) \tag{19}$$

where $\frac{U}{FT}$ implies the average number of users being associated with a RB by assuming uniform user association.

D. MANY-TO-ONE MATCHING BETWEEN USERS AND RBS

In the literature, matching algorithms have been discussed and implemented to address user grouping problems for wireless networks and showed superior performance [36], [15]. Inspired by prior works, we formulate the wideband scheduling problem as a many-to-one matching game aiming to associate users with RBs based on users' estimated SINR obtained with estimated CCI.

Specifically, we consider the set of users $\mathcal{U}$, and the set of RBs $\mathcal{B}$, as two disjoint sets of selfish and rational players to be matched with each other [37]. Each player has its own preference list which ranks the players in its opposing set from the most to the least favorite based on their preference [28]. The preference of a user $i \in \mathcal{U}$ over a RB $(f,t) \in \mathcal{B}$ is defined by user i's estimated SINR if being associated with RB (f,t), which can be computed according to:

$$\Upsilon_i^{US}(f,t) = \left(\sigma^2 \cdot \|\boldsymbol{h}_i\|^{-2} + \Gamma_i^{f,t}\right)^{-1} \tag{20}$$

On the other hand, the preference of an RB (f,t) for user i is defined by the Euclidean distance between user i's embedding at subband f and the t-th cluster center:

$$\Upsilon_{f,t}^{RB}(i) = -\left\| \boldsymbol{z}_i^f - \boldsymbol{\mu}_t^f \right\|. \tag{21}$$

This means that RBs favor users close to their respective cluster centers. Also, we define the relation operator $\succ$ such that $i \succ_{(f,t)} j$ if and only if $\Upsilon_{f,t}^{RB}(i) > \Upsilon_{f,t}^{RB}(j)$, which means that RB (f,t) prefers user i over user j.

Algorithm 3: Gale-Shapley Matching Algorithm

Input: $\Upsilon_i^{US}(f,t)$, $\Upsilon_{f,t}^{RB}(i)$, $\forall i \in \mathcal{U}, \forall (f,t) \in \mathcal{B}$
Output: Feasible matching function η

Initialize $\widetilde{\mathcal{B}}_i \leftarrow \mathcal{B}, \forall i \in \mathcal{U}, \eta(e) \leftarrow \emptyset, \forall e \in \mathcal{U} \cup \mathcal{B}$
while *not converge* **do**
  for $i = 1$ **to** $i = U$ **do**
    if $\eta(i) = \emptyset$ *and* $\widetilde{\mathcal{B}}_i \neq \emptyset$ **then**
      $(f^*, t^*) = \arg\max_{(f,t)\in\widetilde{\mathcal{B}}_i} \Upsilon_i^{US}(f,t)$
      $\widetilde{\mathcal{B}}_i \leftarrow \widetilde{\mathcal{B}}_i \setminus \{(f^*, t^*)\}$
      if $|\eta(f^*, t^*)| < m_{f^*,t^*}$ **then**
        $\eta(f^*, t^*) \leftarrow \eta(f^*, t^*) \cup \{i\}$
        $\eta(i) \leftarrow \{(f^*, t^*)\}$
      else
        $j^* = \arg\min_{j\in\eta(f^*,t^*)} \Upsilon_{f^*,t^*}^{RB}(j^*)$
        if $i \succ_{(f^*,t^*)} j$ **then**
          $\eta(f^*, t^*) \leftarrow \eta(f^*, t^*) \setminus \{j^*\} \cup \{i\}$
          $\eta(i) \leftarrow \{(f^*, t^*)\}$
          $\eta(j^*) \leftarrow \emptyset$
        end
      end
    end
  end
end

If user i is associated with RB (f,t), we say user i and RB (f,t) are matched with each other and form a matching pair. The goal of our considered matching game is to find a matching function η that map from $\mathcal{U} \cup \mathcal{B}$ to $2^{\mathcal{U}\cup\mathcal{B}}$ such that $\forall i \in \mathcal{U}$ and $\forall (f,t) \in \mathcal{B}$, the following conditions are satisfied

1) $\eta(i) \subseteq \mathcal{B} \cup \emptyset$ and $|\eta(i)| \in \{0, 1\}$
2) $\eta(f,t) \subseteq 2^{\mathcal{U}}$ and $|\eta(f,t)| \leq m_{f,t}$
3) $(f,t) \in \eta(i) \iff i \in \eta(f,t)$

Condition 1) allows each user to either be matched with one RB from $\mathcal{B}$ or remain unmatched. Condition 2) allows each RB (f,t) to match with at most $m_{f,t}$ users from $\mathcal{U}$, where $m_{f,t}$ is the quota of RB (f,t). Condition 3) denotes reciprocity, i.e., user i is matched with RB (f,t) if and only if RB (f,t) is matched with user i.

A matching function η is stable if and only if there is no blocking pair $(i,(f,t))$ with $i \notin \eta(f,t)$ and $(f,t) \notin \eta(i)$ such that $(f,t) \succ_i \eta(i)$ and $i \succ_{(f,t)} \eta(f,t)$, which means that user i and RB (f,t) prefer each other to the matching partner they obtain. The efficient Gale-Sharpley algorithm [30], or deferred acceptance algorithm [15], is a well-known method for solving the optimal stable matching [38] problem. We adopt Gale-Sharpley to address our matching game as summarized in Alg. 3.

Since the matching result is obtained from the estimated CCI instead of the actual CCI, we treat the matching result as a preliminary grouping decision and design the corresponding post-processing procedure to improve the performance in the next subsection.

E. GRAPH-BASED POST PROCESSING AND REFINEMENT

To narrow the performance gap between the preliminary scheduling decision obtained by Alg. 3 and the optimum solution, we design a post-processing procedure consisting of three efficient steps:

1) Clique Detection: The first step follows the idea of our previous work [19]. For every P-RSG $\eta(f,t)$, $\forall (f,t) \in \mathcal{B}$ obtained by Alg. 3, we define an undirected weighted graph $\mathcal{G}_{f,t}^{dis} = (\eta(f,t), \mathcal{W}_{f,t}^{dis})$ to capture the structure of CSI dissimilarity among users in P-RSG (f,t), where $\eta(f,t)$ denote the set of users in P-RSG (f,t) and $\mathcal{W}_{f,t}^{dis} = \{w_{i,j}^{f,t}\}_{i,j\in\eta(f,t)}$ is the set of weighted edges. Unlike previously defined CSI correlation graphs $\mathcal{G}_f^{sim}, \forall f \in \mathcal{F}$ using Eq. (11), here we define the weight by introducing a small threshold ϵ as follows

$$w_{i,j}^{f,t} = \begin{cases} 1 - \rho^2\left(\boldsymbol{h}_i^f, \boldsymbol{h}_j^f\right) & \text{if } \rho\left(\boldsymbol{h}_i^f, \boldsymbol{h}_j^f\right) \leq \epsilon \\ 0 & \text{otherwise,} \end{cases} \tag{22}$$

Note that $w_{i,j}^{f,t} \neq 0$ implies the CSIs of users i and j are dissimilar (i.e., low correlation) in subband f and hence they pose low mutual CCI when scheduled on the same RB, whereas $w_{i,j}^{f,t} = 0$ implies the opposite. Although several candidate functions can be used to define the weight value, we adopt these threshold-based weights for simplicity. We then define a diagonal degree matrix $\boldsymbol{E}_{f,t}^{dis} = \{e_{i,j}^{f,t}\}_{i,j} \in \mathbb{R}^{|\eta(f,t)|\times|\eta(f,t)|}$ for graph $\mathcal{G}_{f,t}^{dis}$ such that

$$e_{i,j}^{f,t} = \begin{cases} \sum_{k\in\mathcal{U},k\neq i} w_{i,k}^{f,t}, & \text{if } i = j \\ 0 & \text{otherwise.} \end{cases} \tag{23}$$

The goal of this step is to uncover the maximal clique (defined below) hidden in the CSI dissimilarity graph $\mathcal{G}_{f,t}^{dis}$. The motivation is that P-RSGs obtained by Alg. 3 could still contain user pairs that exhibit high correlation while ideally, the CSI correlation of *any* user pair in RSGs should be kept low and thus their CSI dissimilarity graph is a clique according to Eq. (22). By ensuring that $\mathcal{G}_{f,t}^{dis}$ is a clique, every user in $\mathcal{G}_{f,t}^{dis}$ will indeed enjoy low CCI in cochannel MIMO access.

Definition 1: An undirected weighted graph is a *clique* if it is complete, or fully connected. That is, a graph $\mathcal{G}$, $|\mathcal{G}| \geq 1$, is a clique if all its edge weights $w_{i,j} \neq 0, \forall i,j \in \mathcal{G}$ with $i \neq j$.

Ironically, finding the maximal clique in a graph is also a NP-hard problem [39]. Existing works, e.g., [13], often rely on exact algorithms such as MaxCliqueDyn to find the maximum clique in CSI dissimilarity graphs. However, it has exponential complexity in the worst case. Fortunately, we find the following heuristic solution to be quite effective. We first compute the degree matrices using (23). Next, we iteratively remove the node of the lowest degree until the

remaining nodes (i.e., remaining users in P-RSG (f,t)) form a clique. This is because low-degree nodes are less likely to form a clique than other nodes. We use a buffer set $\mathcal{A}$ to store all nodes removed from the graph $\mathcal{G}^{dis}_{f,t}$. We repeat the process until all CSI dissimilarity graphs $\mathcal{G}^{dis}_{f,t}, \forall (f,t) \in \mathcal{B}$ become cliques.

2) Node Reassignment: To ensure every user can be allocated one RB, in the second step we reassign the nodes we removed in the previous step into P-RSGs aiming to minimize their CCI to existing users. Note that it is possible that some users may remain unmatched after executing Alg. 3, depending on the quota of RBs. In that case, we would add these users to the buffer set $\mathcal{A}$ and reassign them as well.

We sequentially remove an user i from the buffer $\mathcal{A}$ and add it to P-RSG (f^*, t^*), which has the minimum sum of squared CSI correlations between user i and users in $\mathcal{G}^{dis}_{f^*,t^*}$ plus normalized noise in subband f, i.e.,

$$(f^*, t^*) = \arg\min_{(f,t)\in\mathcal{B}} \frac{\sigma^2}{\left\|\boldsymbol{h}_i^f\right\|^2} + \sum_{j\in\mathcal{G}^{dis}_{f,t}} \rho^2\left(\boldsymbol{h}_i^f, \boldsymbol{h}_j^f\right)$$

which suggests that this P-RSG has maximum SINR for user i. We repeat this process until $\mathcal{A}$ is empty.

3) Iterative Refinement: The final step aims to iteratively refine the preliminary scheduling decision (P-RSGs) in terms of sum rate until convergence. In each iteration, we sequentially chose a user i from $\mathcal{U}$ and evaluate the sum-rate improvement if it switches to another P-RSG (f,t), $\forall (f,t) \in \mathcal{B}$ by comparing the effective sum rate of users in both P-RSG (f,t) and P-RSG $\eta(i)$ (i.e., the RB user i currently associate with), which can be calculated by

$$\begin{aligned} R^{before}_{\eta(i),(f,t)} &= \sum_{j\in\eta(\eta(i))} \widetilde{R}_j^{\eta(i)} + \sum_{j\in\eta(f,t)} \widetilde{R}_j^{f,t} \\ R^{after}_{\eta(i),(f,t)} &= \sum_{j\in\eta(\eta(i))\setminus\{i\}} \widetilde{R}_j^{\eta(i)} + \sum_{j\in\eta(f,t)\cup\{i\}} \widetilde{R}_j^{f,t} \end{aligned} \tag{24}$$

We then switch it to the P-RSG with the best sum-rate improvement. In other words,

$$(f^*, t^*) = \arg\max_{(f,t)\in\mathcal{B}} R^{\text{after}}_{\eta(i),(f,t)} - R^{\text{before}}_{\eta(i),(f,t)}$$

as long as its sum-rate improvement is positive. We repeat this process until no single user can find a better P-RSG to switch, or until a maximum number of allowed iterations.

Alg. 4 summarizes the three post-processing steps above and the overall ULAWS algorithm is presented in Alg. 5.

V. WEIGHT TRANSFORMATION FOR GRAPH EMBEDDING

This section presents the design of our weight transformation function ψ used in (11). Without loss of generality, we consider CSI correlations between users as i.i.d. random variables, with distributions determined by system settings such as channel model and number of antennas. Thus, $\mathcal{G}^{\text{dis}}_f, \forall f \in \mathcal{F}$ defined in (11) are random and dense graphs whose weights are non-zero and given by CSI correlations among users.

Algorithm 4: Graph-Based Post-Processing

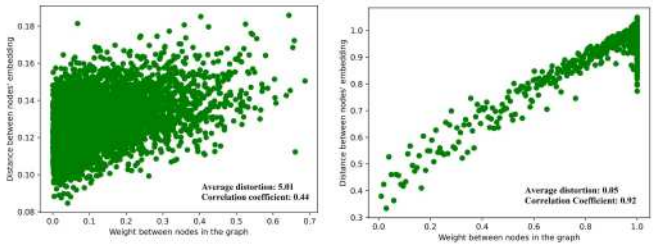
Input: CSI matrix $\boldsymbol{H} = \{h^f_{i,\ell}\}^{U,L,F}_{i,\ell,f}$, matching result η
Output: Scheduling decision $X = \{x_i^{f,t}\}^{U,F,T}_{i=1,f=1,t=1}$

Initialize $\mathcal{A} = \{i|\eta(i) = \emptyset\}$, $x_i^{f,t} \leftarrow 0,\ \forall i,f,t$
for $f = 1$ **to** $f = F$ **do**
  for $t = 1$ **to** $t = T$ **do**
    $\mathcal{G}^{dis}_{f,t} \leftarrow$ CSI dissimilarity graph of users in t-th cluster of subband f
    while $\mathcal{G}^{dis}_{f,t}$ *is not a clique* **do**
      Remove node $i^* = \arg\min_{i\in\mathcal{G}^{dis}_{f,t}} e_{i,i}$ from $\mathcal{G}^{dis}_{f,t}$
      $\mathcal{A} \leftarrow \mathcal{A} \cup \{i\}$
    end
    $x_i^{f,t} \leftarrow 1, \forall i \in \mathcal{G}^{dis}_{f,t}$
  end
end
while $\mathcal{A} \neq \emptyset$ **do**
  Randomly remove a user i from $\mathcal{A}$
  $(f^*, t^*) = \arg\min_{(f,t)\in\mathcal{B}} \frac{\sigma^2}{\|h_i^f\|^2} + \sum_{j\in\mathcal{G}^{dis}_{f,t}} \rho^2(\boldsymbol{h}_i^f, \boldsymbol{h}_j^f)$
  $x_i^{f^*,t^*} \leftarrow 1$
end
while *not converge* **do**
  for $i = 1$ **to** $i = U$ **do**
    $(f^*, t^*) = \arg\max_{(f,t)\in\mathcal{B}} R^{after}_{\eta(i),(f,t)} - R^{before}_{\eta(i),(f,t)}$
    if $R^{after}_{\eta(i),(f^*,t^*)} > R^{before}_{\eta(i),(f^*,t^*)}$ **then**
      $x_i^{\eta(i)} \leftarrow 0,\ x_i^{f^*,t^*} \leftarrow 1$
      $\eta(i) \leftarrow \{(f^*, t^*)\}$
    end
  end
end

Algorithm 5: Unsupervised Learning Aided and Matching-Based Wideband Scheduling, (ULAWS)

Input: CSI matrix $\boldsymbol{H} = \{h^f_{i,\ell}\}^{U,L,F}_{i=1,\ell=1,f=1}$
Output: Scheduling decision $X = \{x_i^{f,t}\}^{U,F,T}_{i=1,f=1,t=1}$

for $f = 1$ **to** $f = F$ **do**
  $\mathcal{G}^{sim}_f \leftarrow$ CSI correlation graph of subband f
  $\boldsymbol{Z}_f \leftarrow$ embedding of $\mathcal{G}^{sim}_f$ obtained by Alg. 1
  $\delta^f \leftarrow$ clustering users into T clusters by Alg. 2
end
Compute $\Upsilon^{US}_i(f,t)$, $\Upsilon^{RB}_{f,t}(i)$, $\forall i \in \mathcal{U}, (f,t) \in \mathcal{B}$ by (20) (21)
$\eta \leftarrow$ preliminary scheduling decision by Alg. 3
Perform Post-processing and refinement using Alg. 4



(a) Classic MDS ($w^f_{i,j} = \rho^2_{i,j}, \forall i, j$) (b) MDS with weight transformation ψ

FIGURE 3. Scatter plot of embeddings in different schemes. Here we set $U = 100$ and $L = 8$. Each dot denotes one of $\binom{U}{2}$ pairs of users.

In practice, it is challenging to embed such graphs into Euclidean space on a large scale while accurately preserving their local structures. For example, Fig. 3 (a) shows the embedding result of a CSI correlation graph $\mathcal{G}_f^{\text{sim}}$, if we apply classic MDS [33] without any weight transformation (i.e., $w^f_{i,j} = \rho^2(\boldsymbol{h}^f_i, \boldsymbol{h}^f_j)$ in (11)). We can see that the distance among users' embeddings substantially deviates from their weight value in $\mathcal{G}_f^{\text{sim}}$ since classic MDS fails to find a feasible solution that satisfies (12) in the given Euclidean space. In fact, the stress value (13) would remain high in such cases when the embedding algorithm terminates.

The weight definition of a graph can severely alter the result of graph embedding [40]. To further improve the quality of graph embedding, we introduce a systematic transformation method that finds a weight distribution suitable for embedding using MDS. Through extensive experiments, we found SMACOF is especially adept at embedding random graphs with a negatively skewed weight distribution (i.e., most weights fall in high-value regions while only a few weights have low values). Embeddings of such graphs tend to uniformly spread on the surface of a hyperball when SMACOF is converged. In this way, we can better utilize the limited Euclidean space and can more accurately preserve the graph structure. Therefore, our transformation function ψ aims to map weights of arbitrary graphs into an "artificially-created" distribution that is negatively skewed.

Without loss of generality, we create our target distribution by uniformly sampling the function $f(x) = e^{\alpha} - e^{\alpha x}$, where α is a scaling factor, U^2 times in region [0,1]. This gives us the distribution $\boldsymbol{\Theta}$ such that:

$$\boldsymbol{\Theta} = \left\{ \theta_i \in \left[0, \frac{1}{U^2}, \frac{2}{U^2}, \ldots, 1 \right] \middle| 1 - e^{-\alpha\theta_i} \right\} \tag{25}$$

Our transformation function $\psi : \{\rho^2(\boldsymbol{h}^f_i, \boldsymbol{h}^f_j)\}_{\forall i,j \in \mathcal{U}} \to \boldsymbol{\Theta}$ apply the exact histogram matching [41] to map the given set of squared CSI correlation $\{\rho^2(\boldsymbol{h}^f_i, \boldsymbol{h}^f_j)\}_{\forall i,j \in \mathcal{U}}$ into target distribution $\boldsymbol{\Theta}$. Specifically, we first sort $\rho^2(\boldsymbol{h}^f_i, \boldsymbol{h}^f_j), \forall i, j$ in ascending order before assigning weights $w^f_{i,j}, \forall i, j$ with elements from $\boldsymbol{\Theta}$ that match the index of their CSI correlations. Because $\boldsymbol{\Theta}$ is independently generated from the CSI correlation set, it is invariant to different system parameters, implying our transform function can be applied to arbitrary graphs.

To illustrate the improvement of embedding quality brought by our designed transformation method, we first evaluate the average distortion between a graph $\mathcal{G}_f^{\text{dis}}$ and its embedding $\boldsymbol{Z}_f$, using definition [42]:

$$\frac{1}{\binom{U}{2}} \sum_{i \in \mathcal{U}} \sum_{j \in \mathcal{U}} \frac{\left| w^f_{i,j} - d^f_{i,j} \right|}{w^f_{i,j}} \tag{26}$$

On the other hand, since the ultimate goal of embedding is to find intrinsic groups in which mutual CSI correlation among users is *relatively* low, the absolute distortion does not matter in our problem. Thus, we also adopt the correlation coefficient between $\mathcal{W}_f^{sim}$ and $\boldsymbol{D}_f$ as a metric to measure the embedding quality. This can be given by:

$$\frac{\sum_{i \in \mathcal{U}} \left(w^f_{i,j} - \bar{w}^f \right) \left(d^f_{i,j} - \bar{d}^f \right)}{\sqrt{\sum_{i \in \mathcal{U}} \left(w^f_{i,j} - \bar{w}^f \right)^2} \sqrt{\sum_{i \in \mathcal{U}} \left(d^f_{i,j} - \bar{d}^f \right)^2}} \tag{27}$$

which measures the correlation between the original graph structure and the structure in the embedding space. Here $\bar{w}^f$ and $\bar{d}^f$ represent the average weight of edges in $\mathcal{G}_f^{sim}$ and average distance between embeddings with given $\boldsymbol{Z}_f$. Fig. 3 (b) shows that our transformation function significantly improves the embedding compared with the classic MDS, in terms of average distortion and coefficient correlation.

VI. COMPLEXITY AND CONVERGENCE ANALYSIS

In this section, we analyze the complexity of the proposed ULAWS in terms of the key system parameters.

We start from Alg. 1. For lines 4-5, it requires complexity of $\mathcal{O}(KU^2)$ to compute the distance matrix $\boldsymbol{D}_f^n$ and auxiliary matrix $\boldsymbol{Q}_f^n$. The matrix multiplication involved in updating $\boldsymbol{Z}_f^n$ using (16) in line 6 has a complexity of $\mathcal{O}(KU^2)$. Thus, given the maximum iteration count N_1, the overall complexity for embedding a graph using Alg. 1 is of order $O(N_1 KU^2)$.

In each iteration of Alg. 2, we need complexity of $\mathcal{O}(KT)$ to find the cluster center closet to user i's embedding and thus it requires $\mathcal{O}(KTU)$ complexity to update the cluster indicator of all users in line 3. Also, it requires a complexity of $\mathcal{O}(KTU)$ to update all cluster centers in line 4. Hence, given the maximum iteration count N_2, the overall complexity to cluster embeddings using Alg. 2 is of the order $\mathcal{O}(N_2 KTU)$.

In each iteration of Alg. 3, we only need $\mathcal{O}(1)$ in line 5 for users to propose to their most preferable RB since users' preferences over RBs are computed beforehand. Also, it costs $\mathcal{O}(U)$ to compare the proposing user and the least preferable user currently matched with a RB when the number of users matched with it exceeds the quota. Namely, in the worst case, we need to run FTU iterations for users to propose every possible RB until Alg. 3 terminates. Therefore, a complexity of $\mathcal{O}(FTU^2)$ is required to find a stable matching between users and RBs using Alg. 3.

TABLE 4. Time complexity of each algorithm used in ULAWS.

Algorithm	Embedding (Alg. 1)	Clustering (Alg. 2)	Matching (Alg. 3)	Post-processing (Alg. 4)	Overall (Alg. 5)
Complexity	$\mathcal{O}(KU^3)$	$\mathcal{O}(KTU^2)$	$\mathcal{O}(KTU^2)$	$\mathcal{O}(FTU^3)$	$\mathcal{O}(FU^3(T+K))$

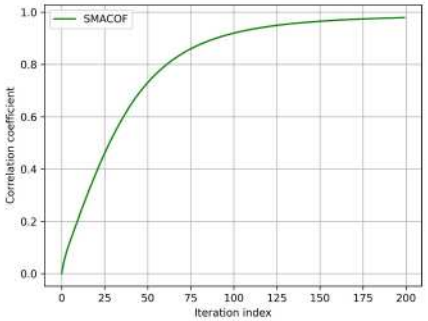


FIGURE 4. Convergence of SMACOF (Alg. 1). Runtime: 0.028sec.

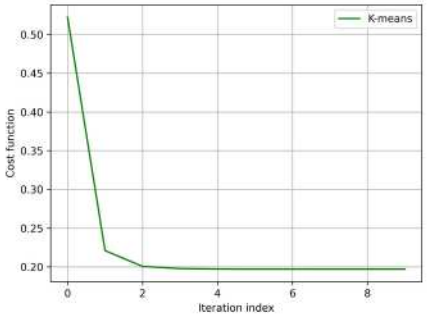


FIGURE 5. Convergence of K-means (Alg. 2). Runtime: 0.008sec.

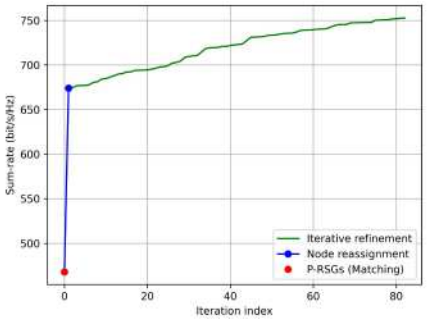


FIGURE 6. Convergence of Post-processing (Alg. 4). Runtime: 0.081sec.

Alg. 4 checks the degree of all nodes in $\mathcal{G}_{f,t}^{\text{dis}}$ to determine if it is a clique. This step requires complexity of $\mathcal{O}(U^2)$. Also, removing the node and updating the graph has a complexity of $\mathcal{O}(U)$. Thus, lines 5-8 need complexity of $\mathcal{O}(U^2)$ and the total cost for the first step of post-processing (lines 2–11) is $\mathcal{O}(FTU^2)$. In line 14, it needs $\mathcal{O}(FTU)$ to evaluate the normalized noise plus CCI for all RBs. Consequently, allocating remaining users in $\mathcal{A}$ for the second step (lines 12–16) would cost $\mathcal{O}(FTU^2)$. Finally, line 19 has a complexity of $\mathcal{O}(FTU^2)$ to find the optimal RB (f,t) for a user i to switch to in terms of sum-rate improvement. As a result, given the maximum iteration count N_3, the complexity of the third step (line 17–25) is $\mathcal{O}(N_3FTU^2)$. Combining all of the steps, the overall complexity for post-processing using Alg. 4 is $\mathcal{O}(N_3FTU^2)$.

The complexity of Alg. 5 is analyzed as follows. First, the complexity of computing a CSI correlation pair is $\mathcal{O}(L)$. Thus, it requires $\mathcal{O}(LU^2)$ to construct the CSI correlation graph for a subband in line 2. Note that we reserve the value of the computed CSI correlations for later procedures so as to reduce unnecessary complexity. Now let us assume the maximum iteration counts used in both Alg. 1, Alg. 2 and Alg. 4 are set to be less than the number of users, i.e., $N_i \leq U, i = 1,2,3$. We further assume that the user number is greater than the number of antennas, subbands and timeslots, i.e., $U \geq L, F, T$. Based on the complexity analysis above, it requires complexity $\mathcal{O}(F(LU^2 + N_1KU^2 + N_2KTU)) = \mathcal{O}(FKU^3)$ for line 1-5. Then, line 6 has a complexity of $\mathcal{O}(FTU^2)$, which is the same as line 7 (i.e., Alg. 3). Combining lines 1-8, the overall complexity of ULAWS is of order $\mathcal{O}(F(KU^3 + TU^2 + TU^3)) = \mathcal{O}(FU^3(T+K))$. The above complexity analysis is summarized in Table 4.

Next, we illustrate the convergence of steps used in ULAWS (i.e., Alg. 1, Alg. 2, Alg. 4) in Fig. 4, Fig. 5 and Fig. 6 respectively. To generate results in these figures, we executed ULAWS once for a problem instance with $U = 200$ users, $L = 8$ antennas, $F = 8$ subbands, $T = 6$ timeslot and $K = 200$ embedding dimension. Moreover, we adopt the correlation coefficient, the cost function of Eq. (18) normalized by U and K and sum-rate, as the convergence metric for Alg. 1, Alg. 2, Alg. 4, respectively, according to their design objectives. Note that the execution of Alg. 3 is not included here since it is a deterministic algorithm. We observe the growing performance improvement of both Alg. 1 and Alg. 4. The reason is that in each iteration we minimize the upper bound of stress function Eq.(13) in Alg. 1, and we switch a user to another RB if and only if the sum-rate difference is positive in the iterative refinement step of Alg. 4.

In summary, it takes ULAWS 0.138 seconds to solve this problem instance. The matching (0.013 seconds) and clustering (0.008 seconds) steps have the lowest runtime while embedding (0.028 seconds) and post-processing (0.081 seconds) steps are the complexity bottleneck of ULAWS, an observation corroborated by Table 4. Furthermore, we observe that the algorithms under consideration converge within 200 iterations, which verifies our guideline that the practical maximum iteration count of Alg. 1, Alg. 2, Alg. 4, can be set below the number of users.

VII. NUMERICAL EXPERIMENTS

In this section, we provide performance evaluation for the proposed wideband user grouping and scheduling algorithms via numerical simulations.

TABLE 5. Time complexity (worst case) and execution runtime (second) of comparison algorithms.

Algorithm	Complexity	50 users	100 users	200 users	300 users	600 users	900 users	1200 users
ULAWS	$\mathcal{O}(FU^3(T+K))$	0.0174	0.0506	0.1379	0.2709	1.006	2.196	3.537
SC-MS	$\mathcal{O}(FU^3)$	0.0061	0.0185	0.0878	0.1565	0.876	1.941	3.385
CARAD	$\mathcal{O}(U^4)$	0.0379	0.1266	0.2638	0.6578	1.877	3.658	6.482
HNG	$\mathcal{O}(FTU^3)$	0.0007	0.0034	0.0074	0.0168	0.066	0.147	0.258
MADOC	$\mathcal{O}(FTU^2)$	0.0005	0.0010	0.0031	0.0067	0.044	0.088	0.160
HRS	$\mathcal{O}(FTU^2)$	0.0003	0.0007	0.0026	0.0052	0.025	0.050	0.095

A. SIMULATION SETTINGS

Unless otherwise stated, we consider one BS equipped with $L = 8$ antennas serving $U = 600$ users. Users are uniformly distributed within a circular region centered at a BS with a radius of 50 m. Users are served with radio resources consisting of $F = 30$ subbands and $T = 5$ timeslots, leading to a sum of 150 RBs. The CSI 3D-array $\boldsymbol{H} \in \mathbb{C}^{U\times L\times F}$ is generated randomly, considering both path-loss, shadowing effect, and Rayleigh fading, with elements independent from each other. Note that since the cumulative time of considered timeslots is shorter than channel coherence time, channels can be viewed as quasi-static. The noise power is set to be $\sigma^2 = 0.01$ and bandwidth $\beta = 5$ MHz. Users' rate requirement is set as their achievable data rate in the subband with the lowest SNR. We use a correlation threshold $\epsilon = 0.07$ in the proposed ULAWS. The target distribution used for weights transformation in ULAWS adopts a scaling factor of $\alpha = 4.5$.

B. BENCHMARKS

We compare our proposed ULAWS with the following approaches applicable to wideband MU-MIMO scheduling problems with fixed RBs.

1) Spectral Clustering-aided Multi-band Scheduling (SC-MS) [19].

2) MADOC [9]: This method initialized every RB with an user with the highest SNR in the corresponding subband. In later iterations, it picks the user i with the highest SNR in all possible subbands and then assigns the user to the RB (f^*, t^*) in the selected subband f^* such that $t^* = \arg\min_{t\in\mathcal{T}} \max_{j\in\mathcal{U}} x_j^{f^*,t} \rho(\boldsymbol{h}_i^{f^*}, \boldsymbol{h}_j^{f^*})$.

3) CARAD [21]: This scheme first matches users with RB based on their preference lists through the Gale-Shapley algorithm [30]. Next, it iteratively searches for swap-blocking user pairs and swaps their matching until there is no swap-blocking user exist or the maximum allowed iterations $T_{max} = U$ is reached.

4) Heuristic algorithm (HRS) [12]: This algorithm starts with FT empty RBs. In each iteration, HRS sequentially adds an user i to RB (f^*, t^*) such that the CCI between user i and existing users in RB (f^*, t^*) is the minimum among FT RBs. Thus, HRS has a lower bound of $(1 - 1/k)$ times the optimal value for a general MAX k-CUT problem [43], where k is the number of clusters to assign.

5) Hungarian algorithm (HNG) modified from [44]: This matching algorithm initializes with FT empty RBs. In each iteration, HNG constructs a cost matrix $\boldsymbol{C} = \{c_{i,(f,t)}\}_{i,(f,t)} \in \mathbb{R}^{U\times(FT)}$ where $c_{i,(f,t))}, \forall i \in \mathcal{U}, \forall (f,t) \in \mathcal{B}$ is the CCI between user i and existing users in RB (f, t). Based on $\boldsymbol{C}$, Hungarian algorithm *optimally* assigns FT users to RBs, to minimize the sum matching cost. The process repeats until all U users are assigned to an RB, taking $\lceil \frac{U}{FT} \rceil$ iterations.

6) Round Robin (RR): It iteratively assigns an user from the user set, one at a time, to a RB in sequential order until the user set is empty. **Complexity**: $\mathcal{O}(U)$.

The complexity and execution runtime of these comparison algorithms are summarized in Table 5. We can observe that all compared schemes have polynomial worst-case time complexity. As illustrated in Table 5, the proposed ULAWS and SC-MS [19] have a similar complexity. Scheduling scheme based on swap-matching, i.e., CARAD [21], has the highest complexity while heuristic schemes, i.e., MADOC [9] and HRS [12], have the lowest complexity. In our runtime results, all algorithms are implemented with Python 3.6.8 and are executed on a Windows 10 laptop with Intel i7-9750H CPU and 16GB RAM. Although runtime in real systems depends on various factors such as the optimization of codes, the hardware used, and the choice of programming language and operation system, our result can provide insights into the scalability and relative complexity of different scheduling methods.

C. PERFORMANCE METRICS

The sum rate of users (i.e., throughput of the system) is widely used in the literature [45], [46] to evaluate the performance of scheduling policies. In this work, we consider two types of sum rate 1) achievable sum rate according to Eq. (7) and 2) effective sum rate using the definition of Eq. (9).

Note, however, that a scheduling policy aiming to maximize the sum rate may not always be reasonable in terms of system design. For example, due to the nature of the SINR function, very often the sum rate will be higher if we allocate an RB to one user with a decent channel condition rather than to multiple users with moderate channel conditions. This poses a fairness issue. Hence, in addition to the sum rate, we also adopt Jain's fairness index [47] as our metrics. The Jain fairness of data rates between users allocated to RB(f, t) is

computed by

$$J_{f,t} = \frac{\left(\sum_{i\in\mathcal{U}} x_i^{f,t} R_i^{f,t}\right)^2}{\left(\sum_{i\in\mathcal{U}} x_i^{f,t}\right) \cdot \sum_{i\in\mathcal{U}} \left(x_i^{f,t} R_i^{f,t}\right)^2} \in [0, 1]. \quad (28)$$

when $J_{f,t} = 1$, the data rates of every user in the RSG are exactly the same.

Other than fairness, we also would like to see how reasonable a scheduling decision is. To this end, we consider SINR loss to measure the ratio of noise plus CCI that users would experience given the scheduling decision and the noise they would experience if they were scheduled to the RB with the best channel gain without co-channel users. Such SNR loss can be measured by:

$$\frac{1}{U}\sum_{i\in\mathcal{U}} 10\log\left(\frac{\sigma^2\left\|\boldsymbol{h}_i^f\right\|^{-2} + \sum_{j\in\mathcal{U}} x_j^{f,t}\rho_{i,j}^2\left(\boldsymbol{h}_i^f, \boldsymbol{h}_j^f\right)}{\sigma^2 \cdot \left[\max_{\forall f\in\mathcal{F}}\left(\left\|\boldsymbol{h}_i^f\right\|^2\right)\right]^{-1}}\right) \text{dB} \quad (29)$$

According to (29), 0 dB loss corresponds to the ideal case when all users are scheduled to their most preferred subband with fully orthogonal co-channel users and zero CCI.

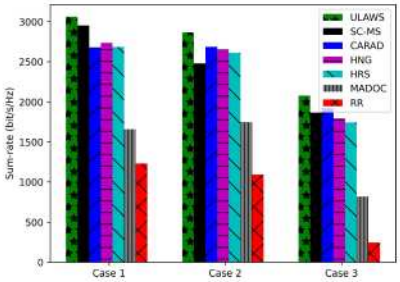


FIGURE 7. Users' sum rate of all schemes in three considered scenarios.

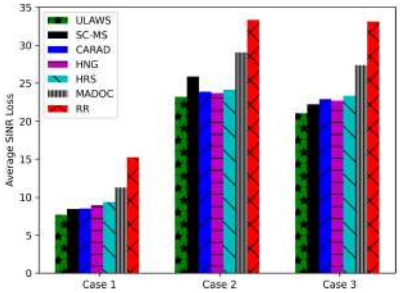


FIGURE 8. Average SINR loss of all schemes in three considered scenarios.

D. PERFORMANCES UNDER DIFFERENT SYSTEM SCENARIOS

Based on numerical tests, we now present the performance of scheduling algorithms in three different scenarios described below.

1) Case 1 (Achievable Rate for Equal and Unit CSI Gain): We let CSI amplitudes be the same for every user in every subband while the phase of CSIs remains random. In this way, users' sum rate will only be affected by CCI from co-channel users according to (5). This case helps to show how well different algorithms can form RSGs in each subband such that CSI correlations among users in the same group are low. It also represents the scenario where channel gain variation across different subbands is trivial.

2) Case 2 (Achievable Rate for Heterogeneous CSI Gains): We consider frequency-selective CSIs such that users' sum rate will be affected by both channel gain and CCI from co-channel users. This case can represent the scenarios where user channel gains and phases vary in different subbands.

3) Case 3 (Effective Rate for Heterogeneous Gains): Here channels are frequency-selective. However, in contrast to the previous two cases, we adopt the effective rate as the metric to eliminate rates contributed by users whose rate requirements are not met. Note that we apply the same rule to define the effective SINR loss (i.e., the effective SINR loss of an user is 0 if its requirement is not met).

Fig. 7 illustrates the resulting sum rate of all schemes in three considered scenarios. We can see that the proposed ULAWS has the highest sum rate in all cases. This implies that, in RSGs formed by ULAWS, users have acceptable CCI and are scheduled at their preferred subband. In contrast, MADOC and RR perform relatively worse in *Case 3* since they fail to find a solution to separate users with strong CCI in MIMO groups, leading to some failures to satisfy the rate requirement of certain users and the loss of overall effective rate. We also observe that the performance of SC-MS in *Case 2* is not as good as in the two other cases by comparison since SC-MS only considers user CSI correlations regardless of their channel gain. As expected, SC-MS performs the best in the *Case 1* among the three scenarios. In contrast, CARAD and MADOC perform relatively better in *Case 2* than *Case 1*. This is because their scheduling mechanisms favor users with good channel gain while such practice is not helpful in the *Case 1*, where channel gains of all subbands are assumed equal.

Fig. 8 compares the SINR loss in various scenarios. ULAWS shows the lowest SINR loss in all cases, because it forms RSGs for which noise plus CCI experienced by users does not deviate much from the lowest possible noise they experience. Generally, all the tested schemes exhibit high SINR loss in *Case 2* due to the effect of channel gain variation in different subbands, whereas all CSI gains are equal in *Case 1*. In *Case 3* only the SINR loss of satisfactory users is counted. As expected, We find that the SINR loss of SC-MS is relatively high in *Case 2*. Thus, even if user CCI in the same RSG obtained by SC-MS is low in *Case 2*, noises severely degrade SINR loss when users are assigned to less favorable subbands.

Fig. 9 shows distributions of the Jain index of all schemes in all scenarios. The proposed ULAWS achieves the highest average Jain index in all test cases. This confirms that ULAWS can achieve a high sum rate while preserving users' fairness. Both CARAD HNG, HRS and MADOC have similar performance in Jain index. Additionally, we

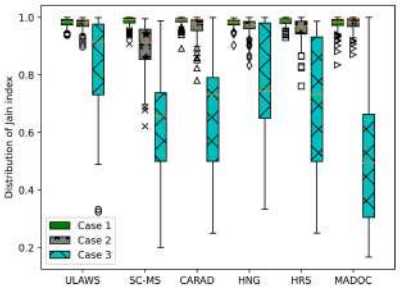


FIGURE 9. Distribution of fairness index of all RSGs in three considered scenarios.

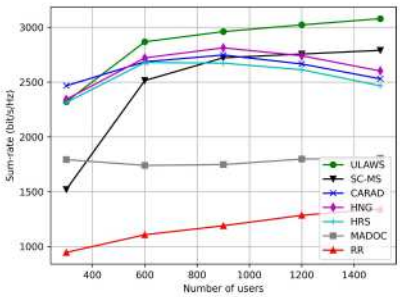


FIGURE 10. Users' sum rate for different numbers of users in different schemes.

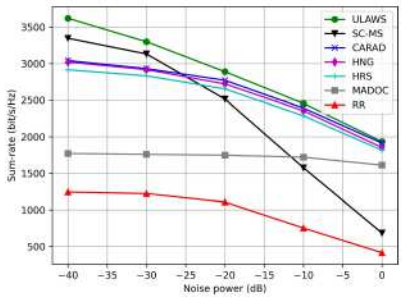


FIGURE 11. Users' sum rate for different noise power in different schemes.

can see that because a user's data rate is 0 in *Case 3* if its requirement is not met, the variation of data rate among users in the same RSG tends to be large, generating a much smaller Jain index in all schemes. Although SC-MS has the second-highest sum rate, as shown in Fig. 7, it has the lowest Jain index in *Case 3*. This means that its high sum rate is mostly contributed by users scheduled on the subband with decent channel gain, whereas data rate of users scheduled on unfavorable subbands is greatly degraded, creating a large variation of data rate in RSGs.

E. TESTING SYSTEM PARAMETERS

The main system parameters that affect the performance of scheduling algorithms are examined as follows.

1) User Number: We first test the performance of all schemes by increasing the user number while fixing the number of RBs. We can see from Fig. 10 that system's sum rate generally increases with the user number. CARAD, HNG, and HRS have similar performance in all cases. However, their sum rates start to decrease after $U \geq 900$. Such behavior is related to the nature of the sum rate and the constraints of each scheme. As long as the CCI remains low among users in each RSG, we can increase the sum rate by adding more users. However, for a given number of RBs (i.e., RSGs), when the number of users exceeds a certain level (e.g., $U = 900$), RSGs become over-populated such that both CARAD, HNG and HRS can not find a scheduling decision to separate users with high CSI correlation and CCI in different RBs within their design rules. As a result, user sum rate decreases due to the increased CCI within the RSGs.

On the other hand, both ULAWS, SC-MS, MADOC and RR show no decline in sum rates for $U \leq 1500$ users, albeit for different reasons. Since RR forms RSGs randomly, users suffer from severe CCI in all cases, even for very small numbers of users. Hence, its achieved sum rate does not show an elbow point as for HRS and HNG and CARAD. The is also the case for MADOC. In contrast, the sum rates of both ULAWS and SC-MS increase well beyond $U \geq 900$ because users in RSGs they formed do not suffer severe CCI. Compared with CARAD, HNG and HRS, ULAWS and SC-MS achieve a better overall balance by leveraging the combination of graph clustering and post-processing although their methodologies are not the same. Hence, they can still generate scheduling decisions to separate users with high CCI even for large user numbers, e.g., $U \geq 900$, whereas CARAD, HNG and HRS fail to do so. This result illustrates that ULAWS is more scalable for a larger number of users and more robust. That being said, sum rate will eventually decline when a sufficiently high number of users leads to severe CCI.

2) Noise Level Effect: Fig. 11 shows the resulting sum rates at different noise power levels under Additive White Gaussian Noise (AWGN). Generally, users' sum rate drops with increasing noise power. However, the performance of MADOC did not change much with the noise level. This implies the performance bottleneck of MADOC is CCI among users instead of noise power. In contrast, ULAWS is effective in adapting to different noise powers and performs relatively well in low-noise scenarios. On the other hand, the performance of SC-MS is relatively low in the highly noisy regime, but its performance improves when additive noise diminishes. This is because the CCI effect is more dominant when noise is low but SC-MS only aims to form RSGs with low CCI regardless of their channel gain.

3) Rate Requirement: Fig. 12 illustrates the effect of rate requirement on outage rate (i.e., ratio of users whose rate requirements are not met). This metric is highly correlated to the effective sum rate. The x-axis indicates the level of rate requirement compared to the original settings (e.g., 1.5 means 50% more). We observe that the outage rate grows with the rate requirement. Moreover, our ULAWS achieves a lower outage rate under high-rate requirements. As corroborated by Fig. 8, since the SINR losses of users

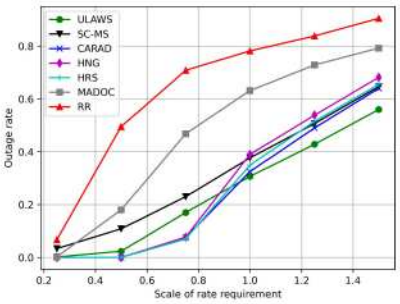


FIGURE 12. Outage rate of users versus different levels of rate requirement.

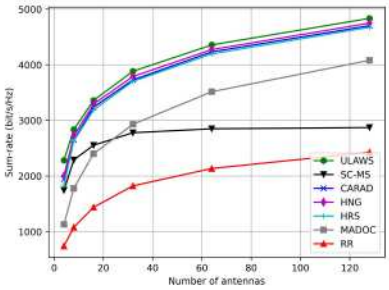


FIGURE 13. Users' sum rate for different numbers of antennas.

in RSGs obtained by ULAWS is low, it is more likely to satisfy users' rate requirement than CARAD, HRS and HNG, especially when the rate requirement is high.

However, when the rate requirement is low, ULAWS satisfies fewer user than either CARAD, HNG or HRS. This can be explained as follow. ULAWS's matching step and its clique-finding process aim to find sets of users *suitable* to be scheduled together at their preferred subband to mitigate their SINR loss. However, the outcome does not necessarily lead to the highest user satisfaction ratio for the low-rate requirement scenario. Since ULAWS does not specifically consider users' rate requirements, a user may be assigned to a less favorable subband to avoid causing high CCI at its favorite subband, thereby failing to meet its rate requirement. On the other hand, both HRS and HNG treat users equally to minimize the sum effect of noise and CCI experienced by all users in a greedy manner. Without considering the trade-off when selecting co-channel users, this can work well against low rate requirement but fails to satisfy most users under high rate requirement due to the greedy nature of the solution.

4) Antenna Number: We observe from Fig. 13 that generally, users' sum rate increases with the number of antennas. This is because increasing antenna numbers can reduce CSI correlation among users, thereby leading to lower CCIs and higher data rate. However, this rate improvement gradually saturates as the number of antennas grows asymptotically large. We conclude that ULAWS performs well in both scenarios with higher and lower antenna numbers. Note that in contrast to Fig. 10, the sum rate of MADOC indeed increases with antenna number. This verified our view that the performance bottleneck of MADOC is CCI among users. On the other hand, we observe the sum rate of SC-MS saturates after $L \geq 32$. This implies the performance bottleneck of SC-MS lies in the noise level instead of user CCI because SC-MC creates RSGs based on users' channel correlation without considering their SNR.

VIII. CONCLUSION

To explore and utilize the promised performance gain of MU-MIMO in wideband wireless systems, users should be scheduled on RB by jointly considering their channel characteristics across multiple subbands. To address this challenging NP-hard problem, we develop a novel Unsupervised Learning Aided Wideband Scheduling algorithm (ULAWS). Specifically, we first consider the scheduling between users and RBs as a matching game aimed at forming Preliminary Resource Sharing Groups (P-RSGs) based on users' estimated SINR as a way to provide a broader perspective of the problem. We suggest to apply SMACOF for graph embedding and to utilize the classic K-means clustering algorithm to find intrinsic groups based on CSI correlation among users. We then use the obtained intrinsic groups to estimate user CCI in the matching process. We find a stable matching of the formulated game (i.e., P-RSGs) using the Gale-Sharpley algorithm. Finally, a universal weight transformation function enhances the quality of graph embedding for arbitrary graphs. Our numerical results show that the proposed ULAWS method for MU-MIMO user scheduling and resource allocation outperforms other benchmark schemes in terms of sum rate, fairness and outage rate, under different system parameters and scenarios.

REFERENCES

[1] J. G. Andrews et al., "What will 5G be?" *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1065–1082, Jun. 2014.
[2] E. Larsson, F. Tufvesson, O. Edfors, and T. Marzetta, "Massive MIMO for next generation wireless systems," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 185–195, Feb. 2014.
[3] Y. Chen, Y. Wu, Y. T. Hou, and W. Lou, "mCore: Achieving sub-millisecond scheduling for 5G MU-MIMO systems," in *Proc. IEEE 40th Conf. Comput. Commun.*, 2021, pp. 1–10.
[4] E. Castañeda, A. Silva, A. Gameiro, and M. Kountouris, "An overview on resource allocation techniques for multi-user MIMO systems," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 1, pp. 239–284, 1st Quart., 2017.
[5] C. Singhal and S. De, *Resource Allocation in Next-Generation Broadband Wireless Access Networks*. Hershey, PA, USA: IGI Global, 2017.
[6] A. Goldsmith, S. Jafar, N. Jindal, and S. Vishwanath, "Capacity limits of MIMO channels," *IEEE J. Sel. Areas Commun.*, vol. 21, no. 5, pp. 684–702, Jun. 2003.
[7] J. Dai and S. Wang, "Clustering-based spectrum sharing strategy for cognitive radio networks," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 1, pp. 228–237, Jan. 2017.
[8] M. Alkhaled, E. Alsusa, and W. Pramudito, "Adaptive user grouping algorithm for the downlink massive MIMO systems," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, pp. 1–6, 2016.
[9] K.-U. Storek and A. Knopp, "Fair user grouping for multibeam satellites with MU-MIMO precoding," in *Proc. IEEE Global Commun. Conf.*, 2017, pp. 1–7.
[10] X. Yang, S. Zhang, B. Gao, and J. Cao, "A low complexity joint user grouping and resource allocation algorithm in massive MIMO systems," in *Proc. IEEE 19th Int. Conf. Commun. Technol. (ICCT)*, 2019, pp. 914–919.

[11] O. Saatlou and S. D. Blostein, "User selection for MU-MIMO based on channel estimation and spatial orthogonality," in *Proc. IEEE 34th Annu. Int. Symp. Pers., Indoor Mobile Radio Commun. (PIMRC)*, 2023, pp. 1–6.

[12] L. Liang, S. Xie, G. Y. Li, Z. Ding, and X. Yu, "Graph-based resource sharing in vehicular communication," *IEEE Trans. Wireless Commun.*, vol. 17, no. 7, pp. 4579–4592, Jul. 2018.

[13] B. Ahmad, D. G. Riviello, A. Guidotti, and A. Vanelli-Coralli, "Graph-based user scheduling algorithms for LEO-MIMO non-terrestrial networks," in *Proc. Joint Eur. Conf. Netw. Commun. 6G Summit*, 2023, pp. 270–275.

[14] Y. Xu, G. Yue, and S. Mao, "User grouping for massive MIMO in FDD systems: New design methods and analysis," *IEEE Access*, vol. 2, pp. 947–959, 2014.

[15] J. Cui, Z. Ding, P. Fan, and N. Al-Dhahir, "Unsupervised machine learning-based user clustering in millimeter-wave-NOMA systems," *IEEE Trans. Wireless Commun.*, vol. 17, no. 11, pp. 7425–7440, Nov. 2018.

[16] C. Feres and Z. Ding, "An unsupervised learning paradigm for user scheduling in large scale multi-antenna systems," *IEEE Trans. Wireless Commun.*, vol. 22, no. 5, pp. 2932–2945, May 2023.

[17] Z. Cheng, J. Yang, Z. Wei, and H. Yang, "User clustering and scheduling in UAV systems exploiting channel correlation," in *Proc. IEEE 30th Annu. Int. Symp. Pers., Indoor Mobile Radio Commun. (PIMRC)*, 2019, pp. 1–6.

[18] R.-F. Trifan, R. Lerbour, G. Donnard, and Y. L. Helloco, "K-means MU-MIMO user clustering for optimized precoding performance," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, 2019, pp. 1–5.

[19] C.-H. Hsu, C. Feres, and Z. Ding, "Spectral clustering aided user grouping and scheduling in wideband MU-MIMO systems," in *Proc. IEEE Int. Conf. Commun.*, 2023, pp. 4292–4297.

[20] B. Makki, T. Svensson, and M.-S. Alouini, "Throughput analysis of large-but-finite MIMO networks using schedulers," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, 2018, pp. 1–6.

[21] J. Zhao, Y. Liu, K. K. Chai, M. Elkashlan, and Y. Chen, "Matching with peer effects for context-aware resource allocation in D2D communications," *IEEE Commun. Lett.*, vol. 21, no. 4, pp. 837–840, Apr. 2017.

[22] B. Huang, C. Zhang, X. Bai, J. Li, M. Sun, and W. Kong, "Energy-efficient resource allocation for machine-type communications in smart grid based on a matching with externalities approach," *IEEE Access*, vol. 7, pp. 104354–104364, 2019.

[23] Y. Meng, Z. Zhang, Y. Huang, and P. Zhang, "Resource allocation for energy harvesting-aided device-to-device communications: A matching game approach," *IEEE Access*, vol. 7, pp. 175594–175605, 2019.

[24] N. Jindal, "MIMO broadcast channels with finite-rate feedback," *IEEE Trans. Inf. Theory*, vol. 52, no. 11, pp. 5045–5060, Nov. 2006.

[25] E. Bjornson and E. Jorswieck, "Optimal resource allocation in coordinated multi-cell system," *Found. Trends Commun. Inf. Theory*, vol. 9, nos. 2-3, pp. 113–381, 2013.

[26] T. K. Y. Lo, "Maximum ratio transmission," *IEEE Trans. Commun.*, vol. 47, no. 10, pp. 1458–1461, Oct. 1999.

[27] R. Hou, K. Huang, and K.-S. Lui, "An overlapping coalition formation game based multicast scheme in Backhaul-limited small cell networks," *IEEE Trans. Broadcast.*, vol. 66, no. 3, pp. 647–655, Sep. 2020.

[28] X. Ye and L. Fu, "Joint MCS adaptation and RB allocation in cellular networks based on deep reinforcement learning with stable matching," *IEEE Trans. Mobile Comput.*, vol. 23, no. 1, pp. 549–565, Jan. 2024.

[29] Z. Yang, L. Cai, and W.-S. Lu, "Practical scheduling algorithms for concurrent transmissions in rate-adaptive wireless networks," in *Proc. IEEE 29th Conf. Comput. Commun. (INFOCOM)*, 2010, pp. 120–128.

[30] A. E. Roth and M. A. O. Sotomayor, *Two-Sided Matching: A Study in Game-Theoretic Modeling and Analysis* (Econometric Society Monographs). Cambridge, U.K.: Cambridge Univ., 1990.

[31] H. Jia, S. Ding, and X. Xu, "The latest research progress on spectral clustering," *Neural Comput. Appl.*, vol. 24, pp. 1477–1486, Jun. 2014.

[32] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, Aug. 2000.

[33] W. S. Torgerson, "Multidimensional scaling of similarity," *Psychometrika*, vol. 30, no. 4, pp. 379–393, 1965.

[34] D. Chen, B. Fan, C. Oliver, and K. Borgwardt, "Unsupervised manifold alignment with joint multidimensional scaling," in *Proc. 11th Int. Conf. Learn. Represent.*, pp. 1–24, 2023. [Online]. Available: https://openreview.net/forum?id=lUpjsrKItz4

[35] J. de Leeuw and P. Mair, "Multidimensional scaling using majorization: SMACOF in R," *J. Statist. Softw.*, vol. 31, no. 3, pp. 1–30, 2009.

[36] X. Shi, D. Wu, C. Yue, C. Wan, and X. Guan, "Resource allocation for covert communication in D2D content sharing: A matching game approach," *IEEE Access*, vol. 7, pp. 72835–72849, 2019.

[37] B. Di, S. Bayat, L. Song, and Y. Li, "Radio resource allocation for downlink non-orthogonal multiple access (NOMA) networks using matching theory," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, 2015, pp. 1–6.

[38] D. Gale and L. S. Shapley, "College admissions and the stability of marriage," *Amer. Math. Mon.*, vol. 69, no. 1, pp. 9–15, 1962. [Online]. Available: http://www.jstor.org/stable/2312726

[39] Q. Wu and J.-K. Hao, "A review on algorithms for maximum clique problems," *Eur. J. Oper. Res.*, vol. 242, no. 3, pp. 693–709, 2015.

[40] I. Gallagher, A. Jones, A. Bertiger, C. E. Priebe, and P. Rubin-Delanchy, "Spectral embedding of weighted graphs," *J. Amer. Statist. Assoc.*, pp. 1–10, Jul. 2023. [Online]. Available: https://doi.org/10.1080/01621459.2023.2225239

[41] J. P. Rolland, V. Vo, B. Bloss, and C. K. Abbey, "Fast algorithms for histogram matching: Application to texture synthesis," *J. Electron. Imag.*, vol. 9, no. 1, pp. 39–45, 2000. [Online]. Available: https://doi.org/10.1117/1.482725

[42] F. Sala, C. De Sa, A. Gu, and C. Re, "Representation tradeoffs for hyperbolic embeddings," in *Proc. 35th Int. Conf. Mach. Learn.*, vol. 80, 2018, pp. 4460–4469. [Online]. Available: https://proceedings.mlr.press/v80/sala18a.html

[43] R. Y. Chang, Z. Tao, J. Zhang, and C. C. J. Kuo, "Multicell OFDMA downlink resource allocation using a graphic framework," *IEEE Trans. Veh. Technol.*, vol. 58, no. 7, pp. 3494–3507, Sep. 2009.

[44] A. A. Khan, R. S. Adve, and W. Yu, "Optimizing downlink resource allocation in multiuser MIMO networks via fractional programming and the hungarian algorithm," *IEEE Trans. Wireless Commun.*, vol. 19, no. 8, pp. 5162–5175, Aug. 2020.

[45] R. Tian, Y. Liang, X. Tan, and T. Li, "Overlapping user grouping in IoT oriented massive MIMO systems," *IEEE Access*, vol. 5, pp. 14177–14186, 2017.

[46] A. C. Cirik, K. Rikkinen, and M. Latva-aho, "Joint subcarrier and power allocation for sum-rate maximization in OFDMA full-duplex systems," in *Proc. IEEE 81st Veh. Technol. Conf.*, 2015, pp. 1–5.

[47] R. Jain, D. Chiu, and W. R. Hawe, "A quafntitative measure of fairness and discrimination for resource allocation in shared computer systems," Eastern Res Lab, DEC Res., Hudson, MA, USA, Rep. TR-301, 1984.