2010 3rd International Conference on Computer and Electrical Engineering (ICCEE 2010)
IPCSIT vol. 53 (2012) © (2012) IACSIT Press, Singapore
DOI: 10.7763/IPCSIT.2012.V53.No.2.100

Low Complexity Multi-user Scheduling with Quantization

Juan Hang[1+],Yuanfeng Zhang[2] and Feng She[2]

[1]Engineering Training Center, Shanghai Second Polytechnic University

Shanghai 201209, P. R. China

[2]Department of Electronic and Electrical Engineering, Shanghai University of Engineering Science

Shanghai 201620, P. R. China

Abstract—This paper deals with the design and analysis user scheduling algorithm in multi-antenna broadcast (downlink) systems with limited feedback of channel state information. By using quantized codebook, the channel direction can be divided into serval unoverlapped regions. Based on the quantized channel regions, we can get semi-orthogonal region sets by setting semi-orthogonal threshold. Simulation results show that the achieved sum rate by the presented algorithm can achieve nearly optimal sum rate close to the full searching algorithms while with much lower complexity than that of the previous algorithms.

Keywords-MIMO, broadcast channel, downlink scheduling, spatial multiplex.

1. Introduction

Multiple-input multiple-output (MIMO) system is well motivated due to the potential improvements in transmission rate or diversity gain [1], and it is well known that multiple antennas can be easily deployed at base station in cellular systems. However, mobile terminals usually have a small number of antennas due to the size. Thus, it may not be able to obtain significant capacity benefit from multiple transmit antennas[2]. To solve the problem, multiuser must be served simultaneously. One way to accomplish this is called dirty paper coding (DPC), while DPC is with high complexity. As a much simple transmit strategy, zero forcing beamforming (ZFBF) techniques have been proposed, which can greatly reduce the complexity while keeping the throughput region close to optimal[3]. Generally, finding the optimal active users requires an exhaustively search over all users. Currently, this problem has attracted great interest [4], [5]. In [5], the authors proposed a semi-orthogonal user scheduling algorithm to reduce interference among different data streams. In [6], a similar idea was used to develop a greedy user sets selection, which is shown to achieve the optimal asympotic sum rate. In [3], a better user sets selection scheme based on clique (full connected subgraph) graph was proposed. Unfortunetly, the above algorithms are all with high complexity. Thus, it is necessary to find an user scheduling algorithm with low complexity. In [2], a semi-orthogonal user scheduling (SUS) algorithm was proposed. Though SUS is with low complexity, it can not guarantee to get the optimal user sets, which motivates us to find a low complexity user scheduling algorithm which can get better user sets than SUS.

In this paper, we propose a low complexity user scheduling algorithm. The underlying idea is that the multiuser channel can be modeled as a weighted graph by quantized channel.

[+] Corresponding author.
E-mail address: juana124@163.com

The paper is organized as follows. We outline the system model in section II. The proposed scheduling algorithm is presented in section III. We analyze the complexity in section IV. The simulation results are presented in section V. Finally we conclude this paper in section VI.

Notations used in this paper are as follows: $(\cdot)^T$ denotes matrix transposition, $(\cdot)^H$ denotes matrix conjugate-transposition, and $tr(\cdot)$ is trace of channel matrix, $E[\cdot]$ denotes statistical expectation, and $\|\cdot\|^2$ denotes the mean square norm of a vector.

2. Multiuser Broadcast System and Transmit Strategy

2.1. Multiuser broadcast system

We consider a single-cell MIMO system with a single base station supporting data traffic to K users. The base station is with N_t transmit antennas and each of the user terminal has single antenna, and $K \gg N_t$. The signal received by user k is given by

$$y_k = h_k x + n_k, \; k \in \{1, \ldots, K\}, \tag{1}$$

where $x \in \mathbb{C}^{N_t \times 1}$ is the transmit signal vector, and n_k is complex Gaussian noise with unit variance per vector component, and $h_k \in \mathbb{C}^{1 \times N_t}$ is the multiple-input single-output (MISO) channel gain matrix to the k th user.

We employ ZFBF as transmit strategy, where the transmitter first selects an active user set $S \subset \{1, 2, \ldots, K\}$, where the set size $|S| \le N_t$.

Denote $h_i, i \in \{1, \ldots, |S|\}$ as the channel to the i th active user, and define $H(S) = [h_1^T, \ldots, h_{|S|}^T]$. Then the transmit signal is represented as

$$x = \sum_{i=1}^{|S|} \sqrt{P_i} w_i s_i, \tag{2}$$

where s_i, w_i, and P_i are data symbol, beamforming vector, and transmit power for the i th active user, respectively. Then the received signal at the i th active user is given by

$$y_i = \sqrt{P_i} h_i w_i s_i + \sum_{j=1, j \ne i}^{|S|} \sqrt{P_j} h_i w_j s_j + n_j. \tag{3}$$

From channel realization, $H = \{h_1^T, h_2^T, \ldots, h_K^T\}$, Multiuser MIMO can be represented as a node weight graph [3].

2.2. Multiuser Transmit Strategies (ZFBF)

[2] showed that employing ZFBF to a set of N_t nearly orthogonal users with large channel norms is asymptotically optimal as the number of users grow large.

In ZFBF, we first select a user subset S to be served together, and then build the corresponding channel matrix $H(S)$. The beamforming matrix $\mathcal{W}(S)$ is written as

$$\mathcal{W}(S) = H(S)^H (H(S) H(S)^H)^{-1}. \tag{4}$$

As a result, the achievable throughput of ZFBF for a given user set S is given by

$$R_{ZFBF}(S) = \max \sum_{i \in S} \log_2(1 + P_i). \tag{5}$$

We define b_i as the effective channel gain to the i th user. Then the power constraint is

$$\sum_i P_i / b_i \le P, \tag{6}$$

where $b_i = \dfrac{1}{[(H(S)H(S)^H)^{-1}]_{i,i}}$, and P_i in (5) can be easily obtained by waterfilling as

$$P_i = (\mu b_i - 1)^+, \tag{7}$$

where $(x)^+$ denotes $\max\{x, 0\}$, and μ is waterlevel.

3. Scheduling under ZFBF Multiplexing

In this section, we provide scheduling algorithm based on channel quantization.

For finite user number, the probability of existence of an orthogonal set is zero. Thus, we consider the user sets which are ``nearly" orthogonal in scheduling scheme. To be precise, we define two vectors v_1 and v_2 to be $\alpha-$orthogonal if

$$\frac{|v_1 v_2^H|}{\|v_1\| \cdot \|v_2\|} \le \alpha. \tag{8}$$

3.1. Channel quantization and codebook

In this paper, we attempt to divide the channel space into several unoverlapped channel regions. For each of these regions, there is a codeword to denote the channel vector in the region. The set of codewords is called codebook.

We consider codebooks construction from FFT matrices [7, 8]. This class of codewords in the codebook can be thought of as subset of m columns of the $N \times N$ FFT matrix. More precisely, the codebook consists of m distinct columns chosen from an $N \times N$ FFT matrix, with index set $u = [u_1\, u_2 \ldots u_m]$, denoted $\mathcal{C}_{FFT}(u, N)$, be the codebook of size N with codewords taken to be columns o

$$\frac{1}{\sqrt{m}} \begin{pmatrix} 1 & 1 & \cdots & 1 \\ e^{j\frac{\pi}{N}u_1} & e^{j\frac{\pi}{N}u_2} & \cdots & e^{j\frac{\pi}{N}u_m} \\ e^{j\frac{\pi 2}{N}u_1} & e^{j\frac{\pi 2}{N}u_2} & \cdots & e^{j\frac{\pi 2}{N}u_m} \\ \vdots & \vdots & \ddots & \vdots \\ e^{j\frac{\pi(N-1)}{N}u_1} & e^{j\frac{\pi(N-1)}{N}u_2} & \cdots & e^{j\frac{\pi(N-1)}{N}u_m} \end{pmatrix} \tag{9}$$

These codebook is known to achieve the smallest μ for a given N in very special cases.

Using this construction, the inner product magnitude is the same for all codewords. That is,

$$\begin{aligned} \mu_{rms}(\mathcal{C}_{FFT}) &= \max_j \sqrt{\sum_{i \ne j} |c_i^H c_j|^2} \\ &= \sqrt{\sum_{i \ne k} |c_i^H c_k|^2}, \forall c_k \in \mathcal{C}_{FFT}. \end{aligned} \tag{10}$$

3.2. Selecting semi-orthogonal user sets by codebook

The main idea of our user scheduling algorithm with quantized channel is to use semi-orthogonal codewords to construct the semi-orthogonal user sets. The user scheduling algorithm is intended for sum-rate maximization.

Before user scheduling, we firstly construct semi-orthogonal relationship among codewords in a codebook with a certain α which is defined in (8). Because the each codeword denote a channel region, then we can thought the semi-orthogonal relationship among codewords is the same as the semi-orthogonal relationship as channel regions. Thus, from a codebook, we construct a channel-region semi-orthogonal relationship, and α captures pair-wise semi-orthogonal relation between the regions. when $\frac{|c_i \cdot c_j^H|}{\|c_i\|\|c_j\|} \le \alpha$, the two regions R_i and R_j are $\alpha-$orthogonal region (note c_i and c_j are codewords in region R_i and R_j respectively).

Based on the semi-orthogonal region sets, the user set can be chosen. In other words, the semi-orthogonal channel region sets reflects the orthogonal relationship between users. Thus, in user scheduling, we only need to consider users in the $\alpha-$orthogonal channel regions sets.

With certain α and certain codebook, we can calculate the channel region sets which meet with the semi-orthogonal condition. To decide which region that the channel vectors belong to, we define distance as $d(c,h) = |c \cdot h^T|^2$. Then we use the following nearest condition

$$if \quad d(c_i,h) < d(c_j,h) \quad then \quad h \in R_i \tag{11}$$

where c_i and c_j are the codewords in channel region R_i and R_j, respectively.

3.3. Low complexity scheduling algorithm with vector quantization

In this section, we provide user scheduling algorithm which is based on the $\alpha-$orthogonal channel region sets. The motivation of using those sets is that the optimal ZFBF user sets is a clique with high probability when the number of user K is very large.

As we assumed the transmitter knows perfect channel state information (CSI) of users, the transmitter know which channel vector belong to which channel region.

Based on the knowledge, the transmitter carrys out the following scheduling algorithm.

1. In each time slot, the transmitter calculate the project term of each user by

$$p_{hi} = | h_i c_k^H |^2, \tag{12}$$

where h_i is the channel vector of user i which is in channel region k, and c_k is the codeword denoting the channel region. Then in each channel region, the user with the largest project term p_{hi} will be selected.

2. Calculating the capacity of each user set. The user set with the highest capacity will be the selected active user set.

The semi-orthogonal user sets can be constructed by the principle mentioned above. To each of those user sets, we first sort the channel gain $| h_i h_i^H |$ in decreasing order, i.e., $h_1^2 \geq h_2^2 \geq \ldots \geq h_\ell^N$, where $\ell \leq N_t$ is the user number in the user set. Then, in each set, we calculate

$$g_t = h_t - \sum_{i=1}^{t-1} \left(\frac{h_t g_i^H}{\|g_i\|^2} \right) \cdot g_i,\ t = 1, \ldots, \ell. \tag{13}$$

Let $q_r = \sum_{i=1}^{\ell} \|g_i\|^2$, where $r = 1, \ldots, R$. Here, R is the number of semi-orthogonal channel or user set of a certain codebook with certain α constraint. Then, the user selection criteria is as following

$$r = \text{argmax}\, q_r,\ r \in \{1, \ldots, R\}. \tag{14}$$

As a result, the users in channel user set r is the selected active user set in this time slot.

In the others time slots, the transmitter only need to repeat $Step1$ and $Step2$.

4. Complexity analysis

In this section, the complexity of the proposed user scheduling algorithm is analyzed. Due to DPC and greedy user selection algorithm are with high complexity, we only compare the complexity of proposed algorithm with that of SUS algorithm which is with low complexity. The SUS algorithm which is mentioned in [2], which consists two stages: user selection using semi-orthogonal algorithm and a beamforming weight vector calculation. We note that the latter stage has a small fixed complexity, requiring only one $N_t \times N_t$ matrix inversion $W(S) = H(S)^{-1}$ to obtain beamforming weights. We first analysis the complexity of SUS algorithm.

In Step2 of SUS algorithm presented in [2], the complexity of computing $\frac{| g_{(i)}^H g_{(i)} |}{\|g_i\|^2} g_{(i)}$ is $N_t^2 + N_t + 1$, and each user need one $(1 \times N_t) \times (N_t \times N_t)$ vector-matrix multiplication with the complexity is N_t^2.

In Step 3 of SUS algorithm, each user need to calculate the channel vector norm, and the complexity is N_t.

In Step 4, each user need to compute $\frac{| h_k g_{(i)}^H |}{\|h_k\| \cdot \|g_i\|}$, thus the complexity is $N_t + 3$.

Let T be the number of total times of user search in SUS algorithm, then the total computing complexity of SUS algorithm is

$$\begin{aligned} C_{SUS} &= [N_t^2 (T - K) + (N_t^2 + N_t + 1)(M - 1)] \\ &\quad + N_t T + (N_t + 3) T \\ &= (2N_t + 3) K + (N_t^2 + 2N_t + 3)(T - K) \end{aligned} \tag{15}$$

Now, let us analysis the complexity of the proposed scheduling algorithm. As mentioned in section III, there are 2 Steps in user set selection.

In Step1, each user needs to compute $\|h_i c_k^H\|^2$. Then the complexity is N_t+1, and the total calculating complexity is $(N_t+1)K$, where K is user number.

In Step2, we need to compute $\|h_k\|^2$, $\|g_i\|^2$ and $\left(\frac{h_t g_i^H}{\|g_i\|^2}\right)g_i$. The complexity of computing $\|h_k\|^2$ is N_t+1, and the complexity of computing $\|g_i\|^2$ is N_t, which needs to be computed in each channel regions (N times); The complexity of $\left(\frac{h_t g_i^H}{\|g_i\|^2}\right)g_i$ is $(N_t+1)+N_t=2N_t+1$, which needs to be computed $M(M-1)R/2$ times per group. Thus the total computational complexity is

$$C_{FFT}=(M+1)K+MR+(2M+3)M(M-1)R/2. \tag{16}$$

From (15), and (16), we can see that the computational complexity of SUS and our proposed algorithm approximate to linear function of user number K by following result

$$\lim_{K\to\infty}\frac{C_{SUS}}{C_{FFT}}=\frac{2N_t+3}{N_t+1}, \tag{17}$$

When K is large enough, the complexity of the proposed algorithm will be sure lower than that of SUS. For example, when $N_t=4$, $\alpha=0.3$, $T\approx 1.3K$, and codebook size of 24, and the semi-orthogonal sets is $R=78$. Then we get

$$C_{SUS}-C_{FFT}\approx(11K+27\times 0.3K+63)-(5K+4.24+66.78)=14.1K-5181. \tag{18}$$

When $K>368$, the complexity of vector quantization based algorithm is lower than that of SUS algorithm.

5. Simulation Result and Discussion

In this section, we provide some numerical examples to illustrate the performance of the proposed user scheduling algorithm. In the considered multiuser MIMO downlink systems, the number of transmit antenna is $N_t=4$, and each user has single antenna. The power in simulation be $P=10dB$.

We assume that the discrete-time channel impulse response is generated according to the Hiperlan2 Channel Model C in [10]. The channels between different transmit and receive antennas are assumed to be independently.

Experiment *1*: This experiment is about the capacity of the presented user scheduling algorithm, DPC, TDMA and semi-orthogonal user scheduling (SUS) algorithm presented in [2]. For SUS algorithm, we choose optimal α range from 0.25 to 0.36. Fig. 1 shows that the presented algorithm, DPC and SUS can achieve higher capacity than that of TDMA. We can also find that the codebook of size 24 with 78 semi-orthogonal channel region sets and the codebook of size 32 with 144 semi-orthogonal channel region sets can achieve a moderate higher capacity than SUS. This implies that the presented scheduling algorithm will get higher capacity while has much lower complexity than SUS which will also be shown in the next experiment.

Experiment 2: This experiment is about the complexity of the the presented scheduling algorithm and that of SUS algorithm. In this experiment, $N_t=4$, $\alpha=0.3$, $T\approx 1.3k$, the codebook size is 24, and the number of semi-orthogonal channel region sets $R=78$. From Fig. 2, we can see that the proposed algorithm is with much lower complexity than that of SUS when user number is greater than 400. This implies that the complexity of our algorithm will be lower than that of SUS algorithm when the system is with large number.

Experiment 3: This experiment is about the complexity of the the presented scheduling algorithm and that of SUS algorithm. By analyzing SUS algorithm proposed in [2]. In this experiment, let $N_t=4$, $\alpha=0.3$, the FFT codebook size are 128 and 256 respectively, and the codebook used in SUS is of size 256. From Fig. 3, we can see that the proposed algorithm is with much lower complexity than that of SUS. This implies that the codebook based user schduling complexity is not only with much lower complexity, but also only need less amount of feedback bits than SUS algorithm

6. Conclusion

In this paper, we present a low complexity user scheduling algorithm with channel vector quantization in MIMO broadcast systems . The objective of the user scheduling is to reduce the computing complexity of user selection and achieve sum-rate optimization. The proposed user scheduling algorithm is with very low complexity in user scheduling than previous works. We also show that the sum-capacity will increase with the increment of the number of user.

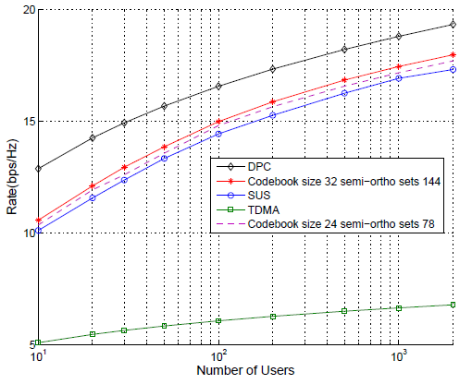

Fig.1: The throughput of different transmit strategy

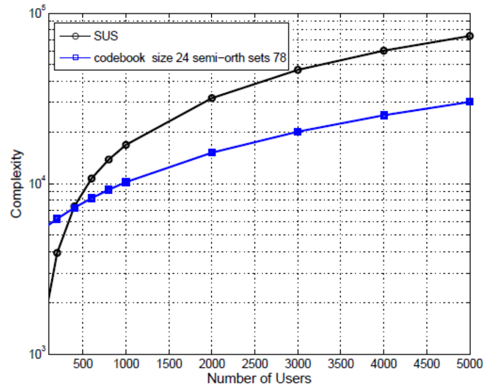

Fig.2: The comparison of complexity between quantization and SUS

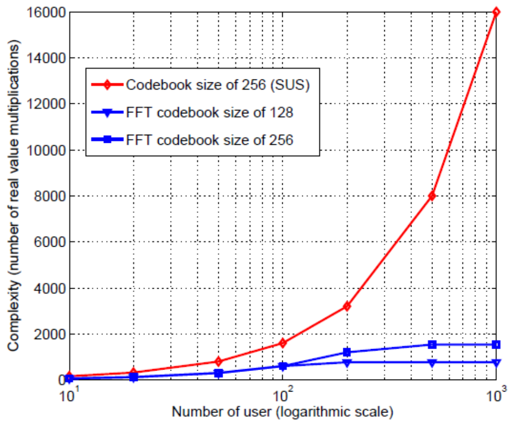

Fig.3: The complexity of codebook based user scheduling algorithm and SUS.

7. Acknowledgment

This work is supported by Shanghai Second Polytechnic University foundation QD209013.

8. References

[1] S. M. Alamouti, “A simple diversity techniques for wireless communications,” IEEE J. Select. Areas commun., vol. 16, pp.1451-1458, Oct. 1998.

[2] T. Yoo, A. Goldsmith, “On the optimality of multi-antenna broadcast scheduling using zero-forcing beamforming,” IEEE J. Sel. Areas Commun., vol. 24, no. 3, pp. 528-541. May. 2006.

[3] T. Yoo, A. Goldsmith, “Sum-rate optimal multi-antenna downlink beamforming strategy based on clique search,”in Proc. IEEE Globecom, Nov. 2005.

[4] M. Andrews, K. Kumaran, K. Ramannan, A. Stolyar, and P. Whiting, “Prociding quality of service over a shared wireless link,” IEEE Communications Magazine, vol. 39, no.2 pp.150-154, Feb. 2001.

[5] P. Viswanath, D. Tse, and R. Laroia, “Opportunistic beamforming using dumb antennas,” IEEE trans. Inf.Theory, vol. 48, no. 6, pp. 1277-1294. May. 2002.

[6] G. Dimić, N. D. Sidiropoulos, “On the downlink beamforming with Greedy user selection: Performance analysis and a simple new algorithm,” IEEE trans. Signal Processing, vol. 53, no. 10, pp. 3857-3868. Oct. 2005.

[7] P. Xia, S. Zhou, and G. Giannakis “Achieving the welch bound with difference sets,”IEEE trans. Inf. Theory. vol. 51, no. 5, pp. 1900-1907. May. 2005.

[8] C. Swannack, G. W. Wornell, and E. Uysal-Biyikoglu “MIMO broadcast scheduling with quantized channel state information” in Proc. IEEE international symposim on information theory,, Seattle, washington, Jul. 2006.

[9] G. Caire, S. Shamai, “On the achievable throughput of a multiantenna Gaussian broadcast channel,” IEEE trans. Inf. Theory., vol. 49, no. 7, pp. 1691-1706. Jul. 2003.

[10] J. Medbo and P. Schramm, “Channel models for HIPERLAN/2 in different indoors scenarios,” in ETSI/BRAN 3ERI085B, Mar. 1998.