

# Towards Programming the Radio Environment with Large Arrays of Inexpensive Antennas

Zhuqi Li<sup>1</sup>, Yaxiong Xie<sup>1</sup>, Longfei Shangguan<sup>1</sup>, R. Ivan Zelaya<sup>2</sup>  
Jeremy Gummeson<sup>3</sup>, Wenjun Hu<sup>2</sup>, Kyle Jamieson<sup>1</sup>

<sup>1</sup>Princeton University, <sup>2</sup>Yale University, <sup>3</sup>UMass Amherst

## Abstract

Conventional thinking treats the wireless channel as a constraint, so wireless network designs to date target endpoint designs that best utilize the channel. Examples include rate and power control at the transmitter, sophisticated receiver decoder designs, and high-performance forward error correction for the data itself. We instead explore whether it is possible to reconfigure the environment itself to facilitate wireless communication. In this work, we instrument the environment with a large array of inexpensive antenna (LAIA) elements, and design algorithms to configure LAIA elements in real time. Our system achieves a high level of programmability through rapid adjustments of an on-board phase shifter in each LAIA element. We design a channel decomposition algorithm to quickly estimate the wireless channel due to the environment alone, which leads us to a process to align the phases of the LAIA elements. Variations of our core algorithm then improve wireless channels on the fly for single- and multi-antenna links, as well as nearby networks operating on adjacent frequency bands. We implement and deploy a 36-element LAIA array in a real indoor home environment. Experiments in this setting show that, by reconfiguring the wireless environment, we can achieve a 24% TCP throughput improvement on average and a median improvement of 51.4% in Shannon capacity over baseline single-antenna links. Over baseline multi-antenna links, LAIA achieves an improvement of 12.23% to 18.95% in Shannon capacity.

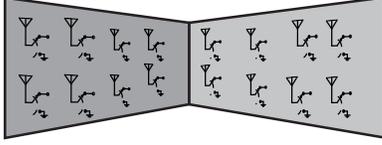
## 1 Introduction

While the vision of the Internet of Things is rapidly becoming a reality, more and more wireless devices now crowd the wireless spectrum both at home and in enterprise. Largely unplanned wireless networks such as Wi-Fi and Zigbee have proven their utility, yet still degrade in performance when the number of radios operating in close proximity scales too far. Network designers are aware of this wireless “success disaster,” and have proposed many different approaches to

address the problem, but most—if not all—of the solutions thus far formulated treat the wireless channel as a constraint within which endpoint radios work to maximize throughput.

Prior work on improving wireless networks in isolation has followed two broad themes: (i) wireless modulation and channel coding schemes that aim to best utilize the wireless channel, and (ii) diversity schemes that aim to minimize periods of time when the wireless channel is inoperable, *i.e.*, outages. Modulation and channel coding have proved a remarkable success, achieving the information theoretic channel capacity both for static channel scenarios through fixed-rate LDPC codes [12], and for dynamic channel scenarios through rateless Spinal Codes [27]. Diversity schemes have taken on many forms in terms of time, space, and frequency diversity. Space diversity schemes generally fall into two categories. The first involves exploiting diversity among different communication endpoints to route data around failures of individual links. Schemes such as ExOR [7] leverage multiple relays, and MRD [26] and SOFT [39] improve performance through multiple APs serving mobile clients. Multiple Input, Multiple Output (MIMO) links [14,28,29,34,42] use multiple antennas to exploit spatial diversity across the multiple paths that make up a single wireless link. OFDM [35] makes efficient use of the wireless channel across different frequencies.

While the above ideas have reaped significant performance benefits, the vast majority of the design innovation has heretofore taken place at the *endpoints* of the wireless links, leaving the wireless channel itself unchanged. This paper explores a new approach: can we instead build a smarter radio environment, one that electronically reconfigures itself to the communication happening at any particular instant in time? After all, the wireless channel is the result of multipath signal propagation through the ambient environment. If we could configure signal propagation behavior at will, we could instead *create* more favorable channel conditions for wireless communication over the same spectrum. Since the number of reflectors, diffractors, and absorbers in the environment potentially dwarfs the number of antennas at the communication endpoints, more degrees of freedom may result from chang-



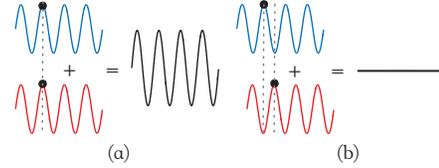
**Figure 1:** Building walls instrumented with LAIA, which modulate wireless signals incident on the wall, configuring the environment to improve wireless communication.

ing the environment itself rather than the communication endpoints. For this reason, our vision qualitatively differs from recent proposals to cover walls with conductive paint [43], add static reflectors [8, 16, 41, 44], or programmable phased-array reflectors [1, 2] for 60 GHz links in the environment. Furthermore, our vision raises new questions: is dynamically reconfiguring the environment even feasible—let alone in real time—in the face of constantly-changing channel conditions?

The approach we take in this paper is to augment the indoor environment with a *large array of inexpensive antennas (LAIA)*, as illustrated in Figure 1. Each array element is programmable and capable of dynamically shifting the phase of the wireless signal propagating through it. These elements form a *substrate* that can rapidly modulate the signal propagation characteristics of the *environment* itself as needed to improve communication. Since the main determinant of wireless multipath fading and interference are the phase offsets between signals arriving along different paths at each radio receiver, shifting the phases of the signals passing the LAIA array can generate desirable constructive or destructive signal superposition at the receiver. Therefore, the LAIA substrate is uniquely positioned to program these signals, and hence the radio environment.

Different LAIA configurations are desirable for different goals. For instance, if the desired outcome is to boost the performance of an individual link, LAIA will likely effect different phase shifts on the wireless signals passing through it compared to a scenario where the goal is to optimize two different links. Other possibilities include the handling of single-antenna (single-input, single-output, *i.e.*, *SISO*) links versus MIMO links. Therefore, if we can adjust the phase offsets on these paths, we can generate desirable signal alignment or separation—this process is explained further in §2.

In Section 3.1 we present the design of the LAIA element, which consists of two antennas and a programmable phase shifter. To control as many paths as possible, we deploy a large array of LAIA elements in the environment, connecting the array to a *controller*. We then tackle the central challenge of the LAIA architecture: how to design a control algorithm for the substrate, whose input consists of measurements at the endpoints and whose output consists of a time series of phase shifts (one such time series for each LAIA element) that the substrate should actuate. The design space of this control algorithm is significant, and so LAIA’s control algorithm



**Figure 2:** Signals propagating on two paths combine (a) constructively and (b) destructively based on their relative phase offsets.

focuses on a particular subset of that design space:

1. Scaling to large numbers of LAIA elements given the channel coherence time constraint;
2. Incorporating wireless channel measurements at communication endpoints, as supported by Wi-Fi standards, but piggybacking on regular data traffic;
3. Configuring the LAIA array to accomplish several different objectives simultaneously.

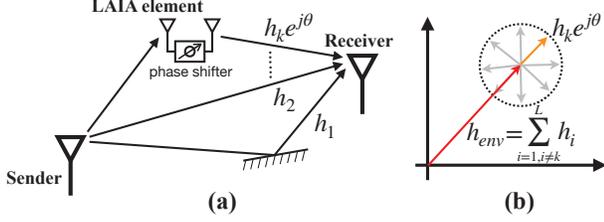
We describe our hardware and software implementation in Section 4. Section 5 describes a LAIA testbed comprised of 36 array elements deployed in a residential house. Extensive experiments on this 36-element testbed show LAIA can achieve a 24% TCP throughput improvement on average and a median improvement of 51.4% in Shannon capacity over the baseline single-antenna links. Over the baseline multi-antenna links, LAIA achieves an improvement of 12.23% to 18.95% in Shannon capacity.

## 2 Programming the radio environment

The indoor wireless channel is comprised of a collection of *propagation paths* in the environment. Radio signals radiated from the transmitter are reflected, diffracted, and scattered by multiple surfaces and objects, which cause them to traverse different propagation paths before reconvening at the receiver. The net effect of each path is captured by an attenuation in signal strength and a phase shift, and so can be modeled by  $h_i = \alpha_i \cdot e^{j\phi_i}$  with amplitude  $\alpha_i$  and the phase  $\phi_i$  (for the  $i^{\text{th}}$  path). The  $L$  propagation paths superimpose at the receiver, constructively or destructively, based on their relative phase offsets, resulting in a wireless channel  $h_{env} = \sum_{i=1}^L h_i$ . Figure 2 shows two examples where two signals are completely in phase (add constructively) or out of phase (add destructively).

Consequently, if we can configure individual signal propagation paths, we can generate different channel profiles. This can be achieved by changing the phase of a path: as Figure 2 shows, when we change the phase of a propagation path, the amplitude of the combined signal at the receiver may change significantly, in the case that we turn destructive superposition into constructive superposition, for instance.

In other words, by reconfiguring the phase offset on each propagation path, we can in fact program the overall radio environment! This suggests deploying a collection of phase-



**Figure 3:** (a) The signal travels through the LAIA element superimposes with signals travels along all other propagation paths at the receiver. (b) By adjusting the phase shift  $e^{j\theta}$ , the signal from LAIA element to the receiver aligns completely with other signals from the sender to the receiver.

shifting signal relays (which we term LAIA *elements*), each with one antenna for transmission and reception, as shown in Figure 3(a). The effect of a LAIA element can be modeled by adding a phase shift of  $\theta$  on the  $k^{\text{th}}$  propagation path term passing through the LAIA element between the sender and the receiver, *i.e.*,

$$h_{comb} = \left( \sum_{i=1, i \neq k}^L h_i \right) + h_k \cdot e^{j\theta} \quad (1)$$

as shown in Figure 3(b), we can adjust  $\theta$  so that the signal from the LAIA element to the receiver aligns completely with the other signals from the sender to the receiver.<sup>1</sup>

## 2.1 LAIA: System goals

With the ability to configure individual propagation paths, there is potential to address many link- or even network-level performance goals by adopting different signal alignment or separation strategies. LAIA takes the first step in this direction, highlighting the potential of programming the environment by addressing the following performance goals:

**Goal 1:** Removing the channel *null* that is associated with frequency-selective fading, for a single link, thus improving its throughput (§5.1).

**Goal 2:** Optimizing the throughput performance of individual single- (§5.2) and multi-antenna links (§5.3).

**Goal 3:** Jointly optimizing the throughput of two networks operating on different frequencies in close proximity (§5.4),

**Goal 4:** Substituting for and/or improving the performance of multi-AP diversity schemes (§5.2).

**Discussion.** Goal 1 is the least ambitious, simply requiring that different propagation paths be co-phased at a particular frequency, to fill a null at that frequency, analogous to transmit beamforming [4]. Goal 2, however, requires considering the different effects of each LAIA element on multiple

<sup>1</sup>It is also possible to use single-antenna elements that alter the *reflected* propagation paths from the sender to the receiver, but we focus on a relay design in this work.

OFDM subcarriers (frequencies) at the same time, substantially complicating the problem. Multi-network and AP diversity scenarios (Goals 3 and 4) can be viewed as a problem of beamforming on multiple links (with different channels) simultaneously with a single beamforming matrix.

## 2.2 LAIA: Design challenges

**A substrate design balancing cost and efficacy.** As each LAIA element provides only limited programmability, we need a large array of inexpensive antennas to effectively program the whole environment. The minimalist element design of §3.1 addresses this challenge, and §4.1 gives an indication of the cost.

**A scalable control plane design.** The LAIA control plane orchestrates the many LAIA elements to achieve the desired effects. The first challenge here is accurate *estimation* of each LAIA element’s wireless channel  $h_{pi}$  as well as the environmental channel  $h_{env}$  between the two communication endpoints, where only measurements of the combined wireless channel are available at the receiver. The second challenge is calculating the most desirable individual element phase settings to *align* the wireless signals, given an enormous search space in the control plane. Further, both channel estimation and alignment challenges are exacerbated by the array size of LAIA and the need to completion both tasks within the channel coherence time. Our channel estimation, *flip-and-align*, and *channel alignment* algorithms (§3.2) address each of these challenges in turn.

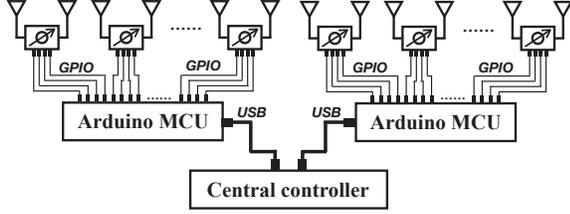
**Multi-objective control.** At any given time instant, the LAIA substrate can be in at most one configuration. Therefore, when considering multiple subcarriers’, links’ and/or networks’ simultaneous operation, we require algorithms to address these joint optimization challenges. §3.3 explains how LAIA solves these.

## 3 Design

We first describe the individual element design for LAIA, followed by the overall system architecture including the hardware and control infrastructure (§3.1). We then delve into the control algorithm (§3.2), covering channel estimation (§3.2.1) and alignment of the entire LAIA array (§3.2.2). We conclude the section by explaining algorithms to optimize wireless channels on the fly for single- and multi-antenna links, as well as nearby networks operating on adjacent frequency bands (§3.3).

### 3.1 Element design and control architecture

Each LAIA element is minimalist and passive, simply consisting of two antennas connected to either end of a phase shifting device and without drawing any power, as shown in Figure 4.



**Figure 4:** The architecture of the LAIA array. The LAIA element is controlled by Arduino MCU through GPIO. One Arduino is able to control up to 10 elements. A large array is controlled by multiple Arduino boards (two in the figure). All of the Arduino boards are connected to a central controller through USB 3.0 interface. The control algorithm is implemented on the central controller.

A wireless signal received by one antenna is thus shifted in phase and then emitted to the environment through the other antenna, and vice-versa for the wireless signal received by the other antenna. A purely passive radio chain significantly reduces the energy consumption and costs of each LAIA element. Each LAIA element is programmed by modulating the phase shifter, using the micro-controller (MCU) of an Arduino board as a controller to configure the phase shifter through general purpose I/O pins, as shown in Figure 4. Since each LAIA element is a passive device, it is low cost, but provides only a limited impact on the environmental channel, which we quantify in Section 5.1.

LAIA’s control architecture is also shown in Figure 4. The phase shifters, which control up to 10 individual elements, are configured through the low-cost Arduino MCU (the figure gives an example of a LAIA array using two Arduino boards). Each Arduino board is connected to a central controller via a USB 3.0 interface. The central controller collects channel measurements from the AP, runs our control algorithm and distributes the control signals, *i.e.*, the amount of phase shift on each element, to the Arduino boards, which in turn configure the phase shifters of all the connected elements.

## 3.2 Control algorithm

We now present the control algorithm implemented on the LAIA controller. Our algorithm aims to find the configuration, *i.e.*, the phase shifts of LAIA elements in the array, that can accomplish a certain performance goal. We first introduce our channel estimation algorithm, followed by the channel alignment algorithm that aligns all the programmable element channels with the environmental channel for maximum capacity. Variations of the channel alignment algorithms optimize wireless channels on the fly for multi-antenna links, as well as nearby networks operating on adjacent frequency bands.

To program the environment, we need to first estimate the channels of all LAIA elements and the environmental channel and then align them, just as Figure 3(b) shows. The element

channels and environmental channel are linearly superimposed at the receiver. To estimate each individual channel using only the superimposed channel measurement, we can randomly configure the elements and measure the superimposed channel multiple times, using regular data frames from the on-going communication for overhead-free measurement. Individual channel measurements are uncorrelated, so we can estimate the element and environmental channel by solving the linear equations. On the other hand, channel alignment requires all elements to be configured with the optimal phase rotation so that they can add constructively. We implement a controller which resolves such a conflict and simultaneously estimate and align the channels.

### 3.2.1 Channel decomposition

Our first task is to separate (decompose) the ambient environment’s channel  $h_{env}$  from the channel corresponding to the propagation paths traversing the  $i^{th}$  LAIA element. Suppose the  $i^{th}$  LAIA element is configured with phase setting  $\theta_i$ . We refer to the channel through that LAIA element in isolation as  $e^{j\theta_i}h_{p_i}$ . With LAIA in mind, we can model the combined wireless channel differently than the traditional wireless channel model of Equation 1 and instead write the combined channel in terms of the ambient environment and each of  $M$  LAIA elements:

$$h_{comb} = h_{env} + \sum_{i=1}^M e^{j\theta_i}h_{p_i} \quad (2)$$

We start from a simplified algorithm to decompose the channel in the presence of only a single LAIA element before generalizing to the realistic, multi-element case.

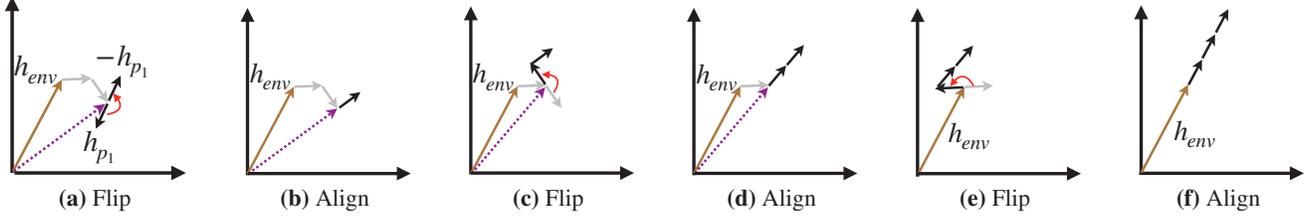
**Single LAIA element case.** In the case of single element, the channel is comprised of the environment  $h_{env}$ , plus that element’s contribution,  $h_{p_1}$ , supposing the element is initially configured at  $0^\circ$  phase. We first measure the channel with the element in the  $0^\circ$  state ( $h_{env} + h_{p_1}$ ), then flip the element’s phase to  $180^\circ$  and measure the channel again ( $h_{env} - h_{p_1}$ ). The two resulting measured channels can be represented in matrix form as

$$\begin{bmatrix} h_{comb}^1 \\ h_{comb}^2 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} h_{p_1} \\ h_{env} \end{bmatrix} \quad (3)$$

and the resulting linear system solved for  $h_{env}$  and  $h_{p_1}$ . Once the environmental channel is known, we choose  $\theta_1$  such that  $\angle e^{j\theta_1}h_{p_1} = \angle h_{env}$ , thus phase-aligning the environment and LAIA element channels with each other.

**Multiple LAIA elements.** We now consider the realistic case where  $M$  LAIA elements are present. We form the channels to be decomposed into an  $(M + 1)$ -dimensional vector comprised of the  $M$  LAIA elements along with the environmental channel itself:

$$\mathbf{h} = [h_{p_1}, h_{p_2}, \dots, h_{p_M}, h_{env}]^T \quad (4)$$



**Figure 5:** Channel estimation process for three LAIA elements. Without conscious alignment, the environmental channel and all LAIA elements superimpose randomly in (a). Instead, we estimate and then align the channels of elements 1, 2 and 3 in (b), (c), and (d), respectively.

Then  $M + 1$  channel measurements with different LAIA element phase settings can be expressed as another  $M + 1$  dimensional vector  $\mathbf{h}_{\text{comb}} = [h_{\text{comb}}^1, h_{\text{comb}}^2, \dots, h_{\text{comb}}^{M+1}]^\top$ , where

$$\mathbf{h}_{\text{comb}} = \mathbf{Q} \cdot \mathbf{h} \quad (5)$$

and  $\mathbf{Q}$  is a *control matrix* whose  $M + 1$  rows correspond to  $M$  different phase settings of each LAIA element (adjoined with 1 for the environmental channel which we cannot change) and whose  $M + 1$  columns correspond to each of  $M + 1$  channel measurements. We now decompose the channel, obtaining  $\mathbf{h}$ , by solving the linear system of Equation 5, and that this solution exists whenever matrix  $\mathbf{Q}$  is full rank.

We could construct a full rank control matrix by randomly configuring the LAIA elements for each of the  $M + 1$  measurements involved. But this would generate random contributions to the perceived  $\mathbf{h}$ , and so the intermediate combined channel  $\mathbf{h}_{\text{comb}}$  would be suboptimal. Hence we instead adopt a channel decomposition strategy such that we can achieve some alignment along with each additional channel measurement, even before we construct the entire control matrix  $\mathbf{Q}$ .

**Flip-and-align algorithm.** LAIA’s approach simultaneously decomposes the wireless channel and increasing the signal to noise ratio of the channel being decomposed while the measurements are taking place, thus making it more practical for networks where traffic is ongoing and present in the background of the measurement process. In fact, the channel measurements are piggybacked on ongoing traffic.

We use a three-element example to illustrate the flip-and-align algorithm. We begin with a “cold start” scenario where all element channels and the environmental channel are unknown. Since all channels are unknown, we set the phase shifter of all LAIA elements to  $0^\circ$  and collect the first channel measurement  $h_{\text{comb}}^1$ . We flip the phase of the first element, just as Figure 5(a) shows and then collect the second channel measurement  $h_{\text{comb}}^2$ . According to Equation 3, we can now calculate the channel  $h_{p_1}$  of the first element and the superimposed channel  $h_{\text{env}} + h_{p_2} + h_{p_3}$ , where  $h_{p_2}$  and  $h_{p_3}$  are the channels of the second and third elements respectively. Next, we rotate the channel  $h_{p_1}$  to align with the superimposed channel, as Figure 5(b) shows. We then flip the channel of the second element, as Figure 5(c) shows, and collect the third channel measurement  $h_{\text{comb}}^3$ . Similarly, the channel  $h_{p_2}$  of

the second element and the superimposed channel  $h_{\text{env}} + h_{p_3}$  is calculated using three channel measurements  $h_{\text{comb}}^1, h_{\text{comb}}^2$ , and  $h_{\text{comb}}^3$ . All of the estimated channels, *i.e.*,  $h_{p_1}$  and  $h_{p_2}$ , are then aligned with the superimposed channel, as shown in Figure 5 (d). We repeat until all channels are calculated and aligned, as Figures 5(e), (f) show.

The flip-and-align control matrix  $\mathbf{Q}$  is thus given by:<sup>2</sup>

$$\mathbf{Q} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & 1 & 1 & 1 \\ e^{j\theta_{1,1}} & -1 & 1 & 1 \\ e^{j\theta_{1,2}} & e^{j\theta_{2,1}} & -1 & 1 \end{bmatrix} \quad (6)$$

The control matrix  $\mathbf{Q}$  in Eq. 6 can be extended to include an arbitrary number of LAIA elements.<sup>3</sup>

When an ongoing round of channel measurements finishes, LAIA begins a new round, following the same flip-and-align algorithm with the only difference being the initial phase of each LAIA element. In the new round each LAIA element begins configured with its previously-computed phase rotation, to leverage any correlation in the wireless channel between the two measurement rounds that will be present if both rounds complete at time intervals close to the channel coherence time.

### 3.2.2 Channel alignment

Once we have decomposed the wireless channel, the LAIA *alignment* algorithm calculates the most desirable phase settings for individual elements to optimize a particular performance metric, *i.e.*, we find the solution  $\Theta^*$  to the following optimization problem

$$\Theta^* = \underset{\Theta}{\text{arg max}} F(h_{\text{comb}}(\Theta)) \quad (7)$$

where  $\Theta = [\theta_1, \theta_2, \dots, \theta_M]$  is the phase settings of all  $M$  elements,  $h_{\text{comb}}(\Theta)$ <sup>4</sup> is the combined channel of applying  $\Theta$  to

<sup>2</sup> $\theta_{1,1}$  is the phase rotation calculated to align the first element with the superimposed channel in the third channel measurement, as shown in Figure 5(b);  $\theta_{1,2}$  and  $\theta_{2,1}$  are phase rotations calculated to align the first and second elements with the superimposed channel, as shown in Figure 5 (d).

<sup>3</sup>We prove that the control matrix  $\mathbf{Q}$  is full rank in the Appendix.

<sup>4</sup> $h_{\text{comb}}$  can be a scalar, a vector, or a matrix. We use the scalar notation here and explain the other scenarios later in the section.

LAIA, and  $F(h_{comb}(\Theta))$  is the objective function characterizing the performance metric.

**Example.** Say we want to maximize the Shannon capacity of a *narrowband single-antenna* link. The combined channel  $h_{comb}(\Theta)$  (a *scalar*, like  $h_{env}$ ) after applying  $\Theta$  is:

$$h_{comb}(\Theta) = e^{j\Theta} \cdot \mathbf{H}_p + h_{env} \quad (8)$$

where  $\mathbf{H}_p = [h_{p_1}, h_{p_2}, \dots, h_{p_M}]^T$  represents all the LAIA element channels, and the objective function  $F(h_{comb}(\Theta))$  (Shannon capacity in bits per second per Hertz) is

$$F(h_{comb}(\Theta)) = \log_2(1 + |h_{comb}(\Theta)|^2 \rho) \quad (9)$$

where  $\rho$  is the signal-to-noise ratio (SNR) on the channel. Shannon capacity is maximized at the highest SNR for the combined channel, which can be achieved by aligning the phases of all LAIA element channels with the environmental channel  $h_{env}$ , as shown in Figure 5, to maximize  $|h_{comb}|^2$ . The  $i^{th}$  element in  $\Theta$ ,  $\theta_i$ , is simply set to the phase difference between the channels  $h_{p_i}$  and  $h_{env}$ .

### 3.3 Multi-objective control

**Wideband single-antenna links.** Wireless links typically operate over a wide frequency band and therefore experience frequency selective fading. This means that even a single-antenna link is comprised of a set of distinct channels, corresponding to different frequency components (subcarriers). In other words, the environmental channel  $\mathbf{h}_{env}$  and the combined channel  $\mathbf{h}_{comb}$  become vectors, whose length is the number of subcarriers. We can still resort to Eq. 8, but now each element channel  $\mathbf{h}_{p_i}$  is also a vector. Since our phase shifters operate over the entire frequency band without regard for subcarriers, and so the same phase shift will be applied to all subcarriers. Clearly we cannot optimize for different channels simultaneously with one phase shift configuration. Therefore, while we can still define an objective function  $F$  over all subcarriers, for example, the total Shannon capacity (*i.e.*, summing the per-subcarrier Shannon capacity),  $F$  can no longer be maximized analytically.

**Multi-antenna (MIMO) links.** MIMO links, narrowband or wideband, are another form of "multi-channel" links. Therefore, following a similar approach used to handling wideband single-antenna links, we can derive the appropriate expressions for the channels (using Eq. 8) and the objective function, then again use the same optimization problem formulation (Eq. 7). For example, the situation for *narrowband MIMO* links (though not seen in practical wireless technologies) is analogous to that of *wideband single-antenna* links. The environmental channel  $\mathbf{h}_{env}$  and the combined channel  $\mathbf{h}_{comb}$  are now both *matrices*, whose dimensions are determined by the number of antennas at the wireless sender and receiver respectively. Eq. 8 is still valid, but each element channel  $\mathbf{h}_{p_i}$

is now also a matrix of the same dimension as that of  $\mathbf{h}_{env}$  and  $\mathbf{h}_{comb}$ . The MIMO channel capacity is an example for the objective function.

Extending this further to *wideband MIMO* links used in Wi-Fi or LTE, the environmental channel  $\mathbf{h}_{env}$ , the combined channel  $\mathbf{h}_{comb}$ , and each element channel  $\mathbf{h}_{p_i}$  are all matrices of the form  $N_{rx} \times N_{tx} \times N_{subcarriers}$ , where  $N_{rx}$ ,  $N_{tx}$ , and  $N_{subcarriers}$  are the numbers of the receiver antennas, the sender antennas, and the subcarriers respectively.

**Multiple links.** Only single-link (SISO or MIMO) scenarios have been considered so far. Next, we extend LAIA to multi-link scenarios and discuss two representative cases: (i) a single cell with multiple clients associated with the same access point (AP) and on the same frequency; and (ii) multiple concurrent links on different frequencies.

In the first case, while the AP only communicates with one client at a time, LAIA cannot predict which client will be active and cannot configure the element array accordingly before the transmission. Therefore, LAIA configures the element array to maximize the total capacity of all AP-client links and keep the configuration until any client joins or leaves the network. In the second case, LAIA must find a configuration of the element array that works for multiple links simultaneously. This can again be reduced to maximizing the total capacity. In either case, we use the total capacity expression as the objective function in our optimization problem (Eq. 7). In addition, we adjust the objective function to consider fairness or other performance goals, *i.e.*, we maximize the aggregate capacity of all the links with the constraint that no individual link will be harmed in terms of capacity.

#### 3.3.1 Iterative search

In above multi-channel cases, there is no analytic solution to the optimization problem in Eq. 7. Given a large array size and the number of possible per-element phase choices for LAIA, exhaustively searching for the optimal  $\Theta$  that maximizes the objective function is too complex computationally. For example, a 4-bit phase shifter, used in LAIA, provides  $2^4 = 16$  possible phase shifts per element and results in  $16^M$  possible phase combinations for an  $M$ -element LAIA array! We therefore identify heuristics to prune the search space.

**Basic version.** We adopt a multi-round iterative algorithm to prune the search space. In each round, we sequentially align all element channels with the environmental channel. Specifically, in round one, we first align the channel  $h_{p_1}$  with  $h_{env}$  by searching through all possible phase settings of the first element that maximizes  $F$ , fixing the phases of the rest of the array. Once we find the result,  $\theta'_1$ , we then align the element channel  $h_{p_2}$  with the intermediate combined channel  $e^{j\theta'_1} h_{p_1} + h_{env}$ , again fixing the phases elsewhere in the array. We repeat the process for all LAIA elements until we obtain a vector  $\Theta' = [\theta'_1, \theta'_2, \dots, \theta'_M]$ , when this round completes.

With  $\Theta'$  applied to the LAIA elements, the combined channel becomes  $h_{comb}(\Theta')$ : While  $F(h_{comb}(\Theta'))$  usually improves over  $F(h_{env})$ , the improvement is not guaranteed to be optimal. Therefore, multiple rounds of the alignment may be needed. These rounds proceed in much the same way as the first round, except that in subsequent rounds of alignment, when we align LAIA element  $k$ , we leave the phase of all the other elements  $\Theta^{(k)} = [\theta_1, \dots, \theta_{k-1}, \theta_{k+1}, \dots, \theta_M]$  constant so we align element  $k$  with the combined channel,  $h_{comb}(\Theta^{(k)})$ . We iterate until there is barely an additional improvement (*i.e.*, below a threshold) in  $F$  from a new round. The algorithm typically converges in two to three rounds in our experiments.

By tuning one element phase at a time, the iterative algorithm already reduces the search complexity dramatically from exponential to linear:  $16M$  phase settings are checked instead of  $16^M$  in each round. Strictly speaking this finds the local optimal result, but we find empirically that the local optimal is typically very close to the global optimal (Figure 11).

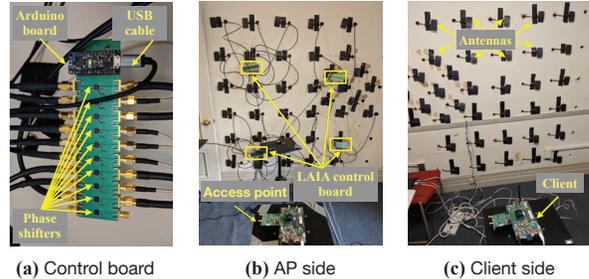
**Prioritizing “influential” elements.** Still, the execution time of our search algorithm may exceed the channel coherence time, especially when given a large LAIA array. Therefore, we further refine the basic iterative search as follows.

Recall that the number of *dominant multipath components* in a typical indoor environment is limited to a small number [10, 21]. This suggests that the number of *influential elements* in the whole LAIA array is also limited. Therefore, we can speed up the search and reap most of the benefit by prioritizing these influential elements. We assign a priority level, initialized to 0, to each LAIA element to indicate the order of search. In each round of iteration, the elements will be reconfigured to the optimal phases in the order of their priority levels. After an element is configured, its priority level is increased by its contribution to the overall channel improvement weighted by a random probability uniformly distributed between 0 and 1. The element with the highest contribution tends to get a large increment of its priority level and will likely be tuned again sooner in the next round.

The advantages of the prioritized search algorithm are two-fold. First, the priority level ensures the element that contributes most to the channel is configured first, within the channel coherence time, and thus adapting LAIA quickly to changing environmental channels, which helps to mitigate the effect of mobility. Second, the algorithm does not exclude less influential elements and can still converge to the result calculated by the basic iterative algorithm.

## 4 Implementation

In this section, we first introduce the LAIA element implementation (§4.1) and then describe the way to handle channel measurement errors (§4.2).



**Figure 6:** (a) A LAIA control board prototype, with ten phase shifters and one Arduino board. A real-world deployment of a 36-element LAIA array, one antenna of each LAIA element is on (b) the AP side of the wall and the other is on (c) the client side.

### 4.1 LAIA hardware

The hardware of LAIA includes two major parts: antenna and control board. The antenna we use is small panel antenna, with relatively narrow beam width (around 40 degree). Each antenna only costs about 1.75 USD. We build a prototype of LAIA control board using ten MACOM MAPS-010144 four-bit phase shifter [25] and an Arduino Adafruit Metro Mini MCU [3] on a four-layer printed circuit board (PCB), as shown in Figure 6(a). All the control boards are connected to the central controller (a laptop) via a USB 3.0 hub. The four-bit phase shifter can shift the phase of an incoming signal at a granularity of  $\pi/8$ .

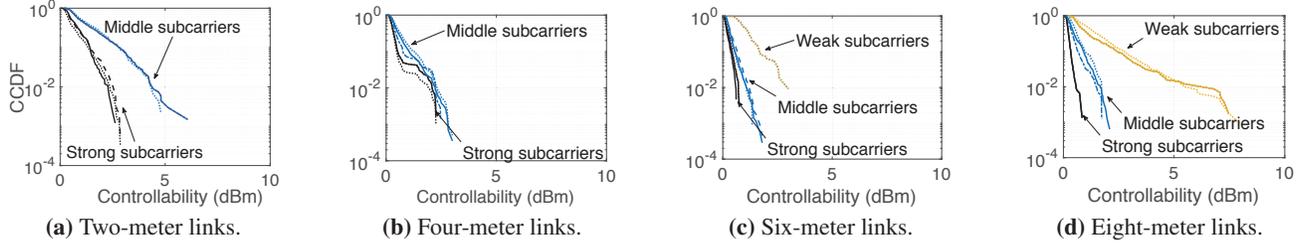
### 4.2 Channel measurement error

Our control algorithm relies on a very accurate channel measurement at the receiver end, which is challenging to achieve in practice. First, *carrier frequency offset* (CFO) describes the frequency difference between the two transceivers’ oscillators. The phase offset caused by CFO changes over time, causing a random phase offset between channel measurements from different packets. The channel decomposition algorithm relies on multiple packets to decompose the channel, thus suffering from the random phase offset. We describe the detail of our CFO estimation algorithm in Appendix.

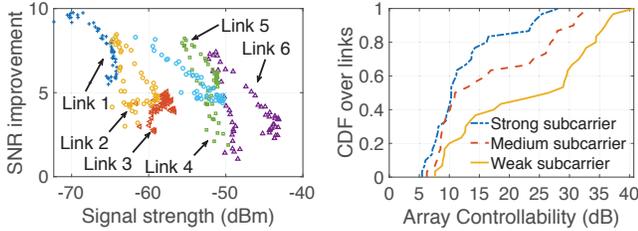
Second, *symbol timing offset* (STO) introduces a sub-sample delay in the time domain, equivalent to a phase slope in the frequency domain. As STO varies across different packets, it introduces a phase offset across packets and thus affects our channel decomposition algorithm. We leverage fractional interpolation [22, 40] to detect and remove the phase slope in channel readings, thus compensating for STO.

## 5 Evaluation

In this section, we evaluate LAIA’s performance, starting with microbenchmarks (§5.1) to characterize basic wireless



**Figure 7:** Distribution (across elements and subcarriers) of LAIA element controllability on strong ( $> -53$  dBm), middle ( $> -65$  dBm and not strong), and weak ( $< -65$  dBm) subcarriers of three links, for various link lengths noted. LAIA controls weaker subcarriers more easily.



**Figure 8:** LAIA SNR improvement versus signal strength for six links (one point per subcarrier). **Figure 9:** Controllability distribution of the 36-element LAIA array, across 30 links.

channel controllability, accuracy of channel decomposition and the gap between the LAIA control algorithm and the optimal method. We then conduct field studies to quantify the gain of channel capacity and TCP throughput of LAIA for both SISO (§5.2) and MIMO (§5.3) links. An evaluation of multiple links using both the same and adjacent channels (§5.4) concludes our evaluation.

**Experiment setup.** We deploy 36 passive LAIA elements on an interior wall of a house, as shown in Figures 6(b) and (c). Each element is a passive relay with antennas attached to either side of the wall. The drywall provides a 1.5 dB attenuation for signal at 2.4GHz, which matches RADAR’s *Wall Attenuation Factor* [5] and the results of a construction material signal attenuation test [31, 38]. We use WARP v3 software-defined radios [37] as Wi-Fi senders and receivers on the 2.4 GHz Wi-Fi band with 20 MHz channel bandwidth. The sender and receiver are deployed in two different rooms with a wall in between blocking the line-of-sight (LoS) path.

## 5.1 Microbenchmarks

We conduct microbenchmarks to better understand the performance of each part of LAIA’s design. The experiments to evaluate the controllability of LAIA emulate link optimizations to remove (*i.e.*, “fill”) null subcarriers.

**LAIA element controllability.** We define the difference between the maximum and the minimum signal strength that one LAIA element can induce at a particular subcarrier as the *element controllability*. We vary the link distance from 2 m to 8 m, and measure the controllability of each individual

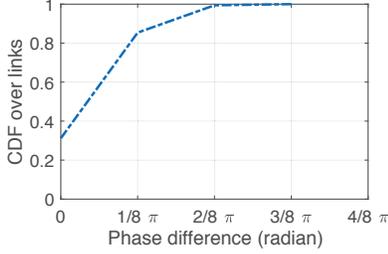
LAIA element, over three different links, for each link length. Due to frequency-selective fading, the environmental channel differs across subcarriers, which leads to differences in controllability. To investigate this effect, we separate subcarriers into *strong* ( $[-53, +\infty]$  dBm), *middle* ( $[-65, -53]$  dBm), and *weak* ( $[-\infty, -65]$  dBm) strength ranges and test the controllability of each subcarrier. Figure 7 shows the distribution of controllability across different LAIA elements, with one curve per physical link and subcarrier strength combination. We see that two-meter links have no strong subcarriers, while six meter and eight-meter links have the most weakest subcarriers. We also see consistent controllability across link length after binning the data into the above subcarrier strength ranges. Overall the clear trend is for greater controllability for weak subcarriers than for medium and strong ones.

Considering absolute controllability, elements at different locations achieve from *ca.* 0 dB to 7.5 dB apiece, since their  $40^\circ$  directional antennas capture differing amounts of RF signal from different transmitter locations. As link distance increases, signal power received from each element decreases, resulting in generally higher controllability.

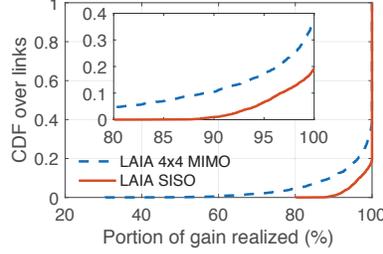
**LAIA array controllability.** Given the modest amount of controllability a single LAIA element provides, we investigate 36 LAIA elements working together. We configure the array to maximize link Shannon capacity. The SNR changes on each subcarrier of six representative links are shown on the y-axis of Figure 8. This scatter plot plots the original signal strength of each subcarrier on the x-axis. We see that LAIA improves the SNR for all subcarriers and generally the weaker the subcarrier, the larger the improvement.

In a separate larger-scale experiment involving 30 links, we bin the subcarriers of each individual link into the top, middle, and lowest thirds by SNR. The controllability distribution is shown for each bin (respectively, *strong*, *medium*, and *weak* in Figure 9). From the figure, we can see that LAIA provides up to 42 dB controllability. Median controllability for the strongest, medium, and weakest bins is about 10.4 dB, 11.8 dB, and 27.3 dB, respectively. In summary, multiple elements provide substantial controllability, especially when transceivers experience poor channel conditions.

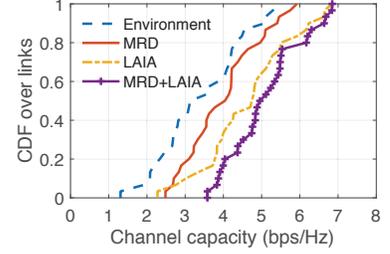
**Accuracy of channel decomposition.** We next evaluate the accuracy of the channel decomposition algorithm. In these experiments, we randomly select a 2.4 GHz Wi-Fi channel



**Figure 10:** Predicted and measured phase difference for constructive superposition.



**Figure 11:** Percent improvement of LAIA compared with exhaustive search.



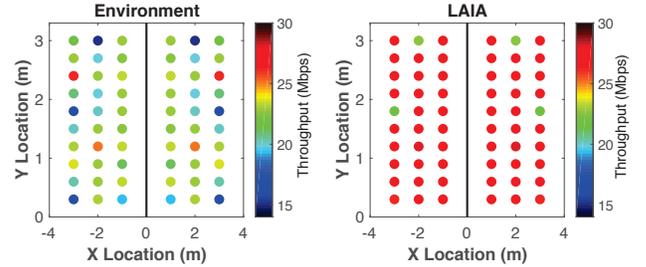
**Figure 12:** Channel capacity achieved by MRD [26], LAIA, and MRD+LAIA.

and estimate the CSI of an individual LAIA element channel and the environmental channel using the channel decomposition algorithm presented in §3.2.1. With the decomposed channel, we compute the phase setting for this LAIA element that can maximize the reference SNR. At the same time, we apply different phases to this LAIA element to measure the difference between the computed phase setting and the real phase setting that maximizes SNR. Since the accuracy of the computed phase setting is only determined by the accuracy of channel decomposition algorithm, we can use the deviation between the computed and real phase setting as a metric to measure channel decomposition accuracy.

We repeat the phase decomposition test 2,000 times with different LAIA elements and different transmitter/receiver locations and plot the absolute phase difference between the computed and real phase settings in Figure 10. We can see that 85.4% of the LAIA elements can be configured within an error of one phase shift step ( $\pm 1/8\pi$ ) and 99.4% of the LAIA elements can be configured within an error of two phase shift steps ( $\pm 1/4\pi$ ). Based on our experience, we attribute most of the error to the small phase oscillation of phase shifters.

**LAIA control algorithm—optimal gap.** We evaluate the gap between the LAIA control algorithm and optimal exhaustive search. We conduct extensive trace-based simulations on both SISO OFDM links and 4x4 MIMO OFDM links. We collect 5,000 traces for both cases, one trace for each link. Each trace includes the environmental channel and each LAIA element’s channel. In order to make the comparison feasible within reasonable time, we consider a subset of the search space: five LAIA elements, each having eight phases. Consequently, exhaustive search can find the answer in  $8^5 = 32,768$  search combinations for each trace. We compute the Shannon capacity of the result of both the LAIA control algorithm and exhaustive search. Additionally, we use percent Shannon capacity increase  $\frac{C(\text{LAIA}) - C(\text{Env})}{C(\text{Opt}) - C(\text{Env})}$  as a metric to evaluate how close the LAIA control algorithm approaches the optimal.

We observe in Figure 11 that the LAIA control algorithm provides a close approximation of the optimal. For SISO OFDM, LAIA finds the exact optimal in 80.76% of the cases. For 4x4 MIMO OFDM, LAIA finds the exact optimal in 61.6% of the cases. On average, LAIA can achieve 99.23%



**Figure 13:** TCP throughput at different locations without (left) and with (right) LAIA. The location of the transmitter and receiver is symmetric to each other.

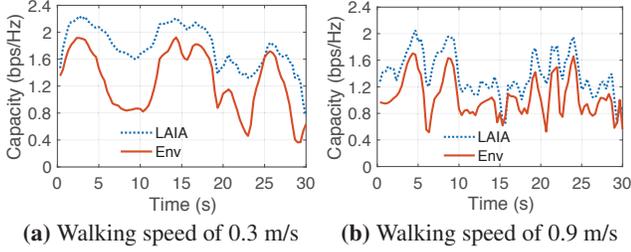
and 96.89% capacity increase compared to the optimal solution for SISO and 4x4 MIMO cases, respectively.

**Element density.** We evaluate the impact of LAIA element density in the array. Our 36-element array is deployed on a  $2 \times 1.8$  m wall. We build a 9-element array by selecting nine adjacent elements from a  $0.7 \times 0.5$  m wall area and measure the controllability of each such array. We then build another 9-element array by randomly selecting nine elements from all 36 elements. We repeat this random selection process 1,000 times. The controllability (averaged over subcarriers) achieved using nine adjacent and nine randomly selected elements (averaged over 1,000 experiments), is 2.57 dB and 2.50 dB, respectively. We see that the density of the LAIA elements does not significantly affect achieved controllability.

## 5.2 SISO performance

In this section, we evaluate the end-to-end performance of LAIA for single-antenna links.

**Channel capacity.** We first measure the channel capacity achieved by LAIA in SISO communication systems and compare it with that achieved by MRD [26], a link diversity scheme that deploys two APs and always connects the client to the AP with better channel. In these experiments, we place two receivers in one room and a transmitter in another room. The LAIA elements are deployed on the wall between these two rooms. We compare LAIA with MRD in the following



**Figure 14:** Snapshot of the channel capacity when walking with speed of (a) 0.3 m/s, and (b) 0.9 m/s.

way. We first connect the transmitter to the first receiver and measure the channel capacity achieved by LAIA. We next pick the better link between two transmitter-receiver links and measure the capacity achieved by the MRD algorithm. At last, we then pick the better link and measure the capacity achieved by running LAIA. As a baseline, we also measure the capacity of the environmental channel.

We repeat this experiment 30 times at different Tx/Rx locations and plot the CDF of the capacity of these four methods in Figure 12. We can see that the pure environmental channel achieves the lowest median capacity (3.13 bps/Hz), followed by the MRD algorithm (4.04 bps/Hz). In contrast, we can see that LAIA achieves a capacity of 4.74 bps/Hz, which is 51.4% higher than the pure environmental channel. This capacity further increases to 4.96 bps/Hz as we combine LAIA and the MDR algorithm.

**TCP throughput of a single user.** We further evaluate the TCP throughput achieved by LAIA using the WARP 802.11 reference design [36]. In these experiments, we set up a transmitter-receiver link across a wall, and measure the TCP throughput on different locations. The transmitter is placed on the location symmetric to the receiver. During the transmission, LAIA's uses the channel alignment algorithm described in Section 3 to configure the phase of every element. The left figure of Figure 13 shows the TCP throughput of the environmental channel where we terminate all LAIA elements deployed on the wall. We can see that the TCP throughput varies significantly across locations. In contrast, from the right figure we can see that LAIA successfully improves the TCP throughput to above 26 Mbps for most locations. The average TCP throughput achieved by LAIA is 26.9 Mbps, a  $1.24\times$  improvement over the environmental channel (21.7 Mbps). Additionally, LAIA can achieve a maximum of  $1.56\times$  TCP throughput improvement.

**Mobile case.** The previous experiments demonstrate that LAIA can successfully improve the channel capacity and TCP throughput for static wireless links. In this section, we evaluate the performance of LAIA in a mobile environment. We place the receiver (AP) at a fixed location and move the transmitter at two constant speeds: slow (0.3 m/s) and normal (0.9 m/s). At the same time, the LAIA controller configures

the LAIA array in real time based on the CSI information from the AP. We then measure the SNR and estimate the Shannon capacity on the basis of that measurement during the transmitter's movement. Figure 14 shows a snapshot of the real-time channel capacity. LAIA achieves consistently higher channel capacity than the pure environmental channel at both the slow and normal movement speeds. Specifically, LAIA successfully improves the average channel capacity from 1.28 bps/Hz to 1.79 bps/Hz for slow movement speed, and from 1.04 bps/Hz to 1.39 bps/Hz for normal movement speed, respectively, demonstrating that LAIA effectively improves the channel capacity in mobile environments. We notice that LAIA works slightly better when the moving speed is slower. This is because the slower movement gives LAIA more time to react to the channel change. We will further discuss how to scale LAIA to high speed in future work.

### 5.3 MIMO links

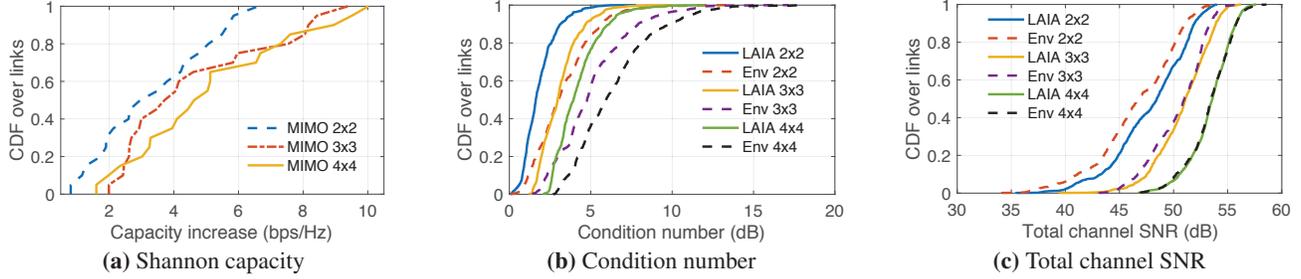
We next evaluate the capacity improvement achieved by LAIA for MIMO communication systems. We set up a multi-antenna transmitter-receiver system in our testbed and measure the channel capacity improvement at 20 different locations. Figure 15(a) shows the CDF of the channel capacity improvement LAIA achieves on  $2\times 2$ ,  $3\times 3$  and  $4\times 4$  MIMO links. We see that the improvement increases with the number of antennas used in the MIMO system. Specifically, the median capacity improvement is 2.96 bps/Hz, 3.71 bps/Hz and 4.6 bps/Hz for  $2\times 2$ ,  $3\times 3$ ,  $4\times 4$  MIMO, respectively, which corresponds to 18.95%, 15.68% and 12.23% capacity improvement compared with the baseline.

Since both a received power increase and a better MIMO conditioning can lead to channel capacity improvement in MIMO systems, we plot the condition number and the total channel SNR (summation of SNR for all LAIA elements) in Figure 15(b) and Figure 15(c), respectively, to understand the root cause of the capacity improvement in our experiments. From Figure 15(b) we observe a significant condition number gap between the LAIA channel and the environmental channel, indicating that the channel conditioning makes a substantial contribution to the channel capacity gain. In contrast, Figure 15(c) shows that the difference of channel SNR between the LAIA channel and the environmental channel is small. Hence we conclude that most of the capacity improvement comes from a channel conditioning improvement.

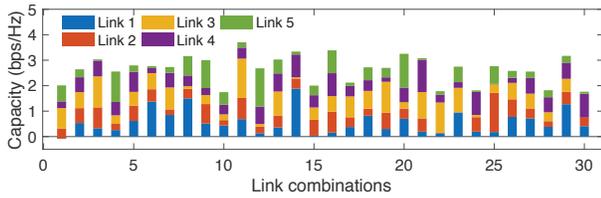
### 5.4 Multiple links

We also evaluate LAIA's performance in cases where multiple links operate in close proximity.

**Multiple users.** We first evaluate the performance in a Wi-Fi network consisting of five single-antenna clients and one single-antenna AP. Only one client can communicate with the AP at any given time instance. Since the LAIA controller



**Figure 15:** LAIA improvement on MIMO links (a) Shannon capacity increase achieved by LAIA, for  $2 \times 2$ ,  $3 \times 3$  and  $4 \times 4$  MIMO communications; (b) Condition number changed by LAIA, for  $2 \times 2$ ,  $3 \times 3$  and  $4 \times 4$  MIMO communications; and (c) Total channel SNR.



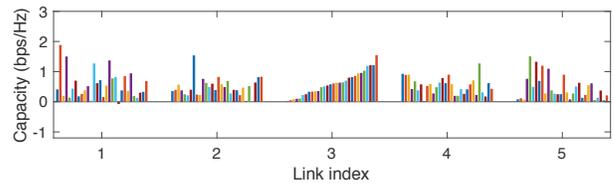
**Figure 16:** Shannon capacity improvement achieved by LAIA over five wireless links in 30 groups of experiments.

does not know which link is active, it configures the antenna array to maximize the total channel capacity of the five links, keeping the configuration until one user disassociates with the AP. We move the AP and all five clients to 30 different locations. At each location, we measure the CSI of all five links and calculate Shannon capacity.

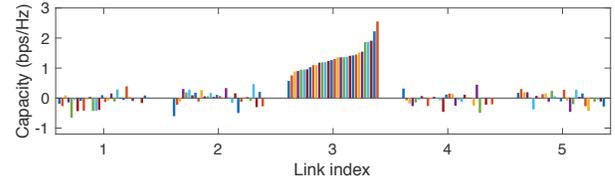
Figure 16 shows the Shannon capacity improvement (with respect to the environmental channel) LAIA achieves. We observe that LAIA achieves a 3.43 bps/Hz capacity improvement on average over 30 locations. The maximum and minimum capacity improvement is 5.54 bps/Hz and 2.10 bps/Hz, respectively. We also observe diverse capacity improvements among different links, *e.g.*, the capacity of Link 1 in Group 1 decreases, while Link 3 in Group 22 achieves a 72.9% capacity improvement. This is because our current control algorithm aims at maximizing the total capacity of all five links instead of optimizing any individual link.

We further sort the results of these 30 groups of experiments in ascending order of the capacity improvement over Link 3. In Figure 17(a), LAIA aims at maximizing the total capacity of all five links, while in Figure 17(b), LAIA aims at maximizing the capacity of Link 3 only. LAIA achieves much higher capacity improvement for a specific link when we target at maximizing that link alone. However, the capacity improvement of all the remaining links decreases significantly. Therefore, the control algorithm of LAIA should consider the fairness among all clients when serving them.

**Concurrent transmissions.** We next evaluate the performance of LAIA in the presence of concurrent wireless SISO links over two non-overlapping channels (channels 1 and 11 in the 2.4 GHz Wi-Fi band). LAIA’s controller optimizes the



(a) Capacity increase of five links (maximizing all links)



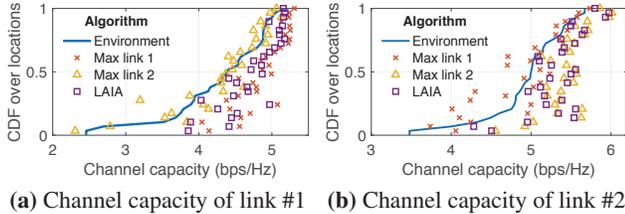
(b) Capacity increase of five links (maximizing link 3)

**Figure 17:** The sorted capacity improvement when LAIA aims at (a) maximizing the total capacity of five links; and (b) maximizing the capacity of link three.

channel capacity of these two SISO links with the fairness constraint (denoted *LAIA*). We then measure the capacity of the environmental channel as the baseline (denoted *environment*). For comparison purposes, we also measure the channel capacity when LAIA aims to maximize the channel capacity of link one (*Max link 1*) and link two (*Max link 2*). We conduct the above experiments in 30 random locations and plot the CDF of the channel capacity in Figures 18(a) and (b). We observe that LAIA can improve the aggregate channel capacity of these two links without reducing the capacity of any link. These experimental results demonstrate the feasibility of improving the capacity of two non-overlapping links simultaneously with LAIA. When LAIA aims to maximize the channel capacity of a single link, we can see that the achievable capacity improvement over this link increases, but sometimes the capacity of the other link decreases, which is unfair to the user of the other link.

## 6 Related Work

**Directional antennas and static Wi-Fi signal shapers.** Use of directional antennas is perhaps one of the most straightfor-



**Figure 18:** Channel capacity across locations when two wireless links transmit simultaneously on non-overlapping channels.

ward mechanisms to shape the wireless signals. Multiple of these antennas can be placed strategically to avoid interference between networks [24]) or generate desirable network coverage [30]. However, the granularity of the control is usually coarse, at the mercy of the beam specification and fluctuations in the environment. In a similar spirit, there have been several attempts at making the environment more amenable to wireless communications, for example, by adding 3D-printed static reflectors around the sender or receiver to shape outgoing or incoming signals [8, 41] or deploying static reflectors away from the AP [16]. However, these lack reconfigurability and cannot adapt to changing environment conditions. In contrast, LAIA aims to adapt to the environment in real time.

**Millimeter-wave reflectors.** Recent work uses reflectors, static mirrors [44], or programmable phased-array reflectors [1, 2] to generate alternate paths, which can circumvent obstacles blocking the direct path. While this is promising in the context of millimeter wave, the problem is qualitatively different for lower frequencies. For communications below 10 GHz, the number of reflectors, diffractors, and absorbers in the environment often creates a dense, high-dimensional channel, unlike the sparse matrix for 10 GHz and above. Optimizing this dense matrix is a qualitatively different problem than the one we undertake.

**Wi-Fi extenders.** Commercial Wi-Fi extenders are trending recently [11, 13, 32, 33], typically decoding packets from AP and forwarding them to clients located in longer ranges. This is again coarse-grained, however, and simply shifts the same coverage problem to a region further away from the transmitter. In contrast, LAIA controls the phase offset on different signal propagation paths to improve the SNR of the received signals and can be customized to practically any location within the antenna range. FastForward [6] uses a relay node to improve the communication channel. The fundamental difference between FastForward and LAIA is FastForward’s active versus LAIA’s passive design.

**Backscatter systems.** Much recent work [19, 20, 23] harnesses ambient signals in the environment as the power source for communication and computation. Another line of research [17, 18] presents active-passive hybrid designs (*i.e.*, traditional plus backscatter) for low-power radios as well as addressing power asymmetry between end-user devices.

LAIA is diametrically different since it alters communication signals instead of reusing them for another purpose. LAIA instead reconfigures signal propagation.

**Massive MIMO.** The advent of massive MIMO systems has ushered in a rapid growth in next generation wireless networks [14, 15, 28, 29, 42]. Massive MIMO leverages a large number of antennas to focus energy into ever smaller beams to serve multiple users simultaneously. These systems customize transmission based on the channels between the base station and the end users. In contrast, LAIA reconfigures the propagation paths of the wireless signals and can work synergistically with massive MIMO systems.

## 7 Discussion

Our current design merely scratches the surface of the design space. We briefly discuss limitations and future work.

**Limitations.** LAIA’s channel decomposition algorithm requires multiple packet exchanges to fully decompose wireless channels and hence takes time that scales with the number of elements. The channel decomposition requires highly accurate CSI, which is not provided by current commodity Wi-Fi devices, but may be in the future. Therefore, our current system only works on WARP software defined radio.

**Future work.** Many open questions remain and there are exciting avenues to explore. For example, what is the minimal subset of LAIA elements to control for competitive results? Further questions abound in the design of individual elements themselves: investigating the design spectrum between passive and active elements is ongoing work.

## 8 Conclusion

We have taken a first step towards programming the radio environment, a qualitatively different approach to the conventional strategy of optimizing communication endpoints. Our prototype implementation reconfigures the radio environment in real time, with an extensive evaluation demonstrating communications throughput enhancements that complement many other state of the art methods.

## Acknowledgements

We thank the PAWS group, the reviewers and our shepherd, Xinyu Zhang for their insightful comments. This material is based upon work supported by the National Science Foundation under Grant Nos. CNS-1617161, CNS-1763212, and CNS-1763309. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

## References

- [1] ABARI, O., BHARADIA, D., DUFFIELD, A., AND KATABI, D. Cutting the Cord in Virtual Reality. In *HotNets* (2016).
- [2] ABARI, O., BHARADIA, D., DUFFIELD, A., AND KATABI, D. Enabling high-quality untethered virtual reality. In *NSDI* (2017).
- [3] Arduino Adafruit Metro Mini. [Website](#).
- [4] ARYAFAR, E., ANAND, N., SALONIDIS, T., AND KNIGHTLY, E. Design and experimental evaluation of multi-user beamforming in wireless LANs. In *MobiCom* (2010).
- [5] BAHL, P., AND PADMANABHAN, V. N. RADAR: an in-building rf-based user location and tracking system. In *IEEE INFOCOM* (2000).
- [6] BHARADIA, D., AND KATTI, S. Fastforward: Fast and constructive full duplex relays. *SIGCOMM* (2015).
- [7] BISWAS, S., AND MORRIS, R. ExOR: Opportunistic Multi-hop Routing for Wireless Networks. In *SIGCOMM* (2005).
- [8] CHAN, J., ZHENG, C., AND ZHOU, X. 3D Printing Your Wireless Coverage. In *HotWireless Workshop* (2015).
- [9] Chinese remainder theorem. [https://en.wikipedia.org/wiki/Chinese\\_remainder\\_theorem](https://en.wikipedia.org/wiki/Chinese_remainder_theorem).
- [10] CZINK, N., HERDIN, M., ÖZCELİK, H., AND BONEK, E. Number of multipath clusters in indoor MIMO propagation environments. *Electronics letters* 40, 23 (2004), 1498–1499.
- [11] NETGEAR N300 Wi-Fi Range Extender. <https://www.netgear.com/home/products/networking/wifi-range-extenders/WN3000RP.aspx>.
- [12] GALLAGER, R. G. *Low-Density Parity-Check Codes*. PhD thesis, MIT, 1963.
- [13] Google Wi-Fi. [https://store.google.com/us/product/google\\_wifi](https://store.google.com/us/product/google_wifi).
- [14] HAMED, E., RAHUL, H., ABDELGHANY, M. A., AND KATABI, D. Real-time Distributed MIMO Systems. In *SIGCOMM* (2016).
- [15] HAMED, E., RAHUL, H., AND PARTOV, B. Chorus: truly distributed distributed-mimo. In *SIGCOMM* (2018).
- [16] HAN, S., AND SHIN, K. Enhancing wireless performance using reflectors. In *INFOCOM* (2017).
- [17] HU, P., ZHANG, P., AND GANESAN, D. Laissez-Faire: Fully asymmetric backscatter communication. In *SIGCOMM* (2015).
- [18] HU, P., ZHANG, P., ROSTAMI, M., AND GANESAN, D. Braidio: An integrated active-passive radio for mobile devices with asymmetric energy budgets. In *SIGCOMM* (2016).
- [19] IYER, V., TALLA, V., KELLOGG, B., GOLLAKOTA, S., AND SMITH, J. Inter-technology backscatter: Towards internet connectivity for implanted devices. In *SIGCOMM* (2016).
- [20] KELLOGG, B., PARKS, A., GOLLAKOTA, S., SMITH, J. R., AND WETHERALL, D. Wi-fi backscatter: Internet connectivity for rf-powered devices. In *SIGCOMM* (2014).
- [21] KOTARU, M., JOSHI, K., BHARADIA, D., AND KATTI, S. SpotFi: Decimeter level localization using WiFi. In *SIGCOMM* (2015).
- [22] LAAKSO, T. I., VALIMAKI, V., KARJALAINEN, M., AND LAINE, U. K. Splitting the unit delay [FIR/all pass filters design]. *IEEE Signal Processing Magazine* 13, 1 (1996), 30–60.
- [23] LIU, V., PARKS, A., TALLA, V., GOLLAKOTA, S., WETHERALL, D., AND SMITH, J. R. Ambient backscatter: wireless communication out of thin air. In *SIGCOMM* (2013).
- [24] LIU, X., SHETH, A., KAMINSKY, M., PAPAGIANNAKI, K., SESHAN, S., AND STEENKISTE, P. DIRC: Increasing indoor wireless capacity using directional antennas. In *SIGCOMM* (2009).
- [25] MACOM Maps-010144 four-bits phase shifter. <http://cdn.macom.com/datasheets/maps-010144.pdf>.
- [26] MIU, A., BALAKRISHNAN, H., AND KOKSAL, C. E. Improving loss resilience with multi-radio diversity in wireless networks. In *MobiCom* (2005).
- [27] PERRY, J., IANNUCCI, P., FLEMING, K., BALAKRISHNAN, H., AND SHAH, D. Spinal codes. In *SIGCOMM* (2012).
- [28] RAHUL, H., KUMAR, S., AND KATABI, D. JMB: Scaling Wireless Capacity with User Demands. In *SIGCOMM* (2012).
- [29] SHEPARD, C., YU, H., ANAND, N., LI, L., MARZETTA, T., YANG, R., AND ZHONG, L. Argos: Practical many-antenna base stations. In *MobiCom* (2012).
- [30] SHETH, A., SESHAN, S., AND WETHERALL, D. Geo-fencing: Confining wi-fi coverage to physical boundaries. In *PerCom* (2009).
- [31] STONE, W. C. Electromagnetic signal attenuation in construction materials. Tech. rep., National Institute of Standards and Technology, 1997.
- [32] Linksys - AC750 Boost Range Extender. <http://www.linksys.com/us/p/P-RE6300/>.

- [33] TP-Link RE450 1750 Wi-Fi Range Extender. [https://www.tp-link.com/us/products/details/cat-5508\\_RE450.html](https://www.tp-link.com/us/products/details/cat-5508_RE450.html).
- [34] TSE, D., AND VISWANATH, P. *Fundamentals of Wireless Communication*. Cambridge University Press, 2005.
- [35] VAN NEE, R. D., AND PRASAD, R. *OFDM for wireless multimedia communications*. Artech house, 2000.
- [36] Warp 802.11 throughput benchmarks. <https://warpproject.org/trac/wiki/802.11/Benchmarks/Throughput>.
- [37] Rice Univ. WARP platform (v. 3). <https://mangocomm.com/products/kits/warp-v3-kit>.
- [38] WILSON, R. Propagation losses through common building materials 2.4 ghz vs 5 ghz. *Magis Networks Inc.: San Diego, CA, USA* (2002).
- [39] WOO, G. R., KHERADPOUR, P., SHEN, D., AND KATABI, D. Beyond the bits: Cooperative packet recovery using physical layer information. In *MobiCom* (2007).
- [40] XIONG, J., SUNDARESAN, K., AND JAMIESON, K. ToneTrack: Leveraging frequency-agile radios for time-based indoor wireless localization. In *MobiCom* (2015).
- [41] XIONG, X., CHAN, J., YU, E., KUMARI, N., SANI, A. A., ZHENG, C., AND ZHOU, X. Customizing indoor wireless coverage via 3d-fabricated reflectors. In *BuildSys* (2017).
- [42] YANG, Q., LI, X., YAO, H., FANG, J., TAN, K., HU, W., ZHANG, J., AND ZHANG, Y. BigStation: Enabling Scalable Real-time Signal Processing in Large MU-MIMO systems. In *SIGCOMM* (2013).
- [43] ZHANG, Y., YANG, C. J., HUDSON, S. E., HARRISON, C., AND SAMPLE, A. Wall++: Room-scale interactive and context-aware sensing. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (2018), ACM, p. 273.
- [44] ZHOU, X., ZHANG, Z., ZHU, Y., LI, Y., KUMAR, S., VAHDAT, A., ZHAO, B., AND ZHENG, H. Mirror Mirror on the Ceiling: Flexible Wireless Links for Data Centers. In *SIGCOMM* (2012).

## A Appendix

### A.1 Probing Matrix Construction

**Theorem 1.**  $\mathbf{Q}$  in the form of

$$\mathbf{Q} = \begin{bmatrix} 1 & 1 & \dots & 1 & 1 \\ -1 & 1 & \dots & 1 & 1 \\ e^{j\theta_{2,0}} & -1 & \dots & 1 & 1 \\ \vdots & \vdots & \dots & \vdots & \vdots \\ e^{j\theta_{M,0}} & e^{j\theta_{M,1}} & \dots & -1 & 1 \end{bmatrix} \quad (10)$$

is a full rank matrix regardless of the value of  $\theta_{i,j}$

*Proof.* To prove matrix  $\mathbf{Q}$  is full rank, we try to prove the value of the determinant of  $\mathbf{Q}$  is non-zero. We first consider a transfer matrix  $\mathbf{Q}'$ ,

$$\mathbf{Q}' = \begin{bmatrix} -1 & 1 & \dots & 1 & 1 \\ e^{j\theta_{2,0}} & -1 & \dots & 1 & 1 \\ \vdots & \vdots & \dots & \vdots & \vdots \\ e^{j\theta_{M,0}} & e^{j\theta_{M,1}} & \dots & -1 & 1 \\ 1 & 1 & \dots & 1 & 1 \end{bmatrix} \quad (11)$$

which swaps the first row of  $\mathbf{Q}$  with the last row. Notice that the absolute value of the determinant of  $\mathbf{Q}$  and  $\mathbf{Q}'$  are the same.  $|\det \mathbf{Q}| = |\det \mathbf{Q}'|$ . For  $\det \mathbf{Q}'$ , We can subtract the last row from the first  $n-1$  rows, and we can get

$$\begin{aligned} \det \mathbf{Q}' &= \det \left( \begin{bmatrix} -2 & 0 & \dots & 0 & 0 \\ e^{j\theta_{2,0}} - 1 & -2 & \dots & 0 & 0 \\ \vdots & \vdots & \dots & \vdots & \vdots \\ e^{j\theta_{M,0}} - 1 & e^{j\theta_{M,1}} - 1 & \dots & -2 & 0 \\ 1 & 1 & \dots & 1 & 1 \end{bmatrix} \right) \\ &= (-2)^{n-1}. \end{aligned} \quad (12)$$

So the absolute value of the determinant of  $\mathbf{Q}$  is not zero. Since  $|\det \mathbf{Q}| = |\det \mathbf{Q}'| \neq 0$ ,  $\mathbf{Q}$  is a full rank matrix.  $\square$

### A.2 CFO Estimation

Figure 19 demonstrates why CFO can cause a random phase offset in CSI estimation. In the figure, the x-axis denotes time and y-axis denotes the phase difference between the carrier signals of sender and receiver. The slope of the green line is CFO, denoted by  $\Delta f$ . The blue boxes denote packets. In the figure, the time gap between the first packet and the second packet is  $\Delta t_1$ . The time gap contributes to a phase difference of  $\Delta \phi_1$  between the CSI of two packets. In addition, the phase difference will fold back modulo  $2\pi$ . If  $\phi_0$  is phase difference between upconverter signal and downconverter signal at time 0, we can write the phase error of the  $i$ th packet as  $\phi_i = (\phi_0 + \Delta f \cdot t_i) \bmod 2\pi$ . Our goal is to estimate  $\Delta f$  accurately. To achieve this, we can change the formula to the

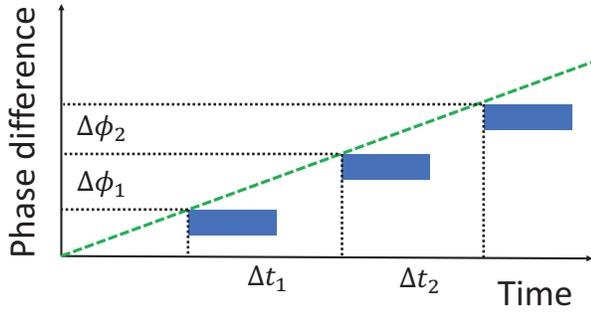


Figure 19: A example to demonstrate how LAIA estimate CFO.

form of the Chinese remainder theorem [9]  $\phi_0 = (\phi_i + 2\pi \cdot n_k) \bmod t_i$ , where  $n_k$  is a integer: we compute an accurate CFO and compensate for the random phase offset accordingly.

This method has a more accurate CFO estimation than the Schmidl-Cox method because it uses a much larger time window to estimate CFO. The time window that Schmidl-Cox uses to estimate CFO is typically a symbol time ( $4\mu s$ ), while the time window that LAIA uses is typical several packet times (around 4 ms). Since phase errors due to CFO accumulate with time, a longer observation time window can provide better CFO estimation.