# Measurement-Based Transmission Schemes for Network MIMO

Krishna C. Garikipati
Dept. of EECS
University of Michigan–Ann Arbor
gkchai@eecs.umich.edu

Kang G. Shin
Dept. of EECS
University of Michigan–Ann Arbor
kgshin@eecs.umich.edu

## ABSTRACT

The scaling of network MIMO (netMIMO) comes with the challenges of interference that arise from the aging of the Channel State Information (CSI). Particularly, in scenarios of high channel mobility—where netMIMO is expected to be used—the performance degradation from interference can be severe. In this paper, we design transmission schemes to address the two sources of CSI aging: delay in acquiring CSI feedback, and transmission period over which CSI remains unchanged. We introduce a two-phase training and feedback protocol that balances CSI aging across users by allowing users with high interference to have a shorter feedback delay. We also introduce an adaptive adjustment to the transmission length to reduce the decoding errors caused from interference. The proposed protocols adapt to the measurements collected from the users and hence applicable to various mobility scenarios. Our evaluation in a real and synchronized netMIMO setup shows significant gains (5–10dB) in user performance, especially in mobile channels.

## Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design—*wireless communications*; C.2.2 [**Computer-Communication Networks**]: Network Protocols

## General Terms

Design, Experimentation, Performance

## Keywords

Multi-user MIMO networks, CSI feedback, CSI aging

## 1. INTRODUCTION

A distributed multiuser MIMO (MU-MIMO) network, commonly known as *network MIMO* (netMIMO), has the potential of achieving gigabit capacity in dense wireless networks, such as enterprise networks, where most of the wireless traffic is generated and received. A typical netMIMO setup consists of a central controller that does the baseband processing, and a set of distributed Access Points (APs) that concurrently transmit data to multiple users or stations (STAs). In contrast to traditional multi-cell WLAN deployments where adding more cells increases co-channel interference, netMIMO eliminates the inter-cell interference, or converts it into useful information through tight integration of APs into a giant virtual MIMO transmitter, thus preserving capacity gains from the additional cells. The network capacity scales linearly with the number of transmitters, or APs [1], and therefore, a simple installation of a large number of APs is expected to meet the bandwidth requirements of modern wireless networks.

However, in reality, realizing the true gains of a large-scale netMIMO has been challenging for a number of reasons. First, the distributed APs or transmitters have to be accurately synchronized in time and frequency through periodic exchange of synchronization frames. Second, fast backhaul links that connect the APs to the controller and support real-time baseband processing have to be deployed. Last, the fundamental wireless capacity that is under-achieved due to the channel feedback overhead and channel estimation errors has to scale with the size of the network. Note that although reciprocity of the wireless channel precludes the need for channel feedback, the calibration requirements [2] make the reciprocity assumption untenable for netMIMO deployments.

Nevertheless, recent advances have shown that high-capacity backhaul can be realized with optical fiber technology, while synchronization was achieved with a large number of APs [3]. On the other hand, the problem of STA interference that arises from the errors between estimated and observed channel remains a major challenge [4].

In netMIMO, as in any MU-MIMO network, the APs broadcast training frames that are used by the STAs to compute their respective Channel State Information (CSI). Each STA then transmits its CSI, either compressed or uncompressed, back to the APs during the feedback phase. This implies the delay in acquiring the CSI feedback from STAs grows with the number of participating STAs. In certain scenarios, the excessive feedback delay may see the channel undergo significant changes. Subsequently, when data frames are transmitted, one or more STAs may see a channel that is completely different from the channel used for computing the CSI. This phenomenon is technically known as *CSI aging*, and the delay between CSI estimation and ac-


*MobiHoc'14,* August 11–14, 2014, Philadelphia, PA, USA.

http://dx.doi.org/10.1145/2632951.2632980 .

tual transmission is called the *CSI delay*. Since netMIMO relies on zero-forcing techniques based on past CSI, improper cancellation of streams due to CSI aging leads to interference at the STAs. This unwanted interference can result in significantly lower throughput than expected from perfect cancellation. In addition, when data frames are sent for a long period of time using the past CSI, the interference may get accentuated due to further increase in CSI delay. Therefore, both the feedback delay and the length of transmission are important factors that determine the effect of CSI aging on netMIMO performance.

In this paper, we propose transmission schemes for netMIMO that are designed to address the following two questions associated with CSI aging: Q1) *how to get CSI feedback without adversely affecting one or more STAs' performance?*; Q2) *how to determine the length of transmission over which the CSI remains unchanged?* Since the effects of CSI aging are strongly dependent on the deployment, we propose a protocol that adapts to the measurements received from the STAs. We implement and evaluate these schemes in a real netMIMO testbed with real-world wireless channels. This paper makes the following contributions:

- A non-heuristic, non-threshold-based training and feedback protocol, and an adaptive transmission scheme that effectively overcome the effect of CSI aging.
- A novel measurement-based approach where interference observed at the STAs is used in adapting the netMIMO protocol. We show that interference is a useful metric, and support our claims with analysis and observations.
- Construction of a fully synchronized netMIMO testbed on WARP Software-Defined Radio [5] with MIMO-OFDM processing, and development of a new channel-measurement framework with custom hardware and software design.

The rest of the paper is organized as follows. Ssection 2 motivates the problem of CSI aging while Section 3 specifies the netMIMO model. Section 4 describes our proposed transmission schemes which are followed by their implementation and evaluation in Ssection 5. Finally, we discuss the related work in Section 6 and conclude the paper with Section 7.

# 2. BACKGROUND AND MOTIVATION

We consider an example netMIMO deployment in Fig. 1, where a group of APs, all connected to a controller through a common backhaul link, concurrently transmit data of multiple STAs. To suppress the interference that may occur when a STA receives unwanted data of other STAs (e.g., STA1 receives signals of STAs 2–4), netMIMO uses coordinated transmission like zero-forcing and block-diagonalization [6]. This coordination is possible if the baseband transceiver processing is implemented at the controller, and the APs effectively function as remote antennas.

## 2.1 Protocol Design

In netMIMO, the downlink CSI of STAs that is required for canceling interference is obtained through a training and feedback protocol. We assume a mechanism similar to the IEEE 802.11ac MU-MIMO explicit feedback protocol [7, 8]. Training happens via a sequence of Null Data Packet (NDP)

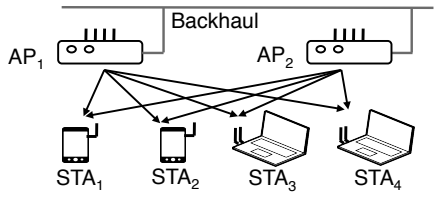



**Figure 1: APs transmit to multiple STAs**

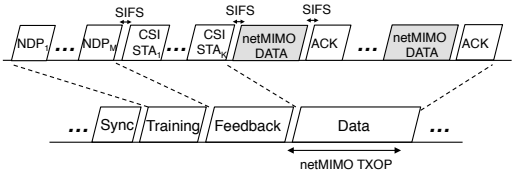



**Figure 2: Timeline of a netMIMO transmission**

| | | | |
|---|---|---|---|
| CSI compression | Givens | Bits per angle | 8 |
| Feedback rate | MCS0 | AP,STA Antennas | 2,1 |
| SIFS duration | 16$\mu$s | Tx Power | 18 dBm |
| NDP duration | 30$\mu$s | Bandwidth | 20MHz |
| CSI Grouping | 2:1 | Processing | ZF |

**Table 1: netMIMO protocol parameters**

sounding frames sent by the netMIMO APs. Each STA estimates the channel matrix by listening to the NDP frames and computes its CSI. The format of the CSI may vary, e.g. channel matrix, beamforming matrix, precoding matrix indicator, signal-to-noise ratio, etc., and is specified by the controller. The STAs encode this CSI into CSI-feedback frames and transmit them to one or more APs in a pre-determined order. Once the controller obtains the CSI, it generates the precoding weights that are applied to the antennas of the APs to send the netMIMO data packets.

Fig. 2 illustrates the basic netMIMO transmission protocol. It begins when a sync frame is transmitted by the master AP to synchronize the multiple APs and to reserve the wireless medium [3], followed by a sequence of NDP frames in the training period. This is followed by sequential CSI feedback from the STAs (Poll frames for soliciting CSI feedback are not shown). The data packets of STAs are sent in the netMIMO Transmit Opportunity (TXOP) period, where the TXOP duration is determined during the channel reservation process. Once the STAs receive and decode their data frames, they send ACK/Block ACK frames to acknowledge reception.

**Remark**: The CSI may be encoded by the STA to reduce the amount of feedback overhead. For example, quantization [9], CSI grouping across subcarriers [7] and differential encoding [10] can be readily applied. In addition, matrix operations like Givens rotation [11] may be applied to reduce the size of channel representation.

## 2.2 Effect of CSI Aging

The wireless channel between two nodes varies with time due to relative motion between them, motion in the surroundings, random fading and other physical impairments. The rate of variation is generally captured through the notion of *coherence time*—defined as the time period during which the wireless channel remains almost constant. For example, the coherence time is of the order of hundreds of milliseconds for stationary nodes but can be much less ($<$ 10ms) for moving nodes [12]. It therefore follows that aging

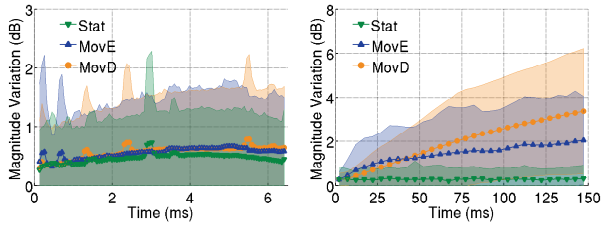



**Figure 3: Measured channel magnitude variations at micro (left) and macro (right) scale**

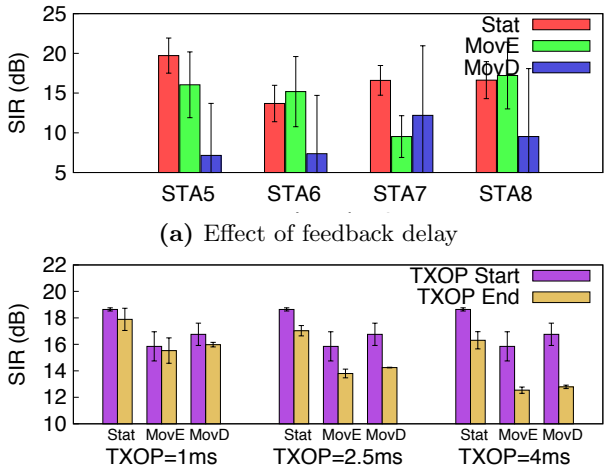
of CSI is inversely related to the coherence time. To study the effect of this CSI aging, we perform some preliminary measurements of the protocol in Section 2.1 on a netMIMO testbed (see Section 5 for details). We consider three mobility profiles for the STAs—*stationary* (Stat), *moving environment* (MovE) and *moving device* (MovD)—that capture a range of different channel characteristics. In each of these scenarios, the measured channel behavior (mean and deviation) across 32 wireless links as a function of time is shown by plots in Fig. 3. The variations in the channel magnitude is shown to increase over time for mobile channels but remain relatively stable for a stationary environment. The parameters for the netMIMO transmission protocol are chosen from the 802.11ac standard [7] and are shown in Table 1. In Section 3, we describe the transmit and receive processing. To isolate the effect of channel aging on the performance, we assume a noiseless channel estimation and an error-free feedback channel, which is justified by using the best feedback resolution (8 bits per angle), and the best modulation and coding scheme (MCS0) for uplink transmission.

**Nonlinear increase in feedback delay**. Table 2 lists the duration of the CSI feedback frame for each STA and the total feedback delay as the number of STAs is varied. In each scenario, the number of transmit antennas is taken equal to the number of STAs. The calculation, based on the parameters in Table 1, is done as follows: Given a total of $K$ transmit antennas and single receive antenna per STA, the CSI feedback is a $K \times 1$ unitary matrix (§3). This implies the total number of angles (both phase and rotation) after parameterization by Givens rotation is $2K - 2$ ( Eq. (30) in [11]). Therefore, the feedback size of each STA is given by $(2K - 2) \times$ #BitsperAngle $\times$#Subcarriers /#CSIgrouping. The feedback duration is then determined using the MCS0 (6Mbps) feedback rate, which is then summed for all $K$ STAs including the packetization and the inter-frame overheads to arrive at the total feedback duration. Clearly, in this design, the total feedback duration grows quadratically with the number of STAs.

| | 4 STAs | 6 STAs | 8 STAs | 10 STAs |
|---|---|---|---|---|
| Each (ms) | 0.224 | 0.288 | 0.480 | 0.608 |
| Total (ms) | 0.96 | 1.82 | 3.96 | 6.24 |

**Table 2: Duration of feedback frames**

We measure the Signal-to-Interference (SIR) values for a single STA in various scenarios in a 10–STA netMIMO (Fig. 4a). The total feedback delay here is 6.24ms, and as one can see, the SIR in the MovD environment is at least 50% less than the stationary environment. Since the SIR in each of the scenarios is generally highest in case of the stationary channel, it suggests that SIR degradation (4–15dB) in non-stationary channels is due to the interference from CSI aging.




**(a)** Effect of feedback delay

**(b)** Effect of TXOP duration

**Figure 4: Measured SIR at the STAs suggests that CSI delay leads to performance loss**

**Role of TXOP duration**. When the precoding weights based on past CSI are used in sending data packets for a prolonged period, it may also lead to CSI aging. This is confirmed with a netMIMO of 6 STAs as shown in Fig. 4b, where there is a marked difference in the SIR values at the start of TXOP and at the end of TXOP. This is particularly significant (=6 dB) as the duration of TXOP is increased to 4ms, where CSI aging corresponds to a total 5.82ms (including the initial feedback delay of 1.82ms).

## 2.3 Measurement-Based Approach

To overcome the effects of CSI aging requires changes to the netMIMO protocol. However, given that the indoor wireless channels are difficult to model because of their dependence on the topology and disturbances, we focus on a passive measurement-driven protocol that inherently accounts for the CSI delay of each STA.

**Why netMIMO?** Although channel variations are also applicable to a MU-MIMO network consisting of a single AP, we emphasize CSI aging only in context of netMIMO for two reasons. First, netMIMO is designed for high-density indoor environments like conference rooms, theaters, shopping malls, etc., that inherently accompany mobility. Second, as compared to the single AP installation, links between STA and APs in netMIMO are more diverse, which implies lower coherence, and therefore, a greater CSI aging effect.

# 3. NETWORK MIMO MODEL

We briefly describe the netMIMO signal processing of the transmit and receive baseband process at the APs and STAs, respectively. The system uses OFDM, and in the description that follows we drop the subcarrier index to indicate that it is applicable to each OFDM subcarrier. Assume there are $M$ APs, each with $N_m$ antennas, and $N_t = M \times N_m$ transmit antennas in total, that together serve $K$ STAs ($K \leq N_t$). Let $\mathcal{K} = \{1, \ldots, K\}$ denote the set of STAs each of which is equipped with $N_r$ receive antennas. The STA $k$ estimates the composite $N_r \times N_t$ channel matrix, $\tilde{H}_k$, from the $M$ NDPs in the training phase and computes the right singular matrix $V_k$. Assume that a single data stream is supported per STA and an equal amount of power is assigned to each

STA. In this case, the CSI of STA $k$ is the strongest right singular vector, $v_k$, of $V_k$ [13]. Each STA $k$ quantizes, compresses and encodes $v_k$ into the CSI feedback frame, and broadcasts it to the APs. The CSI feedback frame is decoded by the APs and the contents passed on to the controller. After collecting the CSI from every STA, the controller uses a zero-forcing (ZF) precoding to generate the precoding matrix, $W = \Gamma(\Gamma^\dagger\Gamma)^{-1}$, where $\Gamma = [v_1\ v_2 \dots\ v_K]$ is a matrix of singular vectors, and $(\cdot)^\dagger$ represents the Hermitian transpose. The precoding weights, $\tilde{w}_1, \tilde{w}_2, \dots, \tilde{w}_K$, are the normalized column vectors of $W$ used by the controller to modulate the $K$ STA data streams. The APs then collectively transmit these modulated streams which are received and processed by each STA as a single netMIMO frame.

Let $x_k$ denote the data of STA $k$, then the signal $y_k$ received by STA $k$ is given by

$$y_k = H_k\tilde{w}_k x_k + \overbrace{\sum_{i\neq k} H_k\tilde{w}_i x_i}^{\text{interference}} \quad (1)$$

where $H_k$ is the observed channel of STA $k$ in netMIMO data packet. In this equation, the term $R_k = \sum_{i\neq k} H_k\tilde{w}_i x_i$ is the unwanted interference from other STAs' data while the term $H_k\tilde{w}_k x_k$ is the desired signal. In ideal conditions, the interference $R_k$ is 0 because the estimated channel $\tilde{H}_k$ is equal to the observed channel $H_k$, and the precoding weights $\tilde{w}_i$ satisfy $H_k\tilde{w}_i = 0$ for all $i \neq k$ . However, in practice, $H_k$ and $\tilde{H}_k$ are different due to the time-lag between them (CSI aging effect) which results in non-zero interference.

In this paper, we assume all STAs are equipped with a single antenna, i.e., $N_r = 1$. We also assume the data symbols, $x_1, \dots, x_K$, are zero-mean Gaussian i.i.d. with equal average power, $P/K$, where $P$ is the total input power. Therefore, the interference power, $I_k$, observed by STA $k$ is

$$I_k = \sum_{i\neq k} \frac{P}{K}|H_k\tilde{w}_i|^2 \quad (2)$$

while the SIR is given by

$$\text{SIR}_k = \frac{|H_k\tilde{w}_k|^2}{\sum_{i\neq k}|H_k\tilde{w}_i|^2} \quad (3)$$

Unless specified, the term *interference* in this paper refers to the interference power in Eq. (2). Each STA $k$ decodes its data $\hat{x}_k$ with a Zero-Forcing (ZF) receiver given by $\hat{x}_k = (H_k\tilde{w}_k)^\dagger y_k/|H_k\tilde{w}_k|^2$. Similar to the 802.11ac MU-MIMO protocol [7], the precoded channel terms, $H_k\tilde{w}_i$, needed for ZF/MMSE estimation are known from the VHT-LTF portion of the netMIMO packet. Therefore, for all practical purposes, a netMIMO STA is simply a MU-MIMO STA of existing WLAN deployments. We note that the channel noise in the above formulation, albeit present, has been ignored, considering the fact that netMIMO is designed for high SNR regime.

The interference at a STA arises from the mismatch between the observed channel, $H_k$, and the precoding weights derived from the estimated channel, $\tilde{H}_k$, and is therefore a good metric to quantify channel fluctuations ( §4.3). Furthermore, it may be approximated as: $|H_k\tilde{w}_k|^2\text{E}[|\hat{x}_k - x_k|^2]$, which is calculated at the receiver after demodulation by averaging over multiple OFDM symbols. We validate this approximation in our setup by precoding with random precoding vectors at Tx antennas and measuring the average symbol error after demodulation at a receiver.

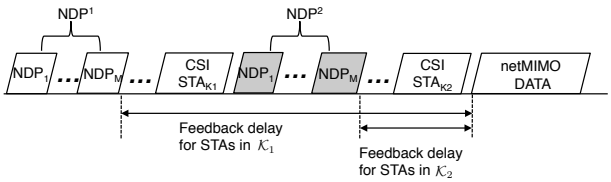



**Figure 5: The proposed training protocol with additional NDP frames inserted in the feedback sequence**

# 4. ADAPTIVE NETWORK MIMO PROTOCOL

We present a netMIMO transmission protocol that combats CSI aging by modifying the CSI delay of STAs. We use the interference experienced by the STAs to quantify CSI aging as the metric of interest. The first part of the protocol balances the interference across STAs through two-phase training in which STAs with mobile channels are accorded a smaller CSI delay. The second part of the protocol adjusts the TXOP duration in order to meet the decoding threshold required throughout the TXOP duration. We begin by describing our proposed modification of the training protocol.

## 4.1 Two-Phase Training

The idea of two-phase training is to have a second NDP broadcast from APs close to the start of data transmission. STAs with high channel mobility and therefore with large channel variation relative to the first NDP use the new NDP frames to estimate their channel. The training protocol is depicted in Fig. 5. Here NDP$^1$ and NDP$^2$ refer to the original and the additional set of training frames, respectively. The set of STAs, denoted by $\mathcal{K}_1$, estimate the downlink channel w.r.t. NDP$^1$, and broadcast their CSI feedback, similar to the standard feedback procedure. The remaining set of STAs, $\mathcal{K}_2$, estimate the channel from NDP$^2$ and send their CSI feedback after NDP$^2$ is transmitted. The partitions, $\mathcal{K}_1$ and $\mathcal{K}_2$, therefore identify the two phases of training, and are specified by the controller at the start of the netMIMO transmission.

Let $T_{\text{NDP}}$ be the collective duration of the set of NDP frames and let $T_k$ be the fixed feedback duration of STA $k$. Given a partition $\mathcal{K}_2$, the feedback delay (CSI delay) of STAs in $\mathcal{K}_1$ is given by $T^1 = \sum_{k\in\mathcal{K}} T_k + T_{\text{NDP}}$, while for STAs in $\mathcal{K}_2$, the delay is given by $T^2 = \sum_{k\in\mathcal{K}_2} T_k$. Clearly, $T^1$ is fixed while $T^2$ varies with the size of $\mathcal{K}_2$. For simplicity, the constant overhead terms like SIFS are not shown. Let $\bar{I}_k(\mathcal{K}_2)$ be the average interference of STA $k$ where the average is taken over all OFDM symbols and subcarriers in the first data packet of the netMIMO TXOP. Its dependence on CSI delay, $T^2$, is indicated implicitly through the set $\mathcal{K}_2$. To keep analysis tractable, the duration of the data frames in netMIMO TXOP is fixed at 1ms. We ignore channel variations within the data frame and define the CSI delay w.r.t. the start of data frame, which is also the feedback delay of the protocol.

Initializing with a few mobile STAs in $\mathcal{K}_2$ results in a smaller feedback delay and, possibly less interference from CSI aging. But as more STAs are included in $\mathcal{K}_2$, the feedback delay, and possibly the interference, of STAs trained

with NDP$^2$ increases. Consequently, there exists an optimal partition that balances interference across all STAs, i.e., minimizes the worst case. Taking the objective function OBJ $= \max_{k\in\mathcal{K}} \bar{I}_k(\mathcal{K}_2)$, we express this as a minimization problem:

$$\min_{\mathcal{K}_2\subseteq\mathcal{K}} \{\max_{k\in\mathcal{K}} \bar{I}_k(\mathcal{K}_2)\}. \quad (4)$$

Although straightforward, the problem cannot be solved because the observed channel and interference of STAs are not known *a priori* during training. Therefore, we take a passive measurement-based approach. The current location of NDP$^2$ is decided based on interference measurements reported by the STAs in the ACK frames from previous netMIMO transmission. We use a greedy approach: At each step the STA with the maximum reported interference is trained with NDP$^2$ as long as the maximum interference decreases in the next step. Algorithm 1 shows the procedure for NDP placement. It is initialized by sending only NDP$^1$, i.e., $\mathcal{K}_2 = \emptyset$ and measuring the interference at STAs.

**Algorithm 1** NDP placement

1: *Initialize*: $t = 0$, $\mathcal{K}_1 = \mathcal{K}$,$\mathcal{K}_2 = \emptyset$, $\bar{I}_1(\emptyset)\ldots\bar{I}_K(\emptyset)$, $\bar{I}^0_\star = \max_k \bar{I}_k(\emptyset)$
2: **for** iteration t **do**
3: Find $\bar{I}^t_\star = \max_k \bar{I}_k(\mathcal{K}_2)$, STA $k_\star = \arg\max_k \bar{I}_k(\mathcal{K}_2)$
4: **if** $\bar{I}^t_\star > \bar{I}^{t-1}_\star$ and $\mathcal{K}_2 \neq \emptyset$ **then**
5: Remove the last placed STA from $\mathcal{K}_2$ into $\mathcal{K}_1$
6: **else**
7: Place STA $k_\star$ in $\mathcal{K}_2$ if $k_\star$ was not previously moved from $\mathcal{K}_2$ into $\mathcal{K}_1$
8: **end if**
9: Train STAs in $\mathcal{K}_1$ with NDP$^1$ and STAs in $\mathcal{K}_2$ with NDP$^2$. Complete netMIMO transmission.
10: Obtain average interference, $\bar{I}_k(\mathcal{K}_2)$, of each STA $k$
11: **end for**

At every iteration of the NDP placement algorithm, the maximum observed interference among STAs is compared with previous maximum interference (line 4). If there is a decrease, the algorithm tries to further decrease OBJ by training the STA with maximum interference with NDP$^2$ (line 7). If there is an increase in OBJ, the last placed STA in $\mathcal{K}_2$ is shifted to $\mathcal{K}_1$. Intuitively, the NDP placement algorithm addresses CSI aging by reducing CSI delay of some STAs while increasing the CSI delay of the rest by a negligible $T_{\text{NDP}}$ period ($\approx$ 0.2ms for a 5-AP netMIMO).

## 4.2 Adaptive TXOP sizing

The netMIMO transmit opportunity (TXOP) is the common transmission period in which a single or a burst of netMIMO data packets of STAs are transmitted. Since the transmission is based on precoding from past CSI, a limit on the duration of TXOP is required to avoid the interference that arises from excessive CSI delay. A similar TXOP limit has been defined in 802.11 EDCA mechanism, albeit to improve MAC-level efficiency. Also, since different STAs have different channel characteristics, the interference in the TXOP can be different for each STA. This suggests that each STA needs to have its own TXOP limit within the netMIMO TXOP according to its tolerance towards interference.

We propose a netMIMO protocol that adapts the TXOP limit of each STA while satisfying its decoding requirements. The basic idea is to observe the interference in the TXOP of a previous netMIMO transmission to set the limit for the next, as shown in Fig. 6. Since a TXOP is made up of multiple data packets, which are also utilized in sending the interference reports, the TXOP limit is specified in the integral number of constant sized 1ms data packets. Algorithm 2 shows the procedure for the proposed netMIMO TXOP Adjustment that is run at every netMIMO transmission.

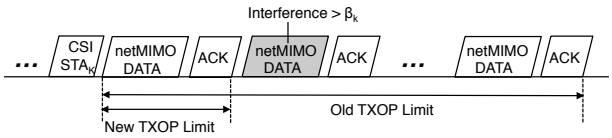



**Figure 6: Adjusting TXOP based on the interference**

**Algorithm 2** TXOP Adjustment

1: *Initialize*: $t = 0$, $L_k$ = TXOP Limit of STA, $\beta_k$ = interference threshold of STA $k$
2: **for** iteration t **do**
3: Obtain average interference, $\bar{I}^e_k$, from the end of TXOP of STA $k$
4: **for** each $k$ with $\bar{I}^e_k \geq \beta_k$ **do**
5: Set $L_k$ to the first time instant ($>$ minimum TXOP limit) in TXOP where interference of STA $k$ exceeds $\beta_k$
6: **end for**
7: **for** each $k$ with $\bar{I}^e_k < \beta_k$ **do**
8: $L_k \leftarrow \max(L_1, \ldots, L_K)$
9: **end for**
10: Set netMIMO TXOP Limit, $L = \max(L_1, \ldots, L_K)$
11: **end for**

The algorithm initializes with the default TXOP duration $L_k$ and interference threshold $\beta_k$ of each STA $k$. This threshold depends on the fixed modulation and coding scheme (MCS) used by the STA in the TXOP. Its calculation is given in Section 4.3. In lines 4–5, the algorithm decreases the TXOP Limit of STA $k$ when it exceeds its interference threshold. On the other hand, the TXOP limit of STA whose interference is below the threshold is set to the maximum value (line 8) of $L$ which is also the netMIMO TXOP limit. Similar to MU-MIMO transmission in 802.11ac [7], zero-padding is done for STAs whose TXOP limit is less than the transmission duration.

In summary, the two-phase training mitigates the interference at the start of TXOP while the TXOP Adjustment algorithm tries to reduce the interference that shows up towards the end of transmission. Both of these approaches therefore complement each other.

## 4.3 Analysis

We now present some key insights into the underlying variables of CSI delay and interference. We back these insights with measurements obtained from our testbed. We then show that the proposed protocol follows directly from these observations.

**Insight 1**(Independence): Interference at a STA is independent of CSI delay of other STAs.

*Explanation*: This follows from the fact that given the precoding weights $\tilde{w}_1, \ldots, \tilde{w}_K$ and the estimated channel $\tilde{H}_k$ of STA $k$, the interference $I_k$ is only dependent on its observed channel $H_k$, and is therefore independent of CSI delays of other STAs.

**Insight 2**(Monotonicity): The average interference experienced by a STA is directly related to its channel variations. Moreover, it increases *roughly* monotonically with the CSI delay of the STA.

*Explanation*: We can express the observed channel of STA $k$ in terms of the estimated channel and the error $e_k$ as:

$$H_k = \tilde{H}_k + e_k \tag{5}$$

where $e_k$ is a zero-mean spatially white gaussian error term. The zero-mean behavior was indeed verified in our testbed measurements. Next, consider a single data frame which is made up of a sequence of OFDM symbols. For a single subcarrier, we can express the expected value of interference (using Eq. (2)) as follows:

$$\mathrm{E}[\sum_{i\neq k}\frac{P}{K}|(\tilde{H}_k+e_k)\tilde{w}_i|^2] \overset{(a)}{=} \mathrm{E}[\sum_{i\neq k}\frac{P}{K}|e_k\tilde{w}_i|^2]$$
$$= \sum_{i\neq k}\frac{P}{K}\mathrm{E}[(e_k\tilde{w}_i)^\dagger(e_k\tilde{w}_i)] = \sum_{i\neq k}\frac{P}{K}\tilde{w}_i^\dagger\mathrm{E}[e_k^\dagger e_k]\tilde{w}_i$$
$$\overset{(b)}{=} \sum_{i\neq k}\frac{P}{K}\tilde{w}_i^\dagger\sigma_k^2 I\tilde{w}_i = \sum_{i\neq k}\frac{P}{K}\tilde{w}_i^\dagger\tilde{w}_i\sigma_k^2 \overset{(c)}{=} C\sigma_k^2$$

where $C=(K-1)P/K$ is a constant; (a) follows by using $\tilde{H}_k\tilde{w}_i = 0$ for all $i \neq k$, (b) is result of the white gaussian error term $e_k$, and (c) uses $\tilde{w}_i^\dagger\tilde{w}_i = 1$ for all $i$. This implies the observed average interference* that is obtained from multiple OFDM symbols of STA $k$ (further averaged over subcarriers to mitigate noise) can be approximated as

$$\bar{I}_k \approx C\sigma_k^2 \tag{6}$$

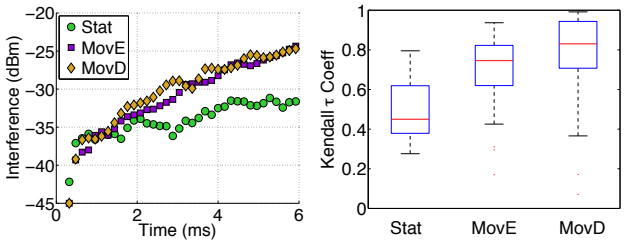
Therefore, the average interference at a STA is directly proportional to the power of channel variations, $\sigma_k^2$. This suggests that as channel variations grow larger with the CSI delay (see Fig. 3 for instance), the interference should also increase. To investigate if this indeed holds, we experimentally measure the interference as a function of STA's CSI delay under different scenarios. Fig. 7a gives an example of the observed STA interference in a 10-STA netMIMO setup. As expected, the interference in general increases with CSI delay, and more so in case of non-stationary environments. We run through these observations with the Kendall $\tau$ rank correlation test [14] which is non-parametric test to measure monotonicity between two variables. A correlation coefficient close to $+1$ indicates a monotonically increasing relationship while a value close to $-1$ suggests a dependency that is monotonically decreasing. Fig. 7b plots the range of Kendall $\tau$ rank correlation coefficient values calculated for the interference with respect to the CSI delay observed over 10 runs. The significance level was set to 0.01. In the majority of runs and under different channel conditions, we observe a rank correlation coefficient, $\tau > 0.7$, which confirms our insight on the monotonic nature of interference as a function of CSI delay. In what follows, we make use of the assumptions on independence and *strict* monotonicity, and make another simplifying assumption that channel behavior remains constant across multiple iterations.

PROPOSITION 1. *The NDP placement algorithm finds the optimal solution to Eq. 4 within $K$ iterations.*

PROOF. We first show that NDP placement does converge. By construction, at every step, a STA from $\mathcal{K}_1$ is moved to $\mathcal{K}_2$, unless the maximum interference among STAs increases. The CSI delay of STAs in $\mathcal{K}_1$ is given by $T^1 = \sum_{k\in\mathcal{K}} T_k + T_{\text{NDP}}$ which remains constant, so interference of STAs in $\mathcal{K}_1$ also remains constant by assumption. On the other hand, from the monotonicity assumption, the interference of STAs in $\mathcal{K}_2$ increases as the CSI delay $T^2 = \sum_{k\in\mathcal{K}_2} T_k$ increases which happens when a STA is moved to $\mathcal{K}_2$. Clearly, if at any point $k_\star \in \mathcal{K}_2$, then partitions don't change further. Also, if at any iteration $t$, $\bar{I}_\star^t > \bar{I}_\star^{t-1}$, STA $k_\star$ form iteration $t-1$ will be moved back to $\mathcal{K}_1$. Subsequently, in iteration $t+1$ STA $k_\star$ will continue to remain in $\mathcal{K}_1$. Therefore, the partitions converge in both cases.

To show that the converged partition, $\mathcal{K}_1, \mathcal{K}_2$, is optimal, assume there exists another partition $\mathcal{G}_1, \mathcal{G}_2$ that does strictly better. Let STA $k_\star$ and STA $r_\star$ have the maximum interference in the two scenarios, respectively. It follows that $I_{r_\star}(\mathcal{G}_2) < I_{k_\star}(\mathcal{K}_2)$. For any $k \in \mathcal{K}_2$, it is true that $k \in \mathcal{G}_2$ because otherwise if $k \in \mathcal{G}_1$, then $I_k(\mathcal{G}_2) \leq I_{r_\star}(\mathcal{G}_2) < I_{k_\star}(\mathcal{K}_2)$ from assumption. Further, $I_{k_\star}(\mathcal{K}_2) < I_k(\mathcal{G}_2)$, because at some point, $T'$, in NDP placement STA $k$ would be moved from the first partition and the maximum interference strictly decreases at each step, and also because the CSI delay of STA $k$ before $T'$ is same as with the partitions $\mathcal{G}_1, \mathcal{G}_2$. As a result, $I_k(\mathcal{G}_2) < I_k(\mathcal{G}_2)$ which is a contradiction. Therefore, we have $\mathcal{K}_2 \subset \mathcal{G}_2$ since the two partitions cannot be the same. Now, consider two cases when $k_\star \in \mathcal{K}_1$ and $k_\star \in \mathcal{K}_2$. In the first, it follows that $k_\star \in \mathcal{G}_2$ because otherwise if $k_\star \in \mathcal{G}_1$ implies $I_{r_\star}(\mathcal{G}_2) \geq I_{k_\star}(\mathcal{G}_2) = I_{k_\star}(\mathcal{K}_2)$, which is not true. Now, consider, $I_{k_\star}(\mathcal{K}_2 \cup k_\star) > I_{k_\star}(\mathcal{K}_2) > I_{r_\star}(\mathcal{G}_2) > I_{k_\star}(\mathcal{G}_2)$, from the stopping condition of the algorithm. This is a contradiction since $\mathcal{K}_2 \cup k_\star \subseteq \mathcal{G}_2$. For the second case, since $k_\star \in \mathcal{K}_2 \subset \mathcal{G}_2$, it follows $I_{r_\star}(\mathcal{G}_2) \geq I_{k_\star}(\mathcal{G}_2) > I_{k_\star}(\mathcal{K}_2)$ which is also contradiction to our assumption. Consequently, there cannot exist another partition with does strictly better than the NDP placement algorithm.

*Average from a single sample path is used as approximation of mean across all paths




**(a)** Example of STA interference (averaged over 10MHz) vs. CSI delay

**(b)** Kendall $\tau$ coefficient for interference vs. CSI delay with 0.01 significance level

**Figure 7: Examples and results of statistical test show that interference increases with CSI delay**

PROPOSITION 2. *If the interference threshold $\beta_k$ of STA $k$ satisfies $\beta_k = \frac{P}{K}(\tilde{\Lambda}_k^{min})^2/\alpha_k$, where $\alpha_k$ is the SIR decoding threshold, and $\tilde{\Lambda}_k^{min}$ is the minimum largest singular value of $\tilde{H}_k$ across subcarriers, then the average SIR after TXOP Adjustment satisfies $\overline{SIR}_k \gtrapprox \alpha_k$ for each data frame.*

PROOF. The TXOP adjustment proceeds by doing a series of adjustments to the TXOP limit of each STA, and sets it either to the minimum TXOP limit or increases to the maximum of the TXOP limits of all STAs. Therefore, after a finite number of steps the TXOP limit of each STA $k$ converges to a constant value such that the average interference, $\bar{I}_k$, throughout the TXOP is less than $\beta_k$ (if not the case then it is set to the minimum possible TXOP limit and there are no performance guarantees). Consider a subcarrier

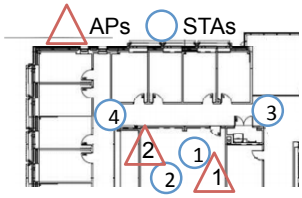



**Figure 8: Deployment of netMIMO testbed**

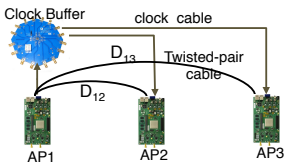



**Figure 9: Time and frequency synchronization**

of a data frame within the TXOP, we can express:

$$\mathrm{E}[SIR_k] = \mathrm{E}[\frac{P}{K}\frac{|(\tilde{H}_k + e_k)\tilde{w}_k|^2}{I_k}] \overset{(d)}{\approx} \mathrm{E}[\frac{P}{K}\frac{|\tilde{H}_k\tilde{w}_k|^2}{I_k}]$$
$$\overset{(e)}{=} \mathrm{E}[\frac{P}{K}\frac{\Lambda_k^2}{I_k}] \overset{(f)}{\geq} \mathrm{E}[\frac{P}{K}\frac{(\Lambda_k^{min})^2}{I_k}] \overset{(g)}{\geq} \frac{P}{K}\frac{(\Lambda_k^{min})^2}{\mathrm{E}[I_k]}$$
$$\overset{(h)}{\approx} \frac{P}{K}\frac{(\Lambda_k^{min})^2}{\bar{I}_k} \geq \frac{(\Lambda_k^{min})^2}{\beta_k} = \alpha_k$$

where (d) is an empirical approximation when $|H_k| >> |e_k|$, (e) uses the fact that $\tilde{w}_k$ is normalized vector of $W = (\Gamma^\dagger)^{-1}$ and $H_k$ has a single non-zero singular value, (f) follows by using the minimum singular value across subcarriers, (g) follows from Jensen's inequality and (h) is obtained by replacing expectation with the average sum. Therefore, for any data frame in a TXOP, we may approximate the average measured SIR measured across the frame's OFDM symbols and subcarriers as $\overline{SIR}_k \gtrapprox \alpha_k$.

# 5. EVALUATION

In this section, we provide the details of our netMIMO testbed, the evaluation methodology, the implementation of the proposed algorithms and their performance evaluation.

## 5.1 netMIMO Testbed

We have implemented a prototype of netMIMO using the WARP SDR platform [5]. The setup consists of a maximum of 5 APs (with 2 Tx antennas each), and 10 STAs, each with 1 Rx antenna. The controller function is implemented on a back-end server connected to the WARP boards via Ethernet with a Gigabit switch. We make use of the WARPLabv7.3 drivers with their efficient WARP-PC interface to interface with the WARP boards.

Fig. 8 shows the layout of the netMIMO testbed deployed in our Department building. To manage the large number of WARP boards we grouped them into separate AP and STA clusters. We have 2 AP clusters, and 3 STA clusters, with a combined total of 10 Tx and Rx antennas. The STA WARP boards are placed on a movable cart while carrying out the experiments.

### 5.1.1 Signal Processing

Following the 802.11n/ac standards, we have implemented a full-fledged netMIMO OFDM system. We use the basic 20MHz 64-fft OFDM (48 data subcarriers) design and upsample it by a factor of 2 to get an effective bandwidth of 10MHz. We use the channel 14 in the 2.4GHz ISM band that is unused by nearby devices.

At the start of every netMIMO transmission, a *sync* signal is sent by the controller to trigger the APs and the STAs to start their transmit and receive chain, respectively. Each AP broadcasts an 802.11 NDP packet composed of a short training field (STF) and a long training field (LTF). Each STA runs an auto-correlation algorithm to detect the STF and extracts the start time of the received packet. It also uses the STF to estimate the frequency-offset and corrects it accordingly. The LTF of an AP consists of OFDM training symbols on each Tx antenna of the AP, and is used by the STA to estimate channel w.r.t. each Tx antenna. The CSI feedback frames are not sent over-the-air and made implicitly available to the controller. Finally, a netMIMO data frame consisting of cyclic-shifted STFs, and precoded training and data is transmitted.

### 5.1.2 Synchronization

For the netMIMO signal processing to be valid, we require that all AP nodes in netMIMO are completely synchronized. By default, we take $\mathrm{AP}_1$ as the master AP and synchronize the rest of the APs with $\mathrm{AP}_1$.

*Time Synchronization*: At a high level, time synchronization means that all APs transmit simultaneously. However, in the context of netMIMO which uses OFDM, it requires that for each STA, the precoded OFDM frames transmitted by the APs as part of a single netMIMO packet must arrive within a cyclic prefix (CP) window. By ignoring the small differences in the propagation delay between APs and STAs, this implies that time synchronization in netMIMO is achieved when all APs begin transmission within a CP period. We ensure this in two steps.

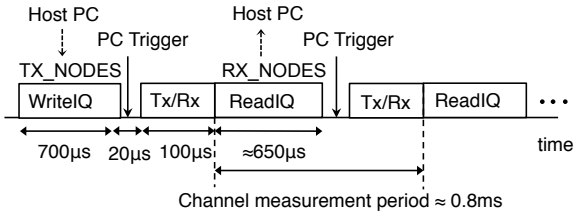
First, we use a twisted-pair cable assembly and connect the debug headers on the WARP board of $\mathrm{AP}_1$ with that of other APs. This ensures that the rest of the APs can be triggered by $\mathrm{AP}_1$ while $\mathrm{AP}_1$'s input trigger is set to a PC trigger that is sent over Ethernet. The twisted-pair cable which extends up to 30m in our setup has a propagation delay of the order of 100ns (typical value of CP is 400 or 800ns). As a result, we observe a significant time-lag in triggering from $\mathrm{AP}_1$. In our second step, we address this lag by introducing transmission offsets among the APs, i.e., we delay the start of packet transmission at an AP by a fixed duration. Let $D_j$ denote the delay of the twisted-pair cable between $\mathrm{AP}_1$ and AP $j$. Note that $D_1 = 0$. Also,let $O_j$ be the transmission offset of AP $j$. We find offsets $O_1$ to $O_M$ that satisfy the following constraint.

$$\max_{1\leq j\leq M}\{O_j + D_j\} - \min_{1\leq j\leq M}\{O_j + D_j\} \leq \mathrm{CP}. \qquad (7)$$

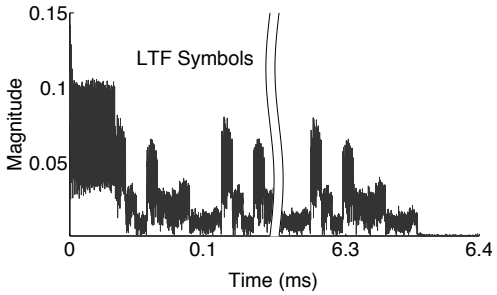
As the above constraint is under-specified, the offsets are found easily through a simple search. To allow for tolerance, we use CP/2 as the window size in our setup.

*Frequency Synchronization*: As illustrated in Fig. 9, we frequency-synchronize all netMIMO APs by sourcing their RF and sampling clock from the 40MHz reference clock of $\mathrm{AP}_1$. This is implemented with the CM-MMCX module of WARPv3 board that provides external interface to the reference clocks. We use the RG-174 coax cables that have an attenuation rating of 6.6dB/100ft at 50MHz for connecting the clock sources. Further, we use ADCLK954 boards as external clock buffers to prevent clock drift that may arise from the attenuation.

*Phase Synchronization*: Two WARP boards that are frequency synchronized may still differ in phase due to a random phase introduced by the Amplifier (LNA) circuit. Fortunately, this random phase remains constant once the circuit is powered on. Thus, a constant phase difference between two APs is easily corrected.




**(a)** Timeline of function calls in the WARP driver




**(b)** LTF symbol stream for channel estimation

**Figure 10: Details of the channel measurement framework that allows real-time evaluation of net-MIMO**

## 5.2 Evaluation Methodology

### 5.2.1 Trace-based evaluation

Our netMIMO setup uses the WARPLAB 7.3 library written in MATLAB to do signal processing. However, the execution time of MATLAB for netMIMO processing and WARP read/write exceeds 100ms and thus renders it useless for real-time evaluation. Therefore, we resort to a trace-based evaluation of netMIMO. The idea is to continuously collect the channel measurements between every pair of antennas in real time. This is achieved by repeatedly sending LTF symbols from the APs and processing the received symbols at the STAs to estimate the channel. We then play back the channel traces with the netMIMO transmission protocols through virtual timers which allows us to do an offline yet realistic evaluation.

### 5.2.2 Channel-Measurement Framework

To do an accurate trace-based evaluation of netMIMO, we require the $N_r \times N_t$ channel measurements to be done as frequently as possible. This implies that we need to either send/receive a single long packet (>10ms) of LTF symbols or send/receive the LTF symbols at a very fast rate (every 1ms or so). The current WARP design, however, has a maximum buffer size of only $2^{15}$ samples (airtime of 0.8ms at 40MHz). Moreover, reading just a single buffer into WARPLab incurs an average delay of around 3ms. Therefore, it is difficult to obtain accurate channel traces using the current WARP framework.

We overcome these limitations by designing our own hardware and software solutions. We modify the WARP FPGA design by halving the sampling clock and increasing the buffer size to $2^{16}$ samples (beyond $2^{16}$ is not possible due to FPGA Bus width limitations). This allows us to send/receive signals up to 6.4ms long—an 8× increase in the transmission airtime. Consequently, we are able to send/receive LTF symbols to accurately track the channel as shown in Fig. 10b. We have also written a multi-threaded C driver that directly interfaces with the WARP hardware [15]. This lightweight driver completely replaces the much slower WARPLAB driver that is written in MATLAB. On comparing the performance of the basic readIQ routine which reads the IQ baseband buffers into the PC, we witness an almost 3× to 10× reduction in read delay as the number of buffers is increased from 4 to 10. In particular, the average readIQ delay for reading $2^{11}$ samples in case of 6 STAs is only 0.8ms, i.e., smaller packets can be sent and received within 1ms (Fig. 10a), which allows us to do channel measurements with high accuracy. The long transmission design is used for validating the STA interference behavior, e.g., Section 4.3, while the WARP driver is used in the evaluation of our algorithms. We would like to point out that measurements with WARP boards are highly sensitive to noise, external interference and other physical impairments. Therefore, we apply a locally-weighted smoothing on the time variations of channel magnitude and phase, similar to the channel tracking procedure of OFDM receivers.

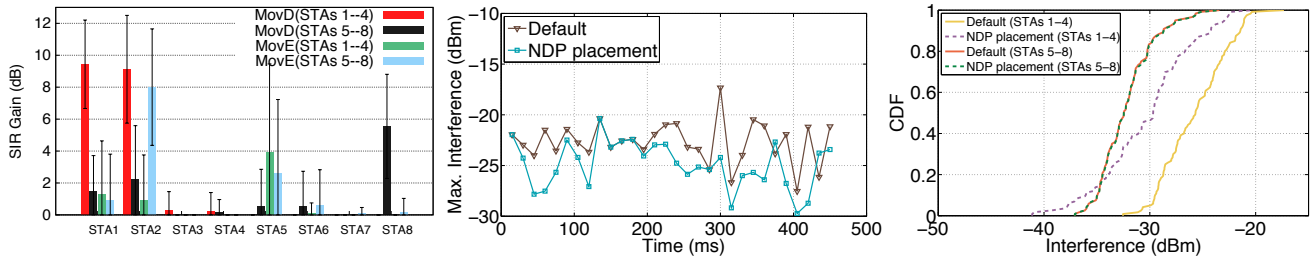
### 5.2.3 Mobility Experiments

In order to account for the channel conditions in real net-MIMO deployments, we consider three mobility scenarios in our evaluation: (1) *Stationary*(Stat)—This corresponds to the scenario when the location of STAs is fixed, and there are no disturbances from the motion of people or nearby objects. (2) *Moving environment*(MovE)—Environmental mobility is simulated through human actions such as arm movements and walking while the nodes remain fixed. (3) *Moving device*(MovD)—Device mobility is simulated by moving the cart on which STA nodes are placed while taking measurements. This motion closely resembles the pedestrian mobility of smartphones and other mobile devices.

## 5.3 NDP Placement

We implement and run the NDP placement given in Algorithm 1 over a period of 450ms where a netMIMO transmission occurs every 15ms. A number of scenarios for STAs' mobility are played out while the channel measurements are taken. Fig. 11a shows the SIR gains of STAs with the NDP placement compared to the default training protocol in four different scenarios: MovD where one among the group of STAs 1–4 and STAs 5–8 are placed and moved on a cart, and repeated again in MovE where instead external disturbances are present around one of the group of STAs. The gain is calculated by measuring the SIR of a single netMIMO packet that is transmitted following the feedback process. As one can see, the gains for some STAs can be significant (5–10dB) and is due to the smaller CSI delay resulting in reduced interference. Most importantly, the gains are strongly correlated to the channel mobility of the STAs. For example, when STAs 1–4 are moved, all of STAs 1–4 see gains while STAs 5–8 see almost none. When STAs 5–8 are moved, STAs 6 and 8 have noticeable gains while STA3's and STA4's performance remains unchanged. This can be explained from the two different CSI delays—a few STAs' delay is reduced (from $NDP^2$) while for others it remains the same ($NDP^1$). Note that due to the coarseness of the measurement readings, the actual gains may deviate slightly than what has been reported.

Results in Fig. 11b and Fig. 11c are shown for the MovD (STAs 1–4) scenario of the 8–STA netMIMO system which according to Table 2 incurs a feedback delay of 3.96ms with




**(a)** Observed gains in four experiments **(b)** Maximum interference among STAs **(c)** Comparison between STA clusters

**Figure 11: Evaluation of NDP placement in 8–STA netMIMO. Results show significant gains for mobile STAs.**

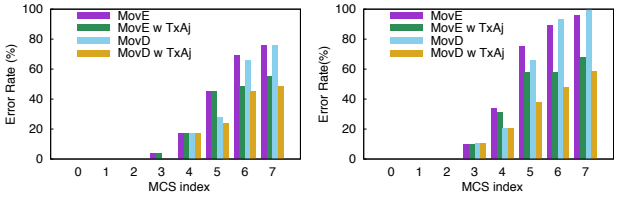
single NDP training. The average readIQ delay of reading each WARP buffer was observed to be 1.07ms. Therefore, channel readings are available at 4 time points within the feedback period. In Fig. 11b, we plot the maximum average interference of STAs ($\bar{I}_\star^t$ in Algorithm 1) as a function of netMIMO iteration. As expected, the NDP placement strives to lower the maximum interference compared to the default scheme. The distribution of the interference across the mobile and the stationary STA clusters is shown in Fig. 11c, where the selective impact of NDP placement on the interference distribution of mobile STAs is visible.

## 5.4 TXOP Adjustment

Similar to the NDP placement, the TXOP Adjustment (TxAj) procedure in Algorithm 2 uses the past interference to adapt every netMIMO transmission. The CSI delay of STAs is fixed while the netMIMO TXOP duration is varied. Since the objective here is to meet the SIR threshold ($\beta_k$) of a STA according to its MCS, we evaluate the performance of our proposed approach with all possible MCS values. We use the widely adopted MCS-threshold lookup table for 802.11 wireless environments [16]. Figs. 12a and 12b show the evaluation results of a single STA in an 8-STA netMIMO setup with TXOP limits of 2ms and 4ms, respectively. The minimum TXOP limit of each STA was set to 1ms in both scenarios. The TXOP limit value is the initial TXOP duration of all netMIMO STAs which remains constant in the default scheme but is adjusted for each STA in our proposed scheme. Note that the CSI delay here consists of the TXOP duration in addition to the feedback delay of 3.96ms. For each MCS, we calculate the percentage of STA's total transmissions that do not meet the corresponding average SIR threshold and hence would result in packet errors. Since the accuracy of measurement is limited to 1ms, the SIR is measured only for a few packets in the TXOP. A higher MCS means higher SIR requirements (e.g., MCS7 requires SIR $>$ 23dB), the plots in Fig. 12 therefore exhibit an increasing trend. With our TXOP Adjustment which changes the transmission length according to the interference, we see a noticeable reduction in error rate, up to 26% in case of 2ms TXOP limit, and 42% in case of 4ms TXOP limit.

## 5.5 Field Test

We conducted a field test of the netMIMO setup by running both NDP placement and TXOP Adjustment simultaneously for a period of 450ms. We consider the ambient wireless environment of our testbed and further orchestrate all STAs in motion together with the motion of people/objects around. We consider two different locations of




**(a)** TXOP limit=2ms **(b)** TXOP limit=4ms

**Figure 12: Error rate measured for each MCS before and after TXOP Adjustment**

| # STAs | $K=4$ | | $K=6$ | | $K=8$ | |
|---|---|---|---|---|---|---|
| Location | (1,2) | (3,4) | (1,2) | (3,4) | (1,2) | (3,4) |
| Default ( §2.1) | 134 | 113 | 199 | 108 | 279 | 150 |
| Proposed ( §4) | 142 | 119 | 220 | 123 | 292 | 174 |

**Table 3: Average netMIMO-throughput (Mbps)**

the STA clusters, with Location (3,4) being in NLOS environment w.r.t. the APs, and placed farther than Location (1,2). The initial TXOP limit was set to 4ms. The average throughput of each STA is calculated by averaging over all possible MCS data rates, which is then added for all STAs to arrive at the average netMIMO throughput, as shown in Table 3. It can be concluded that the throughput gains are modest in the case of 4–STA netMIMO, but become more prominent as the system is scaled—more than 15% in case of 8 STAs. Note that the throughput is per unit time within a netMIMO TXOP.

## 5.6 Discussion

*Theoretical basis*: While the monotonic behavior of interference is loosely observed in our measurements, the channel state across netMIMO iterations can be quite different. Consequently, the theoretical analysis in §4.3 does not necessarily apply, but rather provides a guiding principle.

*Channel fading*: Due to multi-path effects, few of the OFDM subcarriers may experience random fading. The measurement-based approach is robust to fading as the interference is averaged across all subcarriers.

*Rate adaptation and Decoding*: We have used fixed MCS rates for each STA for determining the transmission duration. In addition, decoding is assumed to fail completely if the decoding threshold is not met. However, our TXOP adjustment can be further improved if rate adaptation is applied within and across TXOPs, whereas soft decoding at STA will reduce the TXOP error rate.

# 6. RELATED WORK

Distributed MU-MIMO systems have been implemented through centralized [3, 17] and decentralized WLAN architectures [18]. Furthermore, a downlink MU-MIMO base station with 64 antennas was implemented in [2]. Each of them assumed perfect CSI, or rather stationary wireless channel, which is justified given that their experiments were carried out in a controlled environment. However, we find that user mobility has a significant effect on the performance of such systems. While the other alternative of implicit CSI feedback, as employed in [2], effectively mitigates the CSI aging effects, its calibration requirements make it less practical for WLAN devices, and as such has not been adopted in the 802.11ac standard.

In theory, the results on MU-MIMO capacity with the overhead of training and feedback, imperfect channel estimation, and channel aging have already been known and reported [1, 9, 19]. However, few systems have validated these results in practice. The user mobility profile was considered in [20] to select a transmission strategy. The authors claimed that netMIMO architecture was only suitable for stationary clients. In [12] and [8], experiments were performed to show the relationship between feedback delay and capacity of 802.11n SU-MIMO and MU-MIMO users, respectively. The authors in [8] observed that the performance of MU-MIMO is highly sensitive to CSI delay and aging, and the sensitivity is dependent on the number of Tx/Rx antennas and the interference cancellation ability of the STAs. It is stated that without interference cancellation, capacity decreases considerably within 5ms, which is also consistent with our findings. However, their results are limited to just 4 STAs, assumes multiple Rx antennas, and employs fixed training interval for each STA irrespective of its mobility state. The recent work in [21] on MU-MIMO feedback is closely related to ours. It proposes an adaptive feedback compression scheme based on frequency and time-domain compression techniques, and evaluates them in realistic indoor channels. The setup, however, considers single AP and 2 STAs, and does not take into account the scaling of feedback overhead or the diversity from having multiple APs.

# 7. CONCLUSION

As recent advances in hardware bring netMIMO closer to reality, it is necessary to closely examine the various aspects of such a system. In this paper, we considered the impact of CSI aging on the performance of a large-scale netMIMO in real and accurately measured indoor wireless channels. We proposed a two-phase feedback protocol and an adaptive transmission scheme, both of which are adapted from the interference measured by the STAs. Our approach exploits the underlying relationship between interference and CSI delay which was validated in our experiments; a more thorough study is needed to fully understand this dependency and design better protocols. Other aspects like MCS rate adaptation and receiver interference cancellation further need to be analyzed. All of these are matters of our future inquiry.

## Acknowledgments

The work reported in this paper was supported in part by the NSF under Grants 1160775 and 1317411.

# 8. REFERENCES

[1] D. Gesbert, S. Hanly, H. Huang, S. Shamai Shitz, O. Simeone, and W. Yu, "Multi-Cell MIMO Cooperative Networks: A New Look at Interference," *in IEEE Journal on Selected Areas in Communications*, 2010.

[2] C. Shepard, H. Yu, N. Anand, E. Li, T. Marzetta, R. Yang, and L. Zhong, "Argos: practical many-antenna base stations," in *Proc. of MOBICOM*, 2012.

[3] H. Rahul, S. Kumar, and D. Katabi, "JMB: scaling wireless capacity with user demands," in *Proc. of SIGCOMM*, 2012.

[4] G. Caire, S. A. Ramprashad, and H. Papadopoulos, "Rethinking network MIMO: Cost of CSIT, performance analysis, and architecture comparisons," in *Information Theory and Applications Workshop (ITA)*, 2010.

[5] " Rice University Wireless Open Access Research Platform,"

[6] Q. Spencer, A. Swindlehurst, and M. Haardt, "Zero-forcing methods for downlink spatial multiplexing in multiuser MIMO channels," *IEEE Transactions on Signal Processing*, vol. 52, no. 2, pp. 461–471, 2004.

[7] " Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications ," IEEE Std. 80211ac Draft 3.0, 2012.

[8] M. Gong, E. Perahia, R. Want, and S. Mao, "Training protocols for multi-user MIMO wireless LANs," in *Proc. of PIMRC*, 2010.

[9] N. Jindal, "MIMO Broadcast Channels With Finite-Rate Feedback," *IEEE Transactions on Information Theory*, vol. 52, no. 11, pp. 5045–5060, 2006.

[10] R. Porat, E. Ojard, N. Jindal, M. Fischer, and V. Erceg, "Improved MU-MIMO Performance for Future 802.11 Systems Using Differential Feedback," in *Information Theory and Applications Workshop (ITA)*, 2013.

[11] J. Roh and B. Rao, "Efficient feedback methods for MIMO channels based on parameterization," *IEEE Transactions on Wireless Communications*, vol. 6, no. 1, pp. 282–292, 2007.

[12] E. Perahia, A. Sheth, T. Kenney, R. Stacey, and D. Halperin, "Investigation into the Doppler Component of the IEEE 802.11n Channel Model," in *Proc. of GLOBECOM*, 2010.

[13] F. Boccardi, H. Huang, and M. Trivellato, "Multiuser eigenmode transmission for mimo broadcast channels with limited feedback," in *IEEE Workshop on Signal Processing Advances in Wireless Communications*, 2007.

[14] M. Kendall, *Rank Correlation Methods*. 1990.

[15] K. C. Garikipati and K. G. Shin, "Improving Transport Design for WARP SDR Deployments," in *Proc. of the ACM SIGCOMM Software Radio Implementation Forum (SRIF)*, 2014.

[16] Cisco Systems Inc., "Wireless Mesh Access Points, Design and Deployment Guide, Release 7.4," 2013.

[17] H. Balan, R. Rogalin, A. Michaloliakos, K. Psounis, and G. Caire, "Achieving high data rates in a distributed MIMO system," in *Proc. of MOBICOM*, 2012.

[18] X. Zhang, K. Sundaresan, M. Khojastepour, S. Rangarajan, and K. Shin, "NEMOx: Clustered Network MIMO for Wireless Networks," in *Proc. of MOBICOM*, 2013.

[19] G. Caire, N. Jindal, M. Kobayashi, and N. Ravindran, "Multiuser MIMO Achievable Rates With Downlink Training and Channel State Feedback," *IEEE Transactions on Information Theory*, vol. 56, no. 6, pp. 2845–2866, 2010.

[20] S. Singh, K. Sundaresan, A. Khojastepour, S. Rangarajan, and S. Krishnamurthy, "One strategy does not serve all: tailoring wireless transmission strategies to user profiles," in *Proc. of ACM Workshop on Hot Topics in Networks*, 2012.

[21] X. Xie, X. Zhang, and K. Sundaresan, "Adaptive Feedback Compression for MIMO Networks," in *Proc. of MOBICOM*, 2013.