Achieving Better Channel Orthogonality for Improved User Scaling of Multi-user MIMO

Technical Report 2014-06-28, Rice University

Hang Yu, Lin Zhong, and Ashuthosh Sabharwal

Abstract

In an 802.11ac cell[1], an access point (AP) can employ zero-forcing beamforming to simultaneously transmit to multiple clients. However, as the number of clients approaches that of antennas on the AP, the cell capacity often flattens and even drops, exhibiting a *scalability problem*. In this work, we tackle this problem by optimizing the use of multiple antennas available on modern mobile clients for *post-combining*. Our key insight is that post-combining allows a client to use its antennas to create a *virtual channel* from the AP, with much improved *orthogonality* to the virtual channels of other clients. Better channel orthogonality between clients can significantly improve the user scaling of zero-forcing beamforming. To realize this idea, we present an 802.11ac-based solution called MACCO. MACCO leverages the channel sounding in 802.11ac in two ways to optimize the client post-combining in a distributed fashion. First, when a client reports its channel to the AP, MACCO allows it to report the virtual channel based on its locally optimized post-combining. Second, when a client optimizes its post-combining, MACCO allows it to *overhear* the virtual channels reported by other clients in an overhead-free manner, and use them to improve the channel orthogonality. Using an WARP-based implementation, we demonstrate that on average MACCO can increase the capacity of a MU-MIMO cell with eight AP antennas and eight clients by 35%, compared to existing solutions that use client antennas differently.

1. INTRODUCTION

Multi-user multiple-input multiple-output (MU-MIMO) allows an 802.11ac access point (AP) to simultaneously transmit independent data streams to as many clients as its antennas, using interference-free pre-coding schemes such as zero-forcing beamforming (ZFBF). However, the simultaneous transmissions have a scalability problem, a.k.a. *channel hardening* [10]: as the number of clients approaches that of antennas on the AP, the aggregated capacity over all data streams, i.e., the cell capacity, often improves insignificantly [1, 20]. The key reason is that as the AP pre-codes the data streams to nullify the interference between clients, the pre-coded signal vector is often not aligned with the channel vector to the client, producing reduced pre-coding gain and client signal-to-noise-ratio (SNR). As the number of the clients increases, such client SNR reduction may cause trivial improvement of the cell capacity, or even reduce it when the number of AP antennas and clients is relatively large (e.g., eight), and henceforth lead to the capacity scalability problem.

[1]In this work, we use *cell* to denote an infrastructure basic service set (BSS) in 802.11, which consists of an access point and the clients it serves.

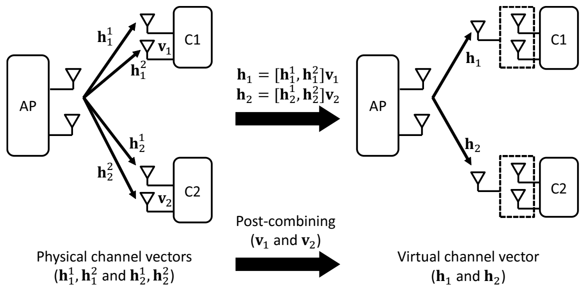


Figure 1: Post-combining by the client creates a virtual channel vector from the AP to the client, which is a weighted combination of the physical channel vectors to the client antennas.

There are three approaches toward solving this problem. The first approach is to let the AP group clients into clusters and only serve the clients in a cluster at a time [9, 19]. While simple, with this approach the capacity gain from MU-MIMO is inevitably limited by the reduced number of clients in a cluster. The second approach is to add many antennas to the AP or connect neighboring APs to form a giant virtual AP [15, 20]. Unfortunately, this approach requires massive modification to the network infrastructure and fine-grained synchronization between the connected APs. The last approach, taken by this work, is to exploit multiple antennas that are often available on modern mobile clients for *post-combining*, a.k.a. receive beamforming. This approach does not mandate changes to the network infrastructure, requires no modification to the ZFBF AP, and can be incrementally implemented on currently existing 802.11ac clients.

By post-combining the signals received at its multiple antennas, a client can use multiple physical channels between the AP and itself to create a *virtual channel*. To see how the virtual channel is created, let us consider the example in Figure 1 where the AP transmits to two clients each with two antennas. The received signal after the post-combining of the client is a weighted combination of the signal received at each antenna, whereas the weights are determined by the post-combining vector. Therefore, the signal can be considered being sent on a virtual channel from the antennas on the AP to a *virtual antenna* on each client. Given that the post-combining vectors of the clients are determined, the AP can perform pre-coding simply based on their virtual channels. By creating the virtual channel, post-combining improves the client SNR in two ways. First, by coherently combining the signals from multiple antennas, post-combining is able to produce a stronger virtual channel with less attenuation. This technique is known as *maximum ratio combining* (MRC). More importantly, post-combining can properly adjust the virtual channel to make it more orthogonal to the virtual channels of other clients. Such *channel orthogonality* is actually the key to make the capacity of MU-MIMO more scalable to the number of clients. Existing solutions that follow this approach [4, 6, 12, 21] remain theoretical and do not consider the channel orthogonality to optimize the post-combining vectors.

The key contribution of this work is a practical, 802.11ac-compliant, overhead-free solution that follows the last approach described above, but is much more effective toward the capacity scalability improvement by considering channel orthogonality between clients. Our solution, called MACCO (Multi-Antenna Clients that acquire Channel knowledge via Overhearing), is motivated by the explicit channel sounding process used by an 802.11ac AP to acquire the downlink channels from all clients. In the channel sounding, the AP sends a sounding frame to all the clients and then the clients report their channels one by one. MACCO exploits this process in two parts to let a client optimize its post-combining. First, when a client responds to the sounding from the AP, instead of reporting its physical channels, it reports the virtual channel that is locally calculated based on the optimized post-combining. Second, other clients that have not yet reported their channels overhear the reported virtual channel from the client and use it to optimize their post-combining with improved channel orthogonality. Because of the distributed nature of the post-combining optimization in MACCO, the clients do not need to report the lengthy physical channels to the AP and the AP does not have to send the centrally optimized post-combining vectors back to the clients, both of which would otherwise incur considerable overhead [26]. Since explicit channel reporting by clients is mandated by 802.11ac, MACCO does not incur any overhead by leveraging the "free" channel information broadcast on the wireless medium.

We report a prototype realization of MACCO using the WARP platform [17], and evaluate its performance in real-world indoor environments. Our experimental results indicate that compared to existing solutions, on average MACCO can boost the capacity by 35% for a MU-MIMO cell with eight AP antennas and eight clients. While the capacity improvement is modest, with only four antennas on each client, MACCO can near-proportionally increase the capacity as the MU-MIMO AP serves more clients, for a majority of SNR regimes and cell topologies.

It is necessary to emphasize that while we target the scalability problem to the number of clients in a congested MU-MIMO cell, this problem also applies to the number of streams on a single client if the AP has fewer clients to serve and delivers more than one stream to the client. MACCO is equally effective for the scalability problem to the number of streams, given that the client has extra antennas for post-combining. We elaborate this issue in Section 4.1.

2. BACKGROUND

We first provide background regarding MU-MIMO and 802.11ac.

2.1 MU-MIMO Primer

MU-MIMO allows an AP with M antennas to simultaneously transmit independent data streams to up to K=M clients. To achieve this, the AP must *pre-code* the data signals to the clients, i.e., by multiplying a complex scalar to the signal sent from each antenna. The set of scalars is denoted as the *pre-coding vector*. The pre-coding vector is able to adjust the magnitude and phase of the transmitted signals. When the client is equipped with multiple antennas, it can similarly apply a *post-combining vector* to the signals received at the antennas.

Pre-coding and post-combining can be either linear or non-linear. Capacity-achieving schemes, such as dirty-paper coding, are non-linear, and prohibitively expensive to implement in practice [7, 25]. As a result, similarly to other experimental work on MU-MIMO [1, 13, 20], in this work we focus on linear pre-coding schemes. To nullify the interference between served clients, denoted as intra-cell interference, the AP leverages *zero-forcing beamforming* (ZFBF) to pre-code the data signals. In ZFBF, the pre-coding vector for one client, $\mathbf{w}_k$, must be orthogonal to the channel vectors of other clients, i.e., $\mathbf{w}_k$=$\mathbf{H}_O^\perp$ where $\mathbf{H}_O = [\mathbf{h}_1, \cdots, \mathbf{h}_{k-1}, \mathbf{h}_{k+1}, \cdots, \mathbf{h}_K]$ and $\mathbf{h}_k$ is the channel vector from the AP to the kth client. ZFBF is known to be suboptimal when K approaches M, i.e., the cell capacity may increase slowly or even drop as more clients are served [20]. To serve a single client, the AP can leverage *conjugate beamforming* (ConjBF) to maximize the client SNR, by choosing $\mathbf{w}_k$=$\mathbf{h}_k^*$. When the client has multiple antennas, the channel vector $\mathbf{h}_k$ becomes a matrix, denoted as $\mathbf{H}_k$, and the AP employs *eigen beamforming* (EigenBF) to maximize the client SNR, by choosing $\mathbf{w}_k$=$\mathbf{v}_{max}(\mathbf{H}_k^*\mathbf{H}_k)$ where $\mathbf{v}_{max}()$ is the eigenvector of a matrix corresponding to its largest eigenvalue. To cope with EigenBF by the AP, the client sets its post-combining vector, $\mathbf{v}_k$, as $\mathbf{v}_k$=$\mathbf{H}_k\mathbf{v}_{max}(\mathbf{H}_k^*\mathbf{H}_k)$ to maximize its SNR. This post-combining technique is known as *maximum ratio combining* (MRC). All the pre-coding schemes require full channel knowledge to all clients by the AP, which the AP acquires through channel sounding (see

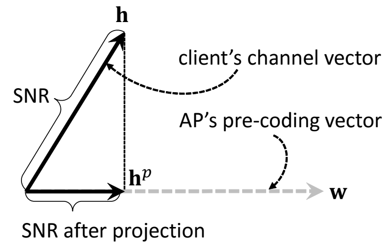


Figure 2: Representation of a channel vector and a pre-coding vector. Employing the pre-coding vector w by the AP projects the channel vector h to the direction of w, leading to a reduced SNR at the client.

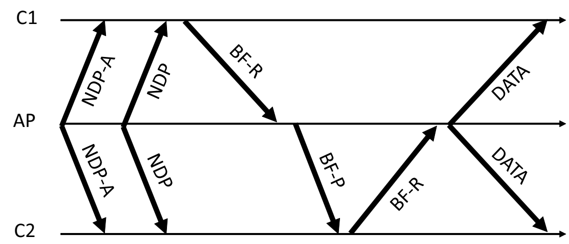


Figure 3: Channel sounding in 802.11ac. The AP first sends a NDP-A frame to initiate the sounding process, and then polls each client except the first one for their estimated channels contained in the BF-R frame.

Section 2.3).

2.2 Channel Orthogonality for MU-MIMO

We next discuss an important metric for the performance of MU-MIMO, the *channel orthogonality* between clients. Linear pre-coding by the AP can be considered projecting the client's channel vector $\mathbf{h}$ to the direction of the AP's pre-coding vector $\mathbf{w}$. As a result, to the client, the resultant channel on which it receives its signal is actually the projected channel, $\mathbf{h}^p$, as shown in Figure 2:

$$\mathbf{h}^p = \mathbf{h} \cdot \cos\langle \mathbf{h}, \mathbf{w} \rangle,$$

where "$\langle \cdot \rangle$" refers to the angle between two vectors. Clearly, when $\mathbf{w}$ is aligned with $\mathbf{h}$, the maximum channel SNR is preserved; when $\mathbf{w}$ is perpendicular to $\mathbf{h}$, the signal is completely nullified. Such relationship can be extended to that between two channel vectors, which we define as the channel orthogonality. For two channel vectors, $\mathbf{h}_1$ and $\mathbf{h}_2$, their orthogonality can be represented by the angle between them:

$$\theta = \arccos \frac{\mathbf{h}_1 \cdot \mathbf{h}_2}{|\mathbf{h}_1||\mathbf{h}_2|}.$$

The channel orthogonality between clients is critical for the capacity of MU-MIMO under ZFBF. Recall that to nullify the interference between two clients, the AP must set $\mathbf{w}_1$=$\mathbf{h}_2^{\perp}$ and $\mathbf{w}_2$=$\mathbf{h}_1^{\perp}$. If $\mathbf{h}_1$ and $\mathbf{h}_2$ are perfectly orthogonal, the maximum channel SNR for both clients is retained and the cell capacity can be linearly scaled up. Otherwise, as the AP serves both clients, the capacity of each client decreases, leading to the capacity scalability problem.

2.3 MU-MIMO Operation in 802.11ac

We next briefly introduce the MU-MIMO operation in 802.11ac [3]. While the pre-coding scheme for the AP is not specified by the protocol and left to the vendor, we anticipate that only linear pre-coding schemes can be feasibly adopted by practical 802.11ac APs. Therefore, in this work we assume the 802.11ac AP performs ZFBF for interference nullification, and ConjBF or EigenBF for SNR maximization.

An 802.11ac AP needs to know the downlink channels to all the clients to calculate the pre-coding vectors. 802.11ac defines an explicit channel estimation and feedback scheme to let the AP acquire such downlink channels. This process is commonly known as *channel sounding* [3]. Figure 3 depicts the frame exchange in the channel sounding process. First, the AP transmits a *Null Data Packet Announcement* (NDP-A) frame to notify the set of clients to be concurrently served in the incoming simultaneous transmission, and to request those clients to prepare their *Beamforming Report* (BF-R) frames that contain the channel estimates. After the NDP-A frame, the AP sends a *Null Data Packet* (NDP) frame containing the training sequences required by the client to compute the channel estimates. Upon reception of the NDP frame, the clients begin replying with their BF-R frames. The AP relies on a *Beamforming Report Poll* (BR-P) frame to request this channel feedback from each client, except the first one, who has a pre-allocated slot at the beginning as indicated by the AP in the NDP-A frame.

3. OVERVIEW OF MACCO

To improve the capacity scalability of a MU-MIMO cell to the number of concurrently served clients, we present an 802.11ac-based protocol called MACCO. The key idea in MACCO is to leverage the channel sounding process in 802.11ac to let a client optimize its post-combining to improve the channel orthogonality in a distributed way. To realize this idea, MACCO features two designs: *local post-combining optimization* and *implicit channel acquisition*.

3.1 Local Post-combining Optimization

MACCO allows a client to locally optimize its post-combining vector and report the corresponding virtual channel to the AP. This is motivated by the observation that in the channel sounding process of 802.11ac, the AP asks the clients to explicitly report the estimated channels from the AP to them. We leverage this to let a client report a virtual channel that is created by post-combining, instead of the physical channels. According to Section 2, the signal after the client's post-combining is a weighted combination of the signal received at each client antenna, whereas the weights constitute the post-combining vector. The resultant combined signal can be considered being received at one single virtual antenna at the client. Therefore, the channel vector from the antennas on the AP to the virtual antenna on the client, i.e., the virtual channel, is a combination of the client's physical channel vectors and post-combining vector. By reporting the virtual channel, the client hides from the AP how its antennas are used for post-combining.

Why Distributed Optimization?

It is natural to ask: what are the benefits of such distributed post-combining optimization done by the clients locally? Notably, an alternative design achieving the same performance can be a centralized optimization where every client reports its physical channels to the AP, the AP optimizes the post-combining vectors for all the clients, and send them back to the latter. However, such centralized optimization have two limitations that are critical in practice.

First, the centralized optimization requires the clients to report more channel information to the AP, and the AP to deliver the optimized post-combining vectors to the clients, both of which consume additional channel air time as overhead. In contrast, MACCO clients only need to report their virtual channels and the MACCO AP does not need to deliver the optimization results at all. Observe that the physical channels are multiple vectors that correspond to the client antennas. The virtual channel, on the other hand, is a single vector irrespective of the number of client antennas. Therefore, reporting the physical channels would amplify the overhead from BF-R frames which is already substantial [26]. When the client uses N antennas for post-combining, the size of the BF-R frame is N times larger. Moreover, the AP must send a frame to each client to deliver the optimized post-combining vector. Such process is also known as "feedforward" [4]. Similar to feedback from clients, feedforward is expensive in 802.11ac since the frame must be sent at base rate (6 Mbps) for reliability concerns. Adding a feedforward frame from the AP to each client can aggregately double the duration of the 802.11ac channel sounding process. It is important to note that the AP cannot "implicitly" deliver the post-combining vectors to the clients without using feedforward frames. A naive but incorrect design would let the AP encode the data frames in a way that allows each client to infer the optimal post-combining vector using MRC. However, the client cannot apply MRC to the data frame which is subject to inter-client interference; to remove the interference, the client must know the optimal post-combining vector beforehand.

Second, the centralized optimization requires modification to the AP, i.e., the AP must optimize each client's post-combining vector and feedforward the optimization results to the clients. MACCO, on the contrary, works with unmodified 802.11ac APs employing ZFBF and 802.11ac clients. Such protocol compatibility of MACCO is an important benefit that allows rapid, incremental deployment of MACCO clients in current and future 802.11ac networks.

3.2 Implicit Channel Acquisition

Table 1: Denotations used in the paper.

Symbol	Definition
M	Number of antennas on the AP
K	Number of simultaneously served clients
N	Number of antennas on the client
$\mathbf{h}_k$	Virtual channel vector of the kth client
$\mathbf{h}_k^n$	Physical channel vector of the kth client's nth antenna
$\mathbf{h}_{k,j}$	The jth virtual channel vector of the kth client
$\mathbf{w}_k$	Pre-coding vector of the AP for the kth client
$\mathbf{v}_k$	Post-combining vector of the kth client
$\mathbf{H}_k$	Physical channel matrix from the AP to the kth client

MACCO allows a client to acquire the knowledge of other clients' optimized virtual channels through overhearing their BF-R frames. As we have shown in Section 2 and **??**, poor channel orthogonality is the major factor that degrades the client SNR and capacity, which the client cannot improve without knowing the channels of other clients. Fortunately, during the channel sounding process of 802.11ac, a client can overhear the BF-R frames from other clients that reported their channels at an earlier time. Such BF-R frames contain the information about the optimized virtual channel of the client. Since the wireless medium is shared by all clients, and the BF-R frame with a known format is not encrypted, a client can easily decode the BF-R frame from other clients and acquire the virtual channel in it. Such channel knowledge is considered "free" for the MACCO client to obtain, since the BF-R frame is mandated by 802.11ac and originally intended to the AP.

It is noticed that such overhearing in MACCO only offers partial virtual channel knowledge to each client, since the client must optimize its post-combining vector right before it reports the virtual channel. Nevertheless, such knowledge is sufficient to significantly improve the capacity scalability of MU-MIMO as we will show in Section 5. We acknowledge that having full channel knowledge by each client may provide further capacity improvement. However, to the best of our knowledge, how to jointly optimize the post-combining vectors of all clients with full channel knowledge remains an challenging and open problem even for theoretical investigation.

4. DISTRIBUTED POST-COMBINING OPTIMIZATION

We next discuss how a MACCO client leverages the overheard virtual channels to optimize its post-combining vector. Toward this, we provide a closed-form solution to the optimal post-combining vectors that maximize the client's capacity, given by the following equation:

$$\mathbf{v}_{k,j} = \mathbf{H}_k \mathbf{H}_L^{\perp} \mathbf{v}_{jmax}((\mathbf{H}_k \mathbf{H}_L^{\perp})^* (\mathbf{H}_k \mathbf{H}_L^{\perp})), \qquad (1)$$

where the denotations are provided in Table 1 and Section 4.4. We will derive this result by analyzing three representative cases.

4.1 Cases 1: M=2 and K=2

We first study a simple case where an AP with two antennas serves two clients (C1 and C2) each with two antennas. To see how the multiple antennas can benefit the client, let us first assume both C1 and C2 employ antenna selection and only use a single antenna, with the corresponding channel vector $\mathbf{h}_1$ and $\mathbf{h}_2$. We denote the precoding vector for C1 and C2 employed by the AP as $\mathbf{w}_1$ and $\mathbf{w}_2$. Then, the SNR of C1 and C2 normalized to its noise power can be expressed as

$$\rho_1 = |\mathbf{w}_1 \cdot \mathbf{h}_1|^2 = |\mathbf{h}_1|^2 \cos^2\langle \mathbf{w}_1, \mathbf{h}_1 \rangle,$$
$$\rho_2 = |\mathbf{w}_2 \cdot \mathbf{h}_2|^2 = |\mathbf{h}_2|^2 \cos^2\langle \mathbf{w}_2, \mathbf{h}_2 \rangle,$$

where $|\mathbf{w}_1|^2$=$|\mathbf{w}_2|^2$=1 due to power normalization. To nullify the interference between C1 and C2 with ZFBF, the AP must employ $\mathbf{w}_1$=$\mathbf{h}_2^\perp$ and $\mathbf{w}_2$=$\mathbf{h}_1^\perp$. Such interference nullification reduces ρ_1 and ρ_2, since $\mathbf{w}_1$ and $\mathbf{w}_2$ are usually not aligned with $\mathbf{h}_1$ and $\mathbf{h}_2$.

When C1 and C2 use both antennas, they can be leveraged to post-combine the received signals. Visually, such post-combining "stretches" and "steers" the physical channel vectors to create a virtual channel vector. For clarity, when the clients use more than one antenna, we use $\mathbf{h}_k^n$ to denote the physical channel vectors of the nth antenna at the kth client, and $\mathbf{h}_k$ the virtual channel vector of the kth client.

Before proceeding to discuss the post-combining optimization by C1 and C2, we need to clarify that in this case the same number of streams can be achieved by letting the AP only serve a single client, say C1, with single-user MIMO (SU-MIMO). However, SU-MIMO similarly has a capacity scalability problem to the number of streams, since the SNR of each stream is reduced when C1 decodes the streams. Such SNR reduction depends on the orthogonality between the two physical channel vectors $\mathbf{h}_1^1$ and $\mathbf{h}_1^2$. This is also known as the "antenna correlation" that reduces the capacity of SU-MIMO. Without extra antennas, C1 cannot improve the channel orthogonality between the two streams. Therefore, we argue that the AP should employ MU-MIMO to serve both clients instead of one of them with two streams, in order to leverage the post-combining gain from the client antennas. Such argument is similarly proposed in [5] to explore the client diversity.

Optimization by C1. We first discuss the optimization performed by C1 and assume it first reports the virtual channel to the AP without the knowledge of $\mathbf{h}_2$. Therefore, C1 optimizes its post-combining vector, $\mathbf{v}_1$, to maximize its SNR solely based on its physical channel vector $\mathbf{h}_1^1$ and $\mathbf{h}_1^2$. To achieve this, C1 employs MRC:

$$\mathbf{v}_1 = \mathbf{H}_1 \mathbf{v}_{max}(\mathbf{H}_1^* \mathbf{H}_1),$$

where $\mathbf{H}_1$=$[\mathbf{h}_1^1, \mathbf{h}_1^2]^\mathrm{T}$. The resultant virtual channel vector, $\mathbf{h}_1$, is a weighted combination of $\mathbf{h}_1^1$ and $\mathbf{h}_1^2$ with maximum magnitude, as shown in Figure 4. From the figure, observe that the post-combining gain from MRC creates a virtual channel that is stronger than either of the two physical channels. However, the channel orthogonality between C1 and C2 may not necessarily be improved since C1 is unaware of the direction of C2's virtual channel. Nonetheless, given its limited channel knowledge, C1 has to employ MRC for its post-combining, wishing a resultant SNR improvement.

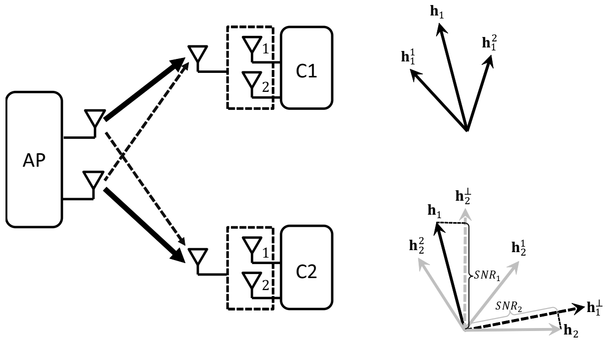


Figure 4: Post-combining optimization by the clients C1 and C2 in Case 1. Depending on the channel knowledge, the client maximizes its SNR before or after the signal projection by the ZFBF AP.

Optimization by C2. Now let us switch the focus to C2 and assume it optimizes its post-combining vector and reports to the AP after C1. As a result, C2 can overhear the BF-R frame from C1, and acquire the knowledge of the virtual channel $\mathbf{h}_1$. Then, C2 knows the direction that the AP will project its signal onto, $\mathbf{h}_1^\perp$, and can maximize its SNR after such projection. To do this, C2 employs:

$$\mathbf{v}_2 = (\mathbf{h}_1^\perp \mathbf{H}_2)^*,$$

where $\mathbf{H}_2$=$[\mathbf{h}_2^1, \mathbf{h}_2^2]^\mathrm{T}$. As shown in Figure 4, the resultant virtual channel vector, $\mathbf{h}_2$, is more orthogonal to $\mathbf{h}_1$, even though its magnitude is not necessarily maximized. However, after the AP's signal projection, $\mathbf{h}_2$ is guaranteed to have the maximum magnitude and the SNR of C2 is maximized thanks to the knowledge of $\mathbf{h}_1$. In addition, we can see the SNR increase of C1 due to improved channel orthogonality.

To summarize, in this case where M=2 and K=2, a client can leverage post-combining to adjust its virtual channel vector such that its SNR is maximized before or after the AP's signal projection, depending on whether the client has the knowledge of the other's virtual channel.

4.2 Cases 2: M=3 and K=3

Now let us consider the case where an AP with three antennas serves three clients, each with two antennas. Consequently, the physical or virtual channel vectors reside in a three-dimensional space. Notice that compared to the previous case we have not changed the number of antennas on the clients, N, since its impact on the post-combining optimization is trivial as we will show in the following. We similarly assume C1, C2 and C3 successively send their BF-R frames to the AP to report their optimized virtual channels.

Optimization by C1. For C1 who does not know $\mathbf{h}_2$ and $\mathbf{h}_3$, the optimization is similar to that in the previous case, i.e., $\mathbf{v}_1$=$\mathbf{H}_1\mathbf{v}_{max}(\mathbf{H}_1^*\mathbf{H}_1)$ whereas N only changes the dimension of $\mathbf{H}_1$.

Optimization by C2. Similar to the previous case, C2 has the knowledge of $\mathbf{h}_1$ by overhearing the BF-R frame from C1. However, since the channel vectors are in the three-dimensional space, $\mathbf{h}_1^\perp$ becomes a two-dimensional plane, represented by a 2 by 2 matrix instead of a vector. Therefore, without knowing $\mathbf{h}_3$, C2 should maximize the magnitude of the projection of $\mathbf{h}_2$ onto $\mathbf{h}_1^\perp$. The resultant optimal post-combining vector is given by

$$\mathbf{v}_2 = \mathbf{H}_2\mathbf{h}_1^\perp\mathbf{v}_{max}((\mathbf{H}_2\mathbf{h}_1^\perp)^*(\mathbf{H}_2\mathbf{h}_1^\perp)), \qquad (2)$$

which we prove in the Appendix.

Optimization by C3. For C3, its signal will be projected by the AP onto a space that is perpendicular to both $\mathbf{h}_1$ and $\mathbf{h}_2$. As a result, extending Equation 2 we have the optimal post-combining vector employed by C3:

$$\mathbf{v}_3 = \mathbf{H}_3[\mathbf{h}_1,\mathbf{h}_2]^\perp\mathbf{v}_{max}((\mathbf{H}_3[\mathbf{h}_1,\mathbf{h}_2]^\perp)^*(\mathbf{H}_2[\mathbf{h}_1,\mathbf{h}_2]^\perp)). \qquad (3)$$

4.3 Cases 3: M=3 and K=2

In the previous two cases we have assumed a congested MU-MIMO cell where K=M, so that the AP serves M clients each with a single stream. As a result, each client uses all antennas to post-combine the signals of the stream, and maximize its SNR which uniquely determines its capacity. While in this work our main focus is on such congested MU-MIMO cells, for the general applicability of MACCO we next discuss the case where $M>K$ so that AP can send more than one streams to certain client. Note, the client needs equal number of antennas to receive multiple streams from the AP. However, if extra antennas are available, MACCO can leverage them to improve the channel orthogonality between streams. To show this, we assume N=4 such that the client can apply MACCO to up to two streams. We suppose the AP transmits one stream to C1, and two streams to C2.

Optimization by C1. Since C1 receives a single stream from the AP, its post-combining optimization does not differ from that in the previous two cases where $\mathbf{v}_1$=$\mathbf{H}_1\mathbf{v}_{max}(\mathbf{H}_1^*\mathbf{H}_1)$.

Optimization by C2. For C2, since the AP delivers two streams to it, it must report two virtual channel vectors to the AP, denoted as $\mathbf{h}_{2,1}$ and $\mathbf{h}_{2,2}$. Naturally, C2 should jointly optimize $\mathbf{h}_{2,1}$ and $\mathbf{h}_{2,2}$, based on the knowledge of $\mathbf{h}_1$ overheard from C1. The optimization can be extended from the previous case with a single stream from the AP: when there are two streams, they should be sent on the best two eigen channels. As a result, the optimal post-combining vectors for C2 are given by

$$\mathbf{v}_{2,1} = \mathbf{H}_2\mathbf{h}_1^\perp\mathbf{v}_{max}((\mathbf{H}_2\mathbf{h}_1^\perp)^*(\mathbf{H}_2\mathbf{h}_1^\perp)),$$
$$\mathbf{v}_{2,2} = \mathbf{H}_2\mathbf{h}_1^\perp\mathbf{v}_{smax}((\mathbf{H}_2\mathbf{h}_1^\perp)^*(\mathbf{H}_2\mathbf{h}_1^\perp)),$$

where $\mathbf{v}_{smax}()$ is the eigenvector of a matrix that corresponds to its second largest eigenvalue. To decode the two streams, C2 individually post-combines the received signals with $\mathbf{v}_{2,1}$ and $\mathbf{v}_{2,2}$, to eliminate the interference between them.

4.4 General Case

The results in the previous three cases can be generalized to cases with arbitrary M and K where $M \geq K$. For the kth client, let use denote the set of known channel vectors of other clients as $\mathbf{h}_1, \mathbf{h}_2, \cdots, \mathbf{h}_l$, and its own channel matrix as $\mathbf{H}_k$. Then, the set of optimal post-combining vectors are given by:

$$\mathbf{v}_{k,j} = \mathbf{H}_k\mathbf{H}_L^\perp\mathbf{v}_{jmax}((\mathbf{H}_k\mathbf{H}_L^\perp)^*(\mathbf{H}_k\mathbf{H}_L^\perp)),$$

where $\mathbf{H}_L = [\mathbf{h}_1, \mathbf{h}_2, \cdots, \mathbf{h}_l]$ and $\mathbf{v}_{jmax}()$ stands for the eigenvector of the matrix corresponding to its jth largest eigenvalue.

4.5 Computational Complexity

The computation involved in the above distributed post-combining optimization (Equation 1) can be modularized with polynomial complexity. Calculating the optimal set of post-combining and virtual channel vectors include the following operations: (*i*) calculating the conjugate of a matrix; (*ii*) calculating the eigenvectors and eigenvalues of a matrix; (*iii*) inverting/pseudo-inverting a matrix; (*iv*) multiplying two matrices. All of them can be accomplished by the following two modules that already have efficient hardware implementation: (*i*) matrix singular value decomposition (SVD); (*ii*) matrix multiplication. In addition, the complexity of Equation 1 is given by $O(M^2L) + O(MNL) + O(N^2M) + O(NL)$, where N, L, and M are all small numbers (e.g., $N \leq 4$, $L<8$ and $M \leq 8$). Therefore, we suspect that the computation in MACCO can be affordable by modern clients.

5. EVALUATION

In this section valuate MACCOwith both experiments and simulations.

5.1 WARP Implementation

We implement MACCO on WARP [17], a flexible FPGA-based software-defined radio (SDR) platform. Our implementation is based on WARPLab [16] with extensive modifications. WARPLab is a framework that combines the advantages of FPGA and MATLAB to enable rapid physical layer prototyping that allows coordination of multiple transmit and receive boards. The extensible framework provides great flexibility to develop physical layer techniques, which is perfect for MU-MIMO system designs. In WARPLab, one or more WARP boards as APs or clients are connected to and controlled by a central computer running MATLAB, through gigabytes Ethernet cables. The baseband processing can be flexibly implemented in either hardware (FPGA on the WARP boards) or software (MATLAB on the central computer).

Given the WARPLab framework, we need solve two challenges to implement MACCO.

Realtime vs. Flexibility. The first challenge is the balance between the realtime capability and flexibility of our implementation. To make it fully realtime, MACCO has to be completely realized in hardware, which is prohibitively complex and inflexible. On the other hand, realizing all processing in software is simpler and more flexible, but incurs significant communication round-trip time (RTT) between the central computer and the WARP boards as the transfer between them needs to be raw baseband samples.

To achieve a good tradeoff, we implement standard MU-MIMO processing including CSI estimation, pre-coding, and post-combining in hardware, while the calculation of the pre-coding vector, post-combining vector, and virtual channel vector in software. This way, the relatively complex matrix operations in MACCO can be easily done in MATLAB, while hardware-friendly MU-MIMO operations are pushed to FPGA. The transfer between the central computer and WARP boards is reduced from raw baseband samples to the calculated pre-coding, post-combining and virtual channel vectors.

Delay of over-the-air frames. The second challenge is the substantial delay involved in sending a frame over the air. This is again due to the slow transfer between the central computer and the WARP boards. Even though we have largely reduced the size of such transfer by implementing MU-MIMO operations in hardware, the constant latency of forming the Ethernet packet in MATLAB, polling the Ethernet interface on the board, and fetching the transferred data into the board buffer is still too long to satisfy the timing constraints of 802.11ac. Our measurement shows that when a client receives the BF-P frame from the AP, it usually takes a few milliseconds for the client to start sending the BF-R frame. The aggregated delay from multiple BF-P and BF-R frames will make the channel sounding process much longer, leading to an potentially outdated wireless channel.

To tackle this, we emulate the process of channel reporting and overhearing in MACCO: when all client boards receive the NDP frame sent from the AP boards over the air, they simply transfer the estimated physical channels to the central computer via Ethernet, and then the BF-P and BF-R frames are emulated on the central computer, providing proper channel knowledge to each client. After the pre-coding and post-combining vectors are calculated, they are sent back to the AP and client boards, and then the MU-MIMO transmission from the AP follows. We claim that such emulation does not lose authenticity due to two reasons. First, since the 802.11ac channel sounding is explicit, the channel information in the BF-R frame is entirely delivered when the frame is successfully decoded. Second, the BF-P and BF-R frames are considered control frames in 802.11ac and required to be sent with base rate (6 Mbps). Therefore,

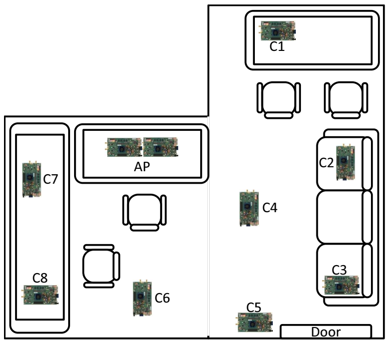


Figure 5: Our experimental setup in a lab room for the evaluation of MACCO. An AP with eight antennas simultaneously serves up to eight MACCO clients each with four antennas.

the likelihood of not capable of decoding them by a client is fairly small. Emulating the BF-P and BF-R frames can greatly reduce the duration of the channel sounding process, allowing us to involve more (up to K=8) clients in the experiment.

Measurement Setup

We deploy a single MU-MIMO cell including one AP with eight antennas and up to eight clients, within a room in a campus building shown in Figure 5. To conduct multiple groups of experiments to average out the performance, we change the locations of the AP and clients to create different wireless channels between them. We conduct the experiments at night, on a clean Wi-Fi channel (# 14) to make sure the clients are not subject to external interference. We measure the client SNR and INR to calculate the Shannon capacity of each client and the cell.

5.2 Results

We next report our evaluation results. We focus on evaluating the effectiveness of MACCO for improving the capacity scalability of a MU-MIMO cell, and investigate the impact from various parameters such as K, M, and N.

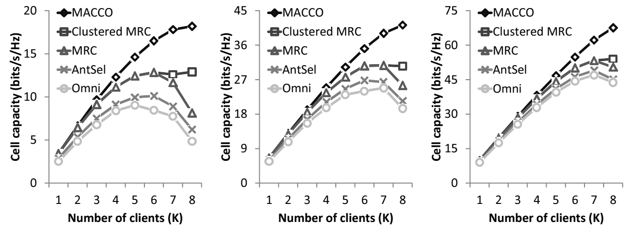
We compare MACCO to the following alternative approaches to use client antennas: (*i*) Omni in which a client only has a single antenna; (*ii*) Antenna Selection (AntSel) in which a client picks the antenna with the best physical channel; (*iii*) MRC in which a client optimizes its post-combining vector to maximize its SNR based on its own physical channels; (*iv*) Clustered MRC in which the clients employ MRC but the AP only serves the best set of clients as a cluster at a time and schedules clusters in a TDMA fashion. There are two notes regarding the alternative approaches for comparison. First, in our evaluation we only approximate the capacity of clustered MRC, assuming no overhead from clustering and scheduling the clients by the AP. However, in practice serving more than one clusters of clients in a TDMA fashion requires extra control frames such as the NDP-A and NDP frames in the channel sounding process. Second, AntSel, MRC and Clustered MRC do not seek to leverage the channel knowledge of other clients to optimize the client post-combining, and thus cannot improve the channel orthogonality. However, comparing MACCO with these approaches is fair, since MACCO acquires channel knowledge in an overhead-free manner by overhearing the BF-R frames that are necessary in the 802.11ac channel sounding and destined to the AP.

5.2.1 *Capacity Scalability Improvement*

To see how MACCO makes the MU-MIMO capacity more scalable to the number of clients, Figure 6 shows the capacity of a MU-MIMO cell where the AP employs M=8 antennas and serves K=$1, 2, \cdots, 8$ clients. We have purposely chosen three regimes with low, medium, and high average SNR at the clients, by adjusting the transmit power from the AP. The average SNR with respect to a single client antenna are 1.47 dB, 11.02 dB, and 19.43 dB for the low, medium, and high SNR regimes, respectively. We obtain the following important observations from Figure 6.

- With Omni, AntSel and MRC, the cell capacity starts to drop as K approaches M. This is due to significantly reduced client SNR from ZFBF as we explained in Section 2.
- When the client employs AntSel instead of Omni, multiple antennas provide trivial capacity improvement. This is because AntSel only offers an antenna diversity gain, which is quite limited due to the physical proximity of the antennas on a single client.
- MRC achieves better cell capacity than AntSel does, since MRC allows the client to pick the best eigen channel instead of the best physical channel.
- Clustered MRC achieves similar performance as MRC does, but prevents the capacity from dropping as K approaches M. This is because the AP does not add more clients to serve as the capacity starts to decrease due to client SNR reduction.
- MACCO outperforms alternative approaches by more effectively using the multiple client antennas based on the overheard channel knowledge. It works most effectively when K approaches M, making the capacity near-proportional to the number of clients. This confirms our channel characterization results in Section **??**, indicating the crucialness of channel orthogonality to determine the capacity scalability of MU-MIMO.
- MACCO provides most capacity improvement in the low SNR regime, where the capacity near-linearly grows with SNR. This means MACCO can provide good performance to clients that are on the edge of the cell.

Larger-scale MU-MIMO cells beyond 802.11ac. While 802.11ac APs are limited to be equipped with up to M=8 antennas, we suspect the applicability of MACCO is not restricted to cells of such scale. To investigate the effectiveness of MACCO beyond current 802.11ac, we seek to evaluate it under MU-MIMO cells of even larger scale, e.g., with 16 and 32 AP antennas and clients. Unfortunately, we find it very difficult to perform experiments of this scale. While building an AP with many antennas is possible [20], properly deploying so many multi-antenna clients is hardly feasible since they need to be distributed and connected to the central computer using Ethernet cables. Therefore, we rely on simulation based on synthetic channels to evaluate the performance of MACCO in large-scale MU-MIMO cells. We adopt the 802.11n/ac MIMO channel model based on the *Ricean* K factor [8] to generate MU-MIMO channel traces, assuming an i.i.d. MIMO channel from the AP to each client. To mimic a realistic indoor environment, in the model we choose the *Ricean* factor K as 1 which offers the MIMO channel both line-of-sight (LOS) and non-line-of-sight (LOS) components.



(a) Low SNR (b) Medium SNR (c) High SNR

Figure 6: Cell capacity achieved by Omni, AntSel, MRC, Clustered MRC and MACCO as function of the number of clients, K, with M=8 antennas on the AP. MACCO can near-proportionally increase the cell capacity while others cannot.

Figure 7 shows the cell capacity achieved by MACCO and alternative approaches under synthetic channels for the large-scale MU-MIMO cell. We first observe that MACCO is still able to considerably improve the capacity scalability. However, the capacity sub-linearity of MACCO becomes more obvious as K approaches M. The key reason is that as more clients are served, it becomes harder to improve the orthogonality between a greater number of channels. In fact, the capacity scalability issue we tackle in this work is intrinsically determined by the physical channel vectors of the clients. Even though MACCO can improve the channel orthogonality, the resultant virtual channel can still suffer from certain SNR loss. Theoretically, MACCO is able to achieve perfect capacity scalability only if each client is equipped with a very large number of antennas [11], which is apparently impractical.

5.2.2 *The Impact of Antenna Numbers*

We next study the impact of two important parameters on the effectiveness of MACCO.

Number of AP antennas. We first study the capacity gain from MACCO compared to Omni and MRC with different

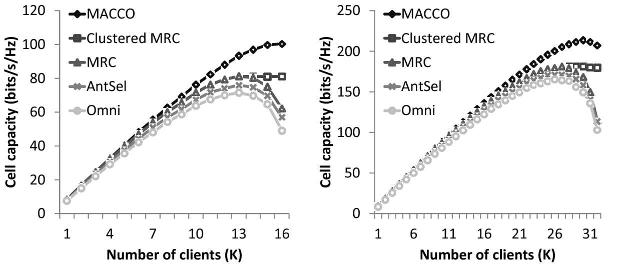


Figure 7: (Simulated) capacity achieved by MACCO and alternative approaches for even larger-scale MU-MIMO cells: (Left) M=K=16 and (Right) M=K=32.

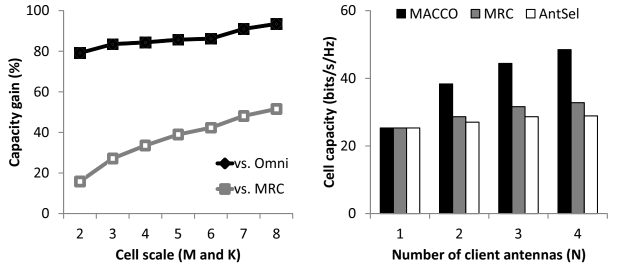


Figure 8: (Left) Cell capacity gain from MACCO with different cell scales (M and K). (Right) Cell capacity achieved by AntSel, MRC, and MACCO with different number of client antennas (N).

number of antennas on the AP (M), as shown in Figure 8 (Left). We assume a congested cell where K=M to investigate the full potential of MACCO. It is seen that as M and K increases, the MU-MIMO cell enjoys more capacity improvement from MACCO. This is because the scalability issue becomes more problematic as the AP is able to serve more clients. Moreover, the capacity gain increase compared to MRC is more obvious, due to the fact that MRC becomes less effective with more clients without improving the channel orthogonality between them.

Number of client antennas. We then study how the number of antennas on the clients, N, affects the capacity gain from MACCO. We use M=K=8 as the example, and show the results in Figure 8 (Right). One can see that with more client antennas, MACCO, MRC and AntSel can all improve the cell capacity. However, the improvement from MRC and AntSel are very limited; on the contrary, MACCO is more capable of increasing the capacity from more client antennas. This is because MACCO can more effectively leverage the precious degrees of freedom provided by the client antennas, by using them to improve the channel orthogonality between clients.

5.2.3 *Client Order does Not Matter*

When the MACCO clients successively send their BF-R frames, they acquire different amount of channel knowledge through overhearing. It is natural to hypothesize that the capacity gain from MACCO is dependent on the client reporting order. To see this, we perform simulation based on synthetic channels used Section 5.2.1, with up to M=K=16. However, our simulation results show that such client order has a negligible impact on the cell capacity (less than 8% capacity drop) even when the clients have very different SNR. We argue that this is because the client with more channel knowledge does not necessarily enjoy more capacity improvement. While this may sound counter-intuitive, our key insight is that MACCO increases the cell capacity mainly by improving the channel orthogonality. Better channel orthogonality can usually lead to an improved capacity for all involved clients while the client with most channel knowledge does not necessarily enjoy most improvement. In other words, the effectiveness of MACCO is toward the entire MU-MIMO cell instead of any individual client.

6. RELATED WORK

MU-MIMO has been demonstrated practical in real-world wireless environments, e.g., [1, 2, 15, 18, 20, 22, 27]. Meanwhile, the capacity scalability problem for a single MU-MIMO cell has been observed from experiments. The authors in [1] observed that when a ZFBF AP with four antennas increases its number of served clients from one to four, the capacity of each client drastically drops. The authors in [20] found out that even the ZFBF AP has 16 antennas, the capacity starts to decrease after more than 12 clients. The authors in [18] observed that the per-client SNR reduction from MU-MIMO is determined by the orthogonality between channels, and proposed a method for the clients to evaluate such channel orthogonality and properly choose the data rate. There are three major approaches toward solving the capacity scalability problem.

The first approach is to let the AP group clients into clusters, only serve the clients in the same cluster at a time, and schedule clusters in a TDMA fashion [9, 19]. However, such an approach does not fully exploit the spatial multiplexing gain of MU-MIMO: the capacity gain is bounded by the reduced number of clients in a cluster. In 802.11ac networks, the inefficiency of such an approach is further amplified by the overhead from channel sounding and channel contention, as the AP needs to contend for and sound the channel for each individual cluster.

The second approach is to increase the number of antennas on the AP, or the Massive-MIMO technique [11]. Argos proposed in [20] features many antennas (M=64) on the base station (AP) to ensure M>>K. BigStation proposed in [27] similarly adopts up to twice as many antennas as clients. However, Massive-MIMO is not suitable for real-world 802.11ac APs that usually have a small form factor or limited cost and energy budget. The authors in [15] and [2] adopt Network-MIMO [24] to connect multiple adjacent APs as a giant virtual AP. Since the virtual AP possesses the antennas from all connected APs, a larger number of clients can be simultaneously served. However, Network-MIMO requires well planned AP deployment and fine-grained time and frequency synchronization between the connected APs, and therefore cannot be applied to 802.11ac networks that are distributed thus uncontrolled.

The third approach leverages multiple client antennas for post-combining as we do, but existing solutions based on it [4, 6, 12, 21] remain theoretical, do not consider channel orthogonality between clients, and suffer from two practical concerns. First, the solutions are often heuristic with expensive iterations, which is hard to implement in practice. Second, the optimization has to be centrally executed by the AP with full channel information. Unfortunately, to enable this in 802.11ac networks, the 802.11ac protocol has to be radically modified to allow the AP to optimize the post-combining vectors for the clients, and explicitly deliver the optimization results to the clients. Such centralized optimization also incurs considerable overhead to the 802.11ac channel sounding. In contrast, by letting the clients overhear others' virtual channels to improve the orthogonality between them, MACCO allows local, per-client post-combining optimization in a distributed, 802.11ac-compliant way.

The focus of this work is the capacity scalability within a single MU-MIMO cell. Neighbouring APs can be subject to inter-cell interference. The authors of [30] proposes a clustering design, NEMOx, that groups adjacent APs into a synchronized cluster and apply Network-MIMO to the smaller number of APs in the cluster. Between clusters, a modified CSMA/CA technique is adopted to avoid inter-cell interference. Clearly, MACCO can be applied to a single cluster in NEMOx.

7. CONCLUDING REMARKS

To address the capacity scalability problem of MU-MIMO, we have proposed a distributed solution based on 802.11ac, called MACCO, to exploit the multiple client antennas for optimal post-combining. In MACCO, each client leverages the channel sounding in 802.11ac to locally optimize its post-combining, based on the successively overheard virtual channel from other clients. We have implemented MACCO on WARP and used both real-world experiments and simulations to demonstrate its capability to significantly improve the capacity scalability of MU-MIMO.

Capacity and Energy Efficiency Tradeoff

The distributed post-combining optimization in MACCO allows a client to explore an interesting tradeoff between its capacity and energy efficiency. That is, the capacity improvement from MACCO depends on two factors: (*i*) the knowledge of other clients' virtual channels, and (*ii*) the number of antennas for post-combining. First, knowing more channels allows the client to optimize its own virtual channel to produce better orthogonality, since it can more precisely predict how the AP will project its data signals. Note, it does not contradict with our conclusion in Section 5.2.3: while MACCO does not provide more gain to clients that report their virtual channels at a later time, for a particular client the number of overheard virtual channels affects the channel orthogonality improvement. Second, more antennas provide the client more flexibility to adjust its virtual channel. They meanwhile offer an increased SNR gain from MRC. However, both factors incur extra energy consumption. To overhear the BF-R frames from other clients, a client must keep its radio transceiver on. To use more antennas, the client has to turn on more RF chains of the transceiver. Therefore, a client has the opportunity to adaptively acquire channel knowledge and use a subset of antennas in MACCO based on its capacity requirement and energy budget. Such capacity and energy tradeoff has been explored by the authors in [14,28,29] for SU-MIMO clients, by changing the number of antennas only. We believe that by letting the client adaptively overhear the channel knowledge, MACCO makes such tradeoff even more interesting.

8. REFERENCES

[1] E. Aryafar, N. Anand, T. Salonidis, and E. W. Knightly. Design and Experimental Evaluation of Multi-User Beamforming in Wireless LANs. In *Proc. ACM MobiCom*, 2010.

[2] H. V. Balan, R. Rogalin, A. Michaloliakos, and K. Psounis. Achieving High Data Rates in a Distributed MIMO System. In *Proc. ACM MobiCom*, 2012.

[3] O. Bejarano, E. W. Knightly, and M. Park. IEEE 802.11ac: From Channelization to Multi-User MIMO. *IEEE Communication Magazine*, 2013.

[4] C. Chae, D. Mazzarese, and T. Inoueand R. W. Heath. Coordinated Beamforming for the Multiuser MIMO Broadcast Channel With Limited Feedforward. *IEEE Trans. Signal Processing*, 2008.

[5] B. Chen, K. C. Lin, and H. Wei. Harnessing Receive Diversity in Distributed Multi-User MIMO Networks. In *Proc. ACM SIGCOMM (Poster)*, 2013.

[6] M. Codreanu, A. Tolli, M. Juntti, and M. Latva-aho. Joint Design of Tx-Rx Beamformers in MIMO Downlink Channel. *IEEE Trans. Signal Processing*, 2007.

[7] M. Costa. Writing on Dirty Paper. *IEEE Trans. Information Theory*, 1983.

[8] V. Erceg, L. Schumacher, P. Kyritsi, A. Molisch, and D. S. Baum. TGn Channel Models. IEEE Doc. 802.11-03/940r2, 2004.

[9] S. Gollakota, S. D. Perli, and D. Katabi. Interference Alignment and Cancellation. In *Proc. ACM SIGCOMM*, 2009.

[10] B. Hochwald and S. Vishwanath. Space-Time Multiple Access: Linear Growth in the Sum Rate. In *Proc. Annual Allerton Conference On Communication, Control and Computing*, 2002.

[11] J. Hoydis, S. Brink, and M. Debbah. Massive MIMO: How many antennas do we need? *arXiv:1107.1709v2*

[cs.IT], 2011.
[12] M. Ku and D. Kim. Tx-Rx Beamforming with Multiuser MIMO Channels in Multiple-cell Systems. In *Proc. Int. Conf. Advanced Communication Technology*, 2008.
[13] S. Kumar, D. Cifuentes, S. Gollakota, and D. Katabi. Bringing Cross-Layer MIMO to Todays Wireless LANs. In *Proc. ACM SIGCOMM*, 2013.
[14] C. Li, C. Peng, S. Lu, and X. Wang. Energy-based Rate Adaptation for 802.11n. In *Proc. ACM MobiCom*, 2012.
[15] H. Rahul, S. Kumar, and D. Katabi. JMB: Scaling Wireless Capacity with User Demands. In *Proc. ACM SIGCOMM*, 2012.
[16] Rice University. WARPLab. `http://warp.rice.edu/trac/wiki/WARPLab`.
[17] Rice University. Wireless Open Access Research Platform (WARP). `http://warp.rice.edu/trac`.
[18] W. Shen, Y. Tung, K. Lee, K. C. Lin, S. Gollakota, D. Katabi, and M. Chen. Rate Adaptation for 802.11 Multiuser MIMO Networks. In *Proc. ACM MobiCom*, 2012.
[19] 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. *IEEE Trans. Signal Processing*, 2006.
[20] C. Shepard, H. Yu, N. Anand, E. Li, T. Marzetta, R. Yang, and L. Zhong. Argos: Practical Many-antenna Base Stations. In *Proc. ACM MobiCom*, 2012.
[21] Q. H. Spencer, A. L. Swindlehurst, and M. Haardt. Zero-forcing Methods for Downlink Spatial Multiplexing in Multiuser MIMO Channels. *IEEE Trans. Signal Processing*, 2004.
[22] 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 *Proc. ACM MobiCom*, 2009.
[23] *Withheld for blind review*. Combating Inter-cell Interference in 802.11ac-based Multi-user MIMO Networks. In *submission to MobiCom (submission ID: 108)*, 2014.
[24] S. Venkatesan, A. Lozano, and R. Valenzuela. Network MIMO: Overcoming Intercell Interference in Indoor Wireless Systems. In *Proc. IEEE Asilomar Conf. on Signals, Systems and Computers (ACSSC)*, 2007.
[25] H. Weingarten, Y. Steinberg, and S. Shamai. The Capacity Region of the Gaussian Multiple-Input Multiple-Output Broadcast Channel. *IEEE Trans. Information Theory*, 2006.
[26] X. Xie, X. Zhang, and K. Sundaresan. Adaptive Feedback Compression for MIMO Networks. In *Proc. ACM MobiCom*, 2013.
[27] Q. Yang, X. Li, H. Yao, J. Fang, K. Tan, W. Hu, J. Zhang, and Y. Zhang. BigStation: Enabling Scalable Real-time Signal Processing in Large MU-MIMO Systems. In *Proc. ACM SIGCOMM*, 2013.
[28] H. Yu, L. Zhong, and A. Sabharwal. Power Management of MIMO Network Interfaces on Mobile Systems. *IEEE Trans. VLSI*, 2012.
[29] H. Yu, L. Zhong, A. Sabharwal, and D. Kao. Beamforming on Mobile Devices: A First Study. In *Proc. ACM MobiCom*, 2011.
[30] X. Zhang, K. Sundaresan, M. A. Khojastepour, S. Rangarajan, and K. G. Shin. NEMOx: Scalable Network MIMO for Wireless Networks. In *Proc. ACM MobiCom*, 2013.

Appendix

Proof of Equation 2. We first formulate the optimization problem faced by C2:

$$\max_{\mathbf{w}_2,\mathbf{v}_2}(|\mathbf{v}_2\mathbf{H}_2\mathbf{w}_2|), \text{ s.t. } \mathbf{w}_2\mathbf{h}_1 = 0.$$

To solve this constrained optimization problem, we convert it into an unconstrained problem. This is achieved by rewriting $\mathbf{w}_2$ as $\mathbf{w}_2$=$\mathbf{h}_1^{\perp}\mathbf{u}_2$ where $\mathbf{u}_2$ is a two-dimensional vector. In other words, since $\mathbf{w}_2$ lies in the orthogonal space of $\mathbf{h}_1$, it can be represented by the linear combination of two basis vectors of $\mathbf{h}_1^{\perp}$. Therefore, the original problem is converted into the following unconstrained optimization problem:

$$\max_{\mathbf{w}_2,\mathbf{v}_2}(|\mathbf{v}_2\mathbf{H}_2\mathbf{h}_1^{\perp}\mathbf{u}_2|),$$

which is the standard eigen beamforming problem under the channel $\mathbf{H}_2\mathbf{h}_1^{\perp}$. According to Section 2, the optimal post-combining vector is given by

$$\mathbf{v}_2 = \mathbf{H}_2\mathbf{h}_1^{\perp}\mathbf{v}_{max}((\mathbf{H}_2\mathbf{h}_1^{\perp})^*(\mathbf{H}_2\mathbf{h}_1^{\perp})).$$