

DiFuse: distributed frequency domain user selection for multi-user MIMO networks

Kyu-haeng Lee¹ · Joon Yoo² · Chong-kwon Kim³

© Springer Science+Business Media New York 2016

Abstract Optimal user selection is important in increasing the capacity of Multi-User Multi-Input Multi-Output Wi-Fi networks, yet it faces a significant challenge; the multi-user diversity gain can be overwhelmed by the formidable Channel State Information (CSI) acquisition overhead. To lessen the overhead, existing schemes adopt the greedy user selection which generally takes the projected norm as the user selection metric, since it considers both the channel power gain and the orthogonality. However, the projected norm suffers from occasional poor user selection, since it does not take the optimal sum capacity gain into account. This paper proposes a new distributed user selection protocol called DiFuse. To employ the sum capacity gain as the user selection metric in DiFuse, each user cleverly computes its own estimated capacity gain by overhearing the CSI feedback from others. The users then simultaneously transmit their feedbacks at the frequency domain via the distributed feedback contention, which effectively reduces the feedback overhead. Then the AP collectively utilizes them for user selection that achieves the maximum positive increment to the sum capacity gain. We implemented the prototype of DiFuse on the USRP

N210, and evaluated its performance via both testbed experiments and trace-driven emulations. The results showed that DiFuse outperforms the throughput of the existing scheme called OPUS by $1.8\times$ on average, while maintaining better fairness.

Keywords MU-MIMO · User selection · Scalability · CSI feedback overhead

1 Introduction

Multi-User Multi-Input Multi-Output (MU-MIMO) technologies have emerged as a key component to increase the capacity of wireless networks. The MU-MIMO scheme either simultaneously transmits to multiple users at the same time or focuses energy towards a single user to enhance the data rate. Cellular technologies such as LTE systems adopted the MU-MIMO technology, and due to its advantages, MU-MIMO has recently been incorporated to the 802.11ac and 802.11ax Wi-Fi standard, as an optional feature [1–3]. The main reason that it is selected as an option is because multi-streaming to multiple users inherently requires large control overheads in the 802.11 based MAC.

Optimal user selection is essential for increasing the capacity of MU-MIMO Wi-Fi networks. However, determining an optimal user set is difficult and impractical since it requires an exhaustive search over all possible user and antenna sets, and its search space is $\sum_{m=1}^M \binom{K}{m}$, where K and M are the number of users and the number of Access Point (AP) antennas, respectively. Many researchers have developed greedy user selection algorithms aimed to provide sub-optimal performance while reducing the feedback

✉ Joon Yoo
joon.yoo@gachon.ac.kr

✉ Chong-kwon Kim
ckim@snu.ac.kr

Kyu-haeng Lee
khlee@popye.snu.ac.kr

¹ Samsung Electronics, Seoul, South Korea

² Department of Software Design and Management, Gachon University, Seongnam, South Korea

³ Department of CSE, Seoul National University, Seoul, South Korea

overhead as well as computational burden [4–9]. The main idea behind the most prior schemes is to *incrementally* select a user in each iteration by some selection criteria instead of conducting exhaustive search for all user and antenna set combinations. For example, one user is selected in each iteration such that the new user minimizes interference to previously selected streams.

To accomplish the benefit of the MU-MIMO user selection in the aforementioned algorithms, we need to exploit the multi-user diversity gain; at a given time, the AP can select the best user (e.g., a user with favorable channel conditions) among candidates to improve the system throughput. To leverage multi-user diversity, two key challenges should be addressed: reducing the *Channel State Information (CSI)* feedback overhead and employing the proper scheduling policy for user selection.

First, the downlink CSI of the candidate users must be efficiently fed back to the AP. Different from cellular systems [10] where separate control channels are used to report the CSI, current 802.11ac Wi-Fi systems use a series of poll-based CSI feedbacks for each user [1], as shown in Fig. 1. Furthermore, the CSI feedback is transmitted at the low basic rate (e.g., 6.5 Mbps), and also grows as the number of transmitter/receiver antennas, quantization level and subcarrier group size increase. The CSI feedback overhead can reach up to $25\times$ compared to the data transmission time in case of 160 MHz of bandwidth and 4×1 MIMO [11]. Such excessive overhead could easily overwhelm the multi-user diversity gain even under optimal user selection.

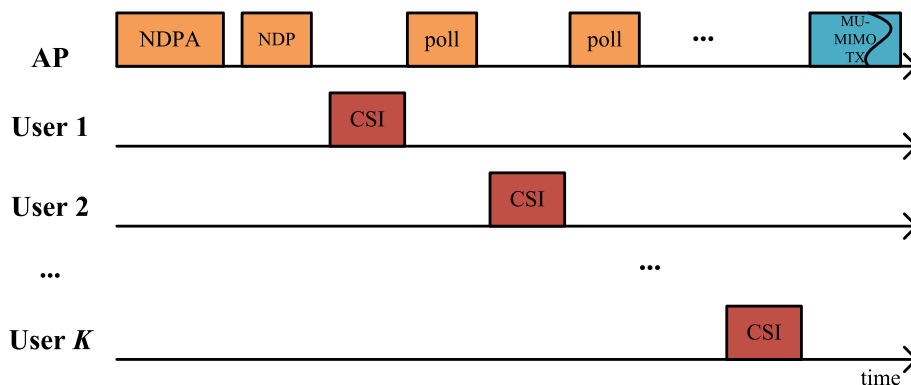
Second, it is vital to select the best user in every user selection step to leverage the MU-MIMO capability by employing the appropriate user selection metric. The *projected norm*, which is defined as the norm of the user channel projected to the orthogonal subspace of the previously selected user channels [6], is widely used, since it considers both the channel power gain and the orthogonality. However, in some cases, the projected norm based scheme may result in undesirable user selection, due to the

fact that it does not consider how the newly joined user channel impacts the already selected ones, if there are any. This may fail to maximize the sum-capacity in each iteration, and occasionally cannot guarantee a positive increment in the sum-capacity. To handle this issue, the AP that employs the norm-based scheme must additionally compute the sum-capacity to assure that it gives positive increment. Here, the feedback report may have induced unnecessary overhead, since the user may not be selected.

In this paper, we present a new user selection protocol called DiFuse (Distributed frequency domain user selection) that uses *the capacity gain* as a scheduling metric in user selection. The key mechanism of DiFuse is to greedily select a user at each iteration which yields maximum positive increment to the sum-capacity of the network. Given a user set, the capacity gain of a new user is defined as the increment in network capacity achieved by including the new user to the user set. DiFuse, as its name suggests, makes the scheduling decisions in a distributed manner. Each user cleverly computes its expected sum-capacity gain by overhearing the CSI feedback transmissions from other users. Then each user sends its sum-capacity gain in a simplified format called a *Selection REQUEST (SREQ)*; the user marks on one particular subcarrier of an OFDM symbol depending on the value of sum-capacity gain. The users concurrently transmit the SREQs via *distributed feedback contention*, which is devised to effectively reduce feedback contention via dynamic threshold design and frequency domain contention [12–14]. The AP receives the SREQs and collectively uses them to select the user that gives the maximum positive increment to the sum-capacity of the network, then polls the user for the actual CSI transmission.

We implement the DiFuse prototype on the USRP N210 and GNURadio [15], and conduct testbed experiments and trace-driven emulations. The results show that DiFuse obtains higher throughput compared to conventional schemes such as SUS [6] and OPUS [9], as well as 802.11ac [1]. Further, DiFuse gives a better degree of

Fig. 1 MU-MIMO transmission in 802.11ac. The AP randomly polls a set of users for MU-MIMO transmission



proportional fairness over other schemes, especially when users experience different channel qualities.

The remainder of the paper is organized as follows. Section 2 provides the related work and we give some background of MU-MIMO and user selection in Sect. 3. We discuss the DiFuse mechanism in Sect. 4. Section 5 shows the performance evaluation and we finally conclude our paper in Sect. 6.

2 Related work

In this section, we review the recent related work on the user selection schemes and channel feedback overhead reduction techniques.

2.1 User selection schemes

In user selection, it is challenging and often impractical to determine the optimal user set due to the large search space. Therefore, some protocols based on heuristics have been proposed. The basic idea of these proposals is to reduce the computation and feedback overhead by incrementally selecting a user in every selection round. ZFS [4] chooses a user that maximizes the sum-capacity. In GWC-ZFBF [5], the AP chooses a user with the largest channel power gain. SUS [6] finds a user with the largest norm of the projected channel to the orthogonal subspace of the previously selected users. Jin et al. [7] propose a volume metric as the product of the diagonal elements of an upper-triangular matrix by performing QR factorization to the selected user channels. The authors try to achieve less computational effort compared to SUS. GUSS [8] considers *delete* and *swap* operations to guarantee positive increment of channel capacity in each selecting round. The aforementioned proposals, however, are impractical in a sense that they simply assume that full CSI feedback from all users is just given prior to the user selection.

OPUS [9] is a user selection scheme for MU-MIMO Wi-Fi systems that bears the most similarity to our work. In OPUS, users estimate their potentials (e.g., SINR) in each round to boost the capacity and initiate a distributed feedback contention. The potential measurement scheme closely follows the main idea of the SUS (i.e., the largest projection power). However, the result based on the projected norm may not satisfy the maximum sum-capacity. Also, the time domain contention employed in OPUS may cause non-negligibly high overhead.

MIMOMate [16] and Signpost [17] are user grouping protocols for uplink MU-MIMO. The main idea of MIMOMate is to compute proper user transmission groups that the authors call “MIMO-Mate” and to let users join the concurrent transmission according to their MIMO-Mate

relation. To do this, a MIMOMate AP should configure MIMO-Mate relations and it needs to announce the result to its users. In Signpost, users use angles between their channel and some predetermined orthogonal directions to contend for the uplink transmission opportunity. In particular, the authors propose a contention scheme utilizing both frequency and time domain resources which they call 2-D contention. Though these schemes are designed for the uplink case, we additionally compare their throughput performance with DiFuse, as will be shown later.

2.2 CSI overhead reduction techniques

We should carefully consider the impact of CSI feedback overhead, or else, we may fail to realize the multi-user diversity gain in MU-MIMO scheduling. One way of reducing the CSI overhead is to use *compression* based on the codebook and quantization techniques [1, 10, 18], which reduces the number of bits in the CSI feedback frame. In Wi-Fi systems, the CSI feedback can be compressed in three dimensions: at the time, frequency, and quantization level. However, it is not easy to select the optimal compression level and also the compression may result in throughput loss since fewer CSI feedback may offer diminishing returns.

We can further reduce the overhead by decreasing the *number of feedbacks*. One way is to employ threshold based techniques using S(I)NR [19, 20] or CNR (Carrier-to-Noise Ratio) [21]. Otherwise, we can exploit the statistical model of channel coherence time for the similar purposes [11]. Although, these schemes reduce the number of feedbacks, the performance may be degraded due to the reduced CSI feedback. Furthermore, the instantaneous CSI must be updated as a mandatory feature in 802.11ac [1] *frame aggregation* to successfully decode all the frames.

3 Background and motivation

3.1 System model

In this paper, we consider a single Basic Service Set (BSS) Wi-Fi network, where an M -antenna AP and K single-antenna user stations communicate with each other, as shown in Fig. 2. Assume that the users experience independent Rayleigh fading. In particular, the channel gain from the m th antenna of the AP to the k th user, denoted by h_{km} , is assumed to be an independent zero mean complex Gaussian random variable with unitary variance. Then, we can characterize the downlink channel of user k , i.e., h_k , as a zero mean complex Gaussian channel vector.

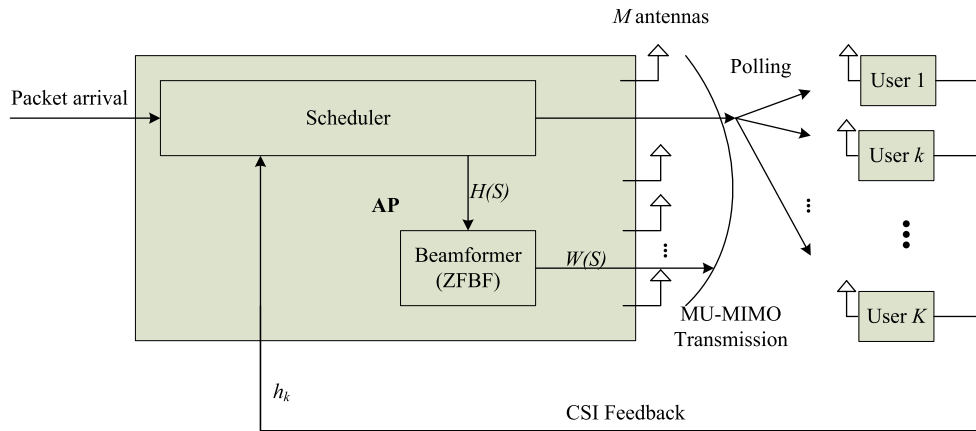


Fig. 2 MU-MIMO downlink system with an M -antenna AP and K single-antenna user stations

In MU-MIMO, the multi-antenna transmitter, e.g., AP, uses precoding to send multiple data streams between several users at a given instant. Similar to most recent schemes, we incorporate Zero-Forcing Beamforming (ZFBF) as the precoding strategy, since it effectively removes the mutual interference among concurrent transmissions by using a low-complexity precoding matrix computation. In ZFBF, the precoding vector of one user is selected to be orthogonal to the channel vector of the other user. In particular, the precoding matrix for the transmission group S , denoted by $W(S)$, is obtained as:

$$W(S) = H(S)^\dagger = H(S)^*(H(S)H(S)^*)^{-1}, \tag{1}$$

where $(\cdot)^\dagger$, $H(S)$, and H^* stand for a pseudo-inverse, the channel matrix of S , and the conjugate transpose of H .

Let $X(S)$ be the signal vector to be transmitted for S . Then, the precoded signal vector, denoted by $X(S)' = W(S)X(S)$, has an average power constraint of $\mathcal{E}[X(S)'*X(S)'] \leq P$, where P and $\mathcal{E}[\cdot]$ are the maximum transmitting power of the AP and the expectation operator, respectively. The sum-capacity for S , denoted by $C(S)$, is modeled as:

$$C(S) = \max_{P_j: \sum_{j \in S} \gamma_j^{-1} P_j \leq P} \sum_{j \in S} \log(1 + P_j), \tag{2}$$

where $\gamma_j = \frac{1}{\|w_j\|^2}$ is the effective channel gain of j th user in S and w_j is an element of $W(S)$.

The main objective of the user selection problem is to maximize the sum-capacity for a user group S :

$$\text{maximize } C(S). \tag{3}$$

$S \subset \{1, \dots, K\}; |S| \leq M$

Generally, (3) is accomplished by taking two steps: optimal user selection and power allocation. Note that the water filling is well known for the optimal power allocation approach [22]. In this paper, we assume the equal power distribution scheme for simplicity.

In OFDM systems the subcarriers may have different capacities due to frequency diversity, so it is hard to directly adopt (2) in the practical system. To handle this, we use a single metric to integrate all capacity values over the subcarriers. There are several ways to do this. One may simply average out the SNRs of the subcarriers as in OPUS [9] and Signpost [17]. Instead, we use ESNR (Effective SNR) [23] which can account for all the SNR values of the subcarriers. To elaborate on how ESNR works and evaluate its performance is out of scope of this paper, but we briefly explain how we adopt it for the capacity estimation.

Let $P_j^{[n]}$ be the SNR on the $n(\in N)$ th subcarrier of the j th user channel. Then, the effective SNR of this user (P_j) is defined as the follows:

$$P_j = B^{-1} \left(\frac{1}{N} \sum_n B(P_j^{[n]}) \right), \tag{4}$$

where $B(\cdot)$ and $B(\cdot)^{-1}$ denote BER function of SNR and its inverse function, respectively [23].

Now, we compute the desired capacity under the OFDM based system by using (2) and (4). Note that the ESNR can also be used for rate adaptation.

3.2 Motivation

For optimal user selection, we should consider all possible user groups, compare their capacities, and choose one group which gives the highest capacity. However, such a naïve and exhaustive search over the entire user set [i.e., $\sum_{m=1}^M \binom{K}{m}$] obviously results in very high feedback overhead as well as computation, especially when K is very large. To reduce the overhead, many schemes adopt the incremental user selection approach: the AP or BS (Base Station) incrementally chooses a user by employing a use selection metric in each iteration, e.g., channel strength,

orthogonality (angle)¹, (projected) norm and capacity gain.

Figure 3 compares the normalized capacity gain of the four selection metrics (random, maximum power, maximum angle, maximum projected norm), computed over that of the optimal selection. We select one user randomly as the first user and then, choose the other users according to the selection metric. The same first user is chosen for each metric, for fair comparison. We set M as 2 and K as 20. For the optimal selection, we consider the *capacity gain* among all possible user groups. We use two types of traces as input: real trace ('R') and synthetic trace ('N'), where the real trace is obtained from the USRP testbed while the synthetic trace consists of Gaussian random channel gains (Fig. 4).

As shown in the result, in random and max-power selection schemes, only about 15 and 20 % result the same as the optimal, respectively. The max-angle metric is better by showing 27–46 % of the optimal. Although the projected norm delivers the best performance (80 % result the same as the optimal), it gives around 20 % of non-optimal sum capacity. Such a performance loss happens because sometimes a user with a high power gain but small orthogonality could be selected. Such a small orthogonality may induce significant power loss to the previously selected user channels, as shown in the example in Fig. 5.

Though for a given user when $M = 2$, the capacity gain always selects the optimal user, but when $M > 2$ the benefit may decrease due to the nature of incremental user selection. Figure 4 shows the average capacity gain of the capacity gain metric against other metrics according to M . In the case of norm [Fig. 4(a)], the gain first increases drastically until $M = 4$, and after that begins to decrease. This is because when M is small, the impact of the power loss incurred by improper users selected by norm is relatively bigger than the case of large M . For the other metrics [Fig. 4(b)], the capacity gain metric shows much better performances as M increases.

Unfortunately, exploiting the capacity gain as a selection metric is a challenging task. First, to compute the capacity gain, all CSI feedbacks from users should be sent prior to the selection, which significantly increases a MAC overhead. One alternative is to compute them in a distributed manner. In other words, we can let each user estimate its own capacity gain. To realize this, users should know the CSI of the previously selected users, and thus the AP may need to send them back to the remaining users, which brings a large amount of frame exchanges. Instead of explicitly sending the CSI feedbacks, the OPUS [9]

¹ Note that the angle (θ) between the channels of users is computed by $\sin \theta = \frac{|h_{\perp} \cdot h|}{\|h_{\perp}\| \|h\|}$, where h is the channel vector of the target user and h_{\perp} is the vector that is orthogonal to the subspace spanned by the already selected user channels.

performs orthogonality probing to realize distributed user selection. However, since the users do not have the CSI of others, the OPUS has no choice but to use the projected norm. Second, an effective contention mechanism should be supported for distributed user selection. A time-domain contention scheme adopted in OPUS [9] is simple and intuitive, but the contention overhead is non-negligible, as mentioned earlier.

In summary, the main goal of this paper is to propose a protocol that enables the capacity gain metric for user selection in a scalable manner and at the same time addresses the time-domain contention overhead issue. Our proposed scheme, DiFuse, meets both requirements by using overhearing-based CSI acquisition and frequency domain signaling. We elaborate the details of our scheme in the next section.

4 Distributed frequency domain user selection

4.1 Protocol overview

Figure 6 illustrates the operation of DiFuse. Let us assume User 1 was first selected (we will explain the first user selection in the later subsection.). User 1 sends its CSI report² to the AP, and all remaining users ($k = 2, \dots, K$) overhear³ it and compute the *capacity gain* by comparing the sum capacities of two possible user groups: with and without itself. Then, the users concurrently send the gain value via SREQ to the AP, only if the gain is above 1. This implies that a user can become a candidate only if it gives a positive increment to the sum-capacity. Furthermore, since the users with gain values under 1 do not transmit the SREQ, the contention can be reduced. Say that User K has the highest capacity gain among all remaining users, i.e., $\frac{C(\{1, K\})}{C(\{1\})} > \frac{C(\{1, k\})}{C(\{1\})}$, $k \in \{1, 2, \dots, K\} \setminus \{1, K\}$. Then, in the second polling round, the AP polls User K via Selection REPLY (SREP) as the next user, and in turn, User K transmits the actual CSI feedback. DiFuse exploits the frequency domain contention to efficiently integrate the concurrent SREQ transmissions from the users. This

² There are two feedback mechanisms, explicit and implicit, but we do not consider the implicit CSI feedback scheme in this paper, since in recent standards and systems only explicit feedback is used. In fact, while in 802.11n, both implicit and explicit CSI feedback are allowed, in 802.11ac, transmit beamforming is limited to only the explicit feedback mechanism [1, 3] (implicit feedback is dropped.)

³ Recent MIMO researches use the overhearing scheme [16, 24, 25]; in MIMOMate [16] and TurboRate [24], overhearing is used for users to join uplink concurrent transmissions; in CoaCa [25] users should decode other users channel information to realize coordinated interference cancellation.

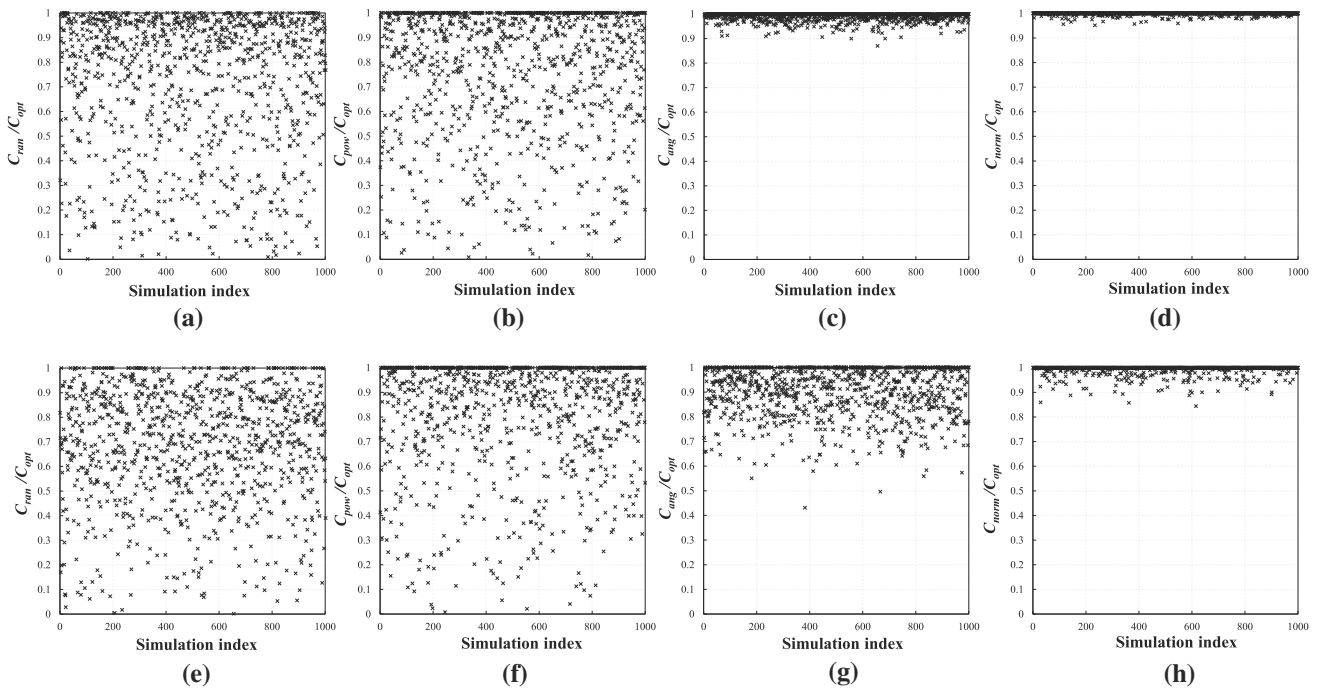


Fig. 3 Normalized capacity gain of each selection metric. We set $M = 2, K = 20$ and use two types of traces: real trace ('R') and synthetic trace ('N'). **a** C_{ran}/C_{opt} , 'R'. **b** C_{pow}/C_{opt} , 'R'. **c** C_{ang}/C_{opt} ,

'R'. **d** C_{norm}/C_{opt} , 'R'. **e** C_{ran}/C_{opt} , 'N'. **f** C_{pow}/C_{opt} , 'N'. **g** C_{ang}/C_{opt} , 'N'. **h** C_{norm}/C_{opt} , 'N'

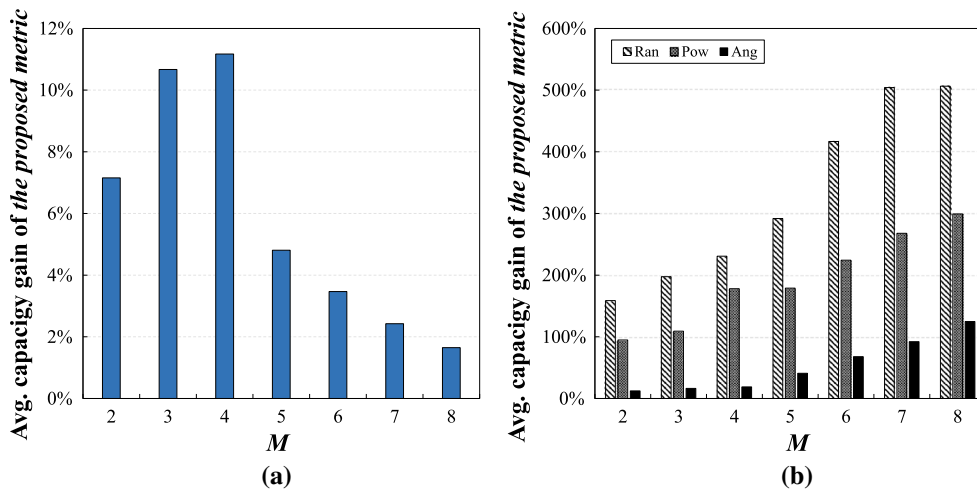


Fig. 4 Average capacity gain of the capacity gain metric against other metrics according to M . Zero-capacity cases are excluded from computing the average. **a** Norm case. **b** Other cases

process repeats while the remaining Degree of Freedoms (DoFs) are available.

4.2 Distributed feedback contention

We conduct the distributed feedback contention at the frequency domain. The capacity gain of each user is first mapped to a particular subcarrier of an OFDM symbol (setting a bit “1”), and then the users concurrently

transmit the symbols via SREQ. The AP can detect these combined multiple SREQs by using the typical FFT. An SREQ lasts only for a few OFDM symbols, and so its overhead is much smaller than that of the time-domain contention [9]. Note that multiple OFDM symbols could be misaligned due to several reasons such as different propagation delay or switching delay, but the total misalignment has been shown to be tightly bounded [12–14]. As long as the misalignment is less than the

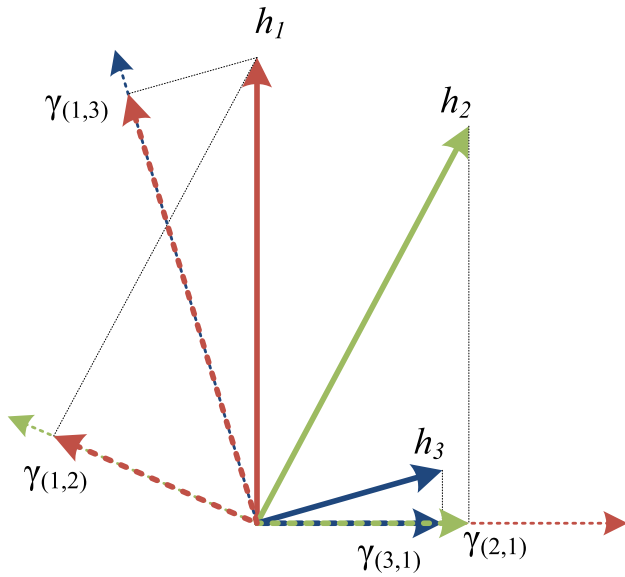


Fig. 5 An illustrative example of the projected norm based scheme when $M=2$ and $K=3$. We assume that User 1 was already selected. $\gamma_{(a,b)}$ denotes the channel of user a projected to the orthogonal subspace of channel of user b . User 2 will be selected if we select the user with the largest projected norm, i.e., $\gamma_{(2,1)} > \gamma_{(3,1)}$. However, the actual sum-capacity becomes bigger when User 3 is selected instead of User 2, i.e., $\log \gamma_{(1,2)} + \log \gamma_{(2,1)} < \log \gamma_{(1,3)} + \log \gamma_{(3,1)}$

Cyclic Prefix (CP), the AP can decode misaligned signals.

In making an SREQ, DiFuse uses a simple threshold based mapping scheme. We assume that for each polling round i , each data subcarrier $n \in N$ has its virtual threshold $E_{i,n}$ (we call this a *slot threshold*). For all i and n , all slot thresholds satisfy the following condition:

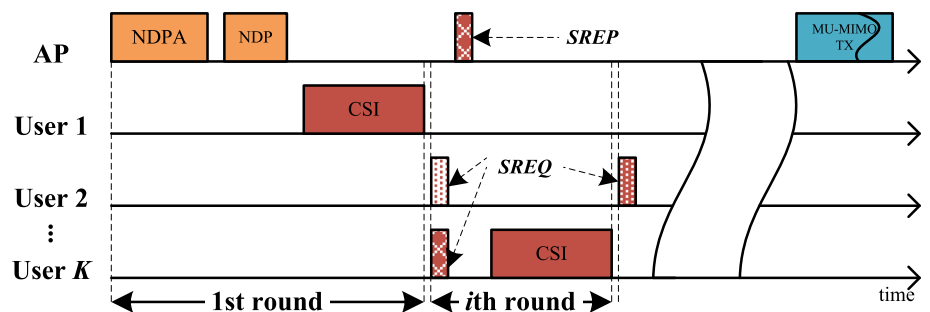
$$1 \leq E_i^{min} \leq E_{i,n} \leq E_i^{max} \tag{5}$$

$$E_{i,n} < E_{i,n+1}, \tag{6}$$

where E_i^{min} and E_i^{max} are the minimum and maximum slot threshold of round i , respectively. Recall that E_i^{min} cannot be < 1 .

We illustrate an example of slot thresholds in Fig. 7. There are several slot thresholds for each selecting round i ,

Fig. 6 Overview of DiFuse operation. The poll and CSI feedback exchanges between the AP and users are conducted via SREQ/SREP based on frequency domain contention. DiFuse requires up to M polling and feedback frames to achieve the user selection gain



and when making SREQ, DiFuse users use one of them. In particular, at the beginning, users use slot thresholds with $\delta = 0.99$, and according to SREQ transmission results, users can use other threshold with the decreased δ . The detailed description of δ and the slot threshold is given in the next subsection. Given slot thresholds, a channel gain of User k ($\lambda_{i,k}$) is mapped to n th subcarrier, if it satisfies the following condition:

$$E_{i,n} \leq \lambda_{i,k} < E_{i,n+1}, \tag{7}$$

where $E_{i,|N|+1} = \infty$.

Since the slot thresholds of round i are monotonically increasing with n , as shown in (5) and (6), a user with a higher capacity gain will activate a higher subcarrier index on SREQ. When users send SREQs, according to the capacity gain status of users, one of following cases may occur.

- *Success case* A success case happens only when a single user is associated with the highest activated subcarrier index of the combined SREQ. The DiFuse AP extracts the SREQ of the selected user from the combined SREQ, which we call SREP-I, which is then broadcast to all remaining users. The user checks if SREP-I matches the previously sent SREQ, if so, then it transmits the actual CSI feedback.

Example Let us assume that the capacity gain of User 2 and User 3 are 1.750 and 1.703, respectively. Then, with slot thresholds of $\delta = 0.99$ and $i = 2$, users will send their SREQs to the AP as shown in Fig. 8(a). Since only User 2 is associated with the highest activated subcarrier index, the contention is successfully resolved.

- *Collision case* A collision case happens when more than one users are associated with the highest activated subcarrier index of the combined SREQ. Since the SREQs do not include any user information such as user ID or address, the SREP-I will invoke multiple users to send their CSI feedbacks at the same time. If the AP fails to apprehend the received CSI due to the collision, it just quits the user selection procedure and starts the data transmission for the already selected users.

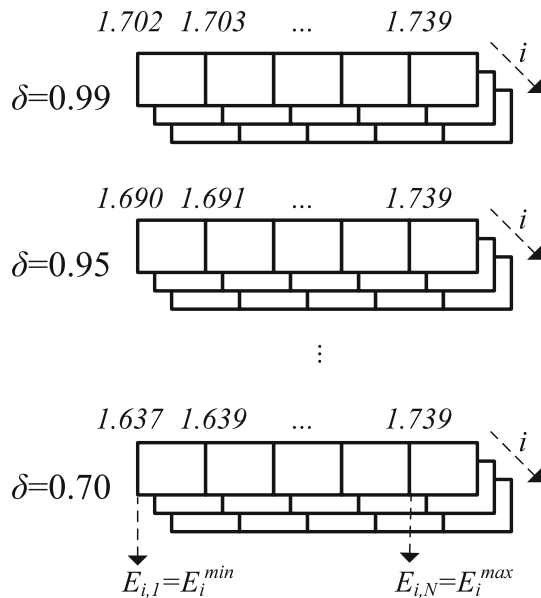


Fig. 7 The main concept of slot thresholds of DiFuse. We generate several slot thresholds for idle case. Here δ presents a seed parameter

Example Let us assume that the capacity gains of User 2 and User 3 are 1.75. Then, their SREQs will be the same and finally CSI collision will occur, as shown in Fig. 8(b).

- *Idle case* An idle case happens if there are no associated users for SREQ, which means $\lambda_{i,k} < E_i^{min}$, for all users [Fig. 8(c)]. Therefore, in this case, the SREQ transmissions do not occur. Instead, after timeout (2 SIFS), the AP transmits the pre-defined OFDM symbols (we call this SREP-II) to inform the users of the idle case. Compared to the collision case, the idle case hardly affects the system performance, since the transmission times of SREQ and SREP are very short (a few microseconds per each). Based on this fact, we allow users to perform re-mapping and re-transmission of SREQs again, when a idle case happens. Specifically, in each retry, users make SREQs with different slot thresholds. Note that the number of retries is limited by the pre-defined threshold. The SREP-I and SREP-II may require some modifications to the legacy standard, but we note that this change can reduce up to 3 OFDM symbol transmissions, compared to the legacy polling frame.

Example Let us assume that the capacity gains of User 2 and User 3 are 1.690 and 1.691, respectively. With slot thresholds of $\delta = 0.99$ and $i = 2$, each user cannot generate SREQ because their capacity gains are too low. After receiving SREP-II, users regenerate SREQs with a decreased δ (e.g., 0.95). Here, users can successfully generate SREQs and User 3 will be selected as the i th user, as shown in Fig. 8(c).

4.3 Slot threshold design

To maximize the number of success cases, we devise an empirical method to dynamically set the slot thresholds of DiFuse. We address several challenges to accomplish this. First, it is not easy to know the exact channel status of users before the user selection. For this reason, we set slot thresholds empirically from the channel status statistics. To do this, the AP collects the channel status of the users and computes the capacity gain distribution. We provide an example of the capacity gain distribution in Fig. 9. Note that the capacity gain distribution is computed offline.

Second, it is difficult to reduce both collision and idle cases since we could not estimate the dynamics of user channels. Thus, we aim to avoid only collision cases, in that the penalty of the idle case is far smaller than that of the collision case. To accomplish this, we let the distance between E_i^{min} and E_i^{max} be relatively short. However, if the two values are too close together, it decreases the multi-user diversity gain because it is likely to have very few or even no associated users between the two values (i.e., idle case). On the other hand, if the two values are too far apart, it increases the probability that more than one users are associated with the same activated sub-carrier (i.e., collision case).

Based on the above insight, we allow multiple slot thresholds for one selection round. In particular, we first fix E_i^{max} (E_i^{max} is set to $F_i^{-1}(1)$, since it reflects the expected gain that the system can obtain) and make several E_i^{min} values according to δ (seed parameter) as follows:

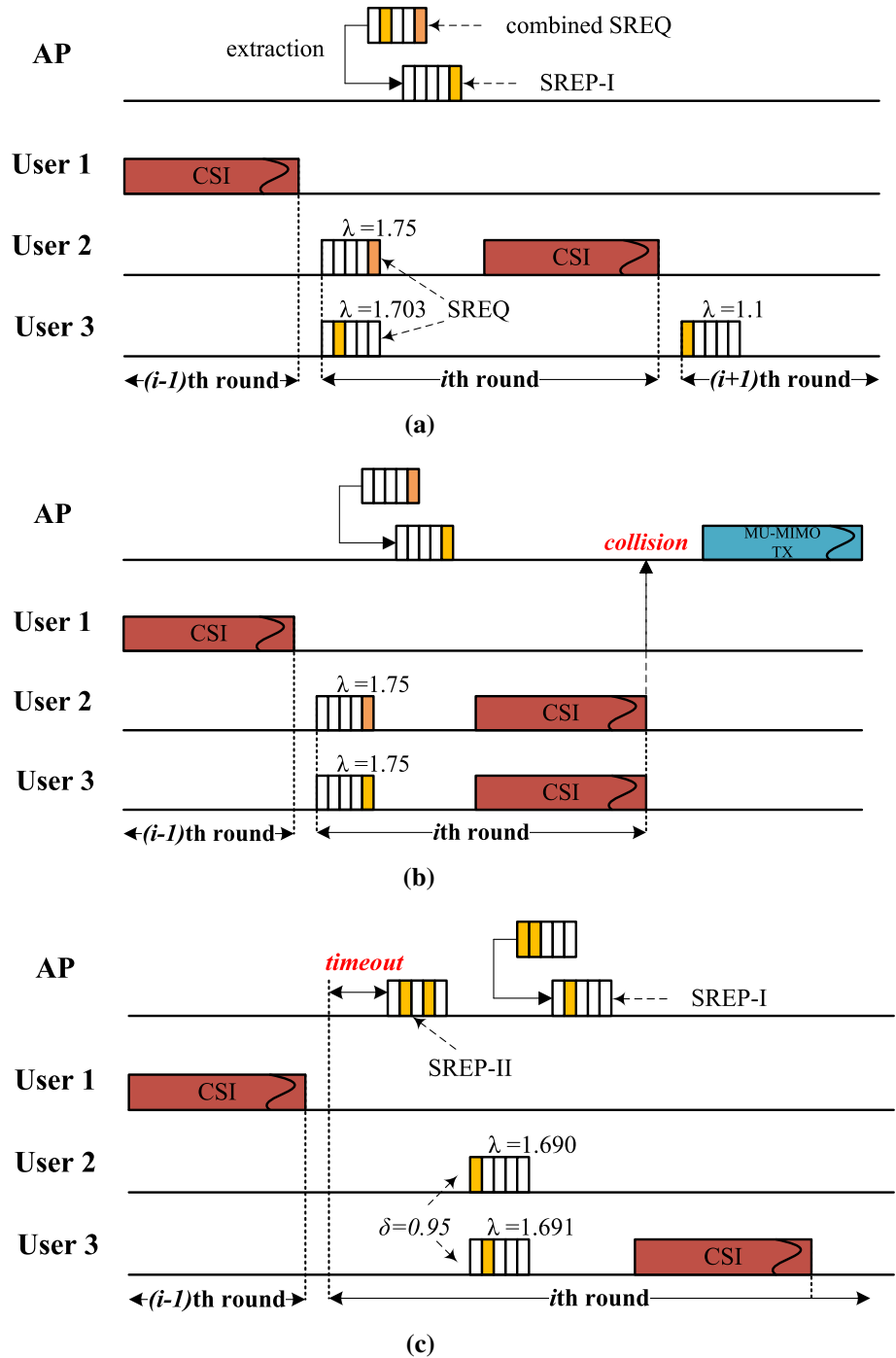
$$E_i^{min} = F_i^{-1}(\delta), \quad (8)$$

where F_i is the CDF of capacity gain distribution for selection round i .

Since the collision probability decreases with δ , we set the initial δ as 0.99. After that, if an idle case happens, δ is decreased and users retry SREQ transmission with slot thresholds of the new δ . In our case the minimum available value of δ (denoted by δ_{thr}) is 70 % ($F_4^{-1}(0.7) \simeq 1$ (see Fig. 9).)

Once E_i^{min} and E_i^{max} are given, other remaining thresholds can be determined in various ways. For example, the gap between the thresholds may increase or decrease according to some functions (e.g., ‘linear’, ‘exponential’), or we can make it by reflecting the capacity gain distribution itself (we call this ‘dist’ design). Figure 10 shows the examples of slot thresholds according to different mapping designs. As we will show later in Fig. 18, the performance of ‘dist’ design is better than that of ‘linear’ design. After determining a set of slot thresholds, the DiFuse AP periodically broadcasts them to all users. Note that we do not need to broadcast slot thresholds for every packet transmission.

Fig. 8 Three cases of SREQ transmissions. **a** Success case. **b** Collision case. **c** Idle case



4.4 Proportional fair selection

The fairness among users is an important factor in communication systems. We introduce DiFuse-PF which considers the proportional fairness based on DiFuse. DiFuse-PF uses the following utility to select users:

$$U_k = \frac{\mathcal{T}_k}{\mathcal{R}_k}, \tag{9}$$

where \mathcal{T}_k , \mathcal{R}_k are the current available data rate and average data rate of user k , respectively.

The main difficulty of implementing the proportional fair selection lies in the fact that the AP does not know the instantaneous data rates of the users (\mathcal{T}_k) when the decision has to be made. However, in DiFuse-PF, each user can easily estimate \mathcal{T}_k from the sum-capacity computation in the polling phase. Moreover, the DiFuse-PF users must use

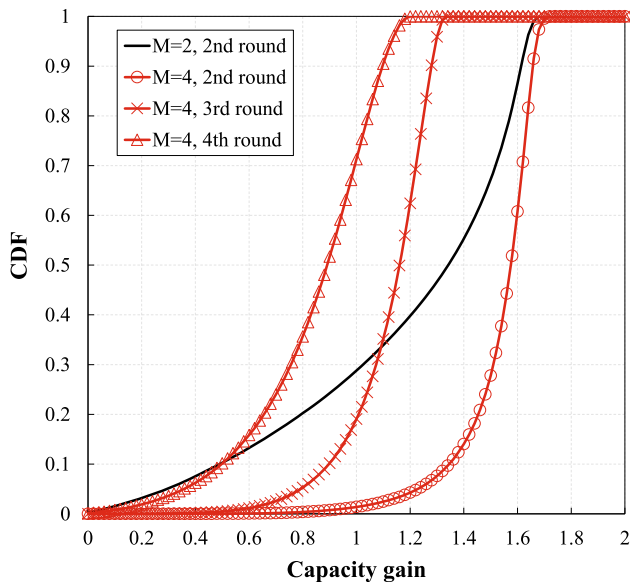


Fig. 9 Capacity gain distribution for $M = 2$ and 4. As the selection round advances, the average capacity gain decreases. This result is predictable, because the power allocated to each user is reduced as a user group size increases

the gain of (9) to mark the subcarrier in their SREQs. Figure 11 shows the utility gain distribution. From the result, we see that the utility gain distribution has a similar pattern with the capacity gain distribution in Fig. 9, but it has a different scale. The remaining procedure is essentially the same with the technique described in 4.3.

4.5 First user selection

The aforementioned user selection of DiFuse actually starts from the second user, but the performance of DiFuse may

also depend on how the first user is selected. Unfortunately, optimally selecting the first user is difficult since no CSI is available at the time of user selection. In other words, the AP, without any instantaneous channel information, must select a first user to send the CSI feedback.

To handle this issue, DiFuse and DiFuse-PF refer to the statistics of the previous channel status of the users, for the first user selection. More specifically, to maximize the sum-capacity, the user with the largest channel gain will be selected as the first user, similar to SUS [6], while in DiFuse-PF, the user with maximum utility (9) will be chosen to achieve proportional fairness. This method should work well when transmissions of the AP happen in channel coherent time which typically ranges from 15 to 100 ms [9, 11, 26]. We summarize the metric comparison of several user selection protocols stated so far in Table 1.

4.6 Discussions

4.6.1 Gain reduction compared to the optimum

DiFuse may not give the optimal result due to the nature of the incremental selection procedure. To obtain better capacities, we may need an additional procedure to switch the existing selected users with new users, similar to GUSS [8]. However, this may lead to more interactions between the users and an AP, thus resulting in higher overhead and diminishing the achieved gain.

4.6.2 Computational complexity

In computing the sum-capacity, the complexity mainly lies on the channel inversion. For a subchannel, the complexity for channel inversion is $O(M^3)$, in the worst case where

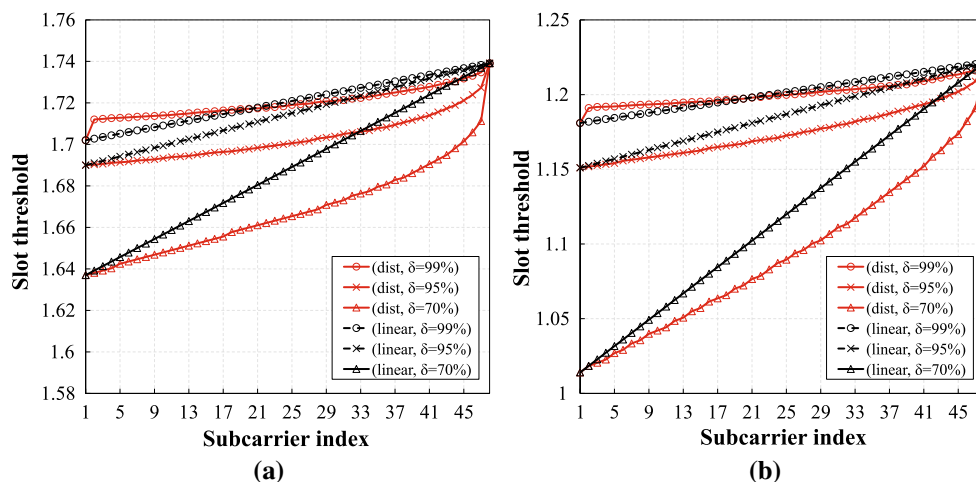


Fig. 10 Examples of slot thresholds for different mapping configurations. In the ‘dist’ design, gaps between slot thresholds are determined by the capacity gain distribution, while in the ‘linear’ design, gaps are determined to be equal. **a** 2nd selection round. **b** 4th selection round

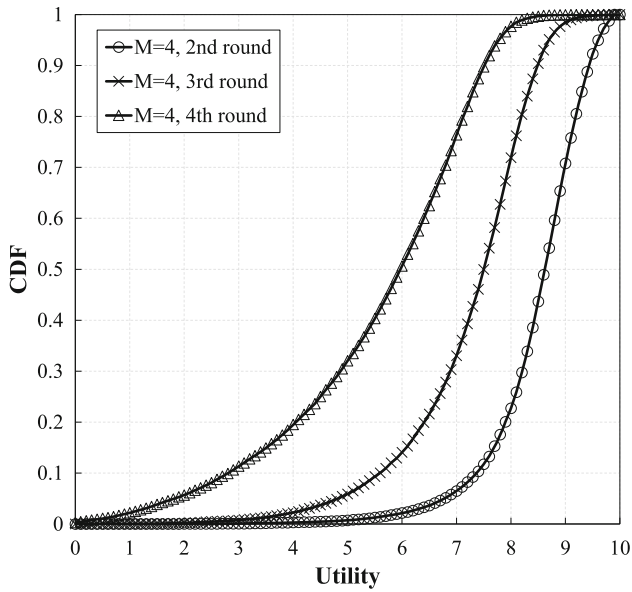


Fig. 11 The gain distribution of the proportional fair utility for $M = 4$

Table 1 Metric comparison between user selection protocols

	First user selection	$i (> 1)$ th user selection
SUS [6]	$\ \gamma_k\ ^2$	$\ \gamma_k\ ^2$
OPUS [9]	Random	SINR
OPUS-PF [9]	$\frac{1}{R_k}$	SINR
DiFuse	$\ \mathbf{h}_k\ ^2$	λ_k
DiFuse-PF	U_k	U_k

$M=|S|$. According to the recent result [26], the absolute processing time is actually affordable when M is modest (<50). For example, $M=|S| < 12$, the channel inversion only takes merely $10 \mu s$.

4.6.3 Selfish user behavior

A selfish user could manipulate the SREQ to be selected for the MU-MIMO transmission by setting a bit on the highest subcarrier index. However, the DiFuse AP can easily detect such selfish user behavior by comparing the actual CSI feedback and the user’s SREQ value.

5 Performance evaluation

In this section, we first evaluate the frequency domain contention scheme through the benchmark testbed. Then we evaluate the system-level performance via the trace-driven emulations.

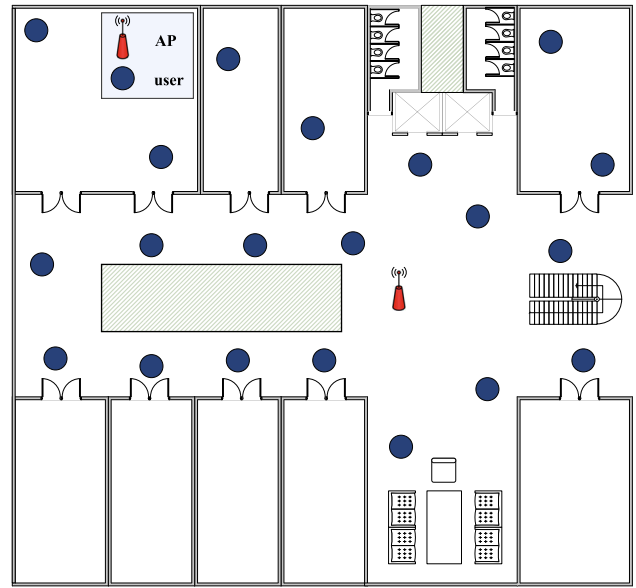


Fig. 12 Experiment environment

5.1 Micro benchmark

5.1.1 Benchmark testbed setup

We implement the USRP/GNURadio testbed on OFDM PHY of 64 FFT size and 48 data subcarriers [15]. All nodes are equipped with USRP N210 on SBX daughterboard and work on 10 MHz bandwidth⁴. The AP is equipped with $M = 2, 4$ antennas and the user nodes are randomly located as shown in Fig. 12. In USRP/GNURadio testbeds, real-time experiments generally have limitations; when the software radios exchange signal samples between the host and the RF front end, it incurs very high latency [27]. This limitation exacerbates under larger number of K . For this reason, we evaluate the performance of frequency domain contention under the USRP testbed, and conduct remaining parts as a trace-driven emulation approach similar to recent experimental works [11, 12, 14, 23].

5.1.2 SREQ detection

To evaluate the feasibility of multiple users simultaneously sending the SREQ, we measure the SREQ detection probability by the AP. We let 8 users each transmit the SREQ to the 4-antenna AP under two different circumstances. For *similar case*, the difference between the maximum and minimum received signal power is less than 5 dB, and otherwise it is *different case*. To control the dedicated power levels, we make gain adjustments using the transmit power control before starting the real

⁴ In trace-driven emulations, the bandwidth is set to 20 MHz.

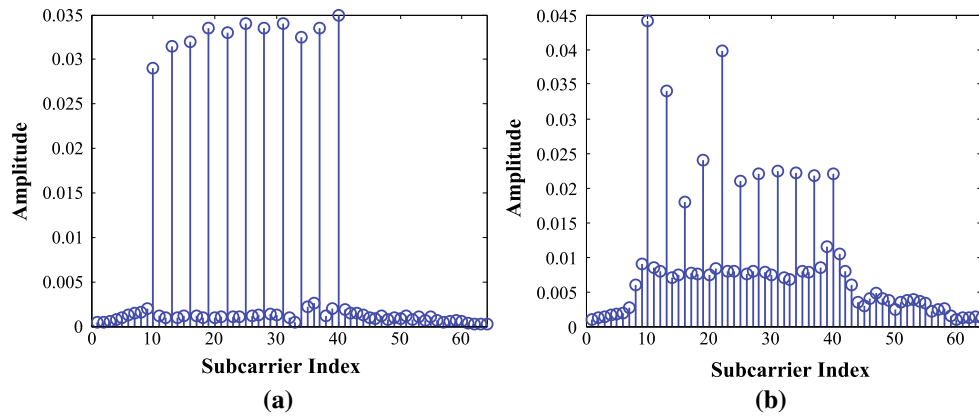


Fig. 13 FFT results under two different transmit power cases. **a** Similar case. **b** Different case

measurements. As for SREQ, we assign different subcarrier indexes from $\{10, 13, \dots, 40\}$ for each user. However, four indexes (31, 34, 37, 40) are assigned to the last user. Also all 8 users are connected to a central controller to achieve transmission synchronization.

The FFT result examples regarding the two cases are compared in Fig. 13. The threshold for detection is set to 10 dB over noise level. As expected, the result of the similar case is more clear. In the different case, though the noise level increases due to the power difference, we can still accurately detect the signals at all 11 dedicated points.

Next, we measure the detection error probability according to the received SNR synthesized from the multiple SREQs. We define the detection error as the event that the AP mis-detects the SREQ of the largest subcarrier index, i.e., 40. Recall that the SREQ of the largest subcarrier index, i.e., SREQ from the highest capacity gain user, matters most in DiFuse. As shown in Fig. 14, in the similar case, the detection error is only about 0.6 % at the high SNR range. The detection error increases in the different case, but is always <5 % even in the low SNR range (the majority of detection error is due to the false negative.). The signal mis-detection in subcarrier level comes from two major sources, interference and misalignment. For example, back2F [12] suffers from the strong self-signal interference by the full-duplexing. In contrast, in DiFuse, only the noise can affect the detection performance, since control frames play a role of holding the medium, like RTS/CTS. Also, as long as the misalignment is less than CP, the AP can still detect all SREQs reliably, as mentioned earlier.

5.1.3 Frequency domain contention

To evaluate the frequency domain contention of DiFuse, we measure two probabilities of collision case (p_{col}) and idle case (p_{idle}) according to different δ_{thr} . Each value is

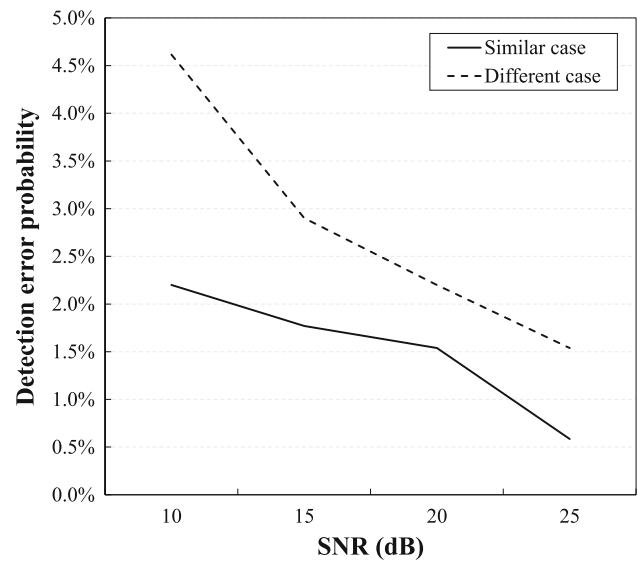


Fig. 14 Detection error probability versus SNR. The SNR is synthesized from the SNRs of multiple SREQs. Even in the worst case, the detection error of DiFuse is still <5 %

averaged over whole polling rounds. Based on the results, we additionally compute the expected throughput according to different transmission lengths (0.5, 3.0, 5.5 ms). For comparison, we conduct an emulation for the case of $K = 100$, and illustrate both results in Fig. 15.

From the result, we can clearly observe the tradeoff between p_{idle} and p_{col} : p_{idle} increases with δ_{thr} , while p_{col} decreases. First, the initial p_{idle} is quite high in both cases. Specifically, in the case of $K = 8$, the actual number of contending users is very small (i.e., 7, 6, 5 for each selection round, respectively), and thus p_{idle} is much higher than that of $K = 100$. Actually this high p_{idle} shows that our slot threshold design fits well in the real scenario. Recall that our design mainly aims to lower p_{col} , at the expense of the increase of p_{idle} . As a result, p_{col} of two cases are 2.5 and 2.9 %, respectively. And thus we can effectively limit

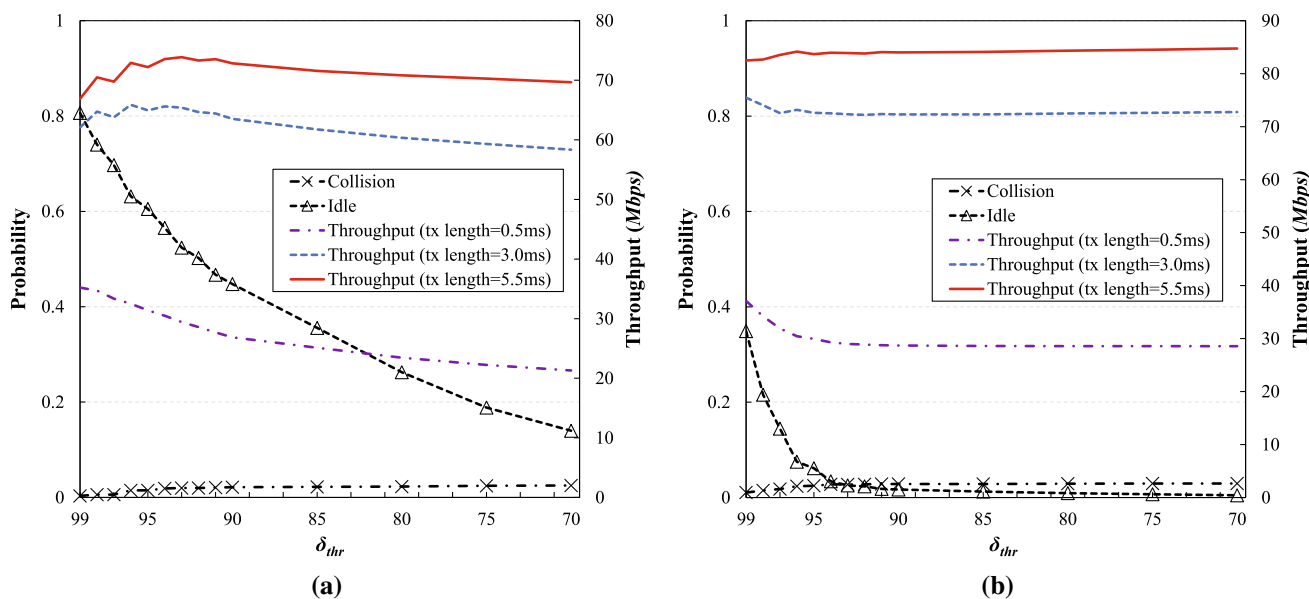


Fig. 15 Performance comparison on the frequency domain contention. The DiFuse can limit the collision probability to 3 % for 100 users. **a** $K = 8$. **b** $K = 100$

the impact of p_{col} on the system performance (e.g., throughput).

From the viewpoint of the throughput of different K , we can see that their patterns are different from each other. For small K , the throughput decreases with δ_{thr} . This result indicates that the throughput of small K is highly affected by the re-transmission overhead in the frequency domain contention. In particular, the small K increases p_{idle} , and thus invokes more re-transmissions of SREQs and SREPs. On the other hand, for large K , p_{idle} converges to 0 quickly, and thus it is very unlikely to go into the re-mapping procedures. As a result, the throughput seems to be not much affected by δ_{thr} . In addition, we can see that there is an optimal δ_{thr} point to maximize the throughput in each case. For small K , except the case of transmission length of 0.5 ms, the throughput is slightly increased before going down. For large K , the results show the similar pattern to those of small K , but the gap is very small because retries of SREQ/SREP transmission rarely happen.

5.2 System-level performance

5.2.1 Setting

As mentioned earlier, our USRP based testbed is limited by the interface latency, so we cannot directly implement all 802.11 MAC functionalities. Furthermore, it is even more challenging to implement distinct schemes of each protocol like frequency domain contention, 2-D contention and orthogonality probing in real-time, and there are far more

difficulties in a large scale network environment. For example, in our case, among three main components needed to build DiFuse (frequency domain contention, user selection and beamforming), user selection and beamforming could not be implemented in real-time since the MAC overhead of implementation exceeds the channel coherence time and thus causing beamforming failures. Fortunately, the frequency domain contention can be implemented because it is not that much affected by the channel coherence time.

For this reason, we conduct a trace-driven evaluation for system-level performance comparison. We have collected CSI traces and contention results during the frequency domain contention evaluation and used them for user selection and beamforming emulation offline. Plus, for MAC time-related intervals, only the capacity and throughput computations are performed by the MATLAB. Though our trace-driven emulation lacks some details, we believe that it is enough to show and compare how the proposed protocols work.

We emulate 802.11ac, SUS⁵, OPT, OPUS, OPUS-PF, DiFuse and DiFuse-PF with the basic parameters defined in the 802.11ac specification [1]. In addition, we also emulate MIMOMate [16] and Signpost [17], but since they are uplink user selection protocols, it is inevitable to make some modifications on them. For these two protocols, we evaluate the throughput performance only. P is set to 15 dB and we use the ESNR based rate adaptation scheme [23].

⁵ We set the parameter α used in SUS as 1, and so it does not render the early termination.

Our traces basically contain 10,000 per-subcarrier CSI traces for 20 users. For large K (e.g., 200), we extend the trace by multiplying random complex numbers. Unless otherwise stated, K , M and transmission length is set to 20, 4 and 0.5 ms, respectively.

5.2.2 The impact of selection metrics

In this section, to evaluate the impact of selection metrics, we compare the sum-capacities of five different selection metrics according to M , in Fig. 16. We add a result of the optimal solution in the figure via exhaustive search. Except for the optimal case, the first user is randomly selected for all cases. Note that this result does not include the CSI feedback overhead.

From the result, we observe that the capacity gain metric consistently outperforms other metrics, while closely approaching the optimal case by up to 95 %. The capacity improvements of capacity gain metric which is used in DiFuse are up to $2.0\times$, $1.9\times$, $1.6\times$ and $1.1\times$ compared to random, max-power, max-angle and projected norm, respectively. As expected the random user selection gives the worst performance. The performance of max-power is better than the random selection, but its gain is marginal. The orthogonality gives more capacity gains than the channel strength, but using only this metric alone limits the performance, especially when M is large. On the other hand, the projected norm metric shows a quite good performance, since it considers the tradeoff between channel strength and orthogonality. However, as discussed in Sect. 3.2, this metric cannot

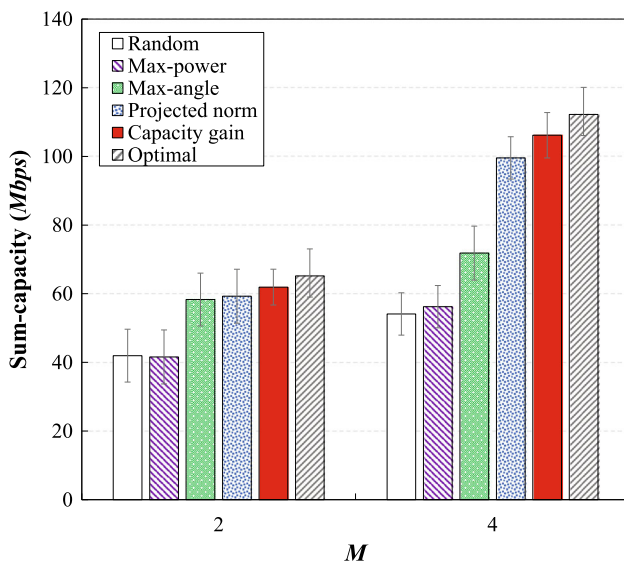


Fig. 16 Sum-capacity versus M . The capacity gain metric which is used in DiFuse achieves the sum-capacity improvement of $2.0\times$, $1.9\times$, $1.6\times$ and $1.1\times$ over random (802.11ac), max-power, max-angle and projected norm (SUS, OPUS), respectively

guarantee the highest sum capacity gain in every selection round. In our result, 16 % of total cases suffer from that problem.

5.2.3 CSI feedback overhead

In this section, we investigate the CSI feedback overhead of each protocol. Figure 17 shows the average CSI feedback duration as a function of K . First, in 802.11ac the CSI feedback overhead is fixed regardless of K because it always gathers CSI of users as many as M . SUS conveys a much larger overhead than other schemes, constantly increasing with K . We observe that OPUS shows a higher overhead than 802.11ac. Even though OPUS limits the CSI overhead by terminating the selection earlier than 802.11ac, it suffers from the inefficient time-domain contention. In the meantime, DiFuse has the lowest overhead over all cases, thanks to the frequency domain contention and quick termination. Such overhead reduction further improves the throughput performance of DiFuse based on the sum-capacity enhancement by the user selection method.

5.2.4 Throughput

We compare the throughput of each protocol including MIMOMate [16] and Signpost [17] in Fig. 18, under different network configurations. Recall that MIMOMate and Signpost are originally designed for uplink MU-MIMO, so we need to first modify them to match our evaluation scenario as follows:

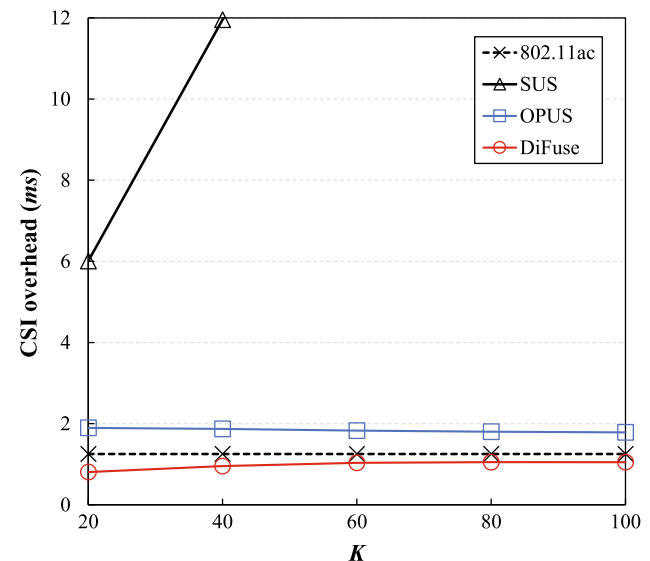


Fig. 17 CSI feedback duration versus K . The average feedback time of DiFuse is lower than those of other protocols

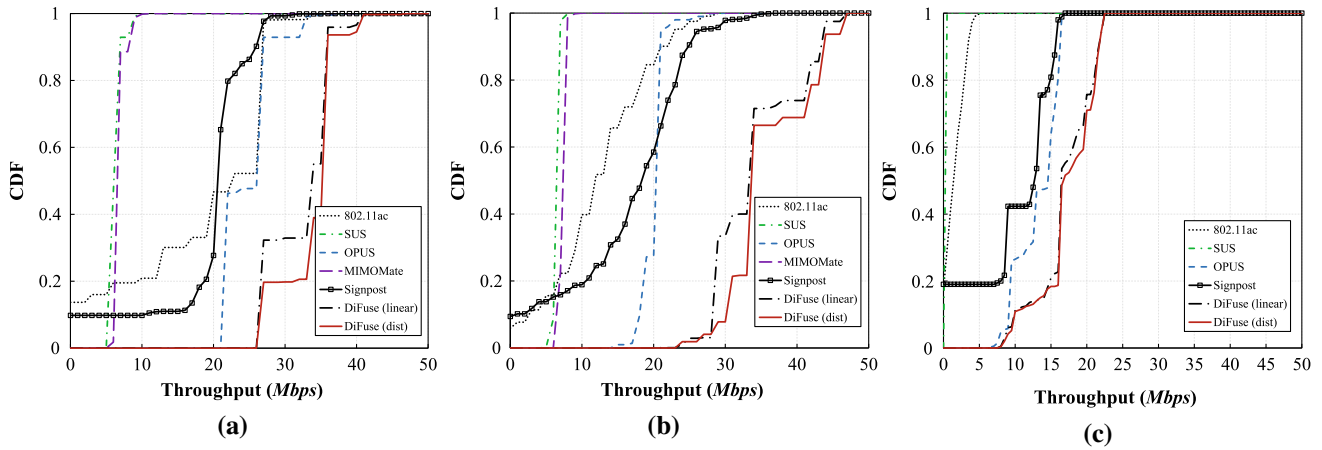


Fig. 18 Throughput versus M . DiFuse obtains the maximum throughput gain of $4.8\times$, $2.8\times$, $1.8\times$, $4.5\times$ and $2.1\times$ on average, over 802.11ac, SUS, OPUS, MIMOMate and Signpost, respectively.

- **MIMOMate** We simply assume that the AP can compute the optimal user group, instead of MIMO-Mate relations, at the expense of receiving CSI feedbacks from all users like SUS [6]. Also we omit the result of the case of $M = 20$, due to its excessive large search space.
- **Signpost** The 2-D contention and angle based user selection of original Signpost are unchanged, but we let one of orthogonal probing directions made by the AP include the channel direction of the first selected user. Besides, instead of performing the collision recovery by users, the AP stops the selection procedure right after it senses any feedback collisions. The total timeslots are set to 50 and 128 for $K = 20$ and $K = 200$, respectively.

As shown in the result, DiFuse outperforms other protocols in all cases. Except for the case of (c), DiFuse obtains the maximum throughput gain of $4.8\times$, $2.8\times$, $1.8\times$, $4.5\times$ and $2.1\times$ on average, over 802.11ac, SUS, OPUS, MIMOMate and Signpost, respectively, especially when DiFuse uses the ‘dist’ slot thresholds design. Through the ‘dist’ design, DiFuse can achieve the additional gain of maximum 5 % than the ‘linear’ design. The gap between DiFuse and OPUS becomes even larger with increasing M . First, the frequency domain contention of DiFuse gives a much smaller overhead than the contention scheme used in OPUS. Second, the effective selection method of DiFuse provides a higher capacity than that of OPUS.

Additionally, we observe zero-throughput cases in the case of 802.11ac. This is the limitation of ZFBF. The AP wastes most transmit power for interference cancellation and the intended signal may have low power. This can lead to zero-throughput. 802.11ac significantly suffers from the zero-throughput, due to the random user selection. SUS achieves

better performance than 802.11ac, but the gain over 802.11ac is not that large due to its long CSI feedback time. Also, MIMOMate obtains only a little gain over SUS even though it can exploit the optimal user set, since the CSI feedback overhead overwhelms the capacity gain. Meanwhile, we see that the throughput of Signpost is not as high as that of OPUS though the 2-D contention is more effective than the contention scheme of OPUS. This is because Signpost uses an angle based user selection. In the original Signpost, angle based user selection could work well since the AP decodes uplink frames by using ZF-SIC, but in downlink scenario where the AP uses ZFBF, norm is more effective than the angle metric from the user selection perspective.

In massive MIMO case (c), DiFuse still obtains modest throughput even under significantly large CSI reports. **a** $M = 2, K = 20$. **b** $M = 4, K = 20$. **c** $M = 20, K = 200$

In massive MIMO [26, 28–30] or distributed MIMO systems [31, 32], we can exploit much more transmit antennas. Figure 18(c) shows the result when $M = 20, K = 200$. Recall that the size of each CSI report is approximately 10 times larger than that of the case (a). In 802.11ac and SUS, the performance of ZFBF drops dramatically, due to the tightness of the DoF. Recall that in both protocols, the number of selected users is M . In result, almost zero-throughput happens in both 802.11ac and SUS. In contrast, the results of DiFuse and OPUS show that they are feasible under many-antenna systems in practice. Also, DiFuse brings a higher throughput than OPUS in this scenario.

Next, we measure the throughput of four protocols, according to K in Fig. 19. The random selection of 802.11ac gives constant throughput regardless of K . We can clearly see that the throughput of SUS decreases with K . Meanwhile, OPUS and DiFuse persistently obtain higher throughput over other two protocols, due to the small number of CSI feedback transmissions, plus the capacity enhancement by selection. Specifically, DiFuse outperforms OPUS over all cases.

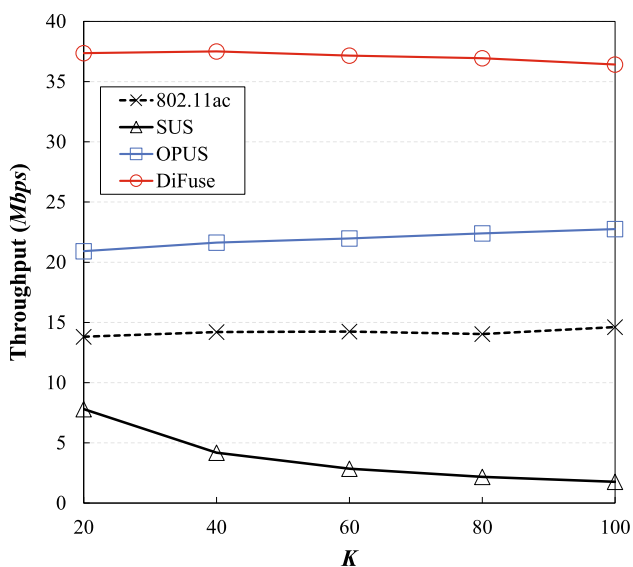


Fig. 19 Throughput versus K . DiFuse consistently outperforms other protocols. DiFuse and OPUS benefit from the multi-user diversity gain

5.2.5 Hidden terminal

Since DiFuse users exploit the CSI feedback overhearing, in some cases, some users may not be able to participate in the selection decision due to the hidden terminal problem. For example, let us assume that user A and user B share the same AP and both employ DiFuse, but cannot hear each other's transmissions, i.e., are hidden terminals. If user A was first selected by the AP, then user A will transmit the CSI to the AP, but user B will not be able to overhear this. Thus, user B will be left out from the selection procedure. Here, we evaluate the effect of the hidden terminal problem on the performance of DiFuse. First, we set K and M as 20 and 4, respectively. To artificially generate hidden terminals, we place three users (User ID 1 ~ User ID 3) apart from others, so that they fail to overhear the other users' transmissions, (i.e., User ID 4 ~ User ID 20), and vice versa.

Figure 20 shows that DiFuse still outperforms other protocols even when the hidden terminal problem occurs. Some of the users cannot join a certain user selection process when a previously selected user is hidden to them. For example, User 1 ~ User 3 cannot join the selection process if User 4 was previously selected and has sent its CSI report to the AP. Even though such users cannot be selected, they may have another opportunity in the later downlink transmissions, e.g., when non-hidden users are selected. On the contrary, we find that in some cases, the hidden terminal problem can even be a benefit to DiFuse since it may help to limit the network size, reducing the contention for feedback. For example, when User 1 is selected as the first user, the only candidates become User 2 and User 3. In this case the

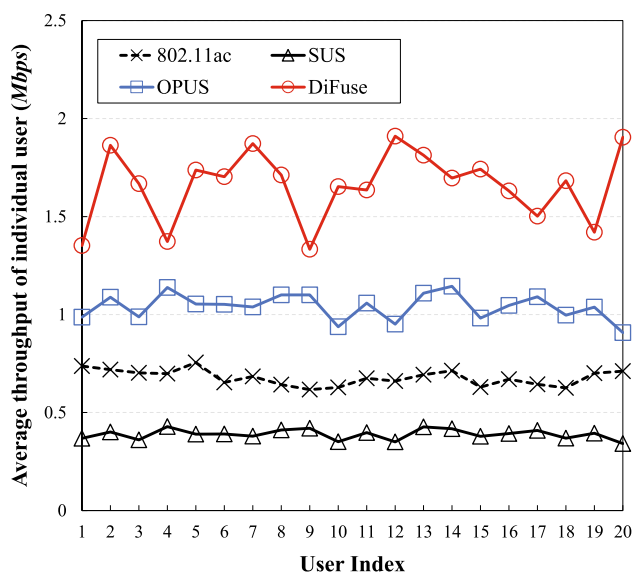


Fig. 20 Throughput of 802.11ac, SUS, OPUS and DiFuse for hidden terminal case. DiFuse still provides a higher throughput than other protocols even when there exist some hidden terminals

three users can achieve higher throughput via beamforming with sufficient DoF(s) (e.g., $3 < M$).

5.2.6 Fairness

To evaluate the fairness performance of DiFuse-PF, we let 20 users experience diverse average SNRs ranging from 5 to 20 dB; the user with a larger ID number has a higher SNR than the user with a lower ID number.

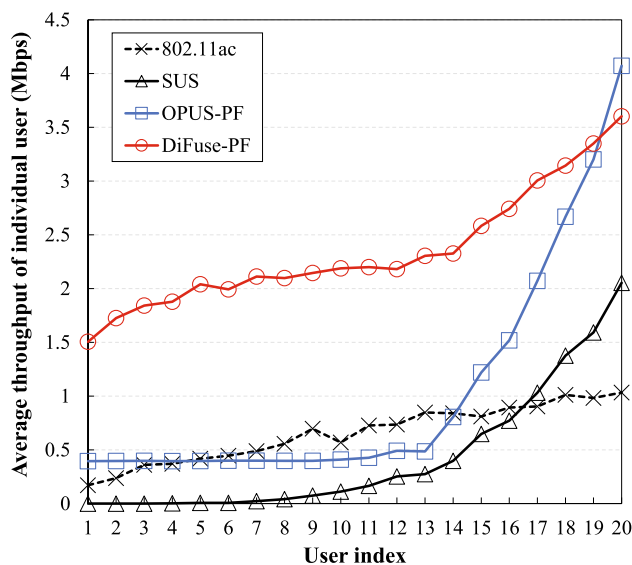


Fig. 21 Throughput comparison of 802.11ac, SUS, OPUS-PF and DiFuse-PF. DiFuse-PF achieves the best fairness even when the users experience diverse channel qualities, while maintaining the throughput

Figure 21 shows the throughput that each user attains under each protocol. 802.11ac severely suffers due to the low SNR users, diminishing the MU-MIMO effectiveness. Although SUS shows higher throughput than 802.11ac, some users with low SNR suffer from the starvation. OPUS-PF [9] is the proportional fair version of OPUS, where the user with the lowest average throughput is likely to be selected as the first user, so that users with low SNRs can maintain throughput. However, just considering the first user for user fairness is not enough, especially when users experience diverse channel qualities. Users with high channel gains will eventually join the MU-MIMO transmission group in the following rounds. Recall that OPUS-PF and OPUS both use the same selection metric to select users except for the first user. Meanwhile, DiFuse-PF shows better fairness than others since the probability that users with low SNRs have a transmission chance is increased.

6 Conclusion

In this paper, we have proposed a new user selection protocol called DiFuse, which employs the sum-capacity gain as the user selection metric and exploits the frequency domain contention to reduce CSI feedback overhead. To evaluate the performance of DiFuse, we conduct USRP/GNURadio based experiments as well as the extensive trace-driven emulations. The results show that DiFuse consistently outperforms other schemes in terms of throughput and proportional fairness.

Acknowledgments This work was supported by the Basic Science Research Program through the National Research Foundation of Korea funded by the MSIP (NRF-2013R1A1A1006823), by the MSIP (Ministry of Science, ICT & Future Planning), Korea, under the National Program for Excellence in Software (R2215-16-1006), by Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIP) (No.B0190-16-2017, Resilient/Fault-Tolerant Autonomic Networking Based on Physicality, Relationship and Service Semantic of IoT Devices) and MSIP (Ministry of Science, ICT and Future Planning), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2015-R0992-15-1023) supervised by the IITP (Institute for Information & communications Technology Promotion).

References

1. 802.11ac. (2013). Wireless LAN Medium Access Control and Physical Layer Specification. *IEEE Std. 802.11ac Draft 5.0*.
2. 11ax, http://www.ieee802.org/11/Reports/hew_update.htm.
3. Aruba Networks. (2014). 802.11ac White Paper.
4. Dimic, G., & Sidiropoulos, N. D. (2005). On downlink beamforming with greedy user selection: Performance analysis and a simple new algorithm. *IEEE Transactions on Signal Processing*, 53(10), 3857–3868.
5. Yoo, T., & Goldsmith A. (2005). Sum-rate optimal multi-antenna downlink beamforming strategy based on clique search. In *Proceedings of IEEE GLOBECOM*.
6. Yoo, T., & Goldsmith, A. (2006). On the optimality of multi-antenna broadcast scheduling using zero-forcing beamforming. *IEEE Journal on Selected Areas in Communications*, 24(3), 528–541.
7. Jin, L., Gu, X., & Hu, Z. (2011). Low-complexity scheduling strategy for wireless multiuser multiple-input multiple-output downlink system. *IET Communications*, 5(7), 990–995.
8. Huang, S., Yin, H., Wu, J., & Leung, Victor C. M. (2012). User selection for multi-user MIMO downlink with zero-forcing beamforming. *IEEE Transactions on Vehicular Technology*, 62(7), 3084–3097.
9. Xie, X., & Zhang, X. (2014). Scalable user selection for multiuser networks. In *Proceedings of IEEE INFOCOM*.
10. Evolved Universal Terrestrial Radio Access (E-UTRA), Downlink Multiple Input Multiple Output (MIMO) enhancement for LTE-Advanced (Release 11). *3GPP TR V11.0.0*.
11. Xie, X., Zhang, X., & Sundaresan, K. (2013). Adaptive feedback compression for MIMO networks. In *Proceedings of ACM MobiCom*.
12. Sen, S., Roy Choudhury, R., & Nelakuditi S. (2011). No time to countdown: Migrating backoff to the frequency domain. In *Proceedings of ACM MobiCom*.
13. Fang, J., Tan, K., Zhang, Y., Chen, S., Shi, L., Zhang, J., et al. (2013). Fine-grained channel access in wireless lan. *IEEE/ACM Transactions on Networking*, 21(3), 772–787.
14. Lee, S., & Kim, C.-K. (2012). D-fi: A diversity-aware wi-fi using an ofdm-based bloom filter. In *Proceedings of IEEE ICNP*.
15. Ettus INC. Universal Software Radio Peripheral. <http://ettus.com>.
16. Kuo, T. W., Lee, K. C., Lin, K. C. J., & Tsai, M. J. (2014). Leader-contention-based user matching for 802.11 multiuser mimo networks. *IEEE Transactions on Wireless Communications*, 13(8), 4389–4400.
17. Zhou, A., Wei, T., Zhang, X., Liu, M., & Li, Z. (2015). Signpost: Scalable mu-mimo signaling with zero csi feedback. In *Proceedings of ACM MohiHoc*.
18. Wireless LAN Medium Access Control and Physical Layer Specification. In *IEEE Std. 802.11n*, 2009.
19. Gesbert, D., & Alouini M.-S. (2004). How much feedback is multi-user diversity really worth? In *Proceedings of IEEE ICC*.
20. Zhang, W., & Letaief, K. B. (2007). MIMO broadcast scheduling with limited feedback. *IEEE Journal on Selected Areas in Communications*, 25(7), 1457–1467.
21. Hassel, V., Gesbert, D., Alouini, M.-S., & Oien, G. E. (2007). A threshold-based channel state feedback algorithm for modern cellular systems. *IEEE Transactions on Wireless Communications*, 6(7), 2422–2426.
22. Tse, D., & Viswanath, P. (2005). *Fundamentals of Wireless Communication*. Cambridge: Cambridge University Press.
23. Halperin, D., Hu, W., Sheth, A., & Wetherall, D. (2010). Predictable 802.11 packet delivery from wireless channel measurements. In *Proceedings of ACM SIGCOMM*.
24. Shen, W.L., Lin, K.C.J., Gollakota, S., & Chen, M.S. (2014). *Rate adaptation for 802.11 multiuser mimo networks* (vol 13, pp. 35–47). IEEE.
25. Yu, H., Bejarano, O., Zhong, L. (2014). Combating inter-cell interference in 802.11 ac-based multi-user mimo networks. In *Proceedings of the 20th annual international conference on Mobile computing and networking* (pp. 141–152). ACM.
26. Yang, Q., Li, X., Yao, H., Fang, J., Tan, K., Hu, W., Zhang, J., & Zhang, Y. (2013). Bigstation: Enabling scalable real-time signal processing in large mu-mimo systems. In *Proceedings of ACM SIGCOMM*.

27. Schmid, T., Sekkat, O., Srivastava, M.B. (2007). An experimental study of network performance impact of increased latency in software defined radios. In *Proceedings of ACM WiNTECH*.
28. Shepard, C., Hang, Y., Anand, N., Li, E., Marzetta, T., & Yang, R., Lin Z. (2012). Argos: Practical many-antenna base stations. In *Proceedings of ACM MobiCom*.
29. Hoydis, J., Ten Brink, S., & Debbah M. (2011). Massive mimo: How many antennas do we need? *CoRR*, abs/1107.1709.
30. Rusek, F., Persson, D., Lau, B.K., Larsson, E.G., Marzetta, T.L., Edfors, O., Tufvesson, F. (2012). Scaling up mimo: Opportunities and challenges with very large arrays. *CoRR*, abs/1201.3210.
31. Zhang, X., Sundaresan, K., Khojastepour, M.A.A., Rangarajan, S., Shin, K.G. (2013). Nemox: Scalable network mimo for wireless networks. In *Proceedings of ACM MobiCom*.
32. Rahul, H.S., Kumar, S., & Katabi, D. (2012). JMB: Scaling wireless capacity with user demands. In *Proceedings of ACM SIGCOMM*.



Kyu-haeng Lee received his B.S. degree in computer science and technology from Tsinghua University, Beijing, China, and Ph.D. in Computer Science and Engineering from Seoul National University in 2009 and 2015, respectively. He has been with Samsung Electronics, as a software engineer since 2015. His research interests include MIMO/OFDM systems, IEEE 802.11, system optimization, and IoT.



Korea, as an assistant professor since 2012. His research interests

include vehicular networks, data center networks, and IEEE 802.11 WLAN.



Chong-kwon Kim received the B.S. degree in industrial engineering from Seoul National University, the M.S. degree in operations research from Georgia Institute of Technology, and the Ph.D. degree in Computer Science from University of Illinois at Urbana-Champaign in 1981, 1982, and 1987, respectively. In 1987, he joined Bellcore as a Member of Technical Staff and worked on Broadband ISDN and ATM. Since 1991, he has been with Seoul National University as a Professor in the School of Computer Science and Engineering. His research interests include wireless and mobile networking, high speed network control, distributed processing, and performance evaluation.