OUS: Optimal User Selection in MU-MIMO WLAN

Yizirui Zhou, Anfu Zhou, and Min Liu
Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
Email: {zhouyizirui, zhouanfu, liumin}@ict.ac.cn

***Abstract*—In multiuser MIMO (MU-MIMO) networks, different users can transmit packets to the AP concurrently, thereby network capacity could be significantly improved. Previous works, using the conventional user selection schemes, have shown the promise of MU-MIMO in practical WLANs. However, we find that, to harness the full potential of MU-MIMO, an optimal user selection scheme is required. In this paper, we formulate the user selection problem in MU-MIMO as a discrete constrained optimization problem, and by decomposing and solving this problem, we design an optimal selection scheme called OUS. OUS efficiently allocates network resources so as to improve both the overall throughput of a network and fairness among users. We conduct extensive simulations to evaluate performance of OUS, and the simulation results show that compared with previous works, OUS improves the overall throughput by 40% and improves the fairness by 30% in most cases.**

I. INTRODUCTION

With the rapid development of mobile Internet, wireless network traffic demands grow exponentially [1]. As a promising technology to improve network capacity, multiuser MIMO (MU-MIMO) has been included in the latest IEEE 802.11ac standard [2]. In MU-MIMO based wireless networks, multiple users communicate concurrently with a multiple-antennas Access Point (AP). In this way, the theoretical capacity of the network increases linearly with the number of antennas on the AP [3].

Recently, many research efforts have been devoted to study MU-MIMO in practical wireless LANs (WLANs). Representative works include SAM in [4], 802.11n+ in [5] and TurboRate in [6]. SAM proposes the chain-decoding method to iteratively decode concurrent frames from uncoordinated devices. 802.11n+ modifies and improves the performance of 802.11n networks, it enables concurrent transmissions between multiple APs and users through nulling techniques. In particular, TurboRate focuses on rate adaptation in MU-MIMO WLAN.

These works have shown the promise of MU-MIMO. However, they simply inherit the traditional CSMA based random access mechanism widely adopted in non-MIMO WLANs [7], and cannot choose an optimal set of concurrent users to exploit the full potential of MU-MIMO. Although random access mechanism works effectively in conventional non-MIMO WLANs [8]–[10], the situation in MU-MIMO WLAN is quite different. Note that in non-MIMO WLANs, the throughput of a user totally depends on the SNR between itself and the AP. However, in MU-MIMO WLAN, the throughput of a user also depends on its channel correlation with other concurrent users [3], [6]. Therefore, user selection in MU-MIMO WLAN should consider channel correlation among users. Based on this insight, MaxT and MaxA [11], [12] take channel correlation into consideration, and select users which produce the maximal throughput. We find that, these mechanisms do help to adapt to the MU-MIMO situation and boost the throughput, but they will result in severe unfairness among users, as we will show in the following sections. Therefore, it is crucial to choose an appropriate set of concurrent users for improving both throughput and fairness.

Motivated by this problem, in this paper we propose an Optimal Uses Selection scheme called OUS. OUS takes both throughput and fairness requirement into consideration, and formulates the complex scheduling problem as a discrete constrained optimization problem. Then it uses the constraints to embody different choices of concurrent users, and sets the objective of the optimization to be the network utility, aiming to achieve a balance between the overall network throughput and fairness. Note that solving a discrete constrained optimization problem often incurs large computational complexity. To remedy this, we decompose the problem and propose an approximating greedy algorithm. The greedy algorithm runs in much less computational time and results in a solution very close to the optimal bound. It is shown that OUS is able to choose an appropriate set of concurrent users and properly allocate the transmission resources so as to achieve high performance.

We implement OUS in a simulation testbed based on ITPP [13], and run extensive simulations to demonstrate performance of OUS. Simulation results show that compared with the random access and other channel-aware user selection schemes, OUS improves the throughput and fairness by about 40% and 30% in average, respectively. Furthermore, results also indicate that OUS is robust as network scale increases.

The rest of the paper is organized as follows. Section 2 describes the background of MU-MIMO WLAN and the motivation to design OUS. Section 3 details the design of OUS. Section 4 presents the evaluation of OUS. We conclude and discuss future work in Section 5.

II. BACKGROUND AND MOTIVATION

In this paper, we mainly focus on uplink transmission as in previous works [6], [14], although the proposed idea is also applied to downlink transmission. Note that uplink transmission requires more considerations since each user independently joins in concurrent transmissions without knowing others. We first briefly describe the background and unique property of uplink MU-MIMO. Then we show the impact of channel correlation on network performance using an example, which leads to the motivation of designing OUS.

A. Uplink MU-MIMO Background

In a typical MU-MIMO WLAN where the AP has two antennas, uplink concurrent transmissions can be represented by Eq. 1.

$$\begin{cases} y_1 = h_{11}x_1 + h_{12}x_2 + n_1 \\ y_2 = h_{21}x_1 + h_{22}x_2 + n_2 \end{cases} \tag{1}$$

where y_i denotes the composite signals received by antenna i, symbols transmitted by user i is x_i, h_{ij} is the channel attenuation coefficients between antenna i of the AP and the user j, and n_i represents the Gaussian noise.

To recover the concurrent transmissions, *zero-forcing*(ZF), which is a technique to mitigate interferences, is widely adopted. By multiplying the inverse matrix of channel coefficients matrix on both sides of the equation, ZF algorithm can decode the concurrent streams successively. However, research has shown that users in concurrent transmissions suffer SNR reduction [6], as Eq. 2 indicates.

$$SNR_{con} = SNR_{ori} + 10log_{10}(r) \tag{2}$$

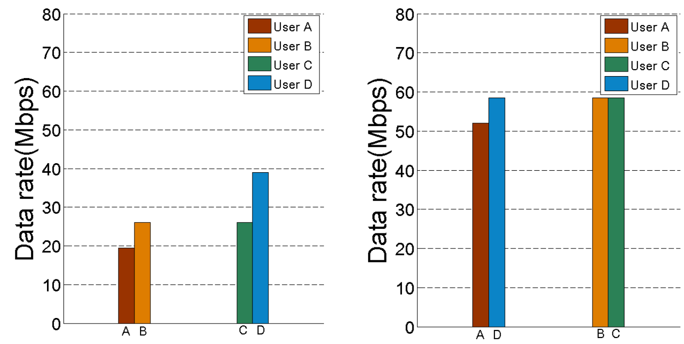
where SNR_{con} and SNR_{ori} represent the SNR(in dB) of the user in concurrent transmissions and the SNR(in dB) when the user transmits alone, respectively. From Eq. 2, the exact value of SNR reduction depends on r $(0 < r < 1)$, which is the channel correlation coefficient that denoted by the square of cosine value of the angle θ between concurrent users' channel vectors (h_{11}, h_{21}) and (h_{12}, h_{22}).

$$r = cos^2(\theta) \tag{3}$$

When the channel of concurrent users is highly correlated, r will be smaller. From Eq. 2, the SNR reduction will be larger, and hence the network performance is degraded.

B. Motivation

As the discussion above indicates, in concurrent transmissions, highly correlated users experience significant SNR reduction, and hence result in low user throughput. This implies that such users should be properly scheduled apart, in order to avoid performance degradation. To illustrate the impact of improper user selection, consider a MU-MIMO WLAN consisting of a two-antenna AP, and four users A, B, C and D. Suppose the original data rates of the four users when they transmit alone are D_{ori}=$\{52Mbps, 58.5Mbps, 58.5Mbps, 58.5Mbps\}$, and the channel correlation between user AB, CD, AD and BC are $r = \{0.06, 0.1, 0.94, 0.97\}$. Fig. 1 depicts the comparison of improper and proper selection in terms of user throughput. As seen from Fig. 1a, with improper selection, either user pair of AB or user pair of CD are scheduled in any concurrent transmissions. Due to the high channel correlation, they suffer a SNR reduction of 12dB and 10dB



(a) Improper user selection

(b) Proper user selection

Fig. 1. Data rate comparison of proper and improper scheduling

respectively, which means these users have to lower their data rates to $D_{con} = \{19.5Mbps, 26Mbps, 26Mbps, 39Mbps\}$ and hence results in a much lower overall throughput of the network. Moreover, since the SNR reduction users suffer varies dramatically, throughput distribution is highly biased, which leads to the unfairness among users. In contrast, Fig. 1b shows an example of proper selection, which schedules the pair of AD or the pair of BC in a concurrent transmission. Thanks to the channel orthogonality, all users have much larger throughputs. In addition, since data rates are not degraded, throughputs distribute evenly among users. To sum up, in the same channel condition, proper user selection achieves a much higher throughput and fairness than improper user selection. Therefore, it is critical to select a proper subset of users with low correlation in each concurrent transmission in order to enhance network performance.

III. OUS Scheme

Now we propose the design of OUS, which is a user selection scheme for MU-MIMO WLANs. Different from random user selection, OUS chooses proper set of concurrent users with low channel correlation, so as to achieve better network performance. Next, we first demonstrate the overview architecture of OUS, then we present the key component of user selection algorithm in OUS.

A. Overview

A typical usage scenario of OUS is a WLAN consisting of one AP and several users. In the network, the AP acts like a central controller which coordinates the transmissions of all the users. Users join in the network and communicate under the control of the AP[1]. Since the AP has a global view of the network, it is able to make wise selection decisions based on current channel condition. Considering the characteristics of dynamic scenarios and the fluctuations of channel quality, the whole scheduling process is divided into periods of fixed length. In each scheduling period, OUS conducts a series of operations to complete the whole scheduling, which are illustrated in Fig. 2. In particular, in each period OUS operates

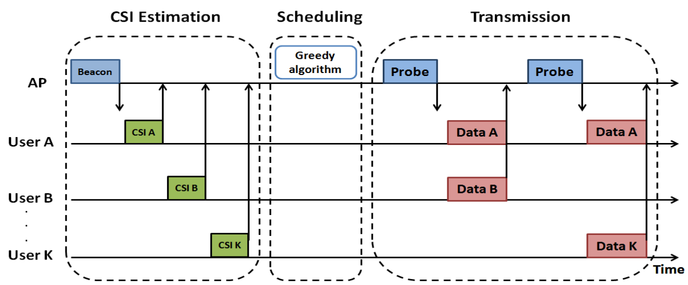


Fig. 2. Flow operations of OUS in a specific scheduling period

as follows:

(i) **CSI Estimation:** First, the AP announces its intention for uplink transmission through a Beacon packet. This packet is broadcasted to all of the users. After receiving the packet, users respond the AP with a packet in order. The AP hence obtains users' CSI through estimating the preambles in these packets.

(ii) **Scheduling:** After getting each users' CSI, the AP calculates the corresponding SNR and channel correlations r between them. Afterwards, it takes into account both the SNR and r, runs the selection algorithm and finally decides the set of concurrent users in each schedule time slot. The selection algorithm, as the key component of OUS, will be presented in detail in next two subsections.

(iii) **Transmission:** After deciding the set of concurrent users, the AP informs these users to transmit. This is accomplished by a probe packet broadcasted to all of the users. The probe packet includes the information of which and in what data rate should a user transmit a packet. After receiving probe packet, the chosen users transmit concurrently. At last, the AP decodes concurrent streams repeatedly and completes the transmissions.

B. Problem Formulation of User Selection

In OUS, time is divided into scheduling periods, where each scheduling period is divided into T time slots. At the beginning of each scheduling period, the AP takes the channel conditions and the available time slots as input and outputs the number of time slots allocated to users. We assume that channel conditions do not change during a scheduling period[2]. In the following, we formulate this allocation problem as a discrete optimization problem. Then we propose an approximation algorithm GreedyMax to solve this problem.

Considering a MU-MIMO WLAN, let M be the number of antennas on the AP and N be the number of users involved. Thus, in one time slot, there can be up to M users transmit concurrently. These users form a transmission set called S. For instance, a two-antenna AP and 3 users form a transmission set S = {A,B,C,AB,AC,BC}. Each element in the set indicates a possible group of concurrent users in one transmission, where 'A' means user A transmit alone and 'AB' means user A and user B transmit concurrently.

The AP thus chooses one element from the set S and allocate it to one of the available time slots. We define that in each scheduling period, the allocation results form a list R, it denotes the number of slots allocated to each element in S and we define its length as R_n. For example, $R[k]$ represents the number of slots allocated to k_{th} element in S. We use $u_i(R)$ to represent the utility function of user i with R as the function input. Consequently, the user selection problem can be described as below.

$$\textbf{Problem}: \quad \max_{R} \sum_{i=1}^{N} u_i(R) \tag{4}$$

$$\text{subject to} \quad \sum_{k=1}^{R_n} R[k] \leq T \tag{5}$$

$$R[k] \geq 0, k \in [1, R_n] \quad \forall k \tag{6}$$

The objective of the above problem is to maximize the total utility of the network, which is described by the summation of each users' utility $u_i(R)$. The first constraint indicates that the total number of time slots allocated to users does not exceed the number of available time slots T. The second constraint shows that $R[k]$ should be nonnegative discrete numbers.

Recall that our ultimate goal is to maximize the throughput of the network on the basis of ensuring fairness among users. However, it is noteworthy that throughput and fairness are potentially conflicting metrics. Maximizing total throughput implies weak users miss the opportunities to be selected, while ensuring high fairness enforces strong users to lower their data rates to transmit concurrently with weak ones. To find an optimal trade-off between throughput and fairness, we define the utility function $u_i(R)$ as follows:

$$u_i(R) = log(\sum_{k=1}^{R_n} p_{ik} R[k]) \tag{7}$$

where p_{ik} represents the unit gain of user i in k_{th} user group, it reflects the throughput contribution a user made if k_{th} user group is allocated one time slot. We further define p_{ik} as follows:

$$p_{ik} = \begin{cases} map(SNR(i)) & \text{if} \quad i \in S(k) \\ 0 & \text{if} \quad i \notin S(k) \end{cases} \tag{8}$$

where $map(SNR(i))$ represents the expected data rate of user i in k_{th} group, we map the expected SNR to the data rate level using existing techniques[4]. For instance, say there is a transmission group 'AB'. In this group, based on the r that calculated before, we find that user A's SNR reduces to 14dB when transmitting concurrently with user B. This SNR is mapped to a specific data rate, and we get the unit gain of user A in group 'AB' is 26Mbps. By calculating the summation of all the users' gain in each pair, we obtain the total expected

[1]Such centralized architecture is widely used in enterprise WLANs, see [15], [16].

[2]Note that the wireless channels usually have a coherent time during which the channel condition won't change very much, especially when the wireless nodes is static or in low-speed movement [3], [17].

Algorithm 1 OUS GreedyMax

Input:

i: User index, j: User group index, t: Slot index, N: User number, p_{ik}: Unit gain, T: Total time slots, R_n: User group number.

Output:

Allocated list R with slot numbers.

for k from 1 to R_n **do**
  $R[k] = 0$
end for
for t from 1 to T **do**
  for j from 1 to R_n **do**
    Try to assign t_{th} slot to j_{th} user group
    if $(\sum_{i=1}^{N} log(\sum_{k=1}^{R_n} p_{ik}R[k])) > max$ **then**
      $max = (\sum_{i=1}^{N} log(\sum_{k=1}^{R_n} p_{ik}R[k]))$
    end if
  end for
  Assign t_{th} slot to the user group which achieves max
end for
return R;

throughput a user achieves in a scheduling period. Finally, a log function is employed to evaluate all users' utility gain. Note that the log function is widely adopted as a utility function for allocation of resources, since it can express a suitable tradeoff between throughput and fairness [18].

C. Approximation Algorithm

In general, solving such a problem incurs high computational complexity. To see this, consider a common scenario with N users. Suppose the AP is equipped with M antennas, then we get the result set size $R_n = \sum_{j=1}^{M} \mathrm{C}_N^j$. In a scheduling period with T time slots, the number of computations grows as high as $(\sum_{j=1}^{M} \mathrm{C}_N^j)^T$. As a result, discrete optimization yields a complexity of $\mathcal{O}(N^{TM})$, exponential to N, which is unacceptable. Given this situation, we next propose a greedy algorithm called GreedyMax, which runs in polynomial time and gives a near-optimal solution.

In a network with N users, given the length of a scheduling period T, Algorithm 1 demonstrates how GreedyMax works. First, GreedyMax initializes all the $R[k]$ to zero, this implies that no time slot has been assigned to users yet. Then, at the beginning of each time slot, GreedyMax tries to assign this slot to each possible user group, and calculates the corresponding utility gain it contributes to the whole network. Afterwards, GreedyMax sorts the calculation results and schedules the user group which achieves the highest utility gain. Finally, GreedyMax stores the results and proceeds to assign the next slot until all of the time slots are consumed. The rational behind GreedyMax is to decompose the global optimization problem into a multi-step optimization problem, so that it can maximize the objective function in each step and approach the optimal solution asymptotically.

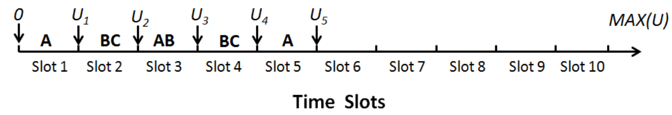


Fig. 3. GreedyMax algorithm

Intuitively, Fig. 3 shows how GreedyMax works in a network with three users(A, B, C) and a two-antenna AP. Say each scheduling period is divided into 10 time slots, GreedyMax starts from the first time slot. As seen from the figure, the highest utility gain in slot 1 is achieved by user group 'A', hence GreedyMax directly assign this slot to 'A'. While in slot 2, as user A has already been selected, assigning this slot to user A again will yield a diminishing utility return. Therefore, GreedyMax turns to assign the second time slot to 'BC'. Finally, GreedyMax repeats this process until all the 10 time slots are assigned.

The GreedyMax algorithm has a complexity of $\mathcal{O}(TN^M)$, polynomial to the user number N. Hence, in a general network, time consumption in scheduling is negligible.

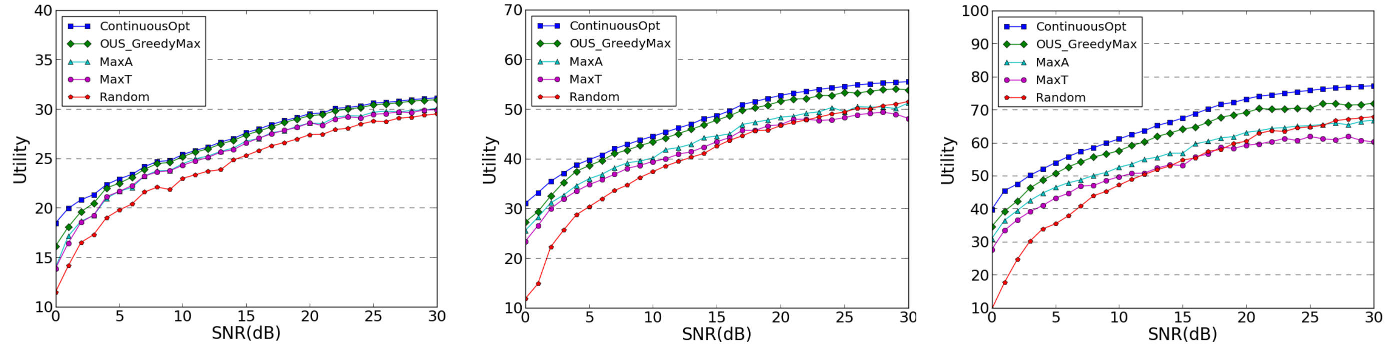
To compare GreedyMax with optimal solution, we express the discrete optimization problem as a continuous optimization problem. The only difference between them is that in continuous optimization, $R[k]$ is continuous real numbers. It is known that continuous optimization always achieves higher objective function value than discrete optimization problem, so it can be used to indicate the upper bound. In the evaluation section, we name it ContinuousOpt and employ the non-linear optimization library OPT++ [19] to solve it.

IV. EVALUATION

In this section, we evaluate performance of OUS extensively using simulations. We first introduce the settings of simulations, and then we compare OUS with other representative scheduling schemes: 1) max-angle (MaxA) [12], which selects the users that share the maximal angle with current user, 2) max-throughput (MaxT) [11], which always selects the users that achieve the maximal throughput with current transmission, 3) Random, which is the scheme widely used in current MU-MIMO works [4]–[6], and 4) ContinuousOpt, which indicates the optimal results.

A. Simulation Environment

The simulations are performed with a simulator written in C++, which is capable of simulating a WLAN with one AP and up to dozens of users. In order to simulate a real MIMO WLAN, we set up the AP in this simulator with two antennas and other users with only one antenna. In the underlayer design, we implemented basic decoding algorithms, including MRC (for single transmission) and ZF (for concurrent transmission) algorithm. In addition, based on ITPP



(a) N = 5 users (b) N = 10 users (c) N = 15 users

Fig. 4. Utility comparison of selection algorithms by varying SNR with client number N = 5, 10, 15. Plots show the average of 1000 runs

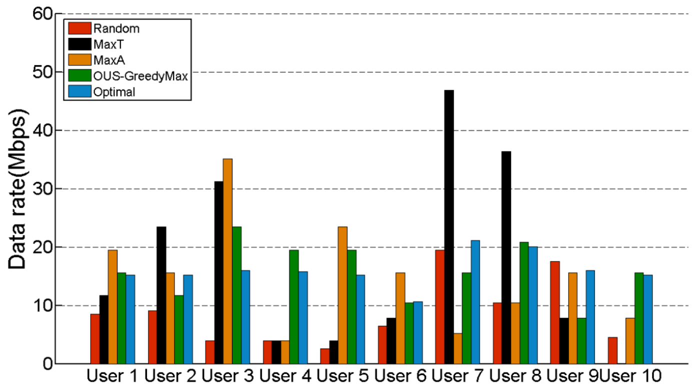


Fig. 5. Data rate comparison of 10 users, with SNR = 20dB.

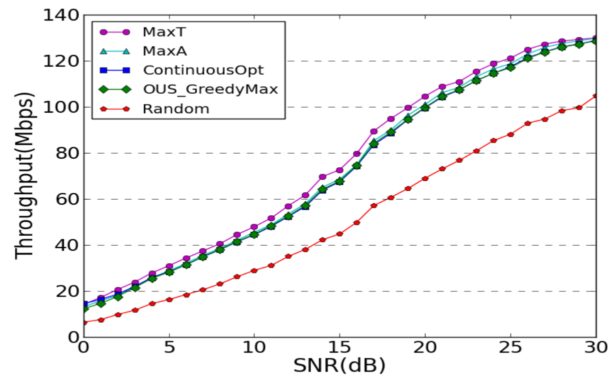


Fig. 6. Network throughput comparison by varying SNR, N = 10 users. Plot show the average of 1000 runs

library [13], this simulator implements {BPSK, QPSK, QAM-16, QAM-64} modulation as well as LDPC channel coding with sophisticated code rates. Thus, it is able to operate with 802.11 standard data rates varying from 6.5Mbps to 65Mbps. We apply TurboRate [6], a MU-MIMO rate adaptation scheme for the user to select the best bitrate. At higher layers of the stack, a MAC protocol is implemented to coordinate the transmissions between the AP and users.

Parameters in simulation include the number of users N, scheduling period length T and average SNR. We performed as long as 20000 time slots in all of the simulations, and in all cases, we set the length of each scheduling period $T = 20$ time slots to ensure conformity (other values of T leads to similar simulation results, so we omit them here).

B. Simulation and analysis

We first make comparisons of the algorithms in terms of network utility by varying SNR and user number N. Fig. 4 depicts the simulation results. As seen from the figure, GreedyMax performs significantly betters in terms of network utility. It achieves a utility very close to ContinuousOpt and much higher than MaxT, MaxA and Random. Since ContinuousOpt gives the up bound of network utility, we can conclude that GreedyMax is very close the optimal results.

We also observe that the utility difference between GreedyMax and other schemes becomes more visible when N increases. Specifically, when N grows from 5 to 15, GreedyMax achieves an average utility 17%, 24% and 30% higher than other algorithms respectively. The reason is that when user number grows, channel correlation among users becomes more complex. Consequently, Random fails to schedule better users, MaxA and MaxT ignore weaker ones, which all lead to low utility. In contrast, GreedyMax maximizes network utility in each transmission, which is capable to ensure high performance in a long term. Next, we break down the metric of network utility, and compare throughput and fairness metrics, respectively.

1) Throughput: We set up a scenario with 10 users and an average SNR of 20dB. Fig. 5 shows the user throughput in a common scheduling period. As shown in the figure, Random fails in both throughput and fairness. While for MaxT and MaxA, despite some users enjoy high performance(User 7 and 8), the throughput distribution among users is highly unequal(User 10 achieves freezing throughput). In contrast, OUS succeeds to ensure both high and balanced throughput around 20Mbps among all users, extremely close to ContinuousOpt's results.

Fig. 6 plots the comparison of network throughput for different algorithms by varying SNR. Network throughput

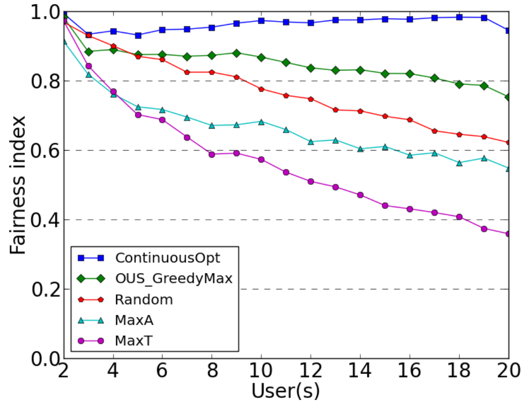


Fig. 7. Fairness comparison by varying user number, SNR = 20dB.

defined here is the average of instantaneous throughput in each scheduling period. As seen from the figure, the throughput of GreedyMax is 40% higher than Random, while just 8% lower than MaxT and MaxA. This indicates that GreedyMax is able to achieve a global balance between users with a little sacrifice on throughput.

2) Fairness: Now we compare the performance of different algorithms in terms of fairness index by varying user number N. We use the Jains fairness index, which is commonly adopted in the literature [20], as the measure of fairness for comparison. A higher value of Jain's fairness means a fairer distribution of the throughputs.

Fig. 7 depicts the fairness index comparison of scheduling algorithms by varying user number. As the figure shows, all algorithms achieve fairness index above 0.9 when the number of users N is very small. While as N grows, fairness achieved by MaxT, MaxA and Random all drop dramatically. Since MaxT and MaxA always select the better users while ignore weak ones, resource allocation among users will certainly be biased, hence fairness index decreases. In contrast, GreedyMax is more robust to the growth of user number, and can still efficiently allocate network resources and keep the fairness index at a high level. Overall, the improvement ratio of GreedyMax over Random, MaxA and MaxT is 15%, 32% and 55%, respectively.

V. CONCLUSION

In this paper we present OUS, an uplink schedule scheme for MU-MIMO LANs. By taking into account channel correlation, OUS enables the AP, to allocate concurrent transmission opportunity to users in a more efficient and fair way. we use simulation to demonstrate that OUS can significantly increase both the throughput and fairness in MU-MIMO WLANs. In this paper, we focus on uplink schedule. In our future work, we plan to extend user scheduling to downlink scenario. It is believed that user scheduling in downlink MU-MIMO WLAN will be more challenging, since power allocation should also be taken into consideration besides channel correlation.

ACKNOWLEDGEMENT

This work has been supported by China Scholarship Council, the National Natural Science Foundation of China (No.61132001, No. 61120106008, No.61272474 and No. 61202410).

REFERENCES

[1] Cisco, "Cisco visual networking index: Global mobile data traffic forecast update, 2013-2018," Tech. Rep., 2014.

[2] E. Perahia and R. Stacey, *Next Generation Wireless LANs: 802.11 n and 802.11 ac*. Cambridge university press, 2013.

[3] D. Tse and P. Vishwanath, *Fundamentals of Wireless Communications*. Plenum Press, New York and London, 2005.

[4] K. Tan, H. Liu, J. Fang, W. Wang, J. Zhang, M. Chen, and G. M. Voelker, "SAM: enabling practical spatial multiple access in wireless lan," in *Proceedings of the 15th annual international conference on Mobile computing and networking*. ACM, 2009, pp. 49–60.

[5] K. C.-J. Lin, S. Gollakota, and D. Katabi, "Random access heterogeneous mimo networks," in *ACM SIGCOMM Computer Communication Review*, vol. 41, no. 4. ACM, 2011, pp. 146–157.

[6] W.-L. Shen, Y.-C. Tung, K.-C. Lee, K. C.-J. Lin, S. Gollakota, D. Katabi, and M.-S. Chen, "Rate adaptation for 802.11 multiuser mimo networks," in *Proceedings of the 18th annual international conference on Mobile computing and networking*. ACM, 2012, pp. 29–40.

[7] I. C. Society, "802.11g: IEEE standard for information technology, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," Tech. Rep., 2003.

[8] G. Bianchi, "Performance analysis of the ieee 802.11 distributed coordination function," *IEEE Journal on Selected Areas in Communication*, vol. 18, no. 3, pp. 535–547, 2006.

[9] M. Heusse, F. Rousseau, R. Guillier, and A. Duda, "Idle sense: an optimal access method for high throughput and fairness in rate diverse wireless lans," in *Proceedings of the 2005 conference on Applications, technologies, architectures, and protocols for computer communications*, vol. 35, 2005, pp. 121–132.

[10] L. Jiang and J. Walrand, "A distributed csma algorithm for throughput and utility maximization in wireless networks," *IEEE/ACM Transactions on Networking (TON)*, vol. 18, no. 3, pp. 960–972, 2010.

[11] Z. Shen, R. Chen, J. G. Andrews, R. W. Heath, and B. L. Evans, "Low complexity user selection algorithms for multiuser mimo systems with block diagonalization," *Signal Processing, IEEE Transactions on*, vol. 54, no. 9, pp. 3658–3663, 2006.

[12] C. Wang and R. D. Murch, "Adaptive downlink multi-user mimo wireless systems for correlated channels with imperfect csi," *Wireless Communications, IEEE Transactions on*, vol. 5, no. 9, pp. 2435–2446, 2006.

[13] "ITPP," http://itpp.sourceforge.net/4.3.1/.

[14] M. Ghaderi and M. Mollanoori, "Uplink scheduling in wireless networks with successive interference cancellation," *IEEE Transactions on Mobile Computing*, p. 1, 2013.

[15] V. Shrivastava, N. Ahmed, S. Rayanchu, S. Banerjee, S. Keshav, K. Papagiannaki, and A. Mishra, "CENTAUR: realizing the full potential of centralized wlans through a hybrid data path," in *Proceedings of the 15th annual international conference on Mobile computing and networking*. ACM, 2009, pp. 297–308.

[16] A. Zhou, M. Liu, Z. Li, and E. Dutkiewicz, "Modeling and optimization of medium access in csma wireless networks with topology asymmetry," *Mobile Computing, IEEE Transactions on*, vol. 11, no. 9, pp. 1559–1571, 2012.

[17] X. Xie, X. Zhang, and K. Sundaresan, "Adaptive feedback compression for mimo networks," in *Proceedings of the 19th annual international conference on Mobile computing & networking*. ACM, 2013, pp. 477–488.

[18] F. Kelly, "Charging and rate control for elastic traffic," *European transactions on Telecommunications*, vol. 8, no. 1, pp. 33–37, 1997.

[19] "OPT++," https://software.sandia.gov/opt++/.

[20] R. Jain, D.-M. Chiu, and W. R. Hawe, *A quantitative measure of fairness and discrimination for resource allocation in shared computer system*. Eastern Research Laboratory, Digital Equipment Corporation, 1984.