Distributed Real-time Implementation of Interference Alignment with Analog Feedback

Seogoo Lee, *Student Member,* Andreas Gerstlauer, *Senior Member,* and Robert W. Heath Jr., *Fellow*

***Abstract*—Interference alignment (IA) is a precoding technique that aligns interfering signals at receivers. It is known that IA achieves the maximum degrees of freedom over an interference channel under ideal assumptions. The real-world performance of IA depends on a range of practical issues, such as imperfect synchronization, channel estimation, and feedback. Practical issues have been studied in simulations and prototypes, but fully-distributed operation of IA network nodes has not been considered. In this paper, we present the first investigation of real-time IA performance on a fully-distributed 2×2 multiple-input and multiple-output (MIMO) prototype system with three physically independent user pairs. Over-the-air algorithms for time and frequency synchronization, as well as analog feedback, are studied and implemented. Sum rates are illustrated as a function of complexity and accuracy of different alignment, synchronization, and feedback algorithms. Corresponding trade-offs are evaluated using an iterative IA method, the injection of residual frequency offset into synchronization, and analog versus quantization-based limited feedback approaches. We demonstrate that, while considering all possible error sources in estimation, synchronization, and feedback, the theoretical multiplexing gain of IA can be reached in practical systems with a constant sum rate loss that remains within 5 bits/Hz/s compared to an ideal simulation.**

***Index Terms*—Interference alignment, prototyping.**

I. Introduction

Interference management plays a crucial role in current wireless systems. Interference alignment (IA) is a precoding technique for efficient interference management. By exploiting multiple signal dimensions and properly aligning interfering streams at the receiver, IA allows for a linear increase in the multiplexing gain as the number of user pairs and dimensions grows large. The concept of IA was first proposed in [1] and [2] for a multiple-input multiple-output (MIMO) X-channel configuration. It was extended to interference channels in [3] and has become the subject of tremendous research [4], [5], [6], [7], [8], [9], [10], [11]. Multidimensionality of signals for IA can be created in time, frequency, or through spatially-separated antennas. A reliable approach for obtaining the dimensionality required for IA is through antenna subspaces available with the use of MIMO communication.

This material is based upon work supported by the Army Research Laboratory under contract W911NF-10-1-0420 and by the National Science Foundation under Grant No. NSF-CCF-1218338.

The authors are with the Wireless Networking and Communications Group, Department of Electrical and Computer Engineering, The University of Texas at Austin, Austin, TX 78712 USA (e-mail: sglee@utexas.edu; gerstl@ece.utexas.edu; rheath@utexas.edu).

Theoretical work on IA depends on perfect assumptions about alignment, synchronization, and feedback. There are analytical, simulation-based, and prototyping approaches to study the impact that practical assumptions have on performance [12], [13], [14], [15], [16]. None of the existing work, however, considers the true nature of interference channels, i.e. distributed operation of transmitters and receivers. In this paper, we investigate real-time IA performance on a fully-distributed network for three independent user pairs, each with a 2×2 MIMO antenna configuration. Using our prototype, we also explore the design space of trade-offs between achievable sum rates and accuracy of different alignment, synchronization, and feedback algorithms.

Several techniques exist to compute the necessary precoding matrices for IA: the closed form method in [3] and iterative methods in [8] and [9]. The closed form solution is simple, but only works for three users. Iterative methods can work with any number of users, but suffer from high computational complexity. In [8], assuming channel reciprocity, the iterations are computed both on the transmitter and receiver sides, and the resulting precoding data is used for both the feedforward and feedback channels. Two optimization problems are suggested for iterative methods: minimizing total interference leakage, or maximizing the signal to interference and noise ratio (SINR). The key in [8] is that the algorithms do not need explicit channel state information (CSI) for all users, and can thus be done in a fully-distributed manner. In [9], the iterations are performed at either the transmitter or receiver side. The base algorithm in [9] is the same as the algorithm that minimizes total leakage in [8], but there is no assumption about channel reciprocity. By contrast, the algorithm in [9] requires CSI feedback and is, therefore, harder to be realized in a truly distributed fashion.

Synchronization issues in IA networks are studied in [13]. It is known that the phase offset caused by time and frequency synchronization errors is not an insurmountable problem for IA. The synchronization error causes not only phase offset, however, but also inter-symbol interference (ISI) or inter-carrier interference (ICI). Both interferences can work as additional error sources, and the synchronization accuracy can be an issue. Over-the-air master-slave synchronization protocols are studied for distributed systems with multiple interfering transmitters and receivers for which no common reference clock or frequency source is available [17], [18], [19]. Within such protocols, one of the transmitters works as the master of the system, while other nodes synchronize to the master. In [17], a precise sample-level time synchronization protocol for wireless LAN systems is given. A frequency synchronization

method based on the same protocol is introduced in [18]. A protocol using orthogonal frequency division multiplexing (OFDM) training symbols via a dedicated synchronization channel is presented in [19]. The protocol in [17] and [18] can be more generally used, but has a higher overhead compared to the protocol in [19], which is, however, limited to frequency division duplexing (FDD) and OFDM systems.

There are two approaches to make the CSI available at the transmitters: to feed CSI back from receivers to transmitters or to exploit channel reciprocity. Relying on reciprocity can be problematic because it is only valid when the uplink and downlink use the same frequency. This is the reason why CSI feedback is more commonly used in commercial wireless systems. For CSI measurement and feedback, there are three major, practical concerns: error caused by additive noise, error by limited precision of the feedback, and error by delayed feedback. Additive noise affects feedforward channel estimation, feedback channel estimation, and feedback CSI decoding. Errors due to limited precision of feedback occur because of the quantization of the estimated CSI. Finally, delays in feedback are caused by a channel's time variation. The effects of CSI feedback limitations on IA are studied using both quanitzed digital [10] and analog feedback [11]. The analysis of analog feedback in [11] is further extended to a more comprehensive analysis in [12]. A main conclusion in [11] and [12] is that analog feedback is a good and practical option for implementing interference alignment.

There is limited work studying alignment, synchronization, and feedback issues in practical IA realizations. From an experimental perspective, IA prototypes are built and the performance of IA is measured [13], [14], [15], [16]. In [13], feasible setups for IA in wireless LAN systems are studied. In [14], the wireless channel is measured, and the sum rate is calculated to show the gain in sum rate that can be achieved under real wireless channel conditions. In [15], the CSI is measured at the receivers, precoding vectors are obtained offline with the previously measured CSI, and the sum rate is calculated from over-the-air precoded transmission. In our previous work [16], we implemented the approach from [15] in National Instruments (NI) LabVIEW and C++ to study computational complexity on embedded platforms. Prior work [13], [14], [15], [16], however, still assumes perfect synchronization and CSI feedback.

The main contribution of this paper is the first real-time implementation and measurement of a fully-distributed MIMO IA system. Our IA prototype has three user pairs with a 2×2 MIMO antenna configuration. Each node works with independent time and frequency references, and the CSI feedback is via wireless channels. Although the system does not work under a frequency selective channel, we adopt OFDM for future extension of the implementation. With this prototype, we study critical issues in distributed IA systems: over-the-air time and frequency synchronization as well as feedback mechanisms. For synchronization, we introduce a variation of existing protocols that ignores phase offsets under the assumption of a constant air propagation delay for any path between two nodes. For feedback, we implement and compare quantization-based limited feedback with analog feedback,

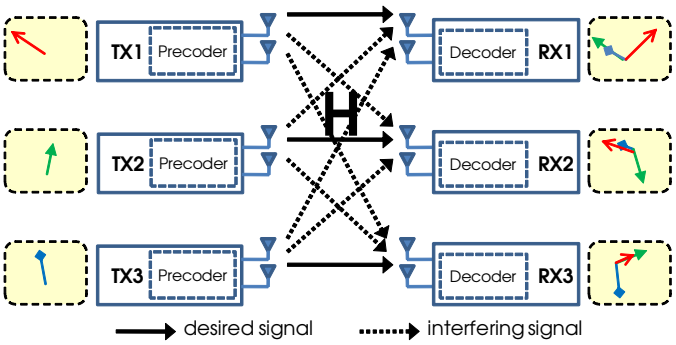


Fig. 1. Block diagram of three-user, 2×2 MIMO system.

which has been proposed for achieving high multiplexing gain with low overhead, but has not been validated in practice.

From our prototype, we observe that IA provides significant gains even in the presence of practical impairments. There is a sum rate loss of up to 5 bits/Hz/s compared to simulation results, but the multiplexing gain of IA is preserved, which further validates the analysis in [11]. Additional measurements are performed considering the accuracy of the IA solution, synchronization, and CSI feedback. In the process, we evaluate trade-offs between computational overhead and IA performance. To summarize the observations from our measurements: 1) the number of iterations needed for an iterative IA to reach achievable sum rates does not depend on the availability of perfect CSI. Furthermore, 2) the residual frequency offset that causes ICI in the OFDM system also introduces a 2 bits/Hz/s loss in the sum rate when assuming a 0.001 % offset with respect to the subcarrier spacing at 30 dB SNR. This result indicates that the synchronization accuracy can be important in IA system design. Finally, and most importantly, 3) a comparison of sum rates for quantization-based versus analog feedback demonstrates an advantage of analog feedback in multiplexing gain even when considering the errors from a realistic implementation of synchronization.

The remainder of this paper is organized as follows: Section II briefly describes our system model and the background of various IA, synchronization, and feedback algorithms employed. Section III presents our approach for realization of a distributed IA system. Section IV and Section V show implementation details and results, respectively. Finally, Section VI concludes our paper.

II. System Model and Background

A. Interference Channel

Our system model is based on a K-user interference channel. We define N_{t} as the number of antennas at the transmitter, N_{r} as the number of antennas at the receiver, and N_{s} as the number of desired data streams for each user's transmitter and receiver pair. It is assumed to be a pre-defined mapping of transmitters and receivers, as seen in Fig. 1 with $N_{\mathrm{s}} = 1$, $N_{\mathrm{t}} = 2$, and $N_{\mathrm{r}} = 2$. Only the data streams from the i-th transmitter are desired at the i-th receiver, and streams from other transmitters are considered to be interfering. All nodes are separated. To provide a simple explanation, we assume

a narrowband, block-fading channel, which is also valid for per-subcarrier operation of OFDM systems under frequency selective channels. The received signal $\mathbf{y}_k$ at receiver node k is given by

$$\mathbf{y}_k = \mathbf{H}_{k,k}\mathbf{F}_k\mathbf{d}_k + \sum_{m\neq k}\mathbf{H}_{k,m}\mathbf{F}_m\mathbf{d}_m + \mathbf{n}_k, \quad (1)$$

where $\mathbf{H}_{k,m}$ is the $N_\mathrm{r}\times N_\mathrm{t}$ channel matrix between the m-th transmitter and the k-th receiver, $\mathbf{F}_k$ is the $N_\mathrm{t}\times N_\mathrm{s}$ precoding matrix, $\mathbf{d}_k$ is the $N_\mathrm{s}\times 1$ data stream for the k-th user pair, and $\mathbf{n}_k$ is the additive white Gaussian noise (AWGN) at receiver k.

B. Interference Alignment

Interference alignment is a technique that aligns multiple interfering streams in the same subspaces, such that receivers can cancel the interference using linear techniques. With availability of antenna subspaces, the degrees of freedom (DOF) for IA can be achieved with limited number of subspaces. This depends on the antenna configuration as follows:

$$\min(N_\mathrm{t}, N_\mathrm{r})\frac{R}{R+1}K, \quad (2)$$

where $R = \frac{\max(N_\mathrm{t},N_\mathrm{r})}{\min(N_\mathrm{t},N_\mathrm{r})}$ [6]. For example, a three-user, 2×2 MIMO system ($K = 3, N_\mathrm{r} = 2$, and $N_\mathrm{t} = 2$) has a DOF of 3. Each user has a limited number of subspaces (two in this example) and can exploit one of the subspaces without interference, while interfering streams are aligned and received in the other subspace.

For IA with MIMO and a zero-forcing combining matrix at the receivers, the received signal becomes

$$\hat{\mathbf{y}_k} = \mathbf{W}_k^\mathrm{H}\mathbf{H}_{k,k}\mathbf{F}_k\mathbf{d}_k + \sum_{m\neq k}\mathbf{W}_k^\mathrm{H}\mathbf{H}_{k,m}\mathbf{F}_m\mathbf{d}_m + \mathbf{W}_k^\mathrm{H}\mathbf{n}_k, \quad (3)$$

where $\mathbf{W}_k$ is the $N_\mathrm{r}\times N_\mathrm{s}$ combining matrix at receiver k. The first term in (3) is the desired signal after decoding, while the second term is the interference, which becomes zero. The last term shows the effect of AWGN. The effect of IA can be summarized as follows:

$$\mathbf{W}_k^\mathrm{H}\mathbf{H}_{k,m}\mathbf{F}_m = 0 \quad (4)$$

$$\mathrm{rank}(\mathbf{W}_k^\mathrm{H}\mathbf{H}_{k,k}\mathbf{F}_k) = N_\mathrm{s}. \quad (5)$$

The sum rate R_sum of the IA system is defined as the summation of each user pair's rates which are derived from the ratio of the desired signal's power over the sum of interference leakage power and noise power as follows:

$$R_\mathrm{sum} = \sum_{k=1}^{K}\log_2\Big|\mathbf{I}_{N_\mathrm{s}} + (\sigma_k^2\mathbf{W}_k^\mathrm{H}\mathbf{W}_k + \mathbf{R}_k)^{-1} \times (\mathbf{W}_k^\mathrm{H}\mathbf{H}_{k,k}\mathbf{F}_k\mathbf{F}_k^\mathrm{H}\mathbf{H}_{k,k}^\mathrm{H}\mathbf{W}_k)\Big|, \quad (6)$$

where

$$\mathbf{R}_k = \sum_{m\neq k}\mathbf{W}_k^\mathrm{H}\mathbf{H}_{k,m}\mathbf{F}_m\mathbf{F}_m^\mathrm{H}\mathbf{H}_{k,m}^\mathrm{H}\mathbf{W}_k.$$

Here, the noise covariance matrix $\mathbb{E}[\mathbf{n}_k\mathbf{n}_k^\mathrm{H}] = \sigma_k^2\mathbf{I}_{N_\mathrm{r}}$, and $\mathbf{I}_N$ is the $N\times N$ identity matrix. For simplicity, the transmit power of $\mathbf{d}_k$ is assumed to be 1. There are different approaches to find solutions for the linear IA precoding matrix $\mathbf{H}_{k,m}$. One approach that works in the case of three user pairs and a 2×2 MIMO system is the closed form solution from [3]. In this approach, two interfering streams at a receiver should be aligned in one subspace. For example, the interfering streams from transmitters two and three can be aligned at receiver one as follows:

$$\mathbf{H}_{1,2}\mathbf{F}_2 = \mathbf{H}_{1,3}\mathbf{F}_3. \quad (7)$$

Then $\mathbf{F}_3$ can be obtained as

$$\mathbf{F}_3 = \mathbf{H}_{1,3}^{-1}\mathbf{H}_{1,2}\mathbf{F}_2. \quad (8)$$

Similarly, at receiver three, interfering streams from transmitters one and two should be aligned, and $\mathbf{F}_1$ is

$$\mathbf{F}_1 = \mathbf{H}_{3,1}^{-1}\mathbf{H}_{3,2}\mathbf{F}_2. \quad (9)$$

At receiver two, interfering streams from transmitters one and three should be aligned. Together with (8) and (9), $\mathbf{F}_2$ is any eigenvector of $\mathbf{E}$, where $\mathbf{E}$ is

$$\mathbf{E} = \mathbf{H}_{3,2}^{-1}\mathbf{H}_{3,1}\mathbf{H}_{2,1}^{-1}\mathbf{H}_{2,3}\mathbf{H}_{1,3}^{-1}\mathbf{H}_{1,2}. \quad (10)$$

In contrast, the iterative methods find an IA solution through numerical approaches [8], [9]. Using an optimization problem that minimizes interference leakage, the formulation in [9] is as follows:

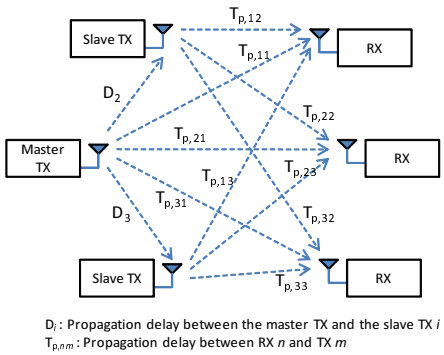
$$\max_{\substack{\mathbf{F}_m^\mathrm{H}\mathbf{F}_m=\mathbf{I}_{N_\mathrm{s}}\\ \mathbf{C}_k^\mathrm{H}\mathbf{C}_k=\mathbf{I}_{N_k-N_\mathrm{s}}}}\sum_{k=1}^{K}\sum_{\substack{m=1\\ m\neq k}}\|\mathbf{H}_{k,m}\mathbf{F}_m - \mathbf{C}_k\mathbf{C}_k^\mathrm{H}\mathbf{H}_{k,m}\mathbf{F}_m\|_\mathbf{F}^2. \quad (11)$$

The following procedure is performed to solve the problem in (11):

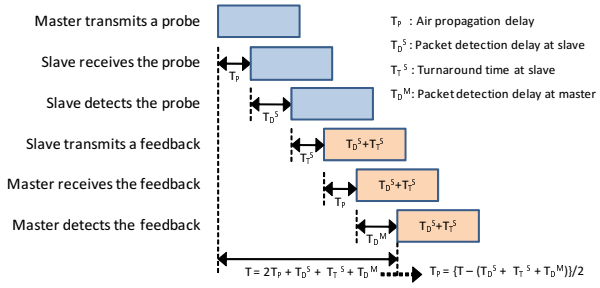
1) Choose the set of precoding matrices $\mathbf{F}_m$ randomly, $\forall m$.
2) Choose the columns of $\mathbf{C}_k$ to be the $N_k - N_\mathrm{s}$ dominant eigenvectors of $\sum_{m\neq k}\mathbf{H}_{k,m}\mathbf{F}_m\mathbf{F}_m^\mathrm{H}\mathbf{H}_{k,m}^\mathrm{H}$ $\forall k$.
3) Choose the columns of $\mathbf{F}_m$ to be the N_s least dominant eigenvectors of $\sum_{m\neq k}\mathbf{H}_{k,m}^\mathrm{H}(\mathbf{I}_{N_k} - \mathbf{C}_k\mathbf{C}_k^\mathrm{H})\mathbf{H}_{k,m}$ $\forall m$.
4) Repeat steps 2 and 3 until convergence.

Both the precoding matrix set $\{\mathbf{F}_m\}$ and the combining matrix set $\{\mathbf{W}_k\} = \mathbf{I}_{N_\mathrm{r}} - \mathbf{C}_k\mathbf{C}_k^\mathrm{H}$ are obtained from the above procedure. In this instance, N_k means the number of antennas for k-th receiver, where each receiver can have a different N_k. In this paper, however, we specialize the results to the case where all receivers have the same number of antennas. The convergence is guaranteed, but the cost function (11) is not convex, i.e. the algorithm can fall into local minima. The iterative method is more flexible than the closed form solution and applies to any number of users, given IA feasibility conditions are satisfied [5].

In this paper, we implement both closed form and iterative IA. Since our system is a three-user, 2×2 MIMO setup, we can apply the closed form solution, which has lower complexity. We use the iterative IA method from [9] to show trade-offs between computational complexity and sum rate performance. Among the numerical approaches [8], [9], the approach in [9] is simple and can be used more generally than others that assume channel reciprocity.



(a) System with different propagation delays.



(b) Delay estimation between master and a slave.

Fig. 2. Master-slave protocol in [17].

C. Synchronization

A distributed IA system requires synchronization among the transmitters and between the transmitter and receiver sides. Once the former is guaranteed, the latter becomes a point-to-point synchronization problem that is well-studied in literature. Thus, most of the effort in our prototype is spent on synchronization among the transmitters. We propose to use a master-slave protocol to solve this synchronization problem.

Master-slave synchronization protocols are efficient ways for synchronizing multiple nodes [17], [18]. Fig. 2(a) shows an example of a system that has different propagation delays between any two nodes [17]. One of the transmitters works as the master transmitter, and the others are slaves. The goal is to align the transmitters in time and frequency such that signals from all the transmitters arrive as close together as possible at the receivers.

To achieve this goal, each transmitter should know the over-the-air propagation delays between itself and the other nodes. This happens by broadcasting a probe and receiving responses that have the internal processing delays of the responding node. Fig. 2(b) shows the delay estimation procedure between the master and a node. When a transmitter first sends the probe, all of the other nodes sense the probe and measure its probe detection delay ($T_{\mathrm{D}}^{\mathrm{S}}$) as well as the turnaround delay when switching from receiving to transmitting mode ($T_{\mathrm{T}}^{\mathrm{S}}$). Each node then sends the sum of those delays ($T_{\mathrm{D}}^{\mathrm{S}} + T_{\mathrm{T}}^{\mathrm{S}}$) back to the transmitter that sent the probe. The transmitter measures the over-the-air propagation delays (T_{P}), including its internal detection time ($T_{\mathrm{D}}^{\mathrm{M}}$) and the received delay information from the slave. Then the other transmitters perform the same task to get the propagation delays from them to other nodes.

Once all required delay information is obtained at all nodes, the system is ready to work in a synchronous manner. The master transmitter sends its training to the other transmitters, and each of them adjusts its local time by the previously-measured delay information. After some time margin, all transmitters send their packets so all arrive at a receiver simultaneously. If, however, multiple receivers exist in the system, it is impossible to align all receivers and transmitters. Instead, the transmitters adjust their packet transmission time so the sum timing error at all receivers is minimized, e.g. by ensuring that all signals are received within the cyclic prefix of the OFDM system.

For frequency synchronization, slave transmitters measure the frequency offset using the master's training in order to recover the offset before they send their own data packet, as originally shown in [18]. Along with the frequency offset, the phase offset between the master and slaves can also be considered. The phase offset caused by oscillator drift is inevitable, and it may affect the performance of wireless systems. To allow the slaves to measure the phase difference with the master, two identical trainings can be sent by the master within a fixed time interval. The slaves compare these two trainings to estimate the phase offset between the two. The slaves can, therefore, compensate the phase offset before their data transmission. Details are described in [18].

Our prototype synchronizes transmitters through a similar master-slave protocol. It is, however, modified under assumption that the propagation delays between any two nodes are the same across the system. Also, phase synchronization is not implemented since it does not affect IA's performance. We do not implement a master election procedure, and instead assume that the master is already pre-defined. The method in [20], which uses the accuracy of local clocks and energy availability of nodes as decision metrics, can be a candidate for an election procedure.

D. CSI Feedback

The CSI is a key requirement for IA. We implement two over-the-air feedback methods: digital and analog. Digital feedback, which we call quantization-based feedback, is robust against AWGN, but suffers from quantization error. Analog feedback, by contrast, does not have quantization error and can have better estimation performance than the quantization-based method, especially at high SNR where IA gain in sum rate is significant. Its feedback estimation performance, however, is a function of AWGN. Our first choice for over-the-air CSI feedback is an analog approach that has lower overhead than quantization-based methods. For comparison purposes, however, we also implement a quantization-based feedback method.

For quantization-based feedback, we consider the explicit beamforming method from the IEEE 802.11n wireless LAN standard [21]. Each receiver first normalizes the measured CSI values, and quantizes these normalized values. It then sends the quantized versions of the amplitude, which is used as the normalization factor, and the normalized CSI values to the transmitters. With this method, N_{gain} bits are used for the

amplitude of CSI, and N_{q} bits are used to quantize each of the real and imaginary values of the measured CSI.

The non-negative integer amplitude index M_i for receiver i's CSI feedback (in dB) is

$$M_i = \min\left[2^{N_{\mathrm{gain}}} - 1, \left\lfloor 20\log_{10}\left(\frac{G_{\mathrm{ref}}}{m_i}\right)\right\rfloor\right]. \quad (12)$$

Here, G_{ref} is the maximum gain reference, m_i is the maximum absolute value among all the real and imaginary values of $\mathbf{H}_{i,j}, j = 1, 2, 3$, and $\lfloor x \rfloor$ is the largest integer smaller than or equal to x. The M_i index is sent to the transmitters via the feedback channel.

The real and imaginary values of $\mathbf{H}_{i,j}$ are normalized with M_i and quantized to $\mathbf{H}^{\mathrm{q}}_{\mathrm{scaled}(i,j)}$, which consists of N_{q} bits in two's compliment fixed-point number representation as follows:

$$\mathrm{Re}(\mathbf{H}^{\mathrm{q}}_{\mathrm{scaled}(i,j)}) = \mathrm{round}\left(\frac{\mathrm{Re}(\mathbf{H}_{i,j})}{M_i^{\mathrm{lin}}}(2^{N_{\mathrm{q}}-1} - 1)\right) \quad (13)$$

$$\mathrm{Im}(\mathbf{H}^{\mathrm{q}}_{\mathrm{scaled}(i,j)}) = \mathrm{round}\left(\frac{\mathrm{Im}(\mathbf{H}_{i,j})}{M_i^{\mathrm{lin}}}(2^{N_{\mathrm{q}}-1} - 1)\right), \quad (14)$$

where M_i^{lin} is the normalization factor given as $G_{\mathrm{ref}}/10^{M_i/20}$. For each receiver, the total number of bits to be transmitted as CSI feedback is $N_{\mathrm{gain}} + 2 \times N_{\mathrm{q}} \times K \times N_{\mathrm{t}} \times N_{\mathrm{r}}$. These bits are mapped to quadrature phase shift keying (QPSK) modulation and OFDM subcarriers.

The transmitters receive the CSI feedback and decode M_i and $\mathbf{H}_{\mathrm{received}(i,j)}$ to find the CSI as

$$\mathrm{Re}(\mathbf{H}_{\mathrm{received}(i,j)}) = \frac{\mathrm{Re}(\mathbf{H}^{\mathrm{q}}_{i,j})}{10^{M_i/20}} \quad (15)$$

$$\mathrm{Im}(\mathbf{H}_{\mathrm{received}(i,j)}) = \frac{\mathrm{Im}(\mathbf{H}^{\mathrm{q}}_{i,j})}{10^{M_i/20}}, \quad (16)$$

where $N_{\mathrm{gain}} = 3$ and $N_{\mathrm{q}} = 4, 5, 6$ and 8 are used for our implementation. G_{ref} is chosen to minimize the mean squared error (MSE) by quantization and is a constant in our implementation. The quantization noise is not fixed regardless of SNR. Thus, it needs more resolution and more bits per coefficient at higher SNR, introducing more overhead into the system.

An alternative to this quantization-based method is to implement the compressed quantization of IEEE 802.11n, which reduces the required feedback by parameterizing only the singular values and right singular vectors of the channel. This approach results in about a 25 % reduction in feedback overhead for our three-user, 2×2 MIMO setup. Due to the flexibility offered by scalar quantization, however, we defer implementation of the compressed quantization mode to future work.

Analog feedback is an alternative approach for conveying CSI [11]. With analog feedback, quantization error is traded off for additional estimation error. Analog feedback gets its name from the fact that a CSI value is directly mapped to the I/Q domain instead of being quantized to a binary representation with a specified number of bits, which is then mapped to a pre-defined modulation method, such as phase shift keying. In [11], errors in the feedback channel estimation, feedback training, and feedback data are considered. It is shown that the multiplexing gain of IA is preserved with only a constant sum rate loss when the transmit power of the CSI feedback linearly relates to the feedforward transmit power. The upper bound of the constant sum rate loss ΔR_{sum} is given as a function of the length of the training for feedback channel estimation (τ_{p}), the length of the CSI feedback (τ_{c}), and the ratio P_{f}/P between the feedback and feedforward transmit powers P_{f} and P:

$$\Delta R_{\mathrm{sum}}(\tau_{\mathrm{p}}, \tau_{\mathrm{c}}) \leq \sum_i d_i \log_2\left(1 + \frac{P}{P_{\mathrm{f}}} c(\tau_{\mathrm{p}}, \tau_{\mathrm{c}})\left(K - \frac{1}{d_i}\right)\right). \quad (17)$$

Here, $c(\tau_{\mathrm{p}}, \tau_{\mathrm{c}})$ is a value that decreases with increasing τ_{p} and τ_{c}, i.e. it indicates that the sum rate loss ΔR_{sum} decreases with an increase in overhead in the channel estimation and feedback data. To maintain the multiplexing gain, P_{f} needs to linearly scale with P, i.e. $P_{\mathrm{f}} = \alpha P$. Otherwise, if P_{f} scales as $P_{\mathrm{f}} = \alpha P^{\beta}$, there exists a loss in the multiplexing gain according to β. An extended study that also considers the feedforward channel estimation error is presented in [12].

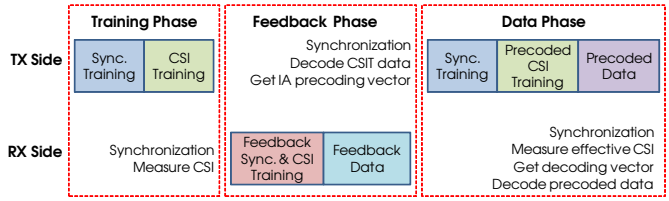


Fig. 3. Operation scenario for the prototype.

III. System Realization

We are targeting a three-user, 2×2 MIMO IA system as shown in Fig. 1. In our system, $N_{\mathrm{t}} = N_{\mathrm{r}} = 2$ for all user pairs, $N_{\mathrm{s}} = 1$, and $K = 3$. We use OFDM for both the feedforward and feedback channels. With this setup, two antenna subspaces are available for a user pair. Two interfering streams align in one subspace at each receiver, while the desired stream exists in the other subspace.

There are three main phases in IA operation: training, feedback, and data transmissions. Fig. 3 summarizes the information that must be transmitted in each of the phases, and also shows the tasks that should be performed by the transmitters and receivers.

- **Training Phase**: This phase is mainly for transmitter synchronization and CSI measurement at the receivers. The master transmitter first sends the synchronization training. The slave transmitters and receivers detect it, and perform associated time and frequency synchronization. Then all the transmitters send synchronized CSI training to the receivers, which perform CSI measurement and processing.
- **Feedback Phase**: This phase is for CSI feedback. Each of the receivers creates a packet that has the synchronization training, CSI training, and CSI feedback data. Then the receivers send the packet in a time orthogonal manner. After receiving and decoding these packets, the

transmitters perform feedback processing to synchronize with the receivers, calculate IA precoding vectors, and generate precoded packets to send in the following data phase.

- **Data Phase**: This phase is for the precoded training and data transmission. The system first performs the same transmitter synchronization as in the training phase. Then the transmitters send the precoded CSI training and precoded data in order. The receivers measure the effective CSI that is the product of precoding vectors and pure CSI, calculate the combining vectors from the precoded CSI training, and decode the precoded data with them. Finally, the receivers calculate the sum rate and the operation ends or repeats.

Details about synchronization protocols and packet structures in each phase will be described in the following sections. We report on the performance of our prototype, including average processing times and delays in Section V.

A. IA Algorithms

We implement both the closed form solution and iterative method from [9] in our prototype. For the closed form solution, we directly realize the formulation shown in the previous section. For the iterative method, the algorithm requires a singular value decomposition (SVD) of a 2×2 matrix. Every iteration requires this SVD. To reduce complexity and to be able to efficiently extend our implementation to a more practical, e.g. embedded form in the future, we use the typical two-step SVD implementation method from [22]. This method first decomposes the target matrix into a real bidiagonal matrix and then finds the SVD of this bidiagonal matrix.

In this paper, we only provide a brief summary of the algorithm. Further details may be found in [22]. The SVD of a matrix $\mathbf{B}$ can be written as follows:

$$\mathbf{B} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\mathrm{H}}, \tag{18}$$

where $\mathbf{U}$ and $\mathbf{V}$ are unitary matrices and $\mathbf{\Sigma}$ is a diagonal matrix with singular values ordered in magnitude. In the first stage, we transform the target matrix as follows:

$$\mathbf{B} = \mathbf{U}_1 \begin{bmatrix} f & g \\ 0 & h \end{bmatrix} \mathbf{V}_1 = \mathbf{U}_1 \mathbf{D} \mathbf{V}_1, \tag{19}$$

where the elements of $\mathbf{D}$, f, g, and h are real values. In the second stage, the SVD of the real bidiagonal matrix $\mathbf{D}$ is found as

$$\mathbf{D} = \mathbf{U}_2 \mathbf{\Sigma} \mathbf{V}_2^{\mathrm{T}}. \tag{20}$$

Combining (19) and (20), the final eigenvectors $\mathbf{U}$ and $\mathbf{V}$ become

$$\mathbf{U} = \mathbf{U}_1 \mathbf{U}_2 \tag{21}$$

$$\mathbf{V} = \mathbf{V}_1^{\mathrm{H}} \mathbf{V}_2. \tag{22}$$

For the combining matrices at the receivers, we use the well-known minimum mean squared error (MMSE) combining matrix presented in [9]. The combining matrix $\mathbf{W}_k$ is

$$\mathbf{W}_k = \left(\sum_{k=1}^{K} \mathbf{H}_{k,m} \mathbf{F}_m \mathbf{F}_m^{\mathrm{H}} \mathbf{H}_{k,m}^{\mathrm{H}} + \sigma_k^2 \mathbf{I} \right)^{-1} \mathbf{H}_{k,k} \mathbf{F}_k. \tag{23}$$

B. Over-the-air Synchronization

There are two differences between the master-slave protocol used in our prototype and the protocols in [17] and [18], which are described in Section II. First of all, since it does not affect IA performance, our prototype does not realize the phase synchronization in [18]. Also, the simplified version of time synchronization from [17] is employed. The over-the-air propagation delays between any two nodes are assumed to be the same. This assumption is valid in our setup because the bandwidth is not high enough and the nodes are not far apart enough to cause a symbol-level timing difference. We also adopt a multi-user OFDM system, which is robust against timing differences because of its cyclic prefix. In a real system, even though the nodes are far enough to cause differences, this is not a problem if they fall into the cyclic prefix as shown in [19]. Under this assumption, there is no need to measure air propagation delays. Furthermore, the processing delays and the turn-around delays at both the transmitters and receivers do not matter if the following two additional requirements satisfied: 1) the transmitting antennas can send a sample exactly at the desired time defined by the transmitters, and 2) the receivers know the exact time when a sample arrives at the receiving antennas. Our implementation hardware and software support these requirements. To validate synchronization performance, we first confirm the decoding of MIMO QPSK symbols in a 6×6 MIMO system before moving to the IA system.

As Fig. 4 illustrates, all the nodes in our prototype synchronize to the master. For our distributed system, we assume that all nodes are physically separated. This means that each node works with its own, independent time and frequency references, and others' references are unknown. In the training phase, which is the starting point of one iteration of the overall IA protocol, synchronization is performed in the following three steps:

1) At time zero, the master transmitter sends its training to the slave transmitters using its local timer. When a slave receives and detects the training, it measures the local time $T_{\mathrm{TP}}^{[n]}$ at which the first sample of training arrives at its antennas. Here, n is the slave index. The slaves also measure the frequency offset from this training.
2) All transmitters know T_{C}, which is the waiting time for synchronized transmission. The master waits until its local time reaches T_{C}, and the slaves wait until $T_{\mathrm{TP}}^{[n]} + T_{\mathrm{C}}$.
3) All transmitters immediately send their training to the receivers when their waiting time ends. Each of the slaves recover the measured frequency offset from their training before it is sent. The training and data are now synchronous in time and frequency.

Since the time between the training and data phases may not be short enough for the slaves to maintain their previous synchronization, the nodes perform the same operation in the data phase. Only the master knows the starting time of the data phase, T_{DP}, and it again sends its training to the slaves. The slaves now detect it at time $T_{\mathrm{TP}}^{[n]} + T_{\mathrm{DP}}^{[n]}$, and all transmitters send the synchronization training and data after T_{C} time. In addition to the synchronization among

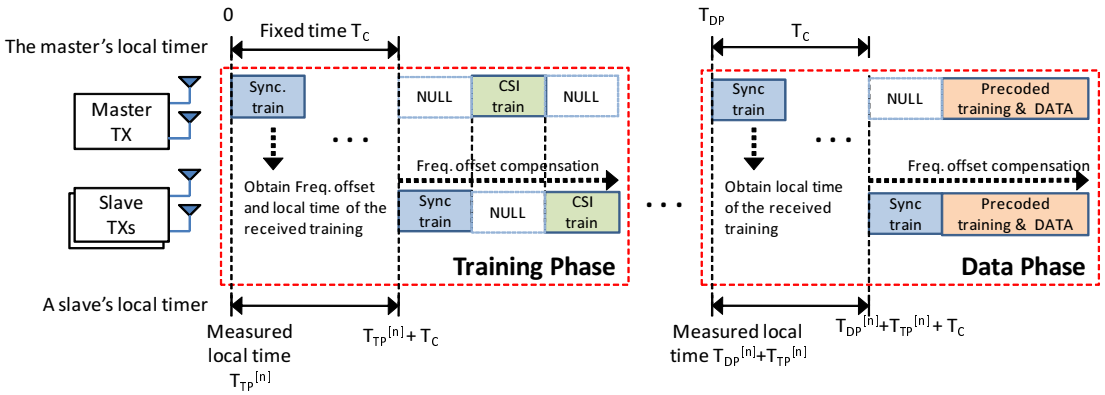


Fig. 4. Transmitter synchronization in our prototype.

transmitters, the receivers also synchronize to the transmitters. Each receiver does this independently, and there is no co-operative synchronization among the receivers.

C. CSI Feedback

The main feedback method in this paper is analog feedback. Under the assumption of a flat fading channel, receiver i measures an estimate

$$\hat{\mathbf{H}}_{i,k}, \quad k = 1, 2, 3. \tag{24}$$

Since each $\hat{\mathbf{H}}_{i,j}$ is a 2×2 matrix, there are $K \times N_{\mathrm{t}} \times N_{\mathrm{r}}$ complex values in the measured CSI matrix at each receiver. These values are mapped to N_{sc} subcarriers in an OFDM symbol by a $N_{\mathrm{sc}} \times (K \times N_{\mathrm{t}} \times N_{\mathrm{r}})$ mapping matrix. As a result, each receiver forms one OFDM symbol for analog feedback. The mapping matrix is a fixed, pseudo-random matrix whose elements are either -1 or 1. To make the average transmitting power of the OFDM symbol equal the other training and data symbols, we normalize the values after subcarrier mapping with a constant gain. The feedback link between a receiver and transmitter also uses the 2×2 MIMO setup, i.e. the OFDM symbol is transmitted via two antennas. Each receiver's feedback transmission is separated in time. After channel equalization and equal antenna gain combining, the transmitters that receive the OFDM symbols apply the pseudo inverse of the mapping matrix to the received symbols in order to find the CSI values. We provide details about our feedback structure in Section IV.

IV. Prototype Setup

We design our system to operate at 2.4 GHz with a bandwidth that would ideally be as large as 20 MHz. In reality, due to hardware constraints in our prototype, it is as low as 250 KHz, resulting in a sample duration of 4 us. Both the feedforward channel and feedback channel use the same frequency. We use OFDM for the training and data. By contrast, since our synchronization algorithm can only exploit time domain auto-correlation of the training, the synchronization training is a time domain signal. Table I summarizes the OFDM parameters of our prototype.

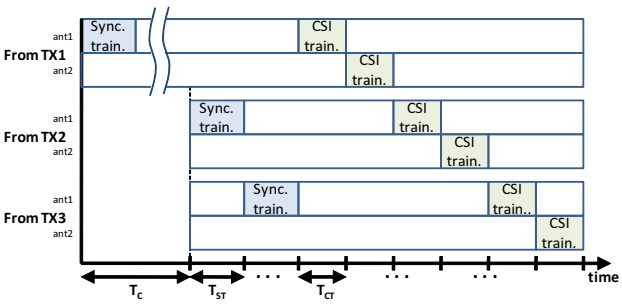
TABLE I
OFDM PARAMETERS FOR THE PROTOTYPE.

Fast Fourier transform (FFT) length	128
Cyclic prefix length	32
Number of null subcarriers	23
Number of data subcarriers (N_{sc})	105

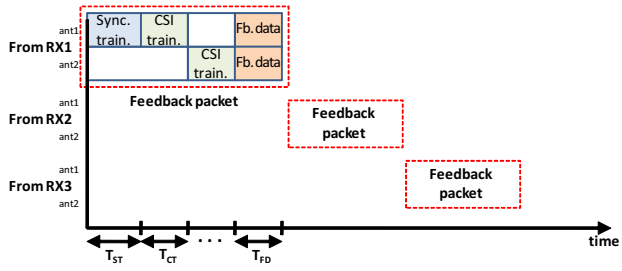
A. Packet Structure

Fig. 5 illustrates the packet structures for the training phase (Fig. 5(a)), feedback phase (Fig. 5(b)), and data phase (Fig. 5(c)).

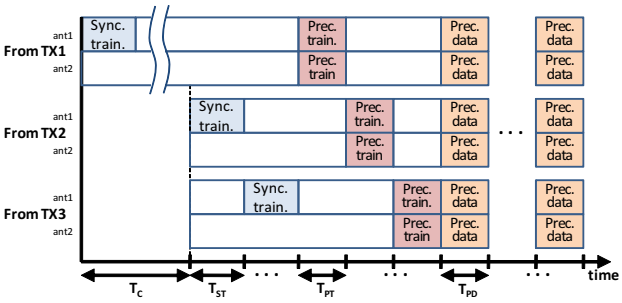
- **Training packet**: This packet starts with a synchronization training that has two parts: short training for coarse time synchronization and long training for fine time and frequency synchronization. A length 17 Zadoff-Chu sequence is repeated seven times as the short training, and a length 29 Zadoff-Chu sequence is repeated eight times as the long training. The total number of samples is 361, and it takes about $T_{\mathrm{ST}} = 1.4$ ms to send this training. The training has repeated patterns in the time domain, where synchronization based on self correlation is employed [23]. CSI training follows the synchronization training. CSI training is used by the receivers to measure channel state. Each CSI training is a single OFDM symbol, which is composed of 160 samples. The time to send one CSI training is $T_{\mathrm{CT}} = 0.64$ ms. This training is sent from each of the transmitting antennas in a time orthogonal manner as shown in Fig. 5(a). As such, six OFDM symbols are transmitted for our three user, 2×2 MIMO system. The training does not experience precoding or decoding. Thus, receivers can find the current wireless channel irrespective of precoding. With this training, each receiver measures a 2×6 CSI matrix.
- **Feedback packet**: Synchronization training and CSI training in the feedback packet are the same as those of the training packet. Training is followed by feedback data, which contains CSI information for precoding vector calculation in the transmitters. Depending on the feedback method, feedback data consists of one or more OFDM symbols ($T_{\mathrm{FD}} = 0.64$ ms each). As shown in



(a) Training packet structure.



(b) Feedback packet structure.



(c) Data packet structure.

Fig. 5. Packet structures.

Fig. 5(b), each receiver sends its feedback packet in a time orthogonal manner.

- **Data packet**: Synchronization training in this packet is also the same as in the previous packets. After the synchronization training, the transmitters send precoded training, which is used for the combining vector calculation in the receivers. This training experiences IA precoding at the transmitters, where each of the three users has one data stream mapped to two antennas via the precoding vectors. In addition, since this training is precoded in the same way as data, the receivers can use it to equalize the precoded OFDM data symbols that follow the precoded training. Each training consists of one OFDM symbol. The two antennas of a transmitter send the training at the same time ($T_{\mathrm{PT}} = T_{\mathrm{PD}} = 0.64$ ms), where each transmitter sends its packet in a time orthogonal manner (Fig. 5(c)).

To account for hardware delays and synchronization processing times in slave transmitters and receivers, we set $T_C = 250$ ms for both training and data packets in our prototype. There can be multiple precoded OFDM data symbols per data packet, and the maximum possible number of OFDM symbols in one packet depends on the coherence time of the channel

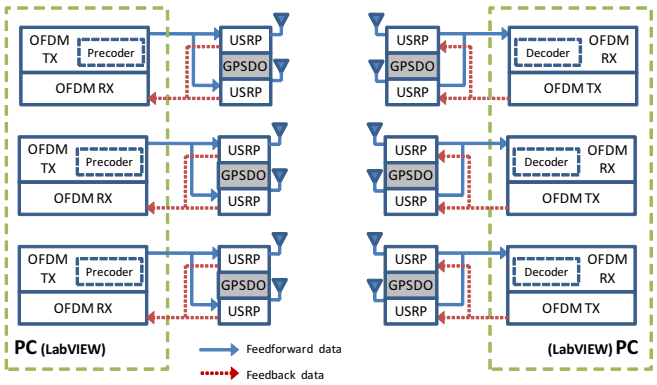


Fig. 6. Software and hardware configuration of the prototype.

environment. We only send one precoded OFDM data symbol to measure instantaneous sum rates.

B. Hardware and Software Setup

Our prototype uses NI's USRP-2921 [24] hardware and LabVIEW software, on two dual-core Intel Xeon 2.67 GHz workstations as shown in Fig. 6. We use one computer each for the transmitters and receivers. The USRPs are used as analog to digital conversion (ADC), digital to analog conversion (DAC), and radio frequency (RF) front-ends. Each USRP drives one antenna and each node has two USRPs attached to it. Accordingly, one computer controls six USRPs. A USRP can switch between transmitting and receiving mode. For example, the USRPs of the transmitters work in transmitting mode for the training and data phases, but in receiving mode for the feedback phase. Each computer is connected to its USRPs via a TCP/IP Ethernet link. Even though only one computer processes the data for all three transmitters and, similarly, only one computer realizes all three receivers, the signal processing for each node is completely independent. In other words, there are three parallel, independent processing chains in each workstation.

We use NI's LabVIEW, which is a model-based digital signal processing implementation environment [25] for our software defined radio (SDR) implementation. LabVIEW's USRP driver is used to control the hardware. To reduce processing time, we optimize our LabVIEW implementation by parallelizing the loop processing. With this optimization, we achieve a 4 times speedup in the signal processing chain. Besides the signal processing time, the system also spends time in the communication between USRPs and PCs. This is a hardware constraint of our prototype, and we do not consider this communication delay in the average processing times we report in Section V.

We use GPS disciplined oscillator (GPSDO) modules [26] to provide a required time (pulse per second (PPS)) and 10 MHz frequency reference to the USRPs. The GPSDO is originally designed for global synchronization to a GPS signal. In our setup, however, instead of providing a global GPS-based time and frequency reference to all the nodes, we only use each GPSDO to provide a better quality oscillator for one node, i.e. two USRPs. Hence, six GPSDOs exist in our prototype. Each

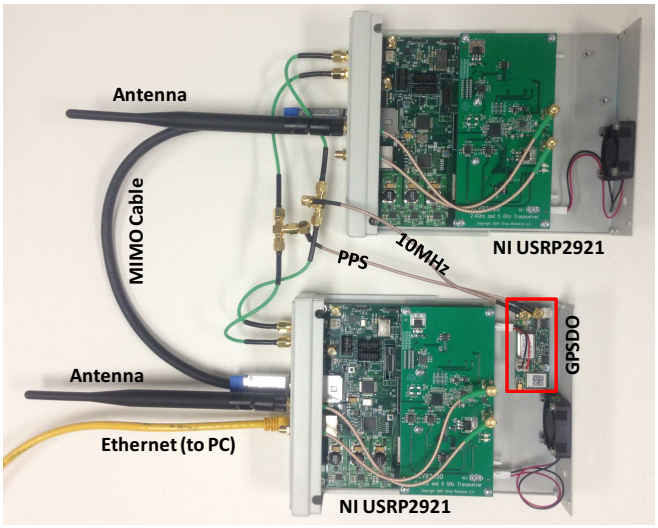


Fig. 7. Hardware configuration of one node.

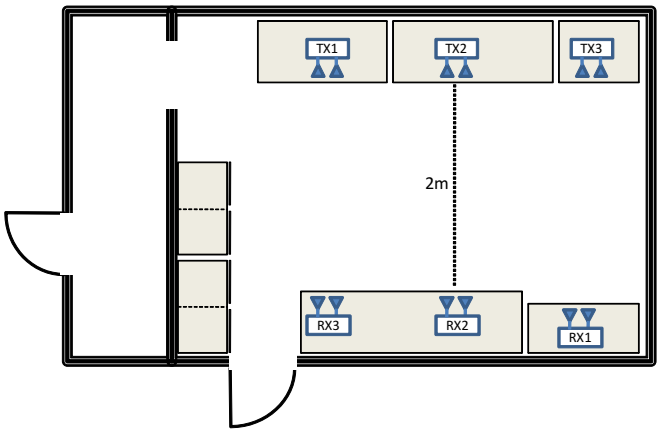


Fig. 8. Test environment of distributed nodes.

GPSDO generates independent time and frequency references. This use of GPSDOs is consistent with the fact that our system is a distributed network in which each node works with its own oscillator. It also does not rely on GPS signals that are unavailable in indoor environments, as is the case in our lab prototype.

Fig. 7 shows the hardware configuration of a node in our prototype. The MIMO cable between the USRPs in Fig. 7 is only used for the Ethernet connection of two USRPs to a computer. Fig. 8 shows the schematic of our indoor test environment. Both the transmitters and receivers are physically separated in order to verify distributed operation of our system. Due to power limitations, we do not test our system over a larger distance. Because OFDM is used with a cyclic prefix of duration 32 us, however, we anticipate that the prototype will work over longer distances. This claim is justified by the plethora of work on network MIMO [27], [28], where similar arguments are used to justify synchronization among multiple distant base stations.

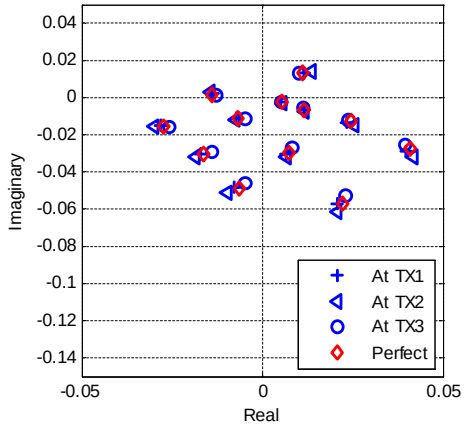


Fig. 9. An example constellation of analog feedback.

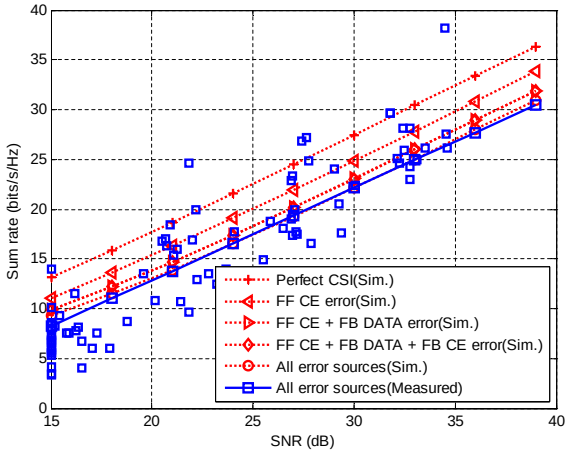
V. Results

In this section, we first present the sum rate of distributed IA with analog feedback, followed by additional measurement results on the relationship between system performance and the accuracy of the IA solution, the residual frequency offset, and CSI feedback. Except for the results in Fig 11, we use the closed form IA solution for our measurements.

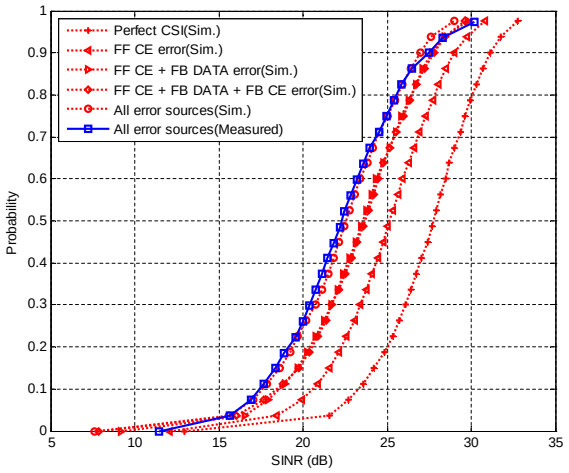
We scale the SNR of the system by changing the RF gain of the USRPs. To measure SNR, the receivers acquire the noise power at a time when there is no signal in the air. The signal power is obtained from CSI training after estimating the channel and low-pass filtering the result in the frequency domain. First, the frequency domain channel estimation result is transformed into the channel's impulse response using an inverse FFT. The first few samples in the impulse response show the actual channel power and the following samples show noise. By nulling the noise samples and performing another FFT, the low pass filtered channel estimation result is obtained, which we use as the signal power. The same transmit power is used for both the feedforward and feedback transmissions. As such, the SNR for both channels is the same.

Fig. 9 shows an example of the CSI received by a transmitter using analog feedback at an SNR of 30 dB. Fig. 9 has 12 constellation groups, and each group is an element of the 2×6 $\{\hat{\mathbf{H}}_{i,1}, \hat{\mathbf{H}}_{i,2}, \hat{\mathbf{H}}_{i,3}\}$ channel matrix from the i-th receiver. The four different points in each group are the received CSIs at transmitters 1, 2, 3, and the perfect reference CSI. A 0.45 % MSE within each group validates the basic operation of analog feedback in our prototype.

Fig. 10 shows the sum rate achieved in our system. Each dot in Fig. 10(a) is the measured instantaneous sum rate in the SNR range between 15 dB and 35 dB. The solid line shows their linear fit. We calculate the sum rate using the precoded training in the data phase. With the effective CSI measured from the training and σ_k^2 measured from the null time duration, the sum rate is obtained using equations (23) and (6). The measured sum rate is compared against results obtained from simulations with various error sources in the system. There are different simulation setups: 1) perfect CSI estimation (CE) and feedback, 2) only with feedforward CE (FF CE) error, 3)

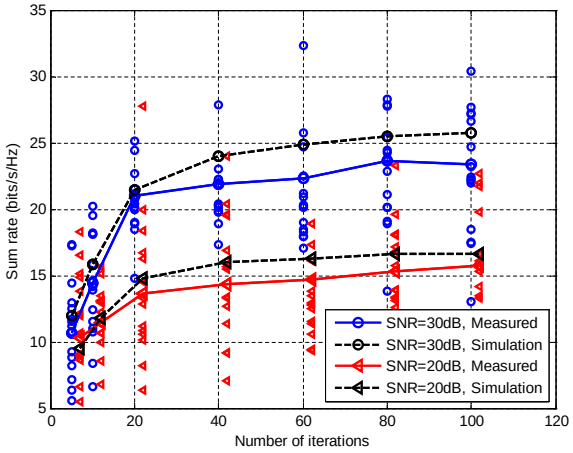


(a) Measured and simulated sum rate.

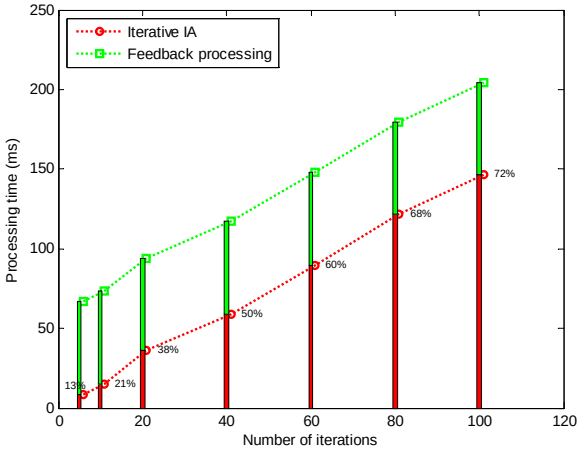


(b) Empirical CDF of measured SINR under 30 dB SNR.

Fig. 10. Distributed IA with analog feedback.



(a) Sum rate with iterative IA method.



(b) Processing time of iterative IA method.

Fig. 11. Complexity vs. performance trade-off of iterative IA method.

with feedforward CE and feedback data (FF CE + FB DATA) errors, 4) with the errors in 3) and feedback CE (FF CE + FB DATA + FB CE) error, and 5) with all error sources, i.e. errors in 4) and feedforward precoded channel estimation error. The most important observation from Fig. 10(a) is that the multiplexing gain remains unchanged, with analog feedback only introducing a constant sum rate degradation. This result verifies the analysis from [11].

We make some interesting observations from the empirical cumulative distribution functions (CDFs) of the same measured and simulated setups under 30 dB SNR (in Fig. 10(b)). As summarized in Table II, the means of the sum rates get smaller and the variances get larger when adding more error sources to the system. Furthermore, the measured mean is smaller and the variance is larger than those in simulation. We believe, however, that the difference of 0.1 is within the measurement error and, as such, is not a meaningful offset. We also believe that this is due to the fact that the actual channel is different from the Rayleigh channel model used in simulation. Note that the decreasing mean and the increasing variance cause a larger outage probability. For example, in the prototype, the outage probability for an SINR < 15 dB is 4.4 %, which is 44 times larger than the perfect CSI simulation and 1.5 times larger than the simulation with all error sources.

Fig. 11(a) shows the sum rate versus the number of iterations for the iterative IA method from [9]. The number of iterations is an important design parameter that determines the accuracy of the IA solution. The mean sum rates at SNR = 30 dB and SNR = 20 dB are measured and compared to those from a simulation with perfect CSI feedback. Since it is limited by CSI estimation and feedback, the sum rate in the prototype is smaller than in simulation. Importantly, however, the number of iterations to achieve the saturated sum rate does not decrease. This shows that the leakage from an imperfect IA solution is independent of other error sources, regardless of the SNR of the system. In addition, we can observe that with enough iterations (> 40) and at the same SNR, the iterative method will approach the performance of the closed form solution that is shown in Fig. 10.

We profile our LabVIEW implementation to measure computational overhead. The delay between the training and

TABLE II
MEANS AND VARIANCES OF SUM RATE UNDER DIFFERENT SIMULATED AND PHYSICAL SETUPS.

	Perfect CSI	FF CE	FF CE+FB DATA	FF CE+FB DATA+FB CE	All error sources	Measures
Mean	27.45	24.9	23.4	23.4	22.4	22.5
Variance	9.3	11.1	12.5	13.3	13.2	15.4

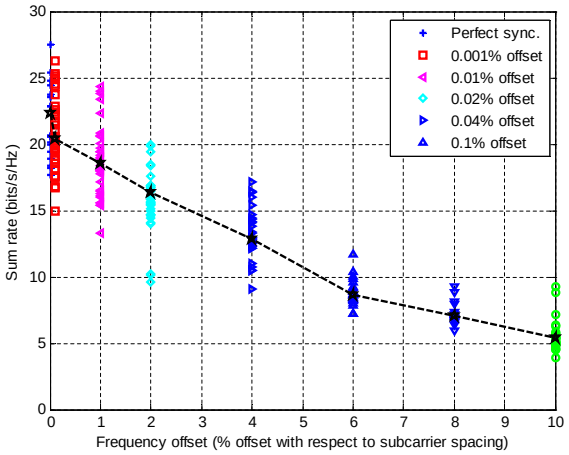


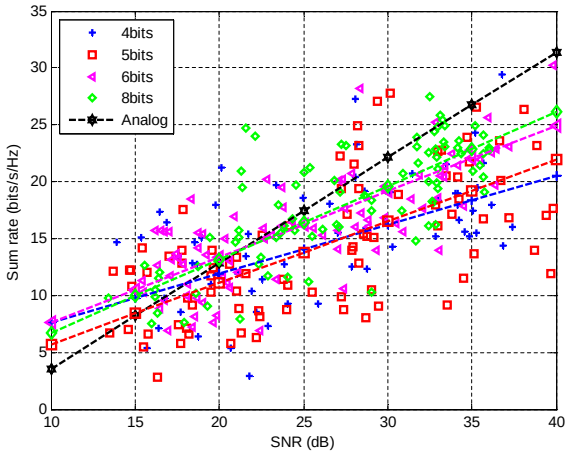
Fig. 12. IA performance with the residual frequency synchronization error.

feedback phases is determined by the processing times for CSI measurement and feedback packet generation in the receivers. The average training processing time is 15.2 ms. Likewise, the final data processing time in the receivers for synchronization, effective CSI measurement, and decoding at the end of the data phase is 28.3 ms.

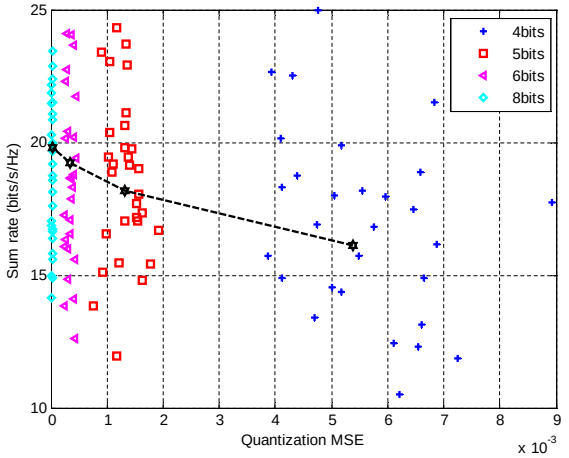
Considering the computational cost of IA in the transmitters, there is a trade-off between the number of iterations and sum rate performance. An increase in the number of iterations provides a more accurate IA solution, but requires more processing time, which results in overhead that can lower time-averaged throughput. Fig. 11(b) describes the portion of processing time that iterative IA takes in the overall feedback computation for each user pair. The feedback computation determines the delay between the feedback and data phases. It includes the processing time for a transmitter's time and frequency synchronization using the method in [23], FFT computation, feedback channel estimation using a least square algorithm, and feedback data decoding as well as precoding vector calculation using the iterative IA method. IA computation contributes about 13 % to overall processing time, even with a number of iterations as small as 5. With 100 iterations, IA computation takes up 72 % of the processing time. An increase in processing time causes an increase in overhead and a degradation in throughput. Although it is a strong advantage of iterative IA methods that they can be used for more than three users, the portion of IA processing time will be more dominant if the number of users increases. This analysis gives us a motivation that, overall, the IA method and the number of iterations need to be carefully chosen in real systems.

We also study IA performance in the presence of synchronization errors as shown in Fig. 12. In OFDM systems, the time synchronization error does not affect IA performance as long as the timing synchronization point is within the cyclic prefix. This requirement is not new to IA. Frequency synchronization, however, can not be perfect, and there always remains a small residual offset even after synchronization. This residual frequency offset causes ICI, which can not be mitigated by IA. As such, it acts as an additional error source in the system. Furthermore, the ICI caused by the residual frequency offset affects all steps in CSI measurement and feedback. Fig. 12 plots the sum rate for different frequency offsets with respect to the subcarrier spacing. The same frequency offset is added to all air links between the transmitter and receiver sides. Measurements are performed for a 30 dB SNR. Results show that with only 1 % residual frequency offset, the sum rate can be degraded by a factor of 3.5, loosing a significant IA gain at high SNR. This means that the training overhead, which determines the synchronization accuracy, is also an important design parameter in IA systems.

Finally, we also compare analog to quantization-based feedback. As described previously, for quantization-based feedback, we adopt the uncompressed explicit scalar quantization from the 802.11n standard [21]. Possible N_{q} values used in our evaluation are $4, 5, 6$ and 8. Each dot in Fig. 13 represents a measured instantaneous sum rate, and the lines are the linear fitting results. From Fig. 13(a), we can observe that the slope of the sum rate trend increases as feedback overhead increases, which implies that more bits are needed as SNR increases to maintain the multiplexing gain of IA. Fig. 13(b) plots the sum rate against the MSE of quantization-based feedback under 30 dB SNR. The dashed line connects the means of the sum rates with different N_{q}. We perform the experiment under 30 dB SNR. We can make some interesting observations given the results in Fig. 13: 1) Compared to quantization-based feedback, analog feedback has a higher sum rate performance in the high SNR region (Fig. 13(a)). This is because the error of analog feedback reduces with increasing feedback SNR, while the error of quantization-based feedback is fixed regardless of SNR. Furthermore, 2) the multiplexing gain of quantization-based feedback increases with increasing N_{q}, which confirms the previous analysis [10] on codebook-based limited feedback for IA (Fig. 13(a) and Fig. 13(b)). Assuming 2×6 CSI matrices and QPSK modulation for the transmission of quantized bits in our setup, the number of QPSK symbols to send for each user's CSI feedback is $12 \times N_{\mathrm{q}} + N_{\mathrm{gain}}/2$. For example, with $N_{\mathrm{q}} = 8$ and $N_{\mathrm{gain}} = 3$ quantization-based feedback requires 98 QPSK symbols. By contrast, using analog feedback, only 12 symbols are required. Such low overhead while maintaining high performance is a strong reason for using analog feedback.



(a) Measured sum rate with limited feedback and analog feedback.



(b) Sum rate vs. MSE of limited feedback.

Fig. 13. IA performance with the quantization-based feedback from 802.11n wireless LAN standard [21].

VI. Conclusions and Future Work

Our prototype for a three-user, 2×2 MIMO IA system is the first implementation that considers a fully-distributed network. Using the prototype, we study the performance of IA in the presence of practical issues with an emphasis on the feedback quality. The nodes are physically separated and work independently without co-operation, obeying the basic assumption of the interference channel. Distributed operation is achieved by developing and applying various over-the-air time and frequency synchronization protocols and feedback methods. Our results show that it is possible to implement IA in a distributed fashion while achieving predicted sum rate scaling. Furthermore, they show the efficacy of analog feedback in a real-world setting. Analog feedback is a viable approach for achieving good performance as an alternative to quantization-based approaches. Future work is needed to design a medium access control (MAC) protocol that allows a fast setup of distributed IA clusters.

Acknowledgments

The authors would like to thank Erik Luther and Ashish Chaudhari of National Instruments, Namyoon Lee and Jackson W. Massey of The University of Texas at Austin, and Omar El Ayach of Qualcomm for their help and support in making this work possible.

References

[1] M. Maddah-Ali, A. Motahari, and A. Khandani, "Signaling over MIMO multi-base systems: Combination of multi-access and broadcast schemes," in *2006 IEEE International Symposium on Information Theory*, pp. 2104–2108, 2006.

[2] S. Jafar and S. Shamai, "Degrees of freedom region of the MIMO X channel," *IEEE Transactions on Information Theory*, vol. 54, no. 1, pp. 151–170, 2008.

[3] V. Cadambe and S. Jafar, "Interference alignment and degrees of freedom of the K-user interference channel," *IEEE Transactions on Information Theory*, vol. 54, no. 8, pp. 3425–3441, 2008.

[4] S. Jafar, "Interference alignment: A new look at signal dimensions in a communication network," *Foundations and Trends in Communications and Information Theory*, vol. 7, no. 1, pp. 1–136, 2011.

[5] C. Yetis, T. Gou, S. Jafar, and A. Kayran, "Feasibility conditions for interference alignment," in *2009 IEEE Global Telecommunications Conference*, pp. 1–6, 2009.

[6] T. Gou and S. Jafar, "Degrees of freedom of the K-user M $\times$ N MIMO interference channel," *IEEE Transactions on Information Theory*, vol. 56, no. 12, pp. 6040–6057, 2010.

[7] N. Lee, D. Park, and Y.-D. Kim, "Degrees of freedom on the K-user MIMO interference channel with constant channel coefficients for downlink communications," in *2009 IEEE Global Telecommunications Conference*, pp. 1–6, 2009.

[8] K. Gomadam, V. Cadambe, and S. Jafar, "A distributed numerical approach to interference alignment and applications to wireless interference networks," *IEEE Transactions on Information Theory*, vol. 57, no. 6, pp. 3309–3322, 2011.

[9] S. Peters and R. Heath, "Interference alignment via alternating minimization," in *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, pp. 2445–2448, April 2009.

[10] J. Thukral and H. Bölcskei, "Interference alignment with limited feedback," in *2009 IEEE International Conference on Symposium on Information Theory*, pp. 1759–1763, 2009.

[11] O. Ayach and R. Heath, "Interference alignment with analog channel state feedback," *IEEE Transactions on Wireless Communications*, vol. 11, no. 2, pp. 626–636, 2012.

[12] O. Ayach, A. Lozano, and R. Heath, "On the overhead of interference alignment: Training, feedback, and cooperation," *IEEE Transactions on Wireless Communications*, vol. 11, no. 11, pp. 4192–4203, 2012.

[13] S. Gollakota, S. D. Perli, and D. Katabi, "Interference alignment and cancellation," in *2009 ACM SIGCOMM Conference*, pp. 159–170, 2009.

[14] O. Ayach, S. Peters, and R. Heath, "The feasibility of interference alignment over measured MIMO-OFDM channels," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 9, pp. 4309–4321, 2010.

[15] O. Gonzalez, D. Ramirez, I. Santamaria, J. Garcia-Naya, and L. Castedo, "Experimental validation of interference alignment techniques using a multiuser MIMO testbed," in *2011 International ITG Workshop on Smart Antennas (WSA)*, pp. 1–8, 2011.

[16] J. Massey, J. Starr, S. Lee, D. Lee, A. Gerstlauer, and R. Heath, "Implementation of a real-time wireless interference alignment network," in *2012 Asilomar Conference on Signals, Systems and Computers*, pp. 104–108, 2012.

[17] H. Rahul, H. Hassanieh, and D. Katabi, "SourceSync: A distributed wireless architecture for exploiting sender diversity," in *2010 ACM SIGCOMM Conference*, pp. 171–182, 2010.

[18] H. S. Rahul, S. Kumar, and D. Katabi, "JMB: Scaling wireless capacity with user demands," in *2012 ACM SIGCOMM Conference*, vol. 42, pp. 235–246, Aug. 2012.

[19] H. V. Balan, R. Rogalin, A. Michaloliakos, K. Psounis, and G. Caire, "AirSync: Enabling distributed multiuser mimo with full spatial multiplexing," *IEEE/ACM Transactions on Networking*, vol. PP, no. 99, pp. 1–1, 2013.

[20] W. Su and I. Akyildiz, "Time-diffusion synchronization protocol for wireless sensor networks," *IEEE/ACM Transactions on Networking*, vol. 13, pp. 384–397, April 2005.

[21] IEEE WG802.11 Wireless LAN Working Group, "IEEE standard for information technology - Local and metropolitan area networks-Specific requirements - Part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications amendment 5: Enhancements for higher throughput," *IEEE 802.11n/D2.0 Standard*, 2009.
[22] G. Golub and W. Kahan, "Calculating the singular values and pseudo-inverse of a matrix," *Journal of the Society for Industrial and Applied Mathematics: Series B, Numerical Analysis*, vol. 2, no. 2, pp. pp. 205–224, 1965.
[23] T. Schmidl and D. Cox, "Robust frequency and timing synchronization for OFDM," *IEEE Transactions on Communications*, vol. 45, no. 12, pp. 1613–1621, 1997.
[24] National Instruments, "NI USRP 2921 data sheet." http://sine.ni.com/ds/app/doc/p/id/ds-355/lang/en, 2012.
[25] National Instruments, "NI LabVIEW - Improving the productivity of engineers and scientists." http://www.ni.com/labview/, 2012.
[26] National Instruments, "GPS disciplined oscillator (GPSDO) kit data sheet." https://www.ettus.com/content/files/gpsdo-kit_4.pdf, 2012.
[27] V. Jungnickel, T. Wirth, M. Schellmann, T. Haustein, and W. Zirwas, "Synchronization of cooperative base stations," in *Wireless Communication Systems. 2008. ISWCS '08. IEEE International Symposium on*, pp. 329–334, Oct 2008.
[28] D. Gesbert, S. Hanly, H. Huang, S. Shamai Shitz, O. Simeone, and W. Yu, "Multi-cell MIMO cooperative networks: A new look at interference," *Selected Areas in Communications, IEEE Journal on*, vol. 28, pp. 1380–1408, December 2010.

Seogoo Lee Seogoo Lee (S'12) is a PhD student in Electrical and Computer Engineering department at The University of Texas at Austin. He received his B.S. and M.S. degrees in Electrical Engineering from Yonsei University, Seoul, South Korea in 2002 and 2004 respectively. Before joining UT Austin in 2011, he was a senior engineer in Digital Media Communication (DMC) Research and Development (R&D) Center at Samsung Electronics, Suwon, South Korea, where he designed Mobile WiMax (IEEE 802.16e and 802.16m) and 3GPP LTE baseband modem Intellectual Properties (IPs). His current research interests are low power implementation of wireless communication systems, approximate computing, and system-level design automation.

Andreas Gerstlauer Andreas Gerstlauer is an Associate Professor in Electrical and Computer Engineering at The University of Texas at Austin. He received his Dipl.-Ing. degree in Electrical Engineering from the University of Stuttgart, Germany, in 1997, and M.S. and Ph.D. degrees in information and computer science from the University of California, Irvine (UCI), in 1998 and 2004, respectively. Prior to joining UT Austin in 2008, he was an Assistant Researcher in the Center for Embedded Computer Systems (CECS) at UCI, leading a research group to develop electronic system-level (ESL) design tools. Commercial derivatives of such tools have been in use at the Japanese Aerospace Exploration Agency (JAXA) and NEC Toshiba Space Systems among others.

Dr. Gerstlauer is co-author on 3 books and more than 70 conference and journal publications, and his paper on OS modeling was reprinted as one of the most influential contributions in 10 years at DATE. He has presented in numerous conference and industrial tutorials, and serves as an Associate Editor for the ACM Transactions on Embedded Computer Systems (TECS) as well as Topic/Track Chair or member of the program committee for major international conferences such as DAC, DATE, CODES+ISSS, ICCAD and ICCD. Dr. Gerstlauer's research interests include system-level design automation, system modeling, design languages and methodologies, and embedded hardware and software synthesis.

Robert W. Heath Jr. Robert W. Heath Jr. (S'96 - M'01 - SM'06 - F'11) received the B.S. and M.S. degrees from the University of Virginia, Charlottesville, VA, in 1996 and 1997 respectively, and the Ph.D. from Stanford University, Stanford, CA, in 2002, all in electrical engineering. From 1998 to 2001, he was a Senior Member of the Technical Staff then a Senior Consultant at Iospan Wireless Inc, San Jose, CA where he worked on the design and implementation of the physical and link layers of the first commercial MIMO-OFDM communication system. Since January 2002, he has been with the Department of Electrical and Computer Engineering at The University of Texas at Austin where he is a Cullen Trust for Higher Education Endowed Professor, and is Director of the Wireless Networking and Communications Group. He is also President and CEO of MIMO Wireless Inc. and Chief Innovation Officer at Kuma Signals LLC. His research interests include several aspects of wireless communication and signal processing: limited feedback techniques, multihop networking, multiuser and multicell MIMO, interference alignment, adaptive video transmission, manifold signal processing, and millimeter wave communication techniques.

Dr. Heath has been an Editor for the IEEE Transactions on Communication, an Associate Editor for the IEEE Transactions on Vehicular Technology, and lead guest editor for an IEEE Journal on Selected Areas in Communications special issue on limited feedback communication, and lead guest editor for an IEEE Journal on Selected Topics in Signal Processing special issue on Heterogenous Networks. He currently serves on the steering committee for the IEEE Transactions on Wireless Communications. He was a member of the Signal Processing for Communications Technical Committee in the IEEE Signal Processing Society and is a former Chair of the IEEE COMSOC Communications Technical Theory Committee. He was a technical co-chair for the 2007 Fall Vehicular Technology Conference, general chair of the 2008 Communication Theory Workshop, general co-chair, technical co-chair and co-organizer of the 2009 IEEE Signal Processing for Wireless Communications Workshop, local co-organizer for the 2009 IEEE CAMSAP Conference, technical co-chair for the 2010 IEEE International Symposium on Information Theory, the technical chair for the 2011 Asilomar Conference on Signals, Systems, and Computers, general chair for the 2013 Asilomar Conference on Signals, Systems, and Computers, founding general co-chair for the 2013 IEEE GlobalSIP conference, and is technical co-chair for the 2014 IEEE GLOBECOM conference.

Dr. Heath was a co-author of best student paper awards at IEEE VTC 2006 Spring, WPMC 2006, IEEE GLOBECOM 2006, IEEE VTC 2007 Spring, and IEEE RWS 2009, as well as co-recipient of the Grand Prize in the 2008 WinTech WinCool Demo Contest. He was co-recipient of the 2010 and 2013 EURASIP Journal on Wireless Communications and Networking best paper awards, the 2012 Signal Processing Magazine best paper award, a 2013 Signal Processing Society best paper award, the 2014 EURASIP Journal on Advances in Signal Processing best paper award, and the 2014 Journal of Communications and Networks best paper award. He was a 2003 Frontiers in Education New Faculty Fellow. He is also a licensed Amateur Radio Operator and is a registered Professional Engineer in Texas.