Mobility-aware Multi-user MIMO Link Scheduling for Dense Wireless Networks

Mengyao Ge and *Douglas M. Blough*
School of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, Georgia 30332-0250
{mengyao, doug.blough}@ece.gatech.edu

***Abstract*—In this paper, we consider the multiuser MIMO scheduling problem for dense wireless networks with access point cooperation. The problem is to maximize the aggregate throughput within a single cluster of access points, while maintaining a general fairness criterion. To alleviate the protocol overhead and sustain the performance of both stationary and mobile users, we propose a mobility-aware scheduling approach, which places users into stationary and mobile groups based on a novel CSI similarity metric. The algorithm then schedules the two groups into separate time slots. To balance fairness between stationary and mobile user groups, we adaptively determine their transmission time fractions. Different scheduling strategies are applied for the two groups. For stationary users, we collect CSI infrequently and perform a computationally expensive scheduling algorithm that is highly optimized to maximize throughput while maintaining fairness. For mobile users, we do per-time-slot CSI measurement and schedule users for each time slot using a very fast but less-optimized algorithm. Numerical results demonstrate that, when accounting for CSI feedback and scheduling overheads, our proposed scheduling algorithm with mobility awareness maintains very good fairness and provides substantial performance gains compared to conventional approaches that do not separate mobile and stationary users.**

I. Introduction

The concept of network MIMO, also referred to as distributed MIMO, has the potential to dramatically increase throughput in dense wireless networks, by allowing access points (APs) to synchronize and share lower-layer information. Nework MIMO is a potential way to accommodate the ever-increasing traffic data demands in wireless local access networks (WLANs). The approach is particularly suited for dense enterprise networks with clusters of closely deployed APs [1]. It can alleviate the high level of co-channel interference introduced by many nearby APs operating on the same frequency. A common scenario is that these APs share a network gateway with one Internet connection. Thus, multiple nearby APs can be grouped into a cluster and cooperate to control the lower-layer parameters and to optimize the overall performance. The simplest way to achieve this goal is to introduce a centralized controller that manages the APs within a cluster [1].

This paper focuses on developing a centralized schedule that achieves both high throughput and a target fairness criterion among users. Our prior research [2] that considers a similar problem targets only static network scenarios. While indoor WLANs, such as in office-type environments, are dominated by stationary clients, there is also limited mobility due to occasional device movements and environmental changes. Client mobility poses a unique problem for the scheduling algorithm. In static environments, the wireless channels remain stable and past information can be relied on to optimize the performance. In contrast, the scheduler for mobile clients needs to accommodate frequent changes of wireless channels. Therefore, given the mix of users with diverse channel and mobility characteristics in next generation enterprise networks, different scheduling strategies are preferable for improving the overall performance.

In this paper, we propose a mobility-aware multiuser MIMO link scheduling algorithm that distinguishes stationary and mobile users based on their channel state information (CSI) and applies different scheduling strategies within each user group. The central controller tracks the channel conditions of clients over time and applies a novel CSI similarity metric based on subspace collinearity to categorize users as either stationary or mobile. Our mobility-aware scheduling algorithm then separates static and mobile users into different time slots, and adaptively adjusts the number of time slots between the two categories to maintain fairness for both stationary and mobile users. The stationary user schedule is calculated in a highly optimized but fairly computationally expensive manner. However, since CSI does not change frequently for these users, this highly optimized schedule can be used for a significant number of scheduling periods. In contrast, the schedule for mobile users is done for each time slot using fresh CSI but in a highly efficient, less optimized fashion. The separation of users into two categories allows us to achieve the promise of expensive but very-high-performing scheduling algorithms that have been presented in the literature for stationary users, while still achieving reasonable performance for mobile users and ensuring fairness both across the two user categories and for individual users. Simulation results demonstrate that, when accounting for CSI feedback and scheduling overheads, our proposed scheduling algorithm with mobility awareness maintains very good fairness and provides substantial performance gains compared to conventional approaches that do not separate mobile and stationary users.

II. System Model and Problem Statement

We consider a scenario in which single-hop wireless networks are densely deployed over a region, where the areas served by different access points (APs) can overlap. We focus on indoor environments, where most devices are stationary for a moderate amount of time between movements. When users' devices are not stationary, they move at low speeds (typically from walking with or rotating a hand-held device). This is a common scenario for most enterprise WLAN settings, which typically covers office-type environments. We focus on downlink transmissions since in typical indoor environments 80% or more of the traffic is on the downlink. We do not mix downlink and uplink traffic in one slot, because scheduling downlink or uplink traffic together helps reduce channel estimation overhead as shown in [3].

A. Access Point Cooperation

Network MIMO is a potential technique to improve the aggregate performance for high-density wireless network deployments by converting the inter-cell interference into multiplexing gain via transmitter cooperation. Due to several constraints, including complexity of coordination, backhaul limitations, and computational limits for scheduling, the practical way to realize network MIMO in dense environments is to group a small number of nearby APs into a cluster as shown in Fig.1. Thus, we divide a large enterprise wireless network into clusters, where the APs within the same cluster can cooperate with each other with the assistance of a network control unit (CU). [1] Determining AP clusters is beyond the scope of this paper. Actually, many environment provides a natural way of clustering APs or any reasonable clustering algorithm can provide the type structure we envision. In this paper, we assume predetermined AP clusters and user association.

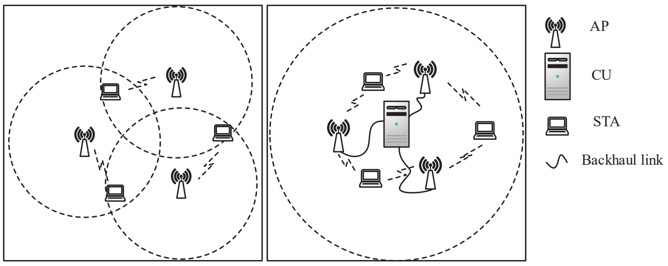

Fig. 1: Traditional multi-cell WLAN (left) and clustered WLAN (right).

B. PHY-layer model

Assume there are M access points (APs) in one cluster, which cooperatively serve K users. We denote the number of antenna elements on the m^{th} AP by $N_{t,m}$ and the number of antenna elements on the k^{th} user by N_r. The user set is denoted by $\mathcal{K} = \{1, \ldots, K\}$. Let $N_t = \sum_{m=1}^{M} N_{t,m}$ be the total numbers of antennas at the AP side. The matrix of complex channel gains between the cooperative APs and the antennas of the k^{th} user is denoted by $\boldsymbol{H}_k \in \mathbb{C}^{N_r \times N_t}$. The data vector $\boldsymbol{x} = \left[\boldsymbol{x}_1^T, \ldots, \boldsymbol{x}_K^T\right]^T$ is jointly precoded by the M APs using the precoding matrix $\boldsymbol{F} = [\boldsymbol{F}_1, \ldots, \boldsymbol{F}_K]$. $\boldsymbol{x}_k \in \mathbb{C}^{N_r}$ is the transmit signal vector for receiver k, and $\boldsymbol{x}_k$ is assumed to be independently encoded Gaussian codebook symbols with $\mathbb{E}[\boldsymbol{x}_k \boldsymbol{x}_k^\dagger] = \boldsymbol{I}$, where $(\cdot)^\dagger$ is the conjugate transpose of $(\cdot)$. It is assumed that the k^{th} user has N_r parallel data streams, although some of the streams can have a rate of zero. $\boldsymbol{F}_k \in \mathbb{C}^{N_t \times N_r}$ is the partition of $\boldsymbol{F}$ applied at the APs to precode the signals of user k.

The received vector at user k for time slot t is given by

$$\boldsymbol{y}_k = \boldsymbol{H}_k \boldsymbol{F}_k \boldsymbol{x}_k + \sum_{l=1, l \neq k}^{K} \boldsymbol{H}_k \boldsymbol{F}_l \boldsymbol{x}_l + \boldsymbol{n}_k \;, \tag{1}$$

where $\boldsymbol{n}_k$ is the vector of Gaussian noise at the k^{th} user with covariance matrix $\sigma_k^2 \boldsymbol{I}$. Assume the received signal is equalized using the linear receive filter $\boldsymbol{W}_k \in \mathbb{C}^{N_{r,k} \times N_{r,k}}$. The received signal of the k^{th} receiver is given by $\hat{\boldsymbol{x}}_k = \boldsymbol{W}_k^\dagger \boldsymbol{y}_k$.

C. CSI Feedback Mechanism

We consider a modification of the explicit feedback mechanism specified in 802.11ac. In 802.11ac, before a MU-MIMO transmission, an AP initiates channel sounding by transmitting a VHT null data packet (NDP) announcement, which specifies the set of users that are going to be polled for CSI feedback. After the NDP announcement, the AP transmits an NDP, which is used by the receivers for channel estimation.

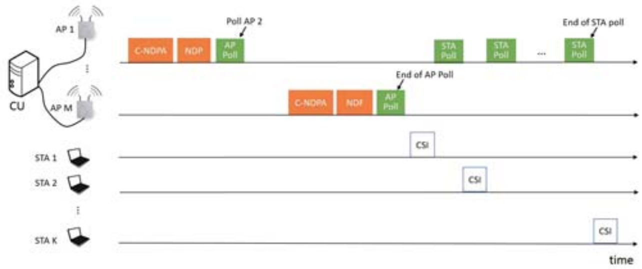

Fig. 2: CSI feedback mechanism for AP cooperation

With AP cooperation, each receiver needs to estimate the composed channel from all APs. This can be done by modifying the single AP mechanism from 802.11ac, as shown in Figure 2. The CU synchronizes the cooperative APs within the same cluster. The APs transmit the cooperative NDP announcement (C-NDPA) and NDP sequentially in a predetermined order to enable the receivers to measure the wireless channels. Each AP will send the AP-poll to notify the next AP for C-NDPA and NDP transmission after finishing its own C-NDPA and NDP transmission. Each client estimates the channel between itself and each AP, i.e. the channel matrix between the m^{th} AP and k^{th} client denoted by $\boldsymbol{H}_{k,m} \in \mathbb{C}^{N_r \times N_{t,m}}$. After receiving the last NDP, the k^{th} client concatenates its channel matrix from M cooperative APs as $\boldsymbol{H}_k = [\boldsymbol{H}_{k,1}, \boldsymbol{H}_{k,2}, \ldots, \boldsymbol{H}_{k,M}]$. For CSI feedback, a master AP is assigned to poll the receives one by one by sending a STA-poll, e.g., AP 1 is selected as the master AP in Figure 2. The first user will send back its CSI immediately after receiving the end of AP-poll, while other users will send their CSI after receiving the corresponding STA-poll.

[1]Our techniques can be applied independently across as many orthogonal channels as are available in a given wireless deployment.

The CSI feedback is always sent at the lowest modulation rate for reliability. To alleviate the feedback overhead, we use a compressed beamforming report as specified in 802.11ac for each polled user. It uses a quantized representation of the estimated channel, based on the SVD of the channel. Let $\boldsymbol{H}_k$ be the channel matrix of k^{th} user, which can be represented via compact SVD $\boldsymbol{H}_k = \boldsymbol{U}_k \boldsymbol{S}_k \boldsymbol{V}_k^\dagger$, where $\boldsymbol{S}_k \in \mathbb{C}^{N_r \times N_r}$ is the diagonal matrix containing the singular values in a decreasing order. $\boldsymbol{U}_k \in \mathbb{C}^{N_r \times N_r}$ and $\boldsymbol{V}_k \in \mathbb{C}^{N_t \times N_r}$ are the left and right singular matrix, respectively.

The explicit feedback in 802.11ac requires the right singular matrix to be decomposed, quantized and then fed back to the AP for transmit beamforming. $\boldsymbol{V}_k$ is a semi-unitary matrix with $\boldsymbol{V}_k^\dagger \boldsymbol{V}_k = \boldsymbol{I}$ and $\boldsymbol{V}_k^\dagger$ forms an orthonormal row basis of the channel matrix $\boldsymbol{H}_{k,n}$. Based on 802.11ac, the right singular matrix $\boldsymbol{V}_k$ can be decomposed using the Givens decomposition:

$$\boldsymbol{V}_k = \left\{ \prod_{i=1}^{N_r} \left(\boldsymbol{D}_k^i \prod_{j=i+1}^{N_t} \boldsymbol{G}_k^{i,j} \right) \right\} \tilde{\boldsymbol{I}} \boldsymbol{\Phi}_k^\dagger , \tag{2}$$

where $\tilde{\boldsymbol{I}}$ is a matrix containing the first N_r columns of an $N_t \times N_t$ unitary matrix. $\boldsymbol{D}_k^i = \operatorname{diag}\left(\boldsymbol{1}_{i-1}, e^{j\phi_{i,1}}, \ldots, e^{j\phi_{i,N_t-i+1}}\right)$ and $\boldsymbol{G}_k^{i,j}$ is the Givens rotation matrix

$$\boldsymbol{G}_k^{j,i} = \begin{bmatrix} \boldsymbol{I}_{i-1} & & & & \\ & \cos\psi_{i,j} & & \sin\psi_{i,j} & \\ & & \boldsymbol{I}_{j-i-1} & & \\ & -\sin\psi_{i,j} & & \cos\psi_{i,j} & \\ & & & & \boldsymbol{I}_{N_t-j} \end{bmatrix}$$

where $\boldsymbol{\Phi}_k$ is a diagonal matrix, which can be absorbed into $\boldsymbol{S}_k$. For example, 802.11ac will quantize and feedback the angles $\phi_{i,j}$ and $\psi_{i,j}$ using a uniform quantizer [4]. The CSI feedback overhead in multicarrier systems can be significantly reduced through subcarrier grouping, where the CSI feedback is performed for each sub-band consisting of several adjacent subcarriers.

D. MIMO Link Scheduling Problem

In the targeted dense environment, there are many users competing for limited resources. Therefore, MIMO link scheduling that can achieve high throughput while maintaining fairness is an essential requirement.

1) Potential aggregate throughput: The achievable data rates of MU-MIMO users depend on the concurrent user group and the corresponding MIMO weights (precoders and combiners). There are $\sum_{i=1}^{N_t} \binom{K}{i}$ possible user groups, also referred as communication sets (CommSets). Assume a certain CommSet $\Pi = \{\pi_1, \pi_2, \ldots, \pi_I\}$ for concurrent transmission with I users. The data rate of user π_k in Π is given by

$$r_{\pi_k} = \log_2 \left| \boldsymbol{I} + \boldsymbol{R}_{\pi_k}^{-1} \boldsymbol{H}_{\pi_k} \boldsymbol{F}_{\pi_k} \boldsymbol{F}_{\pi_k}^\dagger \boldsymbol{H}_{\pi_k}^\dagger \right| . \tag{3}$$

where $\boldsymbol{R}_{\pi_k}$ is the corresponding covariance matrix of the received interference plus noise is given by

$$\boldsymbol{R}_{\pi_k} = \sum_{l \in \Pi, l \neq \pi_k} \boldsymbol{H}_{\pi_k} \boldsymbol{F}_l \boldsymbol{F}_l^\dagger \boldsymbol{H}_{\pi_k}^\dagger + \sigma_{\pi_k}^2 \boldsymbol{I} . \tag{4}$$

2) Scheduling problem description: Our focus is on building a fair scheduler for a single cluster with M cooperative APs and K users. Let $\mathcal{T} = \{t_1, \ldots, t_T\}$ be the scheduling period composed of T time slots of equal duration, $\boldsymbol{\Pi}_j = \{\pi_{1,j}, \ldots, \pi_{I_j,j}\}$ be the CommSet scheduled in time slot t_j with I_j active users, and $\boldsymbol{r}_j = [r_{1,j}, \ldots, r_{K,j}]^T$ be the bit-rates of users in time slot t_j, where $r_{k,j} = 0$ if $k \notin \boldsymbol{\Pi}_j$. For a scheduling period $\mathcal{T}$, we need to schedule a CommSet for each time slot that maximizes the throughput while satisfying a fairness constraint. Mathematically, it can be formulated as follows:

$$\begin{aligned} &\max_{\{\boldsymbol{\Pi}_j\}_{j=1}^T} && \sum_{j=1}^{T} \sum_{k=1}^{K} r_{k,j} \\ &s.t. && \sum_{j=1}^{T} r_{k,j} = b_k \sum_{j=1}^{T} \sum_{k=1}^{K} r_{k,j} \end{aligned} \tag{5}$$

The fairness constraints require that each user achieves a bandwidth that is proportional to its target bandwidth share b_k. For example, the target bandwidth vector $\boldsymbol{b}$ can represent the QoS ratios of the competing users. In this paper, we are particularly interested in achieving time-based fairness, which has been shown in [5] to substantially improve the throughput compared to rate-based fairness in multi-rate WLANs. In [6], the idea of time-based fairness is extended to interfering MIMO channels. Following the idea in [6], the target bandwidth fraction of user k can be set to $b_k = \rho_k / \sum_{k=1}^{K} \rho_k$, where ρ_k is the interference-free data rate of user k. These time-fair b_k's are used in the simulation results of Section IV.

III. Fair MIMO Link Scheduling Algorithm Using Mobility Hints

Stationary users' channels can be stable for hundreds of milliseconds or even longer. For these users, the scheduler can rely on a CSI measurement to remain valid over multiple communication slots. The mobile users, however, require more frequent CSI updates to capture the channel variations. Therefore, it is inefficient to schedule the stationary and mobile users together, especially for a large user population with only a few mobile users. To resolve this problem, we incorporate mobility awareness into our proposed scheduling algorithm. Since the mobility only affects the performance of mobile users and does not affect stationary users during downlink transmission [2], it is possible to enhance the MU-MIMO performance by separating the stationary and mobile users into different time slots.

A. High-level operation of proposed scheduling framework

The operational flow of the proposed scheduling framework is shown in Figure 3. The CU tracks CSI over time and uses it to classify the users into stationary and mobile groups. The number of time slots reserved for the two user groups in each scheduling period, denoted by T_s and T_m, are adaptively adjusted based on the fairness criterion and achieved bandwidth (discussed in Section III-B). The scheduler first calculates an overall schedule for T_s time slots, including only stationary users. Upon completion of the stationary users' transmission,

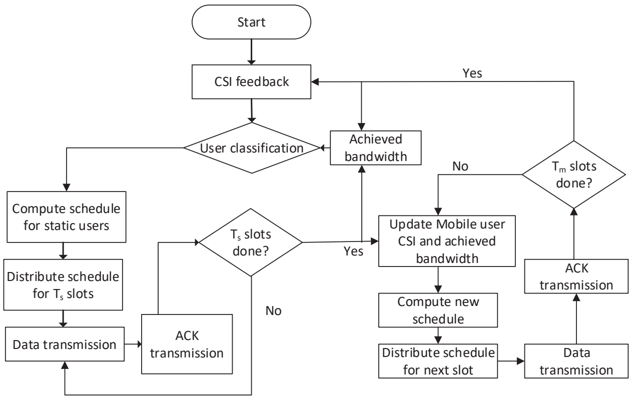

Fig. 3: High-level flow chart of the mobility-aware scheduling framework

the scheduler executes a per-slot scheduling strategy based on fresh CSI of mobile users, measured for each slot. The detailed scheduling algorithm is elaborated in Section III-C.

The communication schedule can be carried out using a TDMA MAC. We aggregate as many packets as can fit within a time slot with fixed duration τ_{slot} and have each receiver simultaneously acknowledge these packets within τ_{slot} using a Bulk ACK. The TDMA operation can be implemented through the 802.11 point coordination function (PCF) or even the distributed coordination function (DCF) as described in [7].

B. User Mobility Classification

To categorize stationary and mobile users, we track the CSI of each user across multiple measurements and identify the channels of stationary users based on CSI similarity. We propose to use subspace collinearity as a metric of CSI similarity. Subspace collinarity is a criterion that reflects the similarity between two matrix subspaces. In general, given two matrices $\boldsymbol{M}_1$ and $\boldsymbol{M}_2$, their subspace collinearity can be represented as

$$col(\boldsymbol{M}_1, \boldsymbol{M}_2) = 1 - \frac{abs(tr(\boldsymbol{M}_1 \boldsymbol{M}_2^\dagger))}{||\boldsymbol{M}_1||_F ||\boldsymbol{M}_2||_F}$$

The value of subspace collinearity varies from 0 to 1. A larger collinearity indicates a lower similarity of the two matrix subspaces.

Let $\tilde{\boldsymbol{V}}_k(t)$ and $\tilde{\boldsymbol{V}}_k(t-\Delta t)$ be the feedback right singular value of the k^{th} user's channel at time instants t and $t-\Delta t$. The similarity between consecutive CSI values is estimated by $f_s(k,t) = col\left(\tilde{\boldsymbol{V}}_k(t), \tilde{\boldsymbol{V}}_k(t-\Delta t)\right)$. For each user, we maintain a moving average of the CSI similarity to track the channel variation as follows:

$$\mathcal{S}(k,t) = \ (1-\beta_k)\mathcal{S}(k, t-\Delta t) + \beta_k f_s(k,t) \ .$$

If the value of $\mathcal{S}(k,t)$ for user k is smaller than a predefined threshold, user k is declared as a stationary user. Therefore, the stationary and mobile user groups are updated accordingly after each channel sounding stage.

Let $\mathcal{U}_s$ and $\mathcal{U}_m$ be the user sets containing stationary and mobile users, respectively. To maintain the fairness between the two user groups, the schedule duration portions reserved for stationary and mobile users should be proportional to their target bandwidth portions by factoring in their achieved bandwidth, which is given by:

$$\frac{T_s}{T_m} = \frac{\sum\limits_{i\in\mathcal{U}_s} b_i \exp(1-u_i/b_i)}{\sum\limits_{i\in\mathcal{U}_m} b_i \exp(1-u_i/b_i)} \ .$$

where T_s and T_m are the number of time slots to accommodate stationary and mobile users, respectively, which are adjusted upon the completion of each entire round of communications based on the achieved bandwidth portion $u_i = \bar{R}_i / \sum\limits_{i=1}^{K} \bar{R}_i$ with R_i representing the average achieved throughput of the i^{th} user. Without loss of generality, we assume $T = T_s + T_m$ is the number of time slots within one entire scheduling period. The objective of the adjustment is to roughly maintain a good fairness between stationary and mobile users. The fairness among each specific user group will be guaranteed by the proposed scheduler for each user group.

C. Calculating a Schedule

For MU-MIMO transmission, the performance of the scheduler is largely dependent on the choice of CommSets and their MIMO weights. For the targeted dense environment, there are typically a large number of users and it is, therefore, computationally prohibitive to explore all possible user combinations.

To balance the aggregate performance and processing overhead, the proposed scheduler works differently for stationary and mobile users. For stationary users, the scheduler calculates a number of high-performance CommSets and corresponding MIMO weights intensively and combines them into a schedule that maximizes throughput and satisfies the target fairness among stationary users. Compared to stationary users, mobile users are much more sensitive to stale CSI. The scheduler for mobile users requires frequent CSI update to accommodate channel variations. A general idea is to calculate a "good" CommSet for each time slot with updated CSI and run a low-complexity MIMO weight calculation algorithm.

1) Scheduling stationary users: $\mathcal{U}_s$ is the stationary user set to be scheduled over a scheduling period $\mathcal{T}_s$ having T_s time slots. The CSI values of the stationary users are updated and expected to be stable for the period of $\mathcal{T}_s$. After collecting the CSI for stationary users, the CU first generates a number of high-performance CommSets and their corresponding MIMO weights and then schedules the CommSets over the slots in $\mathcal{T}_s$, as shown in Figure 3. With stationary channels, the scheduler for stationary users can fully reap the benefits of AP cooperation by performing a fairly expensive optimization procedure to produce the schedule and MIMO weights.

We use an iterative algorithm to generate CommSets. In each iteration, we solve a weighted sum rate maximization problem using the algorithm proposed in [8]. Then, the user weights are updated according to the previously generated CommSets. The user weight update procedure is designed to aid the scheduler in achieving the target fairness criterion [2]. With generated CommSets, the scheduler calculates the num-

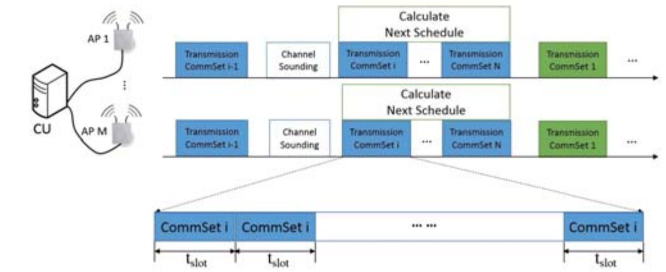

Fig. 4: Flow of operations for AP cooperation with proposed scheduling framework

ber of time slots assigned to each CommSet that achieves maximum throughput while guaranteeing the target fairness among stationary users, as proposed in [2].

Stationary users are less sensitive to processing overhead. If processing overhead becomes too high in certain scenarios, e.g. if the number of subcarrier groups is large, the CU can use parallel processing to speed up schedule calculation. In Section IV, we demonstrate that the computation time of the stationary scheduling procedure is small enough to achieve large throughput gains in practical scenarios.

2) Scheduling mobile users: Unlike stationary users, mobile users require more frequent CSI feedback to accommodate the channel variations caused by environmental changes and/or user mobility. The scheduling approach for stationary users is no longer suitable for mobile users, since the performance of mobile users will degrade as the CSI becomes outdated. Thus, mobile users cannot afford an intensive schedule calculation. Here, we need to develop a more efficient scheduling approach to improve CSI timeliness and accuracy.

Recall that there are N_m time slots, also referred to as *mobile slots*, reserved for $|\mathcal{U}_m|$ mobile users as discussed in Section III-B. The objective of scheduling mobile users within T_m mobile slots is to maximize their aggregate performance while meeting the fairness requirements. Let $b_k^m = \frac{b_k}{\sum_{k\in\mathcal{U}_m} b_k}$ be the normalized target bandwidth portion for mobile user k. When scheduling mobile time slot $t+1$, we use $R_{k,t}$, which is the achieved sum-rate of user k during the first t slots. We have $R_{k,t} = R_{k,t-1} + r_{k,t}$. Let $\mu_{k,t} = \frac{R_{k,t}}{\sum_{k\in\mathcal{U}_m} R_{k,t}}$ be the achieved bandwidth of mobile user k during the first t mobile slots with $\mu_{k,0} = 1, \forall k \in \mathcal{U}_m$ as the initial value. Since the CSI information for mobile users is updated for each time slot, the scheduling problem can be solved for each time slot in sequence. Ultimately, we aim to approach the fairness constraint in problem (5) for mobile users, which can be rewritten as:

$$R_{k,t} = b_k^m \sum_{k\in\mathcal{U}_m} R_{k,t}, \forall k \in \mathcal{U}_m \tag{6}$$

Assuming the equality constraint is satisfied at time slot t, we have

$$\begin{aligned} & R_{k,t} - r_{k,t} = u_{k,t-1}(R_{k,t}/b_k^m - \sum_{k\in\mathcal{U}_m} r_{k,t}) \\ \Longrightarrow\quad & r_{k,t} = (1 - u_{k,t-1}/b_k^m)R_{k,t} + u_k \sum_{k\in\mathcal{U}_m} r_{k,t} \\ & \geq (1 - u_{k,t-1}/b_k^m)R_{k,t}\ . \end{aligned}$$

Thus, the sum rate maximization can be approached by solving a weighted sum rate maximization problem for each time slot, i.e., $\max\ \sum_{k\in\mathcal{U}_m} w_{k,t}R_{k,t}$, where $w_k \propto 1 - u_{k,t-1}/b_k^m$. s_k indicates that larger weights are assigned to users that are below their target bandwidth proportions when considering the previous $t-1$ time slots. Therefore, we can update the user weights as follows:

$$w_{k,t} = \max\left(1 - u_{k,t-1}/b_k^m, 0\right)\ . \tag{7}$$

Thus, any user that is at or above its desired bandwidth proportion is assigned with zero weight and is therefore excluded from the current round of tranmission. To speed up the processing overhead of mobile users, a simple and computationally efficient precoding approach is utilized, namely, block diagonalization (BD). The optimization problem for time slot t can be formulated as:

$$\begin{aligned} \max \quad & \sum_{k\in\mathcal{U}_m} w_{k,t} \log_2 \left| \boldsymbol{I} + \boldsymbol{R}_k^{-1}\boldsymbol{H}_k\boldsymbol{F}_k\boldsymbol{F}_k^\dagger\boldsymbol{H}_k^\dagger \right| \\ s.t. \quad & \sum_{k\in\mathcal{U}_m} Tr(\boldsymbol{\Gamma}_m\boldsymbol{F}_k\boldsymbol{F}_k^\dagger) \leq P_m, m = 1, \ldots, M \\ & \boldsymbol{H}_l\boldsymbol{F}_k = \boldsymbol{0}, l, k \in \mathcal{U}_m, l \neq k\ . \end{aligned} \tag{8}$$

The diagonal matrix $\boldsymbol{\Gamma}_m \in \mathbb{R}^{N_t\times N_t}$ is introduced for each AP to select the partition of $\boldsymbol{F}_k$ applied at the m^{th} AP and P_m is the maximum transmit power of the m^{th} AP. Thus, $\boldsymbol{\Gamma}_m$ contains ones on the diagonal elements corresponding to the antennas of the m^{th} AP and zeros elsewhere.

The maximum number of users in one slot is $\lceil N_t/N_r \rceil$. To reduce computational overhead, we select the $\lceil N_t/N_r \rceil$ users with highest weights for each time slot. The BD precoder can then be designed using QR decomposition with water-filling power loading, as analyzed in [9].

IV. Simulation Results

We conduct simulations of our proposed scheduling algorithm using the WINNER II channel model for indoor office environments [10]. We uniformly distribute M APs and K users in a circular region with a radius of 50 meters. We set each AP to have 4 transmit antennas and each client to have 2 receive antennas. The noise power is -85 dBm and the transmit power of each AP is 23 dBm. Unless otherwise specified, we consider downlink transmission with $M = 3$ cooperative APs.

For comparison, we also consider the following schedulers:

- **Per-slot scheduler**: This is a scheduling algorithm that generates a CommSet by solving a WSRM problem for each time slot. To meet the fairness requirement, the user weights are updated using (7) after the transmission of each time slot. The CSI values are assumed to be updated for each time slot $\tau_{slot} = 5\ ms$. The performance of the basic per-slot scheduler is computed without accounting for the overhead of CSI feedback and processing overhead. This then forms an upper bound on the performance of other schedulers since it optimizes for each time slot and incurs zero overhead. The per-slot scheduler that accounts for CSI feedback and processing overhead is also evaluated and that algorithm is denoted by **Per-slot***.

- **One-shot scheduler**: This is a scheduling algorithm proposed in [2] for a completely static environment. In this paper, we implement this algorithm by treating all users as if they were stationary. Therefore, the channel variation of mobile users within one entire scheduling period will cause performance loss for this algorithm.
- **Conventional TDMA**: This is a basic time-fair TDMA scheduling algorithm, where the MIMO links are scheduled sequentially in a round robin manner. In other words, there is only one user scheduled in each time slot cooperatively served by M APs. The SU-MIMO transmission within each time slot can achieve the interference-free data rates using the optimal SVD MIMO weights.

To evaluate the achieved fairness, we use the fairness index proposed in [6],

$$FI(\boldsymbol{u}, \boldsymbol{b}) = \exp\left(-\sum_{k=1}^{K} |\ln(u_k/b_k)|\right)/K \ , \qquad (9)$$

where u_k is the fraction of bandwidth allocated to the k^{th} user. The fairness index given by (9) takes values in $[0, 1]$, with 1 representing perfect fairness among users.

A. Evaluation of CSI feedback overhead

We first evaluate the CSI feedback overhead of the proposed scheme by comparing with the conventional scheme without user classification. The percentage of mobile users is denoted

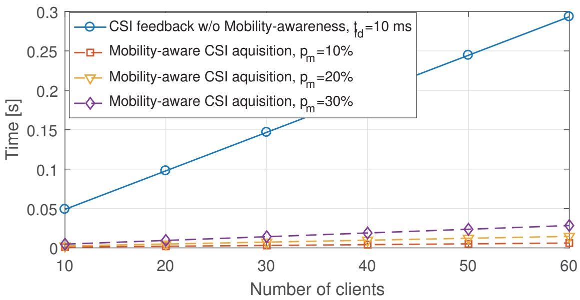

Fig. 5: CSI collection time versus the number of clients within $T = 1\ s$.

by p_m. Figure 5 shows the CSI collection time versus the number of clients within for a period of 1 second. Without user classification, the CSI update period t_{fd} is identical for all users. To guarantee the CSI accuracy of mobile users, the updated period is set to every 10 msec. In this case, the CSI collection time increases rapidly as the number of users increases. For example, it takes more than 0.3 seconds for CSI feedback for 50 users or more, which can overwhelm the data transmission time. The CSI feedback overhead will further scale up with the increase of subcarriers/subbands. By taking advantage of user classification, we can significantly lower the CSI update frequency for stationary users, since their channels can be stable for up to several seconds. In Figure 5, the CSI update period for stationary users is set to 1 second while for mobile users it remains at 10 msec. In this case, we can largely reduce the CSI feedback overhead, while guaranteeing the same CSI accuracy for the mobile users as the conventional scheme. Therefore, the mobility-aware scheme is a promising approach in terms of reducing CSI feedback for the scenarios with limited-mobility. For example, with 30% mobile users, the CSI collection requires about 0.03 seconds within a period of 1 second for 60 users.

B. Evaluation of user classification

In Figure 6, the cumulative distributions of the CSI similarity for stationary and mobile users are plotted using our subspace collinearity metric. The consecutive CSI samples are

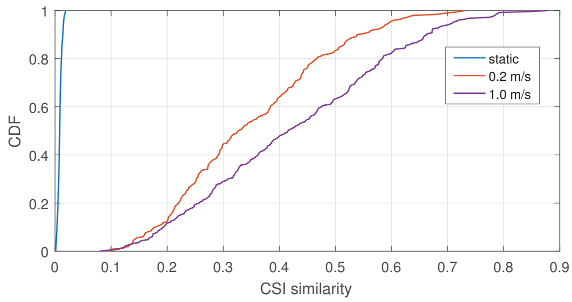

Fig. 6: CDF of the CSI similarity.

collected every 1 second. Clearly, the subspace collinearity metric is a reliable indicator to distinguish stationary and mobile users. For stationary users, the CSI similarity is very close to zero, while the mobile users generate much higher CSI similarity values. Based on these results, we set the similarity threshold $Thr = 0.05$ so that any $Thr > 0.05$ causes a user to be classified as mobile.

C. Evaluation of throughput and fairness

In Figure 7, we evaluate the throughput and fairness performances of the different scheduling algorithms versus the mobile user percentage. The average speed of mobile users is set to 1 m/s. The **per-slot scheduler** provides highest throughput and good fairness, because it neglects the CSI feedback and processing overheads. However, in practice, the intensive CSI feedback for the per-slot update would significantly reduce the data transmission time and lower the achievable throughput. Moreover, the CSI delay caused by the per-slot processing overhead for a large user population (e.g., $K = 45$) would also introduce large fairness loss, especially for mobile users.

Both CSI feedback and processing overhead are accounted for with **per-slot***, as well as with our proposed mobility-aware scheduling algorithm. With user classification, we are able to reduce the CSI overhead for stationary users, while reducing the processing overhead for the mobile users. Besides, the adaptive adjustment of the time slot assignment produces a good fairness among stationary and mobile users. Therefore, our proposed scheduling scheme achieves 25%-35% higher throughput than that of **per-slot***. The one-shot scheduler cannot meet the fairness requirement because the CSI for mobile users become outdated for data transmission. The conventional TDMA schedules a single user for each time slot and guarantees perfect fairness among all users but

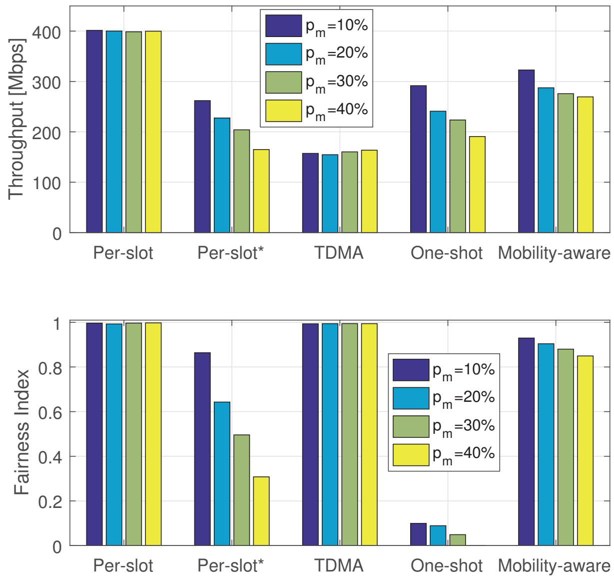

Fig. 7: Throughout and fairness versus mobile user percentage for $K = 45$.

it fails to exploit the multi-user MIMO gain promised by AP cooperation. Thus, it can only achieve about 60% of the throughput of our proposed scheme.

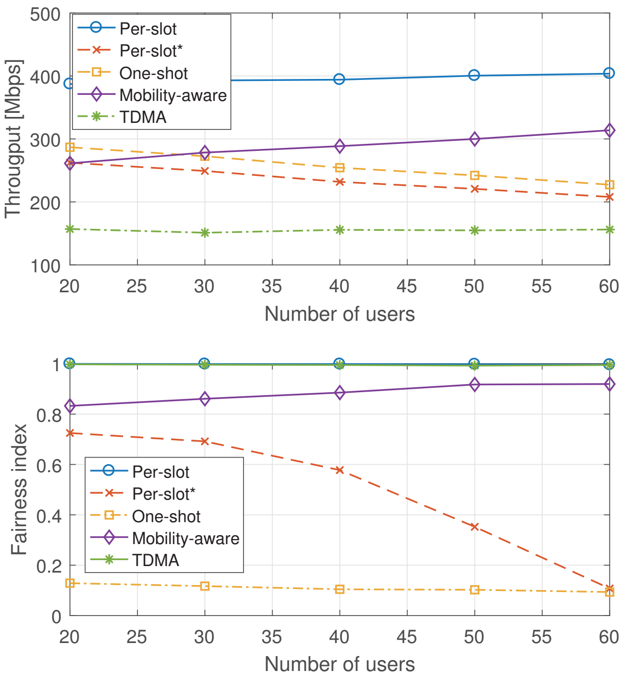

Fig. 8: Throughout and fairness versus mobile user percentage for $K = 45$.

In Figure 8, the achieved throughput and fairness performance is plotted as a function of the number of users. The mobile user percentage is fixed to 20% for all cases. The performance provided by per-slot scheduler without considering the CSI and processing overhead is deemed as the upper bound. With the increase of user numbers, the achievable throughput and fairness of per-slot* scheduler experience sharp decreases. By separating stationary and mobile users and highly optimizing stationary users, we are able to improve throughput and maintain good fairness. The performance of our algorithm actually increases with a large number of users getting to within about 20% of the upper bound throughput and achieving fairness of greater than 0.9.

V. Conclusion

In this paper, we presented a mobility-aware MIMO link scheduling scheme for a cluster of cooperative APs within a dense wireless network. The proposed approach tracks the user channel variation and separates stationary and mobile users into different time slots. Based on the characteristics of stationary and mobile users, different scheduling strategies are applied. On the one hand, for stationary users with slow-varying channels, we combine a set of pre-calculated high-performance CommSets into a high-throughput and fair schedule with sparse CSI update. On the other hand, the performance of mobile users are improved by utilizing timely updated CSI to produce a good CommSet in an efficient way for each time slot. In the presence of limited mobility, our approach exhibits strong performance gains compared to conventional approaches that do not separate mobile and stationary users.

Acknowledgment

This research was supported in part by the U.S. National Science Foundation through Awards CNS–1319455 and CNS–1513884.

References

[1] H. V. Balan, R. Rogalin, A. Michaloliakos, K. Psounis, and G. Caire, "AirSync: Enabling distributed multiuser MIMO with full spatial multiplexing," *IEEE/ACM Trans. Network.*, vol. 21, no. 6, pp. 1681–1695, Dec 2013.

[2] M. Ge and D. M. Blough, "High-throughput and fair scheduling for access point cooperation in dense wireless networks," in *Proc. IEEE WCNC*, March 2017, pp. 1–6.

[3] L. M. Cortes-Pena and D. M. Blough, "MIMO link scheduling for interference suppression in dense wireless networks," in *Proc. IEEE WCNC*, March 2015, pp. 1225–1230.

[4] "IEEE draft standard for IT - telecommunications and information exchange between systems - LAN/MAN - specific requirements - part 11: Wireless LAN medium access control and physical layer specifications - amd 4: Enhancements for very high throughput for operation in bands below 6GHz," *IEEE P802.11ac/D6.0, July 2013*, pp. 1–446, July 2013.

[5] G. Tan and J. Guttag, "Time-based fairness improves performance in multi-rate WLANs," in *Proc. USENIX Conf.*, 2004, pp. 23–23.

[6] D. Blough, G. Resta, and P. Santi, "Interference-aware proportional fairness for multi-rate wireless networks," in *Proc. IEEE INFOCOM*, 2014, pp. 2733–2741.

[7] C. F. Shih, B. Krishnaswamy, and R. Sivakumar, "Rhythm: Achieving scheduled WiFi using purely distributed contention in WLANs," in *Proc. IEEE GLOBECOM*, Dec 2015, pp. 1–7.

[8] M. Ge, J. R. Barry, and D. M. Blough, "Combined user selection and MIMO weight calculation for AP cooperation in dense wireless networks," in *Proc. IEEE WCNC*, March 2017, pp. 1–6.

[9] M. Ge and D. M. Blough, "PBUS: Efficient user selection for block diagonalization in dense wireless networks," in *Proc. IEEE GLOBECOM, Accepted*, 2017.

[10] J. Meinilä, P. Kyösti, T. Jämsä, and L. Hentilä, "WINNER II channel models," *Radio Technologies and Concepts for IMT-Advanced*, pp. 39–92, 2009.