PAPER

Simplified Capacity-Based User Scheduling Algorithm for Multiuser MIMO Systems with Block Diagonalization

Yuyuan CHANG[†a)], ***Student Member and*** **Kiyomichi ARAKI**[†], ***Fellow***

SUMMARY In multiple-input multiple-output (MIMO) systems, the multiuser MIMO (MU-MIMO) systems have the potential to provide higher channel capacity owing to multiuser and spatial diversity. Block diagonalization (BD) is one of the techniques to realize MU-MIMO systems, where multiuser interference can be completely cancelled and therefore several users can be supported simultaneously. When the number of multiantenna users is larger than the number of simultaneously receiving users, it is necessary to select the users that maximize the system capacity. However, computation complexity becomes prohibitive, especially when the number of multiantenna users is large. Thus simplified user scheduling algorithms are necessary for reducing the complexity of computation. This paper proposes a simplified capacity-based user scheduling algorithm, based on analysis of the capacity-based user selection criterion. We find a new criterion that is simplified by using the properties of Gram-Schmidt orthogonalization (GSO). In simulation results, the proposed algorithm provides higher sum rate capacity than the conventional simplified norm-based algorithm; and when signal-to-noise power ratio (SNR) is high, it provides performance similar to that of the conventional simplified capacity-based algorithm, which still requires high complexity. Fairness of the users is also taken into account. With the proportionally fair (PF) criterion, the proposed algorithm provides better performance (sum rate capacity or fairness of the users) than the conventional algorithms. Simulation results also shows that the proposed algorithm has lower complexity of computation than the conventional algorithms.

key words: *MIMO, multiuser diversity, block diagonalization, user scheduling algorithm, simplification, Gram-schmidt orthogonalization*

1. Introduction

Multiple-input multiple-output (MIMO) wireless systems have attracted a lot of attention for their potential of high gain in channel capacity. Recently, attention has shifted to multiuser MIMO (MU-MIMO) systems, which serve several users simultaneously in the domains of frequency and time [1]–[3]. For a system with multiantenna users, block diagonalization (BD) [3] is a technique, where the transmit precoding matrix of each user is designed such that its subspace lies in the null space of all other remaining users, so that multiuser interference (MUI) is completely cancelled.

However, the number of users that can be simultaneously supported with BD is limited by the number of transmit antennas, the number of receive antennas and the richness of the channels [3]. The problem becomes how to choose the subset of users that maximizes the total throughput of a multiuser system containing a large number of users. An exhaustive search over all possible user sets guarantees that the total throughput is maximized, but the computation complexity is prohibitive if the number of users in the system is large.

Two suboptimal user selection algorithms for MU-MIMO systems with BD were proposed in [4] with the aim of maximizing the total throughput while keeping the complexity low. Both algorithms iteratively select users until the maximum number of simultaneously supportable users is reached. The first one is called capacity-based user selection algorithm. In each user selection step, the algorithm selects a user who provides the maximum total throughput with those already selected users. Because the capacity-based user selection algorithm still requires frequent BD capacity calculation, the second one, called Frobenius norm-based user selection algorithm, was also proposed to reduce the complexity of computation furthermore.

Simplified fair scheduling and antenna selection algorithms for BD and successive optimization (SO) system were proposed in [5]. The aim in [5] is to investigate low complexity algorithms for joint user and antenna selection for MU-MIMO system employing either BD or SO. For BD, the user selection algorithm is similar to Frobenius norm-based user selection algorithm in [4], and for further lower complexity, a user grouping technique based on subspace correlation was also proposed, here we call it simplified norm-based user scheduling algorithm. Simplified proportionally fair (PF) scheduling was also introduced for both techniques, where Metric 1 and Metric 2 are PF criteria for BD in [5], to reduce the unfairness of supported rates for the users.

Simplified norm-based algorithm really reduces the complexity of computation of the user scheduling, but also reduces the sum rate capacity performance of the systems. In this paper, we propose a simplified capacity-based user scheduling algorithm, by analysing the capacity-based user selection criterion. We find a new criterion that is simplified by using the properties of Gram-Schmidt orthogonalization (GSO). In simulation results, the proposed algorithm provides higher sum rate capacity than the simplified norm-based algorithm; and when signal-to-noise power ratio (SNR) is high, it provides performance similar to that of the capacity-based algorithm which is very complex.

With the PF criterion, the proposed algorithm provides performance similar to that of the simplified rate-based algorithm (Metric 1) when SNR is higher, but have better fairness of users; when SNR is low, it has fairness similar to that of the algorithm with Metric 1, but provides better per-

Manuscript received November 17, 2010.
Manuscript revised May 23, 2011.
[†]The authors are with the Graduate School of Science and Engineering, Tokyo Institute of Technology, Tokyo, 152-8550 Japan.
a) E-mail: chang@mobile.ee.titech.ac.jp
DOI: 10.1587/transcom.E94.B.2837

formance than it. The proposed algorithm provides fairness similar to that of the simplified norm-based algorithm with PF (Metric 2), but provides higher sum rate capacity than it. Simulation results also shows that the proposed algorithm has lower complexity of computation than the conventional algorithms.

1.1 Notation

In this paper the following notations are used. Uppercase boldface type is used for matrices and lowercase boldface is used for vectors. All scalar quantities are not in boldface. The $(\cdot)^T$ and $(\cdot)^H$ denote the transpose and the conjugate transpose of a matrix, respectively. The $tr(\cdot)$ denotes the trace of a matrix and the $E[\cdot]$ denotes the expectation operator. The $|\cdot|$ and $\|\cdot\|$ denote the determinant and Frobenius norm of a matrix, respectively. Finally, $\mathbf{I}$ and $\mathbf{I}_N$ denotes the identity matrix and the $N \times N$ identity matrix.

The rest of this paper is organized as follows. Section 2 describes the system model and the introduction of BD. In Sect. 3, we introduce the conventional capacity-based and simplified norm-based user scheduling algorithms. In Sect. 4, we analyse the capacity-based user scheduling criteria and propose a simplified capacity-based user scheduling algorithm. The PF is introduced and considered in Sect. 5 for the norm-based and the proposed algorithms. Along with complexity analysis, simulation results are shown in Sect. 6. Finally, we conclude this paper in Sect. 7.

2. System Model

We consider a downlink of a MU-MIMO system with N_t transmit antennas at the base station (BS) and N_k, $k = 1, 2, \ldots, K$, receive antennas at the kth mobile user, where K is the number of multiantenna users. The total number of receive antennas is $N_r = \sum_{k=1}^{K} N_k$, and firstly we consider the simple case that $N_t \geq N_r$.

Let $\mathbf{H}_k$ be an $N_k \times N_t$ complex matrix, and $\mathbf{H}_k$ denote the downlink channel of the kth user. We assume a flat Rayleigh fading spatially uncorrelated channel model so that the elements of $\mathbf{H}_k$, $k = 1, 2, ..., K$ can be modeled as independent and identically distributed (i.i.d.) complex Gaussian random variables with zero mean and unit variance.

The signal vector of user j, $\mathbf{s}_j$ is a vector in $\mathbb{C}^{N_j}$ and is precoded at the transmitter with a $N_t \times N_j$ precoding matrix $\mathbf{M}_j$ to produce transmitted signal vector $\mathbf{x}_j = \mathbf{M}_j\mathbf{s}_j$. The received signal vector of the kth user can be expressed as

$$\mathbf{r}_k = \mathbf{H}_k \sum_{j=1}^{K} \mathbf{M}_j\mathbf{s}_j + \mathbf{n}_k, \tag{1}$$

where $\mathbf{n}_k$ is a vector in $\mathbb{C}^{N_k}$ and denotes the zero mean additive Gaussian noises with variance of σ_n^2. It is noted that this paper is restricted to the case that all the users have the same received noise power level.

When the multiuser interference is eliminated with properly designed precoding matrices $\mathbf{M}_k$, $k = 1, 2, ..., K$ (i.e. $\mathbf{H}_k\mathbf{M}_j = \mathbf{0}\ \forall k \neq j$), the MU-MIMO system is decomposed into parallel single user MIMO channels and the received signal vector for user k becomes $\mathbf{r}_k = \mathbf{H}_k\mathbf{M}_k\mathbf{s}_k + \mathbf{n}_k$. This kind of techniques is called BD, and we will introduce the design of the precoding matrices in the next section.

2.1 Block Diagonalization [3]

Firstly we define an $(N_r - N_k) \times N_t$ aggregate channel matrix $\widetilde{\mathbf{H}}_k \triangleq [\mathbf{H}_1^T \ldots \mathbf{H}_{k-1}^T\ \mathbf{H}_{k+1}^T \ldots \mathbf{H}_K^T]^T$. Zero multiuser interference constraint requires that the precoding matrix $\mathbf{M}_k$ of user k lies in the null space of $\widetilde{\mathbf{H}}_k$, which requires the null space of $\widetilde{\mathbf{H}}_k$ to have dimension greater than 0. Denote the singular value decomposition (SVD) of $\widetilde{\mathbf{H}}_k$ as

$$\widetilde{\mathbf{H}}_k = \widetilde{\mathbf{U}}_k \begin{bmatrix} \widetilde{\boldsymbol{\Sigma}}_k & \mathbf{0} \end{bmatrix} \begin{bmatrix} \widetilde{\mathbf{V}}_k^{(1)} & \widetilde{\mathbf{V}}_k^{(0)} \end{bmatrix}^H, \tag{2}$$

where $\widetilde{\mathbf{U}}_k$ holds the left singular vectors of $\widetilde{\mathbf{H}}_k$ and $\widetilde{\boldsymbol{\Sigma}}_k$ is an $(N_r - N_k) \times (N_r - N_k)$ diagonal matrix with diagonal entries that are the singular values of $\widetilde{\mathbf{H}}_k$. $\widetilde{\mathbf{V}}_k^{(1)}$ holds the first $(N_r - N_k)$ right singular vectors and $\widetilde{\mathbf{V}}_k^{(0)}$ holds the last $(N_t - N_r + N_k)$ right singular vectors. Thus, $\widetilde{\mathbf{V}}_k^{(0)}$ forms an orthogonal basis for the null space of $\widetilde{\mathbf{H}}_k$ (i.e. $\widetilde{\mathbf{H}}_k\widetilde{\mathbf{V}}_k^{(0)} = \mathbf{0}$), and its columns are candidates for the precoding matrix $\mathbf{M}_k$ for user k. Now we define the matrix $\widehat{\mathbf{H}}_k \triangleq \mathbf{H}_k\widetilde{\mathbf{V}}_k^{(0)}$ which can be considered as the block diagonalized channel matrix of user k. This structure allows the SVD to be determined individually for each user. Define the SVD of $\widehat{\mathbf{H}}_k$

$$\widehat{\mathbf{H}}_k = \mathbf{H}_k\widetilde{\mathbf{V}}_k^{(0)} = \widehat{\mathbf{U}}_k \begin{bmatrix} \widehat{\boldsymbol{\Sigma}}_k & \mathbf{0} \end{bmatrix} \begin{bmatrix} \widehat{\mathbf{V}}_k^{(1)} & \widehat{\mathbf{V}}_k^{(0)} \end{bmatrix}^H, \tag{3}$$

where $\widehat{\mathbf{U}}_k$ holds the left singular vectors of $\widehat{\mathbf{H}}_k$ and $\widehat{\boldsymbol{\Sigma}}_k$ is an $N_k \times N_k$ diagonal matrix with diagonal entries, $\sqrt{\lambda_i^{(k)}}$, $i = 1, 2, ..., N_k$, are the singular values of $\widehat{\mathbf{H}}_k$. $\widehat{\mathbf{V}}_k^{(1)}$ holds the first N_k right singular vectors and $\widehat{\mathbf{V}}_k^{(0)}$ holds the last $(N_t - N_r)$ right singular vectors. The product of $\widetilde{\mathbf{V}}_k^{(0)}$ and $\widehat{\mathbf{V}}_k^{(1)}$ now produces an orthogonal basis of dimension N_k and represents the transmission vectors that maximize the information rate for user k subject to producing zero multiuser interference. Thus, the encoding matrix for kth user is defined as

$$\mathbf{M}_k \triangleq \widetilde{\mathbf{V}}_k^{(0)}\widehat{\mathbf{V}}_k^{(1)}. \tag{4}$$

With the encoding matrix $\mathbf{M}_k$, the received signal vector of user k becomes

$$\begin{aligned} \mathbf{r}_k &= \mathbf{H}_k\mathbf{M}_k\mathbf{s}_k + \mathbf{H}_k \sum_{j=1, j\neq k}^{K} \mathbf{M}_j\mathbf{s}_j + \mathbf{n}_k \\ &= \mathbf{H}_k\mathbf{M}_k\mathbf{s}_k + \mathbf{n}_k, \end{aligned} \tag{5}$$

where the second term in the first line represents the MUI and becomes zero because of the precoding matrix design. Then the MU-MIMO channel can be decomposed into individual MIMO channel for each user.

3. User Scheduling Algorithms

When the number of active users K becomes larger and N_r becomes larger than N_t, the BS cannot transmit information to all the users without interference between the users. The BS selects a smaller subset of users from all the K users. The optimal user scheduling strategy is to select the subset of users that maximizes the sum capacity of the MU-MIMO channel at each time slot.

$$\widehat{\mathcal{U}}_s = \arg\max_{\mathcal{U}_l \subset \mathcal{U}} C_{\text{BD}|\mathcal{U}_l}, \tag{6}$$

where $\widehat{\mathcal{U}}_s$ denotes the optimally selected user set, $\mathcal{U}$ denotes the set of all the K users and $\mathcal{U}_l$ is a subset of all the $\mathcal{U}$, and the cardinality of $\mathcal{U}_l$ is less than or equal to the maximum number of simultaneous receiving users $\widehat{K}$. $C_{\text{BD}|\mathcal{U}_l}$ is the sum rate capacity achieved with BD applied to the user set $\mathcal{U}_l$ with average total transmit power P_t and is expressed as

$$C_{\text{BD}|\mathcal{U}_l} = \max_{\{\mathbf{Q}_k:\ \sum_{k\in\mathcal{U}_l} tr(\mathbf{Q}_k)=P_t\}} \sum_{k\in\mathcal{U}_l} \log_2 \left|\mathbf{I} + \frac{1}{\sigma_n^2}\mathbf{H}_k\mathbf{M}_k\mathbf{Q}_k\mathbf{M}_k^H\mathbf{H}_k^H\right|, \tag{7}$$

where $\mathbf{Q}_k = E[\mathbf{s}_k\mathbf{s}_k^H]$ denotes the input covariance matrix of user k and the solution of (7) can be obtained by the water-filling (WF) algorithm over the eigenvalues of $\widehat{\mathbf{H}}_k\widehat{\mathbf{H}}_k^H$, $\forall k \in \mathcal{U}_l$. The transmit power for the user set $\mathcal{U}_l$ is $\sum_{k\in\mathcal{U}_l} tr(\mathbf{M}_k\mathbf{Q}_k\mathbf{M}_k^H) = \sum_{k\in\mathcal{U}_l} tr(\mathbf{M}_k^H\mathbf{M}_k\mathbf{Q}_k) = \sum_{k\in\mathcal{U}_l} tr(\mathbf{Q}_k)$ that is equal to P_t, for that $\widehat{\mathbf{V}}_k^{(0)}$ and $\widehat{\mathbf{V}}_k^{(1)}$ in $\mathbf{M}_k$ are orthonormal bases. It is clear that an exhaustive search over $\mathcal{U}$ is computationally prohibitive if $K \gg \widehat{K}$, so suboptimal user selection algorithms were proposed to reduce the complexity of finding the user set.

In this section, we briefly introduce the capacity-based user scheduling algorithm proposed in [4] and the simplified norm-based user scheduling proposed in [5]. Although antenna selection can increase the performance further [5], we don't employ antenna selection here in order to simplify the problem and gain more insight into the algorithms for user selection.

3.1 Capacity-Based User Scheduling

The capacity-based user selection algorithm is proposed in [4], and is summarized as follows.

Algorithm I: Capacity-based User Scheduling.

1) Initially, let $\mathcal{U} = \{1, 2, ..., K\}$ and $\mathcal{U}_s = \{\phi\}$.
   a. Then select a user u_1 such that $u_1 = \arg\max_{k\in\mathcal{U}} \log_2 \left|\mathbf{I}+\frac{1}{\sigma_n^2}\mathbf{H}_k\mathbf{M}_k\mathbf{Q}_k\mathbf{M}_k^H\mathbf{H}_k^H\right|$.
   b. Set $C_{\text{temp}} = \log_2 \left|\mathbf{I} + \frac{1}{\sigma_n^2}\mathbf{H}_{u_1}\mathbf{M}_{u_1}\mathbf{Q}_{u_1}\mathbf{M}_{u_1}^H\mathbf{H}_{u_1}^H\right|$, and let $\mathcal{U} = \mathcal{U} \setminus \{u_1\}$, $\mathcal{U}_s = \mathcal{U}_s \cup \{u_1\}$.
2) For $i = 2$ to $\widehat{K}$
   a. For every $k \in \mathcal{U}$, let $\mathcal{U}_l = \mathcal{U}_s \cup \{k\}$, calculate the BD sum capacity for the user set $\mathcal{U}_l$: $C_k = \sum_{j\in\mathcal{U}_l} \log_2 \left|\mathbf{I} + \frac{1}{\sigma_n^2}\mathbf{H}_j\mathbf{M}_j\mathbf{Q}_j\mathbf{M}_j^H\mathbf{H}_j^H\right|$.
   b. Let $u_i = \arg\max_{k\in\mathcal{U}} C_k$.
   c. If $C_{u_i} < C_{\text{temp}}$, terminate the algorithm and the selected user set is $\mathcal{U}_s$.
   d. Let $C_{\text{temp}} = C_{u_i}$, $\mathcal{U} = \mathcal{U} \setminus \{u_i\}$ and $\mathcal{U}_s = \mathcal{U}_s \cup \{u_i\}$.
3) Endfor: The selected user set is $\mathcal{U}_s$.

Where $\mathcal{U}$ and $\mathcal{U}_s$ denote the sets of the unselected and selected users, respectively. In step 1, the BS selects the first user with the highest single user MIMO channel capacity, where, for a user k, $\mathbf{M}_k$ holds the first N_k right singular vectors of $\mathbf{H}_k$ and $\mathbf{Q}_k$ is obtained by WF over the eigenvalues of $\mathbf{H}_k\mathbf{H}_k^H$; then the channel capacity of the selected user is memorized by the BS. In step 2, from the remaining unselected users, it finds the user that has the largest value of the BD capacity of the user together with those selected users. But if the total capacity does not increase, the algorithm terminates, even if the number of selected users is smaller than $\widehat{K}$. Algorithm I, which chooses the users with the sum rate capacity of the users, maximizes the total throughput of each step, but frequent BD capacity calculations need to perform a lot of SVDs that is the main load of computation.

3.2 Simplified Norm-Based User Scheduling

The simplified norm-based user selection algorithm is proposed in [5]. We summarize it as follow, but without antenna selection.

Algorithm II: Simplified Norm-based User Scheduling.

1) Initially, let $\mathcal{U}_1 = \mathcal{U} = \{1, 2, ..., K\}$ and $\mathcal{U}_s = \{\phi\}$.
   a. Then select a user u_1 such that $u_1 = \arg\max_{k\in\mathcal{U}} \|\mathbf{H}_k\|^2$.
   b. Let $\mathcal{U}_1 = \mathcal{U}_1 \setminus \{u_1\}$; $\mathcal{U}_s = \mathcal{U}_s \cup \{u_1\}$.
2) For $i = 2$ to $\widehat{K}$
   a. Find the projection matrix $\mathcal{P}_i^{\perp} = \mathbf{I}_{N_t} - \mathbf{V}_{i-1}^H\mathbf{V}_{i-1}$, where $\mathbf{V}_{i-1}$ is the row basis of $\mathbf{H}(\mathcal{U}_s)$, and $\mathbf{H}(\mathcal{U}_s) = [\mathbf{H}_{u_1}^T\ \mathbf{H}_{u_2}^T\ ...\ \mathbf{H}_{u_{i-1}}^T]^T$.
   b. $\mathcal{U}_i = \{k \in \mathcal{U}_{i-1} | (\|\mathbf{H}_k\mathbf{V}_{i-1}^H\| / \|\mathbf{H}_k\|\|\mathbf{V}_{i-1}\|) < \xi\}$
   c. If $\mathcal{U}_i$ is empty, terminate the algorithm and the selected user set is $\mathcal{U}_s$.
   d. Let $u_i = \arg\max_{k\in\mathcal{U}_i} \|\mathbf{H}_k\mathcal{P}_i^{\perp}\|^2 + \|\mathbf{H}(\mathcal{U}_s)\mathcal{P}_k^{\perp}\|^2$ where $\mathcal{P}_k^{\perp} = \mathbf{I}_{N_t} - \mathbf{V}_k^H\mathbf{V}_k$ and $\mathbf{V}_k$ is the row basis of $\mathbf{H}_k$.
   e. Let $\mathcal{U}_i = \mathcal{U}_i \setminus \{u_i\}$ and $\mathcal{U}_s = \mathcal{U}_s \cup \{u_i\}$.
3) Endfor: The selected user set is $\mathcal{U}_s$.

Where $\mathcal{U}_i$ denotes the user selection candidates at the ith step (loop). In step 1, the first user is selected, based on the Frobenius norm of the channel of each user, and in step 2, the row basis $\mathbf{V}_{i-1}$ of the aggregate channel matrix $\mathbf{H}(\mathcal{U}_s)$ can be calculated with SVD or using GSO, and is used to make the projection matrix $\mathcal{P}_i^{\perp}$. The number of candidates is firstly reduced with a threshold of channel spatial correlation ξ, which is designed as in [5], and then from the candidates a user is selected in step 2.d.

4. Proposed User Scheduling Algorithm

In this section, we consider again the user scheduling criterion for the case: The BS has selected one user and then it tries to select the second user from the user pool $\mathcal{U}$ except user u_1. In Algorithm I, the second user is selected to maximize the sum rate capacity with the equation below:

$$u_2 = \underset{\{k\in\mathcal{U};k\neq u_1\}}{\arg\max} \log_2\left|\mathbf{I}+\frac{1}{\sigma_{\mathrm{n}}^2}\mathbf{H}_{u_1}\mathbf{M}_{u_1}\mathbf{Q}_{u_1}\mathbf{M}_{u_1}^H\mathbf{H}_{u_1}^H\right| + \log_2\left|\mathbf{I}+\frac{1}{\sigma_{\mathrm{n}}^2}\mathbf{H}_k\mathbf{M}_k\mathbf{Q}_k\mathbf{M}_k^H\mathbf{H}_k^H\right|, \tag{8}$$

where $\mathbf{M}_{u_1}$ and $\mathbf{M}_k$ are the precoding matrices for user u_1 and user k, lying in the null space of $\mathbf{H}_k$ and $\mathbf{H}_{u_1}$, respectively. $\mathbf{Q}_{u_1}$ and $\mathbf{Q}_k$ are the covariance matrices of the input signals as mentioned in (7) and is optimized by WF over the eigenvalues of $\widehat{\mathbf{H}}_{u_1}\widehat{\mathbf{H}}_{u_1}^H$ and $\widehat{\mathbf{H}}_k\widehat{\mathbf{H}}_k^H$.

For different combination of the selected users, there is a different power allocation obtained by WF. For simplifying the problem, we ignore the effects of the input covariance matrices with setting $\mathbf{Q}_{u_1} = P_{\mathrm{t}}\mathbf{I}_{N_{u_1}}/N_{\mathrm{r}}^{(2)}$ and $\mathbf{Q}_k = P_{\mathrm{t}}\mathbf{I}_{N_k}/N_{\mathrm{r}}^{(2)}$, i.e. considering the signals are transmitted with equal power, where $N_{\mathrm{r}}^{(2)} = N_{u_1} + N_k$ is the sum of the number of receive antennas of user u_1 and k.

And when the BS has selected several users, and tries to select the ith user with step 2.a in Algorithm I, the calculation becomes very complex due to the calculation of the sum rate of BD. For furthermore simplifying the problem, we combine the selected users as one *super user* with channel matrix $\mathbf{H}(\mathcal{U}_{\mathrm{s}})$ as defined in step 2.a of Algorithm II and also use equal power allocation. The selection of the ith user can be simplified as below.

$$u_i = \underset{\{k\in\mathcal{U};k\notin\mathcal{U}_{\mathrm{s}}\}}{\arg\max} \log_2\left|\mathbf{I}+\frac{\rho}{N_{\mathrm{r}}^{(i)}}\mathbf{H}(\mathcal{U}_{\mathrm{s}})\mathbf{M}_{\mathcal{U}_{\mathrm{s}}}\mathbf{M}_{\mathcal{U}_{\mathrm{s}}}^H\mathbf{H}(\mathcal{U}_{\mathrm{s}})^H\right| + \log_2\left|\mathbf{I}+\frac{\rho}{N_{\mathrm{r}}^{(i)}}\mathbf{H}_k\mathbf{M}_k\mathbf{M}_k^H\mathbf{H}_k^H\right|, \tag{9}$$

where $\mathbf{M}_{\mathcal{U}_{\mathrm{s}}}$ and $\mathbf{M}_k$ are the precoding matrices for the super user and user k, and lying in the null space of $\mathbf{H}_k$ and $\mathbf{H}(\mathcal{U}_{\mathrm{s}})$, respectively; $N_{\mathrm{r}}^{(i)} = \sum_{j\in\{\mathcal{U}_{\mathrm{s}},\,k\}} N_j$ is the temporary total number of receive antennas with selected user set where N_j is the number of the jth user's receive antennas; and $\rho = P_{\mathrm{t}}/\sigma_{\mathrm{n}}^2$ is the average SNR.

When the SNR ρ is assumed lower than 1 and close to 0, the maximization of the logarithm functions in (9) approximates to maximize the sum of the eigenvalues of $\mathbf{H}(\mathcal{U}_{\mathrm{s}})\mathbf{M}_{\mathcal{U}_{\mathrm{s}}}\mathbf{M}_{\mathcal{U}_{\mathrm{s}}}^H\mathbf{H}(\mathcal{U}_{\mathrm{s}})^H$ and $\mathbf{H}_k\mathbf{M}_k\mathbf{M}_k^H\mathbf{H}_k^H$, i.e. the Frobenius norm of $\mathbf{H}(\mathcal{U}_{\mathrm{s}})\mathbf{M}_{\mathcal{U}_{\mathrm{s}}}$ and $\mathbf{H}_k\mathbf{M}_k$, respectively. Instead of $\mathbf{M}_{\mathcal{U}_{\mathrm{s}}}$ and $\mathbf{M}_k$, the projection matrices $\mathcal{P}_i^{\perp}$ and $\mathcal{P}_k^{\perp}$ are used to reduce the complexity. Then (9) can be simplified to step 2.d in Algorithm II.

But when SNR is assumed much higher than 1, which is more practical, the simplification in Algorithm II is no more appropriate. We consider the user selection problem in (9) to be the maximization of the product of the determinants.

$$u_i = \underset{\{k\in\mathcal{U};k\notin\mathcal{U}_{\mathrm{s}}\}}{\arg\max} \left|\mathbf{H}(\mathcal{U}_{\mathrm{s}})\mathbf{M}_{\mathcal{U}_{\mathrm{s}}}\mathbf{M}_{\mathcal{U}_{\mathrm{s}}}^H\mathbf{H}(\mathcal{U}_{\mathrm{s}})^H\right| \times\left|\mathbf{H}_k\mathbf{M}_k\mathbf{M}_k^H\mathbf{H}_k^H\right|, \tag{10}$$

where $\left|\mathbf{H}_k\mathbf{M}_k\mathbf{M}_k^H\mathbf{H}_k^H\right|$ can be obtained easily, when $N_k \ll N_{\mathrm{t}}$, but the calculation of $\left|\mathbf{H}(\mathcal{U}_{\mathrm{s}})\mathbf{M}_{\mathcal{U}_{\mathrm{s}}}\mathbf{M}_{\mathcal{U}_{\mathrm{s}}}^H\mathbf{H}(\mathcal{U}_{\mathrm{s}})^H\right|$ becomes very complex, when the number of selected users gets close to $\widehat{K}$. Fortunately, we find that the GSO can reduce the determinant calculation effort, and with the properties of GSO, we propose a simplified capacity-based user scheduling algorithm.

4.1 Gram-Schmidt Orthogonalization (GSO)

Firstly, we assume an arbitrary $N_0 \times N_{\mathrm{t}}$ channel matrix $\mathbf{H}$, where $N_0 \leq N_{\mathrm{t}}$ is the number of row of $\mathbf{H}$. With GSO, $\mathbf{H}$ can be decomposed into a lower triangular matrix $\mathbf{L}$ and an orthonormal set of row vectors $\mathbf{Z}$ that is the row basis of $\mathbf{H}$, i.e. $\mathbf{H} = \mathbf{LZ}$, where $\mathbf{Z} = [\mathbf{z}_1^T \ldots \mathbf{z}_{N_0}^T]^T$. Then the determinant of $|\mathbf{HH}^H|$ becomes the product of the squared diagonal entries of $\mathbf{L}$, i.e. $|\mathbf{HH}^H| = \prod_{i=1}^{N_0} |l_{i,i}|^2$. For emphasising the iterative calculation in GSO, here we define the projection matrix as $\mathbf{B}$, which is equal to $\mathcal{P}^{\perp}$ in Algorithm II.

$$\mathbf{B} = \mathbf{I}_{N_{\mathrm{t}}} - \sum_{i=1}^{N_0} \mathbf{z}_i^H\mathbf{z}_i = \mathbf{I}_{N_{\mathrm{t}}} - \mathbf{Z}^H\mathbf{Z}, \tag{11}$$

where $\mathbf{B}$ is the projection matrix to the null space of $\mathbf{H}$, i.e. $\mathbf{HB} = \mathbf{0}$, and $\mathbf{BB}^H = \mathbf{BB} = \mathbf{B}$. Although the projection matrix calculated by GSO is neither unitary nor orthonormal, GSO is much simpler than SVD; so we use the projection matrix instead of the orthonormal basis for the null space of the channel matrix, like that is calculated in (2), in the user selection. Here we summarize two properties as the lemmas below.

Lemma 1. Consider $\mathbf{H}$ as an aggregate channel matrix that $\mathbf{H} = [\mathbf{H}_1^T\ \mathbf{H}_2^T]^T$. The determinant of $\mathbf{HH}^H$ can be calculated with GSO in steps and becomes

$$\left|\mathbf{HH}^H\right| = \left|\mathbf{H}_1\mathbf{H}_1^H\right| \times \left|\mathbf{H}_2\mathbf{B}_1\mathbf{H}_2^H\right|, \tag{12}$$

where $\mathbf{B}_1$ is the projection matrix to the null space of $\mathbf{H}_1$, that calculated from the row basis of $\mathbf{H}_1$ as (11).

Proof. Let the GSO of $\mathbf{H}$ to be

$$\mathbf{H} = \begin{bmatrix} \mathbf{L}_1 & \mathbf{0} \\ \mathbf{D} & \mathbf{L}_2 \end{bmatrix}\begin{bmatrix} \mathbf{Z}_1 \\ \mathbf{Z}_2 \end{bmatrix}, \tag{13}$$

where $\mathbf{L}_1$ and $\mathbf{L}_2$ are lower triangular matrices, and $\mathbf{D}$ is a rectangular matrix. $[\mathbf{Z}_1^T\ \mathbf{Z}_2^T]^T$ is the orthonormal row basis set for $\mathbf{H}$, where $\mathbf{Z}_1$ and $\mathbf{Z}_2$ are associated with $\mathbf{H}_1$ and $\mathbf{H}_2$, respectively. Then $\mathbf{H}_1 = \mathbf{L}_1\mathbf{Z}_1$, that is the GSO of $\mathbf{H}_1$, and $\mathbf{H}_2 = \mathbf{D}\mathbf{Z}_1 + \mathbf{L}_2\mathbf{Z}_2$. As the definition $\mathbf{B}_1 = \mathbf{I} - \mathbf{Z}_1^H\mathbf{Z}_1$ and $\mathbf{Z}_1\mathbf{Z}_2^H = \mathbf{0}$, we can get the result below.

$$\mathbf{H}_2\mathbf{B}_1 = \mathbf{D}\mathbf{Z}_1\mathbf{B}_1 + \mathbf{L}_2\mathbf{Z}_2\mathbf{B}_1 = \mathbf{L}_2\mathbf{Z}_2, \tag{14}$$

that is the GSO of $\mathbf{H}_2\mathbf{B}_1$. The determinant of $\mathbf{H}\mathbf{H}^H$ is the product of the squared diagonal entries of $\mathbf{L}_1$ and $\mathbf{L}_2$ that is the product of $\left|\mathbf{H}_1\mathbf{H}_1^H\right|$ and $\left|\mathbf{H}_2\mathbf{B}_1\mathbf{H}_2^H\right|$. □

By the way, the projection matrix $\mathbf{B}$ to the null space of $\mathbf{H}$ in Lemma 1 becomes (15), that can be calculated iteratively, in the calculation of GSO for $\mathbf{H}_2\mathbf{B}_1$.

$$\mathbf{B} = \mathbf{I} - \mathbf{Z}_1^H\mathbf{Z}_1 - \mathbf{Z}_2^H\mathbf{Z}_2 = \mathbf{B}_1 - \mathbf{Z}_2^H\mathbf{Z}_2. \tag{15}$$

Lemma 2. Consider another aggregate channel matrix $\mathbf{H}' = [\mathbf{H}_2^T\ \mathbf{H}_1^T]^T$, where $\mathbf{H}_1$ and $\mathbf{H}_2$ are those defined in Lemma 1. Then $\left|\mathbf{H}'\mathbf{H}'^H\right|$ is equal to $\left|\mathbf{H}\mathbf{H}^H\right|$, i.e.

$$\left|\mathbf{H}_1\mathbf{H}_1^H\right| \times \left|\mathbf{H}_2\mathbf{B}_1\mathbf{H}_2^H\right| = \left|\mathbf{H}_2\mathbf{H}_2^H\right| \times \left|\mathbf{H}_1\mathbf{B}_2\mathbf{H}_1^H\right|, \tag{16}$$

where $\mathbf{B}_2$ is the projection matrix to the null space of $\mathbf{H}_2$, that calculated from the row basis of $\mathbf{H}_2$ as (11).

Proof. Firstly, we define the SVD of $\mathbf{H}$ as

$$\mathbf{H} = \mathbf{U}\begin{bmatrix}\mathbf{\Sigma} & \mathbf{0}\end{bmatrix}\mathbf{V}^H = \begin{bmatrix}\mathbf{U}_1 \\ \mathbf{U}_2\end{bmatrix}\begin{bmatrix}\mathbf{\Sigma} & \mathbf{0}\end{bmatrix}\mathbf{V}^H, \tag{17}$$

where $\mathbf{U}$ holds the left singular vectors of $\mathbf{H}$; $\mathbf{\Sigma}$ is a $N_0 \times N_0$ diagonal matrix with diagonal entries are the singular values of $\mathbf{H}$; and $\mathbf{V}$ holds the first N_0 right singular vectors of $\mathbf{H}$. Consequently, the determinant of $\mathbf{H}\mathbf{H}^H$ becomes the product of the eigenvalues of $\mathbf{H}\mathbf{H}^H$, i.e. the product of the squared entries of $\mathbf{\Sigma}$. $\mathbf{U}_1$ and $\mathbf{U}_2$ are defined as the left singular vectors of $\mathbf{H}$ corresponding to $\mathbf{H}_1$ and $\mathbf{H}_2$, respectively; i.e. $\mathbf{H}_1 = \mathbf{U}_1[\mathbf{\Sigma}\ \mathbf{0}]\mathbf{V}^H$ and $\mathbf{H}_2 = \mathbf{U}_2[\mathbf{\Sigma}\ \mathbf{0}]\mathbf{V}^H$ are held[†]. Obviously, when $\mathbf{H}$ becomes $\mathbf{H}'$, only the order of $\mathbf{U}_1$ and $\mathbf{U}_2$ changes, but $\mathbf{\Sigma}$ and $\mathbf{V}$ are unchanged. □

From Lemma 1, it is known that we can get the determinant user by user with GSO; and from Lemma 2, it is possible to change a complex determinant calculation into a simpler one.

In this paper, we try to use GSO to calculate the determinant of the matrices for simplifying the complexity. In the lemmas, the projection matrix becomes much more important than the row basis. Therefore, we define the GSO function as below.

GSO function: calculating the determinant of $\dot{\mathbf{H}}\dot{\mathbf{B}}_0\dot{\mathbf{H}}^H$ with Gram-Schmidt orthogonalization.

1) Initially, the inputs of the function are $\dot{\mathbf{B}}_0$ and $\dot{\mathbf{H}} = [\dot{\mathbf{h}}_1^T\ \dot{\mathbf{h}}_2^T ... \dot{\mathbf{h}}_{N_0}^T]^T$, where $\dot{\mathbf{h}}_i$ is the ith *row vector* of $\dot{\mathbf{H}}$. If $\dot{\mathbf{B}}_0 = \mathbf{I}_{N_t}$, let $\bar{\mathbf{h}}_1 = \dot{\mathbf{h}}_1$, else $\bar{\mathbf{h}}_1 = \dot{\mathbf{h}}_1\dot{\mathbf{B}}_0$.
2) For $i = 1$ to N_0
   a. Calculate $g_i = \bar{\mathbf{h}}_i\bar{\mathbf{h}}_i^H$, the squared length of $\bar{\mathbf{h}}_i$.
   b. $\dot{\mathbf{z}}_i^H\dot{\mathbf{z}}_i = \bar{\mathbf{h}}_i^H(\bar{\mathbf{h}}_i/g_i)$.
   c. $\dot{\mathbf{B}}_i = \dot{\mathbf{B}}_{i-1} - \dot{\mathbf{z}}_i^H\dot{\mathbf{z}}_i$.
   d. If $i < N_0$ then $\bar{\mathbf{h}}_{i+1} = \dot{\mathbf{h}}_{i+1}\dot{\mathbf{B}}_i$.
3) Endfor: return $\widetilde{\mathbf{B}}_0 = \dot{\mathbf{B}}_{N_0}$ and the determinant $\widetilde{g}_0 = \prod_{i=1}^{N_0} g_i$.

Where $\dot{\mathbf{H}}$ is an $N_0 \times N_t$ input channel matrix, where $N_0 \leq N_t$ is the number of row of $\dot{\mathbf{H}}$, and $\dot{\mathbf{B}}_0$ is the input projection matrix; and the output $\widetilde{g}_0$ becomes the determinant of $\dot{\mathbf{H}}\dot{\mathbf{B}}_0\dot{\mathbf{H}}^H$ and the output $\widetilde{\mathbf{B}}_0$ is the projection matrix to the null space of $\dot{\mathbf{H}}\dot{\mathbf{B}}_0$. In the GSO function, we process the input $\dot{\mathbf{H}}\dot{\mathbf{B}}_0$ row by row and the projection matrix is calculated iteratively, where $\dot{\mathbf{B}}_i$ is a temporary result of $\widetilde{\mathbf{B}}_0$ in the ith loop of the "For" loops and the size of $\dot{\mathbf{B}}_i$ and $\widetilde{\mathbf{B}}_0$ is $N_t \times N_t$.

4.2 Simplified Capacity-Based User Scheduling

Back to the user selection problem, we further simplify (10) into the equation below, where we just substitute the projection matrices for the precoding matrices.

$$u_i = \underset{\{k \in \mathcal{U}; k \notin \mathcal{U}_s\}}{\arg\max} \left|\mathbf{H}(\mathcal{U}_s)\mathbf{B}_k\mathbf{H}(\mathcal{U}_s)^H\right| \times \left|\mathbf{H}_k\mathbf{B}_{\mathcal{U}_s}\mathbf{H}_k^H\right|, \tag{18}$$

where $\mathbf{B}_k$ and $\mathbf{B}_{\mathcal{U}_s}$ are the projection matrices to the null spaces of $\mathbf{H}_k$ and $\mathbf{H}(\mathcal{U}_s)$, respectively, which can be calculated with GSO function. When N_k is small, calculating $\left|\mathbf{H}_k\mathbf{B}_{\mathcal{U}_s}\mathbf{H}_k^H\right|$ is a simple task, but calculating $\left|\mathbf{H}(\mathcal{U}_s)\mathbf{B}_k\mathbf{H}(\mathcal{U}_s)^H\right|$ becomes very complex due to the increasing of selected users. With Lemma 2, the determinant of $\mathbf{H}(\mathcal{U}_s)\mathbf{B}_k\mathbf{H}(\mathcal{U}_s)^H$ becomes

$$\left|\mathbf{H}(\mathcal{U}_s)\mathbf{B}_k\mathbf{H}(\mathcal{U}_s)^H\right| = \frac{\left|\mathbf{H}_k\mathbf{B}_{\mathcal{U}_s}\mathbf{H}_k^H\right| \times \left|\mathbf{H}(\mathcal{U}_s)\mathbf{H}(\mathcal{U}_s)^H\right|}{\left|\mathbf{H}_k\mathbf{H}_k^H\right|}. \tag{19}$$

Instituting (19) into (18), and because $\left|\mathbf{H}(\mathcal{U}_s)\mathbf{H}(\mathcal{U}_s)^H\right|$ is constant with respect to selecting an ith user, the user selection problem is simplified as below.

$$\begin{aligned} u_i &= \underset{\{k \in \mathcal{U}; k \notin \mathcal{U}_s\}}{\arg\max} \frac{\left|\mathbf{H}_k\mathbf{B}_{\mathcal{U}_s}\mathbf{H}_k^H\right|^2 \times \left|\mathbf{H}(\mathcal{U}_s)\mathbf{H}(\mathcal{U}_s)^H\right|}{\left|\mathbf{H}_k\mathbf{H}_k^H\right|} \\ &= \underset{\{k \in \mathcal{U}; k \notin \mathcal{U}_s\}}{\arg\max} \left|\mathbf{H}_k\mathbf{B}_{\mathcal{U}_s}\mathbf{H}_k^H\right|^2 \Big/ \left|\mathbf{H}_k\mathbf{H}_k^H\right|. \end{aligned} \tag{20}$$

With (20), we propose a new simplified capacity-based user scheduling algorithm summarized as below.

Algorithm III: Proposed User Scheduling Algorithm.

1) Initially, let $\mathcal{U} = \{1,\ 2,\ ...,\ K\}$ and $\mathcal{U}_s = \{\phi\}$.
   a. For all $k \in \mathcal{U}$, calculate $\left|\mathbf{H}_k\mathbf{H}_k^H\right|$ with GSO function by setting $\dot{\mathbf{B}}_0 = \mathbf{I}_{N_t}$ and $\dot{\mathbf{H}} = \mathbf{H}_k$ (then $N_0 = N_k$), then let $\widehat{\mathbf{B}}_k = \widetilde{\mathbf{B}}_0$.
   b. Then select a user u_1 such that $u_1 = \arg\max_{k \in \mathcal{U}} \left|\mathbf{H}_k\mathbf{H}_k^H\right|$.

[†]Those are not the equations of SVD for $\mathbf{H}_1$ and $\mathbf{H}_2$, but just are the equations developed from (17).

c. Let $\mathcal{U} = \mathcal{U} \setminus \{u_1\}$; $\mathcal{U}_\mathrm{s} = \mathcal{U}_\mathrm{s} \cup \{u_1\}$; $\bar{\mathbf{B}}_1 = \widehat{\mathbf{B}}_{u_1}$ and $C_\mathrm{temp} = \log_2 \left|\mathbf{I} + \frac{\rho}{N_{u_1}} \mathbf{H}_{u_1} \mathbf{H}_{u_1}^H\right|$.

2) For $i = 2$ to $\widehat{K}$

a. Select a user u_i such that $u_i = \arg\max_{k\in\mathcal{U}} \left|\mathbf{H}_k \bar{\mathbf{B}}_{i-1} \mathbf{H}_k^H\right|^2 / \left|\mathbf{H}_k \mathbf{H}_k^H\right|$

b. Calculate $C_{u_i} = \log_2 \left|\mathbf{I} + \frac{\rho}{N_\mathrm{r}^{(i)}} \mathbf{H}_{u_i} \bar{\mathbf{B}}_{i-1} \mathbf{H}_{u_i}^H\right| + \log_2 \left|\mathbf{I} + \frac{\rho}{N_\mathrm{r}^{(i)}} \mathbf{H}(\mathcal{U}_\mathrm{s}) \widehat{\mathbf{B}}_{u_i} \mathbf{H}(\mathcal{U}_\mathrm{s})^H\right|$, where $\mathbf{H}(\mathcal{U}_\mathrm{s}) = [\mathbf{H}_{u_1}^T\ \mathbf{H}_{u_2}^T\ \dots\ \mathbf{H}_{u_{i-1}}^T]^T$, and the current total number of receive antennas is $N_\mathrm{r}^{(i)} = \sum_{k\in\{\mathcal{U}_\mathrm{s},\, u_i\}} N_k$.

c. If $C_{u_i} < C_\mathrm{temp}$, terminate the algorithm and the selected user set is $\mathcal{U}_\mathrm{s}$.

d. Let $C_\mathrm{temp} = C_{u_i}$, $\mathcal{U} = \mathcal{U} \setminus \{u_i\}$ and $\mathcal{U}_\mathrm{s} = \mathcal{U}_\mathrm{s} \cup \{u_i\}$.

e. Set $\dot{\mathbf{B}}_0 = \bar{\mathbf{B}}_{i-1}$ and $\dot{\mathbf{H}} = \mathbf{H}_{u_i}$ (then $N_0 = N_{u_i}$) to calculate $\left|\mathbf{H}_{u_i} \bar{\mathbf{B}}_{i-1} \mathbf{H}_{u_i}^H\right|$ with GSO function, then let $\bar{\mathbf{B}}_i = \widetilde{\mathbf{B}}_0$.

3) Endfor: The selected user set is $\mathcal{U}_\mathrm{s}$.

In step 1.a of the proposed algorithm, the BS calculates $\left|\mathbf{H}_k \mathbf{H}_k^H\right|$ with the GSO function for each user k, and remembers the determinants and the projection matrices which will be used in steps 1.b, 1.c, 2.a and 2.b. In step 1.b, the BS selects the first user, u_1, that has the largest $\left|\mathbf{H}_k \mathbf{H}_k^H\right|$. In step 1.c, the capacity of the MIMO channel of user u_1 with *equal power allocation* for the subchannels is calculated and stored, where N_{u_i} is the receive antenna number of user u_i.

In step 2, the "For" loops, the BS selects the second to the $\widehat{K}$th users, where in step 2.a, the BS selects the ith user, u_i, as (20), and the determinant, $\left|\mathbf{H}_k \bar{\mathbf{B}}_{i-1} \mathbf{H}_k^H\right|$, can be calculated with normal determinant or Gaussian elimination, depending on the computation complexity.

In step 2.b, the simplified BD capacity was calculated considering the SNR and *equal power allocation* for each subchannel, where $N_\mathrm{r}^{(i)}$ is temporarily the total number of receive antennas of the selected users and the u_i in the ith loop. When C_{u_i} is smaller than C_temp, it is considered that the capacity will not increase with the increasing of simultaneous receiving users; the algorithm is terminated and the selected user set is $\mathcal{U}_\mathrm{s}$. Otherwise, the BS remembers C_{u_i} and selects u_i.

Finally, the $\bar{\mathbf{B}}_i$ is updated with the GSO function, considering the channel matrix of the new selected user. In this algorithm, the most complex calculation, step 2.b, is done only once per loop with the projection matrices $\bar{\mathbf{B}}_{i-1}$ and $\widehat{\mathbf{B}}_{u_i}$ obtained at the previous loop and in step 1.a, which keeps the calculation complexity low. With step 2.c, the BS decides the appropriate number of simultaneous receiving users, and the calculation effort furthermore decreases.

5. Proportionally Fair User Scheduling

Scheduling algorithms based on throughput maximization may result in some users denied service due to bad channel conditions. PF scheduling provides fairness among users, and still maintains the multiuser diversity gain [6]. PF and other classes of scheduling algorithms can generally be described through the weighted sum rate maximization requirement [7],

$$\mathcal{U}_\mathrm{s} = \arg\max_{\mathcal{U}_l \subset \mathcal{U}} \sum_{k\in\mathcal{U}_l} \mu_k(t) R_k(t), \tag{21}$$

where $\mu_k(t) = 1/\bar{R}_k(t)$ is the priority weight and $R_k(t)$ is the supported rate of the kth user during the tth scheduling time slot. $\bar{R}_k(t)$ is the average rate that is achieved by user k up to time slot $(t-1)$, which is updated as in [7].

$$\bar{R}_k(t+1) = \begin{cases} \delta\bar{R}_k(t) + (1-\delta)R_k(t), & k \in \mathcal{U}_\mathrm{s} \\ \delta\bar{R}_k(t), & k \notin \mathcal{U}_\mathrm{s} \end{cases} \tag{22}$$

where $\delta = 1 - (1/T_c)$ is the forgetting factor, that means the algorithm keeps tracking the average throughput of each user in a past window of length T_c.

5.1 Simplified User Scheduling Algorithms with PF

The rate-based scheduling metric (Metric 1) and the norm-based scheduling metric (Metric 2) were proposed in [5]; and we briefly introduce the algorithms here.

Metric 1: In step 1.a of Algorithm II, the first user is selected such that

$$u_1 = \arg\max_{k\in\mathcal{U}} \mu_{1,k}(t) \log_2 \left|\mathbf{I} + \frac{\rho}{N_\mathrm{t}} \mathbf{H}_k \mathbf{H}_k^H\right|. \tag{23}$$

And in step 2.d, the ith user is selected such that

$$u_i = \arg\max_{k\in\mathcal{U}_i} \mu_{1,k}(t) \log_2 \left|\mathbf{I} + \frac{\rho}{N_\mathrm{t}} \mathbf{H}_k \mathcal{P}_i^\perp \mathbf{H}_k^H\right|. \tag{24}$$

In (23) and (24), the priority weight $\mu_{1,k}(t) = 1/\bar{R}_k(t)$ is the inverse of the average rate, where $\bar{R}_k(t)$ is updated as in (22), and the supported rate is calculated as below, after completion of selecting user set.

$$R_k(t) = \log_2 \left|\mathbf{I} + \frac{1}{\sigma_\mathrm{n}^2} \mathbf{H}_k \mathbf{M}_k \mathbf{Q}_k \mathbf{M}_k^H \mathbf{H}_k^H\right|, \tag{25}$$

where $\mathbf{Q}_k$ is the input covariance matrix of user k as defined in (7).

Metric 2: In step 1.a of Algorithm II, the first user is selected such that

$$u_1 = \arg\max_{k\in\mathcal{U}} \mu_{2,k}(t) \left\|\mathbf{H}_k\right\|^2. \tag{26}$$

And in step 2.d, the ith user is selected such that

$$u_i = \arg\max_{k\in\mathcal{U}_i} \mu_{2,k}(t) \left(\left\|\mathbf{H}_k \mathcal{P}_i^\perp\right\|^2 + \left\|\mathbf{H}(\mathcal{U}_\mathrm{s}) \mathcal{P}_k^\perp\right\|^2\right). \tag{27}$$

In (26) and (27), the priority weight $\mu_{2,k}(t) = 1/\bar{R}'_k(t)$, and the average rate $\bar{R}'_k(t)$, which is the average of the *changed rate* of user k calculated with norm operation without considering SNR, is updated as below.

$$\bar{R}'_k(t) = \begin{cases} \delta\bar{R}'_k(t) + (1-\delta)\left\|\mathbf{H}_k \mathbf{M}_k \mathbf{Q}_k^{\frac{1}{2}}\right\|^2, & k \in \mathcal{U}_\mathrm{s} \\ \delta\bar{R}'_k(t), & k \notin \mathcal{U}_\mathrm{s} \end{cases} \tag{28}$$

5.2 Proposed User Scheduling Algorithm with PF

In this paper, we also propose a PF scheme for our simplified user scheduling algorithm. In step 1.a of Algorithm III, the first user is selected such that

$$u_1 = \arg\max_{k\in\mathcal{U}} \mu_{3,k}(t) \log_2 \left|\mathbf{H}_k\mathbf{H}_k^H\right|. \tag{29}$$

And in step 2.a, the ith user is selected such that

$$u_i = \arg\max_{k\in\mathcal{U}} \mu_{3,k}(t) \log_2 \frac{\left|\mathbf{H}_k\bar{\mathbf{B}}_{i-1}\mathbf{H}_k^H\right|^2}{\left|\mathbf{H}_k\mathbf{H}_k^H\right|}. \tag{30}$$

In (29) and (30), the priority weight $\mu_{3,k}(t) = 1/\bar{R}_k(t)$, where $R_k(t)$ is as that defined in (25). The average rate $\bar{R}_k(t)$ is also updated as in (22).

6. Simulation Results

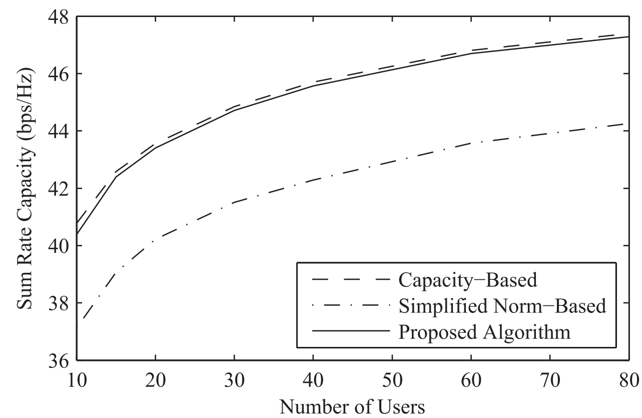
In this section, we show the simulation results of the sum rate capacity of the user scheduling algorithms without PF or with PF, where WF is considered for power control of the subchannels of the selected users in the sum rate capacity calculation. And we also show the calculation complexity of each algorithm. The simulation parameters are set as shown in Table 1, where f_d, T_f and T_c only affect the results of algorithms with PF. The absolute of the three parameters have no specific meaning, but they are designed such that $1 \gg f_d T_f$ to ensure the channels are block fading over time period T_f. In the case of $1 \gg f_d T_f$, when f_d increases, the result of the algorithm with PF gets close to that of the same algorithm without PF, and vice versa.

6.1 Sum Rate Capacity

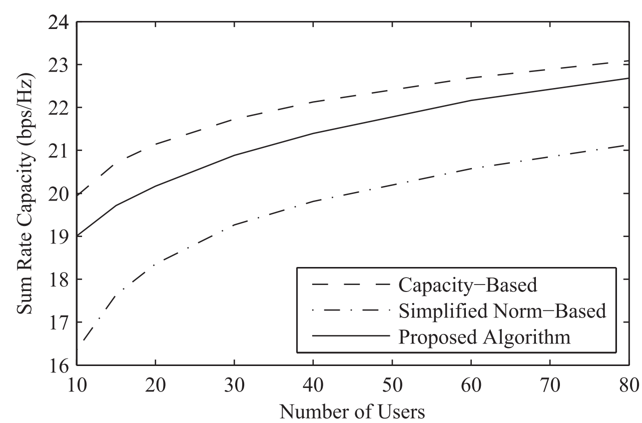
In Fig. 1, we show the sum rate capacity results of the user scheduling algorithms over different average SNRs, where Capacity-Based denotes the result of Algorithm I; Simplified Norm-Based denotes the result of Algorithm II; and Proposed Algorithm denotes the result of Algorithm III. It can be seen that the results of the proposed algorithm are larger than that of the norm-based algorithm. On the other hand, when SNR is large (20 dB), the result of the proposed algorithm is really close to the capacity-based algorithm. When SNR is small (0 dB) and the total number of users is large, the difference between capacity-based and proposed algorithm becomes larger; but when the total number of users is small, the difference becomes smaller for the effect of step 2.c in Algorithm III.

Table 1 Simulation parameters.

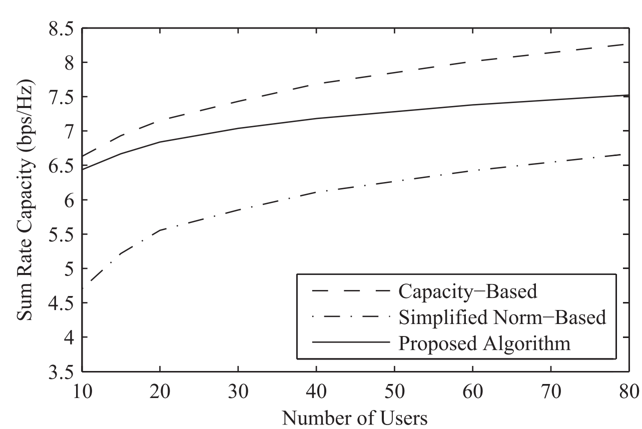
Parameter	Value or Setting
No. of Tx antennas, N_t	8
Total no. of users, K	10, 15, 20, 30, 40, 60, 80
No. of Rx antennas per user, N_k	2
Max. simultaneous users no., $\widehat{K}$	4
Channel model	I.i.d Rayleigh channel
Max. Doppler frequency, f_d	67 Hz
Time period per time slot, T_f	0.5 ms
Width of sliding window, T_c	100 time slots



(a) Average SNR is 20 dB.



(b) Average SNR is 10 dB.



(c) Average SNR is 0 dB.

Fig. 1 Sum rate capacity vs. total number of users.

Table 2 Average number of selected simultaneous receiving users in each user selection when average SNR is 0 dB.

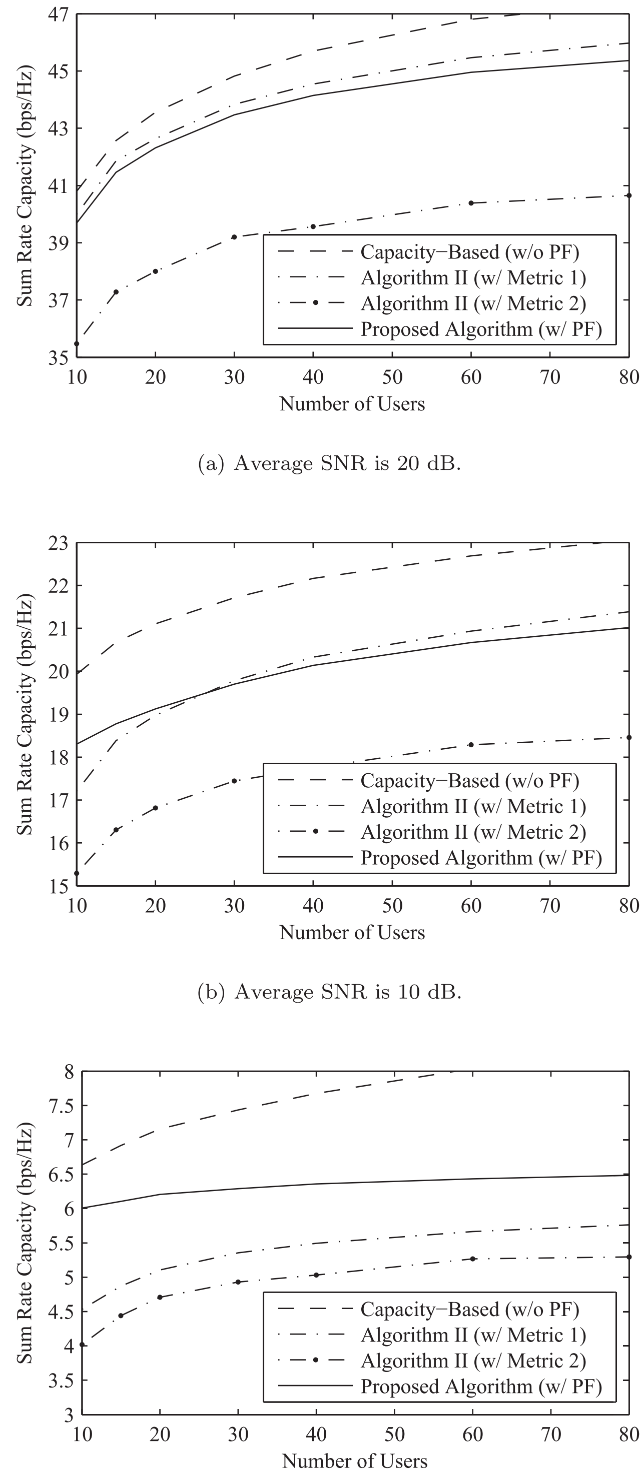
K	10	15	20	30	40	60	80
Algo. I	2.07	2.09	2.12	2.18	2.23	2.38	2.41
Algo. II	4.00	4.00	4.00	4.00	4.00	4.00	4.00
Algo. III	2.52	2.48	2.50	2.70	2.77	2.87	2.92

To explain this phenomenon, we show the average number of selected simultaneous receiving users in each user selection for the algorithms in Table 2, where "Algo." stands for Algorithm. When SNR is low, both Algorithm I and Algorithm II may not select up to $\widehat{K}$ number of users, because of the step 2.c in Algorithm I and Algorithm II, respectively. The results show, when the total user number K is small the average numbers of selected users of Algo. I and Algo. II are close to two, that increases the probability of that the algorithms select the same set of users, so the algorithms have close sum rate capacity results.

In Fig. 2, we show the sum rate capacity results of the user scheduling algorithms over different SNRs considering the PF, where "Capacity-Based (w/o PF)" denotes the result of Algorithm I without PF; "Algorithm II (w/Metric 1)" denotes the result of Algorithm II with Metric 1; "Algorithm II (w/Metric 2)" denotes the result of Algorithm II with Metric 2; and "Proposed Algorithm (w/PF)" denotes the result of Algorithm III with considering PF. It can be seen that the results of the PF scheduling algorithms decrease remarkably than that in Fig. 1 due to the PF criteria. But the results of the proposed algorithm are still higher than that of the norm-based algorithm with Metric 2; and when SNR is large, the result of the proposed algorithm is also closer to the capacity-based algorithm. The performances of Algorithm II (w/Metric 1) and the proposed algorithm are similar to each other when SNR is higher, because they are both simplified capacity-based (or rate-based) algorithms, where we ignore the effect of SNR in the user selection in the proposed algorithm.

For comparing the fairness of the algorithms, we calculate the average and variance of latency, the time between current selection and next selection of each user, for different algorithms. The results are shown in Table 3, in where the unit of the average and variance of latency are in time slot (T_f) and squared time slot (T_f^2), respectively. Because the number of selected user of Algorithm I or the proposed algorithm is smaller than that of Algorithm II, when SNR is lower, the averages of latency of them become larger than Algorithm II. The variance of latency of Algorithm I, that without PF, is larger than that of Algorithm II or the proposed algorithm. The variance of latency of our algorithm is close to that of Algorithm II (Metric 1 or Metric 2) when SNR is low; but it is much smaller than that of Algorithm II (Metric 1) and similar to that of Algorithm II (Metric 2) when SNR is high. From the viewpoint of the variance of latency, the proposed algorithm is almost as fair as Algorithm II (Metric 2) and fairer than Algorithm II (Metric 1).

In the capacity results, we can see that the proposed



(a) Average SNR is 20 dB.

(b) Average SNR is 10 dB.

(c) Average SNR is 0 dB.

Fig. 2 Sum rate capacity (with PF) vs. total number of users.

algorithm provides higher sum rate capacity than Simplified norm-based algorithm; and when SNR is high, it provides performance similar to that of Capacity-based algorithm which is very complex; and we will discuss the complexity further in next section. With the PF criterion, the

Table 3 Average and variance of latency of each user, when total number of users is 40.

(a)Average SNR is 0 dB.

Algorithms	I (w/o PF)	II (Metric 1)	II (Metric 2)	III (w/PF)
Average	12.92	9.98	9.98	13.23
Variance	673.53	154.77	137.48	169.38

(b)Average SNR is 10 dB.

Algorithms	I (w/o PF)	II (Metric 1)	II (Metric 2)	III (w/PF)
Average	12.12	9.98	9.98	10.07
Variance	688.72	211.33	105.84	105.05

(c)Average SNR is 20 dB.

Algorithms	I (w/o PF)	II (Metric 1)	II (Metric 2)	III (w/PF)
Average	9.96	9.98	9.98	9.99
Variance	369.05	212.24	94.61	100.44

Table 4 Complexity of some matrix operations.

Matrix operation	Complexity
$\mathbf{A}_1\mathbf{A}_2$	mnl
$\|\mathbf{A}_1\|^2$	mn
$\lvert\mathbf{A}_3\rvert$	$(m^3-m)/3$
SVD($\mathbf{A}_1$)	$4m^2n+8mn^2+9n^3$
GSO($\mathbf{A}_1$)	$2mn^2+2mn-n^2+m-1$

*Where $\mathbf{A}_1 \in \mathbb{C}^{m\times n}$, $\mathbf{A}_2 \in \mathbb{C}^{n\times l}$ and $\mathbf{A}_3 \in \mathbb{C}^{m\times m}$.

proposed algorithm provides performance similar to that of Algorithm II (w/Metric 1) when SNR is higher, but when SNR is low, it provides better performance than it; and the proposed algorithm is fairer than Algorithm II (w/Metric 1) when SNR is higher, and almost as fair as it when SNR is low. The proposed algorithm provides fairness similar to that of Algorithm II (w/Metric 2), but has higher sum rate capacity than Algorithm II (w/Metric 2).

6.2 Complexity Analysis

In this section we estimate the complexity of calculation with counting the number of complex multiplication operations. Table 4 shows the complexity of some matrix operations. The complexity of determinant is estimated when using Gaussian elimination. In [8], the complexity of SVD is estimated in flop count, but here we treat all the flop counts as complex multiplications for simplification. Finally, the process of GSO is as shown in GSO function in Sect. 4.1, when the inputs $\dot{\mathbf{H}} = \mathbf{A}_1$ (then $N_0 = m$) and $\dot{\mathbf{B}}_0 = \mathbf{I}$.

In Table 5, we show the increasing order of the complexity of BD and the user scheduling algorithms, when the number of receive antenna of each user is N_0, where $N_0 \ll N_t$ and the total number of users is K, where $K \gg \widehat{K} > 1$. The order of BD is calculated with counting the number of SVDs, and the order of that of Algorithm I is calculated with counting the number of BD capacity calculation. As we consider the situation $K \gg 1$, the steps that executed for all or almost all the unselected users determine the complexity of the algorithms. The detail of the complexity of Algorithm II and III is discussed as below.

In step 1.a of Algorithm II, the complexity of the norm operation is N_0N_t; in step 2.b of Algorithm II, the complexity is $N_0N_t^2+2N_0N_t+N_t^2$ including one matrix multiplication and 3 norm operations; and in step 2.d of the ith loop, the complexity is $i(N_0N_t^2+N_0N_t)$ that including two matrix multiplication and two norm operations, where the size of $\mathbf{H}_k$ and $\mathbf{H}(\mathcal{U}_s)$ are $N_0\times N_t$ and $[(i-1)N_0]\times N_t$, respectively. As step 2.b reduces the number of candidates, step 2.d is not executed for all the unselected users; then the upper bound of the complexity of Algorithm II is approximated to $0.5\widehat{K}^2KN_0N_t^2+1.5\widehat{K}KN_0N_t^2$, for $K\gg\widehat{K}$ and $N_t\gg N_0$.

Table 5 Complexity of the algorithms.

Algorithm	Complexity
BD	$O(\widehat{K}N_t^3)$
Algorithm I	$O(\widehat{K}^2KN_t^3)$
Algorithm II	$O(\widehat{K}^2KN_0N_t^2)$
Algorithm III	$O(\widehat{K}KN_0N_t^2)$

In an $N_0\times N_t$ MU-MIMO system, where the BS selects $\widehat{K}$ users from the K users; $N_0 \ll N_t$ and $K \gg \widehat{K} > 1$.

In step 1.a of Algorithm III, the complexity of the GSO is $2N_0N_t^2+2N_0N_t-N_t^2+N_0-1$; and in step 2.a of Algorithm III, the complexity of the determinant $|\mathbf{H}_k\bar{\mathbf{B}}_{i-1}\mathbf{H}_k^H|$ is $N_0N_t^2+N_0^2N_t+N_0^3/3-N_0/3+2$ including the matrix multiplication operations, except $|\mathbf{H}_k\mathbf{H}_k^H|$ that has been calculated at step 1.a. As step 2.c terminates the function when the capacity is considered does not increase, the total selected user number may be smaller than $\widehat{K}$; then the upper bound of the complexity of Algorithm III is approximately $(\widehat{K}+1)KN_0N_t^2$, also for $K\gg\widehat{K}$ and $N_t\gg N_0$.

When PF is taken into account, the complexity of Algorithm II with Metric 2 and Algorithm III with PF is almost no different with that of the original algorithms. However, with Metric 1, the step 1.a in Algorithm II selects the first user with (23), and the complexity of it becomes $N_0^2N_t+N_0^3/3+N_0^2-N_0/3+1$; and the complexity of step 2.d becomes $N_0N_t^2+N_0^2N_t+N_0^3/3+N_0^2-N_0/3+1$ for selecting the ith user with (24), where we ignore the complexity of the logarithm operations. Then the upper bound of the complexity of Algorithm II with Metric 1 is approximately $2(\widehat{K}-1)KN_0N_t^2$, for $K\gg\widehat{K}$ and $N_t\gg N_0$, too.

In Fig. 3, we show the process time of the algorithms through simulation. The simulation parameters are also shown in Table 1 with average SNR is 20 dB, and both the projection matrices in Algorithm II and Algorithm III are calculated with GSO. The process time is normalized with the process time of the simplified norm-based user scheduling algorithm (Algorithm II) when the total number of users is 10. From the results, the process time of Algorithm I (Capacity-Based) soon diverges as the number of users increases, and the process time of Algorithm III (Proposed Algorithm) is almost half of that of Algorithm II. We also show the normalized process time of Algorithm II with Metric 1 (Algorithm II (Metric 1)) that is drawn between Capacity-Based and Proposed Algorithm. Obviously, the proposed algorithm has the lowest complexity of computation.

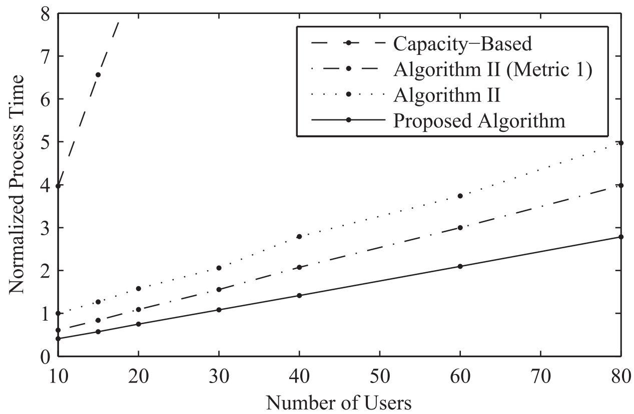


Fig. 3 Normalized process time vs. total number of users.

7. Conclusion

In this paper, we proposed a simplified capacity-based user scheduling algorithm, by analysing the capacity-based user selection criterion. We found a new criterion that is simplified by using the properties of GSO. Taking fairness into account, we also considered the PF in the proposed algorithms. Simulation results showed that the proposed algorithm provided higher sum rate capacity than the simplified norm-based algorithm; and when SNR was high, it provided performance similar to that of the capacity-based algorithm. Fairness of the users was also taken into account. With the PF criterion, the proposed algorithm provided better performance (sum rate capacity or fairness of the users) than the conventional algorithms. Simulation results also showed that the proposed algorithm had lower complexity of computation than the conventional algorithms. In the future work, we will consider systems with unequal SNR users, which is closer to the practical systems.

References

[1] S. Vishwanath, N. Jindal, and A.J. Goldsmith, "Duality, achievable rates, and sum-rate capacity of Gaussian MIMO broadcast channels," IEEE Trans. Inf. Theory, vol.49, no.10, pp.2658–2668, Oct. 2003.

[2] H. Weingarten, Y. Steinberg, and S. Shamai, "The capacity region of the Gaussian MIMO broadcast channel," Proc. IEEE Int. Symp. on Inf. Theory, ISIT'04, p.174, June 2004.

[3] Q.H. Spencer, A.L. Swindlehurst, and M. Haardt, "Zero-forcing methods for downlink spatial multiplexing in multiuser MIMO channels," IEEE Trans. Signal Process., vol.52, no.2, pp.461–471, Feb. 2004.

[4] Z. Shen, R. Chen, J.G. Andrews, R.W. Heath, Jr., and B.L. Evans, "Low complexity user selection algorithms for multiuser MIMO systems with block diagonalization," IEEE Trans. Signal Process., vol.54, no.9, pp.3658–3663, Sept. 2006.

[5] S. Sigdel and W.A. Krzymień, "Simplified fair scheduling and antenna selection algorithms for multiuser MIMO orthogonal space-division multiplexing downlink," IEEE Trans. Veh. Technol., vol.58, no.3, pp.1329–1344, March 2009.

[6] P. Viswanath, D.N.C. Tse, and R. Laroia, "Opportunistic beamforming using dumb antennas," IEEE Trans. Inf. Theory, vol.48, no.6, pp.1277–1294, June 2002.

[7] H. Viswanath, S. Venkatesan, and H. Huang, "Downlink capacity evaluation of cellular networks with known-interference cancellation," IEEE J. Sel. Areas Commun., vol.21, no.5, pp.802–811, June 2003.

[8] G.H. Golub and C.F. Van Loan, Matrix Computations, 3rd ed. The Johns Hopkins Univ. Press, 1996.

Yuyuan Chang was born in 1975. He received the B.S. degree from Department of Control Engineering and the M.S. degree from Department of Electrical Control Engineering, National Chiao Tung University, Hsinchu, Taiwan, R.O.C. in 1997 and 1999 respectively, and another M.S. degree from Electrical and Electronic Engineering Department, Tokyo Institute of Technology, Tokyo, Japan, in 2007. He served in Industrial Technology Research Institute, Hsinchu, Taiwan, R.O.C. from 2000 to 2005. Currently he is pursuing the Ph.D. degree at Electrical and Electronic Engineering Department, Tokyo Institute of Technology, Tokyo, Japan. His research interests include multi-user MIMO systems and user scheduling algorithm.

Kiyomichi Araki was born in 1949. He received the B.S. degree in electrical engineering from Saitama University, in 1971, and the M.S. and Ph.D. degrees in physical electronics both from Tokyo Institute of Technology, in 1973 and 1978, respectively. In 1973–1975, and 1978–1985, he was a Research Associate at Tokyo Institute of Technology, and in 1985–1995 he was an Associate Professor at Saitama University. In 1979–1980 and 1993–1994 he was a visiting research scholar at University of Texas, Austin and University of Illinois, Urbana, respectively. He is currently a Professor at Tokyo Institute of Technology. Dr. Araki is a member of IEEE, IEE and Information Society of Japan. His research interests are information security, coding theory, communication theory, circuit theory, electromagnetic theory, microwave circuits etc.