User Pair Selection for Distributed-Input Distributed-Output Wireless Systems

Yung-Shun Wang∗†, Y.-W. Peter Hong∗, and Wen-Tsuen Chen†‡

*Institute of Communications Engineering, National Tsing Hua University, Taiwan
†Institute of Information Science, Academia Sinica, Taiwan
‡Department of Computer Science, National Tsing Hua University, Taiwan
Email: {yswang@erdos.ee, ywhong@ee, wtchen@cs}.nthon.edu.tw

Abstract—This work examines the user-pair selection problem for distributed-input distributed-output (DIDO) wireless systems. A DIDO system refers to a network of densely deployed transmitter and receiver pairs, where the transmitters are connected to and coordinated by a DIDO server. The system sum rate is known to grow without bound as the number of transmitter-receiver pairs increases. However, when zero-forcing (ZF) beamforming is adopted across the transmitters (as assumed in most existing works on DIDO), the effect of power amplification due to ill-conditioned channel matrices may significantly reduce the system sum rate. In this work, a decremental user-pair selection (DUPS) algorithm is proposed to determine the set of transmitter-receiver pairs that should be simultaneously active in order to reduce the impact of power amplification and increase the system sum rate. A low-complexity variant of DUPS is also proposed and an asymptotic lower bound of its sum rate is derived using extreme value theory. Moreover, inspired by the semi-orthogonal user selection (SUS) algorithm, often adopted in the conventional multiple-input multiple-output (MIMO) literature, a semi-orthogonal DUPS algorithm is also proposed by taking into consideration the orthogonality of the users’ channel vectors in the selection process. Simulations are provided to demonstrate the effectiveness of the proposed schemes.

I. INTRODUCTION

Multiple-input multiple-output (MIMO) technology has been widely adopted in modern wireless communication systems. With multiple antennas at the transmitter, multiple users can be served simultaneously using techniques such as zero-forcing (ZF) [1], [2], minimum mean-square error (MMSE) [3], [4], or maximum signal-to-leakage-plus-noise ratio (SLNR) [5]–[7] beamforming. These techniques serve as promising alternatives to the capacity-achieving (but significantly more complex) dirty paper coding scheme [8], [9]. However, regardless of the technique that is being adopted, the number of users that can be served simultaneously is still limited by the number of antennas at the transmitter. This observation is based on the conventional view that transmitters in the downlink are sparse and many users must be served by the same transmitter (e.g., the macrocell base station (BS)). With the increasing density of small (personal) cells, this limitation can potentially be overcome by the so-called distributed-input distributed-output (DIDO) system [10].

Specifically, the DIDO system proposed in [10] envisioned a wireless system with densely deployed transmitters, possibly one for each user. This assumption is increasingly realistic in modern wireless communication systems where users are more and more likely to deploy personal small cells or access points (APs) in their home or office. The transmitters are connected to a DIDO server that coordinates their transmissions and, thus, can serve as a distributed MIMO system whose spatial degrees of freedom scale linearly with the number of users. In [10], all terminals were assumed to have only a single antenna and ZF beamforming was adopted across the transmitters to eliminate multiuser interference. When the channel vectors experienced by the users are not sufficiently orthogonal (i.e., the combined channel matrix is ill-conditioned), power amplification may occur at the transmitter [11] and can lead to a significant loss in sum rate. Hence, in order to achieve a sum rate that scales favorably with the number of users, it is important to carefully schedule the set of transmitter-receiver pairs that are simultaneously active in each time slot.

The main objective of this work is to propose effective user-pair selection algorithms for DIDO systems. In particular, we propose a decremental user pair selection (DUPS) algorithm that initializes with all transmitter-receiver pairs, but then eliminates in each iteration the pair whose removal yields the highest increase in sum rate. A low-complexity variant of the DUPS algorithm is also proposed to reduce the computational complexity and an asymptotic lower bound of the sum rate is derived using extreme value theory. Moreover, inspired by the semi-orthogonal user selection (SUS) algorithm often adopted in the conventional MIMO literature [1], a semi-orthogonal DUPS algorithm is also proposed by taking into consideration the orthogonality of users’ channel vectors in the selection process. Simulation results are provided to demonstrate the effectiveness of the proposed schemes.

User selection problems have also been investigated extensively in conventional MIMO systems, e.g., in [1], [12]–[14], where the number of transmit antennas are typically fixed. In [1], an SUS algorithm was proposed using an incremental strategy where a user is added into the active set in each iteration. The user selected in each iteration must have channel direction that is sufficiently orthogonal with those already selected. In [12], similar user selection algorithms were proposed.
using the capacity and the Frobenius norm as the selection criterion with block diagonalization precoding. In [13], the authors considered a decremental approach that initiates by selecting all users into the active set, but deletes a user with the smallest effective channel gain in each iteration until no further rate improvement can be observed. In [14], a greedy user selection with swap (GUSS) algorithm was proposed. This algorithm first implements in order an incremental and a decremental approach (i.e., an add and a delete operation) as mentioned above to construct an initial user set. Then, users are checked one-by-one to see if a swap with a user outside of the set would yield a higher sum rate. These schemes are effective in conventional MIMO systems, but are not directly applicable to DIDO systems since, in the latter case, the inclusion of any user into the active set is accompanied by that of an additional transmitter and, thus, causes the channel vectors of all users to change after each selection.

II. SYSTEM MODEL AND PROBLEM FORMULATION

Let us consider a DIDO system with \( N \) transmitter-receiver pairs, as shown in Fig. 1. Each pair consists of a single-antenna AP, i.e., the transmitter, and a single-antenna user, i.e., the receiver. Multiple APs can be selected to be active simultaneously and serve their respective users collaboratively. All APs are connected to a DIDO server that coordinates their transmissions and determines the signals that are to be transmitted by the active APs. Following [10], ZF beamforming is adopted among simultaneously active APs to eliminate interference at the corresponding receivers.

Let \( \Omega = \{1, 2, \ldots, N\} \) be the index set of all transmitter-receiver pairs and let \( \mathcal{S} = \{u_1, u_2, \ldots, u_{|\mathcal{S}|}\} \subseteq \Omega \) be the set of those that are simultaneously active, where \( u_i \) is the index of the \( i \)-th selected transmitter-receiver pair. The transmitter and receiver of pair \( u_i \) are referred to as AP \( u_i \) and user \( u_i \), respectively. The channel vector between the set of active APs and user \( u_i \) can be written as \( \mathbf{h}_{\mathcal{S}, u_i} = [h_{u_1,u_i}, h_{u_2,u_i}, \ldots, h_{u_{|\mathcal{S}|},u_i}]^T \), where \( h_{u_i,u_i} \) is the channel coefficient between AP \( u_j \) and user \( u_i \). The channel coefficients are assumed to be independent of each other. Notice that the channel vector experienced by each receiver varies with any addition or deletion of members in \( \mathcal{S} \).

In DIDO systems, the active APs send data collaboratively to their respective users, acting as a single multi-antenna BS. The transmit signals are first computed by the DIDO server and then sent to the respective APs for transmission. Specifically, suppose that \( v_{u_i} \) with \( \text{E}[|v_{u_i}|^2] = 1 \) is the data symbol intended for user \( u_i \), and that \( \mathbf{w}_{u_i} \) is the collaborative beamforming vector used across the APs to send the data symbol. Then, the transmit signal vector (across the \( |\mathcal{S}| \) APs) can be written as \( \mathbf{x}_\mathcal{S} \triangleq [x_{u_1}, x_{u_2}, \ldots, x_{u_{|\mathcal{S}|}}]^T = \sum_{i=1}^{|\mathcal{S}|} \mathbf{w}_{u_i} v_{u_i} = \mathbf{W}_\mathcal{S} \mathbf{v}_\mathcal{S} \), where \( x_{u_i} \) is the signal transmitted by AP \( u_i \), \( \mathbf{W}_\mathcal{S} = [\mathbf{w}_{u_1}, \mathbf{w}_{u_2}, \ldots, \mathbf{w}_{u_{|\mathcal{S}|}}]^T \), and \( \mathbf{v}_\mathcal{S} = [v_1, v_2, \ldots, v_{|\mathcal{S}|}]^T \). The received signal at user \( u_i \) is

\[
y_{u_i} = \mathbf{h}_{\mathcal{S}, u_i}^H \mathbf{x}_\mathcal{S} + n_{u_i} = \sum_{j=1}^{|\mathcal{S}|} \mathbf{h}_{\mathcal{S}, u_i}^H \mathbf{w}_{u_j} v_{u_j} + n_{u_i},
\]

where \( n_{u_i} \) is the additive noise at user \( u_i \). By letting \( \mathbf{y}_\mathcal{S} = [y_{u_1}, y_{u_2}, \ldots, y_{u_{|\mathcal{S}|}}]^T \) and \( \mathbf{H}_\mathcal{S} = [\mathbf{h}_{\mathcal{S}, u_1}, \mathbf{h}_{\mathcal{S}, u_2}, \ldots, \mathbf{h}_{\mathcal{S}, u_{|\mathcal{S}|}}] \), the received signal vector \( \mathbf{y}_\mathcal{S} \) can be written as

\[
\mathbf{y}_\mathcal{S} = \mathbf{H}_\mathcal{S}^H \mathbf{x}_\mathcal{S} + \mathbf{n}_\mathcal{S} = \mathbf{H}_\mathcal{S}^H \mathbf{W}_\mathcal{S} \mathbf{v}_\mathcal{S} + \mathbf{n}_\mathcal{S},
\]

where \( \mathbf{n}_\mathcal{S} = [n_{u_1}, n_{u_2}, \ldots, n_{u_{|\mathcal{S}|}}]^T \) is the additive Gaussian noise vector with entries that are independent and identically distributed (i.i.d.) with zero mean and variance \( \sigma_n^2 \).

By adopting the ZF beamforming as in [10], the matrix of beamforming vectors can be written as \( \mathbf{W}_\mathcal{S} = \mathbf{H}_\mathcal{S}^H \mathbf{D}_\mathcal{S}^{-1} \mathbf{P}_\mathcal{S} \) [15], where \( \mathbf{H}_\mathcal{S}^H \mathbf{D}_\mathcal{S}^{-1} \) is inverse of the channel matrix, \( \mathbf{P}_\mathcal{S} = \text{diag} \{ \sqrt{P_{S,u_1}}, \sqrt{P_{S,u_2}}, \ldots, \sqrt{P_{S,u_{|\mathcal{S}|}}} \} \) is the power allocation matrix and \( \mathbf{D}_\mathcal{S} = \text{diag} \{ \sqrt{|\mathcal{S}| \gamma_{S,u_1}}, \sqrt{|\mathcal{S}| \gamma_{S,u_2}}, \ldots, \sqrt{|\mathcal{S}| \gamma_{S,u_{|\mathcal{S}|}}} \} \) with \( \gamma_{S,u_i} = \frac{1}{|\mathcal{S}|} \) for all \( i \). Here, \( \gamma_{S,u_i} \) can be viewed as the effective channel gain experienced by user \( u_i \) and depends not only on \( u_i \) but also on the members in \( \mathcal{S} \). The i-th column of \( \mathbf{H}_\mathcal{S}^H \mathbf{D}_\mathcal{S}^{-1} \) represents the unit-norm beamforming vector for the data intended for user \( u_i \) and \( P_{S,u_i} \) is the power used to transmit the data. The received signal at user \( u_i \) then becomes

\[
y_{u_i} = \sqrt{|\mathcal{S}| \gamma_{S,u_i} P_{S,u_i}} v_{u_i} + n_{u_i},
\]

for all \( i \). The corresponding receive SNR is \( \gamma_{S,u_i} P_{S,u_i} / \sigma_n^2 \) and the achievable rate is \( \log_2(1 + \gamma_{S,u_i} P_{S,u_i} / \sigma_n^2) \). The transmit signal is assumed to satisfy the total average power constraint \( \text{E}[||\mathbf{x}_\mathcal{S}||^2] = \text{tr} \{ \mathbf{P}_\mathcal{S} \mathbf{D}_\mathcal{S} \} = \sum_{i=1}^{|\mathcal{S}|} P_{S,u_i} \leq \mathcal{P}_T \), which scales linearly with the number of active pairs.

The main objective of this work is to select the set of simultaneously active user pairs with the goal of maximizing the system sum rate. The problem is formulated as follows:

\[
\begin{align*}
\max_{\mathcal{S} \subseteq \Omega, \mathcal{P}_S} & \quad \sum_{i=1}^{|\mathcal{S}|} \log_2 \left( 1 + \frac{\gamma_{S,u_i} P_{S,u_i}}{\sigma_n^2} \right) \\
\text{subject to} & \quad \sum_{i=1}^{|\mathcal{S}|} P_{S,u_i} \leq |\mathcal{S}| \mathcal{P}_T,
\end{align*}
\]

where \( \mathcal{P}_S \triangleq \{P_{S,u_1}, \ldots, P_{S,u_{|\mathcal{S}|}} \} \). Notice that the user-pair selection and the power allocation problems are coupled. However, for fixed \( \mathcal{S} \), the power allocation problem can be obtained in closed-form as the standard water-filling solution

\[
p_{S,u_i} = \left( \frac{1}{\lambda} - \frac{\sigma_n^2}{\gamma_{S,u_i}} \right)^+,
\]
where \( \lambda \) is chosen so that \( \sum_{i=1}^{\left| S \right|} p_{u_i}^{\star} = \left| S \right| P_r \). However, as in many user selection or admission control problems in conventional MIMO systems [1], [12]–[14], the complexity of the user-pair selection problem (i.e., the search for the optimal \( S \)) grows exponentially with the total number of user-pairs. In particular, an exhaustive search requires an exploration of \( \sum_{i=1}^{N} \frac{N!}{|S|!} \) possible combinations. Hence, low-complexity user-pair selection algorithms are proposed in the following section to efficiently determine the set of active user pairs.

### III. PROPOSED DECREMENTAL USER PAIR-SELECTION (DUPS) ALGORITHM

In this section, decremental user-pair selection (DUPS) algorithms are proposed to determine the optimal set of active user pairs with the goal of maximizing the system sum rate. The algorithm is decremental in the sense that the set of active user-pairs is gradually reduced in size by removing one user-pair in each iteration until no further improvement is observed.

Specifically, the DUPS algorithm is initialized by selecting all user-pairs into the active set \( S \) (i.e., by setting \( S = \Omega \)). Let us define \( R_S = \sum_{u \in S} \log_2 (1 + \gamma_{S,u} p_{u}^2 / \sigma_n^2) \) as the sum rate of the set of active user-pairs \( S \), where the optimal power allocation \( \{ p_{u} \} \) can be computed by (6). In each iteration, the user-pair whose absence yields the maximum sum rate is removed from the active set. In other words, that is the active set in the current iteration, then the user-pair to be removed from the set is given by \( u^\star = \arg \max_{u \in S} R_S \setminus \{ u \} \). This process is repeated until there is no further increase in sum rate. The algorithm is summarized as follows.

**Decremental User-Pair Selection (DUPS):**

**Initialize:**

\[ \text{Set } S = \Omega = \{1, 2, \ldots, N\} \]

**Repeat:**

(i) Compute \( R_{S \setminus \{ u \}} \) for all \( u \in S \).

(ii) Find \( u^\star = \arg \max_{u \in S} R_{S \setminus \{ u \}} \).

(iii) If \( R_{S \setminus \{ u^\star \}} \geq R_S \), then set \( S \leftarrow S \setminus \{ u^\star \} \) and go to step (i).

Otherwise, take \( S \) as the desired solution.

**Notice** that, to compute each sum rate in step (i), an inversion of an \((|S| - 1) \times (|S| - 1)\) matrix is required. This must be done \(|S|\) times in each iteration, one for each of the \(|S|\) candidate user-pairs. The complexity of the channel inversion is \(O(|S|^3)\) which dominates the \(O(|S| \log |S|)\) complexity of the water-filling algorithm [16]. With a maximum of \( N \) iterations, the worst-case complexity of the DUPS algorithm is \(O(N^3)\). This is much smaller than the exponential complexity required for exhaustive search.

From the above discussion, we can see that the complexity of the proposed algorithm comes mainly from the iterative nature of the algorithm and requires \(|S|\) matrix inversions in each iteration. In the following, we propose a low-complexity alternative that removes multiple users at once, using only a single iteration. Notice that, by substituting (6) into the sum rate expression in (4) with \( S = \Omega \), we have

\[
R_{\Omega} = \sum_{i \in \Omega} \log_2 \left[ 1 + \left( \frac{\gamma_{i,u}^\star}{\lambda \sigma_n^2} - 1 \right)^+ \right].
\]

From this expression, we can see that the user-pairs associated with smaller effective channel gains contribute less to the total sum rate and, thus, may potentially be removed to reduce the interference on other user-pairs. In particular, let us define the ordering \( \pi_{\Omega} \) such that \( \gamma_{\Omega, \pi_{\Omega}(1)} \leq \gamma_{\Omega, \pi_{\Omega}(2)} \leq \cdots \leq \gamma_{\Omega, \pi_{\Omega}(N)} \). Then, in the proposed low-complexity DUPS (LC-DUPS) algorithm, the first \( k \) users given by the ordering \( \pi_{\Omega} \) (i.e., the users with the \( k \) worst effective channel gains) are removed at once. The optimal \( k \) can be found by searching over all \( k \in \{0, \ldots, N-1\} \). The algorithm is summarized as follows.

**Low-Complexity DUPS (LC-DUPS):**

(i) Compute \( \{ \gamma_{\Omega, i} \} \) \( i \in \Omega \) and set \( U_0 \leftarrow \emptyset, U_1 \leftarrow \{ \pi_{\Omega}(1) \}, U_2 \leftarrow \{ \pi_{\Omega}(1), \pi_{\Omega}(2) \}, \ldots, U_{N-1} \leftarrow \{ \pi_{\Omega}(1), \ldots, \pi_{\Omega}(N-1) \} \).

(ii) Find \( k^\star = \arg \max_k z_{k} \cdot \gamma_{\Omega, k^\star} \).

(iii) Take \( S \leftarrow \Omega \setminus U_k \) as the desired solution.

Notice that, different from the original DUPS algorithm, the LC-DUPS algorithm requires only a total of \( N \) matrix inversions, with the maximum matrix dimension being \( N \times N \). Hence, the worst-case complexity reduces to only \(O(N^4)\) at the cost of a slight reduction in sum rate. The performance of the LC-DUPS algorithm is analyzed in the following section and is demonstrated via simulations in Section VI.

### IV. ASYMPTOTIC ANALYSIS OF DUPS

In this section, we derive performance guarantees for the DUPS algorithm described above. Yet, instead of analyzing the original DUPS algorithm, which can be difficult to do due to its iterative nature, we consider the analysis of the LC-DUPS algorithm where the worst \( k \) users are removed in one shot. For simplicity, we shall assume that all channel coefficients are i.i.d. Gaussian with zero mean and unit variance and that equal power \( P_r \) is allocated to each data symbol.

Specifically, by [17], the effective channel gain observed by user \( i \) under the active set \( S = \Omega \) can be expressed as

\[
\gamma_{\Omega,i} = \left( [H_{\Omega,i}^H H_{\Omega,i}]^{-1} \right)_{i,i} = h_{\Omega,i}^H P_{\Omega,i}^\dagger H_{\Omega,i}^{\Omega,(i)} h_{\Omega,i} \quad (8)
\]

where \( P_{\Omega,i} = I - H_{\Omega,i} (H_{\Omega,i}^H H_{\Omega,i})^{-1} H_{\Omega,i}^H \) and \( H_{\Omega,i}^{\Omega,(i)} \) is a submatrix of \( H_{\Omega,i} \) with the \( i \)-th column removed. By this expression, we can see that the removal of a user from \( \Omega \) increases the dimension of the null space of \( H_{\Omega,i}^{\Omega,(i)} \) (which increases the effective channel gain of all other users), but also reduces the number of entries in the channel vectors. The user-pair selection should thus exploit the tradeoff between these two effects. Moreover, by [18], we know that \( \gamma_{\Omega,i} \) is exponential with parameter \( 1/2 \), and thus its cumulative distribution function (CDF) is given by

\[
F_{\gamma}(z) = 1 - e^{-\frac{z}{2}}.
\]

Suppose that the worst \( k \) pairs \( \pi_{\Omega}(1), \ldots, \pi_{\Omega}(k) \) are removed simultaneously in the LC-DUPS algorithm, where \( \pi_{\Omega} \)
is the ordering defined previously. Even though the channel vectors are different after the removal of the worst \( k \) user-pairs, it can be verified through simulations that, for \( k = k^* \), the minimum rate of all remaining users is in fact closely approximated by \( \log_2(1 + \gamma_{\Omega\setminus\{k+1\}} P_r / \sigma_n^2) \), i.e., the rate of user \( \pi_\Omega(k+1) \) before the removal of the worst \( k \) users. However, the order statistics of the \((k+1)\)-th worst user rate is difficult to obtain in closed-form, especially when the users’ rates are correlated. Hence, for analytical tractability, we resort to the large system analysis and the i.i.d. approximation on the rates are correlated. Hence, for analytical tractability, we resort to the large system analysis and the i.i.d. approximation on the rates are correlated. Hence, for analytical tractability, we resort to the large system analysis and the i.i.d. approximation on the rates are correlated. Hence, for analytical tractability, we resort to the large system analysis and the i.i.d. approximation on the rates are correlated. Hence, for analytical tractability, we resort to the large system analysis and the i.i.d. approximation on the rates are correlated. Hence, for analytical tractability, we resort to the large system analysis and the i.i.d. approximation on the rates are correlated. Hence, for analytical tractability, we resort to the large system analysis and the i.i.d. approximation on the rates are correlated. Hence, for analytical tractability, we resort to the large system analysis and the i.i.d. approximation on the rates are correlated. Hence, for analytical tractability, we resort to the large system analysis and the i.i.d. approximation on the rates are correlated. Hence, for analytical tractability, we resort to the large system analysis and the i.i.d. approximation on the rates are correlated. Hence, for analytical tractability, we resort to the large system analysis and the i.i.d. approximation on the rates are correlated. Hence, for analytical tractability, we resort to the large system analysis and the i.i.d. approximation on the rates are correlated.

In particular, by Theorem 7.5 in [19], it follows that, for \( \rho \in (0,1) \) with \( f_\gamma(F_\gamma^{-1}(\rho)) \neq 0 \), we have

\[
\sqrt{N}[\gamma_{\Omega\setminus\{k+1\}} - F_\gamma^{-1}(\rho)] \sim \mathcal{N}\left(0, \frac{\rho(1-\rho)}{[f_\gamma(F_\gamma^{-1}(\rho))]^2}\right),
\]

where \( F_\gamma \) and \( f_\gamma \) are the CDF and the PDF of \( \gamma_{\Omega\setminus\{k+1\}} \), respectively. Notice that the variance of \( \gamma_{\Omega\setminus\{k+1\}} \) goes to zero as \( N \to \infty \). Therefore, \( \gamma_{\Omega\setminus\{k+1\}} \) can be approximated asymptotically by \( F_\gamma^{-1}(\rho) = -2 \ln(1-\rho) \) for \( N \) sufficiently large. The sum rate of the remaining set of user-pairs can then be approximately lower bounded as

\[
R_{\Omega\setminus\{k\}} \gtrsim (N-k) \log_2 \left( 1 + \frac{P_r}{\sigma_n^2} \gamma_{\Omega\setminus\{k+1\}} \right)
\approx N(1-\rho) \log_2 \left[ 1 - \frac{2P_r}{\sigma_n^2} \ln(1-\rho) \right].
\]

This implies that, with user-pair selection, the sum rate of the system increases linearly with rate of at least \( (1-\rho) \log_2[1-(2P_r/\sigma_n^2) \ln(1-\rho)] \) as \( N \) increases. This lower bound can also be viewed as the sum rate achievable when all admitted user-pairs are to be served by the same rate, due to fairness considerations.

V. SEMI-ORTHOGONAL USER-PAIR SELECTION IN DIDO SYSTEMS

The semi-orthogonal user selection (SUS) algorithm [1] is a user selection technique often adopted in the conventional MIMO literature to determine the set of users to serve under a fixed number of transmit antennas. A similar technique can also be adopted in DIDO systems (as to be discussed in this section), but is shown to be less effective due to fundamental differences between the user-pair selection problems in the two systems. In conventional MIMO systems, the channel vectors experienced by users remain the same regardless of the users participating in the transmission. On the contrary, in DIDO systems, the addition of any user-pair into the active set changes the number of transmit antennas as well as the number of entries in the channel vectors. Hence, the orthogonality (or semi-orthogonality) of the users’ channel vectors varies after the addition of any new user-pair into the set.

To examine the performance of the conventional SUS algorithm in DIDO systems, let us first consider the following modification. In particular, let us define the orthogonality of a user-pair set \( S \) as

\[
\phi_S \triangleq \max_{i,j \in S, i \neq j} \frac{|h_{i,j}^H h_{S \setminus i,j}|}{\|h_{S,i}\| \|h_{S,j}\|},
\]

and say that a set \( S \) is \( \beta \)-orthogonal if \( \phi_S \leq \beta \). The SUS algorithm takes an incremental approach where a user is added into the active set in each iteration while maintaining the \( \beta \)-orthogonality of the set until no further improvement in sum rate is observed. In particular, suppose that \( S \) is the set already chosen up to the previous iteration. Then, in the current iteration, the DIDO server first determines a candidate set \( C \) that includes all user-pairs \( u \in \Omega \setminus S \) such that \( \phi_{S\cup\{u\}} \leq \beta \). That is, a user-pair is considered as a candidate only if its addition to the set maintains the \( \beta \)-orthogonality of the set. Then, the user-pair with the maximum channel norm is selected and added into the active set, i.e., the user

\[
u^* = \arg \max_{u \in C} \|h_{S\cup\{u\},u}\|.
\]

The process continues until the candidate set \( C \) becomes empty. The algorithm is referred to as the semi-orthogonal user-pair selection (SUPS) algorithm and is summarized as follows.

Semi-Orthogonal User-Pair Selection (SUPS) Algorithm:

Initialize: Set \( S = \emptyset \) and \( C = \Omega \).

Repeat:

(i) Set \( S \leftarrow S \cup \{u^*\} \), where \( u^* = \arg \max_{u \in C} \|h_{S\cup\{u\},u}\| \).
(ii) Find the candidate set \( C = \{u \in \Omega \setminus S : \phi_{S\cup\{u\}} \leq \beta\} \).
(iii) If \( C = \emptyset \), then goto step (i). Otherwise, take \( S \) as the desired set of user-pairs.

In [1], \( \beta \) was chosen as 0.2-0.4 for a system with more than 100 users, but with only 4 transmit antenna. However, this scheme is less effective in DIDO systems because the orthogonality of channel vectors depends on the dimension of the transmit antennas, which is fixed for conventional MIMO systems, but varies after each addition of user-pairs in DIDO systems. In fact, as more user-pairs are selected in DIDO systems, the number of transmit antennas increases and the probability that the channel vectors become \( \beta \)-orthogonal converges to 1. Hence, for any fixed \( \beta \), it is often the case that the selection process either terminates prematurely (because \( \beta \)-orthogonality is difficult to maintain with small number of transmit antennas) or does not terminate until all user-pairs are selected (since the channel vectors become near-orthogonal when the number of transmit antennas is large).

Nonetheless, motivated by the SUS algorithm, we propose in the following a variation of the SUPS algorithm where the user removed from the set in each iteration is determined solely by the orthogonality of the set after its remove. More specifically, suppose that \( S \) is the user-pair set obtained after the previous iteration. Then, in the current iteration, the user-pair \( u^* = \arg \min_{u \in S} \phi_{S\setminus\{u\}} \) is determined and removed from the set. The process continues until there is no further increase in sum rate. The algorithm is summarized as follows.
transmit SNR per AP is defined as $P$.

In i.i.d. channels, the results are averaged over 1000 channel realizations whereas, for the case with non-i.i.d. channels, the results are averaged over 60 different user locations and 500 channel realizations for each location. The transmit SNR per AP is defined as $P_T/\sigma_n^2$.

In Fig. 2, we show the average sum rate of the DUPS, LC-DUPS, S-DUPS, SUPS (with $\beta = 0.55$) algorithms and the exhaustive search method versus the transmit SNR per AP. Here, we set $N = 15$. We can see that the performance of DUPS is near the optimal result. For the proposed methods, the DUPS algorithm performs the best, and followed in order by LC-DUPS and S-DUPS. The gains with respect to the baseline already near-orthogonal, causing almost all users to be selected in each iteration.

In this section, simulations are provided to demonstrate the effectiveness of the proposed DUPS algorithms. The case where all user-pairs are active is used as the baseline for comparison. In Figs. 2-4, we assume that the channel coefficients are i.i.d. $\mathcal{CN}(0, 1)$, whereas in Fig. 5, the channel coefficients are weighted by their respective path loss. For the case with i.i.d. channels, all results are averaged over 1000 channel realizations whereas, for the case with non-i.i.d. channels, the results are averaged over 60 different user locations and 500 channel realizations for each location. The transmit SNR per AP is defined as $P_T/\sigma_n^2$.

In Fig. 2, we show the average sum rate of the DUPS, LC-DUPS, S-DUPS, SUPS (with $\beta = 0.55$) algorithms and the exhaustive search method versus the transmit SNR per AP. Here, we set $N = 15$. We can see that the performance of DUPS is near the optimal result. For the proposed methods, the DUPS algorithm performs the best, and followed in order by LC-DUPS and S-DUPS. The gains with respect to the baseline is about 3, 2, and 1 dB, respectively. However, the rate advantage with DUPS comes at the cost of much higher complexity. Notice that, even though LC-DUPS and S-DUPS have the same worst case complexity, S-DUPS actually requires much less computational time since the process often requires much less than $N$ iterations. The SUPS algorithm (i.e., a variant of the SUS algorithm for DIDO systems) performs actually worse than the baseline due to earlier termination, as described in Section V.

In Fig. 3, we show the average sum rate of the above-mentioned algorithms versus the total number of user-pairs $N$. The transmit SNR per AP is 15 dB. In this figure, we can see that the sum rate of DIDO increases linearly with the number of user-pairs, regardless of the user-pair selection algorithm. However, the rate of increase is different for each scheme, causing the rate improvement to increase as the number of user-pairs increases. It is also interesting to observe that, as $N$ increases, the performance of the SUPS algorithm gradually converges towards that of the baseline algorithm. This is due to the fact that, for large number of users, the channel vectors are already near-orthogonal, causing almost all users to be selected under the $\beta$-orthogonality condition. This justifies our claim in Section V.

In Fig. 4 we show the average number of selected user-pairs versus total number of user-pairs $N$. The transmit SNR $P_T/\sigma_n^2$ is 15 dB.

VI. NUMERICAL RESULTS

The complexity of this algorithm is similar to that of LC-DUPS since only a single matrix inversion is required when computing the sum rate $R_{S\setminus\{u^*\}}$ in each iteration.
user-pairs is the about the same for DUPS and LC-DUPS, but the sum rate of the former is considerably higher, as shown in Fig. 3. This implies that even though a similar number of user-pairs is removed, DUPS is able to select the user-pairs more appropriately than LC-DUPS through its iterative procedure. Moreover, we can see that both S-DUPS and SUPS tend to select almost all user-pairs when $N$ is large. This is because the channel vectors become increasingly orthogonal as the number of transmit antennas increases, as described in Section V. Yet, S-DUPS is able to outperform the baseline algorithm in terms of sum rate, but SUPS is not. This shows that a decremental strategy is often better when the desired number of selected user-pairs is generally large.

In Fig. 5, we consider the case where the channel coefficients are non-i.i.d. and plot the average per user rate as well as the average rate of the worst user versus transmit SNR per AP. In particular, we assume that all APs and users are uniformly distributed in a circle with radius 100 meters. The results are averaged over 60 locations and 500 channel coefficients each. More specifically, suppose that $d_{j,i}$ is the distance between AP $j$ and user $i$. Then, the channel coefficient of AP $j$ and user $i$ is given by $h_{j,i} = \min(d_{j,i}, d_0)^{-\alpha/2} \tilde{h}_{j,i}$, where $\alpha = 2$ is the path loss coefficient, $d_0 = 1$ meter is the reference distance and $\tilde{h}_{j,i} \sim \mathcal{CN}(0, 1)$. We can see that the average rate per user and the average rate of the worst user do not vary significantly. This is because, in DIDO systems, the transmitters are more distributed over the area and, thus, there is no user that is significantly farther away from all the transmitters. This is in contrast to conventional MIMO systems where all transmit antennas are at a single location and can be farther away from some users than others. Also, since most users are served in each transmission as shown in the previous figure, a better long-term service guarantee can be provided to the users.

**VII. CONCLUSION**

In this work, we examined the user-pair selection problem for DIDO wireless systems that employ ZF beamforming. Different from conventional MIMO systems, adding or removing any user-pair from the active set causes the channel vectors of all users to change. By taking this into consideration, the DUPS algorithm was first proposed with the goal of maximizing the system sum rate and the performance is near the optimum. Then, a low-complexity variant, called LC-DUPS, was proposed to reduce the computational complexity of the algorithm and was analyzed asymptotically using extreme value theory. Moreover, inspired by the SUS algorithm in the conventional MIMO literature, we also proposed the S-DUPS algorithm that further reduces the complexity by considering only the orthogonality of channel vectors in its selection process. The effectiveness of the proposed schemes were demonstrated through numerical simulations.

**REFERENCES**


