Enabling Real-Time Interference Alignment: Promises and Challenges

Kyle Miller, Atresh Sanne
Department of ECE
University of Texas, Austin
Austin, TX - 78712
kmil16@gmail.com
atresh.sanne@utexas.edu

Kannan Srinivasan
Department of CSE
The Ohio State University
Columbus, OH - 43210
kannan@cse.ohio-state.edu

Sriram Vishwanath
Department of ECE
University of Texas, Austin
Austin, TX - 78712
sriram@ece.utexas.edu

ABSTRACT

As its name suggests, "interference alignment" is a class of transmission schemes that aligns multiple sources of interference to minimize its impact, thus aiming to maximize rate in an interference network. To our knowledge, this paper presents the first real-time implementation of interference alignment. Other implementation in the literature are either done offline or assume a backchannel between participating nodes to perform alignment. On the other hand, this paper presents a blind interference alignment scheme, one that does not require channel state information at the transmitters or the knowledge of other transmitters data or the knowledge of data between receivers and functions in real-time.

Categories and Subject Descriptors

C.2 [**Computer-Communication Networks**]: Network Protocols

General Terms

Interference Alignment

Keywords

Interference Management, Wireless Networks

1. INTRODUCTION

Interference alignment (IA), a fresh and exciting transmission strategy recently introduced as a mechanism for improving the performance of interference networks, is known to have tremendous gains over traditional orthogonal coding schemes in literature [1]. Specifically, alignment is shown to achieve the optimal degrees of freedom in time varying $K > 3$ user interference channels in [2], and X channels in [3]. Here, degrees of freedom denotes the ratio of the sum-rate achieved by the channel and the logarithm of the signal to noise ratio ($\log SNR$) as $SNR \to \infty$. Mathematically, this can be summarized as:

$$DoF = \lim_{SNR \to \infty} \frac{C(SNR)}{\log SNR}$$

where $C(SNR)$ is the sum-rate achieved using any transmission scheme. Using alignment, it is shown in [2] that a DoF of $K/2$ can be achieved for the K user time-varying interference channel. Using time-division, frequency division or other orthogonal schemes, the DoF is limited to 1. Similarly, alignment can improve the DoF of X channels from 1 to 4/3 [3]. The degrees of freedom achieved by alignment in these two cases can be shown to be optimal, i.e., that no other scheme exists that can achieve a higher DoF. Overall, in general, whenever there are multiple sources of interference, IA can potentially increase the achievable sum-rate in the network over traditional orthogonal transmissions schemes.

Ever since its discovery less than four years ago, there has been a rapid growth in literature associated with alignment, including the study of interference alignment for non-time varying (constant) channel gains [4], ergodic interference alignment [5], and most recently, lattice alignment schemes have been developed for finite SNR that achieve close to the performance of asymptotic alignment schemes presented in [6]. Typically, these alignment schemes are derived under many assumptions: including perfect and instantaneous knowledge of channel state and perfect synchronism between all parties. Fortunately, there are no additional requirements such as data cooperation between the nodes (as required by cooperative MIMO). However, the assumption of perfect and instantaneous channel state information (CSI) is very stringent and is very difficult to accomplish in practice.

In view of this, recent papers have focused on alignment under relatively relaxed assumptions on CSI. One body of work focuses on alignment in the presence of limited channel state feedback [7, 8]. The second body of work studies alignment based on delayed CSI [9]. Finally, the body of work most relevant to this paper accomplishes alignment *without* any CSI [10]. This blind interference alignment (blind IA) scheme is described in greater detail in Section 2.1 of the paper.

Given the excitement created by interference alignment, there has been some work at bringing it closer to practice. In [11], the authors implement a cancellation mechanism designed for a multiple input multiple output (MIMO) local area network (LAN), thus increasing the throughput of such a system. The scheme presented in [11] is novel, however,

We gratefully acknowledge the support of ONR under Award # N000141010337.

MobiHoc'12, June 11–14, 2012, Hilton Head Island, SC, USA.

it requires considerable cooperation over an Ethernet backbone that goes beyond the needs of an alignment algorithm as studied in literature. Furthermore, their work does not address how to carry out channel estimation needed for interference alignment. In fact, their interference requires receiver channel knowledge at the transmitters. Furthermore, they do not study the effects of transmitter synchronization mismatch. Channel estimation and synchronization are critical for IA performance. In this paper, we look at these aspects.

1.1 Related Work

An implementation of alignment with limited feedback was studied in [12]. The authors implement an iterative alignment scheme to determine alignment vectors given limited channel state information. The results in [12] present improvements in performance in terms of throughput over orthogonal schemes. However, there are two issues with the work in [12]. First, the implementation effort is done offline; second, the iterative optimization scheme presented is an approximation and may not always match the globally optimal alignment scheme.

1.2 Our Contributions

Our paper is distinct from this existing body of work in that, to the best of our knowledge, it represents the *first real-time* implementation of an interference alignment scheme. Its only requirement is symbol-level time synchronism between transmitters and receivers. It *does not* require any back-end cooperation in either data or channel state, nor does it make any idealized assumptions on the wireless medium, estimation, feedback and other associated algorithms.

The rest of this paper is organized as follows: the next section motivates the need for IA and presents the blind IA scheme implemented in this paper. Section 3 presents the main challenges in accomplishing interference alignment in practice. Section 3.3 presents the frame format that enables blind alignment, while Section 4 provides a detailed description of the experimental setup. Finally, Section 5 presents the main results of this paper while Section 6 discusses its implications for networks.

2. BACKGROUND & MOTIVATION

The number and types of wireless-featured devices has grown tremendously placing a heavy demand on limited spectral resources. Interference between these devices can significantly degrade performance and reduce the throughput per user. Therefore, the key to improving the performance of wireless systems lies in managing interference effectively. Interference alignment accomplishes this goal. Fundamentally, the concept of alignment is to extend modulated symbols to a higher dimensional space such that the interfering signals are all aligned and occupy the smallest subspace at each receiver. In this case, the remaining dimensions can be used to receive the desired signal essentially free of interference. Note that IA is not the same as spread spectrum, multiple access schemes or other interference minimization or avoidance schemes. IA makes no attempt to avoid, cancel or minimize interference. Instead, it aims to align the interference along dimensions that are different from that of the signal. As such, it has been found to perform much better than all the existing interference management schemes.

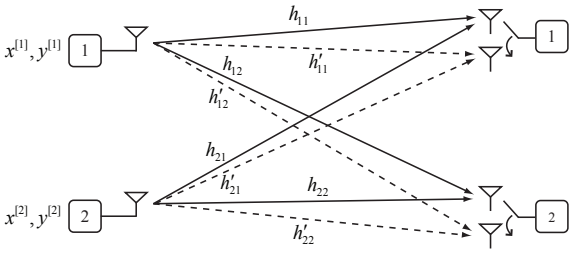


Figure 1: Blind Interference Alignment 2x2 system. Each receiver has 2 antennas to switch between.

As mentioned in the introduction, IA can achieve a DoF that increases linearly with the number of users in interference systems [1]. Note that this is quite amazing, as traditional time or frequency division multiple access schemes have no increase in DoF with users, thus each user only obtains a fraction of the total throughput which decreases with increasing number of users in the system. It is important to emphasize that this linear growth result is a theoretical one, and significant work needs to be done to bring the growing body of literature to practice.

2.1 Basic Principle

As discussed in the introduction, typical IA schemes require perfect global channel knowledge (i.e., perfect knowledge of all the channel coefficients in the system) at all the transmitters and receivers. But, this requirement is impractical in most systems. In this paper, we consider the blind (channel-state unaware) alignment scheme first presented in [10]. In this section, we provide a short description of the theory behind the blind IA technique, while challenges in practical implementation are detailed later. In our experiment setup, we consider two transmitters (TX1 and TX2) and two receivers (RX1 and RX2) as shown in Figure 1. Each transmitter $i = 1; 2$ has two messages $x^{[i]}; y^{[i]} \in \mathbb{C}$ which need to be recovered at RX1 and RX2 respectively. This is commonly referred to as the X-channel. Although the blind IA scheme is much more general, we focus our attention below to the specific 2-transmitter 2-receiver (referred to as 2x2 IA) setup for the sake of illustration. Our testbed will be more generic and allow for a higher order IA module.

In the 2x2 blind IA scheme, each receiver is equipped with a reconfigurable antenna which can switch between two different receive modes, but the receivers have only one RF chain. The receivers switch modes based on a predetermined pattern known to everyone. For $i; j \in 1; 2$, let $h_{ij} \in \mathbb{C}$ and $h_{ij}^{'} \in \mathbb{C}$ denote the coefficient of the channel between transmitter i and receiver j with the receive mode set to 1 and 2 respectively. The channel coefficients have a generic (nondegenerate) continuous probability distribution. It is assumed that the channel stays constant across a supersymbol, which constitutes three transmission slots.

At time instant t, suppose that $u^{[1]}(t); u^{[2]}(t) \in \mathbb{C}$ are transmitted from TX1 and TX2 respectively. In this case, the received signal at user j using receive mode 1 is given by

$$z^{[j]}(t) = h_{1j}u^{[1]}(t) + h_{2j}u^{[2]}(t) + w^{[j]}(t), \qquad (1)$$

where $w^{[j]}(t) \sim \mathbb{C}\mathcal{N}(0, \sigma^2)$ is the additive white Gaussian

Time Slot	1	2	3
Transmit	$x^{[i]} + y^{[i]}$	$x^{[i]}$	$y^{[i]}$
RX1	mode 1	mode 2	mode1
RX2	mode 1	mode 1	mode2

Table 1: Transmissions and receiver modes across 3 timeslots: mode corresponds to the antenna to which a receiver switches.

noise (AWGN). The received signal using mode 2 is obtained analogously using the channel coefficients $h^{'}_{1j}$ and $h^{'}_{2j}$ in Equation 1. The transmitters have absolutely no knowledge of the channel state. On the other hand, each receiver j is assumed to know only the local channel coefficients h_{1j}, h_{2j}, $h^{'}_{1j}$ and $h^{'}_{2j}$. Note that for many physical layer techniques these channel estimates are available for every received packet. For example, Orthogonal Frequency Division Multiplexing (OFDM) systems use known pilot subcarriers to compute channel estimates that they later use to equalize for the channel impairments. Therefore, this requirement on the receivers is very practical.

Recall that transmitter i has two messages $x^{[i]}, y^{[i]} \in \mathbb{C}$ intended for RX1 and RX2 respectively. Assuming perfect synchronization between the transmitters and receivers, the blind IA scheme operates in 3 time slots as shown in Table 1.

In time slot 1, each transmitter sends the sum of its two messages, while in timeslots 2 and 3 the messages for RX1 and RX2 are transmitted separately. The receivers switch between the two receive modes according to the pattern shown in Table 1. Due to the symmetry, the recovery process is analogous at the two receivers, and so we focus on RX1 below. The received signal $z^{[1]}(t)$ at RX1 over the three time slots, assuming the channel remains constant, is given by

$$\begin{aligned} z^{[1]}(1) &= h_{11}(x^{[1]} + y^{[1]}) + h_{21}(x^{[2]} + y^{[2]}) + w^{[1]}; \\ z^{[1]}(2) &= h^{'}_{11}x^{[1]} + h^{'}_{21}x^{[2]} + w^{[1]}; \\ z^{[1]}(3) &= h_{11}y^{[1]} + h_{21}y^{[2]} + w_{[1]}; \end{aligned} \tag{2}$$

Note that we have used the fact that RX1 switches to mode 2 in the second time slot. We now take advantage of the particular manner in which the interfering and desired signals are aligned in order to remove (i.e., zero-force) the interference and obtain the following system of equations:

$$\begin{pmatrix} z^{[1]}(1) - z^{[1]}(3) \\ z^{[1]}(2) \end{pmatrix} = \begin{pmatrix} h_{11} & h_{21} \\ h^{'}_{11} & h^{'}_{21} \end{pmatrix} \begin{pmatrix} x^{[1]} \\ x^{[2]} \end{pmatrix} + \begin{pmatrix} w^{[1]}(1) - w^{[1]}(3) \\ w^{[1]}(2) \end{pmatrix} \tag{3}$$

Since the channel coefficients are chosen from a continuous distribution, the channel vectors involved in Equation (3) are linearly independent with high probability. Therefore, RX1 can recover the desired messages $x^{[1]}$ and $x^{[2]}$.

A high- signal to noise ratio (SNR) performance metric of wireless networks is its degrees of freedom (DoF). DoF is defined as $lim_{SNR\to\infty} C_{sum}(SNR)/logSNR$, where $C_{sum}(SNR)$ denotes the maximum sum throughput achievable in the network. Thus, DoF denotes the asymptotic growth rate of throughput with SNR. Using the alignment scheme described above, we find that we can achieve a DoF of $\frac{4}{3}$. It has been shown that this is, in fact, the optimal DoF for this channel [3]. Thus, alignment promises in theory to achieve rates that are significantly larger than other schemes in existence.

In this paper, we explore the implementation of this promising theoretic idea. In our implementation, we look at a generic interference alignment framework; the 2x2 system discussed above is just to showcase the benefits and powerfulness of interference alignment. Here, we wish to highlight alignment, a physical layer coding scheme. Therefore, traffic models, including burstiness, are not currently incorporated into our framework. With different loads and varying traffic patterns, we anticipate different subsets of users to be chosen across time for the alignment procedure, where users participating in the alignment algorithm have non-empty queues. In the following section, we list the challenges in this implementation effort and discuss how our implementation overcomes them.

3. BLIND IA - IMPLEMENTATION CHALLENGES

Although blind interference alignment is considerably less complex in terms of the control and coordination needed between nodes in a network, it still presents implementation challenges inherent to any interference alignment technique. In this section, we go over these challenges and then present our plans to address them.

3.1 Implementation Assumptions

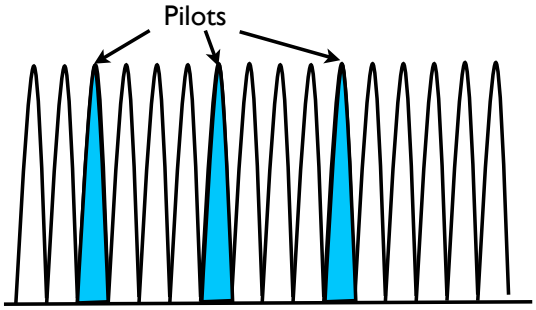
The first implicit assumption made in any interference alignment system, including blind alignment, is that transmitted symbols are synchronized. It is important to point out that this does not require carrier-level synchronism or sub-microsecond precision in synchronism. Even though it is relatively coarse, synchronization is essential - the lack of synchronism can lead to undesirable combinations at each receiver, significantly impacting the ability to align interference.

The second assumption is that each of the receivers can estimate the channel between them and each of the transmitters. This receiver side-information is a standard feature of any coherent communication system.

The third assumption is that each receiver can switch between antennas in real-time, i.e., choose between different receive 'modes'. As in the case of channel estimation, this antenna-switching is a relatively standard feature of multiple antenna systems, although implementing it in a testbed in real-time can be considerably involved as described further in Section 4. Note that asynchrony in the transmitted symbols can also impact the receiver's ability to switch between modes - the receivers rely on symbol boundaries to decide which antenna to switch to.

Finally, the channel is assumed to have a larger coherence time than the rate of antenna switching to be performed. Specifically, we assume that the channel gains between a particular transmit antenna-receive antenna pair remain fixed for a duration longer than the duration needed to affect alignment in interference. For example, the channel gains in Equation (3) remain the same in time-slots 1 and 3. This is, arguably, the most non-trivial assumption

(a) A Non-OFDM System



(b) An OFDM System

Figure 2: OFDM aids alignment by turning a frequency selective medium into a flat fading one per sub-carrier, easing symbol-level synchronism and channel estimation.

necessary for blind alignment. As we find in later sections, this assumption is valid for our test-setup. In general, such an assumption would hold true when the channel coherent time is much larger than symbol duration, which is typically the case for a majority of wireless systems.

3.2 Implementation

In this paper, we work with Orthogonal Frequency Division Multiplexing (OFDM) based physical layer. This is a natural choice for both our testbed and most existing and emerging wireless standards. There are multiple advantages to using an OFDM framework for our system. First, it enables us to transform the original frequency selective wireless channel into multiple flat-fading equivalents. This allows for the blind alignment technique as described in Section 2.1 to be utilized as is within each sub-band. Moreover, the duration of an OFDM symbol is considerably longer than duration of a symbol in a non-OFDM system. This larger symbol duration in an OFDM system helps improve error tolerance on the otherwise stringent requirement of symbol-level synchronization imposed by interference alignment. Finally, OFDM systems typically assign specific sub-carriers to pilot symbols. Figure 2(b) illustrates such an assignment of pilots to subcarriers. These pilots are typically used to estimate the channel at the receiver. This channel estimation plays a critical role in the design of the receive-chain at each receiver as well as in the blind alignment algorithm used in this paper.

An OFDM transmission typically includes a cyclic prefix (CP) before every OFDM symbol. For a tutorial on details underlying the physical-layer frame structure of OFDM transmission, see [13]. Figure 3 shows a physical layer frame with multiple OFDM symbols separated by CPs. This CP proves useful in facilitating antenna switching in our blind alignment scheme. When an antenna switching process is

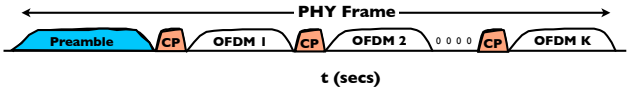


Figure 3: The physical layer frame showing multiple OFDM symbols. It starts with a preamble followed by OFDM symbols interspersed with cyclic prefixes.

initiated, a transient oscillation is observed in the received signal which stabilizes on the timescale of a few microseconds. The CP offers an excellent opportunity during which the switch between modes can be performed without impacting the successful reception of the OFDM symbol that follows the CP.

The preamble in every frame is conventionally used by the receivers to carry out time synchronization and frequency offset correction. To enable IA, the preamble can also be used to synchronize all transmitters in the system. For example, to enable synchronization, a single transmitter is chosen at any one time to send a preamble while the remainder of the transmitters capture and use this preamble to synchronize transmission. Note that many other mechanisms for achieving symbol-level synchronism exist in literature [14, 13], and that the mechanism described here is fairly simple but effective in satisfying the requirements for symbol-synchronization for IA algorithms.

3.3 Frame Format

The frame format for our setup consists of a preamble, pilots, and payload data symbols. The preamble consists of 2 OFDM symbols which are used in frame detection and frequency offset correction using the Schmidl-Cox algorithm [15]. The pilots are used for channel equalization when decoding the data symbols. In our implementation of the X-channel, both transmitters transmit their respective pilots simultaneously in the same OFDM symbol, but on alternating subcarriers. Transmitter 1 sends pilots on all odd subcarriers while transmitter 2 sends pilots on all even subcarriers. The orthogonal nature of the pilot transmission scheme allows the receiver to extract the channel state of both transmitters from a single OFDM symbol. Figure 4 shows the pilot to subcarrier mapping of the transmitted and received symbols. Our choice of pilots may impose some constraints on the scalability of the algorithm. However, scaling may be accomplished by determining suitable subsets of users that align their interference. Within those subsets, the pilots need not be scaled. Furthermore, alignment as structured in our paper is designed for environments with large coherence times. In such environments, the pilots consume a smaller fraction of resources compared to the case with smaller coherence times.

3.3.1 Frame Structure

Initially, we consider a somewhat naive frame format that is overhead-intensive, but constitutes a great starting point for our implementation. This format is shown in Fig. 5. This format is useful when each receiver possesses two receive chains. With two receive chains, all channel estimates can be obtained within one OFDM timeslot. This simple frame is sufficient to perform blind IA. However, it is highly inefficient, as the overhead incurred is substantial. An al-

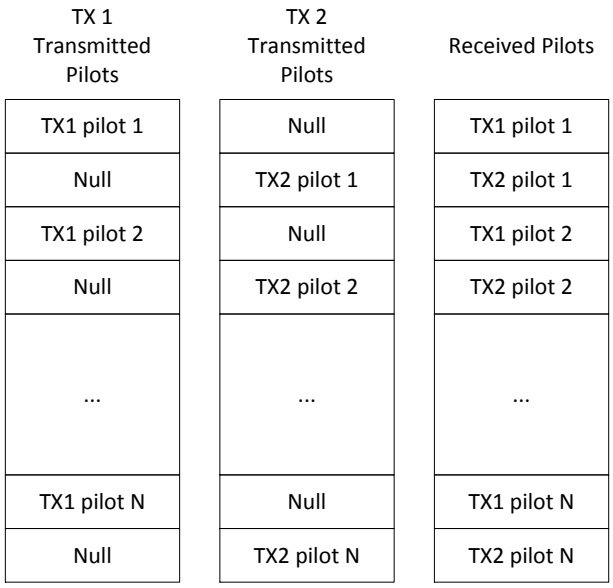


Figure 4: Pilot to subcarrier mapping

ternate frame structure that minimizes overhead is highly desirable, which is presented next.

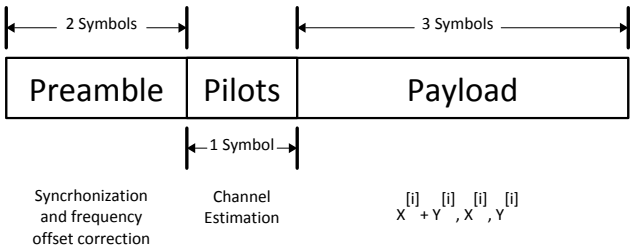


Figure 5: The first frame structure we consider to enable alignment. Here, there is a preamble and pilot for every 3 symbols, which is clearly in excess of what is needed.

3.3.2 Alternate Frame Structure

An alternate frame structure for IA includes added pilot symbols before each payload block and is shown in Figure 6. A payload block consists of a variable number of OFDM symbols of the same type. The first payload block consists of N OFDM symbols of the type $x[i] + y[i]$. The second and third payload blocks consist of N symbols of type $x[i]$ and N symbols of type $y[i]$, respectively. By grouping these symbols together in chunks, we can use the same preamble and pilots for more data, thus decreasing the frame overhead. This can increase efficiency so long as the channel coherency is maintained throughout the frame. Note that it is only necessary to have two pilot symbols per frame while our approach has three. This is because, an IA receiver only needs to get the channel estimates twice – one from each antenna. The third (optional) pilot symbol may be used to verify if the three payload pieces are still within channel coherence time.

3.4 Channel Coherence Time

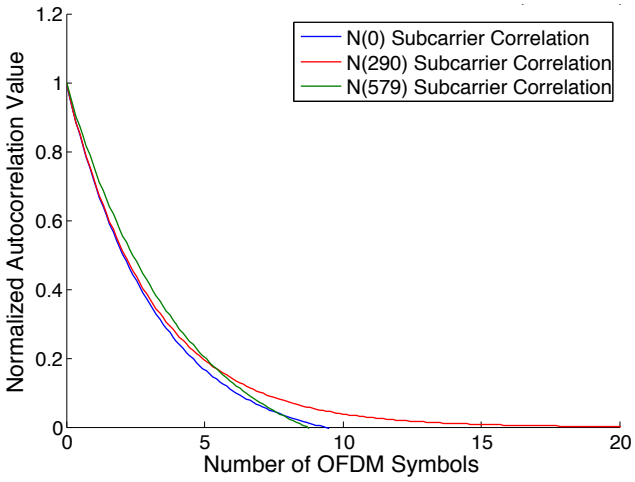


Figure 7: The channel coherence time in our environment. Showing only the phse of auto-correlation across multiple sub-carriers. The magnitude of autocorrelation (not shown here) had a coherence time of several hundreds of OFDM symbols.

As noted earlier, a key assumption made for blind IA operation is that the channel coherence time is much larger than a single OFDM symbol. To quantify coherence time, we ran an experiment in which a transmitter sends a series of OFDM symbols following a preamble. The preamble helps a receiver with synchronization and frequency offset correction. From the following OFDM symbols, a receiver computes for how long its channel to its transmitter remains correlated. We crunched this quantity for many sub-carriers over several experimentation runs. Figure 7 shows the auto-correlation across multiple sub-carriers for a transmitter-receiver pair. This result is representative of all of the other experiments. It shows only the phase of the auto-correlation and not the magnitude. The magnitude of autocorrelation showed correlation over nearly 600 OFDM symbols. The phase, however, as shown in the figure is still correlated over many multiple OFDM symbols before it becomes uncorrelated (with an autocorrelation of 0). The number of symbols in the case of phase varies from 8 to 20 depending on the sub-carriers. However, on average, the coherence time across all sub-carriers was around 12 OFDM symbols. For this reason, in the rest of the experiments, we restrict the number of OFDM symbols per payload (N in Figure 5) to 3. Thus, we have 3 pilot symbols and 9 (3x3) payload symbols equaling 12 OFDM symbols per physical layer frame. Note that IA does not require preamble to have the same channel realization as the rest of the frame.

4. EXPERIMENT DESCRIPTION

The hardware used in this experiment is the National Instruments' Software-Defined Radio (SDR) platform. This consists of a PXIe-8130 real-time controller, PXIe-7965R FlexRIO FPGA module with Xilinx Virtex 5 SX95T, NI-5781 100M samples/sec baseband transceiver, and an Ettus Research XCVR2450 daughterboard. A PXIe-1082 8-slot chassis with a high speed PXI-Express backplane bus can

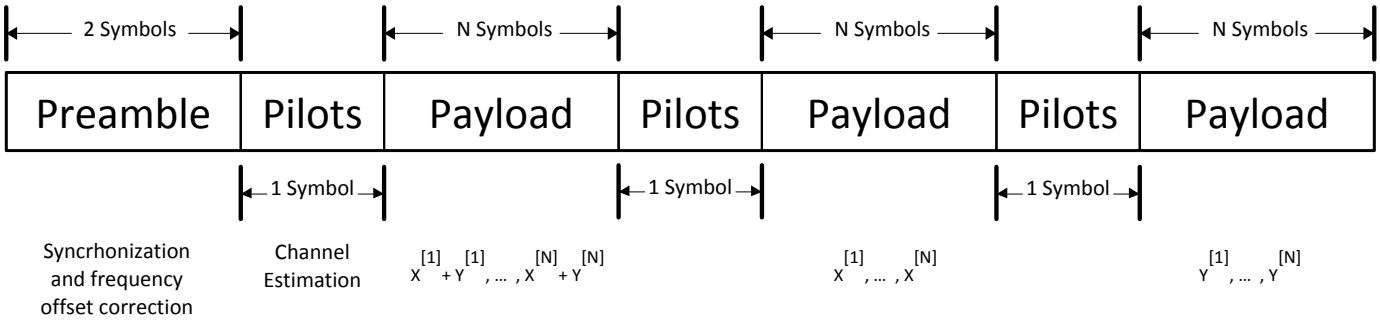


Figure 6: The efficient frame structure we use to enable blind IA

be mounted with up to 8 PXIe cards. Each module consisting of these components provides for a receiver or a transmitter. Figure 8 is a picture of the experimental hardware. The three white modules are the PXIe-1082 chassis, which are TX 1 & 2, RX1, and RX2 (going from the right to left). Each module is connected to an Ettus Research XCVR2450 daughterboard (two daughterboards for the transmit module) through SPI lines.All antennas are spaced such that.

The modulation scheme for this experiment is OFDM modulation with QPSK signaling. The operating frequencies are in the 2.4 GHz ISM band with a center carrier frequency of 2.437 GHz. Our OFDM modulation consists of subcarrier spacings of 15 kHz. Each symbol lasts $\frac{1}{12}$ ms in time, including a cyclic prefix length of 16.67 μs. These parameters are chosen based on the LTE standard. The 12 carriers around DC are nulled resulting in *N'= N - 12*. Furthermore, the $N = 600$ subcarriers are assigned to 580 data subcarriers (Nd) and 20 frequency pilots (Np). These frequency pilots are used to correct for a residual frequency offset.

Three PXIe-1082 chassis modules are employed in this experiment. One of these modules is fitted with two FlexRIO SDRs to control two transmitters TX1 and TX2. The remaining two modules function as receivers RX1 and RX2. Each receiver chassis has two receive modes provided by the Ettus XCVR2450 transceivers. The three chassis are connected through a local area network to development computers. LabVIEW, National Instruments graphical programming environment, is used to design and program the real-time controller. The FlexRIO FPGA module extends the LabVIEW to develop for the FPGAs in our experiment.

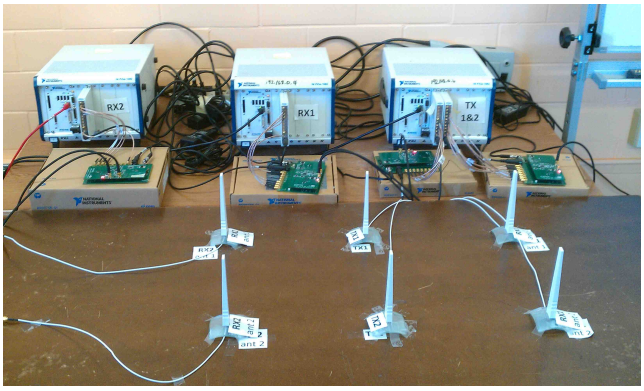


Figure 8: National Instruments SDR platform controls two transmitters (middle) and two recievers, each with two recieve modes (right and left antennas).

4.1 Transmitter

A block diagram of the transmitter is shown in Figure 9. Known psuedo-random bitstreams of length $2N'$ are extended to generate $2N$ bits. These bits are modulated with QPSK signaling to generate two streams of N samples $\boldsymbol{x}^{[i]} = (x_1^{[i]}, ..., x_N^{[i]})$ and $\boldsymbol{y}^{[i]} = (y_1^{[i]}, ..., y_N^{[i]})$. These samples are then precoded into the data symbols required for IA. The result in a payload of $3N$ samples $\boldsymbol{x}^{[i]} + \boldsymbol{y}^{[i]}$, $\boldsymbol{x}^{[i]}$, $\boldsymbol{y}^{[i]}$. Here $2N$ samples of preamble and N samples of pilots are inserted between each data sample to produce a $12N$ sample frame. We note that the second and third preambles are not neccesary to decode IA symbols. They are used for debugging purposes and are not included in the results obtained. Finally, the samples are OFDM modulated: the second and third preambles are modulated with a 3854 length and appended with a 1024 length cyclic prefix. This is done so it is not detected by the FPGA correlation algorithm. The final frame structure is 12 OFDM symbols in length as shown in Figure 6. TX1 and TX2 transmit their corresponding frames in a synchronous pattern.

4.2 Receiver

A block diagram of the reciever is shown in Figure 9. The reciever begins by aquiring two successive frames of samples. The beginning of the frame is detected by correlating over the known preamble. The preamble is also used to correct the frequency offset using the Schmidl-Cox algorithm [15]. The preambles are then removed and the rest of the frame is demodulated to result in $3N'$ payload samples and the 1 pilot symbol for each timeslot (3 total pilot symbols). Channel estimation is performed by selecting the appropriate pilot symbols depending on which reciever is operating. IA decoding then solves equation (2) to recover the $2N'$ payload data. Finally the aquired channel coefficients are used in ML detection to recover the transmitted bits.

4.3 FPGA Correlation Method and Antenna Switching

As mentioned earlier, for Blind IA to function, a receiver should be able to switch between its available antennas in

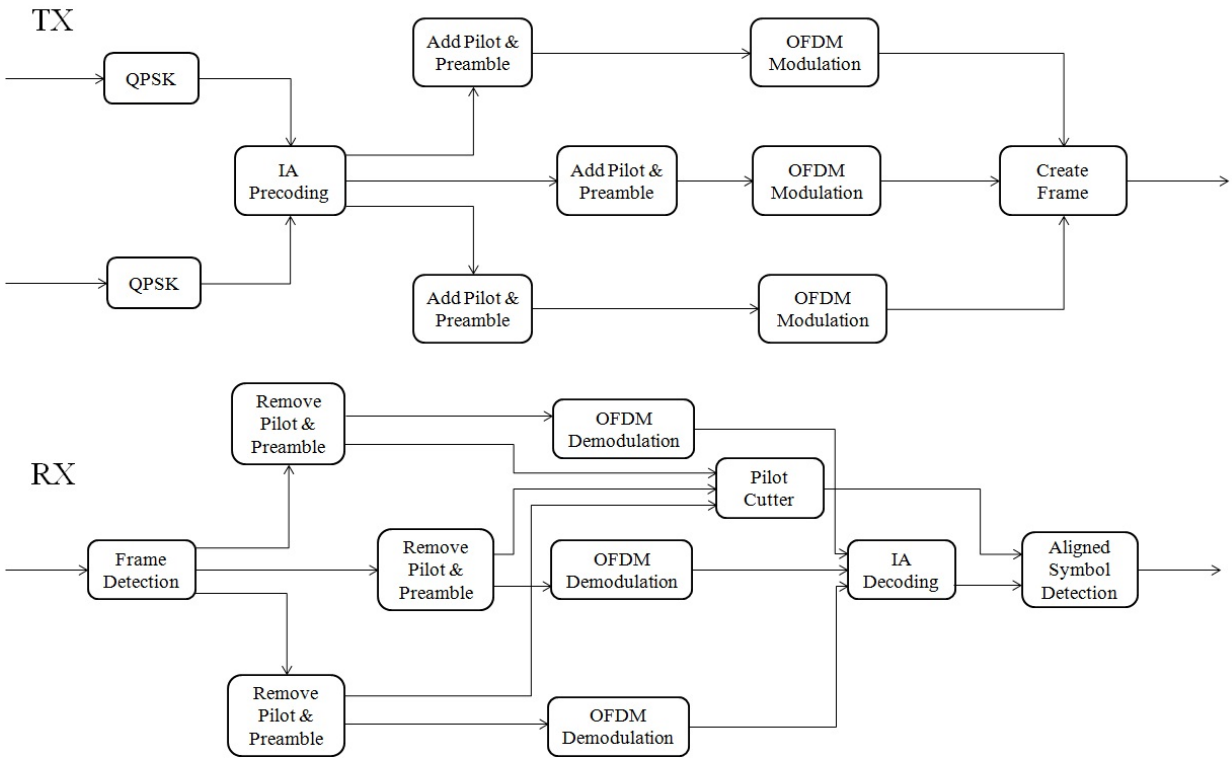


Figure 9: Block diagram of the transmitter (top) and receiver (bottom).

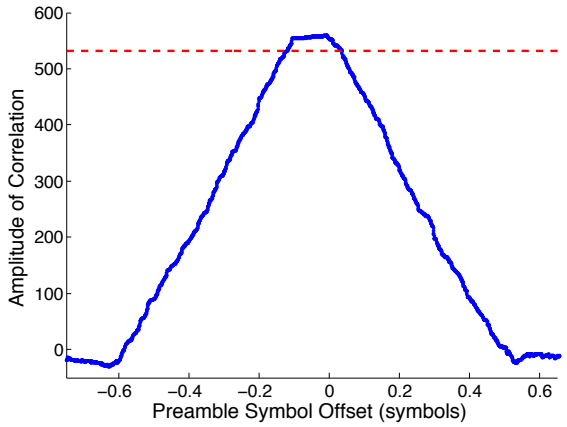


Figure 10: Symbol timing estimation performed on the FPGA. The horizontal dotted line is the user-specified threshold value for detecting peaks. The threshold value tracks the highest amplitude peak.

real-time. The accuracy of antenna switching is entirely dependent on OFDM symbol timing estimation on the FPGA. In this real-time implementation, we move the Schmidl-Cox symbol timing estimation [15] to high speed FPGAs. The challenges of moving this algorithm to FPGAs are the inherent computational restrictions imposed by the FPGA. As a result, FPGA correlation is sensitive to channel variations and SNR because we cannot normalize our estimated timing. Our method employs a variable amplitude correlation with a threshold peak detection. The largest peak over time is tracked and any amplitude that exceeds a threshold percentage of the largest peak is determined to be that frame start. A fluctuation in the channel will irreversibly spike the threshold value to a false peak, causing future preambles to evade detection. Figure 10 is a correlation plot of two identical preamble symbols with a 512 length cyclic prefix. Notice this correlation is not normalized due to previously mentioned FPGA processing constrictions.

Once the preamble is detected, the precise switching time is simply a known offset from the beginning of the frame. The FPGA issues commands to the baseband transceiver through SPI interfacing on a 40MHz clock. The antenna switching command can now be issued through a finite state machine that is configured for SPI protocol. This process automates antenna switching, making it invisible to the reciever. Recieved frames now have the proper antenna switches performed and IA decoding can run independently.

5. RESULTS

This section presents the bit error rate and throughput performance results of our blind IA system compared to a

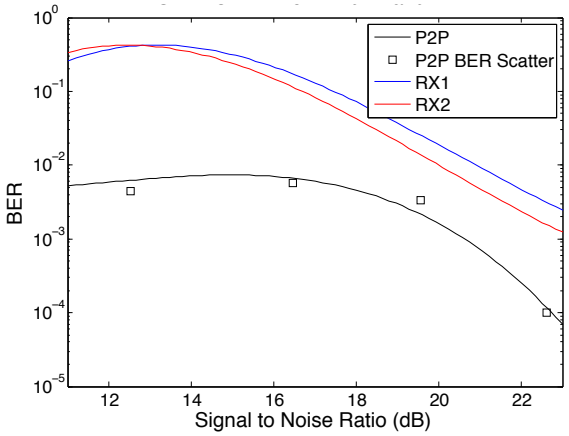


Figure 11: Signal to noise ratio (SNR) vs bit error rate (BER) performance of IA and time division point-to-point (P2P) systems. BER for IA is always worse than P2P.

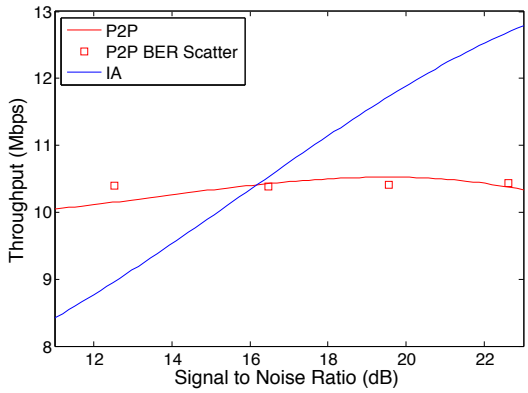


Figure 12: Signal to noise ratio (SNR) vs throughput performance of IA and point-to-point systems. Throughput of IA is worse than that of point-to-point up to an SNR of 16dB. In the very high SNR region (beyond 16dB), however, IA's throughput is better than P2P.

time division point-to-point system. It also presents the results for IA when the transmitters are not perfectly synchronized. Overall, when the two transmitters are perfectly synchronized, IA delivers superior throughput when the SNRs to the receivers are very high. As the transmitters get out-of-sync, however, IA's BER performance remains consistent up to a certain value and deteriorates sharply after that. This shows that IA is a promising strategy even in practice.

5.1 Bit Error Rate (BER) Performance

Figure 11 shows the bit error rate (BER) performance of IA and time division point-to-point (P2P) systems. BER for IA is worse than that for time division across the whole range of SNR. Note that, in our IA system, the two transmitters send the same preamble. Therefore, SNR for just the preamble is twice as high as it is for the corresponding time division P2P system. However, the rest of the frame contains symbols from both transmitters at similar power levels. Therefore, the rest of the frame would have a very low signal to interference and noise ratio (SINR). However, in the time division scenario, the SNR remains the same across the entire frame. Therefore, to illustrate the results for the same setup, our plots show the SNR for the point-to-point time division system for the same transmit power levels as in the IA case.

5.2 Throughput Performance

The measured throughput is at the link level for both IA and P2P systems. We calculated throughput as the amount of data bits recieved per frame (Figure 6), or packet, as bounded by the sampling rate 30.72 $\frac{Msamples}{sec}$. The total amount of transmitted bits per second were then multiplied by the successful recovery probability, $1-BER$, to compute the final aggregate througput in bps. Additionally, we assume an infinite input buffer for our input traffic, and we focus on physical layer performance. Thus, we do not retransmit any packets. Therefore, there is no ARQ/HARQ in our setup. The BER and throughput values are based on the raw bits that are transmitted and received.

Figure 12 shows the aggregate (or sum) throughput results for both the time division P2P and IA systems. It

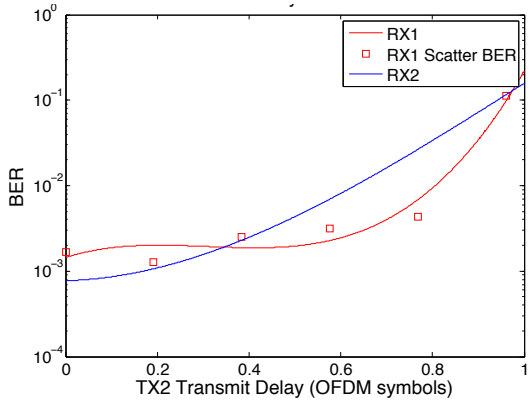


Figure 13: Signal to noise ratio (SNR) vs bit error rate (BER) performance of IA and time division systems. BER of IA is better than time division at very low and very high SNR regions. In the low and high SNR regions, pilots from two transmitters slightly improve BER performance for IA.

shows that the sum throughput for IA is worse than point-to-point up to an SNR of 16dB. Beyond that, however, IA starts to outperform P2P. IA achieves a throughput gain of close to 1.3 when the SNR hits 23dB. Note that this is very close to the maximum gain (its DoF corresponds to $\frac{4}{3} = 1.33$) for a 2x2 IA scheme over time-division point-to-point system. It is important to note that the 1.33x gain is under information-theoretic degrees of freedom calculations; and that in practice, the MAC and other practical coding and signaling aspects of the system impact both the non-aligned and aligned systems, reducing the performance below the information theoretic optimum.

5.3 Synchronization Mismatch

One of the aspects that is critical for IA operation is the synchronization of symbols across transmitters. Figure 13 shows the synchronization mismatch between the two transmitters. The x-axis corresponds to the extent to which

transmitter 2's transmission is delayed. This delay (or mismatch) is given in terms of an OFDM symbol, which is about $\frac{1}{12}$msec. It shows that IA's BER performance worsens as the mismatch increases. A mismatch equalling 1 OFDM symbol causes IA to completely fail. However, a small asynchronism in the order of a cyclic prefix (of 0.2 OFDM symbol) corresponds to a reasonable receiver BER. This is a mismatch of roughly 17usecs for our system. This shows that IA's performance is fairly robust to small mismatches.

5.4 Higher-Order Blind IA

This paper focuses its efforts on implementing a 2x2 blind IA system. It shows that the throughput gain promised in theory is achievable in practice using blind IA in a 2x2 system. For a higher-order (3x3 and beyond) IA system, however, the gains from IA will be higher and will increase with the order [3]. Our current RF front-end only supports switching between 2 antennas. Due to this hardware limitation, we are unable to demonstrate throughput gains for higher-order blind IA systems. However, our physical layer design including the frame format can be easily extended to a higher-order blind IA set up.

6. DISCUSSION

This section discusses the implications of employing interference alignment (IA) in existing networks and network architectures.

Implications to Access Networks: In an access network, multiple wireless clients connect to the Internet through a wireless Access Point (AP). As an AP serves multiple clients, a single AP can serve the purpose of multiple transmitters in an IA system. For example, in our 2x2 setup, an AP could perform the duties of two transmitters and its two clients could serve as its two receivers. By having two transmitters combined into a single AP, issues such as transmitter synchronization would disappear.

Higher Layer Implications: Interference alignment (IA) is a cooperative technique in which multiple transmitters send data to multiple receivers simultaneously. This is unlike MIMO, where only one transmitter is sending data to one receiver using multiple antennas. A MIMO-like point-to-point system works without any modifications to the current wireless network architecture. However, since IA requires synchronization among multiple nodes, the current network architecture needs modifications to take full advantage of the throughput gains from IA.

Channel Estimation: In IA, every receiver needs to be able to estimate the channel between itself and every IA transmitter. To achieve this, different transmitters may be assigned different subcarriers for pilots and when the pilots are transmitted no other transmitter may use that subcarrier to send anything (as discussed in Section 3.3). Thus, the number of pilots for IA increases linearly with the number of transmitters. Such a scalability requirement is inherent to any technique that needs coordination. However, this overhead can be made considerably small when channel coherence is very large (compared to an OFDM frame) or by adjusting the number of IA transmitter-receiver pairs.

MAC Layer Functionality: In a traditional wireless network, a node is responsible for doing carrier sensing before it transmits any data to avoid collisions. In an IA network, the responsibility is on all the nodes that participate in an IA session. Note that not all the nodes in a network may participate in every IA transmission. IA promises orders of magnitude throughput gains only when all of the participating receivers have excellent signal to noise ratios (SNRs) from all of the participating transmitters. Therefore, a network may need to be split into several sets. Each set still needs to do carrier sensing before it chooses to transmit a packet, to avoid collisions. So, with IA, nodes within a set share the responsibility of channel sensing. To save energy, only a subset of nodes could participate in channel sensing.

Network Layer and Above: Note that, in IA, every transmitter can potentially send packets to multiple receivers simultaneously. In a mesh network, where multiple flows crisscross, the crisscross point becomes the bottleneck for all the flows. IA may be used at the crisscross point to mitigate such bottlenecks.

Acknowledgements

We would like to thank Prof. Syed Jafar, Prasanth Anthapadmanabhan and Saul Duran for their discussions. We would also like to thank our shepherd, Supratim Deb.

7. REFERENCES

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

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

[3] M. Maddah-Ali, A. Motahari, and A. Khandani, "Communication over MIMO X channels: Interference alignment, decomposition, and performance analysis," *Information Theory, IEEE Transactions on*, vol. 54, 2008.

[4] A. Motahari, S. Gharan, and A. Khandani, "Real interference alignment with real numbers," *Arxiv preprint arxiv:0908.1208*, 2009.

[5] B. Nazer, S. Jafar, M. Gastpar, and S. Vishwanath, "Ergodic interference alignment," in *Information Theory, 2009. ISIT 2009. IEEE International Symposium on.* IEEE, 2009, pp. 1769–1773.

[6] A. Jafarian, J. Jose, and S. Vishwanath, "Algebraic lattice alignment for K-user interference channels," in *Communication, Control, and Computing, 2009. Allerton 2009. 47th Annual Allerton Conference on.* IEEE, 2009, pp. 88–93.

[7] R. Krishnamachari and M. Varanasi, "Interference alignment under limited feedback for MIMO interference channels," in *Information Theory Proceedings (ISIT), 2010 IEEE International Symposium on.* IEEE, 2010, pp. 619–623.

[8] H. Bolcskei and I. Thukral, "Interference alignment with limited feedback," in *Information Theory, 2009. ISIT 2009. IEEE International Symposium on.* IEEE, 2009, pp. 1759–1763.

[9] M. Maddah-Ali and D. Tse, "On the degrees of freedom of MISO broadcast channels with delayed feedback," *EECS Department, University of California, Berkeley, Tech. Rep. UCB/EECS-2010-122, Sep*, pp. 2010–122, 2010.

[10] T. Gou, C. Wang, and S. Jafar, "Aiming perfectly in the dark-blind interference alignment through staggered antenna switching," in *GLOBECOM 2010, 2010 IEEE Global Telecommunications Conference*. IEEE, 2010, pp. 1–5.

[11] S. Gollakota, S. Perli, and D. Katabi, "Interference alignment and cancellation," in *ACM SIGCOMM Computer Communication Review*, vol. 39, no. 4. ACM, 2009, pp. 159–170.

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

[13] A. B. Narasimhamurthy, C. Tepedelenlioğlu, and M. K. Banavar, *OFDM Systems for Wireless Communications*. Morgan and Claypool Publishers, 2010.

[14] D. Lee and K. Cheun, "Coarse symbol synchronization algorithms for OFDM systems in multipath channels," *Communications Letters, IEEE*, vol. 6, no. 10, pp. 446–448, 2002.

[15] T. Schmidl and D. Cox, "Robust frequency and timing synchronization for OFDM," *Communications, IEEE Transactions on*, vol. 45, no. 12, pp. 1613–1621, 1997.