An Unsupervised Learning Paradigm for User Scheduling in Large Scale Multi-Antenna Systems

Carlos Feres, *Member, IEEE* and Zhi Ding, *Fellow, IEEE*

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

Abstract

The tremendous growth of mobile networking and IoT demands efficient and reliable service for massive wireless systems. Multi-input-multi-output (MIMO) technologies are successfully utilizing spatial diversity to substantially improve spectral efficiency by scheduling multiple devices for simultaneous spectrum access. Efficient solutions to the NP-hard problem of scheduling large number of users are vital to interference mitigation and spectrum efficiency. Despite successes of machine learning in tackling large-scale optimization problems, direct adoption of supervised learning in MIMO user scheduling is difficult as there is no optimum solution to use as labeled training data, and unsupervised learning would identify similar user channel features instead of promoting channel diversity. In this work, we propose an effective and scalable user scheduling paradigm based on unsupervised learning to enhance spatial diversity in both uplink and downlink. Given channel state information (CSI) of users, we first cluster CSIs over the Grassmannian manifold to identify users with high CSI similarity, before scheduling them into MIMO access groups with low co-channel interference. Our paradigm is generalizable to a variety of different simple and scalable unsupervised learning tools and different diversity optimization criteria. Numerical tests demonstrate substantial gain in terms of spectrum efficiency and interference suppression at modest computation complexity.

Index Terms

User scheduling, unsupervised learning, manifold clustering, MIMO systems, co-channel interference.

This material is based upon work supported by the National Science Foundation under Grants No. 2009001 and No. 2029027.

I. INTRODUCTION

In the new cyber-era of Internet-of-Things (IoT), billions of wireless devices are being connected globally [1], [2]. The tremendous growth rate of wireless communication networks in areas such as transportation, environmental monitoring, robotics, and smart cities continue to demand higher spectrum efficiency over the limited bandwidth resources. Multiple-Input-Multiple-Output (MIMO) technologies have been playing a central role in achieving high network throughput and spectrum efficiency [3] for both uplink and downlink links. In uplink multiple access channel (MAC), classical multi-user detection (MUD) receivers such as the maximum-likelihood, decorrelator, MMSE receivers and variants support simultaneous recovery of multiple signals sharing the same physical resource [4]–[12]. Similarly, broadcast (BC) enables shared spectrum and high efficiency in downlink [13].

Despite the intense interest and broad application of uplink MAC and downlink multiuser MIMO (MU-MIMO, also known as BC) systems, their efficiency and performance depend critically on user scheduling of co-channel user groups. In particular, high spectrum efficiency requires low mutual interference among co-channnel users scheduled for BC downlink or MAC uplink. The ultimate goal of *scheduling* is to assign users such that more users can share resources with little mutual interference [14] without sacrificing performance in terms of e.g., sum-rate, capacity, outage probability, among others. In other words, user scheduling aims to identify co-channel users with minimal or controlled co-channel interference (CCI).

In MIMO systems, CCI depends directly on spatial channel diversity among users [15], [16], i.e., on CSI similarities. To effectively mitigate CCI within MIMO user groups, one needs to assess for each possible co-channel MIMO user group their spatial diversity based on updated user CSIs. For a base station with M antennas and single antenna mobile devices, up to M users can be scheduled simultaneously in either MIMO downlink or uplink directions. The co-channel user CSI vectors must exhibit sufficient linear independence [17], [18]. Given a large number of serviced devices N and increasing large number of base-station antennas M, the number of user scheduling options is of order $\mathcal{O}(N^M)$. Thus, to optimally schedule users in resource-sharing groups (RSGs), one needs to examine the CSIs of each possible user grouping among potentially very large number of users in e.g., thousands.

Since MIMO user scheduling is a combinatorial, NP-hard problem, even for a moderate number of users (e.g. hundreds), it is difficult to exhaustively evaluate all possible user combinations

as MIMO user groups against one or more objectives. For example, some recent methods take advantage of proportional fairness (PF) and the determinant pairing algorithm (DPS) [19]–[21]. However, these schemes rely on exhaustive computation of spatial cross-correlations for various possible user groups, and as such require heavy computational workload. Typically, algorithms rely on heuristics when the user number becomes very large as in the case of IoT.

Other approaches exploit localized characteristics shared among small subsets of users. One proposal examines $N(N-1)/2$ pairwise CSI correlations among all N users and proposes to form user groups by setting a maximum correlation threshold [22], [23]. Other schemes such as [24], [25] directly group users of similar CSI covariance matrices. Nevertheless, information on pairwise or small user subsets fail to capture broad comprehensive characteristics on the entire user set. In the case of pairwise correlation, e.g., multiple users with low pairwise correlation in an MIMO user group may still suffer from significant interference accumulation. Localized approaches are also very sensitive to choices of manual parameter tuning and are harder to scale, as different choices might lead to drastically different performance as the user number changes or as the CSI models vary.

Recently, (machine) learning based approaches have been applied to a diverse array of difficult networking problems, including MAC user scheduling [26]–[29]. In fact, both supervised and unsupervised learning algorithms have found applications in wireless CSI characterization [26] that could be utilized in MIMO user scheduling. Machine learning is particularly attractive for large, NP-hard problems such as user scheduling that would require very high complexity to solve directly. In the context of wireless networking, supervised learning requires a rich labeled training set of diverse CSIs and correspondingly optimized scheduled user groups that attain strong performance. Such labeled training set must account for different system conditions such as number of antennas, number of users, wireless channel characteristics, noise levels, different power constraints, among others. However, it is not practically feasible to build such a huge set of optimum solutions. Ironically, supervised learning itself cannot be trained to find such optimum solutions for training.

In contrast, unsupervised learning explores underlying data features and characteristics without relying on labeled training set. Importantly in the context of large-scale user scheduling, unsupervised learning can effectively identify users with highly similar CSIs, as proposed for direct user grouping in downlink multi-cast [28], [29]. However, for the more general multi-user MIMO systems operating in either uplink MAC or downlink BC, scheduling users with similar CSI

leads to poor joint spatial diversity. Such outcome deviates from the original goal of diversity-based user scheduling, aimed at scheduling users with highly dissimilar CSIs into MIMO co-channel groups to promote spatial diversity and to mitigate mutual interference. Clearly, a direct application of traditional learning algorithms over CSI vectors is incompatible with the MIMO user scheduling task. Additionally, unsupervised learning is often performed in Euclidean space, but Euclidean distance of CSI does not correspond to spatial diversity as the latter is related to subspace span.

To this end, we develop a new dual-step approach to indirectly utilize the outcome of unsupervised learning in order to form co-channel user groups with high spatial diversity and low mutual interference. Recognizing the difficulty for unsupervised learning to classify dissimilar user CSIs, we *first* apply unsupervised learning to identify users with highly *similar* CSIs. To properly identify spatial CSIs that lead to large CCI, we map user CSIs to the complex Grassmannian manifold during learning. On this manifold, distances between CSIs relate directly to CSI correlation, regardless of phase and magnitude variations. We can apply *any solid* clustering algorithm over this geometry to cluster users that can generate high mutual interferences. Our *second step* schedule co-channel diversity users in MAC or BC systems by barring user groups with highly similar CSIs from Grassmannian manifold clustering. Applicable to any well established performance metrics such as maximum sum-rate or maximum signal-to-interference-and-noise ratio (SINR), our proposed MIMO user scheduling can improve spatial diversity and effectively mitigate CCI. The major contribution of this work is the development of this two-step strategy for MIMO user scheduling that (a) leverages unsupervised learning to promote spatial diversity of resource-sharing users, and (b) is completely based on the Grassmannian manifold.

We organize this manuscript as follows. Section II presents the signal model for user scheduling, CCI and system performance metrics. Section III introduces our general strategy for unsupervised user scheduling, Section IV presents numerical simulations in different scenarios, and finally Section V concludes the work.

Notations: In the following, vectors and matrices will be denoted with small and capital boldface letters, such as $\boldsymbol{z}$ and $\boldsymbol{Z}$ respectively. Sets are denoted with calligraphic capital letters. The transpose, element-wise complex conjugation and conjugate transpose are denoted by $\boldsymbol{z}^{\mathsf{T}}$, $\overline{\boldsymbol{z}}$ and $\boldsymbol{z}^{\mathsf{H}}$, respectively. The norm of vectors is denoted by $\|\cdot\|$. $\boldsymbol{0}_k$ and $\boldsymbol{I}_k$ represent the zero vector and the identity matrix of size k, respectively. Finally, $|\mathcal{S}|$ denotes the size of a set $\mathcal{S}$.

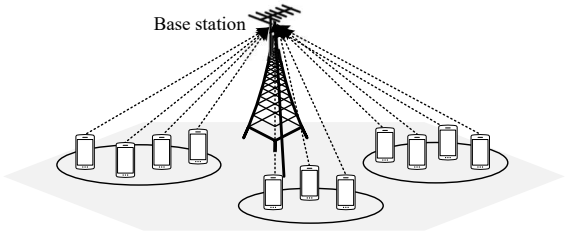


Fig. 1: Illustration of user scheduling for spatial multiplexing, where the BS has already assigned users into RSGs. Users in the same group deliver (MAC) or receive (BC) independent signals in a multiplexing mode on the same time-frequency resource.

II. System model and Problem Statement

Fig.1 depicts a wireless system with N single-antenna users and a single BS with M receiver antennas that has already assigned users into RSGs. We assume $N \geq M$ and a single-carrier system. Without loss of generality, we consider both uplink (MAC) and downlink (BC) MIMO systems that leverage spatial diversity to accommodate spectrum-sharing user groups allocated into distinct resources in multiple access strategies such as OFDMA and TDMA, among others.

Previous works on resource-sharing MIMO systems have studied optimal decoder (MAC) and precoder (BC) designs such that an RSG achieves channel capacity, such as the MMSE-SIC receiver [30] for MAC or dirty-paper precoding [31] with MMSE filters for BC. However, these designs were proposed with the premise that users have already been scheduled in MIMO user groups. To benefit from these exciting works in the literature, we investigate the scheduling problem that is a foundation step to MIMO precoding and rate maximization.

A. A General System Model

Let $\boldsymbol{h}_n \in \mathbb{C}^M$ be the uplink/downlink CSI vector of the n-th user. We assume the CSIs are random and independent from each other. We also assume that the CSI vectors are fixed within a transmission slot, which has been designed such that its duration is lower than the MIMO channel coherence time. Let G the total number of groups to be assigned, and let $\pi_{g,n} \in \{0, 1\}$ indicate whether the n-th user is scheduled in group $g \in \{1, \ldots, G\}$ exclusively, i.e.

$$\pi_{g,n} = \begin{cases} 1 & \text{the } n\text{-th user belongs to group } g \text{ only,} \\ 0 & \text{otherwise,} \end{cases} \tag{1}$$

and the set $\mathcal{S}_g = \{n | \pi_{g,n} = 1, n \in \{1, \ldots, N\}\}$ denotes the scheduled user set of the g-th group with cardinality $|\mathcal{S}_g|$.

In both MAC and BC, we assume the BS has access to perfect CSI of *all* active users, and therefore the BS shall manage user scheduling of all users into RSGs in each transmission slot. To ensure linear independence of user CSIs within each RSG, we impose that RSGs shall have at most M users. We use user SINR and group sum-rate as performance metrics in both uplink and downlink, and hence we derive a per-group signal model in both scenarios. Moreover, we adopt linear receivers/precoders for their straightforward implementation and low computational cost, according to typical IoT node characteristics.

B. Uplink (MAC) Scenario

In MAC uplink, let x_n denote the $n-$th user data symbol of zero mean and unit average power, i.e. $\mathbb{E}\left[|x_n|^2\right] = 1$. Let p_n represent the transmit power of the $n-$th user. At the BS, the $|\mathcal{S}_g|$ user signals from the scheduled MAC user group g arrive at the BS receiver through their respective channel responses $\{\boldsymbol{h}_n\}_{n\in\mathcal{S}_g}$, which is then decoded using a linear receiver $\boldsymbol{Q}_g$ such as the MRC, ZF and MMSE receivers [30], to generate the decoded signal vector

$$\boldsymbol{y}_g^{\mathrm{MAC}} = \boldsymbol{Q}_g^{\mathsf{H}}\Big(\sum\nolimits_{n\in\mathcal{S}_g} \boldsymbol{h}_n\sqrt{p_n}x_n + \boldsymbol{w}_g\Big) = \boldsymbol{Q}_g^{\mathsf{H}}\Big(\sum\nolimits_{n=1}^{N} \boldsymbol{h}_n\sqrt{p_n}x_n\pi_{g,n} + \boldsymbol{w}_g\Big) \tag{2}$$

where $\boldsymbol{w}_g \sim \mathcal{CN}(\mathbf{0}_M, \sigma^2\boldsymbol{I}_M)$ represents the AWGN vector corresponding to the resource block assigned to group g.

C. Downlink (BC) Scenario

In the case of BC, we operate under equal assumptions. However, the signal model changes as the single-antenna receivers will experience CCI but will not be aware of the CSI of other co-users. Furthermore, the received signal and noise are scalars, and the BS uses a linear, unitary beamforming precoder $\boldsymbol{z}_n$ for each user n, which can be selected as the weighted MMSE [32], MRT [33] or ZF precoder [34] among others. For notational simplicity, we abuse notation and we use g to also denote the group index of the group that contains user n. Therefore, the signal model for the signal that user n receives in BC mode with AWGN $w_n \sim \mathcal{CN}(0, \sigma^2)$ is:

$$y_n^{\mathrm{BC}} = \boldsymbol{h}_n^{\mathsf{T}}\sum\nolimits_{i\in\mathcal{S}_g} \boldsymbol{z}_i\sqrt{p_i}x_i + w_n = \boldsymbol{h}_n^{\mathsf{T}}\sum\nolimits_{i=1}^{N} \boldsymbol{z}_i\sqrt{p_i}x_i\pi_{g,i} + w_n. \tag{3}$$

D. Co-Channel Interference and Sum-Rate

The interference that user $n \in \mathcal{S}_g$ experiences is measured by their signal-to-interference-and-noise ratio or SINR. In the case of MAC, the SINR of user n is

$$\mathrm{SINR}_n^{\mathrm{MAC}} = \frac{p_n |\boldsymbol{Q}_g^{\mathsf{H}} \boldsymbol{h}_n|^2}{\sigma^2 + \sum_{i \in \mathcal{S}_g, i \neq n} p_i |\boldsymbol{Q}_g^{\mathsf{H}} \boldsymbol{h}_i|^2}. \tag{4}$$

Among several receiver designs, without loss of generality we adopt the MMSE receivers [30] for their straightforward implementation and to fully leverage spatial diversity. The MMSE design defines the g-th decoder matrix $\boldsymbol{Q}_g$ as

$$\boldsymbol{Q}_g = \Big(\sigma^2 \boldsymbol{I}_M + \sum_{i \in \mathcal{S}_g, i \neq n} p_i \boldsymbol{h}_i \boldsymbol{h}_i^{\mathsf{H}} \Big)^{-1} \boldsymbol{h}_n, \tag{5}$$

with a resulting SINR of

$$\mathrm{SINR}_n^{\mathrm{MAC}} = p_n \boldsymbol{h}_n^{\mathsf{H}} \Big(\sigma^2 \boldsymbol{I}_M + \sum_{i \in \mathcal{S}_g, i \neq n} p_i \boldsymbol{h}_i \boldsymbol{h}_i^{\mathsf{H}} \Big)^{-1} \boldsymbol{h}_n. \tag{6}$$

For BC, the SINR of user n corresponds to

$$\mathrm{SINR}_n^{\mathrm{BC}} = \frac{p_n |\boldsymbol{h}_n^{\mathsf{T}} \boldsymbol{z}_n|^2}{\sigma^2 + \sum_{i \in \mathcal{S}_g, i \neq n} p_i |\boldsymbol{h}_n^{\mathsf{T}} \boldsymbol{z}_i|^2}, \tag{7}$$

and, without loss of generality and for the sake of simplicity, we use the MRT precoders [33]

$$\boldsymbol{z}_i = \frac{\overline{\boldsymbol{h}_i}}{\|\boldsymbol{h}_i\|}, \ \forall n \in \{1, \ldots, N\}, \tag{8}$$

which yield a resulting SINR for user n of

$$\mathrm{SINR}_n^{\mathrm{BC}} = \frac{p_n \|\boldsymbol{h}_n\|^2}{\sigma^2 + \sum_{i \in \mathcal{S}_g, i \neq n} p_i |\boldsymbol{h}_n^{\mathsf{T}} \overline{\boldsymbol{h}_i}|^2 / \|\boldsymbol{h}_i\|^2}. \tag{9}$$

The normalized sum-rate of the g-th group is given by

$$R_g = \sum_{n \in \mathcal{S}_g} \log_2 \big(1 + \mathrm{SINR}_n \big), \tag{10}$$

with R_g^{MAC} and R_g^{BC} denoting the sum-rates for uplink and downlink, respectively, using the corresponding expressions for SINR.

E. The Scheduling Problem

Ideally, we aim to optimize the design of the indicator variables $\pi_{g,n}$ and user power allocation p_n to maximize the efficiency of resource usage in terms of sum rate, that is, maximizing the sum-rate of each RSG and minimizing the number of groups simultaneously. A mathematical formulation of this approach, valid for either MAC or BC scheduling, is

$$\mathcal{P}: \max_{G,\,\pi_{g,n},\,p_n} \quad \frac{1}{G}\sum\nolimits_{g=1}^{G} R_g \tag{11a}$$

$$\text{s.t.} \quad \sum\nolimits_{n=1}^{N} \pi_{g,n} \leq M,\ \forall g\,, \tag{11b}$$

$$\pi_{g,n} \in \{0,1\},\ \forall g\,, \forall n\,, \tag{11c}$$

$$\sum\nolimits_{n=1}^{N} p_n \pi_{g,n} \leq p_{\mathrm{BS}}^{\max},\ \forall g \text{ (BC)}, \tag{11d}$$

$$0 \leq p_n \leq p_{\mathrm{UE}}^{\max},\ \forall n \text{ (MAC)}. \tag{11e}$$

In Problem $\mathcal{P}$, constraint (11b) limits the number of MAC/BC users up to the number of BS antennas M without requiring further non-orthogonal multiple access. By design, each user belongs to one group only (11c). Additionally, in practical BC systems the BS transmission power is limited by $p_{\mathrm{BS}}^{\max}$ in every time slot and needs to be properly allocated (11d), whereas in MAC each user has a maximum transmission power $p_{\mathrm{UE}}^{\max}$ (11e). In order to further mitigate CCI, additional constraints can be considered when more design parameters are available, such as resource availability, cooperation in BC, individual rate requirements, among other criteria. Note that our formulation of problem $\mathcal{P}$ for maximizing spectrum efficiency can be modified as required to attain different objectives, and as such is without loss of generality. Other tractable performance metrics for MAC/BC user scheduling include the minimization of MSE [35], [36], weighted MSE [37], maximization of SINR [38], or minimization of BER [39], among others.

Regardless of the selected objective, $\mathcal{P}$ is NP-hard and shares similar complexity as general nonlinear mixed integer programming. To find the optimum MAC/BC user grouping $\pi_{g,n}^{\star}$, a direct exhaustive search method would need to evaluate all possible $\pi_{g,n}$ in terms of mean sum-rate (11a) to determine the optimum MAC/BC user grouping solution that achieves the best spectrum efficiency. However, the resulting search space is combinatorial even with a modest number of users and fixed G and p_n, and as such requires very high computational load.

Therefore, the challenge in MIMO user scheduling for massive wireless systems is to develop a low complexity and effective algorithm that can achieve high spectrum efficiency and low CCI

with relative independence of system parameters such as the total number of users, number of users within a group, BS antennas, channel realizations, etc.

F. Proposed Novel Solution Paradigm

Any solution to the scheduling challenge will essentially try to find MAC or BC groups such that all users in each group enjoy low CCI, or in other words, their CSIs are distinct enough in the spatial sense, while still incurring in reasonable computational cost. To attain this goal, such solution has to study the whole dataset, instead of looking at portions of it (such as pairwise relationships). Even then, the solution needs to measure dissimilarity, which is not well-defined in a general form and instead is variable, highly dependant on the particular realization of CSIs and the system itself. This leads to either trial-and-error approaches to define dissimilarity in particular scenarios, or the need of a dynamic metric of dissimilarity that accounts for system parameters and CSI variability in several different scenarios, which is both hard and impractical.

Nevertheless, the fast development of scalable solutions of challenging problems in the field of machine learning offers hope in tackling the scheduling problem. These techniques analyze all data points and are able to adapt to several changes in the dataset, known or not. Moreover, machine learning techniques have been thoroughly used across a large variety of computationally difficult problems with the goal of reducing complexity and/or runtime, and has offered novel perspectives and approaches in different aspects of wireless systems [26]–[29], [40].

Regrettably, both supervised and unsupervised learning cannot be directly applied to the scheduling problem. On one hand, supervised learning (which usually enjoys better performance) requires a labeled dataset for training, which in the context of CSI scheduling is nearly unfeasible: the large number of system parameters and possible channel characteristics implies the need for an incredibly large dataset to avoid sampling biases, and there are not have known ground-truth labels due to the fact that the optimum scheduling solution of a particular system is unknown due to the very nature of the scheduling problem.

On the other hand, unsupervised learning cannot be directly applied in user scheduling: a proper scheduling scheme will avoid grouping users with similar CSI, which diminishes performance and channel capacity, and conversely assigns users with dissimilar CSIs in RSGs to reduce CCI. However, unsupervised learning techniques excel at finding common features in a dataset in a efficient manner among data points, such as user CSI, without having to know beforehand which features to study and without the need of labeled datasets. This realization

leads to the main contribution of this manuscript: a general and scalable two-step strategy that uses unsupervised learning techniques to *first* identify in a global manner which users share similar CSI, to *then* exploit that information and define MAC/BC RSGs such that their users do not share spatial similarities.

In the following section, we present the details of our scheduling approach, valid for both MAC and BC systems, that tackles the inherent complexity of the scheduling problem without sacrificing performance.

III. Principled User Scheduling Through Unsupervised Learning

To accomplish our goal, we first examine CSI similarity and introduce a corresponding transformation to a geometric manifold that contains CSI vectors. Using this similarity measure, we directly apply unsupervised learning on active users to identify similar CSIs in terms of subspace span and form clusters of similar CSIs. Thus, users within each similarity cluster tend to exhibit strong CCI due to low CSI diversity (high similarity) such that no two users from a particular cluster should be jointly scheduled in an RSG. Based on the outcomes of unsupervised learning, we define a scheduling algorithm to group users from different CSI clusters into RSGs, thereby achieving high CSI diversity within a group to generate lower mutual interference and higher spectrum efficiency. Figure 2 illustrates our two-step strategy, and we further summarize our scheduling approach in Algorithm 1.

Algorithm 1 Scalable User Scheduling Strategy

Input: $\boldsymbol{h}_n \in \mathbb{C}^M$, $n \in \{1, \ldots, N\}$

Learning-based CSI Clustering:

1: Identify user CSI with high similarity (subspace span) through unsupervised learning;

Similarity-Assisted User Grouping:

2: Assign users from different clusters in RSGs for MAC/BC operation, such that no two users from the same CSI cluster are in any scheduled group, and further exploit clustering results in user selection.

A. Geometric Perspective of CSI Similarity

In our setting, channel similarity is directly related to the colinearity of user CSIs in spatial domain, or equivalent in subspace span. Hence, we start by examining the pairwise CSI

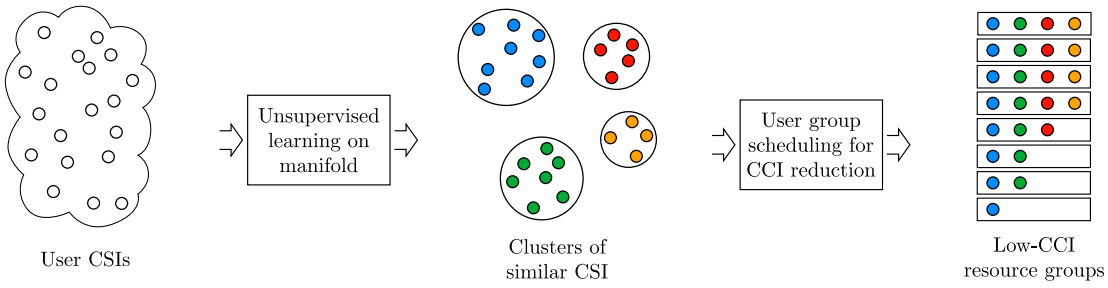


Fig. 2: Proposed user scheduling strategy based on unsupervised learning. In the *first step*, we employ unsupervised learning to classify user CSIs into clusters with strong similarity in the sense of subspace span. In the *second step*, we allocate users into RSGs that exploit CSI similarity such that not 2 users of the same cluster share resources.

correlation coefficient

$$\rho\left(\boldsymbol{h}_n, \boldsymbol{h}_i\right) = \frac{\left|\boldsymbol{h}_n^{\mathsf{H}} \boldsymbol{h}_i\right|}{\left\|\boldsymbol{h}_n\right\| \left\|\boldsymbol{h}_i\right\|}, \tag{12}$$

which determines the amount of CCI between two co-channel users. Note that (12) corresponds to the absolute value of the cosine similarity [41], and it also corresponds to the absolute value of the elements of the channel covariance matrix [42] normalized by channel gains. In particular, if two user CCIs are orthogonal, then there is zero CCI when scheduled in the same RSG. In practice, full CSI orthogonality is rare. One practical solution is to set a upper threshold to limit the pairwise CSI correlation coefficient within each RSG. The challenge lies in that setting such a threshold cannot guarantee the level of CCI among users in an RSG. First, the CCI among users in an RSG would vary depending on the magnitude of user CSIs, and on multi-lateral geometric relationship among CSIs. Second, direct user scheduling also depends on the user selection order considered for scheduling, whose optimization involves a combinatorial and computationally intensive process. To ensure overall system efficiency (11a), consistency, and fairness for both MAC and BC, we need to develop a consistent, simple, and scalable scheduling method to effectively limit CCI among users in a RSG for both BC and MAC scenarios.

Note that traditional unsupervised learning in Euclidean space is incompatible with identifying similar/dissimilar CSI vectors, as the Euclidean distance does not measure spatial correlation/diversity. Instead of using Euclidean distance, Eq.(12) shows that the spatial similarity or dissimilarity of CSI vectors is insensitive to phase rotations and/or magnitude variation of the individual CSI vectors, as

$$\rho\left(\boldsymbol{h}_n, \alpha \mathrm{e}^{i\theta} \boldsymbol{h}_i\right) = \rho\left(\boldsymbol{h}_n, \boldsymbol{h}_i\right) \; \forall \alpha \in \mathbb{R}/0, \; \theta \in [0, 2\pi).$$

To account for CCI invariance in a global manner, we can redefine the geometry of the space when analyzing CSIs, transforming from Euclidean space to a manifold geometry. Such transformation in unsupervised learning has been used to characterize the underlying low-dimension space of data [43]. In our case, by analyzing and clustering CSIs on a manifold that naturally measures the desired notion of diversity, the resulting clusters will effectively identify users that have highly similar CSIs and consequently strong CCI.

As explained, CSI correlation disregards the common phase and magnitude of each vector. Therefore, we need a manifold geometry invariant to magnitude and/or phase variations. Formally, we define an equivalence relation

$$\boldsymbol{h}_n \sim \boldsymbol{h}_i \quad \text{if} \quad \boldsymbol{h}_i = a\mathrm{e}^{i\theta}\boldsymbol{h}_n,\, a \in \mathbb{R}/\{0\},\, \theta \in [0, 2\pi), \tag{13}$$

which states that any two vectors that differ in magnitude and/or phase are considered the same. With Eq. (13) we can define an equivalence class for each CSI vector

$$[\boldsymbol{h}_i] = \big\{a\mathrm{e}^{i\theta}\boldsymbol{h}_i : \theta \in [0, 2\pi], a \in \mathbb{R}/\{0\}\big\} = \big\{\alpha\boldsymbol{h}_i : \alpha \in \mathbb{C}/\{0\}\big\}. \tag{14}$$

In other words, the equivalence class $[\boldsymbol{h}_i]$ is the complex line that passes through $\boldsymbol{h}_i$ and the origin. The set of all such lines is known as the complex Grassmann manifold of complex lines in $\mathbb{C}^M$, or Grassmannian, which we denote by $\mathrm{GR}(M, 1)$. This is a well-known geometry that has been extensively studied for both clustering and optimization [43]–[47]. For computation purposes, every equivalence class $[\boldsymbol{h}_n] \in \mathrm{GR}(M, 1)$ is represented by its unit vector $\boldsymbol{h}_n\|\boldsymbol{h}_n\|^{-1}$ on behalf of all the points contained in the class.

To cluster data points in a manifold, we need to define: (1) a Riemannian distance that measures the space; (2) the tangent spaces, which are linear spaces that approximate the manifold in a neighborhood of a particular point; and (3) geodesics, which are the minimal smooth curves in the manifold that connect two of its points. The complex Grassmannian $\mathrm{GR}(M, 1)$ can be endowed with the following distance function:

$$\mathrm{dist}\big([\boldsymbol{h}_n], [\boldsymbol{h}_i]\big) = \arccos\Big(\frac{|\boldsymbol{h}_n^{\mathsf{H}}\boldsymbol{h}_i|}{\|\boldsymbol{h}_n\|\|\boldsymbol{h}_i\|}\Big) = \arccos\big(\rho(\boldsymbol{h}_n, \boldsymbol{h}_i)\big).$$

Note that this distance is a function of the CSI correlation, and as such, is invariant to scale and phase variations as intended.

The tangent space $\mathrm{T}_{[\boldsymbol{h}_n]}\mathrm{GR}(M, 1)$ is a linear space that contains the tangent directions of all 1-dimensional curves on the manifold passing through $[\boldsymbol{h}_n]$. In the case of $\mathrm{GR}(M, 1)$, we have

$$\mathrm{T}_{[\boldsymbol{h}_n]}\mathrm{GR}(M, 1) = \{\boldsymbol{v} \in \mathbb{C}^M : \boldsymbol{h}_n^{\mathsf{H}}\boldsymbol{v} = 0\}. \tag{15}$$

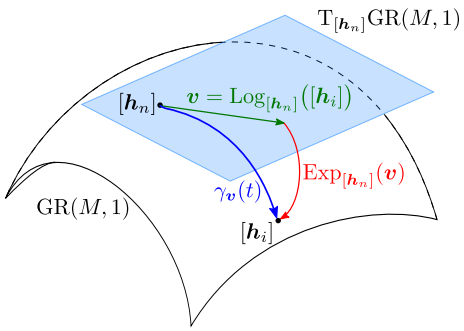


Fig. 3: Depiction of Grassmannian manifold: we show the tangent space at $[\boldsymbol{h}_n]$, the geodesic connecting $[\boldsymbol{h}_n]$ and $[\boldsymbol{h}_i]$, the corresponding exponential and logarithm maps, and their relationships.

We define a Riemannian metric for the linear $\mathrm{T}_{[\boldsymbol{h}_n]}\mathrm{GR}(M,1)$:

$$\langle \boldsymbol{u},\boldsymbol{v}\rangle_{[\boldsymbol{h}_n]} = \mathrm{Re}\big(\boldsymbol{u}^{\mathsf{H}}\boldsymbol{v}\big),\ \ \boldsymbol{u},\boldsymbol{v}\in \mathrm{T}_{[\boldsymbol{h}_n]}\mathrm{GR}(M,1) \tag{16}$$

that induces a norm $\|\boldsymbol{v}\|_{[\boldsymbol{h}_n]} = \sqrt{\langle \boldsymbol{v},\boldsymbol{v}\rangle_{[\boldsymbol{h}_n]}}$ for tangent vectors $\boldsymbol{v}\in \mathrm{T}_{[\boldsymbol{h}_n]}\mathrm{GR}(M,1)$.

Finally, we characterize geodesics connecting two points in $\mathrm{GR}(M,1)$. Formally, we define $\gamma(t)$ as the geodesic from the starting point $[\boldsymbol{h}_n]=\gamma(0)$ reaching the point $[\boldsymbol{h}_i]$ at $\gamma(1)$ at $t=1$. This scaling implies that the geodesic has a defined initial velocity $\boldsymbol{v}=\gamma'(0)$. By construction, $\boldsymbol{v}\in \mathrm{T}_{[\boldsymbol{h}_n]}\mathrm{GR}(M,1)$, which can be computed with the *logarithm map*:

$$\mathrm{Log}_{[\boldsymbol{h}_n]}\big([\boldsymbol{h}_i]\big) = \frac{\boldsymbol{u}}{\|\boldsymbol{u}\|}\arctan\big(\|\boldsymbol{u}\|\big),\ \ \boldsymbol{u} = \frac{\|\boldsymbol{h}_n\|\boldsymbol{h}_i}{\boldsymbol{h}_n^{\mathsf{H}}\boldsymbol{h}_i} - \frac{\boldsymbol{h}_n}{\|\boldsymbol{h}_n\|}.$$

Conversely, for a geodesic $\gamma_{\boldsymbol{v}}(t)$ starting at $[\boldsymbol{h}_n]$ and with initial velocity $\boldsymbol{v}\in \mathrm{T}_{[\boldsymbol{h}_n]}\mathrm{GR}(M,1)$, the *exponential map* yields the point $\gamma_{\boldsymbol{v}}(1)$, and is given by

$$\mathrm{Exp}_{[\boldsymbol{h}_n]}\big(\boldsymbol{v}\big) = \frac{\boldsymbol{h}_n}{\|\boldsymbol{h}_n\|}\cos\big(\|\boldsymbol{v}\|_{[\boldsymbol{h}_n]}\big) + \frac{\boldsymbol{v}}{\|\boldsymbol{v}\|_{[\boldsymbol{h}_n]}}\sin\big(\|\boldsymbol{v}\|_{[\boldsymbol{h}_n]}\big).$$

We therefore have $[\boldsymbol{h}_i] = \mathrm{Exp}_{[\boldsymbol{h}_n]}\big(\mathrm{Log}_{[\boldsymbol{h}_n]}\big([\boldsymbol{h}_i]\big)\big)$. We can use both maps above to move on the manifold. Note that these expressions are equivalent to the logarithm and exponential maps for general Grassmannians $\mathrm{GR}(M,p)$ based on singular value decompositions [48], but simplified for the particular case of $\mathrm{GR}(M,1)$. Fig. 3 visually depicts the Grassmannian manifold discussed above and the relationship among tangent space, geodesic, logarithm and exponential map.

B. Unsupervised CSI Clustering

Among the plethora of unsupervised learning algorithms, e.g., [49], that are generally all useful in our user scheduling paradigm, we consider two simple and well-known data clustering

methods: K-means clustering and agglomerative hierarchical clustering [50]. We adapt both unsupervised learning algorithms in the first step of manifold CSI clustering. Our goal is to classify N users into K clusters $\{\mathcal{C}_1, \cdots, \mathcal{C}_K\}$ based on the available CSI $\{\boldsymbol{h}_i\}_{i=1}^N$ at the scheduling server such that users in each cluster exhibit high CSI similarity in $\mathrm{GR}(M,1)$.

1) Grassmannian K-means (GKM) Clustering: The basic K-means algorithm applies a greedy iterative approach to find a data partition that minimizes the distance between cluster members and their respective cluster centers. At the t-th iteration, the center of the k-th cluster $\mathcal{C}_k^t$ is defined by $\boldsymbol{\mu}_k^t \in \mathbb{C}^M$, which in Euclidean space is given by

$$\boldsymbol{\mu}_k^t = \frac{1}{|\mathcal{C}_k^t|} \sum_{n \in \mathcal{C}_k^t} \boldsymbol{h}_n. \tag{17}$$

In case of manifolds, the cluster centers are given by the intrinsic mean of cluster members,

$$\boldsymbol{\mu}_k^t = \underset{[\boldsymbol{u}] \in \mathrm{GR}(M,1)}{\arg\min} \sum_{n \in \mathcal{C}_k^t} \mathrm{dist}\big([\boldsymbol{h}_n], [\boldsymbol{u}]\big), \tag{18}$$

which has no closed-form solution as it depends on reference points, which are not fixed. The computation of the intrinsic mean is shown in Algorithm 2, where we use the unit-vector representatives of the equivalence classes for computation.

Algorithm 2 Intrinsic Mean for Cluster k in $\mathrm{GR}(M,1)$

Input: $[\boldsymbol{h}_n] \in \mathcal{C}_k^t$, threshold ϵ, maximum number of iterations T_{im}

1: Initialize $t = 1$, $[\boldsymbol{u}] = [\boldsymbol{h}_i]$ for a random i in the cluster
2: **while** $t \leq T_{\mathrm{im}}$ or $\|\boldsymbol{v}\|_{\boldsymbol{u}} \geq \epsilon$ **do**
3: Compute tangent vector $\boldsymbol{v} = |\mathcal{C}_k^t|^{-1} \sum_n \mathrm{Log}_{[\boldsymbol{u}]}\big([\boldsymbol{h}_n]\big)$
4: Update $[\boldsymbol{u}] = \mathrm{Exp}_{[\boldsymbol{u}]}(\boldsymbol{v})$
5: Set $t = t + 1$
6: Return $[\boldsymbol{\mu}_k^t] = [\boldsymbol{u}]$

At the t-th iteration, each user is assigned to a cluster $\mathcal{C}_{k^\star}$ based on the center that is closest to the user, that is,

$$k^\star = \arg\min_k \mathrm{dist}\big([\boldsymbol{h}_n], [\boldsymbol{\mu}_k^t]\big). \tag{19}$$

The K cluster centers are then updated. The clustering and center update steps continue until all clusters stay the same (or any other stopping criteria). It is well-known that performance of K-means could suffer due to poor initialization, and hence we initialize using K-means++

to mitigate this effect [51]. Our implementation of Grassmannian K-means is summarized in Algorithm 3. Note that Algorithms 2 and 3 are also derived in previous work [46], [47], but have been tailored for the particular case of $\mathrm{GR}(M,1)$ by using the exponential and logarithm maps derived in Section III-A.

Algorithm 3 Grassmannian K-means

Input: $\boldsymbol{h}_n \in \mathbb{C}^M$, $n \in \{1,\dots,N\}$, intrinsic mean parameters ϵ and T_{im}

Normalize CSIs to obtain Grassmannian representatives
Set $t=0$ and $\mathcal{C}_k^0 = \emptyset \ \forall k$
Select initial cluster centers according to K-means++ using the Grassmannian distance
while $\mathcal{C}_k^t \neq \mathcal{C}_k^{t-1}$ for any k **do**
    Set $t = t+1$ and $\mathcal{C}_k^t = \emptyset \ \forall k$
    for each user n **do**
        Find the center $[\boldsymbol{\mu}_k]$ closest to $[\boldsymbol{h}_n]$
        Assign user n to $\mathcal{C}_k^t$
    for each cluster k **do**
        Update cluster center $[\boldsymbol{\mu}_k]$ using Algorithm 2

2) Agglomerative Hierarchical Clustering (AHP): This bottom-up hierarchical clustering approach begins by treating each data point as a single point cluster. It proceeds to successively merge the most similar cluster pairs until reaching the target number of clusters using a "linkage" rule to define the distances among merged pairs. Here, we use the complete linkage rule [28], i.e., at the t-th agglomeration, the similarity measure between clusters $\mathcal{C}_k^t$ and $\mathcal{C}_j^t$ is

$$d\left(\mathcal{C}_k^t, \mathcal{C}_j^t\right) = \max_{\boldsymbol{h}_m \in \mathcal{C}_k^t,\, \boldsymbol{h}_n \in \mathcal{C}_j^t} \mathrm{dist}\big([\boldsymbol{h}_m], [\boldsymbol{h}_n]\big). \tag{20}$$

In particular, the linkage between two single-user clusters is $d\left(\mathcal{C}_m^t, \mathcal{C}_n^t\right) = \mathrm{dist}\big([\boldsymbol{h}_m], [\boldsymbol{h}_n]\big)$. Given a set of clusters $\{\mathcal{C}_1^t, \cdots, \mathcal{C}_{K'}^t\}$, where $K' \geq K$, at each iteration, we determine the most similar pairs of clusters according to the linkage rule (20). After merging the two closest clusters, the process is repeated on the new set of clusters until the target number of clusters K is reached.

Note that in the context of our scheduling framework assisted by unsupervised learning in Grassmannian manifold, any effective clustering approach is a valid option. We only focus on these two simpler approaches for their low complexity and ease of exposition.

It is also important to note that our method shares some similarities with spatial division schemes such as JSDM [52], [53]. JSDM attempts spatial division to allocate spatially correlated users into RSGs in MU-MIMO (BC), based on known channel covariance matrices. In particular, one JSDM variation implements K-means using the chordal distance [54], which is a valid Grassmannian distance. Hence, it is a type of GKM that is later exploited for scheduling. However, there are also major differences. First, JDSM is designed under a particular channel model in FDD. It assumes previous knowledge of spatial correlation of CSIs and requires precoding and pre-beamforming, whereas our approach is valid for general CSI realizations with no prior information and simple receiver/precoder designs for scalability. Second, JSDM uses several precoding and pre-beamforming matrices, a large number of eigendecompositions, and solving joint diagonalization problems (and eventually estimating channel correlation matrices), which can require significantly higher complexity.

C. CSI-Based User Scheduling

1) Direct Greedy Scheduling: One way to control CCI among users scheduled in the same group is to apply a simple greedy algorithm to form RSGs. This direct greedy method can be used a basic benchmark. Starting from a user group of one random user, we can consider each new user by examining its pairwise CSI correlation with all users in the group against a set threshold β, and only add the new user if each pairwise correlation is below β until the group size reaches M. We can then continue to schedule additional groups. This **direct greedy scheduling**, which we denote DS, does not rely on any supervised learning. Its scheduling results would vary significantly according to the order of the users being considered during scheduling.

2) User Scheduling with Unsupervised Learning: MIMO user scheduling optimization can consider different performance criteria. However, CCI among users scheduled for the same RSG should always be reduced for any sensible performance metric. Based on the outcomes of CSI clustering, users within the same cluster have highly similar CSIs in terms of strong pairwise correlation coefficient, which can lead to strong CCI and challenge the receiving accuracy. From this perspective, users from different clusters are dissimilar and should induce low CCI, and thus are good candidates to be scheduled for MIMO resource sharing.

Our proposed GKM and AHP algorithms exploit the outcomes from CSI learning in the form of CSI clusters. Since there are multiple CSI clusters, our proposed GKM scheduling would compute the inter-cluster distances and sort clusters in descending order of minimum

inter-cluster distance. We can then start GKM scheduling by forming user scheduling groups by considering clusters that are as far apart as possible to contain CCI. In the case of AHP, we apply cluster merging from the smallest cluster sizes.

D. Power Control in MAC Scheduling

CSI gains and power control must be considered differently in MAC and BC scheduling systems. In MAC, different receiver designs will benefit from different strategies: joint MMSE receiver benefits from CSIs with similar gains, whereas interference cancellation (e.g. SIC) receiver thrives when CSIs have large gain differences. In practice, power control in MAC plays an important role to mitigate the near-far problem. It is well known that optimal power control is achieved by waterfilling with respect to a target interference and noise level [30]. However, it can be hard to accurately apply power control at the scheduling stage for a large number of distinct groups. Thus, we can consider two scenarios: (a) equal power transmission, where maximum user transmit power $p_n = p_{\mathrm{UE}}^{\max}$ is used for all n without power control, despite having different channel gains $\|\boldsymbol{h}_n\|$ (MAC-U); (b) effective power control such that $p_n\|\boldsymbol{h}_n\|^2$ is nearly constant at the receiver (MAC-P).

Consider each user signal quality in MAC. Note that using an MMSE receiver with σ^2 being the noise power,

$$\mathrm{SINR}_n^{\mathrm{MAC}} = \frac{\boldsymbol{h}_n^{\mathsf{H}}}{\|\boldsymbol{h}_n\|^2}\left(\frac{\sigma^2\boldsymbol{I}}{p_n\|\boldsymbol{h}_n\|^2} + \sum_{i\in\mathcal{S}_g, i\neq n}\frac{p_i\boldsymbol{h}_i\boldsymbol{h}_i^{\mathsf{H}}}{p_n\|\boldsymbol{h}_n\|^2}\right)^{-1}\boldsymbol{h}_n,$$

which means that SINR depends on all pairwise correlations of users within a group and the ratio of their received powers. Hence, the MMSE receiver benefits when the users within a group have similar received power $p_n\|\boldsymbol{h}_n\|^2$, and thus the received power ratios are close to 1. Such power ratios have minimum near-far effect and more consistent performance. These power ratios are often achieved under power control. Hence, we define the following grouping rule for MAC-P:

$$\varphi_{\mathrm{P}}^{\mathrm{MAC}}(\boldsymbol{h}_u, \mathcal{S}_g) = \begin{cases} 1 & \rho(\boldsymbol{h}_u, \boldsymbol{h}_\ell) \leq \beta \ \ \forall \ell \in \mathcal{S}_g \ \ \wedge \ \ |\mathcal{S}_g| < M, \\ 0 & \text{otherwise.} \end{cases} \tag{21}$$

E. MAC Scheduling without Power Control

In large scale systems such as IoT deployment, power control may not be practical. Without power control (MAC-U), our proposed scheduling algorithm shall attempt to reduce CCI by forming RSGs of similar channel gain and low similarity.

Specifically, we partition CSI gains of all N users into B levels. Users scheduled in group $\mathcal{S}_g$ must have CSI belonging to the same gain partition, denoted $b(\mathcal{S}_g)$. Hence, we modify the grouping rule $\varphi_{\mathrm{P}}^{\mathrm{MAC}}$ to include this additional criteria:

$$\varphi_{\mathrm{U}}^{\mathrm{MAC}}(\boldsymbol{h}_u, \mathcal{S}_g) = \begin{cases} 1 & \rho(\boldsymbol{h}_u, \boldsymbol{h}_\ell) \leq \beta \ \ \forall \ell \in \mathcal{S}_g \ \ \wedge \ \ |\mathcal{S}_g| < M \ \ \wedge \ \ \|\boldsymbol{h}_u\| \in b(\mathcal{S}_g), \\ 0 & \text{otherwise.} \end{cases} \tag{22}$$

F. BC Scheduling for Low Complexity Transceivers

Practical individual receivers in BC systems do not share CSI information. For massive deployment, dirty paper coding (DPC) [55] is also challenging to implement practically. Our user scheduling will target low complexity transceivers that only utilize local CSI. Therefore, power control for scheduled user groups could prove useful. We consider a simple power control by allocating uniform transmit power among BC group members, i.e. $p_n = p_{\mathrm{BS}}^{\max}|\mathcal{S}_g|^{-1}$. Other simple schemes could be applied, e.g. allocating power such that users within a group exhibit close to identical received signal power $p_n\|\boldsymbol{h}_i\|^2$.

Furthermore, users with weaker CSI gain experience lower SNR. To compensate, our BC scheduling algorithm considers the CSI gain and allocate fewer users of similarly low CSI gain into an RSG to maintain sufficiently high SINR. As a simple two-tier implementation example, we shall partition downlink CSI gains into two levels with a threshold δ. We assign users with weaker CSI gains below δ into weaker CSI groups, up to a maximum of E users in such groups. Conversely, we assign users with stronger CSI gains above δ into stronger CSI groups, up to a maximum of D users in such groups where $D \geq E$.

The corresponding grouping rule is:

$$\varphi^{\mathrm{BC}}(\boldsymbol{h}_u, \mathcal{S}_g) = \begin{cases} 1 & \rho(\boldsymbol{h}_u, \boldsymbol{h}_\ell) \leq \beta \ \ \wedge \ \ |\mathcal{S}_g| < E \ \ \wedge \ \ \|\boldsymbol{h}_u\|, \|\boldsymbol{h}_\ell\| \leq \delta, \ \ \forall \ell \in \mathcal{S}_g, \\ 1 & \rho(\boldsymbol{h}_u, \boldsymbol{h}_\ell) \leq \beta \ \ \wedge \ \ |\mathcal{S}_g| < D \ \ \wedge \ \ \|\boldsymbol{h}_u\|, \|\boldsymbol{h}_\ell\| > \delta, \ \ \forall \ell \in \mathcal{S}_g \\ 0 & \text{otherwise.} \end{cases} \tag{23}$$

To summarize, Fig.4 depicts a flowchart of our proposed user scheduling principle based on unsupervised learning outcomes. The application for uplink MAC-P, MAC-U or downlink BC depends on the choice of the selection rule $\varphi(\boldsymbol{h}_u, \mathcal{S}_g)$ according to (21), (22) or (23), respectively. Moreover, DS uses the same grouping rules for benchmarking purposes. Furthermore, note that if $\beta = 1$, grouping rules dismiss CCI thresholding at the scheduling level, and particularly DS is effectively implementing traditional scheduling schemes. For MAC-P, DS with $\beta = 1$

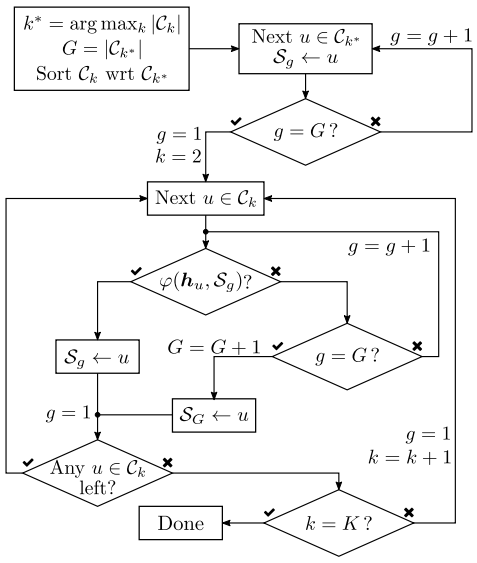


Fig. 4: Flowchart of the proposed scheduling scheme, using rule (21) for MAC-P, (22) for MAC-U, (23) for BC.

corresponds to a round robin (RR) strategy as all users get equal opportunity to be assigned into groups. For MAC-U and BC, this setting corresponds to a proportional fair (PF) strategy: users are grouped only with users that have similar channel power, having equal opportunity within these power-based user subsets, and the BS serves all users fairly because no user is subject to severe interference produced by a user with significantly better channel conditions.

More generally, variants of the proposed scheduling algorithm can exploit both the knowledge of CSI similarity and CSI gains in different ways without changing significantly the underlying methodology supporting our proposed strategy. Even more broadly, the unsupervised learning paradigm can also accommodate CSI features based on system performance and design choices.

G. Complexity Analysis

1) Clustering: First, we determine the complexity of Grassmannian K-means as follows. The CSI normalization step has a cost of $\mathcal{O}(MN)$. The Grassmannian distance has a cost of $\mathcal{O}(M)$, and thus the k-means++ initialization has a cost of $\mathcal{O}(KMN)$. Each iteration of the algorithm consists on two steps. First, assigning N users to K clusters, with a total cost of $\mathcal{O}(KMN)$. Then, the cluster center update for K clusters, which requires the computation of N logarithm maps and K exponential maps, both operations with a cost of $\mathcal{O}(M)$. This process is

iterative, but in practice takes only a few intrinsic mean iterations and the cost of center updates is $\mathcal{O}(KMN)$. Accordingly, the total cost for t iterations of GKM is $\mathcal{O}(tKMN)$.

In the case of AHP, optimal implementations depend on the linkage criteria [56]. When considering complete linkage, it is known that the optimal algorithm has complexity $\mathcal{O}(N^2)$ with respect to similarity comparisons. Furthermore, the pairwise CSI correlation has a total cost of $\mathcal{O}(MN^2)$ as there are $N(N-1)$ pairwise computations, and thus the complexity of this clustering approach is $\mathcal{O}(MN^2)$.

2) User Grouping: All user grouping approaches (similarity-assisted and DS) exploit pairwise correlation information, which has cost $\mathcal{O}(M)$. For the similarity-assisted approaches (GKM and AHP), the algorithm checks $K-1$ clusters with a total of $N' = N - |\mathcal{C}_k^*|$ users, which we can approximate in average with an uniform partition as $N' = N - N/K = N(K-1)/K$. In the case of DS, there is no clustering information, and hence $N-1$ users need to be tested for assignment. Hence, the computation complexity of similarity-assisted methods and DS is of order $\mathcal{O}(MN)$. Of course, in average we expect to observe some computational gains in similarity-assisted grouping, depending on system parameters and selected threshold, but cannot guarantee that they are going to be dramatically significant.

3) Total Computational Cost: In certain mobile scenarios when channels vary rapidly, our scheduling algorithm may be applied periodically, like any existing scheduling scheme. Periodic scheduling over a period of L frames rescheduling once every L frames. Our simple and low cost scheduling algorithm is advantageous since the computation complexity of periodic scheduling only grows at the rate of $1/L$ as L shortens because of channel fading.

IV. Numerical Experiments

In this section we test our proposed scheduling principle based on unsupervised learning. Considering a variety of user service needs and CSI properties, we provide a simple MIMO system model for both uplink and downlink applications.

We examined one BS equipped with $M = 8$ antennas, serving N single-antenna users. The mobile user CSI in uplink/downlink is assumed to be static within one transmission frame, and is modeled as random vectors that incorporate both shadowing and Rayleigh fading. A circularly complex normal vector of size M represents MIMO Rayleigh fading. The shadowing effect is modeled as a power gain that follows a lognormal distribution, with zero mean and standard deviation σ_L of 3dB in logarithmic scale. Additive channel noise is included in both

uplink/downlink directions, corresponding to 20dB of average SNR. The BS has access to user CSI in every scheduling period. Scheduling occurs for each transmission frame independently, which is consistent with common standards as 4G/5G [57] and 802.11ac [58] which provide mechanisms for explicit CSI feedback.

We generate 100 different channels and perform 10 runs per channel realization in our Monte-Carlo simulations. All tests are performed in MATLAB using a 64-bit Windows PC with an i7-7700K processor and 32GB RAM.

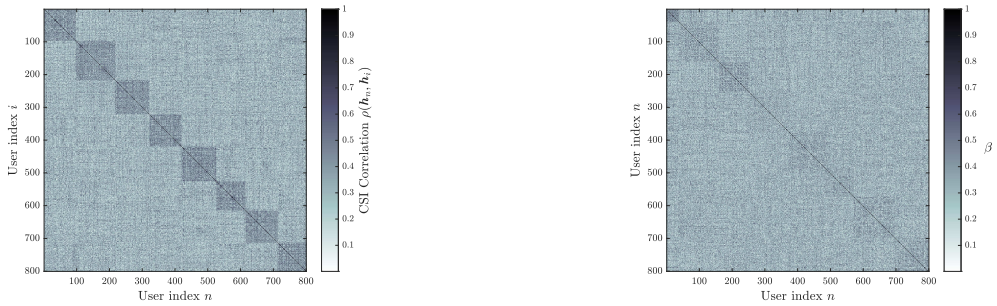
A. Performance Metrics

1) Clustering: Recall that both K-means and AHP clustering algorithms use a pre-determined cluster number K. There are several metrics that can help determine the number of clusters, such as the Silhouette value [59], the Krzanowski-Lai index [60], and the Hartigan index [61]. However, choosing the best K is still a non-trivial problem which is problem dependent and does not have a best consensus solution [62]. Moreover, there is no clear indication that the best K in terms of clustering criteria would lead to better scheduling performance. Regardless, we know that the nature of the MAC and BC systems imply that RSGs cannot have more than M users, since more users necessarily lead to more ill-conditioned group CSI matrix, impacting performance significantly. Additionally, choosing $K > M$ implies that the scheduling rule also needs to consider additional criteria to determine which cluster is used to pick users from, with no obvious principled option. Hence, we set the number of clusters to be $K \leq M$ heuristically, depending on the scenario, and show the pairwise correlation matrices obtained with both clustering methods.

2) CCI Evaluation: CCI leads a loss of signal-to-interference and noise ratio (SINR) in reference to SNR. To evaluate the efficacy of user scheduling, we compute the loss in SINR that each user experiences. In other words, we normalize the SINR of user n by its SNR if it were scheduled without co-channel users (a singleton group), such that it does not share resources and is only hampered by noise. Let p_n be the transmit power of the n-th user and σ^2 be the additive channel noise variance. This SINR loss is given by

$$10 \log_{10} \left(\frac{\mathrm{SINR}_n}{p_n \|\boldsymbol{h}_n\|^2 / \sigma^2} \right) \text{ dB}. \tag{24}$$

Thus, 0dB SINR loss denotes fully orthogonal CSIs among co-channel users and zero CCI.



(a) GKM. (b) AHP.

Fig. 5: Example of of pairwise CSI correlation coefficient matrices after clustering $N = 800$ users with $K = 8$. Larger correlation values are darker. (a) Grassmannian K-means. (b) Hierarchical Clustering.

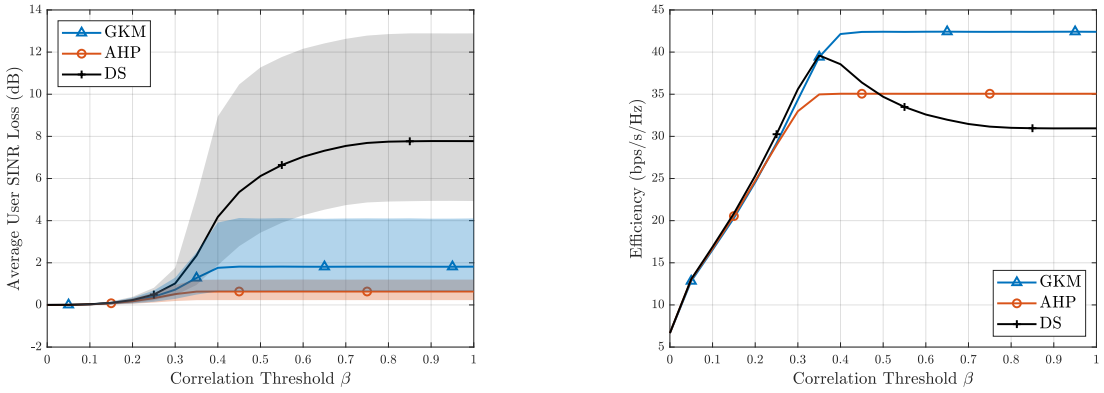
3) Resource Efficiency: To compare how well each scheduling method utilizes limited spectrum resources, we consider the overall MIMO spectrum efficiency by assuming that each scheduled MIMO user group is allocated the same bandwidth (or same amount of spectrum resource). In such case, the spectrum efficiency of the scheduling can be measured by averaging the achieved sum-rate of all groups over G total RSGs: $G^{-1} \sum_{g=1}^{G} R_g$.

4) Runtime: As a proxy for computational complexity, in all simulations we store the "wall-clock" time for successful execution of the algorithms under test. Naturally, runtime often depends on computer, platform and code implementation. Thus, runtime by itself it may not fully capture the complexity of scheduling algorithms. Hence, we apply the same built-in functions, establish similar program structure and implementation for all algorithms under test to mitigate platform biases. Additionally, we also test scheduling for different number of users N to compare the scalability of the algorithms.

B. Uplink MAC MIMO Performance

In MAC, we set the number of clusters to $K = 8$ and we adopt linear MMSE receiver based on CSI of all received user signals in an RSG. This choice of K ensures that the scheduling process will attempt to utilize all spatial degrees of freedom for high spectrum efficiency.

1) Clustering Outcomes: Fig. 5 shows the pairwise correlation matrix of $N = 800$ user CSIs after GKM and AHP clustering. Using grayscale of $[0, 1]$, larger values correspond to darker colors. An ideal clustering outcome should show 8 blocks of darker squares along the diagonal. GKM yields user clusters that show strong within-cluster similarity to be used later in scheduling.



(a) Average user SINR loss.

(b) Average efficiency.

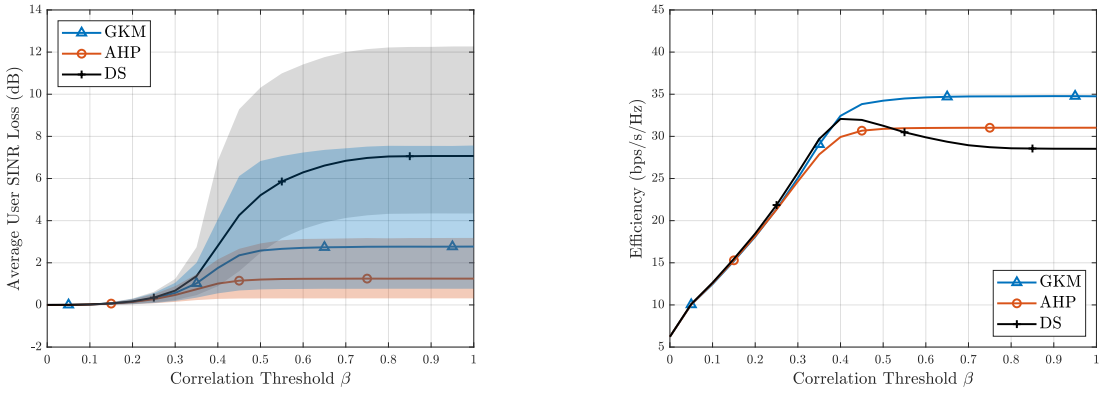
Fig. 6: Performance of scheduling algorithms in MAC-P. Here, $M = K = 8$ and $N = 800$. Solid lines represent averages over channel realizations, and shaded areas show all values within the 10th and 90th percentiles.

All 8 clusters contain many pairs that exhibit high spatial correlation. AHP shows less defined clusters: each cluster exhibits less highly correlated pairs, or in other words, users have weaker CSI correlation (i.e. weaker similarity) within each cluster.

2) Scheduling Performance: To systematically evaluate the effect of exploiting similarity identified during clustering, we analyze our approach in MAC by considering the two scenarios described in the previous section: (1) MAC-P (perfect power control), where each received user signal has unit power, i.e. $p_n \|\boldsymbol{h}_n\|^2 = 1 \, \forall n \in \{1, \ldots, N\}$; and (2) MAC-U (no power control), where user CSI powers follow a log-normal distribution with $\sigma_L = 3$dB, and $p_n = p_{\text{UE}}^{\max} = 1 \, \forall n$.

We first test the MAC-P scenario, which helps isolate the benefits of unsupervised learning without the effect of variable receive powers. We consider three scheduling algorithms: the proposed GKM scheduling, AHP scheduling, and DS scheduling. For these three methods, Fig. 6a shows the comparison of SINR mean and SINR distribution (10% to 90% percentiles) of the resulting user SINR loss. The corresponding spectral efficiency is shown in Fig.6b.

From our test results, when β is set low, only very small amount of CCI is tolerated. In such cases, scheduling performance continues to be dominated by channel noise (i.e., SNR) and allowing little if any resource-sharing. Thus it is natural that, for $\beta < 0.25$, all three scheduling algorithms exhibit little SINR loss and relatively low spectrum efficiency. As we increase β, the spectrum efficiency starts go grow for all algorithms, as shown in Fig.6b. Eventually, spectrum efficiency of both GKM and AHP saturate. More specifically, GKM achieves higher spectrum



(a) Average user SINR loss.

(b) Average efficiency.

Fig. 7: Performance of scheduling algorithms in MAC-U. Here, $M = K = 8$ and $N = 800$. Solid lines depict averages over channel realizations, and shaded areas show all values within the 10th and 90th percentiles.

efficiency but also larger SINR loss. AHP on the other hand, achieves lower spectrum efficiency but also lower SINR loss. In terms of SINR loss, both GKM and AHP exhibit very modest amount of SINR loss that saturates at 2dB and 1dB on average, respectively. These tests show that MIMO user scheduling can achieve spectrum efficiency and SINR loss tradeoff, with GKM being more efficient in spectrum utilization.

For DS, spectrum efficiency would peak at $\beta = 0.35$ before decreasing with increasing β. For $\beta = 1$, DS is a RR scheme that allocates users into groups with equal chance, with a large average SINR loss and lower efficiency. The spread of SINR is also large for DS, with some users experimenting more than -12dB of SINR loss. Both GKM and AHP have a tighter spread, which means most of the users will only experience 5dB of loss at most for GKM, and 2dB for AHP. This demonstrates the failure of DS in grouping users of low mutual CCI to achieve good tradeoff between spectrum efficiency and SINR loss.

We next test the more practical case of unequal CSI gains in a network without power control, MAC-U. Against users of unequal CSI gain, for the scheduling rule $\varphi_{\mathrm{U}}^{\mathrm{MAC}}$ we partition users according to their CSI powers uniformly with power interval of 3 dB such that users are divided according to power level boundaries of $\ldots, -4.5, -1.5, 1.5, 4.5, \ldots$ in dB. Figs.7a and 7b show the SINR loss and average efficiency of all methods in MAC-U.

Again we observe that for smaller β, channel noise dominates the scheduling performance and all algorithms attain similar performance. With increasing β, users experience more SINR losses

as more of them share resources, while the system enjoys the corresponding improvement in spectrum efficiency. The efficiency of GKM and AHP saturate for large β, with a modest increase of 1dB of SINR loss in average compared to MAC-U. Again, GKM achieves better performance with a small tradeoff in user SINR, compared to AHP that grants better user SINR at the expense of reduced efficiency. These tests confirm that similarity-assisted methods improve MIMO user scheduling, even when considering random channel gains and uniform power allocation.

In contrast, DS achieves peak efficiency at $\beta = 0.35$ and decreases steadily for larger β. It also incurs greater SINR losses without any benefit in spectrum efficiency, being smaller than the efficiency of GKM and AHP in almost all cases. In average, users experience 7dB of SINR loss, and even more, the users that experience the largest losses are at least 4dB worse than with GKM or AHP. Also, recall that DS with $\beta = 1$ corresponds to a power-based PF strategy, showing that both GKM and AHP achieve better performance than a straightforward PF scheme. It is clear that DS offers no reasonable tradeoff between user SINR and spectral efficiency.

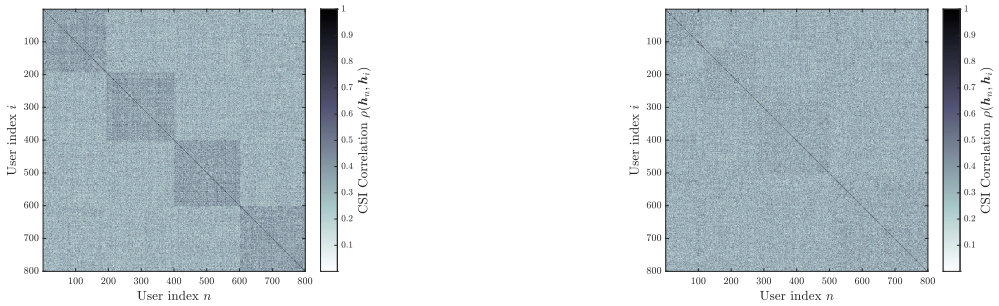
Clearly, the performance of MAC-P is naturally better than MAC-U, as the former enjoys perfect power control and can effectively deal with CSI power variability. Nevertheless, our proposed algorithms still attain good performance under the simpler setup of MAC-U, with only minor performance loss.

In both MAC-U and MAC-P tests, GKM and AHP deliver much better spectrum efficiency and lower SINR loss. Compared with DS, our tests strongly support the efficacy of the proposed MIMO user scheduling principle based on unsupervised learning on the Grassmannian manifold.

C. Performance for Downlink (BC)

In BC, we set the number of clusters to $K = 4$. Recall that the MRT precoders do not exploit the CSI of users in an RSG, and hence a smaller K better controls the SINR experienced by each user without incurring in severe degradation, due to oversharing resources.

1) Clustering Outcomes: We first analyze the CSI correlation matrices for GKM and AHP in Fig. 8, where darker colors in grayscale denote larger correlation coefficient. Even for a low number of clusters, GKM is able to produce discriminative clusters that contain several user pairs with relatively high correlation compared to the members of other clusters. In this case, however, AHP produces clusters that are even less defined, which hints that users within a cluster are not highly similar. This can also be explained by the bottom-up nature of hierarchical clustering of a large number of users, as the last steps (merging large subclusters) are not very discriminative.



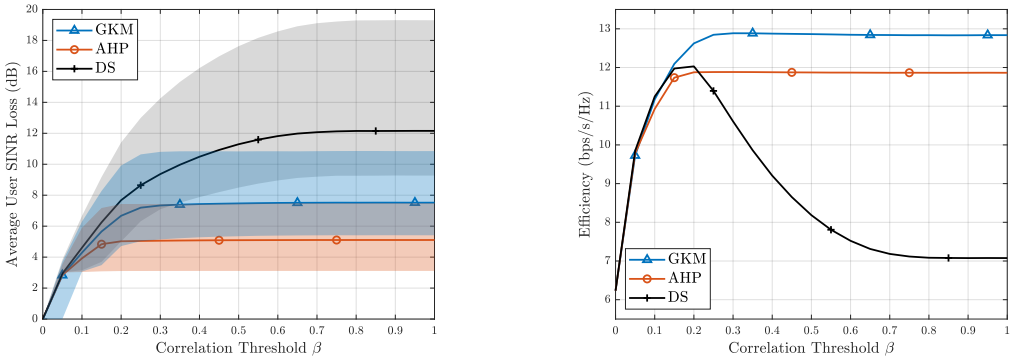
(a) GKM. (b) AHP.

Fig. 8: Example of of pairwise CSI correlation coefficient matrices after clustering $N = 800$ users with $K = 4$. Larger correlation values are darker. (a) Grassmannian K-means. (b) Hierarchical Clustering.

2) Scheduling Performance: For BC systems, we only consider the case where all users have random log-normal distributed CSI power, as the SINR of each user does not depend on the power of the rest of the co-channel users. Here, for the BC grouping rule φ^{BC} we consider a maximum number of users $D = K = 4$, a gain threshold for weak users $\delta = 0.5$ corresponding to -6dB of CSI power with respect to the average channel power, and a maximum number of weak users in an exclusive group $E = 2$. We also set $p_{\mathrm{BS}}^{\max} = M$ and use uniform power allocation within scheduled groups, although a different scheme can also apply.

Figs. 7a and 7b depict the SINR loss distribution and corresponding spectrum efficiency of all tested scheduling methods in BC. We first observe higher achieved SINR loss for BC than for MAC. Such outcome is expected since the selected UE receiver cannot utilize CSI of other users in its co-channel MIMO group. Correspondingly, the BC spectrum efficiency is lower. Similar to MAC systems, all three methods under comparison show comparable performance in terms of efficiency and SINR loss for $\beta \leq 0.15$. For larger β, AHP stalls and GKM offers a modest growth in efficiency, reaching 13% higher efficiency than AHP for $\beta > 0.3$. By comparison, AHP scheduling shows the best SINR losses, whereas GKM offers a good tradeoff and outperforms AHP in spectral efficiency over all values of β.

In BC, the efficiency of DS peaks at at $\beta = 0.2$ before dropping by 58% from its peak. Both the proposed GKM and AHP out-perform DS for $\beta \geq 0.25$. DS also suffers the worst losses for all users, with an average SINR loss of up to 12dB, and 18dB SINR loss for the 10% of users that experience the worst conditions. Moreover, most users experience significantly worse SINR conditions on DS than GKM or AHP. This includes the case $\beta = 1$, where DS is effectively a



(a) Average user SINR loss.

(b) Average efficiency.

Fig. 9: Performance of all scheduling algorithms in BC. Here, $M = 8$, $N = 800$, $K = D = 4$, $E = 2$ and $\delta = 0.5$. Solid lines represent the average over channel realizations, and shaded areas show all values within the 10th and 90th percentiles.

PF scheme over channel power partitions with different number of users per group depending on channel power. Clearly, our test results in BC show that our proposed scheduling strategy, by exploiting similarity obtained via unsupervised learning, provides better SINR for the UE receivers and achieve higher spectrum efficiency than a direct approach.

D. Runtime and Scalability

Table I summarizes the average runtime of all 3 algorithms under test for various number of users N under the same test settings specified earlier. In particular, we set a correlation threshold of $\beta = 0.8$, where there is significant performance difference between DS and the two proposed methods based on learning. As expected, the runtime grows with increasing N in both uplink and downlink. The runtime is not affected by different channel gains (MAC-P vs. MAC-U). Because of its clustering complexity, AHP requires the largest runtime, whereas DS can be faster without clustering for smaller number of users. On the other hand, the proposed GKM only requires a modest level of computation, in view of the performance benefits shown in terms of SINR loss and spectrum efficiency. For example, GKM scheduling under uplink MAC only requires a modest increase of computation to provide significant performance improvement with respect to DS. For larger number of nodes N, the runtime of GKM scheduling scales mildly, as opposed to the sharper rise of runtime for AHP. In BC downlink, we observe similar results, in which the complexity gap between GKM scheduling and DS is less significant. The BC runtime

comparison shows that for large N deployment smart scheduling methods based on unsupervised learning offer improved performance with little or no increase in computation complexity.

TABLE I: Average runtime of all algorithms, in seconds, for $\beta = 0.8$.

	Mode	MAC-P			MAC-U			BC		
	Method	GKM	AHP	DS	GKM	AHP	DS	GKM	AHP	DS
	800	0.41	0.77	0.11	0.42	0.78	0.11	0.76	1.22	0.38
	1600	1.97	4.05	0.96	2.03	4.12	1.00	2.96	6.29	3.03
N	2400	5.95	11.02	2.85	6.05	10.99	2.87	8.41	13.78	8.93
	3200	11.20	25.28	8.37	11.47	25.69	8.08	19.30	32.45	22.93
	4000	20.25	42.56	14.29	20.29	41.98	14.27	36.21	54.12	40.74

V. Conclusions

Massive MIMO systems rely on spatial diversity to support a large number of user devices. To efficiently utilize MIMO spatial diversity for high spectrum efficiency, mutual interference among co-channel users must be mitigated. A key challenge is to design efficient user scheduling schemes that suppress the resulting CCI among scheduled users in RSGs. MIMO user scheduling is an NP-hard problem and is difficult to generate labeled training dataset for supervised learning. Unsupervised learning is effective at extracting shared underlying data features such as CSI similarity but its direct application is incompatible with the goal of promoting spatial diversity within RSGs. This work presents two major contributions: (a) we develop a scalable scheduling strategy that first identifies CSIs of high similarity by leveraging unsupervised learning, and then exploits the learning outcomes to schedule user groups with high CSI diversity (dissimilarity) to minimize CCI; (b) recognizing the CCI to be invariant to MIMO CSI magnitude and common phase rotation, we adopt unsupervised CSI clustering on the complex Grassmannian manifold, to accurately assess the CSI spatial diversity. Our test results show that in both uplink and downlink, our scheduling principle indirectly based on outcomes of unsupervised learning demonstrate performance gains and robustness, both in terms of receiver SINR and spectrum efficiency.

VI. Acknowledgement

The authors thank visiting student Q. Wei from Beijing University of Posts and Telecommunications (China) for helpful discussions and exploration on directly related technical issues.

REFERENCES

[1] Qualcomm Technologies Inc., "The 5G unified air interface: Scalable to an extreme variation of requirements," Tech. Rep., Nov. 2015.

[2] J. G. Andrews *et al.*, "What will 5g be?" *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1065–1082, 2014.

[3] H. Holma and A. Toskala, *LTE Advanced: 3GPP Solution for IMT-Advanced*. New York, NY, USA: Wiley, 2012.

[4] S. Verdu, "Minimum probability of error for asynchronous Gaussian multiple-access channels," *IEEE Trans. Inf. Theory*, vol. 32, no. 1, pp. 85–96, 1986.

[5] P. Vandenameele, L. Van Der Perre, M. Engels, B. Gyselinckx, and H. De Man, "A combined OFDM/SDMA approach," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 11, pp. 2312–2321, 2000.

[6] L. Hanzo, T. Keller, M. Muenster, and B.-J. Choi, *OFDM and MC-CDMA for Broadband Multi-User Communications, WLANs and Broadcasting*. USA: Wiley, 2003.

[7] S. Verdu, "Optimum multiuser asymptotic efficiency," *IEEE Trans. Commun.*, vol. 34, no. 9, pp. 890–897, 1986.

[8] X. Zhu and R. Murch, "Performance analysis of maximum likelihood detection in a MIMO antenna system," *IEEE Trans. Commun.*, vol. 50, no. 2, pp. 187–191, 2002.

[9] R. Lupas and S. Verdu, "Linear multiuser detectors for synchronous code-division multiple-access channels," *IEEE Trans. Inf. Theory*, vol. 35, no. 1, pp. 123–136, 1989.

[10] Z. Xie, R. Short, and C. Rushforth, "A family of suboptimum detectors for coherent multiuser communications," *IEEE J. Sel. Areas Commun.*, vol. 8, no. 4, pp. 683–690, 1990.

[11] U. Madhow and M. Honig, "Mmse interference suppression for direct-sequence spread-spectrum cdma," *IEEE Trans. Commun.*, vol. 42, no. 12, pp. 3178–3188, 1994.

[12] R. Heath and D. Love, "Multimode antenna selection for spatial multiplexing systems with linear receivers," *IEEE Trans. Signal Process.*, vol. 53, no. 8, pp. 3042–3056, 2005.

[13] L. Liu, R. Chen, S. Geirhofer, K. Sayana, Z. Shi, and Y. Zhou, "Downlink MIMO in LTE-advanced: SU-MIMO vs. MU-MIMO," *IEEE Commun. Mag.*, vol. 50, no. 2, pp. 140–147, 2012.

[14] 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, 2003.

[15] 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, 2017.

[16] T. F. Maciel and A. Klein, "On the performance, complexity, and fairness of suboptimal resource allocation for multiuser MIMO–ofdma systems," *IEEE Trans. Veh. Technol.*, vol. 59, no. 1, pp. 406–419, 2010.

[17] Z. Hong, K. Liu, R. Heath, and A. Sayeed, "Spatial multiplexing in correlated fading via the virtual channel representation," *IEEE J. Sel. Areas Commun.*, vol. 21, no. 5, pp. 856–866, 2003.

[18] D.-S. Shiu, G. Foschini, M. Gans, and J. Kahn, "Fading correlation and its effect on the capacity of multielement antenna systems," *IEEE Trans. Commun.*, vol. 48, no. 3, pp. 502–513, 2000.

[19] X. Chen, H. Hu, H. Wang, H.-h. Chen, and M. Guizani, "Double proportional fair user pairing algorithm for uplink virtual MIMO systems," *IEEE Trans. Wireless Commun.*, vol. 7, no. 7, pp. 2425–2429, 2008.

[20] H.-W. Chang and L.-C. Wang, "A low-complexity uplink multiuser scheduling for virtual MIMO systems," *IEEE Trans. Veh. Technol.*, vol. 65, no. 1, pp. 463–466, 2016.

[21] S. Wang, F. Wang, Y. Wang, and D. Yang, "A novel scheduling scheme based on MU-MIMO in TD-LTE uplink," in *2011 IEEE Wireless Commun. Netw. Conf.* IEEE, 2011, pp. 915–919.

[22] Y. J. Zhang and K. Letaief, "An efficient resource-allocation scheme for spatial multiuser access in MIMO/OFDM systems," *IEEE Trans. Commun.*, vol. 53, no. 1, pp. 107–116, 2005.

[23] B. Kim *et al.*, "Uplink NOMA with multi-antenna," in *2015 IEEE 81st Veh. Technol. Conf. (VTC Spring)*, 2015, pp. 1–5.

[24] D. Kudathanthirige and G. A. A. Baduge, "NOMA-aided multicell downlink massive MIMO," *IEEE J. Sel. Topics Signal Process.*, vol. 13, no. 3, pp. 612–627, 2019.

[25] S. C. R. Gaddam, D. Kudathanthirige, and G. Amarasuriya, "Achievable rate analysis for NOMA-aided massive MIMO uplink," in *2019 IEEE Int. Conf. Commun. (ICC)*, 2019, pp. 1–7.

[26] C. Jiang, H. Zhang, Y. Ren, Z. Han, K.-C. Chen, and L. Hanzo, "Machine learning paradigms for next-generation wireless networks," *IEEE Wireless Commun. Mag.*, vol. 24, no. 2, pp. 98–105, 2017.

[27] M. E. Morocho-Cayamcela, H. Lee, and W. Lim, "Machine Learning for 5G/B5G Mobile and Wireless Communications: Potential, Limitations, and Future Directions," *IEEE Access*, vol. 7, pp. 137 184–137 206, 2019.

[28] 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.

[29] 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, 2018.

[30] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*. Cambridge Univ. Press, 2005.

[31] M. Costa, "Writing on dirty paper (corresp.)," *IEEE Trans. Inf. Theory*, vol. 29, no. 3, pp. 439–441, 1983.

[32] E. Björnson and E. Jorswieck, "Optimal Resource Allocation in Coordinated Multi-Cell Systems," *Found. Trends Commun. Inf. Theory*, vol. 9, no. 2-3, pp. 113–381, 2013.

[33] T. Lo, "Maximum ratio transmission," *IEEE Trans. Inf. Theory*, vol. 47, no. 10, pp. 1458–1461, 1999.

[34] N. Jindal, "MIMO Broadcast Channels With Finite-Rate Feedback," *IEEE Trans. Inf. Theory*, vol. 52, no. 11, pp. 5045–5060, 2006.

[35] R. Mo and Y. H. Chew, "MMSE-based joint source and relay precoding design for amplify-and-forward MIMO relay networks," *IEEE Trans. Wireless Commun.*, vol. 8, no. 9, pp. 4668–4676, 2009.

[36] K. H. Lee and D. P. Petersen, "Optimal linear coding for vector channels," *IEEE Trans. Commun.*, vol. 24, pp. 1283–1290, Dec. 1976.

[37] H. Sampath, P. Stoica, and A. Paulraj, "Generalized linear precoder and decoder design for MIMO channels using the weighted MMSE criterion," *IEEE Trans. Commun.*, vol. 49, no. 12, pp. 2198–2206, 2001.

[38] A. Scaglione, G. Giannakis, and S. Barbarossa, "Redundant filterbank precoders and equalizers. i. unification and optimal designs," *IEEE Trans. Signal Process.*, vol. 47, no. 7, pp. 1988–2006, 1999.

[39] D. Palomar, M. Bengtsson, and B. Ottersten, "Minimum BER linear transceivers for MIMO channels via primal decomposition," *IEEE Trans. Signal Process.*, vol. 53, no. 8, pp. 2866–2882, 2005.

[40] L. Kaufman and P. J. Rousseeuw, *Finding Groups in Data: An Introduction to Cluster Analysis*. New York, NY, USA: Wiley, 1990.

[41] A. Singhal, "Modern information retrieval: a brief overview," *Bull. IEEE Comput. Soc. Tech. Committee Data Eng.*, vol. 24, pp. 35–43, 2001.

[42] D.-S. Shiu, G. Foschini, M. Gans, and J. Kahn, "Fading correlation and its effect on the capacity of multielement antenna systems," *IEEE Trans. Commun.*, vol. 48, no. 3, pp. 502–513, 2000.

[43] A. Goh and R. Vidal, "Clustering and dimensionality reduction on Riemannian manifolds," in *2008 IEEE Conf. Comput. Vision Pattern Recognit.*, 2008, pp. 1–7.

[44] A. Edelman, T. A. Arias, and S. T. Smith, "The geometry of algorithms with orthogonality constraints," *SIAM J. Matrix Anal. Appl.*, vol. 20, no. 2, p. 303–353, Apr. 1999.

[45] N. Boumal and P.-A. Absil, "Low-rank matrix completion via preconditioned optimization on the Grassmann manifold," *Linear Algebra Appl.*, vol. 475, pp. 200–239, 2015.

[46] S. Stiverson, M. Kirby, and C. Peterson, "Subspace Quantization on the Grassmannian," in *Advances in Self-Organizing Maps, Learning Vector Quantization, Clustering and Data Visualization (WSOM 2019)*, A. Vellido, K. Gibert, C. Angulo, and J. D. Martín Guerrero, Eds. Cham: Springer, 2020, pp. 251–260.

[47] Y. Zhang, "K-means Principal Geodesic Analysis on Riemannian Manifolds," in *Proc. Future Technol. Conf. (FTC) 2019*, K. Arai, R. Bhatia, and S. Kapoor, Eds. Cham: Springer, 2020, pp. 578–589.

[48] P.-A. Absil, R. Mahony, and R. Sepulchre, *Optimization Algorithms on Matrix Manifolds*. Princeton, NJ, USA: Princeton Univ. Press, 2008.

[49] M. Usama *et al.*, "Unsupervised machine learning for networking: Techniques, applications and research challenges," *IEEE Access*, vol. 7, pp. 65 579–65 615, 2019.

[50] D. Xu and Y. Tian, "A comprehensive survey of clustering algorithms," *Annals of Data Science*, vol. 2, pp. 165–193, 2015.

[51] D. Arthur and S. Vassilvitskii, "K-Means++: The Advantages of Careful Seeding," in *Proc. 18th Ann. ACM-SIAM Symp. Discrete Algorithms*, ser. SODA '07. USA: SIAM, 2007, p. 1027–1035.

[52] J. Nam, J.-Y. Ahn, A. Adhikary, and G. Caire, "Joint spatial division and multiplexing: Realizing massive MIMO gains with limited channel state information," in *2012 46th Ann. Conf. Inf. Sci. Syst. (CISS)*, 2012, pp. 1–6.

[53] A. Adhikary, J. Nam, J.-Y. Ahn, and G. Caire, "Joint spatial division and multiplexing—the large-scale array regime," *IEEE Trans. Inf. Theory*, vol. 59, no. 10, pp. 6441–6463, 2013.

[54] J. Nam, A. Adhikary, J.-Y. Ahn, and G. Caire, "Joint spatial division and multiplexing: Opportunistic beamforming, user grouping and simplified downlink scheduling," *IEEE J. Sel. Topics Signal Process.*, vol. 8, no. 5, pp. 876–890, 2014.

[55] N. Jindal and A. Goldsmith, "Dirty-paper coding versus TDMA for MIMO broadcast channels," *IEEE Trans. Inf. Theory*, vol. 51, no. 5, pp. 1783–1794, 2005.

[56] M. J. Zaki and W. M. Jr, *Data Mining and Analysis: Fundamental Concepts and Algorithms*. New York, NY, USA: Cambridge Univ. Press, 2014.

[57] *Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures*, 3GPP Tech.l Specification TS 36.213 (v16.7.1), 2021.

[58] "IEEE Standard for Information Technology–Telecommunications and Information Exchange between Systems - Local and Metropolitan Area Networks–Specific Requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," *IEEE Std 802.11-2020 (Revision of IEEE Std 802.11-2016)*, pp. 1–4379, 2021.

[59] P. J. Rousseeuw, "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis," *J. Comput. Appl. Math.*, vol. 20, pp. 53–65, 1987.

[60] W. J. Krzanowski and Y. T. Lai, "A criterion for determining the number of groups in a data set using sum-of-squares clustering," *Biometrics*, vol. 44, no. 1, pp. 23–34, 1988.

[61] J. A. Hartigan, *Clustering Algorithms*. New York, NY, USA: Wiley, 1975.

[62] R. Tibshirani, G. Walther, and T. Hastie, "Estimating the number of clusters in a data set via the gap statistic," *J. Roy. Statistical Soc. Series B*, vol. 63, no. 2, pp. 411–423, 2001.