Concurrent Channel Access and Estimation for Scalable Multiuser MIMO Networking

Share Your Story

Concurrent Channel Access and Estimation for Scalable Multiuser MIMO Networking

Tsung-Han Lin and H. T. Kung
School of Engineering and Applied Sciences
Harvard University
{thlin, htk}@eecs.harvard.edu

Abstract—This paper presents MIMO/CON ("MIMO with concurrent channel access and estimation"), a PHY/MAC cross-layer design delivering throughput scalable to many users for multiuser MIMO wireless networking. The basic idea of MIMO/CON is to allow concurrent launches of multiple data transmissions from multiple users so that a multi-antenna MIMO access point (AP) can fully realize its MIMO capacity gain. Using compressive sensing, MIMO/CON can estimate channel state information (CSI) of multiple channels simultaneously without requiring strict synchronization and coordination among distributed users. Furthermore, MIMO/CON can boost channel utilization without explicit access control by allowing the number of concurrent transmissions to exceed receive antennas momentarily. MIMO/CON has been implemented and evaluated on a 4×4 MIMO testbed with software-defined radios. Further, simulation results suggest that MIMO/CON can achieve a 210% improvement in MAC throughput over existing staggered access protocols in a 5-antenna AP scenario.

I. Introduction

MIMO technologies enable an opportunity of linear increase in wireless channel capacity from the additional degrees-of-freedom created by multiple antennas. However, in single-user MIMO, the capacity gain is limited by the relatively small diversity offered by transmit antennas co-located on the same user platform. Multiuser MIMO (MU-MIMO) [7] removes this limitation with geographically separated users and rich spatial diversity. This allows further boosting of channel capacity.

In this paper, we consider a MU-MIMO scenario where an access point (AP) is equipped with many antennas and every user possesses one antenna. We focus on the uplink case where multiple indoor users (i.e., "senders") concurrently transmit data to a multi-antenna AP. With MU-MIMO, one would expect a throughput speedup factor of K with K receive antennas on the AP given sufficient spatial diversity; however realized throughput in a real-world system can be substantially less due to the difficulty of fully parallelizing channel access.

For proper MIMO packet decoding, channel state information (CSI) must be estimated from packet preambles. Existing MU-MIMO systems (e.g., [20], [13]) stagger data transmissions in order to allow random access yet avoid preamble collisions which could impede CSI estimation. Staggered transissions, however, result in efficiency loss that increases with the number of senders. For example, consider 1500-byte packets transmitted with 39Mbps data rate. Note that each packet transmission spans $300\mu s$. With an average access delay of $100\mu s$ [14], there can be no more than 3 concurrent
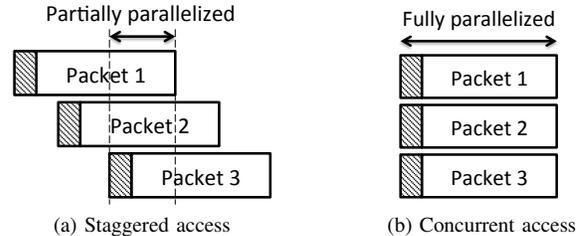

Fig. 1: Two access strategies for multiuser MIMO (MU-MIMO) networks. Shaded areas denote packet preambles. Staggered access means only partially parallelized data transmissions, resulting in low channel utilization. In contrast, concurrent access can realize MIMO capacity gain by fully parallelizing data transmissions.

transmissions. Further, they are only partially parallelized as depicted in Figure 1(a). One may use frame aggregation [2] to send longer payload and amortize the access overhead. However, frame aggregation is not practical for delay sensitive traffics such as VoIP or HTTP. Such protocols are thus not scalable to large K.

We argue that a more efficient approach to coordinate distributed senders is to launch multiple data transmissions concurrently without forcing serialization thereby allowing the transmissions fully parallelized (shown in Figure 1(b)). We call this access strategy *concurrent access*. To realize concurrent access, our proposed design, MIMO/CON, has the following two features:

- MIMO/CON can derive accurate CSI using only *loosely synchronized* and *uncoordinated* concurrent preambles, thus suited to geographically separated users. This, however, usually means an unpractically long preamble sequence with length proportional to synchronization offset. In contrast, by using compressive sensing techniques, MIMO/CON can reduce the preamble length to near the minimum required as in the ideal case where senders are perfectly synchronized and scheduled for transmission.
- MIMO/CON can boost channel utilization without explicitly controlling the number of senders to match the number of AP receive antennas. With random access, senders are likely to either underutilize the available degrees-of-freedom, or overbook with collisions. MIMO/CON mitigates this problem by a novel scheme, called *delay packet*

decoding, that can opportunistically decode packets in collisions at a later time. This means only a subset of packets involved in a collision need to be retransmitted. Hence, MIMO/CON on average need not be hampered by the efficiency loss when senders make uncoordinated access decisions.

MIMO/CON exploits two important insights. First, the CSI, i.e., the channel impulse response, is expected to be *sparse* and constituted of only a few significant taps due to the small delay spread in an indoor environment. Second, allowing concurrent preambles to be loosely synchronized and uncoordinated does not create additional channel state information. Instead, this merely injects the CSI to a higher-dimensional space with many more zero variables due to the additional potential preamble arrival times and potential senders (see Section III-A). To solve for this higher-dimensional CSI, instead of naively increasing the preamble length to match the number of variables, MIMO/CON directly measures the sparse information. MIMO/CON leverages the recent theory of compressive sensing [6] which shows that the sparse information can be derived almost as if the locations of the nonzero unknowns are known a priori. As a result, by formulating a sparse identification problem, MIMO/CON can minimize the concurrent preamble length for practical MU-MIMO systems.

Concurrent channel estimation also provides better handling of collisions when the number of senders exceeds the number of AP antennas. This follows from the observation that concurrent channel estimation is independent from MIMO degrees-of-freedom and can identify senders even in a collision. MIMO/CON thus buffers collisions and exploits later retransmissions to find decoding opportunity. As a result, MIMO/CON can tolerate demand fluctuations better and relax the access control by concerning only average use of the medium and realize statistical gains over random access.

We have prototyped MIMO/CON using software-defined radios, and evaluated it through testbed experiments and simulations. Our evaluation reveals the following:

- The aggregated network throughput of MIMO/CON scales well with users. In particular, our simulation results suggest that MIMO/CON delivers 140% and 210% throughput gains over staggered access with 5-antenna AP under PHY rates 13Mbps and 52Mbps, respectively.
- The compressive sensing based channel estimation scheme works over a wide range of SNRs. In particular, the derived CSI can decode MIMO packets successfully with SNR as small as 5dB, the minimum required for data transmission, and achieves performance comparable to interference-free, serially transmitted preambles.
- Compressive sensing is a natural fit for MIMO channel estimation. The antenna diversity presented by multiple antennas on MIMO AP can speed up compressive sensing decoding. Using 4 receive antennas, the decoding algorithm converges in 1 iteration for most of the experiments.

In short, MIMO/CON can achieve high and scalable MAC efficiency to take advantage of an increased number of receive antennas on the AP, and is amenable to the future trend of massive MIMO designs (e.g., 802.11ac suggests up to 8 antennas on the AP, and an unlimited number of antennas scenario is depicted in [15] for cellular networks).

II. RELATED WORK

MIMO/CON is closely related to and built on prior work on practical MU-MIMO systems [20][13]. For backward compatibility reasons, we share the same goal of realizing MU-MIMO throughput gains with widely used WiFi-like CSMA. As future MIMO designs are expected to see a substantial increase in the number of antennas, MIMO/CON further addresses the scalability issue in MU-MIMO medium access control. A recent proposal, Contrabass [24], has the objective of realizing concurrent access, but the design does not exploit the expected sparsity for channel estimation, and thus suffers from a higher overhead in preamble length and complexity. Throughput scalability issues can also be found in future point-to-point high speed WLAN. FICA [19] and WiFi-Nano [14] both aim at reducing MAC inefficiency under this setting.

Exploiting channel sparsity for channel estimation has a long history of investigation (see [4] for a nice review.) In particular, [18] shares a similar random probe formulation with MIMO/CON. However, MIMO/CON stands out from all prior work in two aspects. First, previous work assume long delay spread environments with perfect synchronization among senders and known sender identity. MIMO/CON instead tackles the short delay spread and loosely synchronized WiFi environment, and uses compressive sensing to resolve timing misalignment and identify transmitting users. Second, previous work focus on PHY layer improvement such as enhancing demodulation performance and pilot size reduction. In contrast, MIMO/CON demonstrates that exploiting channel sparsity can lead to significant throughput gains in MAC level and is of particular importance for multiuser MIMO networking.

Lastly, MIMO/CON's collision handling is based on interference removal techniques, which has previously been used to handle interference under various settings (e.g. [9], [12], [10]). The novelty of MIMO/CON's approach lies in packet identification. With compressive sensing decoding, MIMO/CON can reliably identify packets from concurrent channel estimation while other approaches such as ZigZag [9] rely on the small frequency differences in oscillator between sender hardware.

III. BACKGROUND

MIMO systems rely on distinct spatial signatures (the CSI vector) from each transmit antenna to separate concurrent data streams. Signals received by multiple receive antennas sit in a higher dimensional space, and by projecting the signal onto proper subspace formed by the spatial signatures, individual data streams can be decoupled and decoded (see, e.g., [22]).

The spatial signature is usually measured from a known preamble sequence preceding the data packet. In single-user MIMO, preambles from different transmit antennas are scheduled for serial transmissions by the same sender. However, in

the multiuser scenario, preamble transmissions may not easily be scheduled and thus may suffer from mutual interference. A noisy preamble under such interference prohibits accurate CSI estimation and results in poor MIMO performance. Prior work [20][13] resolves this issue by ensuring that preambles from different senders are not overlapped in time. The AP can then estimate the CSI vectors by sequentially projecting the received signal onto interference-free subspaces.

Serialized non-overlapping preambles, however, impose a significant channel access delay. In order to avoid preamble collisions, a set of K concurrent transmissions has to undergo K access delay. As shown in Figure 1(a), this hampers the MIMO throughput especially under a high PHY data rate (thus shorter packet durations) or a large number of receive antennas (thus many concurrent senders expected).

A. Concurrent channel access and exploitable sparsity

MIMO/CON takes an opposite approach. It launches multiple data transmissions simultaneously so that the contention cost is paid only once for multiple senders. The principle is simple: assuming the backoff operations of senders are in lockstep with each other, generally concurrent access occurs by increasing the sender transmission probability of a time slot. After a set of concurrent transmissions begins, other senders can detect and avoid interfering with ongoing transmissions via carrier sensing. A nice thing about this concurrent access scheme is that the senders behave exactly as in 802.11 DCF and do not even need to be aware of AP's MIMO capability to realize MIMO capacity gain. A larger number of AP's receive antennas will result in a lower contention level and the senders can transmit more aggressively. Conversely, the senders will see a higher contention level and back off when the AP has fewer antennas.

To realize concurrent access, MIMO/CON must maintain synchronization among senders and must obtain CSI from fully overlapped preambles. Note that the synchronization here means only a loose one in the sense that substantial synchronization offsets between transmissions can be tolerated. Such synchronization in general is required in MU-MIMO systems for proper MIMO decoding, and can be realized by exploiting the cyclic prefix (CP) structure in OFDM symbols (see Section VI). Therefore, the major challenge lies in channel estimation from a collection of concurrently received preambles, which this paper studies.

To see how channels can be concurrently estimated, let us first understand the sparsity in channel impulse response. The channel impulse response is the channel distortion from a transmitter to a receiver. Narrowband OFDM subcarriers in general can be modeled as flat fading channels, where the channel distortion can be approximated as a complex value representing the amplitude attenuation and phase shift [22].

We use vector $\hat{\mathbf{h}}$ to represent the channel distortion of the subcarriers. Here we use the "hat" notation to denote the frequency domain representation of a vector. Suppose now a sender transmits a preamble vector $\hat{\mathbf{d}}$ over the OFDM subcarriers. The received signal on the i-th subcarrier can be
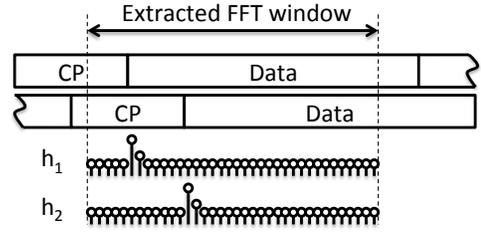

Fig. 2: The locations of the significant taps are determined by the timing misalignment between the beginning of the extracted FFT window and that of preamble sequence.

written as

$$\hat{y}_i = \hat{h}_i \hat{d}_i + \hat{n}_i \qquad (1)$$

where $\hat{n}$ is the noise. Taking an inverse Fourier transform on the received signal yields its time domain representation:

$$\mathbf{y} = \mathbf{h} \otimes \mathbf{d} + \mathbf{n} \qquad (2)$$

The time domain convolution in (2) can be interpreted as the intersymbol interference caused by the multipath effect where signals with different propagation delays overlap with each other. Under this interpretation, the components of $\mathbf{h}$ thus can be viewed as the channel distortion on paths with different propagation delays. In an indoor environment, the delay spread is supposed to be small (30-60 ns or even less, based on various measurement studies [3][8]). Therefore, we can assume $\mathbf{h}$ is a very sparse vector. For example, in 802.11n with 20MHz bandwidth, the sampling interval is 50ns. This means $\mathbf{h}$ has no more than 2 nonzero components.

However, the locations of the few nonzeros in $\mathbf{h}$ are in general unknown at the receiver. Although with short delay spread, the significant taps should always be the first few, their locations in fact vary according to the timing misalignment of the extracted FFT window for an OFDM symbol. This phenomenon is illustrated in Figure 2 that shows two concurrent OFDM symbols and the associated $\mathbf{h}$. One can see that the indices of the significant taps are the differences between beginning of the FFT window and that of preamble sequence.

Let us extend (1) to the case of multiple concurrent senders:

$$\hat{y}_i = \sum_j \hat{d}_j \hat{h}_{ij} + \hat{n}_i \qquad (3)$$

As shown in (3), the signal $\hat{y}_i$ received on the i-th subcarrier is a linear combination of the channel responses from different transmitters j. To solve for $\hat{h}_{ij}$, we need to collect as many equations as the number of unknowns. The number of unknowns is determined by two system parameters: the maximum synchronization offset and the number of potential senders. Figure 3 illustrates the relationship between the number of unknowns and synchronization offsets. For simplicity, we assume every channel has one significant tap. In Figure 3(a), the three OFDM symbols are perfectly synchronized. Since we know the significant tap must be the first one, there are only 3 unknowns. In Figure 3(b), the maximum timing misalignment between the OFDM symbols is 4 samples. Hence

for each channel, there are 4 possible significant tap locations, resulting in 12 unknowns. Similarly, for a larger misalignment of 8 samples in Figure 3(c), the number of unknowns increases to 24. The number of unknowns thus increases proportionally to synchronization offset. Further, given the sender identity is not known a priori, the total number of unknowns needs to be multiplied by the number of potential senders. For geographically distributed users, it is generally difficult and also expensive to provide tight synchronization among them. In this case, it is desirable to use loose synchronization which results in large synchronization offset. One thus would need a very long preamble in order to generate enough equations for solving these many unknowns.

For example, consider a scenario with loose synchronization provided by a simple reference broadcast method which has $2\mu s$ accuracy [19]. This means under 20MHz bandwidth, the significant taps can be at 40 possible locations. Assuming a 4-antenna AP with 100 potential senders and no more than 2 taps in their channel response, the number of unknowns is at least 4000 (40×100) while only 8 of them are not zero. A naive solution would need to send at least $200\mu s$ (4000×50ns) long preambles, which is clearly unpractical due to the very large preamble overhead. In contrast, if we can directly estimate the 8 nonzero variables, 400ns long preambles would generate a sufficient number of equations. This thus motivates exploiting compressive sensing for concurrent channel estimation. We will briefly introduce compressive sensing in the next section, and in Section IV, we will discuss how to design concurrent channel estimation based on compressive sensing.

B. Compressive sensing

Compressive sensing has been shown to be a powerful approach in compressing discrete sparse signals of large sizes [6]. It arises from an interesting question: *Given a K-sparse vector of some large length N, can one recover the K nonzero components using M linear measurements with M < N?* Equivalently, given an $M \times N$ *sensing matrix* $\mathbf{A}$, and the measurement vector $\mathbf{y} = \mathbf{A}\mathbf{x}$, can we recover the unknown vector $\mathbf{x}$ exactly? Since this linear system is underdetermined, the problem in general has infinite solutions for $\mathbf{x}$. However, it has been shown that exact recovery of sparse solutions is possible by taking sufficiently many random projections of $\mathbf{x}$, i.e. use a random matrix for $\mathbf{A}$. These random projections can capture all useful information about the sparse vector with high probability. More precisely, recovery is assured when the sensing matrix $\mathbf{A}$ satisfies, e.g., the restricted isometry property (RIP). In essence, the RIP means that any submatrices of $\mathbf{A}$ are close to orthonormal so that when the submatrices operate on the nonzero components of the sparse vector, the information about the nonzeros is not lost. It has been shown in the literature (see, e.g., [5]) that the number M of measurements required can be as small as $O(K \log \frac{N}{K})$, meaning that the number is approximately a constant multiple of the sparsity K when $\log \frac{N}{K}$ is small. Empirically, the factor is between 3 and 4, which we call the *oversampling ratio*.
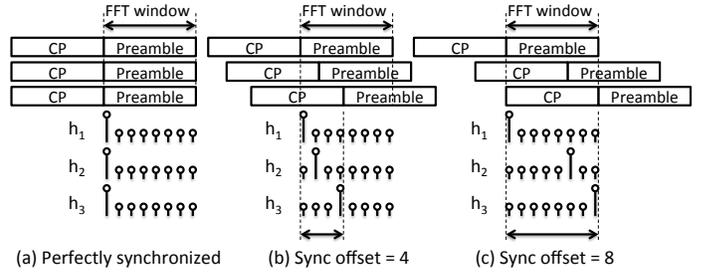

(a) Perfectly synchronized (b) Sync offset = 4 (c) Sync offset = 8

Fig. 3: The number of unknowns in channel impulse response is proportional to the maximum synchronization offset.

IV. CONCURRENT MULTIUSER CSI ESTIMATION

A challenge of estimating CSI from concurrent preambles is that although the information we want to capture is small, there can be a huge number of unknowns. But the AP does not know a priori which senders participate in the concurrent transmission, nor the timing misalignment to determine the location of the significant taps. If the AP had this information, it could have dropped all zero unknowns in CSI estimation and a short preamble would be sufficient.

Fortunately, we can leverage the major insight from compressive sensing: *a few random projections of the unknown vector can preserve sufficient information for sparse recovery with high probability*.

A. Random preamble sequences for CSI estimation

Bearing this insight in mind, in MIMO/CON, users identify themselves with distinct random codes in their preambles. As we will see, the AP will receive random linear measurements of channel impulse response formed from these codes. For simplicity, MIMO/CON uses Bernoulli random codes composed of {1,-1}, which are assigned by the AP during initial association.

To work with OFDM, we assume the preamble length equals the number of subcarriers M, and the preamble is sent over individual OFDM subcarriers. Denote the preamble sequence owned by sender i as a vector $\mathbf{a}_i$. The received signal $\hat{\mathbf{y}}$ at the AP can be written as a linear combination of all concurrently transmitted preambles, passing through the channel with noise:

$$\hat{\mathbf{y}} = \hat{\mathbf{A}}\hat{\mathbf{h}} + \hat{\mathbf{n}} = \begin{bmatrix} \Phi_1 & \Phi_2 & \cdots & \Phi_N \end{bmatrix} \begin{bmatrix} x_1\hat{\mathbf{h}}_1 \\ x_2\hat{\mathbf{h}}_2 \\ \vdots \\ x_N\hat{\mathbf{h}}_N \end{bmatrix} + \hat{\mathbf{n}} \quad (4)$$

where

$$\Phi_i = diag(\mathbf{a}_i) = \begin{bmatrix} a_{i1} & & \\ & \ddots & \\ & & a_{iM} \end{bmatrix} \quad (5)$$

$\hat{\mathbf{h}}_i$ is a complex vector denoting the channel frequency response from transmitter i to the AP. Since only a subset of senders transmit, we use a {0,1} binary variable x_i to indicate whether sender i is active over a total of N senders. In (4), the dimension of $\hat{\mathbf{A}}$ is $M \times MN$, and thus (4) is
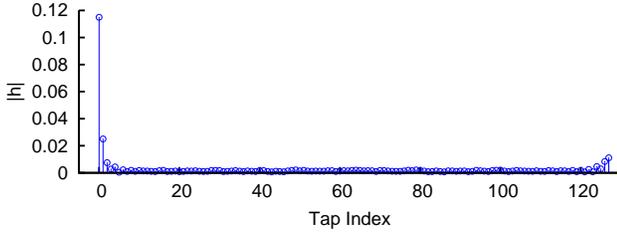
Fig. 4: Channel impulse response measured with 6.25MHz bandwidth. A significant tap is observed at tap 0 with some energy leakage around.

an underdetermined system that cannot be solved by matrix inversion.

As stated earlier, the unknown vector is sparse in the time domain. Hence we first convert the system into a sparse recovery problem by taking inverse Fourier transform on individual channel responses.

$$\hat{\mathbf{y}} = \mathbf{A}\mathbf{h} + \hat{\mathbf{n}} = \begin{bmatrix} \Phi_1\mathbf{F} & \Phi_2\mathbf{F} & \cdots & \Phi_N\mathbf{F} \end{bmatrix} \begin{bmatrix} x_1\mathbf{h}_1 \\ x_2\mathbf{h}_2 \\ \vdots \\ x_N\mathbf{h}_N \end{bmatrix} + \hat{\mathbf{n}} \quad (6)$$

Let us now interpret (6) using compressive sensing. Note that the delay spread D is the same for every channel between the senders and the AP. Assume the number of active senders is K. $\mathbf{h}$ is then a DK-sparse vector of length MN. The received signal $\hat{\mathbf{y}}$ is the compressively sensed measurement vector with length M. To recover $\mathbf{h}$, we only need M to be a small multiple of DK, which can be much less than MN. Note that M is the number of subcarriers and is independent from DK. Therefore one can control the number of measurements to accommodate different sparsity by adjusting the OFDM FFT size.

Before we describe how to solve the sparse recovery problem concerning (6), there are a few points worth noting. First, the formulation can be thought of as a generalized form of CDMA that attempts to multiplex preamble transmissions without creating mutual interference. Traditional CDMA requires that the codes possessed by different senders to be orthogonal to each other. However, this assumes the worst case that all senders will transmit concurrently. Since we know the number of concurrent senders is bounded above by K, we can have a less constraining requirement that asks for only every subset of K codes to be orthogonal. This is exactly the formulation of compressive sensing that leads to a shorter code length.

Second, although the delay spread in an indoor environment is small and should contain only 1 or 2 significant taps, in practice the measured channel impulse response can have more nonzero taps due to leakage [23]. The leakage effect is a result of propagation delays that are not multiples of the sampling intervals. The energy of these delays then leaks into every tap in the discretization process. Figure 4 shows a channel impulse response measured in an indoor environment with

Algorithm 1 CoSaMP algorithm

Input: measurement vector $\hat{\mathbf{y}}$, sensing matrix $\mathbf{A}$, and sparsity level KD

Output: estimated channel vector $\mathbf{h}_i$

1: $\mathbf{h_0} = 0$; $\mathbf{u} = \hat{\mathbf{y}}$; $i = 1$;
2: **while** $|\mathbf{u}| > tol$ **do**
3:    $\mathbf{p} = \mathbf{A}^\mathbf{H}\mathbf{u}$ (Support estimation)
4:    $\Omega = \mathrm{supp}(\mathbf{p}_{\beta KD})$
5:    $T = \Omega \cup \mathrm{supp}(\mathbf{h}_{i-1})$ (Merge previous support)
6:    $\mathbf{b}|_T = \mathbf{A}_T^\dagger\mathbf{y}$ (Least squares to estimate signal)
7:    $\mathbf{b}|_{T^c} = 0$
8:    $\mathbf{h}_i = \mathbf{b}_{KD}$ (Prune estimate to be KD-sparse)
9:    $\mathbf{u} = \hat{\mathbf{y}} - \mathbf{A}\mathbf{h}_i$ (Update signal residual)
10:    $i = i + 1$
11: **end while**

leakage. Fortunately, the leakage is concentrated around the most significant tap, and can be almost entirely captured by measuring a few additional neighboring taps.

Lastly, note that the scheme exploits the sparsity in channel impulse response, or equivalently the correlation between channel response on neighboring subcarriers. Therefore, it is important to obtain measurements from a sufficient number of subcarriers in order to capture the correlation. This suggests that preambles composed of $\{0,1\}$ random sequence will not yield good recovery performance.

B. Sparse CSI recovery

MIMO decoding relies on per-packet spatial signatures, and therefore sparse recovery of CSI must be done within a packet time, which is usually several hundreds of microseconds. However, sparse recovery is generally considered a computationally expensive problem. MIMO/CON exploits the diversity of multiple receive antennas on the the same AP platform to relieve the computation burden and shorten the decoding time.

Specifically, we note that the multi-antenna diversity fits well with a popular class of sparse recovery algorithms based on orthogonal matching pursuit (OMP) [21]. In OMP-type algorithms, the algorithms iteratively make guesses on the locations of potential nonzero unknowns and drop all other zero unknowns. As individual receive antennas obtain independent measurements, they naturally can make the guessing more robust, and make the algorithm converge faster. Our decoding algorithm is built on CoSaMP [17], the state-of-the-art efficient algorithm for sparse recovery, and extends the algorithm to incorporate the multi-antenna diversity.

1) The CoSaMP algorithm: The basic idea of CoSaMP is simple. Given that the solution vector $\mathbf{h}$ is KD-sparse, if we know the locations of the KD nonzero variables (the "support"), we can eliminate all other variables and turn the problem into an overdetermined one. The overdetermined system can then be solved by standard least squares algorithms. If the set of guessed nonzero variables is not entirely correct, the same technique can be used again to improve support

estimation from the residual signal. We display the pseudocode of the algorithm in Algorithm 1.

Support estimation is the most critical step in the algorithm. A good estimation would lead to the correct sparse solution with rapid convergence. The support can be estimated via a *proxy* vector $\mathbf{p} = \mathbf{A}^{\mathbf{H}}\mathbf{A}\mathbf{h}$. Because A is RIP, the submatrices of $\mathbf{A}$ are close to orthonormal. This means that components in $\mathbf{p}$ with large magnitudes will point to nonzeros in $\mathbf{h}$. Since we have $\mathbf{y} = \mathbf{A}\mathbf{h}$, the proxy vector can be computed by a simple matrix vector multiplication:

$$\mathbf{p} = \mathbf{A}^{\mathbf{H}}\hat{\mathbf{y}} \qquad (7)$$

Finally we introduce a simple heuristic on support set selection which simplifies the selection process and also simplifies our future discussion. The heuristic exploits the hierarchical structure in $\mathbf{h}$ that $\mathbf{h}$ can be divided into N blocks corresponding to N senders. One thus can first estimate the active senders and then estimate the significant taps only from the active senders. The active senders can be estimated by forming a *sender proxy* by taking the largest value of each block in the proxy vector. In our implementation, we first estimate αK senders, and $\beta K D$ taps with $\alpha = 1$ and $\beta = 2$. Later we will show that with multi-antenna diversity, choosing proper α and β is not difficult because the proxy vector is robust.

2) Multi-antenna diversity: Multi-antenna diversity arises from an important observation: the $\mathbf{h}$ vectors observed by different receive antennas share the same support. That is because preambles received at co-located antennas come from the same set of active senders, they share the same symbol timing misalignment. One thus can exploits this diversity presented in measurements collected at every antenna.

Incorporating the diversity into the CoSaMP algorithm requires only one modification: replacing line 3 of Algorithm 1 with (8).

$$\mathbf{p} = \frac{1}{K} \sum_{i=1}^{K} abs(\mathbf{A}^{\mathbf{H}}\mathbf{u}_i) \qquad (8)$$

Note that abs denotes taking element-wise absolute of the vector and K is the number of AP antennas. Intuitively, (8) reduces the noise in the proxy estimate by taking the average of multiple estimates. However, this is only true if individual measurements are sufficiently different.

To see where the real diversity is from, let us first consider a simple case where there are only K nonzero variables in $\mathbf{h}$, and say the nonzero variables are the first K ones, h_1 to h_K. Denote the entries of $\mathbf{A}^{\mathbf{H}}\mathbf{A}$ as b_{ij} with its diagonal of all 1's, we can expand Eq.(7):

$$\begin{cases} p_i = h_i + \sum_{j=1, j \neq i}^{K} b_{ij}h_j & \text{if } i = 1...K \\ p_i = \sum_{j=1}^{K} b_{ij}h_j & \text{if } i > K \end{cases} \qquad (9)$$

Note that $\mathbf{A}^{\mathbf{H}}\mathbf{A}$ is diagonally dominant, i.e. b_{ij} is small when $i \neq j$; the proxy $\mathbf{p}$ thus equals to $\mathbf{h}$ distorted by some noise. In other words, finding the support from $\mathbf{p}$ is a detection problem where the two equations in Eq.(9) can

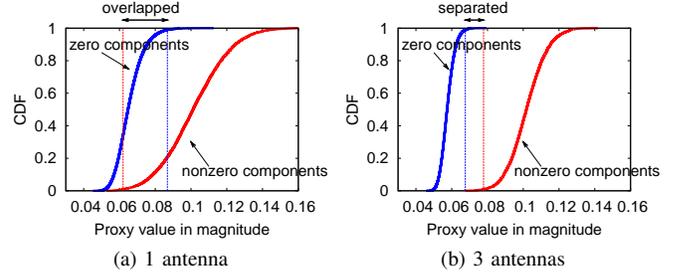

(a) 1 antenna (b) 3 antennas

Fig. 5: Multi-antenna diversity improves the quality of support selection. Measurements from multiple antennas can help distinguish the locations of nonzero and zero variables.

be approximated as two Gaussian distributions with mean h_i and 0. The misidentification rate is then determined by the overlapping region of the tails in the two distributions.

We can write a similar expression for the proxy vector $\mathbf{p}'$ obtained from the second antenna:

$$\begin{cases} p'_i = h'_i + \sum_{j=1, j \neq i}^{K} b_{ij}h'_j & \text{if } i = 1...K \\ p'_i = \sum_{j=1}^{K} b_{ij}h'_j & \text{if } i > K \end{cases} \qquad (10)$$

Eq.(9) and Eq.(10) share the same b_{ij} because the preamble sequences of the senders are the same. The only difference thus lies in the channel impulse response observed by the two antennas. Given the two antennas are located close to each other, the observed signal attenuation will not have too much diversity. However, their *phase shift* can easily be very different. The wavelength of GHz waves is on the order of 10cm. We then can model Eq.(10) as independent Gaussian distributions from Eq.(9). In other words, if we have K proxy vectors, by taking the average over their magnitudes, the variance of the resulting Gaussian distributions can be reduced in a rate of $O(K^{-1/2})$. Therefore, the size of the tail overlapping region diminishes and the proxy estimate quickly becomes very robust.

To demonstrate that the diversity of phase shift in $\mathbf{h}$ improves the robustness of proxy vectors, we conduct a simulation that deliberately sets the signal attenuation in channel response to be the same and varies the phase shift uniform randomly between $[0,2\pi)$. We assume a scenario with $K = 6$ active senders out of total $N = 100$, each with one significant tap of magnitude 0.1 and 4 leakage taps on both sides with magnitude 0.03. The simulation is repeated 1000 times with different random sensing matrices and different locations of the taps.

Figure 5 shows the resulting CDF of the magnitudes of the sender proxy values. In Figure 5(a), misidentification occurs when the proxy values that point to nonactive senders is higher than those pointing to active senders. In this case, the CoSaMP algorithm needs to be run for more iterations to correctly identify all active transmitters. In contrast, in Figure 5(b), with 3 antennas the identification has become relatively easy, due to the reduced variances of the two distributions.

Finally, we note that robust support estimation is not only

beneficial in improving the computation speed; but also helpful in reducing the oversampling ratio of compressive sensing. The result will be shown in Section VII-C.

3) Computational complexity: The computational complexity of the decoding algorithm is dominated by the support estimation. With a naive matrix-vector multiplication $\mathbf{A}^{\mathbf{H}}\hat{\mathbf{y}}$, the computational complexity is $O(NM^2)$. However, $\mathbf{A}$ involves multiple DFT matrices, and we can compute the proxy vector in blocks to exploit the structure in DFT matrices. For example, the i-th block in $\mathbf{p}$ can be computed by:

$$\mathbf{p_i} = \mathbf{F}^{-1}\mathbf{\Phi_i^H}\hat{\mathbf{y}} \tag{11}$$

Given that $\mathbf{\Phi_i^H}$ is a diagonal matrix, the overall complexity is therefore $O(NMlogM)$ with FFT. The other computationally extensive operation in the algorithm is related to solving least squares problems. We can use standard iterative methods such as the conjugate gradient algorithm to achieve $O(MDK)$ computation time.

V. Maximizing Channel Utilization

Beyond concurrent channel estimation, the MIMO/CON MAC layer needs to control the number of concurrent senders to maximize channel utilization. Suppose the AP has K receive antennas. Ideally we want to ensure that always K senders transmit concurrently. However, because random access by distributed senders inevitably leads to fluctuations between underutilizing (less than K senders) and overbooking the channel (collisions), this problem cannot be generally solved without exchanging information between distributed senders.

Instead, MIMO/CON mitigates the problem by *delay packet decoding* to allow momentarily channel overbooking. The opportunity arises from two observations: first, concurrent channel estimation can be decoupled from MIMO degrees-of-freedom. That is, with a proper preamble size, MIMO/CON can learn the sender identities and the associated CSI even in a collision. Second, the MAC layer normally retransmits collided packets at a later time. Therefore, MIMO/CON can exploit the correctly received retransmissions to opportunistically decode packets involved in previous collisions.

To illustrate the idea, consider a simple scenario that the AP has two antennas, and at time t_1 three senders transmit packets p_1, p_2, and p_3 concurrently. Thus the AP receives the following:

$$\mathbf{y} = \mathbf{h_1}p_1 + \mathbf{h_2}p_2 + \mathbf{h_3}p_3 \tag{12}$$

Since the AP has degrees-of-freedom two, at this point the AP cannot decode the concurrent transmissions and this is a collision. Suppose p_3 is retransmitted and received correctly at a later time t_2. We then can regenerate $\mathbf{h_3}p_3$ in the first collision to decode packets p_1 and p_2:

$$\mathbf{y} - \mathbf{h_3}p_3 = \mathbf{h_1}p_1 + \mathbf{h_2}p_2 \tag{13}$$

noting that $\mathbf{h}_3$ has been obtained at t_1 via concurrent channel estimation. Since this equation has only two unknowns left, we can proceed to decode p_1 and p_2.

Now we are left with the question on how to adapt the sender transmission probability to network contention level. MIMO/CON's approach is similar to 802.11 DCF: when a sender sees a transmission opportunity, it tosses a coin to determine whether it will begin a transmission. If a collision occurs, the transmission probability is reduced to avoid future collisions. This strategy fits well with the above collision handling scheme since the retransmission is less likely to be collided and the AP can go back to decode packets in the previous collision. The classic additive increase multiplicative decrease (AIMD) control principle, for example, can be used to probe for optimal transmission probability and achieve fairness among senders.

VI. Discussion

In this section, we discuss several design issues related to implementing MIMO/CON in practice.

(a) Hidden terminal: As in traditional CSMA, MIMO/CON avoids collisions through carrier sensing; therefore it also suffers from the hidden terminal problem where hidden senders cannot be detected and may cause interference. The problem may result in channel overbooking, and asynchronous concurrent transmissions that do not have overlapped preambles. When there are less than K concurrent transmissions, one can still apply the chain decoding technique [20] in staggered access to separate the two data streams if preambles are not overlapped. On the other hand, when the contention level is high, one will need to use RTS/CTS handshakes to contain the traffic. Interestingly, one can easily envision that concurrent preambles can also be a good primitive for building efficient concurrent RTS. A full design of concurrent RTS however is beyond the scope of this paper and left as future work.

(b) Frequency and time synchronization: MIMO/CON does not have more stringent requirement on frequency and time synchronization over existing MU-MIMO systems such as the system described in [13]. Synchronization techniques in previous literature [19][13] can be employed in MIMO/CON. For frequency synchronization, since hardware oscillator frequency is relatively stable, the senders can use the AP's frequency as a reference, and correct the offset periodically. For timing synchronization, symbol timing misalignment between concurrent preambles can be tolerated by using the cyclic prefix design in OFDM as a guard interval (see, e.g., [16]). Therefore one can adjust the CP length to accomodate the synchronization error and in the mean time, scale the data length accordingly to maintain the same CP overhead percentage.

(c) Rate adaptation: In MIMO, the level of inter-stream interference depends on the orthogonality between the spatial signatures of concurrent senders. As a result, rate adaptation is especially difficult if the spatial signatures of concurrent senders cannot be known at the sender a priori. This issue may be addressed by a rateless rate adaptation design (e.g., [11]), where rate adaptation can be totally blind.

(d) Backward compatibility: MIMO/CON builds on random access and carrier sensing and only changes the preamble

structure when transmitting uplink data; therefore it may seem that MIMO/CON nodes can operate with 802.11 nodes. However, since MIMO/CON nodes can transmit concurrently with each other but not with normal 802.11 nodes, they will have a lower collision rate than 802.11 nodes. As a result, 802.11 nodes may spend more time in backoff and are disadvantaged in channel access. A simple strategy to mitigate the problem is to have MIMO/CON nodes transmit less aggressively and sacrifice network throughput somewhat when coexisting with 802.11 nodes.

Lastly, although throughout the paper we assume that each sender is equipped with one antenna, the results can easily be generalized to the case with multi-antenna senders by having the sender operate as multiple single antenna senders.

VII. EVALUATION

We have implemented MIMO/CON on software-defined radios. We use the USRP-N200 boards with WBX daughterboards, and drive them with the UHD software [1]. The radios operate with center frequency 916MHz and a 6.25MHz bandwidth. In the testbed experiments, we focus on evaluating the performance of concurrent preambles. For delay packet decoding, our implementation is based on interference removal; such implementation has been studied extensively in the literature [9][20]. Thus we focus our evaluation on performance gains of delay packet decoding on overall throughput.

In implementing concurrent preambles, a slight change is made in the formulation in (6): the DC subcarrier is not used for avoiding unwanted DC offset from the wireless transceiver. Note that the DC offset can shift the zero values in channel impulse response to a nonzero constant and thus eliminate the sparsity.

### A. MIMO decoding performance with concurrent preambles

We use a 4×4 MIMO scenario to evaluate the performance of concurrent channel estimation in a lab environment. The performance is compared against a baseline case where interference-free preambles are transmitted sequentially. In the setting, we assume there are 100 senders but only 4 of them transmit at any given time. The distance between the transmitters and the receivers is around 2 to 3 meters. We vary the transmission power and the distances to get different SNR values.

For the baseline scheme to which MIMO/CON compares, we apply the standard least squares method [23] to interference-free preambles. The channel is estimated by solving the following equation:

$$\hat{\mathbf{h}} = \Phi^{-1}\hat{\mathbf{y}} \tag{14}$$

where $\Phi = diag(\mathbf{a}_i)$ and $\mathbf{a}_i$ is the known preamble sequence.

In both cases, the obtained channel estimate is then used to decode 4 MIMO data streams immediately followed by the preamble with the standard zero-forcing method and successive interference cancellation [22]. The FFT size of both preamble and data symbols are set to 128 points. We repeat each experiment 300 times with different random preambles.
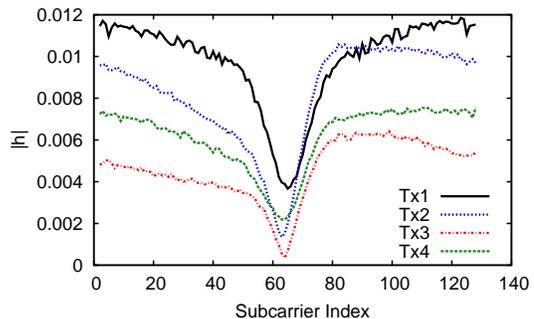
(a) CSI measured with interference-free preambles
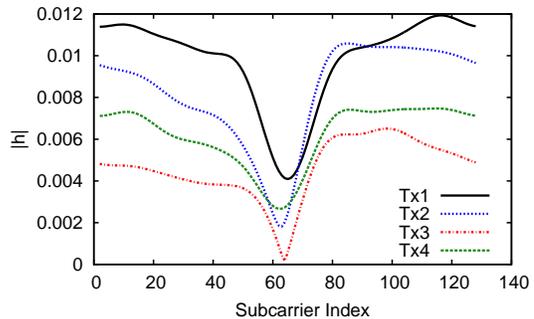
(b) CSI measured with concurrent preambles

Fig. 6: Comparison of the frequency domain CSI measured from interference-free preambles and concurrent preambles.

Figure 6 shows an example of the resulting channel estimate on every subcarrier. The 4 curves correspond to the CSI of the 4 transmitters to 1 receiver. In Figure 6(a), the channel estimation from interference-free preambles gives a less smooth curve due to the noise in channel estimation as the estimation is done with a single preamble symbol without any averaging. In contrast, the channel estimate from concurrent preambles is smoother as shown in Figure 6(b). The smoothness reflects the assumption that the channel has only a few significant taps. It can be seen that the curves in both figures are fairly close to each other.

How does the CSI obtained from concurrent preambles perform in MIMO decoding? Figure 7 shows the scatter plots of the decoded SNR of the subsequent data transmission decoding with the channel estimated from interference-free preambles versus that from concurrent preambles. The experimental results reveal the followings: first, taking 13 taps (6 on each side of the significant tap) is sufficient for channel estimation in all SNRs. Taking fewer taps can result in a degradation in decoded SNR because the recovered CSI is less accurate. The number of taps required also determines the preamble length for sufficient measurement. Second, with a sufficient number of taps such as 13 taps in Figure 7(a), the decoding performance with concurrent preambles is better than interference-free preambles. This shows that MIMO/CON which exploits the channel sparsity in fact can help filter out noise in channel estimation. This is because a nonzero value appearing in a large-delay tap is automatically suppressed during sparse signal recovery. Third, when the signal SNR
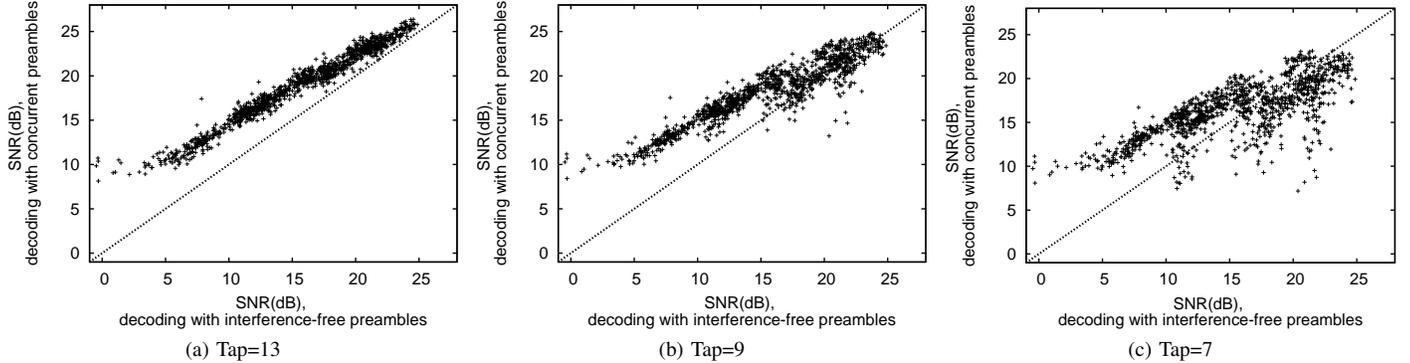
Fig. 7: MIMO decoding performance using CSI estimated from concurrent preambles in 4x4 MIMO. Taking 13 taps is sufficient for reconstructing accurate CSI. Using fewer taps results in degradation in decoding performance, especially when the signal SNR is high.

is high, more taps are required to achieve relatively good decoding performance. For the case with low signal SNR, the accuracy of channel estimation is limited by noise and thus taking fewer taps is sufficient.

B. Impact of multi-antenna diversity in improving decoding efficiency

In this subsection, we conduct experiments to measure the benefits of multi-antenna diversity in sparse recovery using the same 4×4 MIMO setting. The experiments are repeated with different levels of SNRs, which are measured from interference-free packets. We configure similar SNR for all senders.

As discussed in Section IV-B, incorporating measurements from different antennas can make the proxy vector more robust and facilitate support selection. To simplify discussion, we focus on identifying active senders from the proxy vector using the most significant tap. The decoding algorithm can proceed to identify other taps after knowing the senders. We measure the minimum α so that the top αK elements in the reduced proxy vector include all active senders. A larger α indicates more noise in the proxy vector and is more difficult for support selection. A proxy vector with $\alpha = 1$ is optimal, meaning that the top K components in the vector correspond exactly to the K active senders.

In Figure 8, we plot the distribution of α over multiple experiments with varying numbers of receive antennas. We make the following observations: first, the distribution of α under different SNRs is similar, with α slightly larger in low SNR. With 1 receive antenna, more than 50% of the experiments have $\alpha = 1$; however, around 20% of the experiments α is larger than 2. This shows that although in general the proxy vector can identify active senders correctly, at times it cannot especially when the SNR is low. Second, incorporating measurements from 2 antennas, the proxy vector quickly becomes more robust in the high SNR case. In low SNR, it takes 3 antennas to achieve similar performance. When measurements from all 4 antennas are included, almost all of the experiments can identify all senders correctly with $\alpha = 1$.

TABLE I: CSI recovery rate from concurrent preambles.

Number of Antennas	1	2	3	4
Low SNR	89%	99%	99%	100%
Medium SNR	99%	100%	100%	100%
High SNR	100%	100%	100%	100%

Facilitating support selection also improves the algorithm in recovery rate. Shown in Table I, the algorithm does not converge to the right solution in 11% of the experiments with 1 antenna in low SNR. In contrast, when all 4 antennas are included, the algorithm always converges to the correct solution.

C. FFT size of concurrent preambles

The FFT size of concurrent preambles needs to grow with a larger MIMO system or a higher bandwidth with more significant taps in order to maintain sufficient measurements. We conduct simulations to study the number of active senders that can be supported given a preamble length. The simulation setting is the same as in Section IV-B while we change the number of nonzero taps to be 13 as found in the software-defined radio based experiments.

Fundamentally the FFT size has to be greater than the number of unknowns that needs to be solved. This fundamental limit is plotted as the vertical dotted lines in Figure 9. When 1 antenna is incorporated for recovery, the preamble FFT size of 128 and 256 can support up to 4 and 8 active senders, respectively. When 4 antennas are used, the same preamble FFT size can support up to 7 and 14 active senders. We note that in this case, the preamble FFT size has an oversampling ratio of 1.4, which is close to the optimal of 1. We thus conclude that the overhead of concurrent preambles is close to the minimum.

D. Throughput improvement

In this subsection, we investigate the throughput improvement of MIMO/CON. Although we can implement the functionalities of MIMO/CON on software radios, the current hardware system we have in lab cannot run fast enough to support
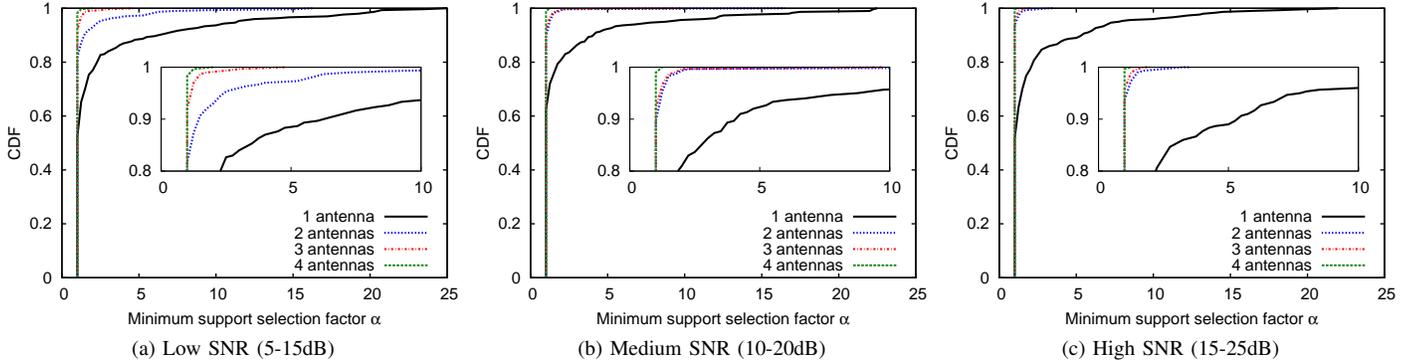
Fig. 8: Impact of multi-antenna diversity in improving decoding efficiency. By incorporating just a few measurements from different antennas, one can estimate CSI from concurrent preambles in only one iteration of the decoding algorithm. Each plot includes a blown-up subplot to show details of CDF for α near 1.
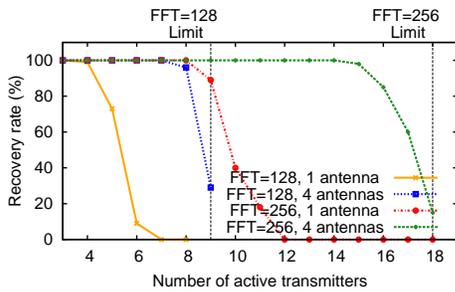

Fig. 9: FFT size of concurrent preambles. Vertical dotted lines indicates the fundamental limit on the active senders that a particular FFT size can support.

carrier sensing and real-time concurrent preamble decoding for a large number of active users. Thus we turn to software simulators to study MIMO/CON's throughput performance for many users scenario. We implemented an event-driven simulator, which assumes standard 802.11n parameters: 28 μs DIFS, 10 μs SIFS, 20 μs PHY preamble, and 9 μs slot time. We assume a standard 1500-byte data packet size and a 14-byte ACK packet size.

We compare MIMO/CON with SAM [20], a staggered access design for MU-MIMO systems. In addition, to evaluate the effectiveness of delay packet decoding, we compare MIMO/CON with and without the feature turned on. For simplicity, we first assume the optimal transmission probability that leads to the highest aggregated throughput is known in the simulation, but will relax this assumption. The throughput under the optimal scheduler is also plotted for reference.

In the first two simulation experiments, we simulate an environment with 20 senders that always have data packets to send. The senders are assumed to have the same PHY data rates, 13Mbps and 52Mbps, to represent the low and high SNR regimes, respectively. Results are shown in Figure 10. First, staggered access performs well when there are fewer antennas; however the throughput quickly saturates when the number of antennas increases, which is due to serializing channel

contention. Assuming an average backoff period of 10 slots, the maximum number of overlapping packets for staggered access is then 8.5 and 2.7 under 13Mbps and 52Mbps. No further throughput improvement will be possible when the number of receive antennas is beyond this limit. Second, MIMO/CON without delay packet decoding (MIMO/CON basic) performs the worst when the number of receive antennas is small. This loss of efficiency comes from the difficulty in balancing between overbooking and underutilizing the channel. However, even without delay packet decoding, the throughput with MIMO/CON scales well to a larger number of receive antennas. Third, delay packet decoding can mitigate the channel utilization problem. With delay packet decoding, the throughput discrepancy between MIMO/CON and the optimal scheduler is reduced by 50%. The remaining gap is mainly due to channel underutilization. Lastly, we add AIMD control to MIMO/CON to dynamically adjust the transmission probability. The results show that adding AIMD is effective and delivers similar throughput performance. Overall, with 5 receive antennas, MIMO/CON can improve the throughput of staggered access by 140% under 13Mbps; a larger improvement of 210% is observed with a higher 52Mbps data rate.

For a heterogeneous scenario where senders have different SNR and thus different data rates, we note that MIMO/CON will still deliver better throughput scalability than SAM. However, the aggregated network throughput of both approaches will be bottlenecked by the sender of the lowest rate. In other words, the throughput performance will be as if all senders operate at the lowest data rate. This is because concurrent transmissions are not independent of each other and must be transmitted in groups. Therefore the time spanned by a set of concurrent transmissions is dominated by the slowest sender. Similar problems exist when senders send packets with various sizes. To improve the efficiency, one may need to cluster senders according to their SNR, or bound the packet duration to avoid few slow senders hampering overall throughput.

Finally, to understand the scalability of MIMO/CON, we conduct a larger scale simulation with 100 senders and up to
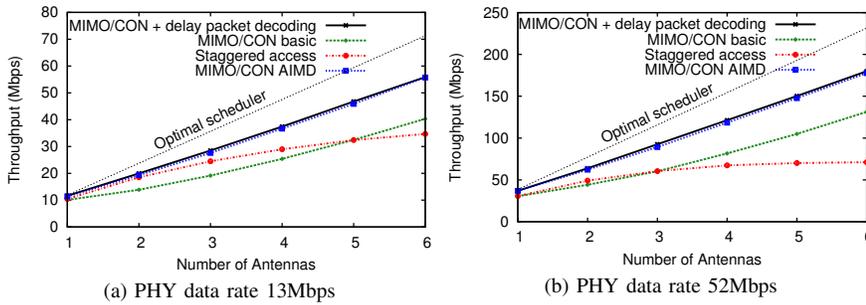
(a) PHY data rate 13Mbps

(b) PHY data rate 52Mbps

Fig. 10: Throughput of MIMO/CON with 20 nodes.

Fig. 11: Throughput of MIMO/CON with 100 nodes and 13Mbps PHY.

32 receive antennas on the AP. Figure 11 shows the throughput with PHY data rate 13Mbps. The trends of the curves stay similar as those in Figure 10. The throughput of staggered access is limited by the packet duration and thus cannot be improved by the additional antennas. MIMO/CON, in contrast, has no such contraints, and scales well with the increased MIMO degrees-of-freedom.

VIII. CONCLUSIONS

In this paper, we have proposed an ambitious scheme for the purpose of achieving full utilization of uplink capacity offered by an AP equipped with many receive antennas. A key to our scheme, called MIMO/CON, is a novel decoding method which can estimate CSI and identify active senders from concurrently received packet preambles. While it may not be surprising that CSI can be derived from fully over-lapped preambles with joint estimation methods, the task becomes significantly more difficult when distributed senders are loosely synchronized and not subject to mutual or central coordination. MIMO/CON leverages the recent theory of compressive sensing to overcome this challenge. In the MAC layer, MIMO/CON addresses the channel utilization issue by a novel strategy called delay packet decoding that exploits normal MAC layer retransmission mechanism to recover otherwise undecodable packets in a collision. In summary, MIMO/CON is a method that allows efficient multiuser MIMO networking among distributed users without requiring strict synchronization and coordination. We believe the proposed concurrent channel access and estimation schemes or similar approaches will be important components for future high-throughput multiuser MIMO networks.

REFERENCES

[1] Ettus research. In *www.ettus.com*.
[2] IEEE 802.11n-2009. Amendment 5: Enhancements for higher through-put. October 2009.
[3] J. Andersen, T. Rappaport, and S. Yoshida. Propagation measurements and models for wireless communications channels. *IEEE Commun. Mag.*, 33(1):42–49, 1995.
[4] W. Bajwa et al. Compressed channel sensing: A new approach to estimating sparse multipath channels. *Proc. IEEE*, 98(6):1058–1076, 2010.
[5] E. Candes and T. Tao. Decoding by linear programming. *IEEE Trans. Inf. Theory*, 51(12):4203–4215, 2005.
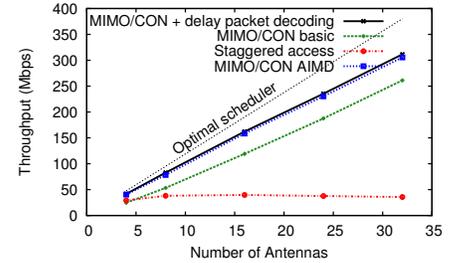[6] M. Davenport et al. Introduction to compressed sensing. *Electrical Engineering*, pages 1–68, 2011.
[7] D. Gesbert et al. Shifting the MIMO paradigm. *IEEE Signal Process. Mag.*, 24(5):36–46, 2007.
[8] S. Ghassemzadeh et al. Measurement and modeling of an ultra-wide bandwidth indoor channel. *IEEE Trans. Commun.*, 52(10):1786–1796, 2004.
[9] S. Gollakota and D. Katabi. Zigzag decoding: combating hidden terminals in wireless networks. In *ACM SIGCOMM 2008*.
[10] S. Gollakota et al. Interference alignment and cancellation. In *ACM SIGCOMM 2009*.
[11] A. Gudipati and S. Katti. Strider: automatic rate adaptation and collision handling. In *ACM SIGCOMM 2011*.
[12] L. Li et al. Retransmission≠ repeat: simple retransmission permutation can resolve overlapping channel collisions. In *ACM MobiCom 2010*.
[13] K. Lin el al. Random access heterogeneous MIMO networks. In *ACM SIGCOMM 2011*.
[14] E. Magistretti et al. WiFi-Nano: reclaiming WiFi efficiency through 800 ns slots. In *ACM MobiCom 2011*.
[15] T. Marzetta. Noncooperative cellular wireless with unlimited numbers of base station antennas. *IEEE Trans. Wireless Commun.*, 9(11):3590–3600, 2010.
[16] M. Morelli et al. Synchronization techniques for orthogonal frequency division multiple access (OFDMA): A tutorial review. *Proc. IEEE*, 95(7):1394–1427, 2007.
[17] D. Needell and J. Tropp. CoSaMP: Iterative signal recovery from in-complete and inaccurate samples. *Applied and Computational Harmonic Analysis*, 26(3):301–321, 2009.
[18] J. Romberg. An overview of recent results on the identification of sparse channels using random probes. In *IEEE CDC 2010*.
[19] K. Tan el al. Fine-grained channel access in wireless lan. In *ACM SIGCOMM 2010*.
[20] K. Tan el al. SAM: enabling practical spatial multiple access in wireless LAN. In *ACM MobiCom 2009*.
[21] J. Tropp and A. Gilbert. Signal recovery from random measurements via orthogonal matching pursuit. *IEEE Trans. Inf. Theory*, 53(12):4655–4666, 2007.
[22] D. Tse and P. Viswanath. *Fundamentals of wireless communication*. Cambridge Univ Pr, 2005.
[23] J. Van de Beek, O. Edfors, M. Sandell, S. Wilson, and P. Borjesson. On channel estimation in OFDM systems. In *IEEE VTC 1995*.
[24] S. Yoon, I. Rhee, B. Jung, B. Daneshrad, and J. Kim. Contrabass: Concurrent transmissions without coordination for ad hoc networks. In *IEEE INFOCOM 2011*.