RICE UNIVERSITY

Argos: Practical Many-Antenna MU-MIMO Systems

by

Clayton W. Shepard

A Thesis Submitted
in Partial Fulfillment of the
Requirements for the Degree
Doctorate of Philosophy

Approved, Thesis Committee:

Lin Zhong, Chair
Professor of Electrical and Computer Engineering and Computer Science

Ashutosh Sabharwal
Professor of Electrical and Computer Engineering

E. Knightly

Edward Knightly
Professor of Electrical and Computer Engineering

Eugene Ng
Professor of Computer Science and Electrical and Computer Engineering

Victor Bahl
Distinguished Scientist
Microsoft Research

Houston, Texas
September, 2017

ABSTRACT

Argos: Practical Many-Antenna
MU-MIMO Systems

by

Clayton W. Shepard

Many-antenna MU-MIMO, or "massive" MIMO at large scale, is a key candidate technology for next-generation wireless systems. However, from a practical design perspective scaling up MU-MIMO presents a number of unique challenges and opportunities.

To efficiently utilize computational, power, and channel resources requires a complete redesign of many aspects of traditional MIMO systems, particularly the base station architecture and control channel. Furthermore, mobility fundamentally limits the achievable rate of many-antenna MU-MIMO systems, thus to efficiently realize practical many-antenna MU-MIMO systems requires performance modeling and testing in real-world environments.

This thesis presents a novel scalable many-antenna base station architecture and control channel design, which are implemented and tested on three generations of large-scale custom built hardware platforms. We derive a theoretical model of many-antenna MU-MIMO system performance in real-world environments, accounting for hardware capabilities and channel estimation overhead. Leveraging these many-antenna MU-MIMO platforms, we conducted a comprehensive channel measurement campaign, spanning the UHF, 2.4 GHz, and 5 GHz bands, with varying degrees of mobility. Based on these measurements, we devise a mobility-aware MU-MIMO system, which is able to optimize performance across various real-world mobile environments. Combined, these innovations enable many-antenna MU-MIMO systems to be efficiently implemented in real-world environments with mobility.

Dedicated to my grandparents, the original Barbie and Ken.

Acknowledgments

First and foremost I would like to thank my advisor, Dr. Lin Zhong, for all of his help, support, patience, and advice that made this project, and my graduate career, possible. I am also very grateful for the continued help from my colleagues, especially Ryan Guerra, Evan Everett, Abeer Javed, Jian Ding, and Narendra Anand, who have directly contributed to the development and direction of this project.

I would also like to thank Dr. Erran Li and the other researchers at Bell Labs for introducing me to this project. Dr. Ashutosh Sabharwal and Dr. Thomas Marzetta have been instrumental to my understanding of the relevant theoretical background; I greatly enjoyed our incredibly enlightening discussions. I appreciate the opportunity to work with Dr. Victor Bahl and the other researchers and Microsoft Research; the experience I gained in deploying multi-cell MU-MIMO base stations and collecting channel measurements was invaluable.

This project would not have been possible without the WARP platform, pioneered by Patrick Murphy. His technical support, as well as that provided by Siddharth Gupta, Gaurav Patel, and Chris Hunter, has been very helpful. I am very grateful to Rahman Doost-Mohammady, Nathan Zuege, Danny Eaton, Chris Harris, Bryan Grandy, Lev Shuhatovich, Josh Blum, and Azalia Mirhoseini for their help and support. Our anechoic chamber measurements were made possible by the Wireless and Communication Group, NASA, Johnson Space Center; we appreciate their ongoing support and collaboration. I would also like to thank the NDSEG, ASEE, and NSF (grants EARS 1444056, CRI 1405937, and CNS 1518916) for their financial support.

Words cannot express how grateful I am to my wonderful parents, who have provided my education and this opportunity.

Contents

Illustrations

Chapter 1

Introduction

Demand for mobile data is increasing exponentially, having grown 18-fold in the past 5 years, and forecasts predict another 7-fold growth in the next 5 years, [3]. Both government and industry are scrambling to keep up with demand: the FCC's "National Broadband Plan" calls for releasing an additional 500 MHz of spectrum for wireless data [4, 5], while industry is pursuing denser wireless deployments as well as new technologies such as mm-wave. Over the past 7 years many-antenna Multi-User MIMO (MU-MIMO), including "massive" MIMO, has quickly advanced from theoretical to experimentally tested. Many-antenna MU-MIMO has the potential to increase wireless capacity by an order of magnitude, and has emerged as one of the key technologies for next-generation wireless systems, including 5G, [6–9].

While laboratory-scale experiments have verified many-antenna MU-MIMO's potential, [10–12], it still faces many challenges for real-world adoption. The size, cost, computational requirements, and power-consumption of 10s, 100s, or even 1000s of radio chains on a single base station pose a significant problem to real-world adoption. However, the fundamental limit to the performance of many-antenna base stations is mobility. MU-MIMO achieves its capacity gains by beamforming signal towards users, while simultaneously suppressing interference between the users. If the users, or the environment, moves, then the users not only lose signal strength, but also receive interference from the signal intended for other users. Thus the base station must repeatedly collect the users' Channel State Information (CSI) then recompute the beamforming pattern to maintain high capacity. In mobile environments, the overhead required to track users can quickly eclipse the performance gains from MU-MIMO, actually leading to lower capacity! Another fundamental challenge facing the adoption of many-antenna MU-MIMO is the “gain gap” in the control channel. Once the base station obtains users’ CSI it can focus all of its radios’ power in a narrow beam towards the users, however collecting CSI is limited to the range of a single antenna, leading to a classic “chicken-and-egg” problem for initiating MU-MIMO channels.

This thesis presents practical solutions to addresses the critical challenges of *system architecture*, *the gain gap*, and *mobility* facing scaling up MU-MIMO for real-world applications. We implemented three generations of many-antenna MU-MIMO platforms: ArgosV1, a non-realtime prototype, ArgosV2, a realtime research platform, and, ArgosV3, a compact, power-efficient, outdoor-deployable system which fully supports the distributed Argos [13] architecture using custom hardware designed to support many-antenna MU-MIMO. To address the gain gap, we devised and implemented a highly-efficient control channel, *Faros*, that enables many-antenna MU-MIMO base stations to reliably cover their entire potential coverage area, effectively reducing the per-antenna transmit power required at the base station. This drastically reduces the size, cost, power, and thermal dissipation requirements on the base station. We derive a theoretical achievable rate model of a real MU-MIMO system with mobile users, and show that computational and channel estimation overhead fundamentally limit the achievable rate in many scenarios with realistic parameters. To verify these results, and gain a deep understanding in to how many-antenna MU-MIMO channels behave in the real-world, we conducted a comprehensive measurement campaign spanning the UHF, 2.4 GHz, and 5 GHz bands with up to 104 antennas serving

8 users. These measurements include results from line-of-sight (LOS) and non-line-of-sight (NLOS) topologies, indoor and outdoor environments, and drastically varying mobility. Our analysis of these results reveal characteristics of MU-MIMO channels that are fundamental to guiding system design, particularly with regard to mobility and environment. In particular we show that in multipath environments mobility is relative to the carrier frequency, whereas in LOS environments mobility is relative to beamwidth, which drastically affects how often the users' position (in channel space) needs to be estimated. Furthermore, we find that even with pedestrian mobility, the achievable rate fluctuates drastically within a millisecond for 2.4 GHz and 5 GHz, whereas stationary users exhibit stable rates indefinitely, regardless of carrier frequency. The channel measurement system, channel measurements, and analysis tools are all released freely online [14], with the hope that they will help guide the development and deployment of next-generation MU-MIMO wireless systems. A key insight from these measurements is realistic environments exhibit drastically varying mobility, thus it will be critical for many-antenna MU-MIMO systems to adapt to current channel conditions and actively select users in realtime. To address this challenge we devise a *Mobility-Aware MU-MIMO* system which is able to quickly adapt to real-world scenarios to optimize the achievable rate.

Combined, the system architecture, control channel, and mobility-aware MU-MIMO techniques enable practical many-antenna MU-MIMO systems to be implemented and adopted in real-world topologies with mobility.

Chapter 2

Background

While MU-MIMO [15], and even many-antenna MIMO [16,17], has been well known for decades, it was not until Marzetta's theoretical work [18] in 2010 that "massive" MIMO became an active area of research. Shortly after, in 2011, we built ArgosV1 [13] to investigate the performance of many-antenna MIMO and the system challenges to scaling up. Each chapter in this thesis contains relevant background, however the underlying fundamental operation of MU-MIMO is critical to understanding the system aspects of scaling up.

2.1 Beamforming and MU-MIMO

Beamforming utilizes multiple antennas transmitting at the same frequency to realize directional transmission. Constructive and destructive interference of the signals from multiple antennas causes the signal strength received to vary spatially, leading to a *beam pattern*, shown in Figure 2.1. This beam pattern can be altered by changing the beamforming *weights* applied to each antenna, effectively altering the amplitude and phase of the signal sent from that antenna. *Open-loop* beamforming uses precomputed beamforming weights (beamweights), such as DFT weights [19], to steer the beam in a desired spatial direction, without knowledge of the users' locations. *Closed-loop* or *adaptive* beamforming employs channel state information (CSI) to calculate the beamweights that maximize the signal strength at intended users and minimize the

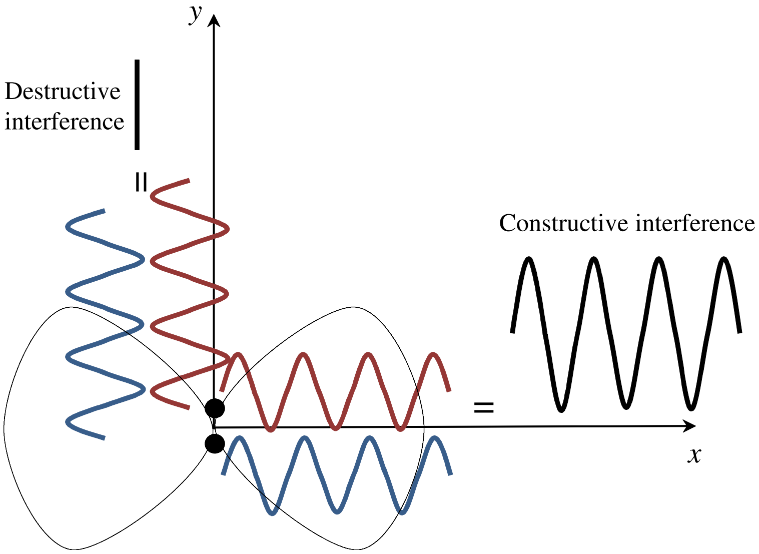


Figure 2.1 : Visualization of beamforming: an overhead view of two antennas, the black circles, spaced a half-wavelength apart, emitting sinusoids. In the horizontal direction the sinusoids are phase-aligned causing constructive interference, whereas in the vertical direction the sinusoids are 180 degrees out of phase, resulting in perfect cancellation. The result is a directional beampattern.

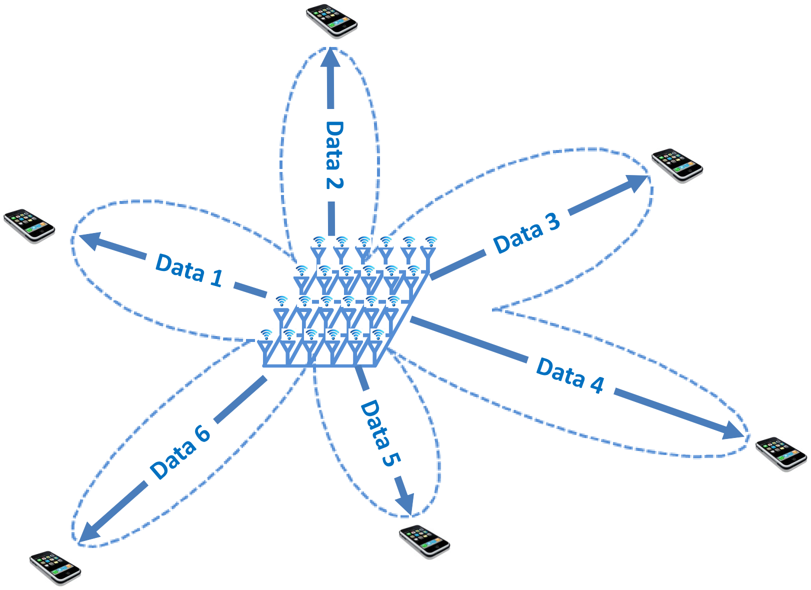


Figure 2.2 : Visualization of multi-user beamforming, also known as MU-MIMO. By carefully cancelling interference between the users, the base station is able to beamform separate signals to each intended user at the same time, effectively multiplying the system capacity.

interference at unintended ones.

Multi-user multiple-input multiple-output (MU-MIMO) base stations leverage multiple antennas, each with its own radio, to serve multiple users simultaneously on the same time-frequency-code resource, typically through closed-loop beamforming, as shown in Figure 2.2. For simplicity, we use the term antenna to include both the radio and antenna. It is well-known that the spectral and energy efficiency of MU-MIMO systems grow with the number of base-station antennas (M) and the number of concurrent users (K), given $M \geq K$.

Many-antenna MU-MIMO: In light of this, several strong theoretical analyses have advocated a very large number of base-station antennas [20–22], commonly referred to as “massive” MIMO, and widely considered one of the few candidate technologies for 5G cellular networks [23–25]. We use the term “many-antenna” to refer to base stations that have many more antennas than users, but are not necessarily “massive”. There have been a number of real-world many-antenna prototypes recently reported, including [13, 26–31], as well as efforts towards commercialization and standardization [6, 23]. The succinct background of many-antenna MU-MIMO relevant to this work is: (*i*) Efficient massive channel estimation requires uplink pilots that are used to infer the downlink CSI via TDD reciprocity. (*ii*) Since channel estimates may only be ephemerally accurate, downlink beamforming must happen quickly after channel estimation.

Chapter 3

Abstract Argos Architecture

The Argos architecture is a distributed many-antenna MU-MIMO system architecture that enables virtually unlimited scalability. At the core of this architecture is a computational split that separates the computation required for each antenna from the required centralized processing. This allows the central processing to be virtually agnostic to the number of physical antennas on the base station, distributing the beamforming computation to each radio module and enabling almost unlimited scalability. To maintain scalable CSI collection overhead, Argos leverages TDD-reciprocity, which requires calibration for implicit channel sounding.

Argos employs a fat tree structure, shown in Figure 3.1 to be highly scalable while limiting end-to-end latency. A central controller is connected to a tree of hubs, each of which connects to multiple chains of radio modules. The central controller performs all centralized computation, e.g., equalization, Forward Error Correction (FEC), and above, including the MAC, whereas the radio modules perform all computation required for each radios, e.g., the filtering, discrete Fourier transforms, and beamforming. This split between the upper and lower PHY, shown in Figure 3.2, not only ensures that every radio module added to the system contains the required computational capacity to support those additional radios, but it also creates a constant throughput requirement along the entire chain of radios, as well as between the hubs and central controller. In this distributed architecture the hubs are very simple devices, and only need to broadcast the downlink data and clock to each chain of radio

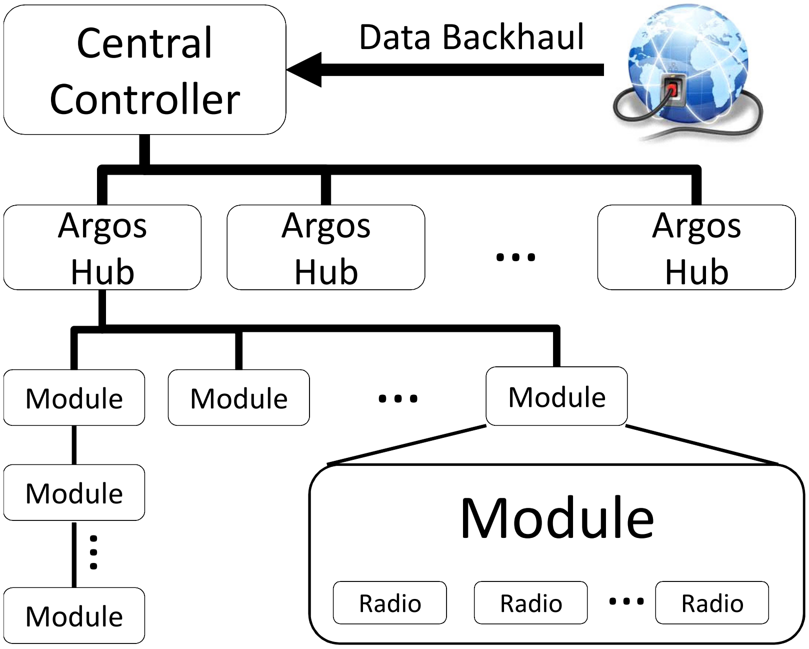


Figure 3.1 : The distributed Argos architecture enables virtually unlimited scalability while limiting end-to-end latency and improving redundancy.

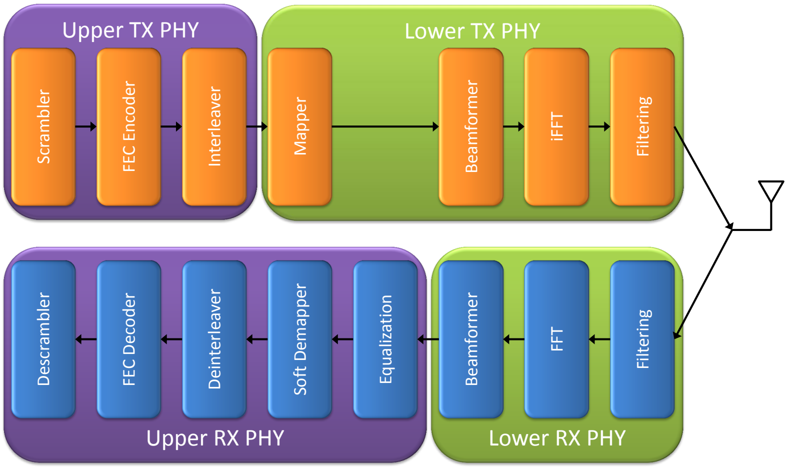


Figure 3.2 : Abstract upper and lower PHY processing split. In Argos, there is one upper PHY per spatial stream, and one lower PHY per radio. Note that in the TX path the upper PHY is broadcast to every radio, whereas in the RX path the lower PHY from every radio is recombined before reaching the upper PHY.

modules. However, to maintain constant throughput requirements in the uplink, each radio module and the hubs have to combine the post-beamformed streams from each radio, which is just a trivial addition operation.

Notably, this architecture enables hubs to be arranged in a multi-level tree structure for further scalability, or even hubs to be inserted after daisy-chained radio modules. However, we find that in real implementations this is rarely necessary, as a single hub can support hundreds of radios, as shown in §5.

Since this abstract architecture is part of our prior work, we defer to [13] to provide a more detailed explanation.

Chapter 4

System Performance Model

To better understand the expected behavior of many-antenna MU-MIMO in the real world, as well as guide hardware design, we developed a detailed system performance model [32]. In particular, traditional linear precoding, i.e., beamforming, techniques do not scale up well with the number of antennas. For example, the predominant MU-MIMO linear precoding techniques, e.g., *zeroforcing* and MMSE, leverage a pseudo-inverse of the channel matrix to nullify or mitigate interference within multiple spatial streams; this requires centralized processing, utilizes non-parallelizable algorithms, and has polynomial complexity with regard to both the number of base station antennas and users served. Thus, to overcome this scalability challenge, recent theoretical work proposed applying the simplest form of linear precoding, *conjugate beamforming*, to many-antenna base stations, and showed that as the number base station antennas approaches infinity their achievable capacity approaches optimal [18]. A modified form of conjugate beamforming can not only be fully distributed and parallelized, but also has linear complexity with the number of base station antennas [13].

Unfortunately, our recent experimental work has shown that even with a substantial number of base station antennas conjugate performs significantly worse than zeroforcing. For example, it only achieves 45% the zeroforcing rate with 64 base station antennas [13]. However these results only indicate the beamformed achievable rate *after* the channel state information (CSI) has been collected and the required computation completed, thus it neglects the computational overhead and the realtime requirements of a practical system. This leads us to an important question in the field of many-antenna base stations: Under what scenarios, if any, does decentralized beamforming (e.g., conjugate) outperform centralized beamforming (e.g., zeroforcing) in real systems?

Towards answering this question, we draw on our experience in building many-antenna base stations to isolate the key practical factors which affect the performance of a real-world system. At a high level these factors can be classified in to two categories: environmental and design. The environmental factors include *channel coherence* and *precoder spectral efficiency.* These factors are completely independent of the base station implementation, and can be measured for a given location. The design factors include *number of antennas* and *hardware capability.*

These factors exhibit complex and nuanced interaction in practice. We derive an analytical model that captures this behavior to predict the achieved spectral efficiency of linear precoding techniques in realtime systems. Using results from ArgosV1, §5.1, we leverage this model to identify and investigate the tradeoff points at which conjugate can outperform zeroforcing. We find that in a low-end, cost-effective, base station conjugate outperforms zeroforcing at coherence times of up to 38 ms, when serving a modest 15 users. However, this coherence tradeoff point is reduced substantially as the number of users decreases or the capability of the hardware increases.

By utilizing our performance model, base station designers can optimize their cost vs. performance tradeoffs and tailor their design to fit specific deployments. Furthermore, since channel coherence and the number of users can vary substantially in real-world deployments, our results suggest that it will be advantageous for base stations to dynamically switch between precoding techniques to optimize performance, which we call *Mobility-Aware MU-MIMO*, further explored in Chapter 8.

4.1 Performance Factors

The factors that affect the performance of base stations employing linear precoding can be classified as either environmental or design. The propagation environment affects the channel coherence and the precoder's spectral efficiency. The base station design determines the number base station antennas, the number of users that can be served, and the precoding algorithm's latency. We next define each factor and their effect on performance, identify how they cause discrepant behavior in conjugate and zeroforcing precoding, and characterize them in real-world systems.

4.1.1 Environmental Factors

Channel Coherence

Channel coherence describes how "smooth" the physical wireless channel is, in both time and frequency. Essentially, it determines how often CSI must be collected. If the channel changes too much over time, then the previously estimated channel state becomes useless. The duration of this interval is the *coherence time*. Similarly, one channel estimate is not valid for the entire spectrum. Thus, the channel state must be estimated at intervals across the entire wideband channel; the width of this interval is the *coherence bandwidth*.

Coherence time is determined by user mobility. Theoretical models simulate coherence time as the amount of time it takes the user or something in the path of the user to move $1/4$ wavelength. For example, at a carrier frequency of 2.4 GHz (wavelength of 12.5 cm) a user moving at 140 mph has a coherence time of 500 μs.

However, this neglects movement in the environment itself and experimental evaluation has shown that vehicular mobility near users results in less than 300 μs coherence intervals in the 2.4 GHz band [33]. Previous work based on LTE channel models often use approximately 1 ms coherence times [18].

Coherence bandwidth is the approximately flat frequency interval of the channel. Delay spread in multipath environments causes the channel's frequency response to become "rough." However, channels can still be approximated as "smooth" over the coherence bandwidth, usually derived as the inverse of the delay spread. This effectively requires the channel to be estimated at regular intervals across the frequency domain to obtain accurate CSI. In LTE models the coherence bandwidth is 210 kHz, as described in further detail in [18].

Channel coherence determines the latency of CSI acquisition and how long that CSI is valid. Since the CSI is only valid temporarily, the overhead of CSI collection and precoding computation results in a direct loss of system capacity. More importantly, however, this overhead is fixed with respect to channel coherence time. Thus, as channel coherence is reduced, the relative performance loss grows. Since conjugate and zeroforcing have drastically different computational overheads they behave differently as coherence time varies.

Precoder Spectral Efficiency

Zeroforcing and conjugate provide vastly different spectral efficiencies during actual data transmissions [13]. We define precoder spectral efficiency as the capacity achieved (bps/hz) using M antennas to serve K users in a given environment neglecting all CSI and computational overhead. Because these factors are neglected, precoder spectral efficiency is independent of base station implementation (for a given

M and K).

This spectral efficiency is determined by the propagation environment, specifically channel orthogonality, user distance, noise, and interference. It is important to note that the relative spectral efficiency of conjugate and zeroforcing varies significantly with SNR, as further explored in [13, 34]. However, zeroforcing is known to perform poorly in low SNR regimes, so a slightly modified form, often referred to as MMSE, should be used in these scenarios. MMSE has negligibly increased performance overhead when compared to zeroforcing, but performs much better at low SNRs, as analytically shown in [35]. While the relative performance to conjugate still varies with SNR, it is not as drastic since its performance is primarily limited by inter-user interference.

One approach to approximate spectral efficiency is to measure each environmental property to create a channel model and simulate precoder spectral efficiency. Alternatively, we employ a more accurate approach that uses a many-antenna base station to measure spectral efficiency directly, thus capturing the combined effect of these properties on performance.

4.1.2 Design Factors

Number of Antennas

The number antennas, both on the base station or with each additional user, drastically affects the system capacity in two ways. While more antennas increase spectral efficiency, they also increase CSI collection and precoding computation overhead, decreasing the amount of time available to send data.

Typically, each additional base station antenna provides a power gain (both by increasing the total transmit power and improving directionality), as well as a potential multiplexing gain (by increasing the possible number of users served simultaneously). However, when zeroforcing, each additional antenna also increases the amount of data sent to the central processor, increasing transport and processing overhead. In contrast, conjugate can be distributed in a manner requiring no additional overhead with more base station antennas.

Each additional user provides a multiplexing gain at the expense of a data slot being converted to a pilot slot, and less transmit power per user. However, in low coherence channels, it may be impossible to collect CSI for all available users and still have time left to send data, thus limiting the number of users that can be optimally served. Notably, the complexity and relative performance of each precoder grows at a different rate with respect to the number of base station antennas and users. Since zeroforcing has polynomial unparallelizable complexity, its computational complexity grows much faster as M and K increase. This indicates that the optimal number of users to serve is dependent on the precoding technique due to these differences in computational overhead.

Hardware Capability

The base station's hardware determines computation and data transport latency. After CSI estimation, the base station must perform the linear precoding computation before data transmission. Any delay caused by this processing results in a direct performance loss. All linear precoding techniques require the same computation to apply the beam weights. Additionally, even traditional baseband processing for wideband systems, such as OFDM, can cause substantial delay. However, since these overheads are common to both zeroforcing and conjugate, we omit them from our analysis as they do not provide additional insight in the performance tradeoffs; they essentially have the effect of further shortening the coherence time.

While conjugate beamforming requires negligible computation beyond the basic linear precoder, zeroforcing has polynomial time complexity with regard to the number of base station antennas and users, and its matrix inverse operations have internal data dependencies which prevent them from being fully parallelized. Additionally, zeroforcing has a central data dependency: i.e., it requires CSI from each base station antenna at a central location to compute the beamforming weights, then these weights must be sent back to each of the radios. When the base station has a large number of radios serving many users across a large bandwidth, this simple data transportation results in significant overhead thereby decreasing the amount of usable coherence time. Thus, the performance of zeroforcing is dependent on the base station's matrix inverse and data transport performance, as well as channel bandwidth, as further described below.

Matrix Inversion. Matrix inversions have internal data dependencies which prevent full parallelization of the algorithm. As the number of simultaneously served users increases, the resulting inverse latency increase cannot be compensated for with additional hardware.

Matrix inversion is an operation that is $O(MK^2)$ and thus the incurred latency scales cubically with the number of concurrently served users (since $M \geq K$). Each of the component operations are CORDIC rotations and divisions which are orders of magnitude more time and resource intensive than simple multiplications and additions (matrix multiplication is also $O(MK^2)$ but far less complex and can be fully parallelized). Furthermore, inverse operations are well known to suffer from numerical precision instability on poorly condition matrices.

To make matters worse, the inversion must be performed for each coherence bandwidth interval across the entire wide band. For example, a system similar to LTE with a 40 MHz bandwidth and a coherence interval of 210 kHz requires 191 of these inverses.

Examples of realtime performance for such a system are dependent on the type of hardware employed. We consider two realistic inversion engines. On the lower, cheaper end, we consider a high performance desktop (Intel-i7, 4 core, using MKL/SSE) CPU and benchmark the matrix inversion performance. Given that each inverse can be computed in parallel, this system can perform 4 inverses at a time, thus, such a system can perform 191 15x15 matrix inversions in approximately 2500 μs. The best case method of performing a matrix inverse is to use dedicated inversion hardware such as an FPGA or ASIC. This method is far more expensive to implement, but would be appropriate for use in a next generation base station. We consider the FPGA complex matrix inversion specified in [36] and compute the expected inverse latency. For this ideal system, 191 15x15 inversions can be computed in approximately 260 μs, almost an order of magnitude less than the CPU method. Note that due to the non-parallelizable nature of the inverse algorithm, this overhead is not easily addressed by Moore's law, as additional cores cannot reduce the latency of an inverse, which grows with the number of users being served.

Data Transport Performance. Current data transport hardware, such as Ethernet or InfiniBand, range in throughput from 1 Gbps to over 100 Gbps. Along with inversion latency, data transport latency significantly detracts from the performance of zeroforcing transmissions due to the inherent, centralized data dependency.

This requires each channel vector to be transported from the radio, through a switch, to the central controller. Once the inverse is computed, the beamforming weights must be sent back to the radios. Thus this process requires two data transmissions (CSI forward and weights backward), each of which include the hop latency of traveling through the switch, as well as propagation delay. The propagation delay exceeds 5 μs per kilometer, given the reduced speed of light in fiber optic cables. In general, the amount of data in both directions is symmetric, as there is both a CSI estimate and a beamweight required for each antenna on each coherence bandwidth.

Gigabit Ethernet (GbE) can transport data at a rate of 1 Gbps to 40 Gbps and has an incurred hop latency of approximately 20 μs [37]. Common Public Radio Interface (CPRI), which has a similar performance to Ethernet, is typically used for data transport in cellular systems, however it is specialized for sending continuous synchronized I/Q samples, and would have to be altered to support this application. For the round trip transportation of 191 15x15 matrices (with 32 bit complex values), a 10 GbE system incurs a latency of at least 355 μs. InfiniBand is a faster, more expensive transportation system intended for supercomputing clusters that is capable of 40 Gbps throughput with only 1 μs hop latencies [38]. For the round trip transportation of 191 15x15 matrices, this system incurs a latency of approximately 70 μs.

Notably, the data being sent to each user must also be distributed to all of the radios, however this is a common requirement for all precoding techniques, would likely use a separate data link, and is much less sensitive to latency.

Channel Bandwidth. Common high-speed communication systems use wide channel bandwidths in order to increase system capacity. Unfortunately, as mentioned above, the frequency response of this wide channel is not flat, thus CSI estimation and precoding computation has to be repeated at regular intervals across the channel. Thus, the number of inverses and amount of data transport required both scale linearly with the bandwidth. In current LTE standards the largest channel bandwidth is 40 MHz (20 MHz downlink and 20 MHz uplink, in FDD), whereas the next generation of WiFi, 802.11ac, goes up to 160 MHz bandwidths (two bonded 80 MHz bands).

4.2 Performance Model

Using the factors discussed in the previous section, we now present the model which dictates the real-world performance of these linear precoding techniques. These factors exhibit complex interactions in real-world systems; we use our model to capture these interactions and analyze their impact on practical performance.

4.2.1 Parameters

A list of model parameters, sorted by their category, environment or design, is shown in Table 4.1. If a value is specific to a precoding technique it is denoted with a $_{ZF}$ or $_{C}$ for zeroforcing and conjugate, respectively.

4.2.2 Model Derivation

The goal of this model is to find the real-world achieved rate of a linear precoding system when given the channel coherence, number of base station antennas, number of users, hardware capability, precoder spectral efficiency, and bandwidth. At a high level, the system capacity, Θ, can be shown in terms of θ, which is determined by the environmental factors, and γ, which is a result of the design factors:

$$\Theta = \theta \cdot \gamma \cdot K \tag{4.1}$$

This equation describes simultaneous data transmission to K users at a rate of θ bps/hz each, however due to the overhead of channel estimation (E) and processing

Variable	Description	Unit
C_t	Coherence time	s
C_b	Coherence bandwidth	hz
θ	Spectral efficiency per user	bps/hz/u
K	# users	u
M	# base station antennas	
S	Data transport throughput	bps
L	Data transport hop latency	s
$T_{\text{-}1}$	Time to perform an inverse	s
N_b	# bits per CSI	bits
B	Bandwidth	hz
γ	% of time transmitting data	%
E	Channel est. overhead	s
P	Total processing time	s
Θ	Achieved aggregate rate	bps/hz

Table 4.1 : Parameters. Upper set are model inputs categorized by environment and design. Lower set are model variables.

(P), we can actually only transmit γ percent of each coherence time (C_t), where:

$$\gamma = \frac{C_t - E - P}{C_t} \tag{4.2}$$

For each user, it takes ${}^{1}\!/_{C_b}$ time to collect accurate channel information for the whole spectrum (since each spectrum block can be measured in parallel), thus:

$$E = \frac{K}{C_b} \tag{4.3}$$

Since conjugate does not require central processing, it has no processing overhead, so $P_C = 0$. However, due to centralized processing requirements of zeroforcing, it must spend a large amount of time in data transport and computing inverses, and thus has a substantial additional overhead:

$$P_{ZF} = 2 \cdot \left(\frac{M \cdot K \cdot \frac{B}{C_b} \cdot N_b}{S} + L \right) + \frac{B}{C_b} \cdot T_{\text{-}1} \tag{4.4}$$

The first part of the equation accounts for the time it takes to send the ${}^{B}\!/_{C_b}$ channel vectors, each with K entries that have N_b bits from the M antennas to the central processor over a connection with a speed of S and hop latency of L (which includes propagation delay due to cable length). This is doubled, since the central processor then has to send the beamweights back to each of the M radios. If the size of the beamweights and CSI differ, due to the use of codebooks, compression, or quantization, the forward and reverse links can be trivially separated to account for this asymmetry. The second component accounts for the amount of time it takes to perform the $K \times K$ inverses for each of the ${}^{B}\!/_{C_b}$ coherence bandwidths.

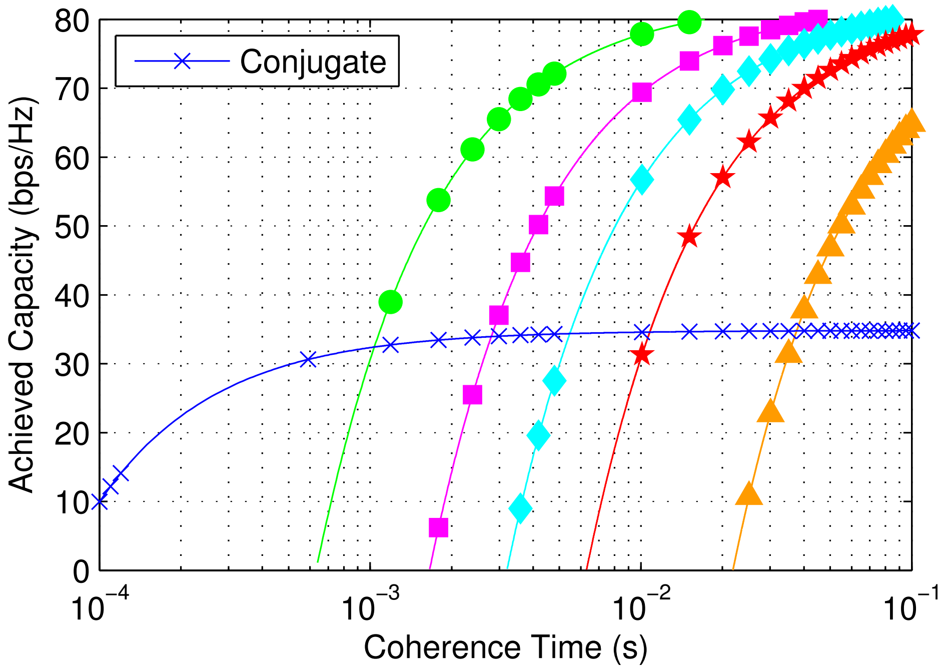
4.2.3 Complete Model

Combining all of the factors we see that the modeled throughput for conjugate is:

$$\Theta_C = \frac{C_t - \frac{K}{C_b}}{C_t} \cdot \theta_C \cdot K \tag{4.5}$$

And for zeroforcing is:

$$\Theta_{ZF} = \frac{C_t - \frac{K}{C_b} - \left(2\cdot\left(\frac{M\cdot K\cdot\frac{B}{C_b}\cdot N_b}{S}+L\right)+\frac{B}{C_b}\cdot T_{\text{-}1}\right)}{C_t}\cdot\theta_{ZF}\cdot K \tag{4.6}$$



	Type	S	L	Inv. Type	Sym.
Super	InfiniBand	40 Gbps	1 μs	FPGA	●
Cluster	4x10GbE	40 Gbps	20 μs	8xIntel i7	■
High	2x10GbE	20 Gbps	20 μs	4xIntel i7	♦
Mid	10GbE	10 Gbps	20 μs	2xIntel i7	★
Low	GbE	1 Gbps	20 μs	Intel i7	▲

Figure 4.1 : Zeroforcing and conjugate performance comparison for different hardware configurations in a M=64, K=15 system.

4.2.4 Simulation

Leveraging our model we analyze the performance of practical many-antenna linear precoding under realistic constraints. We focus on scenarios where the performance of conjugate and zeroforcing cross, as they highlight the conditions when it is important to consider the tradeoffs between the two precoding techniques.

Simulation Methodology

Using the performance model described in §4.2, we input a range of realistic parameter values and analyze their impact on performance. As defined in Table 4.1, there are 11 input parameters to the model; in order to reduce the dimensionality in the presented results, we hold C_b, M, N_b, and B constant, as they yield the least interesting impacts on performance. For all experiments we base the coherence bandwidth, C_b, and channel width, B, on LTE, which defines $C_b = 210$ kHz and $B = 40$ MHz (20 MHz uplink and 20 MHz downlink). Our platform supports up to 64 base station antennas, so $M = 64$. We choose the number of bits in channel estimates and beamweights to be 32 (16 real and 16 imaginary), as this offers low quantization error, and is the width used by our implementation.

We then vary the remaining 7 parameters as follows: We look at channel coherence times, C_t, that range from 500 μs to 100 ms, which are reasonable for real-world mobility, and in-line with the LTE parameters. Using the many-antenna base station implementation described in [13] we collect the real-world spectral efficiency, θ, achieved by conjugate and zeroforcing precoding as the number of users, K, varies from 1 to 15. In order to assess the impact of hardware capability, S, D, L, and $T_{\text{-}1}$, on system capacity, we devise four base stations which range from low-end hardware using Ethernet to high-end custom FPGA designs using InfiniBand; the specifications are provided in Figure 4.1 [37,38]. We assume that processing is local, and thus propagation delay is negligible.

Impact of Factors on Performance

The main factors which affect the performance tradeoffs between conjugate and zeroforcing are coherence time, hardware capability, and number of users. We design simulations which analyze each of these factors, and clearly show their impact on the tradeoff between conjugate and zeroforcing.

Coherence Time and Hardware Capability

We first look at the achieved system capacity of conjugate and zeroforcing with regard to coherence time. Figure 4.1 shows that while serving 15 users simultaneously, conjugate beamforming outperforms zeroforcing at coherence times up to 38 ms in the low-end base station. We clearly see that as the coherence time drops, the overhead of zeroforcing dominates its performance.

However, we can also see in Figure 4.1, that given the specialized “super” high performance central processor and switch we can reduce this tradeoff point to below 1.5 ms. Even using very high-end servers, it is still very difficult to reduce the tradeoff point to below 5 ms.

Number of Users

Finally, we note that as the number of users grows, the performance of zeroforcing quickly degrades under the constraint of low coherence times, as the overhead from data-transport and processing dominate its performance. Figure 4.2 demonstrates a scenario where conjugate begins to outperform zeroforcing with more users; with 4-6

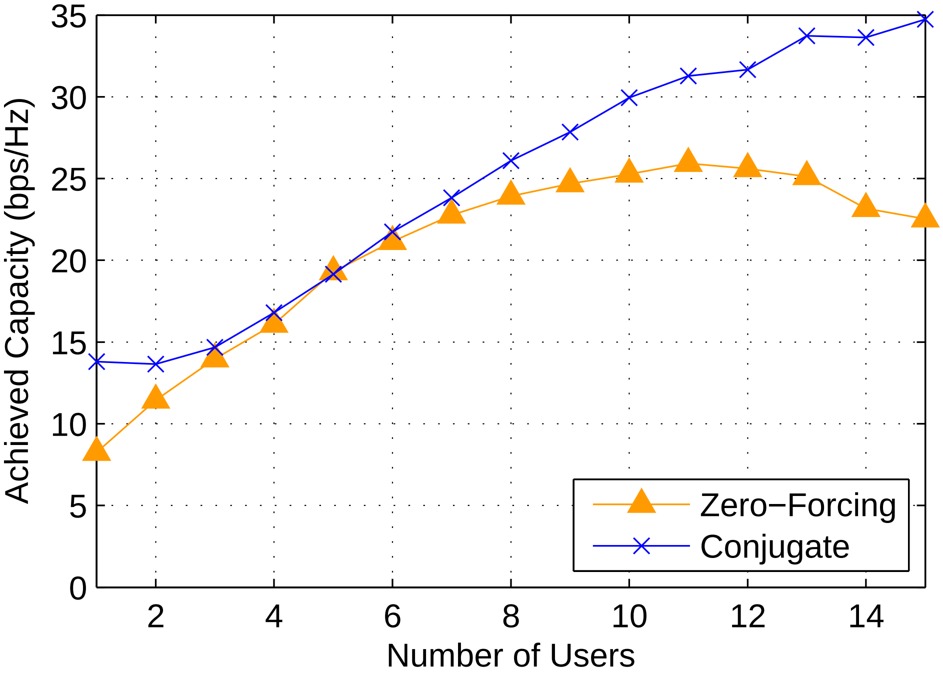


Figure 4.2 : Zeroforcing and conjugate performance comparison for number of users and fixed coherence time of 30 ms with low-end hardware.

users their performance is equivalent, but as the number of users grows to 15, zeroforcing achieves only 65% the system capacity of conjugate. This also demonstrates the criticality of choosing the optimal number of users to serve, as the performance of zeroforcing peaks at 11 users under these constraints. We use the low-end hardware to demonstrate these effects, however higher-end hardware will also show this behavior as the number of users increases; our models show that $\gamma \cdot K$ (an indicator of peak system capacity), under the same 30 ms coherence and 64 base station scenario, is maximal at 49 users, 73 users, 83 users, and 101 users, for the mid, high, cluster, and super hardware configurations, respectively.

4.2.5 Implications

These results indicate that our model can play two important roles in the development of many-antenna base stations: (*i*) guiding base station design and (*ii*) enabling adaptive precoding. We find that conjugate beamforming will be better suited for high frequency bands where coherence is lower and antenna arrays have much smaller form factors, whereas zeroforcing will be more appropriate at lower frequencies with fewer antennas. The actual tradeoff frequencies between these regimes will be a function of user mobility and hardware implementation, and in the tradeoff region adaptive precoding will be useful.

Base station design. Using our model, base station architects can appropriately provision their design to meet real-world performance requirements. By measuring the environmental factors, they can determine the design constraints they need to meet in order to achieve their performance goals. This can help them avoid costly mistakes, such as investing in a zeroforcing system for an environment with very short coherence time.

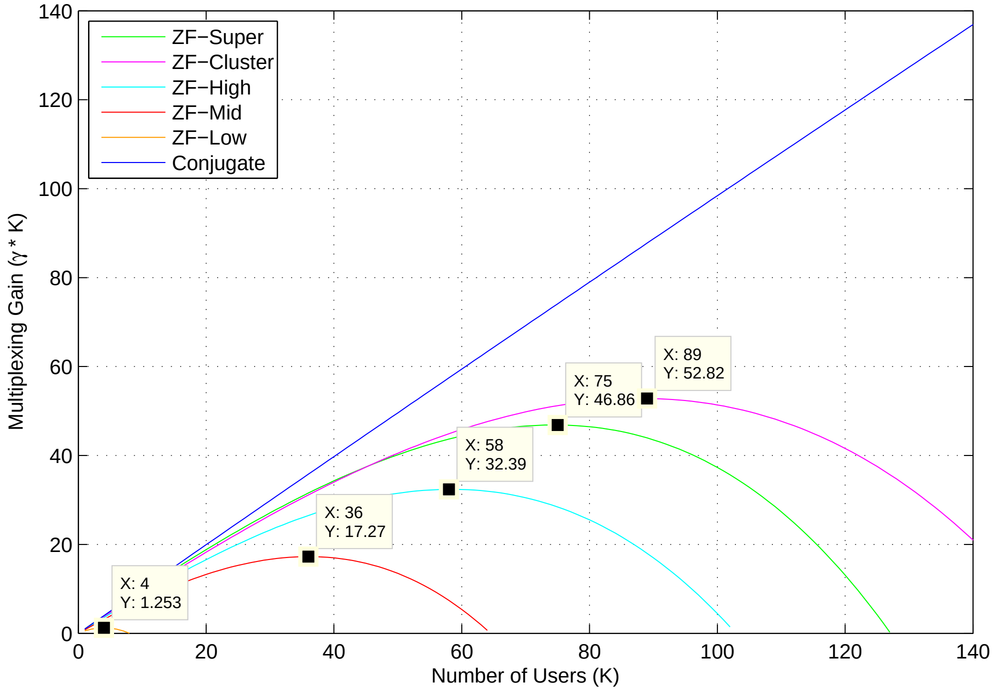


Figure 4.3 : Potential multiplexing gain vs. number of users with 200 base station antennas and a 30 ms coherence time. These results are not scaled by spectral efficiency. This plot highlights the potential impact of hardware capability and processing time on system performance.

Mobility-Aware MU-MIMO. The optimal precoding technique varies according to factors which change in realtime, such as the number of users or channel coherence. Thus, for deployments that encompass the tradeoff points highlighted by our results, it will be advantageous to dynamically switch between conjugate and zeroforcing through mobility-aware MU-MIMO. Since users exhibit widely varying mobility, their coherence time may drop below the threshold where zeroforcing is optimal, and thus the system should dynamically switch to conjugate. Notably, users can be scheduled in groups based on mobility, and thus the precoding can not only be adaptive across time and frequency, but user grouping as well.

4.2.6 Discussion

It is typically very difficult to capture the behavior and performance of complex real-world systems using an analytical model. Our approach addresses this issue by separating the erratic and complex behavior of the environment from the deterministic overhead imposed by the hardware design. This enables system architects to identify and address critical high-level design factors which affect performance from a hardware design perspective then leverage empirical measurements of the environmental factors from the target topology to estimate real-world performance.

Clearly every system design has much more complex internal interactions, such as multiple levels of hardware, software, and data interconnects, as well as possible pipelining, which determine the actual overhead of the high-level factors. These design details can easily be incorporated in to the model. As we develop our own realtime adaptive precoding system we are iteratively refining this abstract model to incorporate concrete implementation details specific to our design. Additionally, as we collect more experimental data from various propagation environments, with more simultaneous users, we will further hone the accuracy and applicability of the model.

We also note that the simulation results presented are a very conservative estimate of the real-world tradeoff points; the parameters chosen are reasonable estimates intended to demonstrate the behavior and trends of the model. Many of the common overheads, such as cyclic prefix, synchronization, control, etc., are omitted from the analysis, and have essentially the same effect as reducing the coherence time. Furthermore, many of the overhead estimates represent idealized, lower-bound, overhead rather than values expected in a full implementation, e.g., data-transport, computation, and CSI collection. However, these values are design and environment specific, and should be determined on a per-system basis, then incorporated in to the model accordingly.

4.2.7 Related Work

While there is plethora of theoretical work on many-antenna base stations, due to the recent nature of this area, to the best of our knowledge, only one explores the tradeoffs between linear precoding techniques. In [34], Yang et al. analyze the radiated power and computational requirements of conjugate and zeroforcing linear precoders. However, when determining the performance of the precoders, the authors do not account for the time it takes to perform these additional computations, nor do they consider other practical implementation issues, such the data transport overhead or the non-parallelizable nature of inverses. Their simulations assume a channel coherence time of 933 μs, which, as we have shown, can cause serious performance degradation in zeroforcing. While this work is very insightful from a theoretical perspective, particularly with regard to energy and spectral efficiency, it neglects the practical implementation challenges facing many-antenna precoding, which drastically affect real-world performance.

Chapter 5

Argos Many-Antenna MU-MIMO Platforms

While models are useful for gaining insight in to the expected performance of many-antenna MU-MIMO, they do not tell the whole story. Real system implementations are required to fully understand the performance of many-antenna MU-MIMO, in particular how the wireless channels behave with mobility.

Until recently, most research regarding massive-MIMO has been restricted to theory that is based on analytical or simulation results. The lack of an experimental platform has prohibited the validation of theoretical results and understanding of the impact of real-world factors on the performance of massive-MIMO. Our previous work [13] demonstrated that a properly designed architecture can realistically scale up to 100s of antennas serving 10s of users simultaneously; however, our initial prototype used antiquated hardware which was bulky, and not flexible and powerful enough to enable realtime wideband MU-MIMO applications. We believe it is critical for the area of massive-MIMO to have an experimental research platform that allows rapid prototyping of new massive-MIMO techniques for real-time streaming applications. Leveraging our experience from the initial Argos prototype, we have refined the Argos design to be more powerful, compact, robust, and scalable in the next two generations of platforms: ArgosV2 and ArgosV3. Unlike Argos, these revisions support rapid prototyping of real-time many-antenna applications with extreme scalability, programmability, portability, and performance. We draw the parallel from ArgosV2 and ArgosV3 to similar platforms used for rapid implementation and testing of small-scale wireless techniques, such as WARP, USRP, and Sora [39–41], which have proven to be invaluable for novel research in PHY and MAC designs.

Building a many-antenna base station to function as a multi-purpose research platform is non-trivial. On one hand, next-generation Argos platforms must have *realtime* capability, i.e., it must support streaming applications. Fast variation of real-world wireless channels, however, is a serious challenge for any MU-MIMO base station, and is the fundamental limiting factor for the performance of massive-MIMO. This is because the many-antenna base station must have accurate channel state information (CSI) to calculate the precoding vectors for each user. The presence of channel fading requires the base station to continuously collect the CSI to each user, within the channel coherence time, which can be as short as 100s of microseconds. As a result, the baseband processing for massive-MIMO must be done rapidly, so that the base station and users can send downlink and uplink data transmission within the channel coherence time. On the other hand, as a research platform, Argos must support programmability of various massive-MIMO techniques such as conjugate beamforming, zeroforcing, MMSE, Tomlison-Harashima, etc. Depending on this massive-MIMO algorithm, often substantial portions of the baseband processing have to be done centrally, whereas other parts can be implemented in a distributed fashion for a performance and scalability gain [13]. Not surprisingly, realizing the real-time capability and programmability of ArgosV2 and ArgosV3 is quite challenging, requiring a careful and flexible design of the base station architecture which enables both centralized and distributed processing in order to accommodate the diverse plethora of emerging massive-MIMO techniques. In addition to the computational and hardware challenges facing a massive-MIMO research platform, we have found that the mechanical and form-factor design are critical components of a flexible, scalable, and usable system. Furthermore, most massive-MIMO techniques share a common requirement of time-frequency synchronization and channel estimation. However, these components are tedious and difficult to implement from scratch, thus we are developing a software framework, *ArgosLab*, to provide these key components out-of-the-box. Finally, to conduct experiments with real-world propagation environments and mobility, we have also developed a battery-powered compact mobile user, *ArgosMobile*.

5.1 ArgosV1

In our previous work we prototyped the first many-antenna MU-MIMO base station, shown in Figure 5.1. This base station was based on the WARPv1 [39] FPGA-defined radio platform, and enabled up to 64 base station antennas beamforming to 15 users. See [13, 42] for more details.

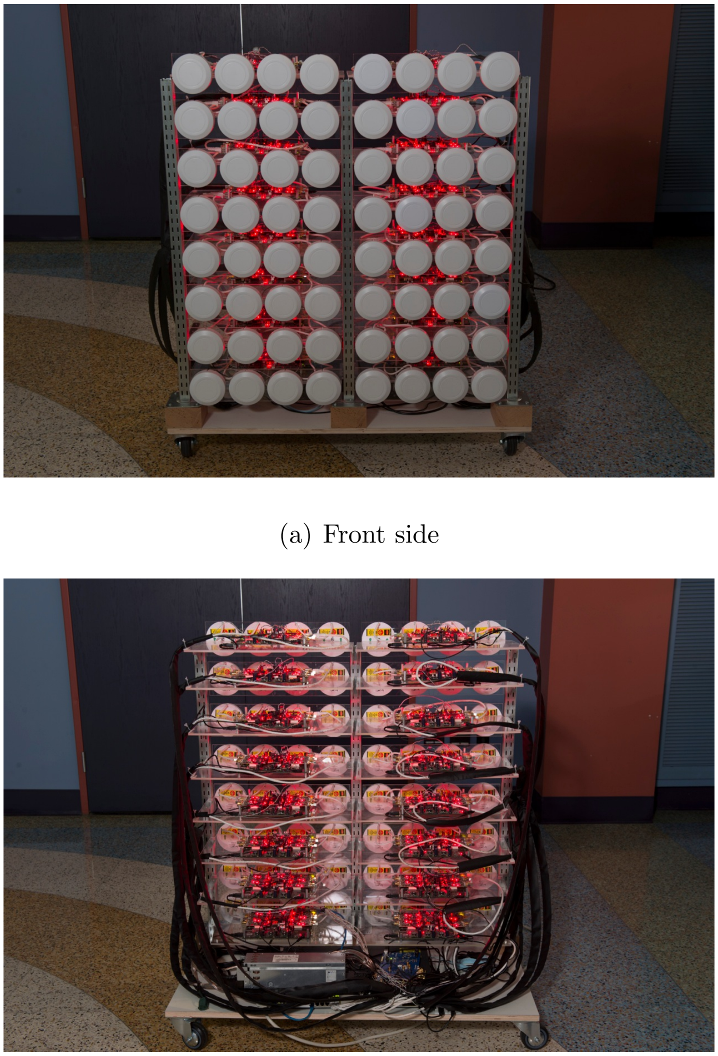
While ArgosV1 enabled real over the air (OTA) MU-MIMO experiments, it was bulky, cumbersome to deploy, and could not support realtime experiments. These limitations lead to the development of ArgosV2.

5.2 ArgosV2

The ArgosV2 platform design can be separated into three categories: 1) the mechanical design, 2) the hardware design, and 3) the software framework. Additionally, we have created ArgosMobile to facilitate real-world experimentation with massive-MIMO techniques.

5.2.1 Mechanical Design

From our previous experience in building a many-antenna base station, we found the mechanical design and form-factor of ArgosV2 is critical for usability, reliability,



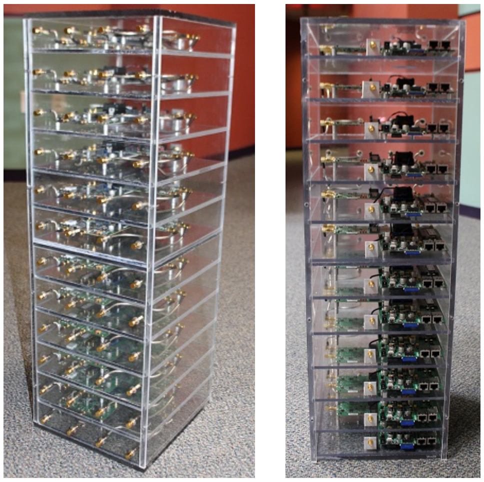
(a) Front side

(b) Back side

Figure 5.1 : Original ArgosV1 platform that supported up to 64x15 MU-MIMO using WARP radio modules.

scalability, resiliency, and performance. The basis of ArgosV2's mechanical design is a custom polycarbonate rack, shown in Figure 5.2. Each rack houses 12 WARPv3 boards on dado-style shelving, for a total of 48 radios. These radios are cabled to 48 panel-mount female SMA connectors spaced a half-wavelength apart at 2.4 GHz; this is the most compact 2D-array design which allows full performance with antennas mounted directly to the rack. The rack was carefully designed to present the SMA connectors on a 9.75 by 29.75 inch face, allowing racks to be stacked next to or on top of each other (using 1/4 inch clips), while maintaining half-wavelength spacing between connectors across racks, both vertically and horizontally. Should different antenna spacing be required, perhaps to test a distributed-MIMO setup or an even more compact array for the 5 GHz band, this configuration enables cables to be easily attached to the panel-mount SMA connectors, allowing antennas to be placed anywhere. For easy-access, troubleshooting, assembly, and field-replacement, the dado-shelves can slide either forward or backward in the rack; all the shelves are held firmly in place by four full-length stops, which can be quickly removed to gain access to all of the WARP modules. This also allows shelves to be easily removed to be placed in remote locations to function as standalone APs, or as part of a distributed antenna system, for maximum flexibility.

We also designed a polycarbonate ArgosHub [13], which uses similar dado-style shelving to house the clock, time-sync, power, and data distribution for up to 32 WARP modules (128 antennas). The hub was designed to be 19.75 inches wide with casters on the bottom, allowing two racks to attach flush to it, creating a single portable base station unit. Moreover, the modular design allows the base station to be easily scaled up with more racks, or split in to multiple small base stations, as well as quickly disassembled and reassembled for additional portability (such as for



(a) Front side (b) Back side

Figure 5.2 : The ArgosV2 base station rack accommodates 12 WARP boards with 48 antennas; multiple racks can be easily connected using an ArgosHub to scale up the base station to support hundreds of antennas.

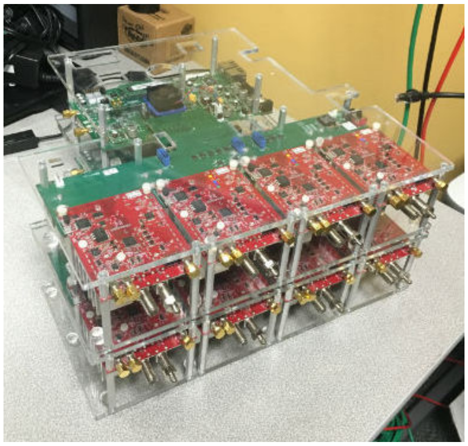


Figure 5.3 : 8-antenna ArgosV2 UHF base station using WURC daughtercards [1].

placement on inaccessible rooftops). Hubs can be easily daisy-chained or placed in a tree topology to support more radios.

Polycarbonate was chosen for its good impact resistance, as well as low cost and weight. While polycarbonate has relatively good anti-static properties, compared to acrylic, to ensure full protection against static discharge damage, we coated the racks and hub Lycron anti-static spray. Ventilation holes were strategically placed to promote airflow around the boards, and the racks were thermally tested to verify that the WARP boards do not overheat; we found that under continuous operation the temperatures of all boards stabilized below 50° C, which is well below the maximum operational temperature rating of 85° C.

This carefully planned mechanical design ensures that ArgosV2 is well-suited for diverse research environments, where usability, reliability, scalability, resiliency, and performance are critical.

ArgosV2 is based on the fundamental Argos system design described in [13]. This hierarchal modular design facilitates both distributed and centralized processing, which are required for massive-MIMO techniques. We leverage the new WARPv3 as the radio module to provide powerful, distributed, real-time processing at the radios. Additionally, most massive-MIMO techniques require tight time and frequency synchronization; we use ArgosHub to provide this synchronization, along with power and data distribution.

For frequency synchronization, ArgosHub utilizes 2 daisy-chained AD9523 clock distribution reference boards, each with a very low noise, accurate CrysTek CVHD-950 VCXO. Time synchronization is implemented using FPGA GPIO over twisted pair cabling. We chose a readily available 1300W server power supply to provide 12V DC power to the WARP modules. Finally, data distribution is achieved using a

Netgear GS752TXS 48-port 1 GbE switch with 4 10GbE SFP+ ports, which provides 2 Gbps to each WARP module and up to 40 Gbps to the central controller, which may be comprised of a single server, an FPGA, or even a full compute cluster depending on performance requirements. While it would be advantageous to enable processing in the switch, as described in [13], it is not necessary for most research applications given the high-bandwidth of the switch, and would require an expensive custom design.

The ArgosV2 hardware design leverages off-the-shelf components to provide a powerful and flexible platform that is capable of accommodating the rigorous computational demands of emerging massive-MIMO techniques.

Software Framework

Massive-MIMO techniques share strict requirements for synchronization and CSI collection, and many use common precoders, which are time-consuming and difficult to implement efficiently. As part of our ongoing work to facilitate rapid prototyping and experimentation, we are designing ArgosLab, a software framework which works with ArgosV2 “out-of-the-box” to provide synchronization, reciprocity calibration, and channel estimation, as well as hardware accelerated linear precoding. Our goal is to enable researchers to develop and test massive-MIMO techniques in real-time, and in real-world environments, without having to spend months, or even years, developing the complex hardware and software required to enable such experimentation.

ArgosMobile

A critical component of ArgosV2 is the ArgosMobile, shown in Figure 5.4, as it enables experimentation in real-world propagation environments. ArgosMobile is an autonomous battery-powered mobile user that employs a WARP board to support

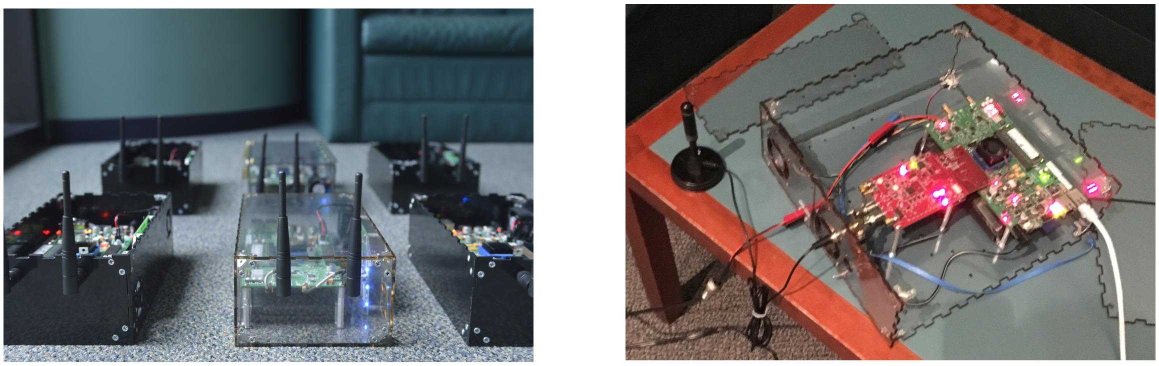


Figure 5.4 : The autonomous ArgosMobile is comprised of a WARP board, a battery, and a dual-band 802.11n bridge, which enables massive-MIMO experiments in real-world conditions with high mobility. *(Left)* Standard ArgosMobiles with 2-antennas. *(Right)* ArgosMobile+ that supports a 4 antenna configuration or a UHF WURC daughtercard.

up to 4 antennas, enabling client-side MIMO techniques. Additionally, ArgosMobile includes a standard 802.11 dual-band WiFi interface for out-of-band feedback and control; this feature is subtle, but crucial for accelerating development, as it allows testing of specific techniques without requiring a fully-functional network stack. The battery life in the ArgosMobile shown is approximately 5 hours.

5.3 ArgosV3

To enable the fully support the Argos architecture, described in §3, we designed a custom ArgosV3 Radio Module, named Iris. We contracted Skylark Wireless LLC, [2], to implement and fabricate the custom Iris Printed Circuit Boards (PCBs). The primary advantage of this custom design is that it supports daisy-chaining, as shown in Figure 5.5, which is critical to correctly implementing the Argos architecture. However, this design has many additional features which make it much more suitable for research, and even commercialization, including Power over Ethernet (PoE), 50

MHz to 3.8 GHz operation, compact and low-power design, modular RF front-ends, and additional computational power.

The compact design enables half-wavelength spacing between radio modules at 3.8 GHz when directly connected in a daisy-chain; spacers or cables can be inserted to support lower frequencies. Each radio module has an LMS7002M with two RF chains, which enables dual-polarization in the most compact form-factor possible.

While development on the ecosystem is ongoing, we have already ported the Argos realtime design, including its channel measurement system, §7.1, as well as OpenAirInterface [43, 44] to ArgosV3. Furthermore, the ArgosV3 modules support the GNU Radio as well as SoapySDR [45] open-source design flows.

5.3.1 Design

Based on our experience building two previous generations of many-antenna base stations, we designed ArgosV3 to be incredibly scalable, flexible, and usable. ArgosV3 fully supports the Argos architecture [13], and is provisioned to support streaming massive-MIMO applications with up to 56 MHz of bandwidth. By employing frequency-agile transceivers and modular frontends, ArgosV3 enables high RF performance from 50 MHz to 3.8 GHz. Each component in ArgosV3 runs a full Linux OS, and the system supports multiple mature software-defined radio projects, including Pothos/SoapySDR, OpenAirInterface, the Argos channel measurement system, and GNU Radio [43, 45–47]. To support truly mobile experiments, ArgosV3 also provides updated battery-powered ArgosMobiles that feature integrated GPS.

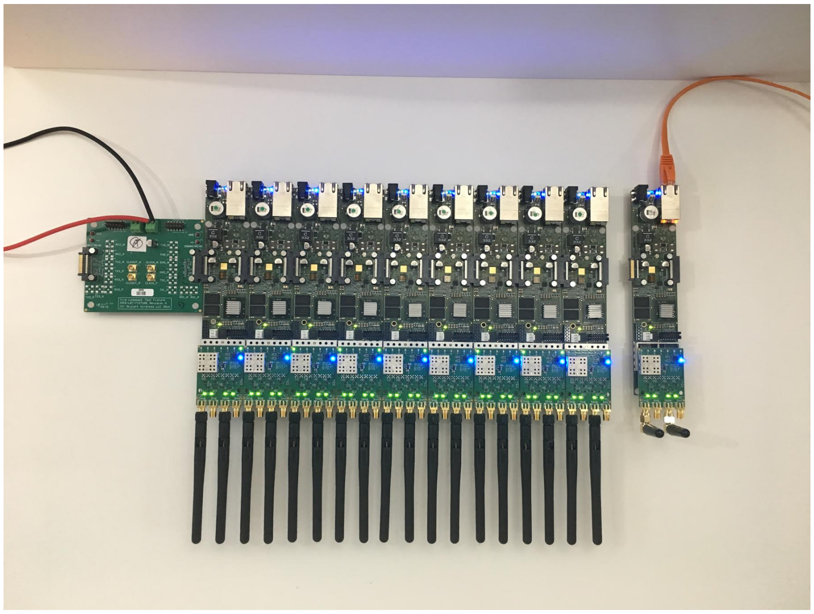


Figure 5.5 : ArgosV3 Iris radio modules. *(Left)* An 18-radio array connected through a bus providing 13.2 Gbps connectivity, shared clocks, GPIOs, and power. *(Right)* A single ArgosV3 radio module with two antennas powered with PoE. [2]

Computational Design

The ArgosV3 base station is composed of three computational components: 1) a central controller, 2) a hub, and 3) multiple radio modules, shown in Figure 5.6.

The central controller is a standard server, or server cluster, that can be provisioned according to the application. For some design flows the central controller provides all of the processing in the system, and simply transmits IQ values to the radio modules. If required, the central controller can leverage optional PCIe FPGA co-processors, e.g., the Xilinx ZC706 or NetFPGA [48]. The central controller is connected to one or more hubs, each with up to four 10 GbE connections, for up to 40 Gbps of connectivity per hub.

The ArgosV3 hub is based on a Xilinx ZCU102 development board with a custom daughtercard to connect to the radio modules. This daughtercard provides high-precision clocking, power, GPIOs, and up to 13.2 Gbps of bi-directional connectivity to each chain of radio modules for coherent MIMO operation. The daughtercard is provisioned to support up to 8 chains of radio modules, and contains integrated GPS to enable global time-frequency synchronization. Additionally, the Xilinx ZCU102 supports Synchronous Ethernet (SyncE) and Precision Time Protocol (PTP) for network time-frequency synchronization.

The most complex and innovative component in ArgosV3 is the radio module, called an Iris, which provides 2 independent radios. Each Iris contains a Xilinx Zynq 7030 SoC, which provides significant computational capacity with two ARM cores and FPGA fabric. To fully support the Argos architecture [13], Irises can be daisy chained, either with a cable, or directly attached, as shown in Figure 5.5. This series connected bus provides clocking, power, GPIOs, and up to 13.2 Gbps of serial connectivity, completely eliminating the need for additional cabling. While the length of chains is only limited by power, which can be injected at multiple points, for streaming designs we typically limit the chain length to 10 to manage end-to-end latency. Conveniently, Irises can also be directly powered by barrel connectors, or via PoE. Notably, individual Irises can also be used as standalone 2 antenna clients.

RF Design

The Iris radio modules in ArgosV3 employ a modular RF design. Each Iris contains a Lime Microsystems LMS7002M frequency-agile transceiver that supports two radio chains from 50 MHz to 3.8 GHz with up to 56 MHz channel bandwidth. However, to support high performance at a given frequency, e.g., in outdoor long-range environments, band-specific components such as filters, Low-Noise Amplifiers (LNAs), and

Power Amplifiers (PAs), and antennas are required. Thus each Iris connects to an interchangeable RF frontend, designed to support the targeted bands. To date, we have designed two band-specific RF frontends for ArgosV3, the first supports 470-700 MHz UHF operation with 30 dBm transmit power, and the second supports 2.4-2.7 GHz and 3.5-3.8 GHz operation with 28 dBm transmit power. For development and testing on the full 50 MHz to 3.8 GHz frequency range, we also built an RF frontend that simply passes through the RF signals to the Lime transceiver, without filtering or amplification.

Additionally, to achieve a compact form-factor, as well as simplify design and cabling, we designed dual-polarized antennas for the UHF, 2.4-2.7 GHz, and 3.5-3.8 GHz bands, each with over 5 dBi gain across their target frequencies.

Mechanical Design

ArgosV3 was designed to be versatile and highly compact, as well as support outdoor deployments. Each Iris radio module is 1.55" wide, and thus can be spaced 1/2 wavelength apart at the highest supported frequency, 3.8 GHz, when daisy-chained. Since every Iris provides two radios, dual polarized antennas can be directly attached while maintaining this compact 1/2 wavelength spacing, as shown in Figure 5.8. For longer wavelengths Irises can be separated by cables or spacer boards, or RF cables can be used to attach the antennas.

While each Iris was designed to be very power efficient, consuming less than 15 W depending on design and transmit power, in dense configurations heat dissipation is a concern. Thermal considerations are even more challenging for outdoor deployments, where the base station needs to be weatherproofed. To address this issue we built a compact enclosure which provides two fully enclosed extruded heatsinks that span every chain of Irises, one for the Zynq processing unit, and one for the RF power amplifier. These heatsinks have fans at both ends to provide forced-air cooling. This weatherproof compact enclosure, containing a hub, 80 Irises, and antennas, is only 18"x14.5"x13", providing 160 radios in just 2 ft^3. A UHF version that is approximately 40"x20"x6", shown in Figure 5.7, is currently being finalized. This UHF enclosure contains 16 radios, and while it can be used standalone for small-scale deployments, to support massive-MIMO multiple enclosures are deployed and connected to an ArgosHub.

Software Frameworks

ArgosV3 contains both CPUs and FPGA resources in all of its components, enabling it to support virtually any software framework. While not required, for convenience we typically run a full Linux OS on each radio module, as well as the hub and central controller. ArgosV3 supports multiple mature software-defined radio projects, including Pothos/SoapySDR, OpenAirInterface, the Argos channel measurement system, and GNU Radio [43,45–47].

Pothos and GNU Radio provide similar features for rapid prototyping of research systems, including high-level language support, e.g., Python, a plethora of sample code, and even GUI interfaces. The Argos channel measurement system provides a framework for flexibly capturing and analyzing high time-frequency resolution channel traces for many-antenna MU-MIMO. Finally, OpenAirInterface provides a production quality realtime LTE implementation entirely in software (C).

ArgosMobile

Since the capacity of MU-MIMO is fundamentally limited by mobility, ArgosV3 provides compact battery-powered clients to enable truly mobile experiments in diverse environments. This ArgosMobile is currently based on the same Iris radio module used in the base station, however it leverages a custom 3D printed enclosure that also contains 26650 Li-Ion batteries in a 3S2P configuration to provide roughly 8 hours of power in a 2"x2.5"x10" form-factor. A modified Iris that includes integrated GPS, a battery controller, and a USB WiFi bridge is currently being finalized.

5.4 ArgosNet

Many-antenna MU-MIMO, or massive-MIMO at large scale, is a key technology for next-generation wireless systems. Researchers have repeatedly demonstrated enormous gains from massive-MIMO in controlled environments [10, 13, 21], but it has yet to be characterized in realistic at-scale deployments or multi-cell topologies. One of the most promising features of massive-MIMO is its ability to reduce inter-cell network interference [18], however this aspect has been left relatively unexplored experimentally due to the lack of a *multi-cell* massive-MIMO testbeds.

We developed *ArgosNet*, [49], the first at-scale multi-cell many-antenna MU-MIMO research platform. ArgosNet leverages our entirely new many-antenna MIMO basestation design, ArgosV3, described above [50]. Leveraging extensive site surveys, we chose 5 locations on Rice University campus to provide not only complete coverage of the campus, but also diverse propagation environments and coverage overlap to facilitate next-generation research. We deployed fiber, power, and mounting to these sites, each with support for multiple sectors. In its default configuration ArgosNet employs four 100-antenna base stations serving 40 mobile clients, however these radios can be flexibly reallocated to more sites or sectors, or consolidated in to a single 400-antenna base station.

These sites are directly connected to the *ArgosCloud*, a 208-core server cluster with FPGA and GPU acceleration. The direct fiber connection allows these base stations to be synchronized by either Next-Generation Fronthaul Interface (NGFI) (SyncE and PTP) or Common Public Radio Interface (CPRI). The outdoor base stations also contain integrated GPS with support for time-frequency synchronization. This high-precision time-frequency synchronization enables advanced wireless techniques such as massive-MIMO Coordinated Multipoint (CoMP) or proposed theoretical techniques such as Pilot Contamination Precoding [51]. Additional small-cells can be deployed anywhere on campus with Ethernet connectivity for heterogeneous experiments.

ArgosV3 supports multiple software frameworks, including SoapySDR, GNU Radio, and OpenAirInterface [44]. Because none of these frameworks currently provide multi-cell integration, we updated the Argos realtime framework, [46], to support ArgosV3 as well as multi-cell measurements. Using this framework we collected initial multi-cell measurements to demonstrate massive-MIMO's ability to reduce network interference, as well as the potential gains from massive-MIMO CoMP.

5.4.1 Multi-Cell Background

Network interference is well known to be the key limiting performance factor in cellular networks [52]. By nature, beamforming techniques reduce both intra and inter-cell interference by focusing their radiated power on the intended user(s), as described in [18]. The more focused the beams, e.g., by using more antennas in a massive-MIMO system, the less resulting network interference there is. While this feature is well-known, and can even be inferred from previous measurements, e.g., this the underlying principle behind multi-user conjugate beamforming, it has been left relatively uncharacterized experimentally due to the lack of multi-cell massive-MIMO testbeds.

Coordinated Multipoint (CoMP) provides multiple techniques to reduce intercell interference and improve cell-edge performance, and has already been adopted in 3GPP standards. These techniques range significantly in complexity and performance, and have varying levels of synchronization requirements [53]. The simplest form of CoMP is coordinated scheduling, where base stations will only schedule transmission from one of the adjacent cells to or from a user on the cell edge to avoid interference. More advanced techniques include null steering towards users in adjacent cells as well as both coherent and non-coherent joint transmission or reception. These techniques can be directly applied to massive-MIMO, however their performance has yet to characterized experimentally. Given the inherent beamforming capabilities of massive-MIMO, the joint coherent transmission and reception (beamforming) methods are particularly attractive, though they require strict time-frequency synchronization. ArgosNet supports all of these CoMP techniques, and we provide some initial measurements in §7.4.4.

5.4.2 ArgosNet Design

The basis of ArgosNet is ArgosV3, 5.3, a scalable massive-MIMO hardware platform designed from scratch to be power, cost, and space efficient in order to enable outdoor multi-cell deployments.

While ArgosV3 flexibly supports frequencies ranging from 50 MHz to 3.8 GHz with 56 MHz bandwidth, ArgosNet leverages a configuration optimized for the 3.55

to 3.7 GHz CBRS band. Shown in Figure 5.9, this CBRS configuration enables 160 radios and antennas in a compact weatherproof enclosure that is less than 2 ft^3 and consumes less than 1.5 kW of power. Each radio provides up to 28 dBm of transmit power, and is connected to a custom dual-polarized antenna element that provides 6 dBi of gain with approximately 100 degrees of directionality in both azimuth and elevation. Internally ArgosV3 consists of 8 chains of 10 radio modules connected in series to a hub; this hub is based on a Xilinx ZCU102 development kit with a custom daughtercard to provide clocking, power, and data to each chain of radio modules. Leveraging the ZCU102's internal four SFP+ ports, ArgosV3 base stations can support up to 40 Gbps Ethernet or CPRI backhaul.

Additionally, ArgosV3 provides an updated autonomous battery-powered mobile user shown in Figure 5.10. This ArgosMobile provides two radios, integrated GPS and WiFi, and over 12 hours of battery life. Since the performance of MU-MIMO is fundamentally limited by mobility, these truly mobile users are critical for realistically characterizing MU-MIMO performance.

ArgosV3 flexibly provides both CPU and FPGA resources at almost every layer of its architecture, and fully enables the original Argos architecture [13]. It currently supports multiple frameworks, including GNU Radio, SoapySDR, OpenAirInterface, and our custom Argos realtime flow.

ArgosNet Architecture

The logical ArgosNet architecture, shown in Figure 5.12, is relatively straightforward: each ArgosV3 base station is connected directly to a cluster, the ArgosCloud, via four single-mode fiber pairs. These fibers can support CPRI, however in our default configuration we use Ethernet with SyncE and PTP support.

The ArgosCloud consists of commodity servers with Nvidia GTX 1080 Ti GPUs and a combined 208 CPU cores. Additionally, ArgosCloud has a 52-port 10 GbE Arista 7150S switch for flexible networking, and a ZCU102 with an SFP+ daughtercard to provide 12 SFP+ ports, expandable up to 20, with CPRI or 10 GbE SyncE/PTP support. In the default configuration, ArgosNet currently contains four 100-radio base stations. Each base station has one 10 GbE SyncE/PTP connection the central ZCU102, one 10 GbE connection to a dedicated server, and two 10 GbE connections to the switch. Notably, this network topology is quite flexible and can be reconfigured for any desired connectivity, e.g., to replicate an LTE network with an Evolved Packet Core (EPC).

Conveniently the individual radio modules in the ArgosV3 base station can be used standalone with 2 antennas and powered by PoE. This allows them to be connected anywhere there is Ethernet on campus to act as either a stationary client or a small cell to enable heterogeneous topologies in the future. Moreover, multiple radio modules can be connected together to provide more antennas at these stationary nodes.

ArgosNet Deployment

To provide flexible experimentation topologies, we deployed power, fiber, and mounting to 5 locations across Rice University campus, 4 outdoor and 1 indoor, shown in Figure 5.11. Each location can support multiple base stations to test multi-sector topologies, as well as provide different coverage areas. All of the outdoor base stations are mounted on the top of buildings to provide complete coverage of the campus, and three of the locations are adjacent to existing cell sites. In our default configuration three base stations are located outdoors on Brown College, Sid Richardson College, and the Stadium, and one base station is indoors, in our lab in Duncan Hall.

Conveniently all 4 outdoor locations have unobstructed LOS connectivity to each other, which enables direct multi-cell implicit calibration techniques. We obtained an FCC license, call sign WI2XLO, that provides experimental coverage of the UHF, ISM, and CBRS bands.

Cell Synchronization

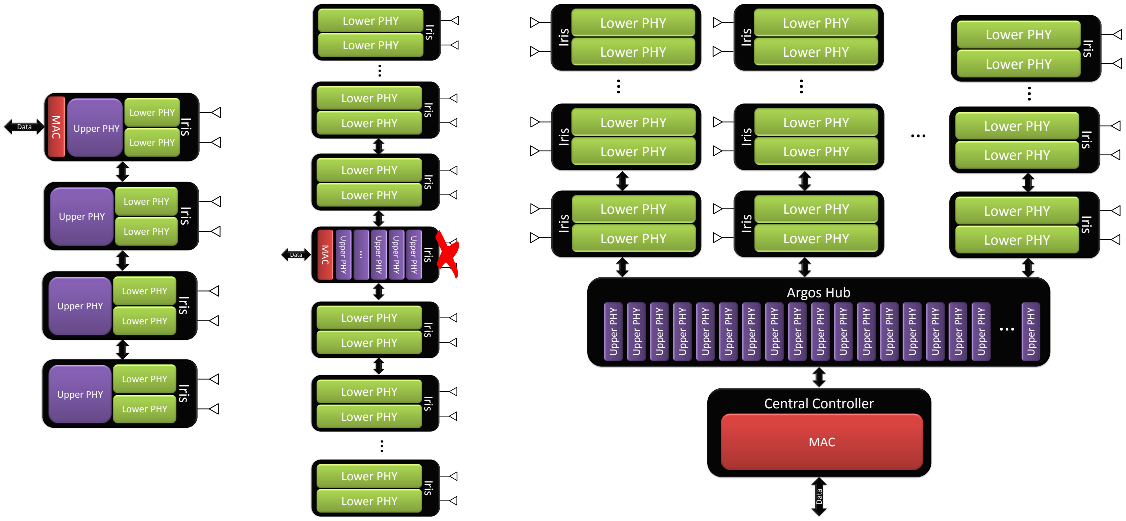
Multi-cell time and frequency synchronization are critical components of next-generation wireless systems, and systems that implement CoMP have even tighter requirements [54]. In general, timing synchronization is used to ensure that protocol frame counters and transmitted symbols across distributed radios are aligned in time, while clock frequency and phase synchronization is used to ensure that transmissions are coherent and beamforming remains accurate [54]. In ArgosNet base stations can be synchronized at long-range using either their direct fiber connection to ArgosCloud or by using GPS. Each approach has its drawbacks and implications for system design.

The direct fiber connection can provide both timing and clock synchronization using CPRI or NGFI, which enables synchronization using the standards-compliant SyncE and PTP protocols. Our ArgosV3 base stations also leverage a u-blox UBX-M8030-KT-FT GNSS module to provide GPS time-frequency synchronization.

All reference clocks go through a clock jitter cleaner in the ArgosV3 base station, ensuring that transmission losses and introduced noise do not diminish radio performance while still allowing clock frequency and phase synchronization across large distributed cellular deployments. We compared three different schemes for clock synchronization: 1) two base stations synchronized to their respective GNSS modules; 2) two base stations synchronized via SyncE over 10 GbE fiber; 3) two base stations in a master/slave configuration with direct clock sharing. Using a Tektronix MSO5054 with DPOJET jitter analysis package, the difference in jitter performance between clocking configurations was undetectable given the noise floor of the measurement equipment.

5.5 Other Platforms

Over the past five years, multiple massive-MIMO and distributed MIMO testbeds have been reported in both academia, [9–13,21,26,50,55–61], and industry, [29,31,62]. These platforms have widely varying capabilities and purposes. [56, 61] are distributed MIMO systems that mainly focus on OTA calibration and 802.11 compatibility. To the best of our knowledge, ArgosV1 and ArgosV2 are the first many-antenna platforms with more than 32 antennas, ArgosV2 is the first realtime massive-MIMO platform, ArgosV3 is the first platform designed from-scratch to address challenges unique to massive-MIMO systems, and ArgosNet is the first at-scale multi-cell massive-MIMO platform. While designing and architecting many-antenna platforms is a significant challenge, it has largely been addressed by [13] and others. Building such a platform is still a substantial engineering challenge, but given the plethora of platforms available we consider the hardware implementation largely a solved problem, and the remainder of this thesis will focus on novel research leveraging our Argos platforms.



(a) Small (b) Medium (c) Large

Figure 5.6 : ArgosV3 base station software and firmware architecture on its 3 scales of base stations. In the small scale base station the upper-PHY and MAC is distributed across the Iris radio modules, eliminating the need for an additional hub. In the medium scale base station a radio module is converted to a combined Argos Hub and Central Controller, computing the MAC and upper-PHY blacks, supporting two chains of Iris radio modules. In the large scale base station a custom Argos Hub performs all of the upper-PHY computations, supporting up to 8 chains of Iris radio modules, while a dedicated central controller implements the MAC. Additionally, a very large scale base station which consists of one hub connected to eight hubs, can support over 1000 radios. [2]

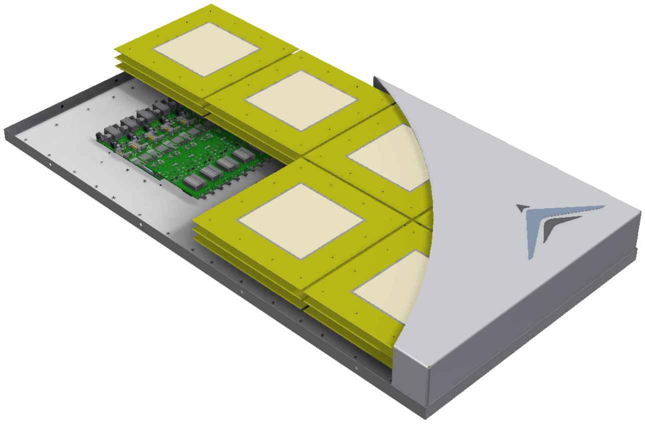


Figure 5.7 : Rendering of UHF enclosure with 16 radios connected to 8 dual-polarized antenna elements. This enclosure is approximately 40”x20”x6” and is currently being finalized for production. To scale to massive-MIMO, multiple UHF enclosures are deployed and connected to an ArgosHub.

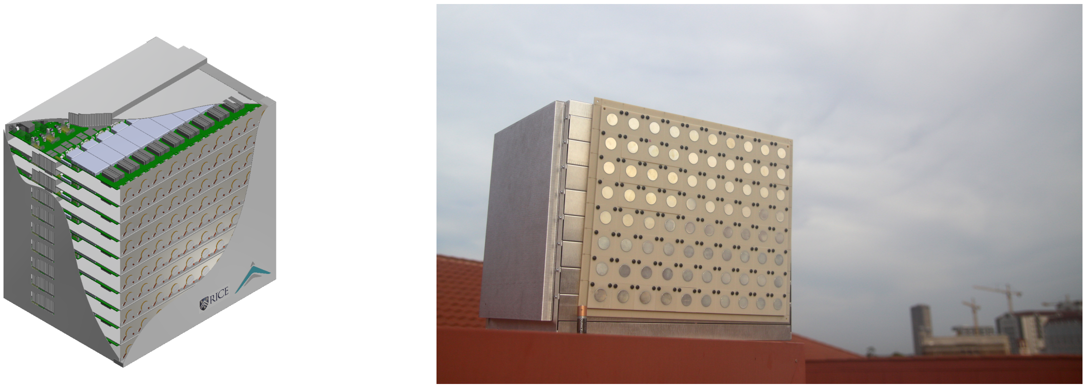


Figure 5.8 : Complete ArgosV3 base station designed for 3.55 to 3.75 GHz operation. *(Left)* Rendering of initial design. *(Right)* First prototype, supporting 160 radios connected to a 10x8 dual-polarized antenna array.

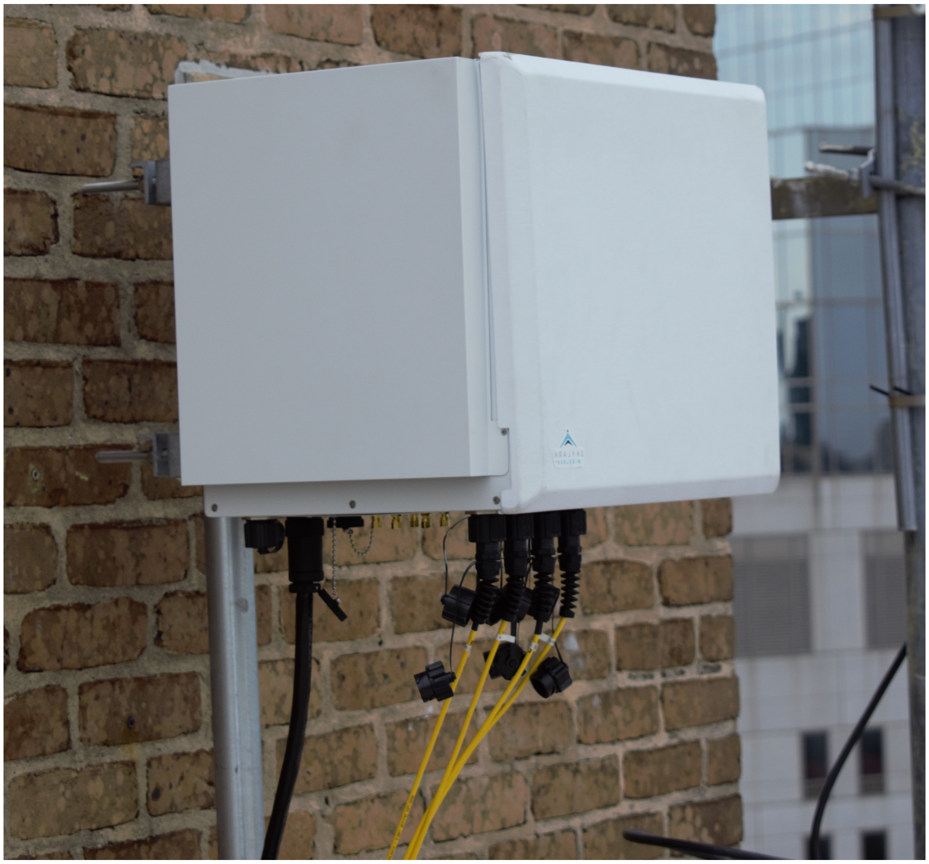


Figure 5.9 : Single 160-radio CBRS weatherproof ArgosV3 base station mounted outdoors. Four single-mode fiber pairs provide up to 40 Gbps connectivity.

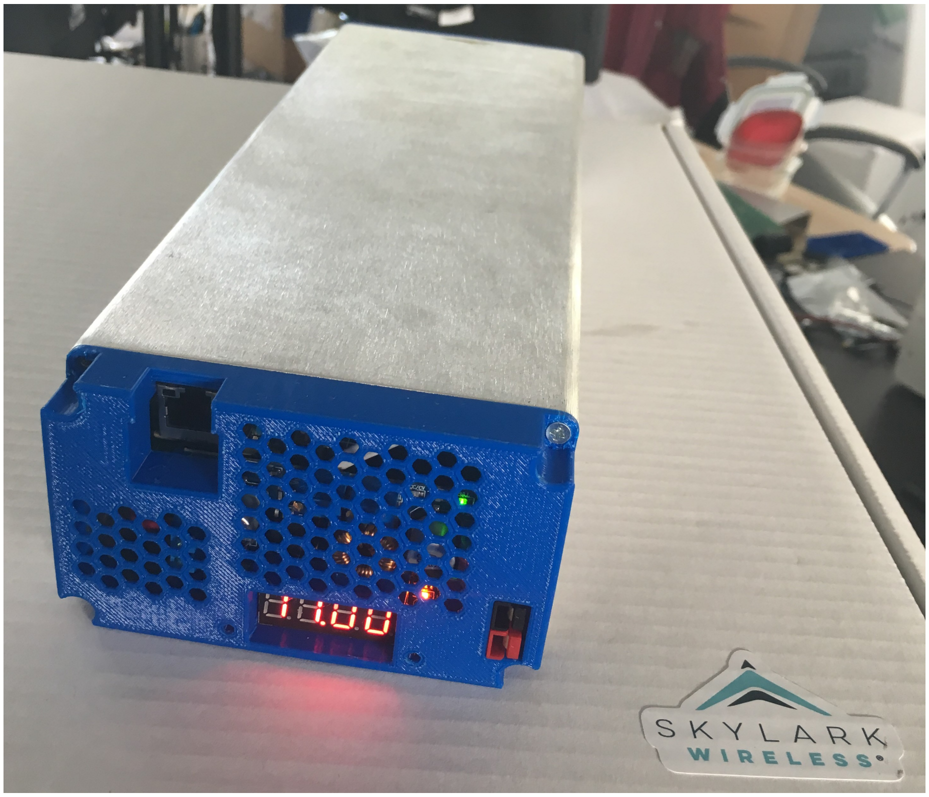


Figure 5.10 : ArgosMobile with integrated WiFi, GPS, and a 12-hour battery life.

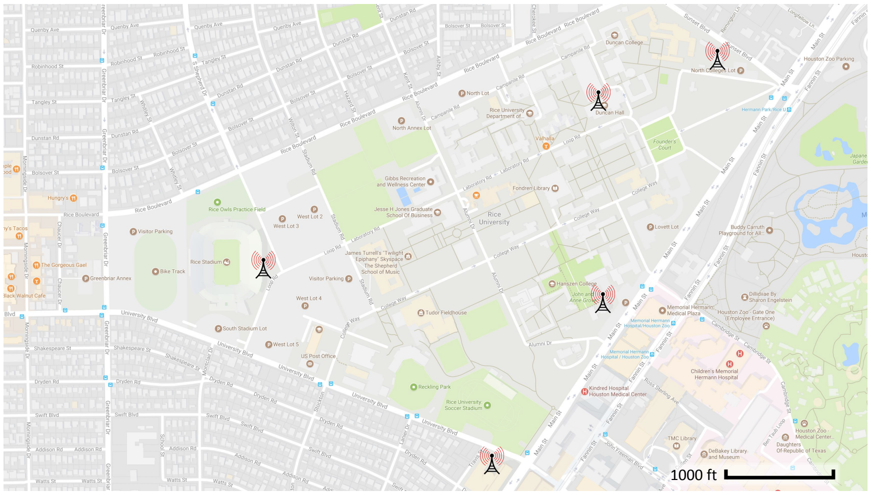


Figure 5.11 : ArgosNet base station locations. Each location has mounting, power, and direct fiber to ArgosCloud. All locations are outdoor sites except for Duncan Hall *top middle*, which is our indoor lab environment.

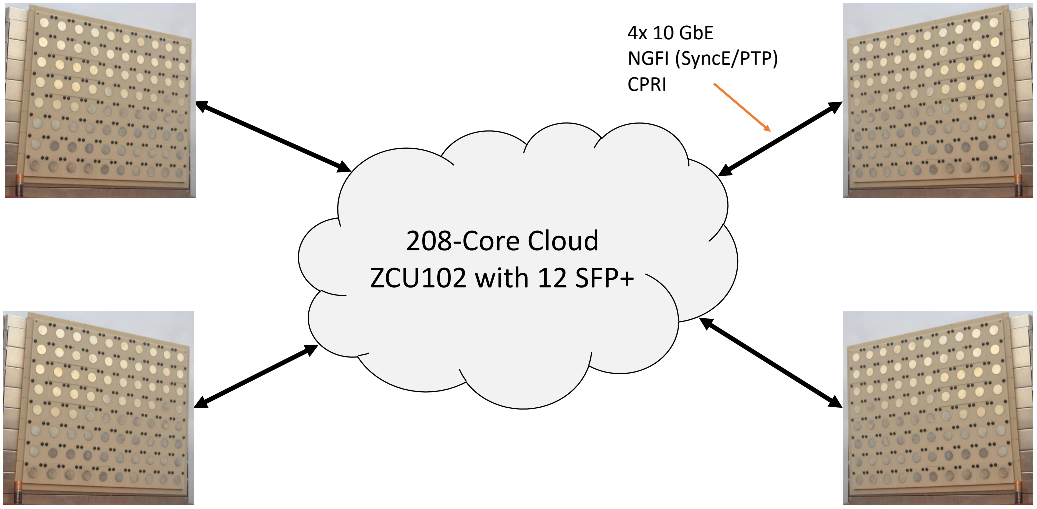


Figure 5.12 : ArgosNet logical architecture. Each base station has four single-mode 10 GbE fiber pairs connected to the 208-core ArgosCloud. In addition to commodity servers, the cloud has a 52-port 10 GbE switch and a Xilinx ZCU102 development kit for distributing NGFI (SyncE/PTP) or CPRI. The ZCU102 has four built-in SFP+ ports, and can be expanded to up to 20 SFP+ ports using FMC expansion cards.

Chapter 6

Faros Control Channel

After building a realtime many-antenna base station, ArgosV2, the first thing we realized is that existing control channels did not scale to many-antenna MU-MIMO. This control channel is required for basic network operations such as time-frequency synchronization, association, channel state information (CSI) collection, random access, and paging, which take place *before* a MIMO channel is established. Today, wireless systems realize the control channel using a single high-power antenna, or simple diversity schemes, but these methods rapidly become very inefficient as the number of base-station antennas (M) increases.

All MIMO base stations have two modes: the *no-CSI mode* that takes place before the base station knows the CSI for the active users, and the *CSI mode* that provides a much more efficient MIMO channel. In order for the base station to collect CSI, it must establish time-frequency synchronization with the users and receive uplink pilots from them; furthermore, once a user becomes inactive, the base station must be able to notify the user of an incoming transmission, i.e. page the user, prompting it to send a pilot. All of these operations are part of the control channel, which is traditionally sent entirely over the no-CSI mode.

In MIMO systems the CSI mode has a gain of up to M^2 higher than the no-CSI mode (see §6.2). When M is small, as in current systems, one can easily overcome this gain gap by simply using a lower modulation rate or a coding gain in the no-CSI mode. However, as M increases, this gap quickly becomes large and problematic. In existing systems *all* control channel operations are performed in the no-CSI mode and sent omnidirectionally to the entire coverage area. Thus, the base station's operational range is limited by the no-CSI mode, which is significantly shorter than that of the CSI mode. One naïve solution is to use higher transmission power in the no-CSI mode. This will lead to more expensive hardware, e.g., power amplifier, possible violation of FCC regulations, and increased inter-cell interference.

We present *Faros*,[*] a novel control channel design that addresses the above gain gap for base stations or access points with multiple antennas. *Faros* leverages two key insights. (*i*) First, *as much of the control channel as possible should be sent over the CSI mode*. We find that the only control channel operations that must use the no-CSI mode are time-frequency synchronization, association, CSI collection, paging, and random access, which are required to establish the CSI mode. By implementing the remaining control channel operations over the CSI mode, we substantially increase their efficiency, as well as avoid the aforementioned gain gap. (*ii*) Our second key insight is that *synchronization and association are not time-critical*. That is, synchronization is valid for 100s of ms and association only happens once; thus by reducing the frequency of synchronization *Faros* is able to substantially reduce the channel overhead of these operations in the no-CSI mode, at the cost of slightly increased association latency at the cell edges.

Guided by these insights, *Faros* leverages open-loop beamforming and coding gains to ensure that many-antenna base stations can achieve their full potential range (see §6.3 and §6.4). Through open-loop beamforming, *Faros* is able to use the full diversity, power, and beamforming gains from all of the antennas on the base station,

[*] $\Phi\acute{\alpha}\rho o\varsigma$, or *Faros*, means "beacon" or "lighthouse" in Greek. The rotation of a lighthouse's strong beam of light is analogous to the beamsweeping employed by *Faros*.

which enables it to scale with M, the number of base-station antennas. Because open-loop beamforming is never as performant as its MU-MIMO counterpart, closed-loop beamforming, *Faros* employs coding gains to further increase the range and to ensure that synchronization and paging are reliable even at the cell edges. To be as efficient as possible, *Faros* only performs these essential tasks and communication outside of the CSI mode, which offers much higher spectral efficiency. Specifically, *Faros* uses open-loop beamforming to sweep extra-long synchronization sequences across the coverage area. This synchronization sequence not only enables users to establish time-frequency synchronization with the base station, but also encodes the base-station ID, and optionally user IDs for paging. *Faros* can dynamically configure important parameters, such as the beam patterns, sweep rate, and sequence length, to match the required gain for full coverage of the desired area. Furthermore, by increasing open-loop beamforming and coding gains in no-CSI mode while reducing the modulation rate or number of users served in CSI mode, *Faros* can be used to extend the range of the base station in remote areas.

We implement *Faros* on Argos, a many-antenna MU-MIMO base station over a 2.4 GHz channel, with 108 antennas (see §6.5) and evaluate the real-world performance and overhead of the implementation (see §6.6). Measurements show that our implementation provides over a 40 dB gain in link budget compared to traditional control channel operations. Anecdotally, this enables us to provide reliable synchronization to mobile users at over 250 meters with less than 100 μW of power per base-station antenna, or 10 mW of total power, using only standard low-gain 3 dBi omnidirectional antennas. Our design facilitates collecting high time-frequency resolution channel measurements in highly mobile environments, with less than 0.5% channel overhead. To reduce the overhead of paging delay, we additionally implement a simple paging scheme that leverages the users last known location for directing the paging signal, which reduces paging delay by 400%.

In designing and implementing *Faros*, we do not invent any new physical layer techniques. Rather, *Faros* contributes a novel synthesis of known methods, such as beamforming, coding, and synchronization, to achieve a very practical and flexible control channel that bridges the gain gap with extremely low overhead. To the best of our knowledge, *Faros* is the first reported control channel design for many-antenna MU-MIMO that can bridge the gain gap.

6.1 Background on Control Channel

In wireless systems, the control channel performs operations required to setup data communication. This includes synchronization, gain control, association, timing advance, random access, paging, setting modulation rates, gain control, scheduling and more. Additionally, in MIMO systems, the control channel must coordinate the collection of CSI across many antennas from multiple users efficiently. This chapter focus on the control channel operations required to establish the MIMO channel, which are *synchronization*, *association*, *CSI collection*, *random access*, and *paging*, as *Faros* performs the remaining control channel operations over the more efficient MIMO channel using existing techniques.

Synchronization: Since nodes in wireless networks do not share oscillators, their time-frequency reference is subject to drift. Thus all high-performance digital wireless communication schemes require tight time-frequency synchronization. In existing systems users establish time-frequency synchronization in four steps: (*i*) First, they auto-correlate the received signal with itself for frame detection and coarse timing. (*ii*) Then, they perform automatic gain control (AGC) to ensure the received signal is within their ADC's dynamic range. (*iii*) Next, they perform a cross-correlation with a pre-known sequence to achieve fine-grained time synchronization. (*iv*) Finally, they leverage the distortion within the known signal, i.e., phase shift, to recover the frequency offset and establish frequency synchronization.

For example, in 802.11 the user continuously performs an auto-correlation to detect the short training sequence (STS) at the start of a packet, which triggers AGC, then performs a cross-correlation on the following LTS for time synchronization. Similarly, in LTE, the user continuously performs an auto-correlation to detect the cyclic prefix of each symbol, then performs a cross-correlation on the PSS and SSS for time synchronization. Typically reference symbols are transmitted throughout the frame in order to maintain this synchronization, as well as compensate for other channel effects. For example, 802.11 dedicates four subcarriers to pilots, and LTE sends reference symbols in a checkerboard-like pattern that are close enough together in time and frequency to continuously correct for drift.

Association: Before a user can transmit or receive data, it must first identify the nearby base stations, select one, then connect to it. To facilitate this association procedure, base stations typically transmit a unique identifier, often called a *beacon*, at a regular interval. Users scan for base stations, often over multiple frequencies, then choose one to associate with based on specific criteria, such as signal strength and authorization. The user then contacts the base station, usually leveraging the same mechanism as random access, to request and coordinate access, e.g., authorization, encryption, and scheduling.

CSI Collection: To obtain CSI, the transmitter sends a pre-known sequence, called a *pilot*, which the receiver uses to compute this amplitude and phase shift for each subcarrier. However, this requires time-frequency synchronization, as without time synchronization the receiver would not reliably know where the pilot starts, and without frequency synchronization there would be inter-subcarrier interference that causes inaccurate channel estimation.

Traditional MU-MIMO systems employ explicit CSI estimation: the base station sends pilots from each of its antennas, the users estimate the CSI to each antenna, then send this CSI estimation back to the base-station. In CSMA systems, such as 802.11, this CSI collection is performed at the beginning of every frame, whereas in scheduled systems, such as LTE, this is performed continuously using reference symbols from each base-station antenna. These techniques do not scale well as the number of antennas and users increase, thus emerging many-antenna systems typically employ implicit CSI estimation: each user sends an uplink pilot which the base station receives on every antenna, which provides uplink CSI, then leverages reciprocal calibration to estimate the downlink CSI based on the uplink CSI [13, 61, 63–65].

Paging and Random Access: Additionally, the control channel handles notifying users when they have incoming data, called *paging*, and coordinating users to randomly access the network when they have outgoing data. Both of these operations must take place before CSI is acquired, as the user has to be paged in order to know it needs to send pilots, or, for random access, it must be able to notify the base station that it has outgoing data so the base station knows to estimate the channel.

6.2 Gain Gap Explained

Multi-antenna base stations operate in two modes: either with CSI or without CSI. With CSI the base station can achieve a gain of M^2 relative to the *peak-power of a single antenna*, whereas without CSI the base station only has a gain of 1 for some

Mode	CSI	no-CSI	**Gap**
Uplink	$M \times P_U$	P_U	$\boldsymbol{M}$
Downlink	$M^2 \times P_{BS}/K$	P_{BS}	$\boldsymbol{M^2/K}$
Gap	$K \cdot P_U/M/P_{BS}$	P_U/P_{BS}	

Table 6.1 : Gain gaps between no-CSI and CSI modes. M is the number of base-station antennas; K the number of concurrently served users; P_U the transmission power of a user antenna; P_{BS} that of a base-station antenna.

control channel operations, illustrated in Figure 6.1. Furthermore, while the physical wireless channel is reciprocal for uplink and downlink transmissions, the transceiver hardware power budget is not, which subsequently creates a second gain gap between uplink and downlink modes. In this section, we take a closer look at these gain gaps, taking in to account real-world constraints and hardware.

Table 6.1 summarizes the analytical results for all modes of operation: no-CSI vs. CSI and downlink vs. uplink assuming an M antenna base station serving K single-antenna users. Each base-station antenna has a transmit power of P_{BS} and each user antenna has a transmit power of P_U. While many theoretical analyses use a total transmit power budget, real systems are constrained by a peak transmit power *per antenna*. For simplicity we assume the average channel and antenna gains are normalized to 1, since they are constant across all modes, and include any non-reciprocal hardware effects, such as the gains from the low-noise amplifiers (LNAs) in the appropriate P, e.g., P_U includes the gain from the base station's LNAs.

We note the above M^2 gain gap is a point of contention, particularly among theoreticians, as they typically assume a total power budget, which reduces this gain to M. In a real system, antennas are peak-power constrained so this would require a single antenna to be provisioned with a much higher total transmit power; in practice this is often infeasible due to cost, power, or regulatory emissions limits. Regardless, there is still at least a gain gap of M, and we analyze both situations in our experiments in §6.6.

6.2.1 Without CSI

To the best of our knowledge there is no existing scheme which performs better than a single antenna for the no-CSI mode control channel operations of synchronization and channel estimation. Thus the no-CSI mode has a gain of 1, which becomes P_{BS} and P_U for downlink and uplink, respectively, as shown in Table 6.1. The gain of an M antenna base station in its no-CSI mode is dependent on what operation it is performing. For CSI collection, there is a fundamental gain limitation of 1 because CSI consists of only information about the link between one antenna and another antenna. Therefore, signals received at other antennas do not contain information about that link's CSI. On the other hand, this theoretical limitation doesn't exist for synchronization, as the desired signal can be sent from all the base-station antennas, which is exploited in our design.

While there are no-CSI mode techniques which achieve a theoretic gain of M, these methods are either impractical, or, in fact, reduce the performance of time-frequency synchronization. One naïve technique would be to use an RF combiner to merge the power output of the M base-station antennas to a single antenna. Not only is this difficult and expensive to implement in hardware, as it requires perfect phase matching to avoid feedback in to the radios, and complex wiring, but it also loses the diversity gain of the M antennas. In essence this is just using a single high-power transmitter, i.e., it is no longer an $M \times K$ system. Despite these drawbacks, we

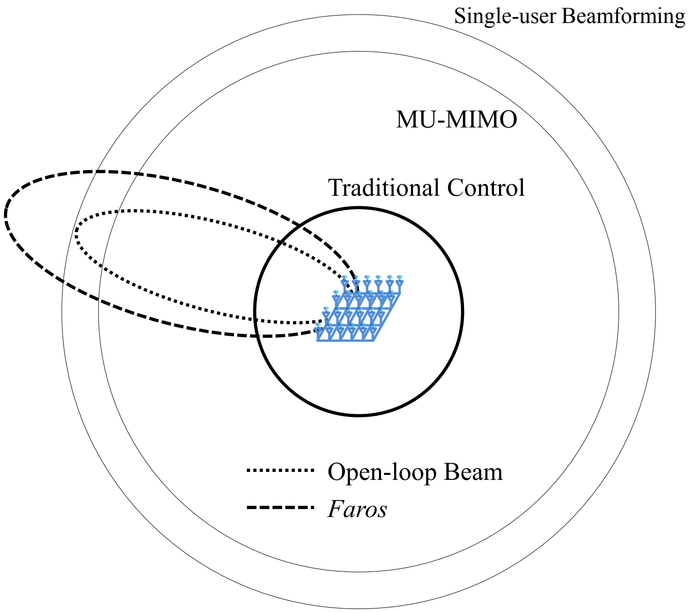


Figure 6.1 : The downlink gain gap. Note that while the figure depicts omnidirectionality, the gap is equivalent for directional antennas.

include this scheme for comparison in our experimental analysis. Another method, which is currently used in multi-antenna systems, such as 802.11n and 802.11ac, is cyclic delay diversity (CDD), which cyclicly rotates the symbols by different amounts of time from each antenna [66]. CDD spreads the power output of all M antennas spatially, and can be thought of as arbitrarily beamforming on different subcarriers. This causes time-domain distortion, which substantially degrades the performance of existing synchronization techniques, and, even worse, this performance degrades rapidly as more antennas are added [66]. Finally, both of these naïve schemes only help in the downlink, and do not provide any gain in the uplink.

6.2.2 With CSI

The potential power gain of an $M \times K$ MU-MIMO system with CSI, in both uplink and downlink, is well known to be $P \cdot M$, where P is the transmission power [67]. Leveraging CSI, the base station can direct radiation towards, or listen to radiation from, the intended K users using beams with an approximate width of $1/M$, which provides a spatial power gain of M. In the downlink, the base station transmits power from all M antennas, but has to split the power among K users, thus providing a per-link power of $P_{BS} \cdot M/K$, assuming equal power allocation among the users. In the uplink, the base station receives power from each user on all M antennas, thus providing a power of P_U. This renders a total gain of $M^2 \cdot P_{BS}/K$ and $M \cdot P_U$, respectively, as shown in Table 6.1. Note that a MU-MIMO base station capable of serving K users likely will not always serve K users simultaneously; with a single user the gap increases to a full M^2.

6.3 Faros Gain Matching

With the gain gaps above, we next present the design of *Faros* in two parts: (*i*) mechanisms to bridge the gain gaps (this section), and (*ii*) the control channel system design that overcomes the limitations of these mechanisms (§6.4).

To bridge the gain gap of the no-CSI and CSI modes in the downlink, *Faros* combines open-loop beamforming with a coding gain. It sweeps open-loop beams carrying orthogonal sequences, which enable the *synchronization* and *paging* operations. In the uplink, *Faros* exploits the natural per-antenna asymmetric transmit power and employs an additional coding gain to enable *CSI collection* and *random access* operations. By encoding a base-station ID in the downlink synchronization sequence and exploiting the random access operation, *Faros* facilitates the *association* operation.

6.3.1 Open-Loop Beamforming

Faros employs open-loop beamforming to exploit the power and diversity of all antennas on the base station. The combined power of the antennas provides a gain of M, and the beamforming provides another gain of M, for a total gain of $O(M^2)$. However, this beamforming gain does not come for free, as it focuses the radiated power on $1/M$ of the antennas' coverage area, thus *Faros* must sweep beams to provide complete coverage. Leveraging our key insight that *association* and *synchronization* are delay-tolerant, *Faros* employs open-loop beamforming for these operations *without* impacting user-perceived performance or creating significant channel overhead.

While there are many MIMO and diversity schemes that exploit the gains from multiple antennas, only open-loop beamforming is effective for time-frequency synchronization, as it provides the full potential combined power and directivity gain from all of the available antennas without causing time-domain distortion. Furthermore, open-loop beamforming has four practical benefits in a real-world MU-MIMO system: (*i*) the increased received power allows the user to employ cheaper RF components, e.g., the LNA, (*ii*) the increased directivity and lower total power reduce the interference to adjacent cells, (*iii*) it does not require any additional hardware or computation, as the beamforming precoders are already required on the base station for MU-MIMO, and (*iv*) it allows the coverage area to be finely tuned.

Beamsweeping

To overcome the spatial selectivity of open-loop beamforming, *Faros* employs *beamsweeping* that transmits a signal, $\mathbf{s} \in \mathbb{C}^{1\times C}$, where C is the coding gain, in different spatial directions using beamforming. Fundamentally, beamsweeping trades off increased spatial coverage with additional time overhead. Guided by our second key insight that some control channel operations are delay-tolerant, we leverage beamsweeping for synchronization, and to help facilitate association.

Each beam is defined by a vector $\mathbf{b_n} \in \mathbb{C}^{1\times N}$, thus an N length sweep pattern can be defined by a matrix, $\mathbf{B} \in \mathbb{C}^{M\times N}$, composed of $\mathbf{b_1}, \mathbf{b_2}, ..., \mathbf{b_n}$. The M-antenna base station transmits an entire sweep pattern in N time-slots, as the transmission in a given time-slot n and given base station antenna m is simply: $s \cdot B_{m,n}$. Thus, if each beam is sent contiguously, then beamsweeping takes N times longer than a single omnidirectional transmission of the same sequence. Because *Faros* sends a beam at the beginning of each frame, an entire beamsweep takes $N \cdot F$, where F is the frame duration, as further described in §6.4.6, and shown in Table 6.2.

Complete Spatial Coverage: If $\mathbf{B}$ forms a complete orthogonal basis, i.e., it consists of $N = M$ orthogonal or pseudo-orthogonal beams, then it provides complete spatial coverage. Any complete M-dimensional basis used for beamsweeping will provide complete coverage of the CSI space, since, by definition, the CSI of any user can be represented by a linear combination of the basis. This ensures that for any given point in the coverage area at least one beam in B will not have a perfect null.

It is important to note that as M increases, the probability that a user detects a given beam is reduced, since the energy is more spatially selective. However, the probability that a user will detect at least one beam in the sweep pattern increases, as, given a complete orthogonal basis, at least one beam is pointed towards the user, and that beam has a higher EIRP since it is narrower. While we find this intuitive, we observed this experimentally, as well as verified it with Monte Carlo simulations, using Hadamard beamsweeping.

Techniques and Range: *Faros* can leverage many beamforming techniques with compelling tradeoffs for specific implementations. Without detailed information about the environment and precise calibration, any orthogonal basis with low peak to average power ratio (PAPR) works well for open-loop beamforming. While a complete basis guarantees spatial coverage, it does not guarantee a strong signal. Since it is statistically impossible that every user will have an open-loop beam pointed directly at them, the gain of beamsweeping is reduced by an inaccuracy factor of a, to M^2/a. As such, an overcomplete $\mathbf{B}$, i.e. $N > M$, can provide extended coverage by statistically reducing a. Otherwise, given careful consideration of the propagation environment and antenna placement, as well as hardware calibration, techniques such as DFT open-loop beamforming can be tuned to provide the desired coverage area. For our implementation we choose Hadamard beamforming weights, as further described in §6.5.

6.3.2 Coding Gain

The use of open-loop beamsweeping will reduce the gain gap between no-CSI and CSI modes. To close the remaining gap, *Faros* additionally employs a variable coding gain in both the downlink and uplink. In theory, a coding gain is achieved by sending a signal over a longer period of time, thus, the total received power, integrated over time, increases linearly as the duration increases. However, this gain comes at a cost of increasing the channel usage overhead linearly as well. Coding gains are ideal for tuning the gains to match between modes because they are easily adjustable and thus can be used to dynamically fine-tune the gain vs. overhead tradeoff.

While Table 6.1 analyzes the gain gap in terms of SINR, not all parts of the frame have the same SINR requirements. For example, data transfer can benefit from a higher SINR by altering the modulation and coding scheme. Higher-ordered modulation requires a higher SINR to be successfully decoded, thus it can be thought of as a negative coding gain in the CSI mode. For instance, in 802.11 OFDM BPSK modulation requires 15 dB SINR, whereas 64-QAM requires 31 dB [68]. In contrast, the detection threshold for a length 128 Kasami sequence is roughly -5 dB [69]. This effectively further reduces the gain gap between the CSI mode, which is used for transmitting data, and no-CSI mode, but how much is dependent on actual data modulation rate. By leveraging a dynamic coding gain, the range and overhead of *Faros* can be tuned to the specific needs of each deployment.

Downlink Coding Method: In the downlink, *Faros* transmits variable length orthogonal synchronization sequences to encode the base-station ID and paging information, while simultaneously providing synchronization and achieving a gain, C_{down}, proportional to the length of the sequence. Orthogonal sequences are extensively used in wireless systems; an overview of them can be found in [70]. Since these downlink sequences need to be detected *prior* to synchronization, they must have low streaming auto-correlations, both with themselves and the other sequences in the orthogonal set. That is, since the sequences must be detectable without knowledge of when they start, the receiver must perform a full correlation at every sample, thus a time-shift of the sequences must produce a low correlation; otherwise it could cause a false detection.

Uplink Coding Method: In the uplink, the *Faros* base station assigns orthogonal pilot slots to active users, and reserves dedicated slots for association and random access, as shown in Figure 6.2(b). These pilot slots are variable length to enable a coding gain based on users' channel quality, e.g., users on the cell edges will use longer pilots to increase the accuracy of their channel estimate.

By orthogonalizing pilots in frequency *Faros* is able to increase the accuracy of the channel estimates, and provide an uplink gain of at least K. Frequency orthogonalization (OFDMA) enables all the users to transmit simultaneously, which increases the instantaneous power received at the base station by a factor of K. To collect complete CSI for every frequency, users are further time orthogonalized, as shown in Figure 6.2b. As such, the total power received for a given user, integrated over time, also increases by a factor of K. Theoretically, to obtain accurate CSI each user must send a pilot for at least a duration of the inverse of the frequency coherence every coherence time interval. However, by scheduling users with poor channel quality to send even longer than required by the frequency coherence interval, *Faros* increases the coding gain, C_{up}; this ensures high-quality channel measurements across the entire cell and fully closes the gain gap.

For association and random access, users send orthogonal synchronization sequences on dedicated time-frequency blocks during the training phase. This allows the users to still achieve a coding gain, while simultaneously enabling collision avoidance and timing-advance estimation, as further discussed in §6.4.4.

6.3.3 Combined Gain

Faros employs a combination of open-loop beamforming and coding gain to close the gain gap, as depicted in Figure 6.1. Beamsweeping provides the majority of downlink gain by focusing the full power of the base station on a small portion of the coverage area; it achieves a gain of M^2/a, where a is the beamforming inaccuracy. In the downlink *Faros* reduces the gap between no-CSI and CSI gains from M^2/K to $M^2/K/(C_{down} \cdot M^2/a) = a/(C_{down} \cdot K)$, thus the coding gain should be tuned so that $C_{down} \approx a/K$.

In the uplink *Faros* leverages OFDMA and coding to achieve a gain of $C_{up} \cdot K$ in the no-CSI mode. This reduces the no-CSI to CSI gap from M to $M/(K \cdot C_{up})$, which suggests C_{up} should be roughly M/K to close the gap.

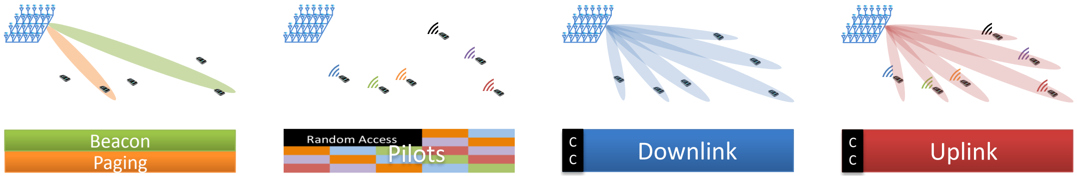
However, once a proper downlink coding gain, C_{down}, is applied, combined with beamsweeping, the *Faros* no-CSI downlink gain is M^2/K. In contrast, the no-CSI uplink gain is only $(C_{up} \cdot K \cdot P_U)$, which leads to a new gain gap. To mitigate this gap, in *Faros* the *total* transmission power of the base station and user need to be roughly the same, e.g. $O(P_U) \approx O(M \cdot P_{BS})$; this is typical of existing bidirectional communication systems, though macro cells can have as high as a 10 to 18 dB difference. This reduces the gap from $(C_{up} \cdot K \cdot P_U)/(M^2/K \cdot P_{BS})$ to $(C_{up} \cdot K^2)/M$, and suggests that the uplink coding gain should be tuned to approximately M/K^2, along with any residual discrepancy between P_U and P_{BS}, to finish closing the gap.

Comparing the C_{up} needed for the no-CSI vs. CSI, M/K, and uplink vs. downlink, M/K^2, we see there is a residual gap of K. Since the range of the base station is limited by the downlink mode, C_{up} should be selected to match the uplink-downlink gap, then the residual gain of K in the CSI uplink can be used to reduce transmission power or increase modulation rate. Notably, this full coding gain is only required at cell edges, where *Faros* uses extra-long pilots.

It is important to realize that when compared to existing systems, for a given coverage area *Faros reduces* the required per-antenna transmission power of the base station by M^2 and of the user by K.

6.4 Faros Control Channel Design

We next describe the design of *Faros* control channel design and how it realizes synchronization, association, CSI collection, random access, and paging.



(a) Beamsweep (b) CSI collection (c) Downlink data (d) Uplink data

Figure 6.2 : An example *Faros* frame structure. First, in (a), the base station beamsweeps a beacon that provides the users with time-frequency synchronization and the base-station ID. If a user needs to be paged, the base station will simultaneously beam a paging sequence towards that user. Next, in (b), users send orthogonal uplink pilots in scheduled slots. Users that require random access or association send an uplink pilot in the one of the reserved slots. Finally, in (c) and (d), the base station leverages the acquired CSI to provide downlink and uplink data connectivity, as well as any remaining control channel information, over the efficient MU-MIMO link.

6.4.1 Synchronization

Faros achieves both time and frequency synchronization by beamsweeping carefully designed, extended-length, sequences from the base station to the user.

Time Synchronization

With *Faros*, users perform a streaming cross-correlation on received samples to detect the synchronization sequence sent from the base station. That is, it computes the correlation of the received signal R with the sequence S, $\sum_{i=1}^{n}(R_{t-i} \cdot S_i^*)$, at every sample. This produces a peak at the single sample when R and S are aligned in time.

While this is the same concept employed by existing systems, *Faros* faces two new challenges: (i) *Faros* needs to detect multiple synchronization sequences simultaneously since it uses both beacon and paging sequences for synchronization, which are sent simultaneously on separate beams. (*ii*) *Faros* needs to perform time synchronization *without* coarse timing information or automatic gain control (AGC). As discussed in §6.1, existing solutions leverage coarse frame detection and AGC to achieve fine-grain time synchronization; however, these techniques are inefficient or even impossible for *Faros* to employ in the no-CSI mode. This is because *Faros*' beamsweeps and MU-MIMO downlink are highly spatially selective and, as a result, users receive every synchronization sequence with highly varying power. While *Faros* could precede every synchronization sequence with a training sequence to facilitate coarse frame detection and AGC, similar to the STS in 802.11, this training sequence would have to have significantly increased length to overcome the gain gap. Moreover, the gains set by this sequence would only be valid for a single beam, making it highly inefficient.

Faros addresses these two challenges with three techniques. First, it employs two full-precision correlators. Existing implementations, such as [39, 71], perform only 1-bit and 3-bit correlations, respectively, and only detect a single pre-set sequence. While this approach is computationally efficient, it does not work well without gain control, and performs poorly when trying to distinguish different sequences. By performing two parallel full-precision correlations, e.g. 12-bit for WARPv3, *Faros* is able to reliably detect synchronization sequences with highly varying signal strengths, as well as reliably distinguish paging and beacon synchronization sequences that are sent simultaneously.

Second, since performing AGC on every sequence is inefficient, *Faros* employs *transmit* gain control. That is, since *Faros* beamsweeps the sequence, a user receives every sequence with a substantially different signal strength. Therefore, the users can simply wait for a sequence in the sweep that is within their dynamic range. If they don't detect any sequences, e.g. before discovering any base stations, the users slowly vary their receive gain settings until they detect sequences. After synchronization is established the users listen to all of the subsequent synchronization sequences and adjust their gain accordingly. Notably, *Faros* performs closed-loop uplink gain control identically to LTE [72] and fine-grain downlink gain control is performed at the beginning of each downlink phase, as depicted by Figure 6.2.

Finally, *Faros* dynamically sets the detection threshold by combining the running average of the correlator output and a spike detector. This is because without traditional AGC, the single-sample correlation peak varies drastically in magnitude. The average correlator output provides the average input power, but is additionally scaled by the power of the correlation sequence so that different sequences can be detected without adjusting the threshold. The spike detector simply raises the threshold exponentially when there is a short burst of power, thus avoiding erroneous false-positives. Existing techniques, such as [39, 71], employ a static threshold for peak detection, as they leverage AGC to consistently set the magnitude of the digital samples, and thus peak. Other reported correlator designs, e.g., [69], use the input power to set the detection threshold, however we found this approach by itself to be inadequate for *Faros*. This approach is susceptible to false-positives from power spikes without retrospective processing, and does not automatically adapt the threshold to sequences with different PAPRs.

Frequency Synchronization

To determine the carrier frequency offset (CFO), the user calculates the phase drift in the downlink synchronization sequence. This sequence consists two repetitions of the same sub-sequence; since the drift from CFO is constant, corresponding received samples in each repetition have the same phase offset. That is, for an n length sequence repeated twice to give the synchronization sequence S, $\theta(S_i, S_{i+n}) = \theta(S_j, S_{j+n})$, where θ is the phase difference between the complex samples. This is because S_i and S_{i+n} are the same symbol, thus in the absence of CFO $\theta(S_i, S_{i+n}) = 0$; with CFO there is a phase drift that is proportional to time n, which is thus constant across all i: $\theta(S_i, S_{i+n}) = drift(n)$. Therefore, we can use the following equation to compute CFO:

$$CFO = \frac{1}{2\pi \cdot n} \sum_{i=1}^{n} \theta(S_i, S_{i+n}) \tag{6.1}$$

Notably, in hardware the division by 2π is not actually performed, since the CFO is multiplied by 2π when generating the correcting complex sinusoid. Thus by selecting n to be a power of 2, the division becomes a trivial bit shift. In the presence of noise, longer sequences become more reliable, as the noise is filtered out by the averaging operation, as shown by our results in §6.6.4. While there are other techniques to compute CFO, such as the conjugate method adopted by LTE [72], *Faros* employs this technique since it enables two synchronization sequences to be simultaneously without affecting CFO recovery. Since both sequences have sub-sequences that repeat twice, the combined signal also repeats twice and can still be used to accurately calculate CFO.

To avoid frequency distortion in multipath environments, typically a cyclic prefix is prepended to the synchronization sequence. However, this cyclic prefix makes time synchronization less robust, as it can cause false positives in the correlator, since it aligns with a subset of the sequence. To avoid this, we use a cyclic *postfix*, then delay the CFO calculation accordingly, i.e., the sum in equation 6.1 starts at the length of the cyclic postfix. Note that this does not affect the correlator performance, as it operates in the time-domain. Equivalently, this can be thought of as cyclic shifting the sequence used in the detector by the length of the cyclic prefix.

6.4.2 Association Procedure

Faros enables association by: (*i*) encoding a unique base-station identifier in the beamswept synchronization sequence, i.e. the *beacon*, (*ii*) having users scan for these beacons to *select a base station*, and (*iii*) providing a *'soft' association* mechanism that allows users to quickly obtain more information about the base station over a MIMO link. We next elaborate on each of these steps.

Beacons: In *Faros*, every base station beamsweeps a synchronization sequence that encodes a locally unique identifier, called a beacon, as shown in Figure 6.2(a) and discussed in §6.3.2. This enables users to simultaneously synchronize with a base station, as well as identify it. For the sake of brevity, we assume that the base stations are coordinated so that they each have locally unique identifiers and can ensure that their beacons do not overlap in time, which prevents random access collisions and reduces pilot contamination. While there are straightforward techniques for achieving this coordination, e.g., through the backhaul or via a user, that discussion is outside the scope of this thesis.

Base Station Selection: Before associating, a user listens for at least one entire sweep interval, perhaps on multiple frequencies, to determine the IDs of all nearby base stations, as well as the average power of the beacons from each base station. Since the beacon is beamformed, its received power does not indicate the actual channel quality between the user and the base station. Thus it is important for the user to listen to beacons for an entire sweep interval to obtain a rough estimate of the signal strength from each base station, but the true SINR and channel quality, cannot be accurately determined until after association due to the beamforming inaccuracy described in §6.3.1. Furthermore, the unique identifier contained in the beacon does not convey any additional information, such as authentication, encryption, and a human-readable identifier (e.g. an SSID). Therefore, the user may soft-associate to multiple base stations in order to search for the best match.

Soft-Association: Since *Faros* beacons only contain a unique identifier, we additionally provide a mechanism called *soft-association* which enables users to gather more information over the CSI mode. Traditional control channel designs broadcast information about the base station in their beacons. For example, 802.11 beacons include the BSSID, SSID, modulation rate, encryption information, and more. This information is essential for users to determine if they want to, or even can, connect to the base station. Moreover, the users need to be able to judge their channel quality to the base station, which can only be done in CSI mode.

Guided by our first key insight, that as much control information as possible should be sent over the more efficient MU-MIMO channel, *Faros* provides soft-association to enable users to quickly establish a MIMO link with the base station to efficiently exchange control channel information. To perform a soft-association, users must first synchronize with the base station by successfully decoding a beacon, then send a pilot in one of the slots reserved for random access, as discussed below. Once the base station successfully receives the pilot it has CSI for that user, which it leverages to open a MIMO link and convey the remaining control channel information. If the user proceeds with a full association, based on authorization, link quality, etc., the base station schedules the user dedicated pilot slots and a unique paging sequence to maintain the link. Otherwise, the user continues to scan for and soft-associate to other base stations.

Var	Description	Overhead	Description
L	Sequence Length	C	Channel Utilization
B	Bandwidth	D_A	Association Delay
F	Frame Duration	D_R	Random Access Delay
N	# of beams		

$$C = \frac{L/B}{F} \qquad D_A = \frac{N \cdot F}{2} \qquad D_R = \frac{F}{2}$$

L	B	F	N	C	D_A	D_R
128	$20\,MHz$	$15\,ms$	100	0.043%	$750\,ms$	$7.5\,ms$
128	$40\,MHz$	$1\,ms$	100	0.32%	$50\,ms$	$0.5\,ms$
256	$20\,MHz$	$10\,ms$	100	0.128%	$500\,ms$	$5\,ms$
256	$20\,MHz$	$5\,ms$	500	0.256%	$1250\,ms$	$2.5\,ms$
512	$40\,MHz$	$2\,ms$	1000	0.64%	$1000\,ms$	$1\,ms$
1024	$80\,MHz$	$1\,ms$	4000	1.28%	$2000\,ms$	$0.5\,ms$

Table 6.2 : Analysis of *Faros*' beacon overhead. *Top:* Variable descriptions. *Middle:* Equations used for analysis. *Bottom:* Expected value of the worst-case overheads of the simplest version of *Faros* given various realistic system parameters.

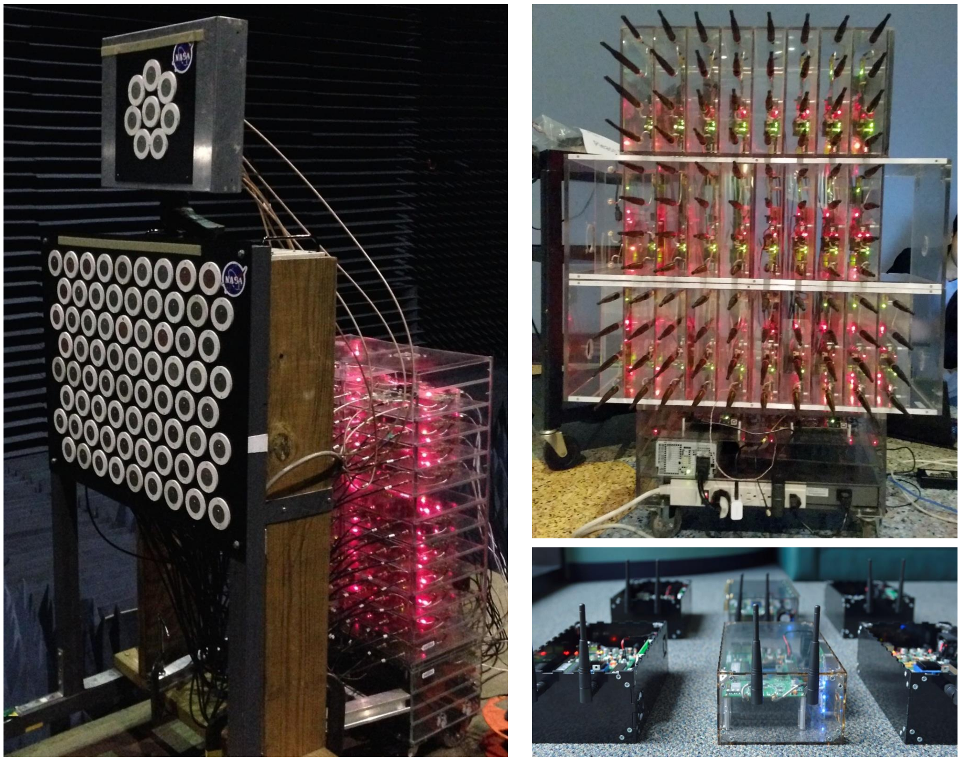


Figure 6.3 : Our prototype, Argos. *Left:* 80-antenna array in an anechoic chamber. *Top Right:* 104-antenna array in an indoor environment. *Bottom Right:* ArgosMobile user devices.

6.4.3 Collecting CSI

After each beacon, all active users send uplink pilots in their scheduled slots which the base station leverages to collect CSI. It is best to think of the CSI collection phase as a number of time-frequency-code resource slots that can be arbitrarily assigned to users, with some resource slots dedicated to random access, including association requests and paging responses. Users which send reference signals in a given resource element gain spatial resource elements in the corresponding time and frequency coherence interval for both the uplink and downlink phases. That is, any given reference symbol provides an estimation that is valid both for the coherence time interval, as well as a wider frequency coherence interval. As noted in §6.3.2, *Faros* assigns longer pilot slots to users that have worse channels in order to improve CSI accuracy.

6.4.4 Random Access

Faros facilitates random access by reserving pilot slots at the beginning of each channel estimation phase, as shown by Figure 6.2(b). To initiate a connection users simply send an uplink pilot in one of these pilot slots. For the user to send in the correct pilot slots, without interfering with other users, it must have successfully received a beacon and thus established synchronization. The base station uses this pilot to estimate the user's channel, as well as timing advance, and create a highly efficient MU-MIMO link to the user. As guided by our first key insight, this link is then used to convey all remaining control channel information, including modulation rates and pilot scheduling, as well as maintain/improve synchronization.

LTE already provides a compelling random access solution which fits well within the *Faros* design, with the exception that *Faros* allows for longer length sequences to be employed to finely tune the gain gap. Thus we defer to [73] to fully describe the

LTE random access scheme, including collision detection and avoidance, as well as timing advance, which we employ in *Faros*.

Multi-antenna Uplink Detection

To detect uplink random access requests with the full diversity and beamforming gains of the array, *Faros* employs a novel open-loop maximum-ratio combining method. Simply detecting the random access request signals the base station that a user needs to be scheduled, however the request is also structured to provide accurate timing information, e.g., for establishing timing advance, and enables the base station to collect CSI for beamforming. Traditional open-loop methods, e.g. Hadamard, are computationally expensive because they require M sets of beamweights to be applied and M correlations (one for each beam), and still do not provide the full array gains. Traditional closed-loop methods, e.g. conjugate, will actually remove timing information when applied without knowledge of when the uplink symbol began. Thus the random access request consists of a sequence, e.g. Zadoff-Chu, repeated twice, where the first repetition is used to remove the phase offset of the second, then the streams from all base station antennas can be coherently combined:

$$\sum_{m=1}^{M} Corr(R_0^m, S) \times Corr(R_n^m, S)^* \tag{6.2}$$

Where $Corr(R, S) = \sum_{i=l}^{l+n}(R_{t-i} \cdot S_i^*)\ for\ l = [1, t]$ is the cross correlation of the received signal, R, with the pre-known sequence, S of length n, and R_i^m denotes the received signal at antenna m with offset i (so R_n^m is the start of the second repetition). Unlike the user, which needs to perform a streaming correlation since it has no timing information, the base station needs to only perform t correlations, where t corresponds to the maximum propagation delay, since the user has time synchronization. As in

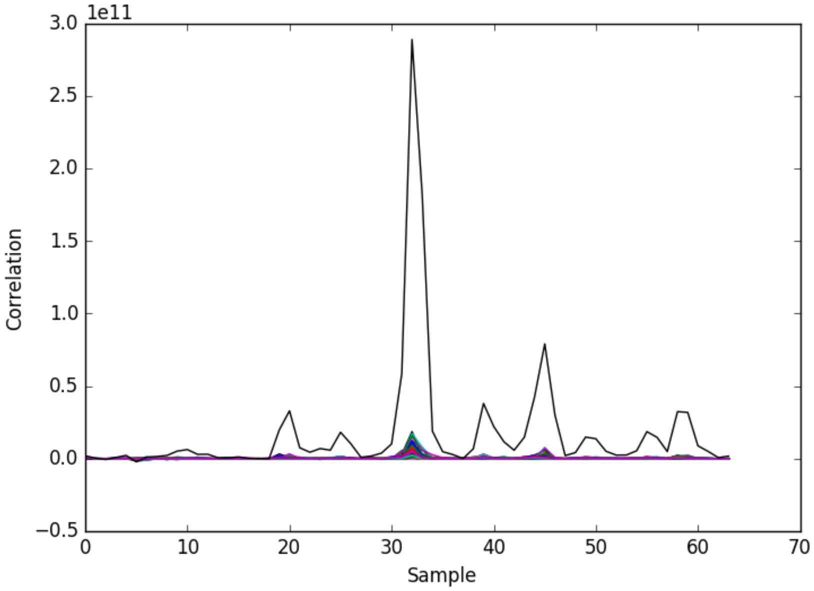


Figure 6.4 : Example of how *Faros* improves uplink detection by leveraging all of the antennas on real indoor channel traces. *Left* High-SNR regime shows the correlation peaks summing constructively. *Right* Low-SNR regime shows the M-fold average noise rejection, drastically improving performance at the cell-edge.

§6.4.1 a threshold is used for detection, and the location of the peak determines the timing offset. This provides the full diversity and beamforming gain of the array, M, *without* removing the critical timing information. This technique can also be used to detect multiple users simultaneously, exactly the same as in the LTE standard, though *Faros* requires an additional step of separating CSI for each user. Notably, while Carrier Frequency Offset (CFO) can affect performance, the user must have already detected a beacon, and thus established time-frequency synchronization with the base station, before sending an uplink random access request, so the CFO should be negligible and the remaining time offset is propagation time, which is used to set the timing-advance.

Since the random access request has a very high coding gain to facilitate CSI estimation at each base station antenna, this method should not be required for detection. However, it does provide additional reliability, particularly in spatially selective environments. Given that random access requests generally expected to be infrequent, this technique is computationally negligible. Moreover, in the Argos system architecture, this method can be employed in a distributed fashion using the idle beamformers on each radio module, which makes it essentially free from a hardware resource perspective.

6.4.5 Paging

Faros enables many-antenna base stations to reliably and *quickly* page users across their entire coverage area. To accomplish this, *Faros* applies the beamsweeping and coding gains described in §6.3; unfortunately, unlike synchronization and association, paging is not delay tolerant. Thus *Faros* leverages the users last known location to substantially reduce the delay from beamsweeping.

Upon association, the base station assigns each user a unique paging sequence. This paging sequence is constructed and transmitted almost identically to the beacon. That is, it is chosen from the same codebook as the beacon to ensure orthogonality, as well as repeated twice to facilitate time-frequency synchronization. To page a user the *Faros* base station beamsweeps their unique paging sequence with the beacon at the beginning of each frame, but on a separate beam, as shown in Figure 6.2(a). This additional spatial separation between the beacon and paging sequence helps improve the detection of either, as it reduces the inter-sequence interference. To detect the paging sequence, users perform the same synchronization correlation used for the beacon, described in §6.4.1. Successful detection similarly provides the user with synchronization, however in the case of a paging sequence the user immediately sends an uplink pilot in the dedicated random access pilot slot. This allows the base station to estimate CSI and begin MIMO communication.

One key challenge facing *Faros* is that while association and synchronization are not time-sensitive, the delay from beamsweeping is likely unacceptable for paging, e.g., up to 2 s in Table 6.2. To solve this challenge, *Faros leverages knowledge of the user's prior location to guide the beamsweep*, even in our naïve implementation this sped up paging by 400%, as demonstrated in §6.6.3. Note that leveraging the users last known location can only improve expected paging delay, as the sweep continues until the user is paged.

Link Maintenance: Additionally, or alternatively, users will periodically send a random access request to the base station. This serves the multi-purpose of maintaining the association, checking for missed page requests, and updating the users' last known location at the base station to assist with efficient paging and inter-base station handovers.

6.4.6 Overhead Analysis

By design, *Faros* has a small, if not negligible, overhead. This overhead can be measured by four metrics: (*i*) total channel overhead, (*ii*) association delay, (*iii*) random access delay, and (*iv*) paging delay. Table 6.2 provides the equations for determining these overheads, then provide example values for reasonable system configurations. For this analysis we assume that frames are sent continuously, with the beacon at the beginning of each frame similar to a scheduled MAC. Since the expected paging delay is dependent on the paging scheme, we discuss its real-world performance using a naïve scheme in §6.6.3, however it is upper-bounded by the association delay as that is how long it takes to perform a full beam-sweep.

It is important to realize that active users *do not* need to receive valid beacons to maintain synchronization, as it is maintained in the CSI downlink control phase. Inactive, but associated users can also maintain synchronization by listening for beacons and paging signals. The duration that time-frequency synchronization is valid depends on the accuracy of the oscillators, frame design, e.g., cyclic prefix, as well as fluctuations in temperature. Given the typical accuracy of oscillators in WiFi and LTE devices, and according to our measurements, the synchronization is usually valid for 100s of ms, but this can be determined on a per-system basis [74]. As such, beacons are only needed for association, and thus the sweep interval can be adjusted accordingly. We also find these overheads are very easy to tune by changing the system parameters. Note that per Table 6.2, *Faros* can support thousands of antennas with less than 2% overhead, at the cost of slightly increased association delay at the cell edges.

6.5 Implementation

We implement *Faros* on ArgosV2 [55], a prototype of many-antenna MU-MIMO base station that consists of an array of 27 WARP boards [39], driving 108 antennas, and 5 battery powered WARP-based ArgosMobiles that can be controlled wirelessly through a WiFi bridge, as shown in Figure 6.3. Both the implementations of *Faros* and Argos can support many times more antennas and users; the reported implementation is only limited by the number of WARP boards available to us. To the best of our knowledge, when first reported this was the largest many-antenna MU-MIMO base with public results.

Our implementation of *Faros* serves as the basis of Argos' realtime design, and involves development across all layers of the Argos architecture. To enable realtime operation we designed multiple custom Xilinx System Generator IP cores for both the base station and mobile nodes' Virtex 6 FPGA. The most computationally complex IP core we developed for the mobile nodes is the streaming correlator. The correlator enables realtime detection of beacon and paging codes simultaneously, can be dynamically reprogrammed with different sequences, and supports multiple rates and lengths. For the base station our most significant IP core is the MU-MIMO precoder, which we modified to support beamsweeping, as well as selecting and sending multiple paging sequences simultaneously on different beams. While System Generator IP cores are built with a graphical model and do not directly have lines of code, we use the Xilinx xBlock scripting language to dynamically build a significant portion of them, which constitutes over 4,000 lines of code. These IP cores are integrated with peripherals and other IP cores, including a Microblaze soft-core that is programmed with over 1,000 lines of embedded C.

We implement two versions of the central controller, one in Matlab, and one in

Python, both of which are over 2,000 lines of code. The Matlab version facilitates flexible non-realtime experiments with rapid analysis, whereas the Python version supports realtime operation, including fully mobile channel estimation with a time resolution up to 200 μs. Our implementation of *Faros* is extremely versatile; it can be compiled to support detecting any code length, given adequate FPGA resources, and can support any beamforming technique by simply reloading the beamsweep buffers with the corresponding precomputed $\mathbf{B}$.

Open-loop beamsweeping: Our implementation uses Hadamard beamweights [75] for beamsweeping for the following reasons. First, they use a minimal number of weights to provide a complete, perfectly orthogonal, basis; this enables a full diversity gain and provides complete spatial coverage with the minimal amount of overhead. Second, they have a perfect peak-to-average power ratio (PAPR) of 1, which allows the antennas to use their full potential transmit power. Finally, calculating Hadamard beamweights does not require any knowledge of the antenna aperture or environment, enabling rapid deployment without calibration or environmental considerations.

Coding: The implementation uses Kasami sequences for the downlink coding. Kasami sequences [76] provide very good detection performance and have low, bounded, streaming correlation both with themselves and the other orthogonal sequences. This allows them to be reliably detected *without* time synchronization, as a streaming correlation on other sequences could produce peaks, and thus false positives, which is important since they are used for time synchronization. Moreover, they provide a large number of orthogonal sequences, e.g., 4096 for a length 256 Kasami sequence, which enables co-located users and base stations to be uniquely identified.

The implementation uses Zadoff-Chu sequences [77, 78] for the uplink channel estimation coding for the following reasons: First, they have a constant amplitude and thus have a perfect PAPR. Second, they can be used to detect multiple users' random access request simultaneously, along with each users' path delay to estimate timing advance, with small computational overhead. This is very similar to LTE's random access preamble [72]. However, in our design we allow variable length sequences in order to match gain requirements, as well as use the sequence for CSI estimation.

Thresholding and Variability: *Faros* leverages a realtime streaming time-domain correlator for the beacon, paging, and synchronization, which creates a very strong single-sample peak when the correct sequence is detected. As such, the performance range and accuracy is highly dependent on the detection *threshold*. This threshold is well understood theoretically with regard to false-positive and false-negative performance, and as such we defer to [69] for a more thorough analysis. Since we do not perform gain control for the beacon or paging code we must set this threshold dynamically based on the input power, as well as increase it during power surges to avoid false-positives. This dynamic threshold in *Faros* can be scaled by a constant via software; for the experiments we set the threshold somewhat aggressively so that it is close to impossible to receive a false positive, as we didn't across the 100,000s of synchronization sequences we sent during our experimentation. This threshold could be further optimized to increase range, particularly with mechanisms to deal with false positives.

6.6 Real-World Performance

We evaluate the performance of *Faros* in bridging the gain gap in real-world topologies. We examine the fully functioning system, and evaluate its performance regarding synchronization, beacons, and paging in diverse environments. Our results demonstrate that *Faros* can extend the no-CSI mode range by over 40 dB when compared

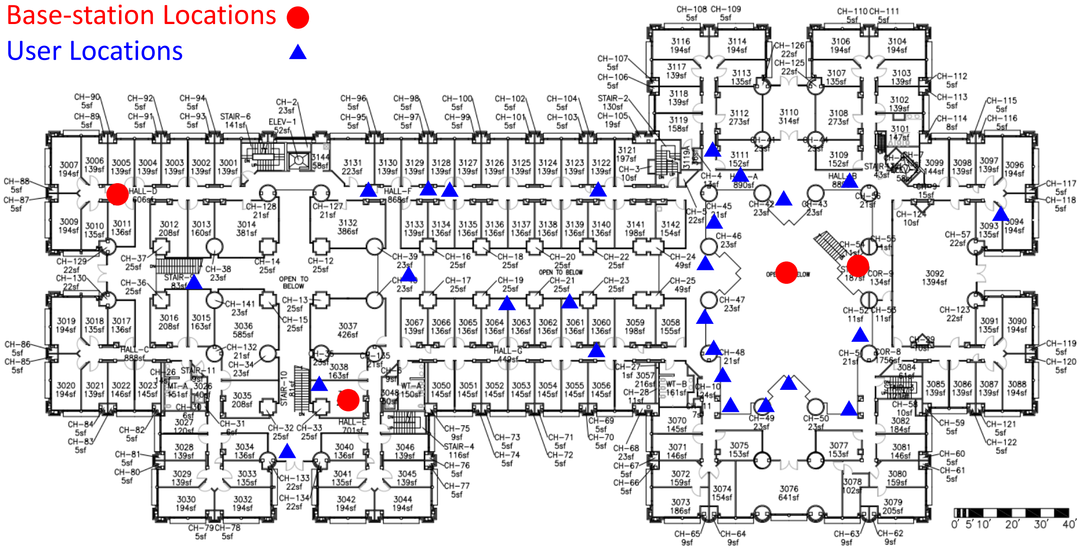


Figure 6.5 : Floorplan depicting example locations of indoor measurements. Both the users and base station locations spanned three floors of elevation.

to traditional control channels. Furthermore, we find that leveraging knowledge of the users previous location can improve paging delay by 400%, and that *Faros* can reliably correct CFO of over 10 kHz. We first describe our experimental setup, then look at how each of the components performs individually.

6.6.1 Experimental Setup

We test the performance of the reported *Faros* implementation in 100 discrete user locations at varying distances from the base station in indoor environments and an anechoic chamber, using five ArgosMobiles simultaneously. Additionally, we perform an outdoor range and mobility test, presented in §6.6.2.

Antenna Configurations: Due to hardware availability, and to test the performance of different antennas, we employed *Faros* with three separate antenna configurations: (i) In the anechoic chamber with 80 directional 6 dBi patch antennas, (ii) indoors and outdoors with 104 omnidirectional 3 dBi monopole antennas, and

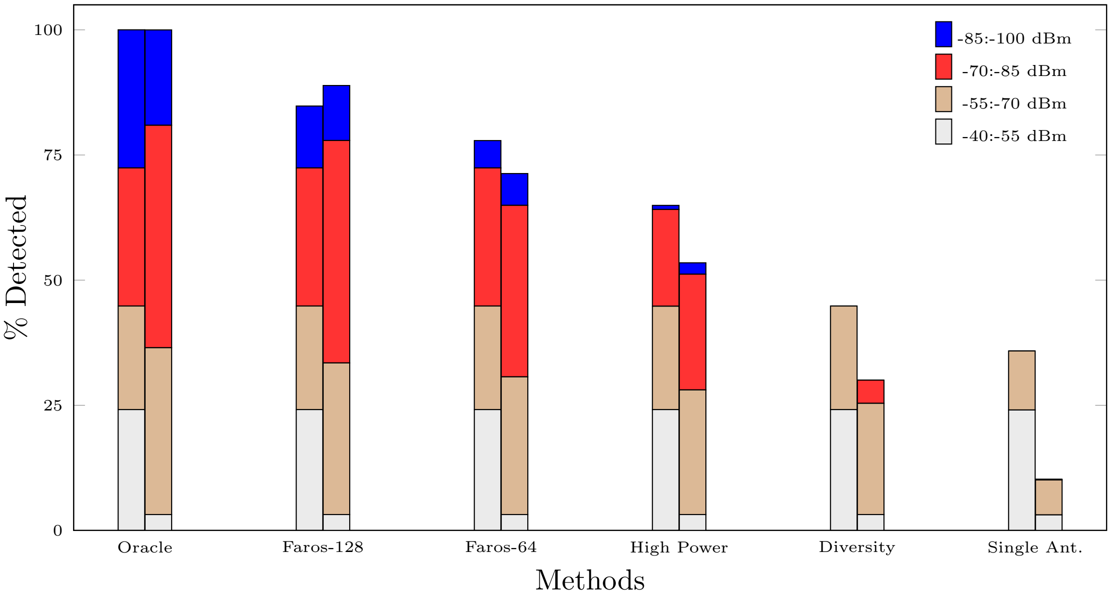


Figure 6.6 : Beacon detection performance across all 32 anechoic chamber (left) and 68 indoor (right) experiment locations. *Oracle* denotes an oracle that detects every beacon sent.

(*iii*) indoors with 108 of the same omnidirectional antennas. In all configurations the users also leveraged the 3 dBi omnidirectional antennas.

Power Settings: In the downlink we use a power of approximately -12 dBm per antenna and in the uplink we use 10 dBm. This downlink power is chosen since it results in a ~10 mW total power and an EIRP of up to 1 W, which is the FCC limit. Our prototype is capable of over 10 W total power, and EIRPs exceeding 1 kW, which is only appropriate in licensed bands. For high-power single antenna mode we use the approximate combined power of all of the base station antennas, which is 8 dBm.

Environments and Range: As Figures 6.6 and 6.7 depict, we selected locations at increasing distances until the beacons couldn't be detected, providing a fairly uniform selection of signal strengths. In indoor locations this required spacing the

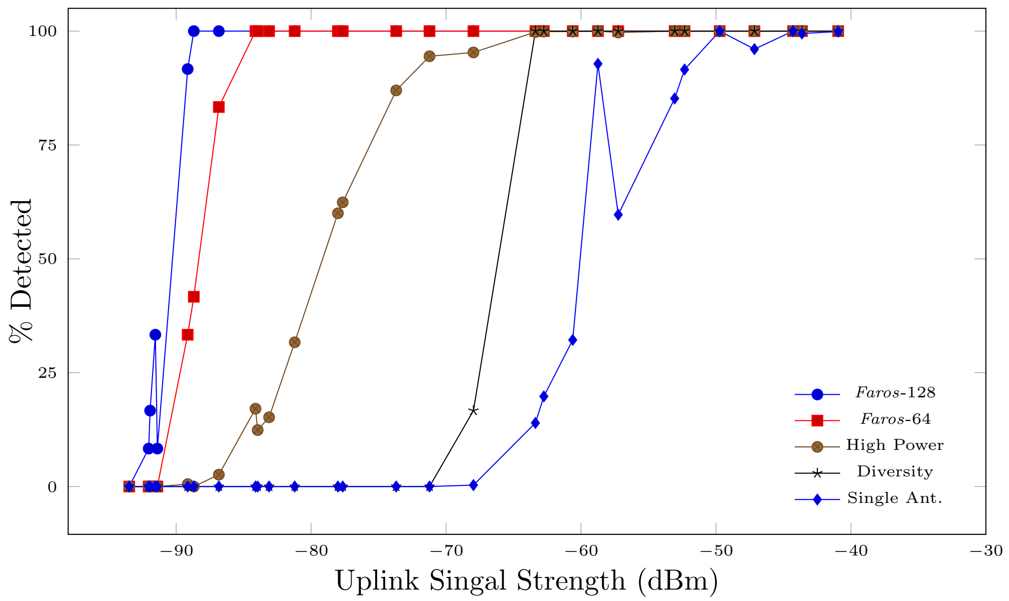


Figure 6.7 : Beacon detection performance vs. uplink RSSI (range) for *Faros* in an anechoic chamber. *Faros* outperforms traditional by over 40 dB. Number indicates beacon length.

users at up to 80 m away, across 3 floors of elevation, as illustrated by the sample of locations presented in Figure 6.5; in outdoor line-of-sight locations it was over 250 m away. For the anechoic chamber experiments users were spaced up to 15 m away from the base station, and we used variable attenuators with up to 60 dB of suppression to simulate increased distance.

Measurement: At each location we test the *Faros* control channel system over a 20 MHz bandwidth at 2.4 GHz and analyze the performance with regard to the accurate detection of the beacon, paging signal, and uplink pilot, which demonstrate *Faros*' performance in no-CSI mode. As a control, we additionally send an unbeamformed beacon and paging signal from each base-station antenna, i.e. a "beamsweep" using the identity matrix, in both low and high-power modes using a 64 length code to compare the performance with traditional single antenna systems and the naïve high-

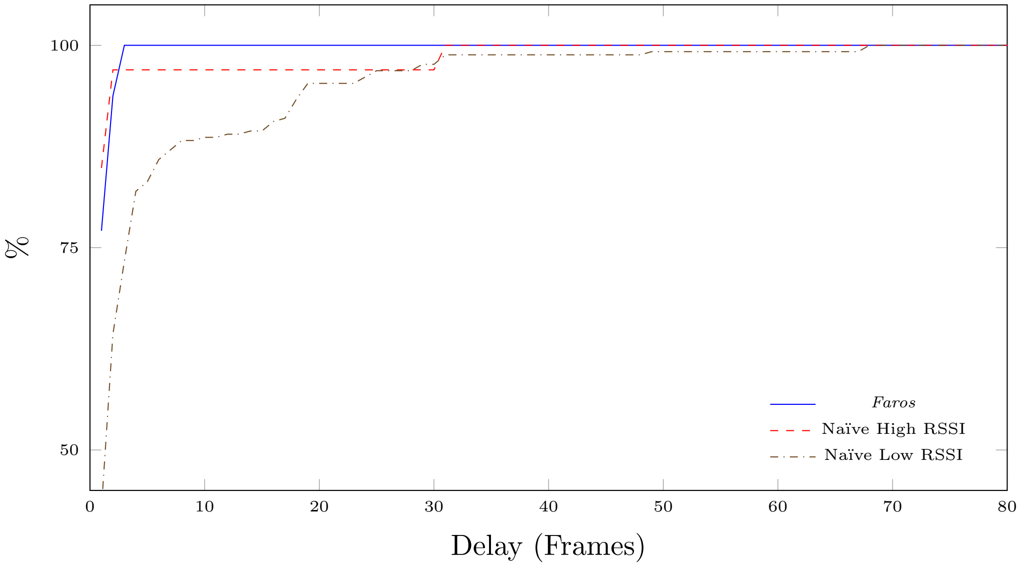


Figure 6.8 : Cumulative distribution functions of paging delay. The naïve method does not use location data to sweep. *Faros* improves mean paging delay by 400% at low RSSIs.

power solution. While *Faros* is capable of running in realtime, we briefly pause after every beam in order to collect performance statistics from the nodes, such as successful detections, false positives, and received signal strength indicators (RSSIs). Because of this measurement delay, these experiments were conducted without mobility, in relatively stationary channels. We use these results to analyze the performance of *Faros*' beacon, paging, and CSI collection vs. traditional methods, which we present below. Additionally we setup a controlled experiment to test the performance of our CFO estimator, presented in §6.6.4.

6.6.2 Beacon Performance

Figures 6.6 and 6.7 show the probability of successfully receiving the base station's beacon, i.e., the synchronization sequence encoded with the base-station ID, with various configuration parameters. We compare single-antenna transmission, both

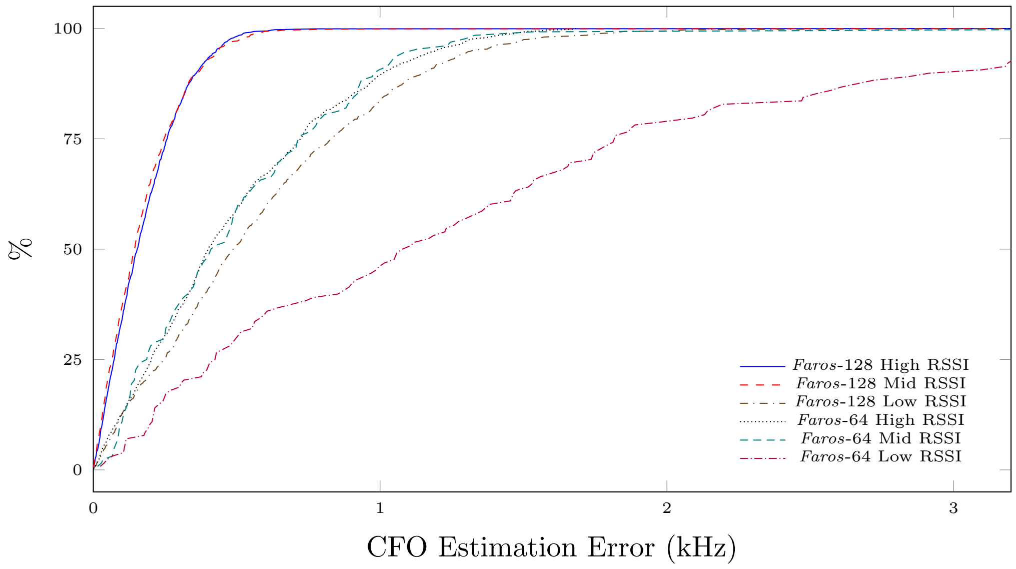


Figure 6.9 : Cumulative distribution functions of CFO estimation error with various sequence lengths and RSSIs. *Faros* provides frequency synchronization within 800 Hz at up to -75 dBm.

high power (*High Power*) and low power (*Single Ant.*), diversity, and *Faros* using code lengths of 64 and 128. In the single antenna diversity mode (*Diversity*) the base station rotates which antenna is transmitting, thus exploiting the full diversity of the array; this is equivalent to *Faros* using the identity matrix for beamsweeping.

The figures sort the results based on the average uplink CSI signal strength across all base-station antennas for the given location, which is an approximation of distance and a fair metric for coverage area. We note that downlink RSSI is not a good metric, since it varies per-beam. Distance is not a good metric since scatterers can significantly alter signal strength. Clearly, changing uplink transmission power will simply shift the same plot either left or right, which indicates how code length and both uplink and downlink transmission powers should be balanced in a real system.

The results across all locations are shown in Figure 6.6, with separate bars for the 36 anechoic chamber locations and 64 indoor, including 104- and 108-antenna, locations. We see that in indoor locations *Faros* is able to reliably serve over 8.8 times more locations than the traditional control channel, and 1.6 times more than a single high power antenna. Even when users have over a -70 dBm average RSSI to the base station, they miss almost 25% of the beacons sent with the high-power single-antenna scheme. This is due to multipath; in some locations, even fairly close, two paths will destructively interfere and create a null, which is not easily overcome with additional signal strength. While the diversity scheme performs better than the single antenna, it is still unable to reliably receive many beacons where users have lower than -70 dBm uplink RSSI. This illustrates the necessity of *Faros*, which leverages both the power and diversity of the entire array, in many-antenna MU-MIMO systems.

The results from the anechoic chamber are shown in Figure 6.7. Since there is no multipath in the anechoic chamber, the detection rate of each technique is very closely related to RSSI, thus these results accurately demonstrate the relative performance of each technique. We find that *Faros* is able to outperform a single-antenna scheme by over 40 dB, and the high-power scheme by 20 dB.

Range and Mobility Performance

To demonstrate the realtime capability of *Faros*, as well as test its range and mobility performance, we performed an outdoor experiment where we ran *Faros* at full speed. Unfortunately, the previous tests required us to pause the experiments after every beacon or paging signal was transmitted and collect measurements, which prevented realtime operation. For this experiment we had the base station continuously beam-sweep the beacon at a frame rate of one beam per 10 ms, then had users move away from the base station at a walking pace. In line-of-sight the users performed reliably, and concurrently, at over 250 meters at multiple angles from the base station, and only began to lose reliability the users had to move behind buildings due to space constraints.

6.6.3 Paging Performance

To demonstrate *Faros*' ability to leverage location information to accelerate paging, we tested a simple scheme which guided the paging sweep based on the intended user's last location. These experiments were performed on the 108-antenna base station configuration in the last 44 locations. In the prior locations we had employed RSSI to guide the sweep, but realized that due to multipath distortion this was not the best detection performance metric, since the time-domain correlation essentially filters individual paths. Instead, we paged mobiles based on each beam's detectability, which is determined by the correlation magnitude to threshold ratio.

We find that *Faros* was able to successfully page 94% of users by the second frame, compared to only 70% without leveraging the user location, as shown in Figure 6.8. When users are near the base station they receive the majority of the beams in a sweep, and thus optimizing based on their location does not provide much benefit, as shown by the low RSSI plot. However, we still see the paging delay reduced from an average of 4.8 frames to 1.2 frames, an improvement of 4 fold, and a worst-case improvement of 68 frames to 3 frames. This system is very naïve, and is intended to demonstrate *Faros*' ability to leverage spatial information to drastically improve the performance of the control channel without additional overhead.

6.6.4 CFO Correction Performance

While successful detection of a beacon or paging sequence inherently provides time-frequency synchronization, to more accurately test the accuracy of our realtime CFO correction we setup a more controlled experiment. We shared a reference clock between the base station and user, effectively removing CFO, and placed the user at 0.5 m from the base station. Then we *induced* a controlled CFO in our beacon sequence by multiplying it with a complex sinusoid ranging from -10 kHz to 10 kHz. To measure the performance vs. coding gain and SNR, we sent beacons of length 64 and 128, as well as used attenuators on the base station to reduce the transmission power from -12 dBm to -42 dBm. These attenuations resulted in the user receiving roughly -60 dBm (*High*), -75 dBm (*Mid*), and -90 dBm (*Low*) RSSIs. We present the cumulative distribution of the error magnitude of our CFO estimates in Figure 6.9. For clarity, these results are derived from a single estimation, however multiple estimates can be employed to reduce the error by an order of magnitude, as shown in [74].

We find that with mid and high RSSI *Faros* is always able to correct CFO within 0.8 kHz using a 128-length beacon, and within 1.3 kHz using a 64-length beacon. This estimation error is sufficient to not restrict the performance of an LTE system, [79]. In the low RSSI regime we see that the 64-length beacon begins to perform poorly, and is only able to correct 80% of the beacons to within 2 kHz error. In contrast, the 128-length beacon with low RSSI is performs similarly to the high RSSI 64-length, which indicates extending the beacon length could further reduce CFO estimation error. The amount of induced CFO did not affect accuracy, and thus is not shown separately.

6.7 Discussion

Broader use of Faros: Our original goal for *Faros* is to provide a very efficient control channel for many-antenna base stations. More fundamentally however, it represents an interesting paradigm that provides fine-grained control over time, code, and spatial resources, enabling previously impossible optimizations both within a single base station, and across the network. *Faros* allows base stations to leverage existing information, such as users' last known location, traffic patterns, and environmental properties to intelligently optimize timing, coding gains, and spatial coverage. Moreover these same properties can be used to further extend the range of the cell in sparse networks, restrict coverage area, carefully tune interference, or dynamically incorporate more antennas to increase the system capacity of a given base station.

MAC and standards: So far we have intentionally avoided discussing the MAC, as the conceptual *Faros* design is MAC-agnostic. The primary requirement of *Faros* is a short regularly scheduled downlink phase in order to perform the beamsweep and paging; and both scheduled and CSMA MACs have this. However, we do not intend *Faros* to be a plug-n-play solution for either LTE or 802.11: both standards must be revised to integrate *Faros*. Applying *Faros* to a scheduled MAC is more intuitive, and likely more efficient: the phases depicted in Figure 6.2 simply have to continuously repeat, though not necessarily in that order. Schemes to adapt *Faros* to CSMA are also fairly straightforward and we provide one example below.

Design sketch of Faros in 802.11: In an 802.11-like CSMA MAC, the beacon would need to be replaced by the beamformed *Faros* beacon immediately followed by the CSI collection phase, including the dedicated random access and association slots. Since 802.11 typically supports only a small number of users with relatively low mobility, each user could have a dedicated CSI slot, which they use at every beacon interval when they are active. The AP can page inactive users during the beacon phase, making them become active, or, if the channel is idle, the AP could page users asynchronously, prompting them to send a pilot immediately. Since these mechanisms allow the AP to maintain accurate CSI for the users, the downlink phase is straightforward: when the channel is idle the AP simply sends a MU-MIMO transmission to the intended users. Uplink MU-MIMO is difficult to efficiently coordinate in CSMA, which, combined with typical asymmetric data requirements, is why current 802.11 standards do not support uplink MU-MIMO. However, one naïve solution would be to allow the users to indicate an uplink request during the CSI collection phase. The AP could then respond with a "clear-to-send" to selected users over the MU-MIMO channel. Of course, to reduce latency users would not have to wait for the beacon to send a single-user uplink packet.

6.8 Related Work

To the best of our knowledge, *Faros* is the first reported control channel design that effectively addresses the gain gap between the CSI and no-CSI modes for many-antenna MU-MIMO systems. Nevertheless, various previous works are related to *Faros* in terms of both problem and solution. The challenge of control channel design for many-antenna MU-MIMO is well-known. The authors of [80] suggest utilizing space-time block coding for the control channel, but do not address the gain gap or suggest a design. The authors of [81] discuss control channel operation from a purely theoretical and feasibility perspective, which is complementary to our work. However, its assumption that "the only reasonable transmit strategy is to spread the power omnidirectionally" is questionable: *Faros* beamforms the control and provides a working counter-example. It also assumes that the total base-station power can be sent omnidirectionally in the first place, e.g., there is no peak-power per antenna constraint, which is incorrect for real systems, as discussed in §6.2. Another recent work from Samsung mentions the control channel briefly, but suggests the solution is to carefully create a wide open-loop beam using all of the antenna elements [27]. This approach requires careful calibration of the antenna elements, is environment and deployment specific, and, more importantly, does not completely serve the full potential coverage area of the base station.

802.11ad suffers from a related gain gap and employs a beamsweeping mechanism with multi-level codebooks to initiate communication from *both* sides of the link. Because 802.11ad does not employ MU-MIMO but phased arrays, its gain gap is fundamentally different and scales with less than M. Moreover, the contiguous Sector Level Sweep (SLS) that 802.11ad performs for synchronization and discovery is naïve, unscalable, and highly inefficient. An 802.11ad SLS with 128 elements can take over 1.5 ms [82], whereas a comparable *Faros* beacon would take less than 150 ns. This indicates that 802.11ad and other mm-wave technology could benefit substantially by incorporating design principles from *Faros*, particularly as they adopt MU-MIMO.

Faros' use of Kasami sequences to send a small portion of the control channel information is inspired by [69], which uses time-domain BPSK modulated Kasami sequences to encode control information in the preamble of 802.11 packets. However, *Faros* addresses an entirely different problem: the gain gap in many-antenna MU-MIMO. As such, it employs different techniques and contributes an entire from-scratch control channel design. Other recent works, such as [83], have also used similar sequences for other purposes including control messages and power reduction.

Like most modern digital wireless systems, *Faros*' synchronization is based on the seminal works in [84, 85]. More recently, some research has focused on over-the-air time-frequency synchronization in distributed antenna systems, including [56,86]. However, these works deal with the distributed antennas, not between the distributed system and users. As such, they do not address the synchronization range gap that emerges with multiple antennas on a single base station, or the challenge of paging in such a system. Since these distributed systems require backhaul, this synchronization can also similarly be solved with CPRI, [87], or PTP and SyncE, as employed by CERN's WhiteRabbit [88].

Open-loop beamforming techniques have been thoroughly researched. While we do not advocate for a specific technique in this work, our experiments leveraged Hadamard matrices for the beamweights; recent work in [75] covers the performance of Hadamard beamforming more thoroughly. Fourier transform based beamforming is a classic technique, however it requires precise antenna calibration in order to be effective, as discussed in [13,61,63–65].

Chapter 7

Channel Measurements

To better understand many-antenna MU-MIMO channels in the real-world, we implemented a realtime wideband many-antenna MU-MIMO channel measurement system. Built on the ArgosV2 [55] platform and Faros control channel design [89], this system enables very reliable high time-frequency resolution measurements, supporting sub-millisecond sounding intervals with 20 MHz bandwidth, across the UHF, 2.4 GHz, and 5 GHz bands. We leveraged this system to conduct one of the most extensive and diverse mobile MU-MIMO measurements campaign ever reported, already containing over 1 billion channel measurements on more than 20 topologies, and continuing to expand. These topologies include LOS and NLOS scenarios in both indoor and outdoor environments with various degrees of mobility and multipath. Additionally we constructed an open-source Python channel analysis toolbox to study the fundamental properties of many-antenna MU-MIMO channels.

Our analysis of these measurements provides real-world reference points and reveals important trends that have a significant impact on MU-MIMO system design. While many of these results are intuitive, this data and analysis provide a ground-truth across a diverse number of environments, mobilities, and frequencies. In this chapter we focus on a few key points. (i) Even under pedestrian mobility, channel stability is very low, e.g., we measured channel coherence times of 16 ms and 7 ms for 2.4 GHz and 5 GHz, respectively, in an indoor NLOS environment. This has important implications for the system design in next-generation wireless systems, as collecting CSI every 10 ms can result in a loss of over 50% of the achievable rate. (*ii*) Multipath, frequency, number of base-station antennas, and rotational mobility significantly affect channel stability, causing up to orders of magnitude change in channel coherence. This suggests that next-generation MU-MIMO systems will need much finer-grained customization than current MU-MIMO systems to operate efficiently under diverse environments, topologies, and levels of mobility. (*iii*) We found that stationary users have indefinitely stable channels, with mobility caused by pedestrians in the environment itself having minor, and ephemeral, impact. Therefore in fixed topologies MU-MIMO systems can virtually eliminate channel sounding overhead. (*iv*) Our measurements also demonstrate that in realistic scenarios channel stability is *bimodal*, that is, users are either mobile or stationary, and thus either have very unstable or very stable channels, respectively. This presents a significant opportunity to reduce system overhead with dynamic channel sounding protocols. (*v*) Finally, we found that system capacity can fluctuate an order of magnitude faster than the channel coherence. Thus channel coherence, as measured by expected correlation, is *not* a good indicator of the channel resounding interval, though it is often used synonymously, e.g., [13, 18, 90]. Next-generation many-antenna MU-MIMO systems must carefully take into account the characteristics of real-world channels, particularly mobility, to achieve their potential performance gains.

We have released the measurement system, channel analysis toolbox, and channel measurements online [14], with the hope that they will help guide next generation MU-MIMO system design and analysis.

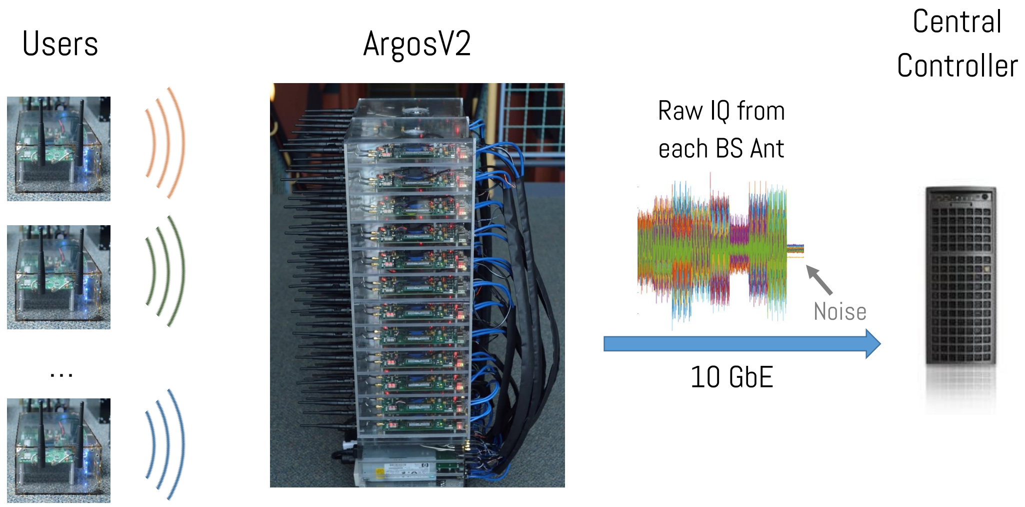


Figure 7.1 : Overview of the channel measurement system design. At the beginning of each frame the base station sends a beacon to synchronize the users. Each user then sends orthogonal pilots, which each base-station WARP node records. At the end of the pilot phase, base-station nodes report the raw pilot IQ samples to the Central Controller, which records them to an HDF5 file.

7.1 System Design

We designed and implemented a realtime wideband many-antenna MU-MIMO channel measurement system that supports high time-frequency resolution across the UHF, 2.4 GHz, and 5 GHz bands. We built this system on the ArgosV2 platform [55], based on WARPv3 [39], and leveraged the Faros control channel design [89] to provide time-frequency synchronization with the users and collect CSI. To support UHF, we ported Argos and Faros to the WARP-based WURC platform [1], which involved converting and merging code, standardizing control interfaces, as well as implementing an IQ only Automatic Gain Control (AGC), as the WURC radio does not have an analog Received Signal Strength Indicator (RSSI) indicator. Using WURC-enabled battery-powered ArgosMobiles as users, this measurement system can easily be con-

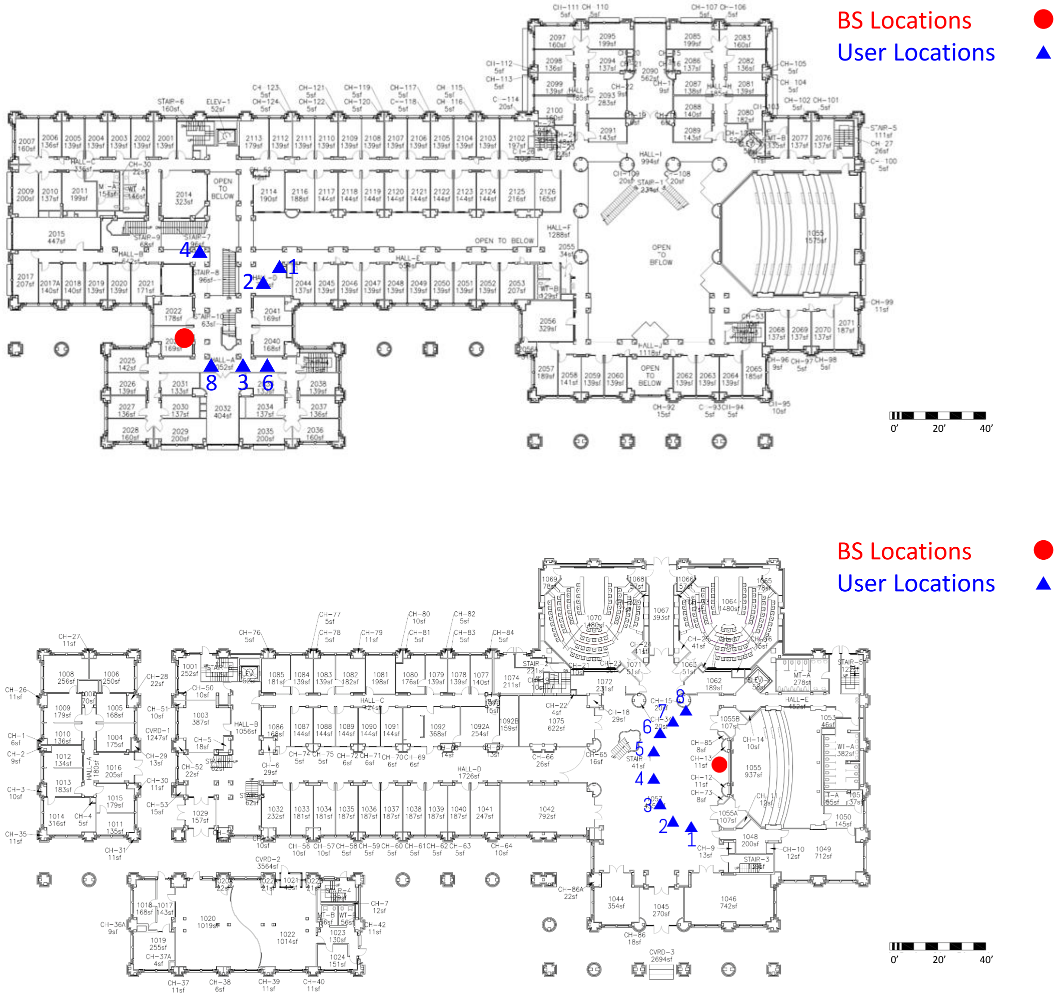


Figure 7.2 : Maps of each experimental setup are included online with each trace. *(Left)* Picture of outdoor environment. *(Top)* Map of example indoor NLOS environment (users not shown are on other floors). *(Bottom)* Map of example indoor LOS environment.

figured to collect fully-mobile channel traces in the UHF, 2.4 GHz, and 5 GHz bands simultaneously, though each band requires separate radios and antennas.

To enable realtime channel measurements, we implemented a from-scratch Python framework that provides complete control over the Argos radio modules, including

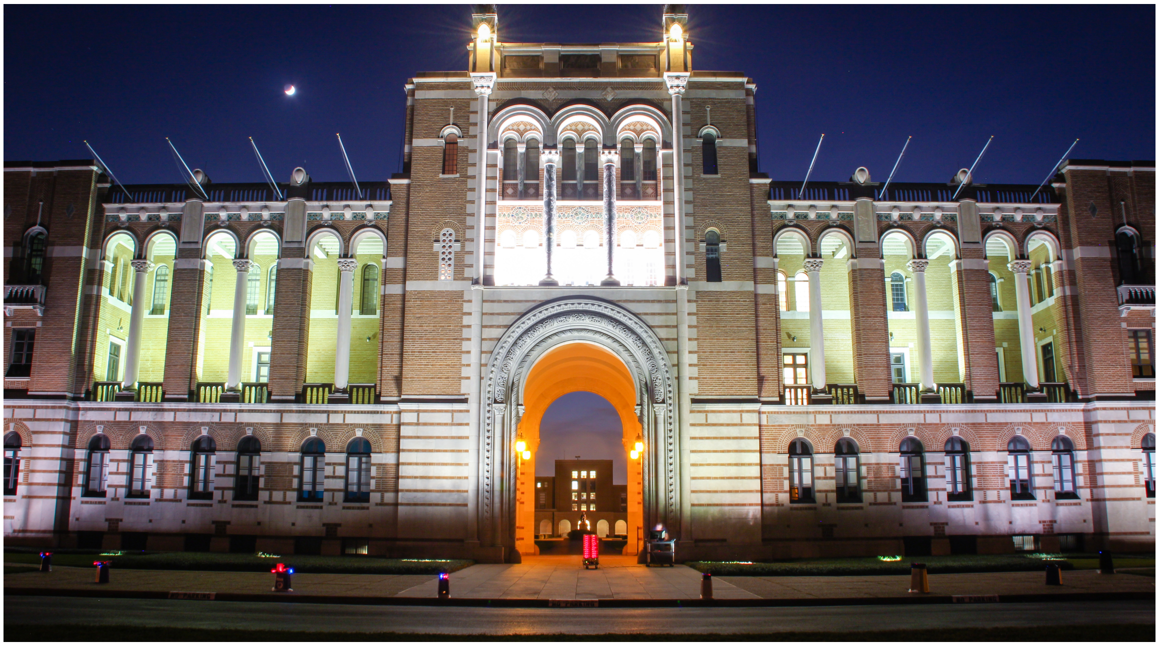


Figure 7.3 : Example of outdoor propagation environment.

parameters such as frequency, number of base-station antennas, number of users, frame length, pilot length, etc. This framework is very similar to the WARPLab [39] framework, which is a Matlab interface that provides arbitrary control of WARP nodes over Ethernet. However, our framework is much faster and utilizes asynchronous I/O to simultaneously interface with all of the nodes, as well as handle node-initiated communication. Notably, our Python framework is actually compatible with the default WARPLab design, and they can even be used together in the same experiment.

Leveraging this framework, we implement the Argos CentralController, as defined in [13], and provide functions such as uplink AGC and centralized beamforming computation. Additionally, the CentralController saves raw uplink pilot IQ samples in realtime to an HDF5 file [91, 92], enabling longitudinal traces lasting for hours, or even days, only limited by storage capacity. Optionally, the base-station nodes can also be configured to report uplink data IQ samples, which can also be stored to the same HDF5 file. This trace storage is performed either by a completely native Python implementation, or an accelerated C shared library, which supports sub-millisecond frame lengths, and therefore CSI resounding interval. Due to channel reciprocity, the uplink channels collected are equivalent to downlink channels, only varying by a constant relative phase shift, transmit power, and any asymmetric noise [13].

The current system supports up to 20 MHz bandwidth and 16 simultaneous users. The number of users supported can easily be expanded, at the expense of time or frequency resolution. The number of base-station antennas supported is virtually unlimited, and is reliant only on the computational and throughput capacity of the central controller along with the frame length. For example, when the central controller becomes the bottleneck, the frame length can be increased to support more antennas, or measurement can be easily parallelized across more servers. Since the central controller is implemented modularly and uses a standard Ethernet interface, it is straightforward to port other Software-Defined Radio (SDR) hardware to this measurement system.

In the results reported in this chapter, users were configured to send 802.11 Long Training Symbols (LTS) pilots, which have 52 subcarriers, in a Time-Division Multiple Access (TDMA) fashion at the beginning of each frame, however the system flexibly supports arbitrary pilot formats. This configuration enables wideband pilots for 8 users, along with a noise estimate, to be collected in less than 100 μs.

More information and updates on the measurement tools can be found online [14].

7.1.1 Continuous Measurements

In order to achieve the highest possible time-frequency resolution, as limited by the hardware sample rate, we also developed a WARPLab [39] based measurement system that is able to take continuous time measurements for up to 6 s at 20 MHz or 3 s at 40 MHz. However, this measurement system does not implement realtime continuous AGC and can only support one user. In this setup, continuous IQ samples are saved locally to memory at each WARP node, then offloaded to the central controller after the trace is complete. These traces are saved in the same format as the Python framework, enabling the same analysis tools to be applied to both types of traces. This system was developed to provide insight in to how user CSI behaves in real world environments at a sample-level resolution.

7.1.2 Multi-Cell Measurements

Synchronizing users in the multi-cell environment is relatively straightforward, we simply treat all the base stations as one large base station. Since the base stations are time-frequency synchronized, as described in §5.4.2, we can ensure that their beacons are aligned. While Faros would typically assign different beacons to different base stations, to simplify the measurement system, all of the base stations send the same beacon. Thus the beacons from different base stations simply look like multiple paths to the users, and the first strong path the user receives will be detected and enable it to synchronize. To avoid inter-symbol interference during the pilot phase given the discrepant path lengths, we use extra-long cyclic prefixes to ensure there is no symbol overlap in time.

The most significant challenge to multi-cell channel traces is AGC. In single base station measurements feedback can be used to adjust user gains, however in the multi-

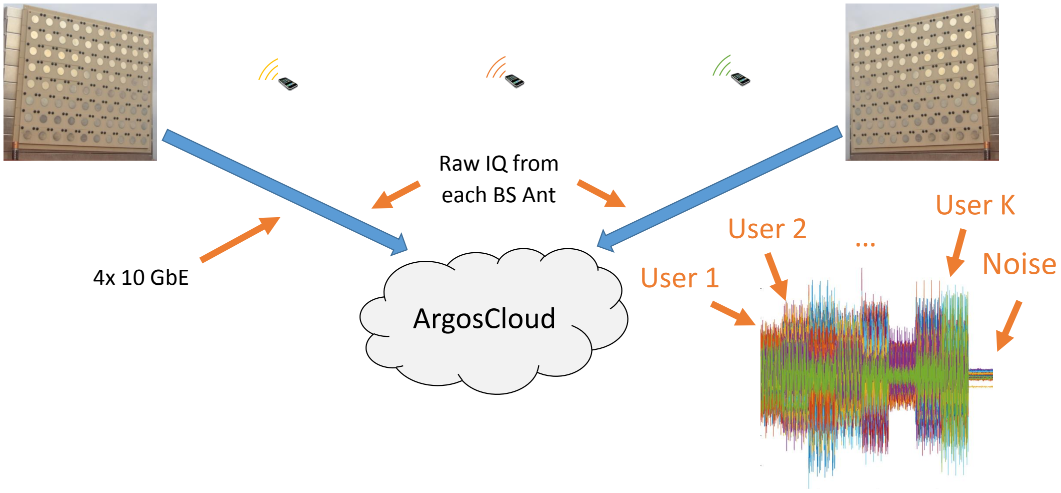


Figure 7.4 : Overview of the multi-cell channel measurement system design. At the beginning of each frame all base stations send a beacon to synchronize the users. Each user then sends orthogonal pilots, which every base station radio module records. At the end of the pilot phase, base-station radio modules report the raw pilot IQ samples to the ArgosCloud, which records them to an HDF5 file.

base station flow it is difficult to set gains for all of the base stations simultaneously due to their potentially highly discrepant paths. Thus to ensure pilots that are received within the dynamic range of all of the base stations, users step through preset transmit gains every frame. This sacrifices some time-resolution, but enables reliable multi-cell measurements. Of course gains can still be set manually, or set using AGC to a specific base station (either once or at each frame), depending on the experimental requirements.

7.2 Measurement Campaign

We conducted an extensive measurement campaign that includes fully mobile traces across the UHF, 2.4 GHz, and 5 GHz bands in diverse environments. At 2.4 and 5 GHz, we collected traces with up to 104 base-station antennas serving 8 users; at UHF, we collected traces with up to 8 base-station antennas serving 6 users. We collected measurements both indoor and outdoor environments, with varying mobility. These traces typically have frame lengths, i.e., time resolution, varying from 2 ms to 50 ms.

Our primary measurement campaign for this chapter consists of 96 base-station antennas serving 8 users at 2.4 and 5 GHz, and thus our analysis will focus on those measurements, particularly since, as shown by Fig. 7.14, comparing traces of different dimensions is not always equitable. The core of these measurements consists of 3 topologies, indoor NLOS, indoor LOS, and outdoor LOS, at 2.4 GHz and 5 GHz, with five levels of mobility: stationary, environmental mobility, user mobility, linear track mobility, and 'naturalistic' mobility. Stationary traces were taken at night with little or no mobility either at the users or in the environment. Environmental mobility consisted of two people intentionally walking around the mobiles and opening and closing doors. User mobility consisted of two people physically picking up the battery-powered mobiles and walking around with them, while other users were stationary. For controlled motion, we used a CineMoco track [93] to linearly move one user, with other users stationary and limited or no other environmental mobility. To simulate somewhat realistic usage, we also conducted a 'naturalistic' scenario where a person picked up one of the user devices and pretended it was a cell phone, then set it back down, with all other users stationary. Additional traces with controlled rotational mobility, two-antenna users, and other test-specific setups were also taken in some topologies. In most traces, AGC was set then disabled to avoid gain jumps during the trace, however this setting, including gain settings, are recorded in each trace.

Unless otherwise specified, these measurements used omnidirectional monopole antennas arranged in an 8 by 12 array spaced 63.5 mm apart, which is half a wavelength at 2.4 GHz, and approximately 1 wavelength at 5 GHz.

For analysis purposes, we found collecting relatively short traces of 20 to 120 s with a 2 ms channel resounding interval worked best for limiting the amount of data and processing time, while enabling us to investigate the behavior of specific scenarios. However, to investigate the longitudinal behavior of users in an office environment, we collected a few traces lasting 20 minutes with reduced time resolution.

Every measurement topology is well documented with experimental descriptions, maps, photos, and, in some cases, even video, which are included with each trace online. Examples of the different environments are shown in Fig. 7.2. More information on the UHF measurements can be found in [94]. To date, the online repository already contains over 100 traces spanning 20 topologies, providing over one billion channel measurements and 1 TB of data.

7.3 Channel Analysis Toolbox

To enable rapid analysis of these traces we provide a modular Python-based analysis toolbox, which leverages NumPy and SciPy [95] to accelerate computation. The toolbox converts the raw IQ samples in trace files to CSI, then is able to efficiently compute many useful channel characteristics, including correlation, coherence, Demmel condition number, achievable rate, impact of delayed CSI, and more. All results presented in this chapter were computed using this channel analysis toolbox, which is made available open-source online [14]. Our analysis is primarily concerned with the following channel characteristics:

Correlation is the normalized product of a user's CSI, $\mathbf{h}_i \in \mathbb{C}^{1\times M}$, with the conjugate of a user's CSI, $\mathbf{h}_j$: $\delta(t) = \frac{|\mathbf{h}_j^H(t)\mathbf{h}_i(t_1)|}{\|\mathbf{h}_j(t)\|\|\mathbf{h}_i(t_1)\|}$. Correlating the same user's CSI, $i = j$, at different points in time provides insight in to the channel stability; this auto-correlation corresponds to the square root of the received power if the base station were to serve that user with single-user beamforming. The cross-correlation of one user's CSI with another user's CSI provides insight in to the channel orthogonality and inter-user interference.

Coherence is the statistically expected auto-correlation of a channel with itself given a time delay of Δ: $\rho_i(\Delta) = \mathbb{E}[\frac{|\mathbf{h}_i^H(t-\Delta)\mathbf{h}_i(t)|}{\|\mathbf{h}_i(t-\Delta)\|\|\mathbf{h}_i(t)\|}]$. Channel coherence provides statistical insight in to the expected behavior of channels across time, and is useful for comparing different mobilities, propagation environments, and frequencies. Notably, some previous work, e.g., [1, 96, 97], considers each SISO channel separately, thus discarding the relative phase information between base-station antennas. In these results we use the expected auto-correlation of the MIMO channel, that is, the auto-correlation of the user's entire CSI vector, which provides much better insight in to the performance of MU-MIMO. *Coherence time* is defined as the delay before the expected correlation falls below a certain threshold, typically 0.95 or 0.90.

Demmel condition number, $d \in [K, +\infty)$ is the ratio of the sum of the eigenvalues to the minimum eigenvalue: $d = \frac{\sum_{k=1}^{n} \lambda_k}{\lambda_n}$, where $\lambda_1 > \lambda_2 > \cdots > \lambda_n$ are the eigenvalues of the matrix $\mathbf{HH}^H$ [98]. The Demmel condition number is a key indicator of MU-MIMO performance for a given set of users. In multi-user conjugate beamforming systems, it indicates the amount of inter-user interference to be expected; in zeroforcing, it indicates the amount of power that has to be sacrificed to form the nulls to each user.

System Capacity, or achievable rate, is computed as the aggregate empirical Shannon capacity of each user's MU-MIMO channel:

$$C(t) = \sum_{i=1}^{K} log_2(1 + \frac{|\mathbf{w}_i(t-\Delta)\mathbf{h}_i^T(t)|^2 P_i}{\sum_{j\neq i} |\mathbf{w}_j(t-\Delta)\mathbf{h}_i^T(t)|^2 P_j + N}) \tag{7.1}$$

where $\mathbf{W} \in \mathbb{C}^{M\times K}$ are the beamweights for a linear beamformer, and P is the per-user transmission power. For multi-user conjugate $\mathbf{W}$ is simply the complex conjugate of the channel matrix, $c\mathbf{H}^*$, $\mathbf{H} \in \mathbb{C}^{K\times M}$, where c is a power scaler. For zeroforcing, $\mathbf{W}$ is the pseudo-inverse of the channel matrix, $c\mathbf{H}^*(\mathbf{H}^T\mathbf{H}^*)^{-1}$. It is important to note that beamforming weights should never be applied to the CSI they were derived from to determine system capacity, as not only is this impossible in a real system, but also because it correlates noise power, resulting in inaccurate achievable rates. To avoid this, our system capacity analysis uses the first LTS pilot symbol to derive the beamweights, then applies them to the second LTS pilot symbol; this provides consistency for any Δ, while enabling accurate analysis for $\Delta = 0$. Thus, we can emulate the achievable rate offline using only recorded CSI.

Expected system capacity is the statistically expected ratio of the achievable rate between when CSI is estimated and when that CSI is used for MU-MIMO given some time delay, Δ: $\gamma(\Delta) = \mathbb{E}[\frac{C(t-\Delta)}{C(t)}]$. Expected system capacity can provide insight in to how a MU-MIMO system will perform given a channel resounding interval (Δ) under different mobilities, propagation environments, and frequencies.

7.4 Results

We analyze the channel traces from the measurement campaign with regard to the impact of mobility, environment, and base station scale on the channel stability and system capacity. Many of these results are intuitive and expected, however our mea-

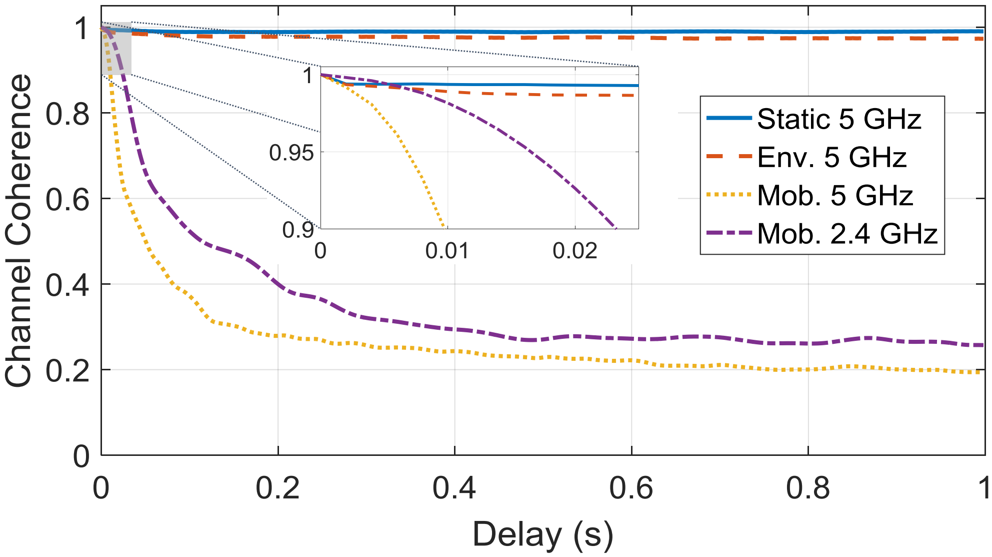


Figure 7.5 : Channel coherence of 5 GHz for completely stationary topologies (Static), topologies with environmental mobility (Env.), and topologies with the users moving at pedestrian speeds (Mob.) in NLOS environments. Stationary and environmental mobility topologies are long-term stable, whereas mobility drastically reduces channel coherence. To demonstrate the impact of frequency, 2.4 GHz with user mobility is also shown, which has a higher channel coherence than 5 GHz with similar mobility.

surements characterize ground-truth points for real-world environments.

7.4.1 Impact of Mobility

We investigate six levels of mobility:

Stationary Environments and Users

Stationary environments are long-term stable over periods of tens of minutes, with no indication of changing, regardless of carrier frequency. As Figs. 7.5 and 7.6 show, stationary channels have a virtually indefinite channel coherence. In many other

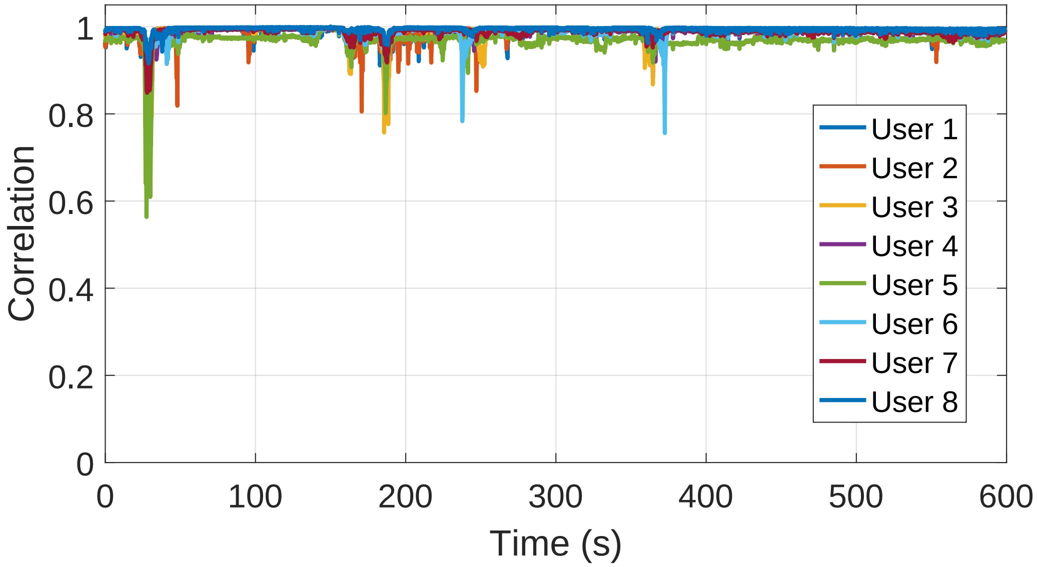


Figure 7.6 : Each users' auto-correlation with a single channel measurement at 300 s in NLOS at 2.4 GHz. Even with environmental mobility stationary users' channels are remarkably stable: other than brief interruptions, their channel correlation typically stays above 0.95.

traces we have observed channels in stationary environments be stable for hours, maintaining a channel correlation above 0.98.

Environmental Mobility

As Fig. 7.5 shows, environmental mobility slightly reduces the coherence of the channel, however it still stays above 0.97 for over a second. Longitudinal studies of the channels in a real-world office environment show similar results; we see in Fig. 7.6 that channels are seemingly indefinitely stable other than brief periods where the channel is altered, e.g., when a person walks between the user and base station, occluding some of the paths. Typically we observe that each user experiences independent

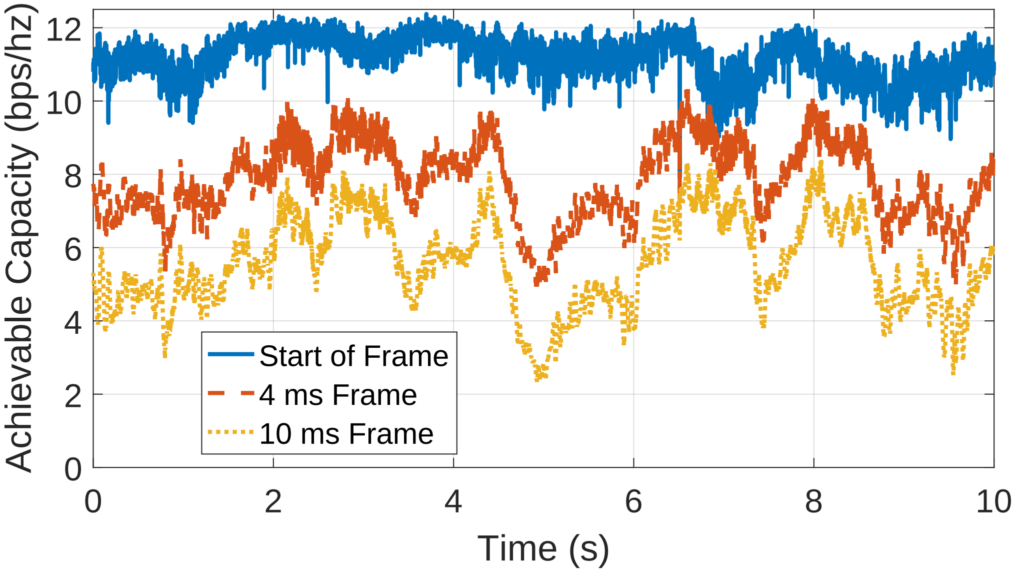


Figure 7.7 : Achievable rate at start of a frame, i.e., immediately after CSI collection, as well as at the end of either a 4 ms or 10 ms frame, for a single user with pedestrian mobility in a 96x8 zeroforcing system at 2.4 GHz.

fades, however in some cases mobility near the base station can cause fades for multiple users, as is the case at approximately 30 s in Fig. 7.6. Thus the achievable rate of many-antenna MU-MIMO systems with environmental mobility is still very stable, but can exhibit brief drops of up to 45%. Notably, the frequency and severity of these fades is quite clearly dependent on the topology and actual mobility.

User Mobility

User mobility drastically impacts channel stability, as it affects all paths across the entire channel bandwidth. As shown in Fig. 7.5, with user mobility 0.90 channel coherence drops to 9.5 ms and 23 ms for 2.4 GHz and 5 GHz, respectively, in a 96x8 MU-MIMO system with NLOS propagation. Similarly, 0.95 channel coherence is 7

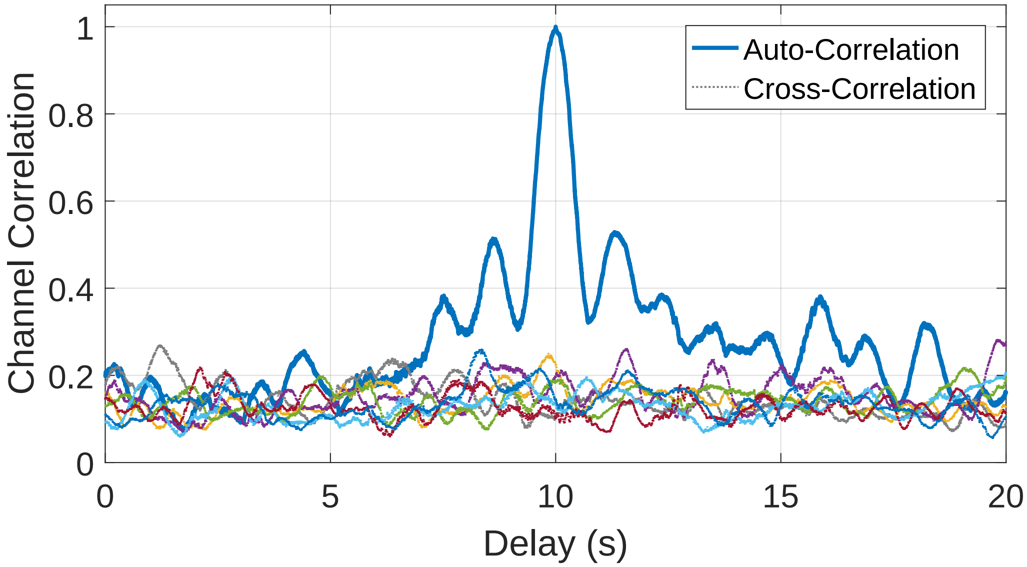


Figure 7.8 : Correlation with user at 10 s on a linear track moving at $\tilde{4}$.6 cm/s in LOS at 5 GHz. The top (blue) curve is the moving user's auto-correlation, the lower curves are the cross-correlation with the other 7 users. The 20 s shown corresponds to approximately 92 cm.

ms and 16 ms, respectively. This has an severe impact on achievable rate, as a system that estimates the user channels every 10 ms will lose an average over 50% of system capacity, with dropouts as high as 80%, even with just pedestrian mobility at 5 GHz, as shown in Fig. 7.7.

Track Mobility

By controlling mobility with a linear track we are able to draw interesting insight in to the interference patterns in different environments and at different frequencies. Fig. 7.8 demonstrates the correlation of the user on the track with itself across time, and thus space, as well as its cross-correlation with the other 7 users, in a 5 GHz NLOS

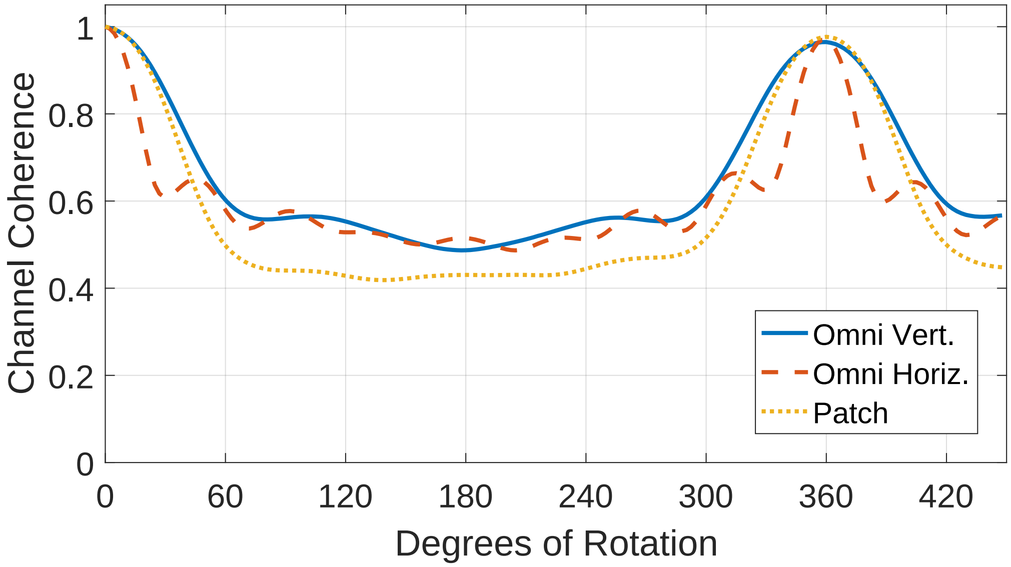


Figure 7.9 : Channel coherence of a user with controlled rotational mobility at 2.4 GHz in NLOS using an omnidirectional antenna and a patch antenna. Rotational mobility can also significantly impact channel stability; the antenna's non-isotropic radiation pattern creates additional spatial selectivity.

environment. This allows us to visualize how the intended signal strength and inter-user interference varies across space. For example, if the system were to use multi-user conjugate, the top (blue) curve represents the amplitude of the intended signal for the user on the track, whereas the other curves each represent the interference from other users. Similarly, this same method can be used to visualize the signal strength and interference of specific beamforming techniques, such as zeroforcing.

Rotational Mobility

We found rotational mobility, where the user's antenna rotates in space, to have a significant impact on channel stability. Leveraging a controlled stepper motor we

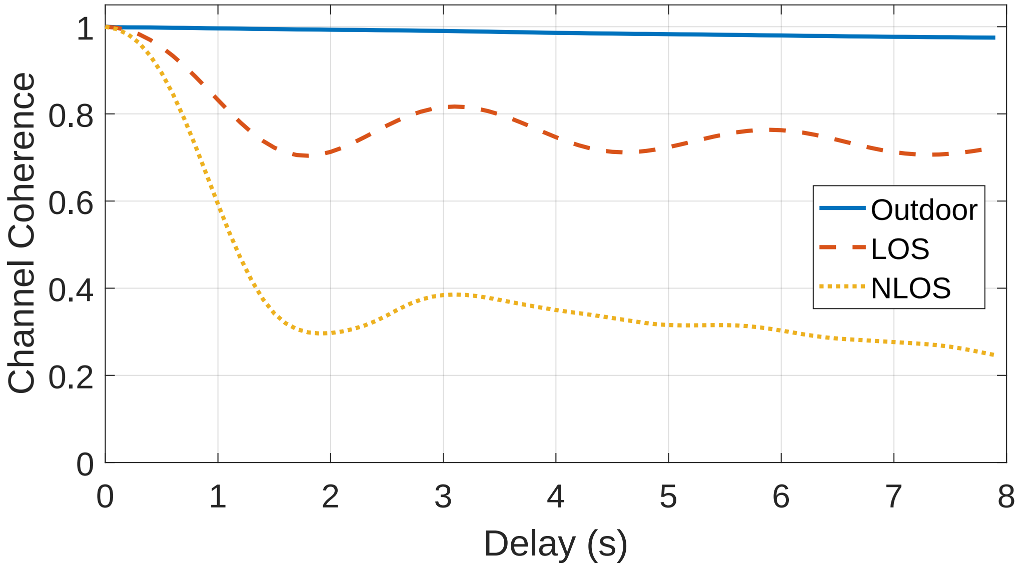


Figure 7.10 : Channel coherence at 2.4 GHz on a linear track moving at 4.6 cm/s. NLOS environments are less stable than LOS environments with mobility, and outdoor environments with little multipath are very stable. The 8 s shown corresponds to approximately 37 cm of movement. Note that the sinusoidal behavior corresponds with a wavelength, e.g., an interference pattern from reflections.

rotated a user's antenna at a constant rate of 18.75 rpm, then measured the channel at 2.4 GHz NLOS to a 96-antenna base station. Due to the connector and mounting, the rotating antenna was offset 2 cm from the center of rotation. Using this experimental setup we collected traces with the following user antennas: (i) a 14 cm omnidirectional 3 dBi monopole oriented vertically; (ii) the same antenna oriented horizontally, its tip thus having a radius of 16 cm from the center of rotation; and (iii) a 6 dBi patch antenna with an 80 degree beamwidth in both azimuth and elevation. As shown in Fig. 7.9, within 25 degrees of rotation, all three scenarios fall below 0.90 channel coherence, with the horizontally mounted antenna reaching 0.90 coherence

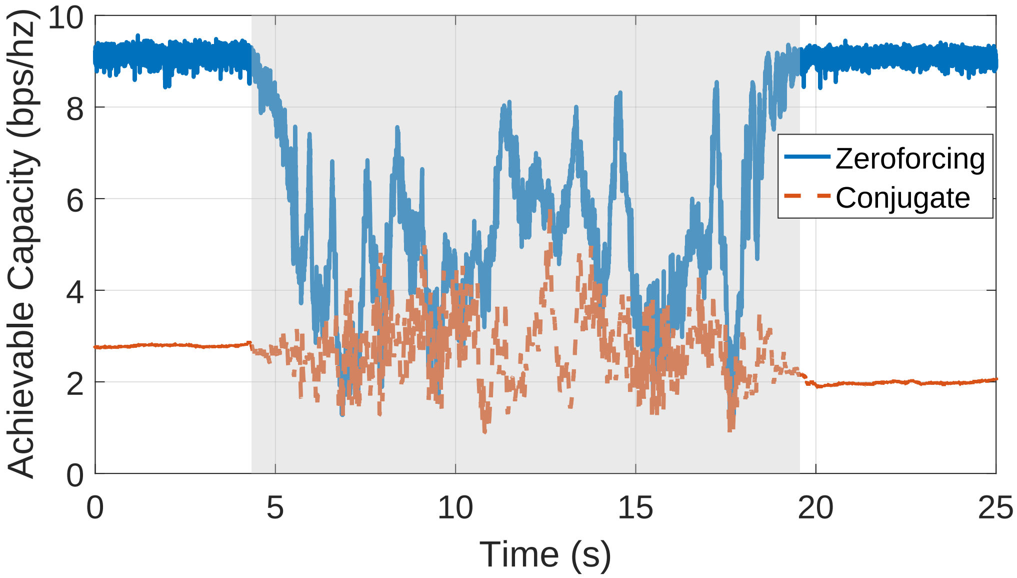


Figure 7.11 : Achievable rate plot for a single user with naturalistic mobilitiy, e.g., a cell phone being picked up and used in a 96x8 system with a 10 ms resounding interval. The shaded region indicates when the user was moving. We see the performance is very bimodal with the channel stable while the user is stationary, and very unstable while the user is moving.

with just 12.5 degrees of rotation. Unsurprisingly, when the antennas reach their original position, at 360 degrees, we see coherence returns to over 0.97, as should be expected. Rotational movement occurs in the pedestrian and naturalistic traces, e.g., Figs. 7.5 and 7.11, and thus likely contributes to the low channel stability. As rotational mobility is known to be common in mobile device usage [99], our result suggests that it should be considered in MU-MIMO system design and modeling.

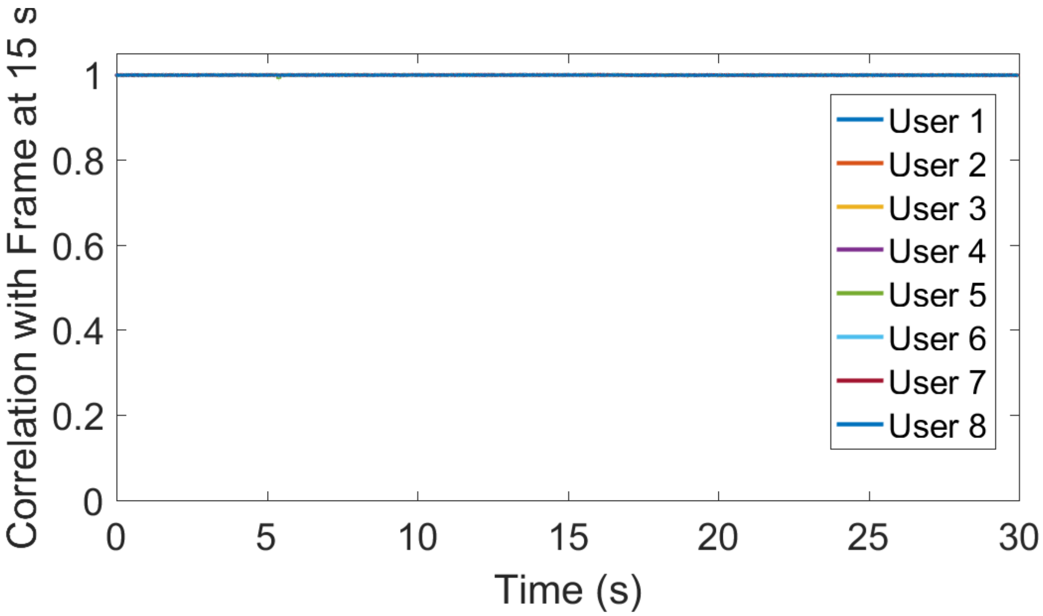


Figure 7.12 : Each users auto-correlation with a single channel measurement at 15 s in NLOS at 2.4 GHz. Stationary channels are long term stable: channel correlation stays above 0.98 for 10s of minutes in UHF, 2.4 GHz, and 5 GHz.

Naturalistic Mobility

We find that naturalistic mobility, e.g., the user equipment is held and moved to mimic cell phone use, results in bimodal channel stability; that is, the user is typically either stationary or moving, which results in very stable and very unstable channels, respectively. As shown in Fig. 7.11, we see that initially the channel is very stationary, but at about 5 s the user receives a call and begins to move around until about 20 s when the call ends and the user sits down, causing the channel to stabilize. A video of this mobility is included with the trace online, and there are other similar traces in other topologies.

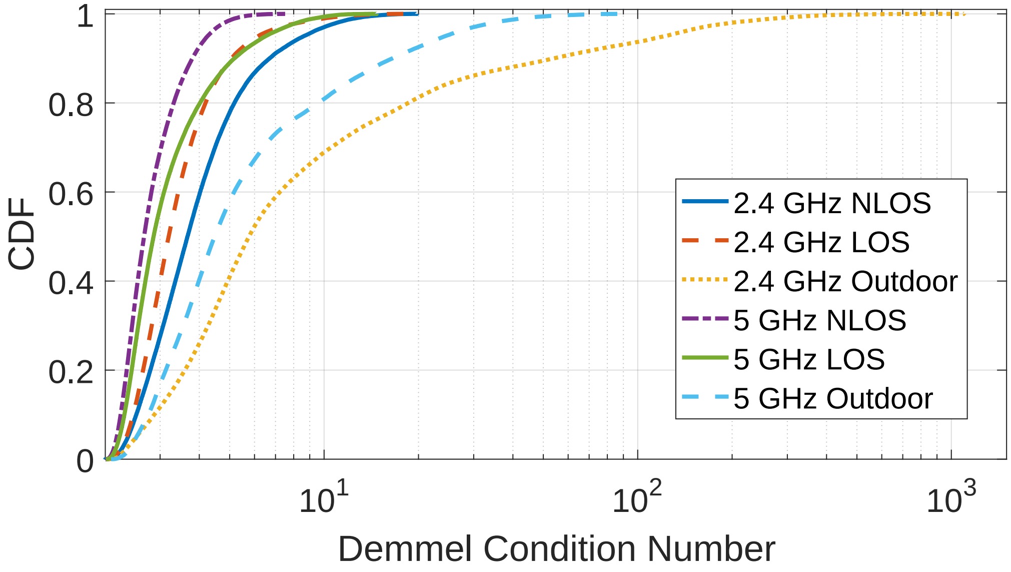


Figure 7.13 : Cumulative distribution function (CDF) of average Demmel condition number for NLOS, LOS, and outdoor propagation environments at 2.4 GHz and 5 GHz in a 96x2 system. In outdoor environments the lack of multipath can make it much harder for MU-MIMO to separate users.

7.4.2 Impact of Environment

The environment has a strong impact on user orthogonality and channel stability.

As expected, more multipath increases user orthogonality, and therefore performance, as MU-MIMO can more easily separate multiple users. As shown in Fig. 7.13, the lack of multipath in outdoor propagation environments makes the channel poorly conditioned, meaning that it is much harder for MU-MIMO to separate users efficiently. This indicates that MU-MIMO can strongly benefit from multipath, and that user selection in high multipath environments is less critical than environments with low multipath, as corroborated in [100].

However, this better channel condition with multipath comes at a cost: the addi-

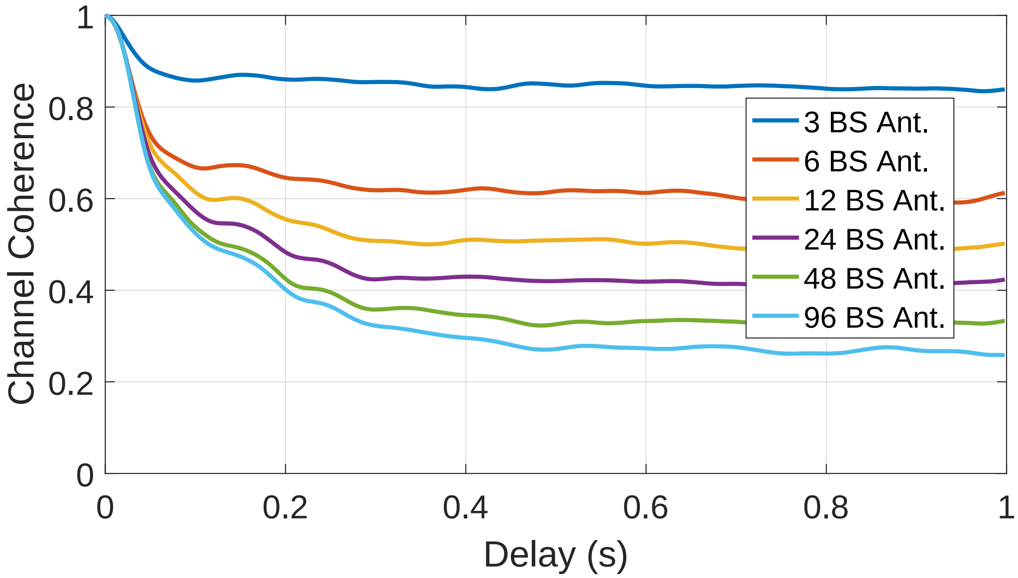


Figure 7.14 : Channel coherence vs. number of base-station antennas with pedestrian mobility in 2.4 GHz NLOS. Scaling up the number of base-station antennas significantly reduces coherence.

tional spatial selectivity makes the channels much less stable in the presence of user mobility, as shown by Fig. 7.10. In outdoor environments the beamwidth created by the base station's antenna aperture dominates channel coherence, whereas in high multipath NLOS environments the channel coherence is dominated by the motion relative to the wavelength. Notably, in our experiments with the track spaced approximately 7 m in front of the array, with a horizontal aperture size of 0.5 m, both 2.4 and 5 GHz have horizontal beamwidths of more than 1.25 m, thus the coherence in the outdoor measurement is expected.

While it is difficult to quantize generally, based on our topologies we also found that NLOS environments are typically not as affected by environmental mobility, since, due to the spatial diversity, it is rare for environmental movement to affect

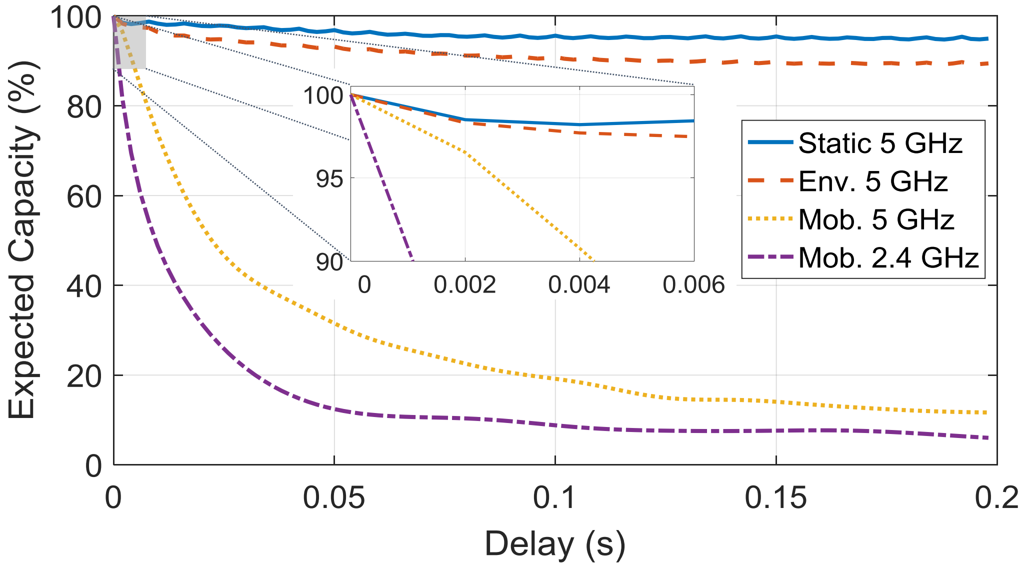


Figure 7.15 : The expected achievable rate of a mobile user in a 96x8 MU-MIMO system vs. channel resounding interval in NLOS. The channel resounding interval required to maintain 90% average system capacity can be an order magnitude lower than the measured channel coherence.

a large portion of paths and frequencies. In contrast, LOS environments show a more bimodal behavior: environmental movement in the direct path causes more extreme changes in the channel, whereas movement not in the direct path rarely has much effect. Thus, with stationary users and environmental mobility, LOS topologies exhibit deep fades more rarely, whereas NLOS topologies exhibit less severe fades more frequently.

7.4.3 Impact of Increasing Number of Base-Station Antennas

In Fig. 7.14 and 7.17 we see that increasing the number of base-station antennas significantly decreases coherence time. More antennas introduce more spatial selectivity, i.e., a narrower beam width, thus movement has a stronger impact on the channel.

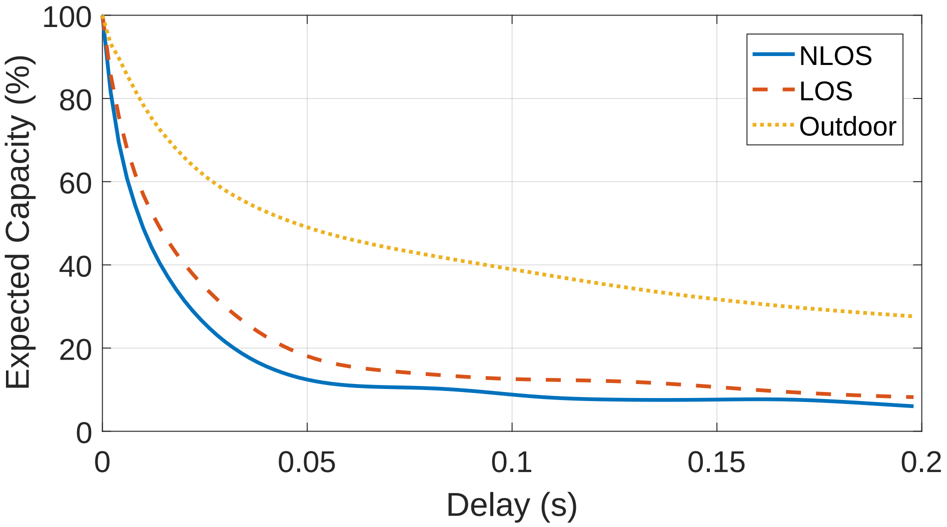


Figure 7.16 : Expected achievable capacity of a mobile user in a 96x8 zeroforcing MU-MIMO system at 2.4 GHz vs. sounding interval for indoor LOS and NLOS, as well as outdoor. The outdoor capacity drops much slower than indoor capacities due to the reduced multipath.

This effect is mathematically expected and fundamental to [18]; as the number of base-station antennas increases, different spatial locations become more orthogonal. Notably, we found this reduction in channel coherence does not necessarily reduce performance, as users can actually be moving away from interference from other users, and additional antennas help suppress inter-user interference.

7.4.4 Multi-Cell Results

We analyze initial multi-cell traces and demonstrate the ability for many-antenna beamforming to reduce network interference, as well as the ability for many-antenna CoMP to drastically increase performance, particularly at the cell edge. These preliminary results are not intended to be comprehensive, but simply demonstrate the

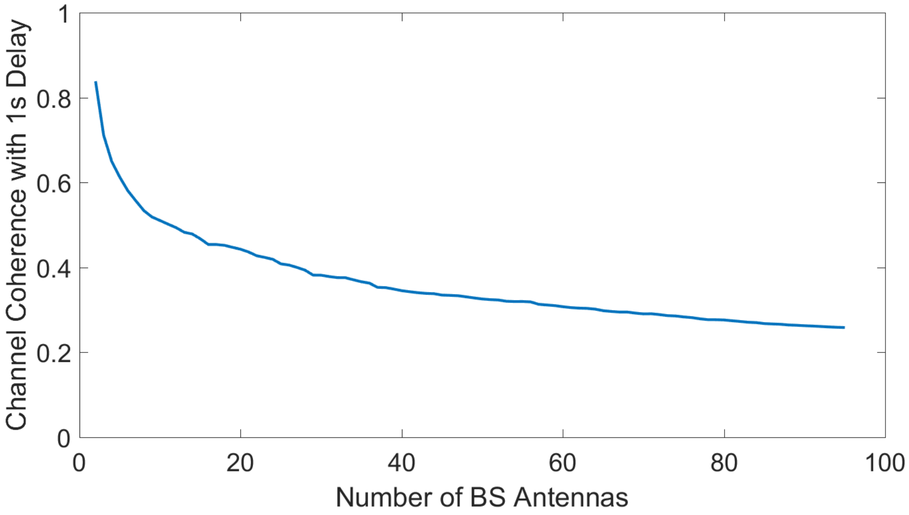


Figure 7.17 : Number of base-station antennas vs. channel coherence with 1s delay at 2.4 GHz. Scaling up number the number of base-station antennas significantly reduces channel coherence.

capabilities of the ArgosNet platform and measurement system. We are currently performing an at-scale measurement campaign leveraging the full ArgosNet installation. When it is complete, the measurement tools, channel traces, and analysis toolbox will be made freely available on the Argos Channel Measurement Repository [14].

Experimental Setup

We setup indoor lab experiments with two 10-antennna base stations placed 3 m apart, operating at 2.484 GHz with two users. Both base stations are locked to a common clock source and they receive a GPIO trigger pulse for initial time synchronization.

Leveraging the Argos channel measurement system described in §7.1.2, we collect full CSI traces at a time resolution of 10 ms and a frequency resolution of 20 MHz. We place a stationary user near the first base station and a mobile user on a 2.5 m

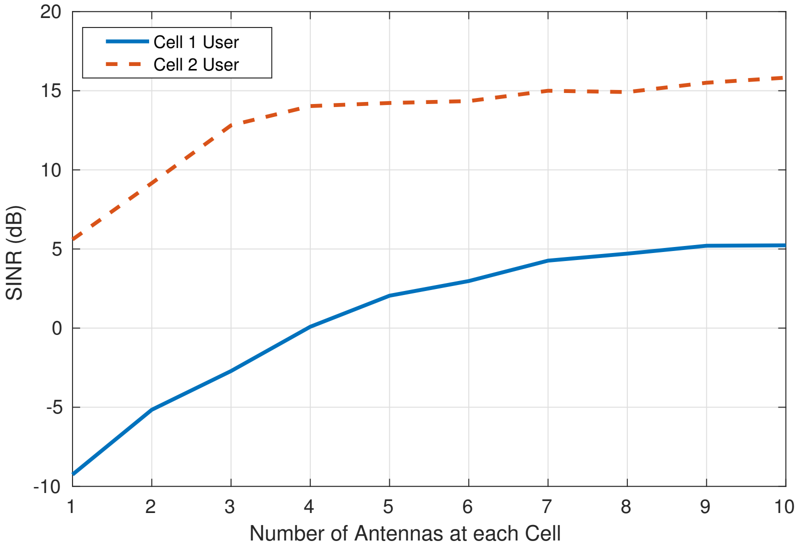


Figure 7.18 : SINR of users vs. number of base station antennas in two cells, each with one user. Increasing the number of base station antennas naturally reduces network interference through beamforming.

Cinetics track to enable constant linear motion at approximately 2.2 cm/s. At the start of the measurement, this mobile user is in close proximity to the second base station then moves away from it toward the first base station at a constant speed. Both users have LOS to both base stations.

Network Interference

To demonstrate the ability for beamforming to naturally suppress network interference, as discussed in [18], we emulate two cells each with one user, where users experience strong inter-cell interference. In Figure 7.18 we see that adding additional antennas drastically improves SINR by reducing inter-cell interference. By increasing the number of antennas on a base station, beamforming maintains the same signal

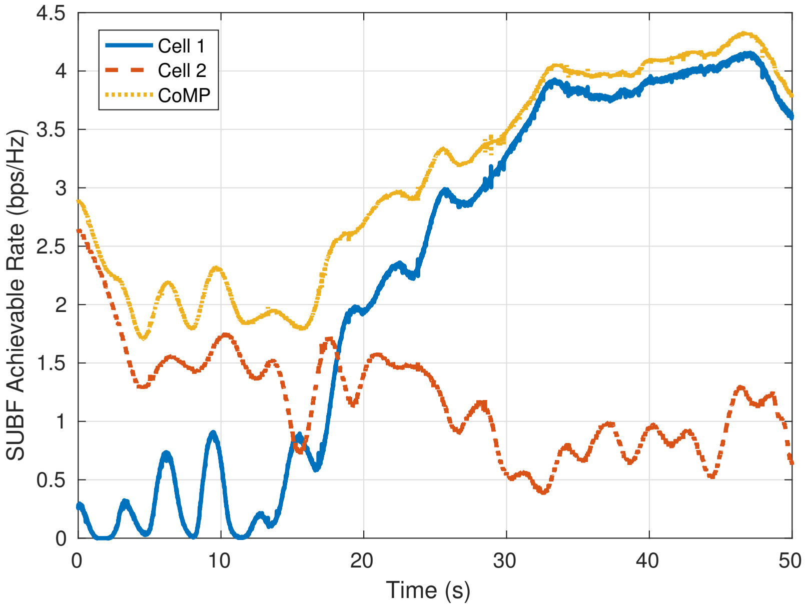


Figure 7.19 : Achievable rate of a user moving from one 10-antenna cell to another over time when be served by each cell individually as well as CoMP with coherent joint transmission, i.e., beamforming.

strength to the intended user, while reducing the total emitted power, thus also reducing network interference.

Coordinated Multipoint

We implement the most advanced form of CoMP, coherent joint transmission, i.e., beamforming, and show its ability to drastically improve performance on the cell edge. Figure 7.19 compares the achievable rate of a single user moving from one cell to the other while served by each cell individually, as well as with CoMP coherent joint transmission. In this scenario we see that CoMP almost doubles performance at the cell edge, at approximately 15 s, the hardest locations to serve in cellular networks. Notably, Figure 7.18 also demonstrates the negligible benefit of complex CoMP schemes when users are not on the cell edge.

7.4.5 System Implications

These channel characteristics have important implications for many-antenna MU-MIMO system design. Due to the instability of system capacity under just pedestrian mobility, e.g., losses of up to 50% within 4 ms of channel sounding, shown in Fig. 7.7, selecting the Modulation and Coding Scheme (MCS) for users will be critical, and likely necessitates fine-grain adaptive or rateless coding schemes, such as [101, 102]. The significant differences we observed in channel coherence and orthogonality across environments and frequencies also must be considered in MU-MIMO systems, particularly for user selection algorithms and CSI collection. Furthermore, the bimodal mobility inherent in naturalistic user movement, shown in Fig. 7.11, can be leveraged to significantly reduce channel sounding overhead.

We also find that measured channel coherence is *not* an accurate estimate of the channel resounding interval in a MU-MIMO system, which is corroborated in [103, 104]. The 90% expected system capacity, shown in Fig. 7.15, for 2.4 and 5 GHz are 1.1 ms and 4.2 ms, respectively. For 2.4 GHz, this is over 20 times lower than the 0.9 coherence time for the same trace. We counterintuitively observe that the expected system capacity drops more quickly for 2.4 GHz than 5 GHz in this scenario. This is because capacity is dependent on a many factors, including the beamformer, the number of other users, their orthogonality, and the environment. In particular, the initial beamformed channel SINR significantly affects expected system capacity, as high SINR channels are inherently more unstable; this is actually the predominant reason for the rapid degradation of 2.4 GHz system capacity in Fig. 7.15. To be clear, channel coherence can be defined, and measured, in many ways, e.g., the theoretical block-fading model assumes the channel does not change during a coherence time interval. Thus using channel coherence interchangeably with the channel resounding interval is not necessarily incorrect, however most *measurements* of channel coherence do not account for all of the factors that affect system capacity, which certainly should be considered in the design of channel sounding protocols for MU-MIMO.

7.5 Related Work

To the best of our knowledge, we are the first to characterize high-resolution (as fast as sub-millisecond collection interval) temporal behavior of many-antenna (up to 104 × 8) MU-MIMO channels, across multiple frequency bands. A number of prior MU-MIMO channel measurements have been reported, including [1,13,96,97,103,105–115]. The evaluation of empirical time-variant channels has mostly focused on small-scale systems with up to 16 base-station antennas [1, 96, 97, 103, 107–110]. Some results from previous work differ from ours, e.g., [1, 109], reported stationary users having unstable channels, whereas our measurements indicate stationary users typically have very stable channels, regardless of environmental mobility.

Many-antenna MU-MIMO channel measurements have been reported in [111–115] with up to 128 base-station antennas. In particular, the authors in [115] measured channels with user mobility on a system with 128 base-station antennas and 8 single-antenna users. These measurement campaigns built a comprehensive foundation for realistic many-antenna MU-MIMO channels in static environments, however they do not report analysis or results regarding environmental or user mobility. Since these measurement platforms leverage virtual antenna arrays or multiplexed antenna arrays, which have inherent RF switching overhead, it seems they are unable to achieve the time-resolution required to accurately sample mobile channels. Our measurements indicate that channels for all users should be collected within 100s of μs, and the resounding interval needs to be on the order of a few ms to provide complete and accurate measurements for NLOS environments with pedestrian mobility.

Chapter 8

Mobility-Aware MU-MIMO

While MU-MIMO can drastically increase the spectral efficiency of wireless systems, mobility fundamentally limits its performance in real systems. This due to the beamformed system capacity degrading over time from movement, as we show in §7.4.1, which in turn requires user locations (their CSI) and the beampattern to be updated, resulting in substantial overhead, as we show in §4.2.4. In fact, MU-MIMO systems that do not adapt to mobility or take in to account overhead can perform worse than traditional Single-Input, Single-Output (SISO) systems, as shown in [100], and we model in §4.2.4. Optimizing MU-MIMO performance in a real-world mobile system is a significant challenge that is affected by channel estimation overhead, computational overhead, beamformer initial performance and degradation, user orthogonality, Signal-to-Noise Ratio (SNR), number of spatial streams, sounding interval, rate adaptation interval, and more. To make the problem even more challenging, many factors are unknown, and can only be statistically modeled, such as future channel states and user traffic. Leveraging the ArgosV2 platform, described in §5.2, we implement a mobility-aware MU-MIMO system, Kinitos[*], which consists of: 1) efficiently maintaining accurate CSI for all active users, 2) user selection and grouping, 3) beamformer selection, 4) choosing beamformer update interval, and 5) MCS rate selection. All five components are heavily impacted by mobility.

[*] $K\iota\nu\eta\tau\acute{o}\varsigma$, or Kinitos, means "mobile" in Greek.

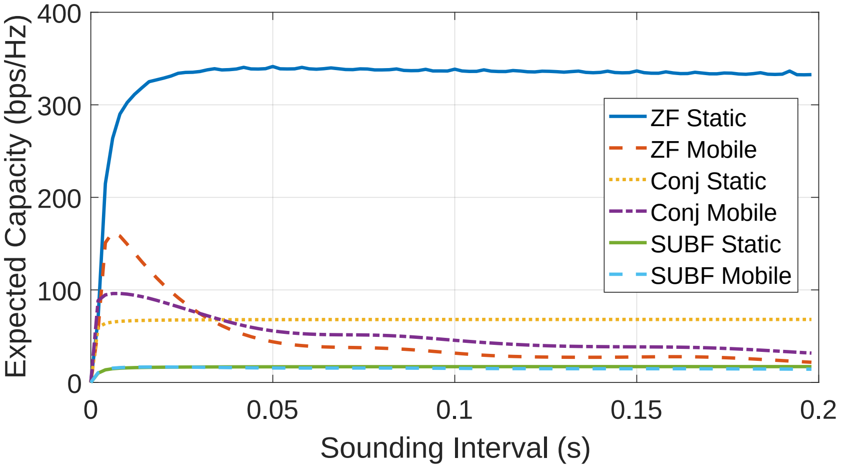


Figure 8.1 : Leveraging real channel traces from a 96x32 massive-MIMO array we emulate downlink user achievable rate with overhead vs. channel sounding interval at 2.4 GHz in an NLOS environment with high SNR (30 dB). We see that the optimal channel sounding interval for mobile scenarios is approximately 6 ms for both zeroforcing and conjugate.

We show that Kinitos always performs within 5% of an oracle system across scenarios with varying mobility, SNR, and number of users.

8.1 Factors that Affect System Performance

As discussed in Chapter 4, characterizing the impact of mobility on MU-MIMO performance is very complex, and depends on many factors. This model is very useful for gaining insight in to how massive-MIMO performs in the real-world, as well as guide system design; however, once a system is built and deployed, most of these factors have already been determined. Moreover, the model presented in Chapter 4 assumes that all users have the same mobility, and that after a coherence period the channel capacity drops to zero, which is clearly not true in real channels, as we measured in Chapter 7.

The parameters that can be chosen at runtime on a real many-antenna base station are pilot scheduling, user selection and grouping, beamformer selection, beamformer update interval, and MCS selection. Notably, there has been substantial prior work on user selection and grouping, [116, 117], as well as rate selection, [118, 119], so we use these existing techniques in Kinitos. User mobility, number of users, and SNR have a significant impact on the optimal pilot scheduling, beamformer selection, and beamformer update interval.

8.1.1 User Mobility

As shown in Chapter 4, user mobility drastically affects MU-MIMO performance, altering the optimal channel sounding interval and beamformer selection. Additionally, based on our channel measurements in Chapter 7, we find users are typically either stationary or mobile, and while mobile users channels vary rapidly, stationary users' channels are almost indefinitely stable, even in the presence of other mobile users. As shown by Figure 8.1, mobile users require frequent channel estimates, and beamformer updates, in order to maintain gains from MU-MIMO.

When stationary users are grouped with mobile users, typically only the mobile users need frequent channel estimates, but every time the base station estimates the channels it also has to update the beamformer for the new CSI to be useful. Therefore, it is not useful to collect CSI for mobile users without updating the beamformer, but it is not necessary to update the CSI of every user before updating the beamformer, since only a subset of users may be mobile. Thus pilot scheduling and beamformer update interval are related in that pilots should only be scheduled as often as the beamformer is updated, but updating the beamformer does not necessarily mean pilots should be scheduled for every user. Notably, the ratio of stationary to mobile users affects the optimal beamformer update interval; the more stationary users the longer the update interval should be to optimize performance. However, this longer update interval is essentially sacrificing the achievable rate of the mobile users, which raises issues of fairness.

User mobility also affects beamformer selection; when users are more mobile, the cost of centralized beamforming is relatively higher, thus, given enough mobility, decentralized beamforming can become more performant in the real world.

On the surface, mobility degrading performance seemingly contradicts prior work, such as [120], which leverages the additional multiuser temporal-spatial diversity from mobility to improve capacity, e.g., by having users transmit when they have higher SNR channels. These techniques operate on long time scales, on the order of seconds or even up to hours. On short time scales, mobility degrades capacity in a beamformed system since the users move out of their intended beams and in to the interference patterns of other users. In fact, on longer time-scales, beamformed systems can also benefit from the same techniques proposed in [120]. Notably, more base station antennas likely reduce the achievable gains from this mobile multiuser temporal-spatial diversity since the system is already oversampled in space, due to channel hardening [121].

8.1.2 Signal to Noise Ratio

The channel SNR both affects the beamformer selection as well as the beamformer update interval. As the SNR decreases, the performance of centralized beamforming approaches the performance of decentralized beamforming, as shown in Figures 8.2

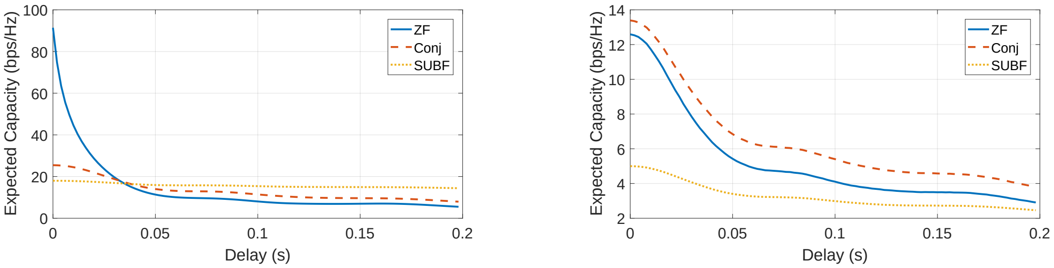


Figure 8.2 : Impact of SNR on beamformer performance in a 96x8 2.4 GHz NLOS system with mobile users. We see that as SNR drops the relative performance of zeroforcing drops quickly, and eventually falls below conjugate. Moreover, it becomes less sensitive to mobility. *(Left)* 30 dB per-link SNR. *(Right)* -3 dB per-link SNR.

and 8.3, as well as our prior work [13].[†] One way to think of this is that as SNR degrades the inter-user interference in conjugate eventually falls below the noise floor and becomes irrelevant; in this scenario the inter-user interference suppression in centralized beamforming is irrelevant, or even harmful since intended signal is sacrificed to form the nulls.

Interestingly, SNR also affects the optimal beamformer update interval: high-SNR user groups require more frequent beamformer updates than low-SNR user groups. This is because the achievable rate of high-SNR groups is inherently less stable than low-SNR groups, as shown by Figure 8.2; we see that the performance of a high-SNR group degrades much more quickly than the low-SNR group. Essentially this is because the high-SNR achievable rate relies on perfect cancellation of inter-user interference (nulls), so any movement causes users to experience interference, substantially

[†] Note that in real systems MMSE would always be chosen over zero-forcing, as it requires very little extra computation, and always performs as well or better than zeroforcing or conjugate, at least ignoring overhead.

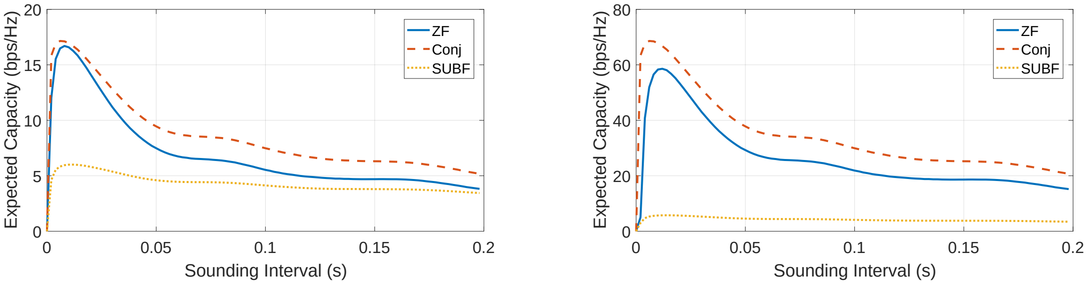


Figure 8.3 : Performance of low SNR (3 dB) 2.4 GHz NLOS system with mobile users, accounting for computational and channel overheads: *(left)* 96x8, *(right)* 96x32. We see that conjugate becomes the optimal choice of beamformer, and increasing the number of users to 32 expands this performance gap further.

degrading their SINR. In the low-SNR regime the noise floor is the dominant factor in SINR, and thus variations of inter-user interference have a much lower impact.

8.1.3 Number of Users

Increasing the number of simultaneously served users, or spatial streams, typically increases system capacity; however it also increases the channel sounding and computational overhead. When the users are stationary the additional spatial streams offer enormous performance improvements, but when the users are mobile the additional overhead drastically reduces the gains, as shown by Figure 8.1. This overhead substantially increases the optimal beamformer update interval, as Figure 8.3 shows the optimal sounding interval for centralized (zeroforcing) beamforming increasing from 6 ms to 14 ms as the number of users goes from 8 to 32. The additional overhead can also affect the beamformer selection, as the overhead from centralized beamforming can overwhelm the gains; as we see in Figure 8.3, increasing the number of users from 8 to 32 further increases the performance gap between centralized and decentralized beamforming. Note that in Figure 8.3 the achievable rate of zeroforcing is actually much higher than conjugate; it is only due to the computational overhead that conjugate outperforms zeroforcing.

8.2 Kinitos System Design

The Kinitos system rapidly adapts system parameters to optimize performance in diverse real-world scenarios. In deployed many-antenna MU-MIMO systems most of the performance factors discussed in our model, Chapter 4, such as hardware capability, are fixed. However, during runtime the system can still determine how often the channel is estimated by scheduling pilots, the user selection and grouping, which beamformer to use, and how often to update the beamformer. Each of these can drastically affect the performance of the system, and each of them is significantly affected by mobility, thus they need to be decided in realtime.

The Kinitos system consists of 1) a mobility-based pilot scheduler, 2) a mobility-based user selector/grouper, 3) an SNR-based or sub-sampled CSI based beamformer selector, and 4) a gradient-descent based beamformer interval selector.

8.2.1 Pilot Scheduler

Kinitos employs a mobility-based pilot scheduler to efficiently maintain accurate CSI for all active users. The core of the Kinitos pilot scheduler is a mobility detection engine that determines mobility on a *per-user* basis by leveraging CSI when it is available, and Error-Vector Magnitude (EVM) when it is not. The Kinitos pilot scheduler collects CSI for all mobile users at each beamformer update interval, but avoids the unnecessary overhead of collecting CSI for stationary users whose CSI has not changed substantially.

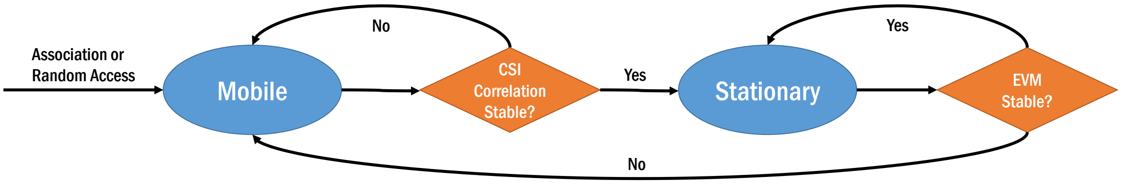


Figure 8.4 : Mobility state diagram.

Mobility Detection Engine

Kinitos determines mobility on a *per-user basis*, using their CSI and the EVM of their beamformed channel. Our extensive channel measurements show that environmental movement, including the movement of other users, typically has minimal impact on a user's performance, thus each user is treated independently. Regardless, our metric does detect environmental mobility, and will classify users as mobile given a strong enough performance impact from this mobility.

We determine user mobility based on the correlation of current CSI with previous CSI, when available, and EVM of the beamformed channel otherwise. As shown in [46], we found that stationary environments are long-term stable and have very little deviation of EVM or correlation. The correlation we use is the instantaneous normalized *correlation* of the user's CSI vector, $\mathbf{h} \in \mathbb{C}^{1\times M}$, between two time points, t_0 and t_f, defined by $C_{f,0} = \frac{|\mathbf{h}_{t_f} \cdot \mathbf{h}^*_{t_0}|}{|\mathbf{h}_{t_f}| \cdot |\mathbf{h}^{\mathrm{T}}_{t_0}|}$. This value is directly related to the signal strength in the beamformed channel, represented by a value ranging between 0 and 1, where 1 represents no change in the channel and 0 represents complete orthogonality from the users' previous location [1, 103, 107, 108].

Note that while stationary users have a stable correlation, there is a constant offset from the expected value of 1. This gap is caused by noise, which causes imperfect correlation. Thus setting a simple threshold on the correlation is not sufficient to robustly determine mobility; fortunately stationary users have constant noise, thus our metric leverages deviation from the mean of the correlation of the last n CSI values to determine mobility at frame f:

$$M_f = \left| C_{f-n,f} - \frac{\sum_{i=1}^{n-1} C_{f-n,f-i}}{n} \right| \tag{8.1}$$

This mobility at frame f, M_f, is then compared to a threshold, γ, to determine if the user should be classified as mobile or stationary. This metric is very functional, as it does not depend on noise or signal strength, and has a constant range of 0 to 1, enabling a constant threshold to work well across environments. Figure 8.5 shows that our metric accurately detects user mobility in a real-world 96 antenna base station serving 4 users at 2.4 GHz.

When CSI is not available, e.g., when the user has determined to be stationary and has not been scheduled a pilot slot, we use the EVM of the downlink beamformed channel to determine mobility. Note that it is important to use the downlink beamformed channel, as the uplink channel for even a stationary user varies based on movement from other users. If the EVM varies beyond some threshold, e.g. 2 dB, then the user is determined to be mobile.

Pilot Scheduling

Once mobility has been determined, the pilot scheduling system is rather straightforward: users that are mobile send pilots at every beamformer update interval, whereas users that are stationary never send pilots. Of course, if a stationary user becomes mobile, they begin sending pilots at regular intervals.

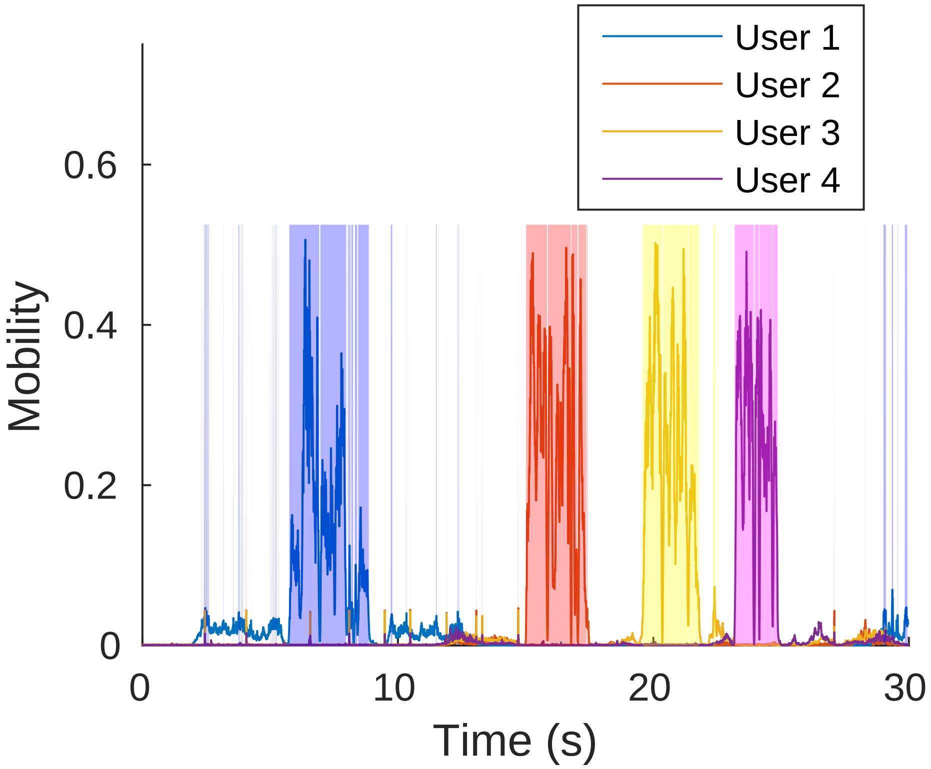


Figure 8.5 : Mobility metric applied to a 96x4 system at 2.4 GHz. Each user is moving, one at a time, and the shaded regions show where the users were determined to be mobile.

8.2.2 Beamformer Selector

Kinitos employs a very simple SNR-based beamformer selector. As discussed in §8.1, as SNR decreases the performance of centralized beamformers approaches the performance of decentralized beamformers. Notably, this transition is relatively gradual, and SNR does not vary rapidly, which enables a simple SNR threshold to work well for determining the beamformer. Thus, Kinitos looks at the average SNR of all users from the previous frame, and if it falls below the threshold it will choose decentralized over centralized beamforming. The threshold selection is, however, related to overhead of centralized beamforming; systems with higher overhead for centralized beamforming will have a higher threshold since the performance of centralized beamforming is reduced. Since centralized overhead varies based on the number of spatial streams, this SNR threshold is implemented as a lookup table based on the number of users being served.

8.2.3 User Grouper and Selector

User grouping and selection is a well researched topic, however in mobile MU-MIMO systems it is important to additionally group users according to their mobility and SNR. As shown in Figure 8.1, if all users are stationary pilots can be suppressed and the beamformer update interval can be greatly extended, essentially eliminating system overhead. However, as soon as even one user in the group is mobile, its CSI has to be updated, as does the beamformer for all users, creating substantial overhead. Similarly, as both the optimal beamformer update interval and beamformer selection is altered by SNR, it is best to group users by SNR to avoid unnecessary overhead.

8.2.4 Beamformer Update Interval Selector

How often the beamformer is updated can drastically affect system performance, and varies substantially based on mobility, SNR, and number of users. Notably, we find the optimal interval varies slowly over time, and system capacity is not sensitive to slight deviations of the interval. Moreover, as shown in Figures 8.1 and 8.3 the system capacity vs. interval function typically has a single inflection point at the maximum and does not have local minimum. Typically this would be a trivial function to optimize, however the difficulty in a real system lies in the lack of full information, particularly future information, as well as continually changing optimum. Thus Kinitos employs a continuous gradient descent to choose the update interval.

That is, Kinitos will begin by increasing the update interval, then check the past two frames to see if system capacity improved or degraded. If performance improved then it will continue increasing the update interval; if performance degraded then it will decrease the update interval, and continue decreasing the interval until performance starts to degrade. In this way, Kinitos converges on the optimal update interval, alternating between the two best, and continuously adapts to changes in mobility and channels.

8.3 Results

We simulate Kinitos on the channel measurements collected in Chapter 7 and compare its performance with a naive system in various configurations as well as an oracle system. While we find the naive systems often perform well in a specific scenario, they cannot adapt to varying mobility, number of users, or SNR. In contrast, Kinitos is capable of achieving within 5% of the oracle in all scenarios.

8.3.1 Trace-Driven Evaluation Setup

We built a comprehensive simulator for real many-antenna MU-MIMO systems that accounts for all of the performance factors outlined in §4.1, and is capable of implementing arbitrary pilot scheduling, beamformer selection, and beamformer update intervals. The simulator leverages channel traces and assumes a fixed minimum frame length, e.g. 2 ms for the Argos measurements, as well as a fixed user grouping and fully backlogged data. At each beamformer update interval the simulator chooses a beamformer, computes the beamforming weights based on the most recently available CSI for each user, then applies the beamweights to the real CSI. The downlink system capacity for each frame is determined by the achievable rate at the end of the frame. This simulates the optimal fixed MCS for an 802.11-like system which only uses frequency interleaving and typically adjusts the MCS at longer timescales than the beamformer update interval. This assumption is pessimistic, since a time-interleaving system could achieve a higher capacity, as could adjusting the MCS on finer time scales or using adaptive coding schemes, but it is applied fairly across schemes and does not affect the relative results.

We compare Kinitos to a naive system, in varying configurations, as well as an oracle system. The naive system has a fixed pilot scheduler, beamformer, and beamformer update interval. The oracle system chooses the optimal pilot schedule, beamformer, and update interval at every frame. Notably the oracle system requires foreknowledge of the channel states, and thus is not possible in a real system, but provides an upper-bound on performance.

When an adequate number of users is not available in traces, the simulator “folds” traces in time to provide more users. However, this only works on mobile users, as static users would have the same CSI, creating orthogonality issues that break MU-MIMO.

It is possible for centralized beamforming systems to perform decentralized beamforming during the time that the centralized beamforming weights are being computed. We do not believe this is a good option for a real system, as it would be fairly complex to implement in practice, and the time would probably be better utilized with uplink data. However, the simulator does separately compute this potential gain, and we show it below in yellow in the system capacity plots.

System Parameters

As discussed in 4.2 and shown in Figure 4.1, the simulated hardware capabilities significantly affect overhead, and therefore centralized beamforming system performance. For these results we use parameters that roughly correspond to somewhere between the super and cluster system configurations described in 4.2: network latency of 20 μs, network throughput of 40 Gbps, and beamformer computational performance that ranges from 200 μs to 1.5 ms depending on number of users and base station antennas.

These values are comparable to those of real systems, e.g., ArgosV3 and the Lund testbed, [28], which reports a 100x10 beamformer processing time of 500 μs. Notably, higher frequency and mobility will reduce channel stability, and thus increase relative overhead, whereas better hardware will reduce overhead. Moreover, different environments and mobility patterns will affect results as well. Therefore these results are useful for identifying trends, comparing relative performance, and highlighting important system design, but should not viewed as general absolute results.

8.3.2 Kinitos Suppresses Overhead in Stationary Topologies

In stationary topologies it is unnecessary to rapidly update users' CSI or the beamformer, so it is best to rarely schedule pilots or update the beamformer in order to avoid unnecessary overhead. Because of this, we see that the naive system with the longest update interval, of 20 ms, performs well, as shown in Figure 8.6. Furthermore, we see that Kinitos adapts well to this scenario, performing within 98% of the oracle.

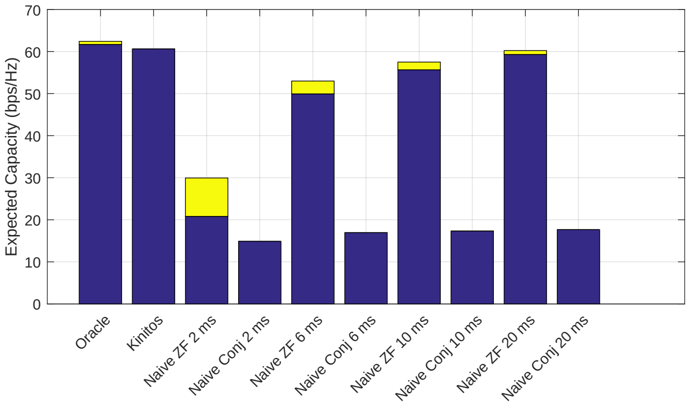


Figure 8.6 : Simulated system performance for 96x6 MU-MIMO systems at 2.4 GHz with all users stationary.

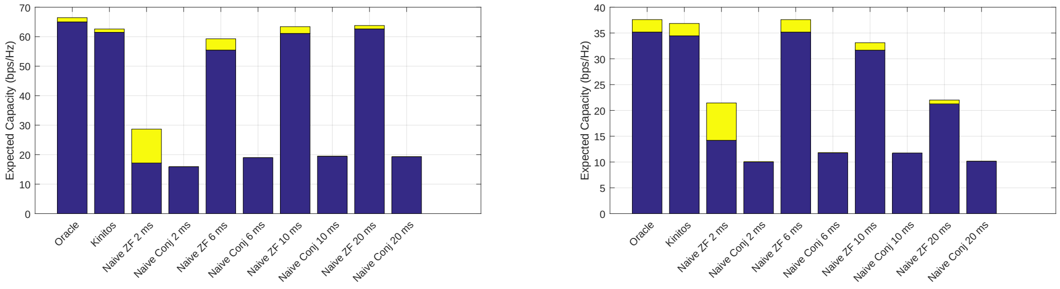


Figure 8.7 : Simulated system performance for 96x8 MU-MIMO systems at 2.4 GHz. *(Left)* 2 users mobile. *(Right)* All 8 users mobile.

8.3.3 Kinitos Adapts to Mobility

As users become mobile, the optimal MU-MIMO system must collect CSI for the mobile users and update the beamformer according to mobility. In Figure 8.7, we see with only two out of eight users mobile, the naive system with the longest update interval still performs well, and even slightly outperforms the Kinitos system. Interestingly, the long update interval is actually sacrificing the performance of mobile users, as the overhead from reducing the update interval would cause the achievable rate of the six stationary users to be reduced. By increasing the number of mobile users to eight, in Figure 8.7, we see that the best naive update interval reduces to 6 ms, and performs comparable to the oracle. Since all users are mobile, their beamformed system capacity degrades rapidly, thus the overhead of collecting CSI and recomputing the beamformer is more than compensated for by the recovered capacity. We still see that Kinitos performs well in these scenarios, performing within 95% and 98% of the oracle for the 2 and 8 mobile user scenarios, respectively.

8.3.4 Kinitos Adapts to Number of Users

As the number of users increases, so does the overhead for centralized beamforming; thus the optimal beamforming update interval increases to overcome the additional overhead. In Figure 8.8 we see that increasing the number of users to 32 causes the 10 ms zeroforcing system to become the most performant naive system. Kinitos performs very well with the increase in number of users, at 99.5% the system capacity of the oracle.

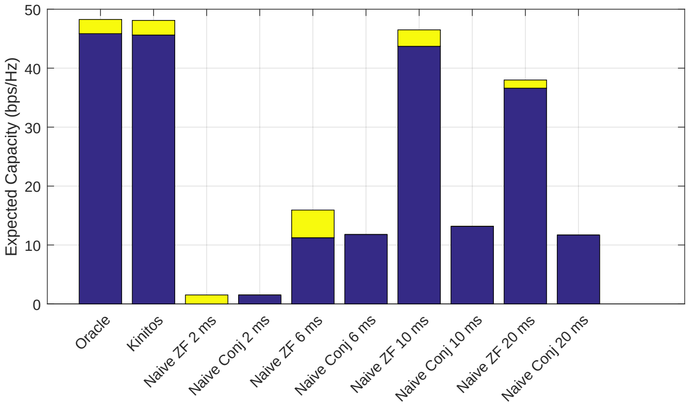


Figure 8.8 : Simulated system performance for 96x32 MU-MIMO systems at 2.4 GHz with all users mobile.

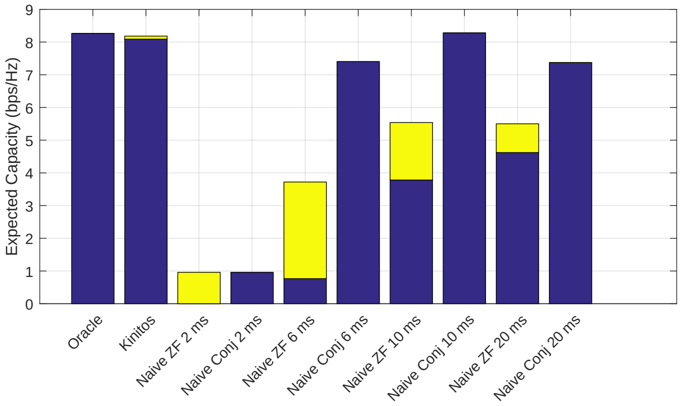


Figure 8.9 : Simulated system performance for 96x32 MU-MIMO systems at 2.4 GHz with all users mobile at very low SNR (0 dB).

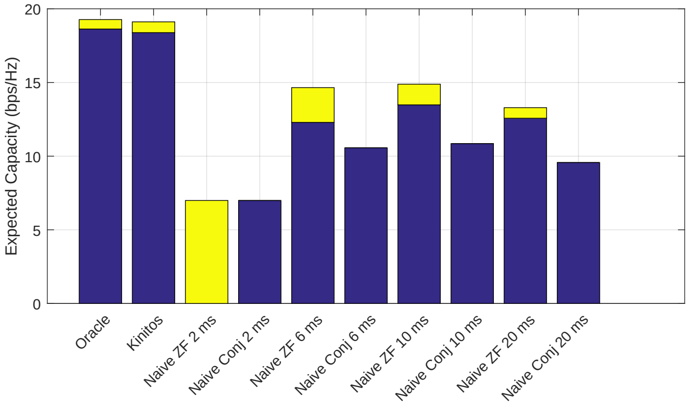


Figure 8.10 : Simulated system performance for 96x16 MU-MIMO systems at 2.4 GHz with mobility ranging from all users stationary to all users mobile, and SNR from 0 dB to 35 dB.

8.3.5 Kinitos Adapts to SNR

As SNR decreases, beamformer selection becomes important. As shown in Figure 8.9, the decentralized conjugate beamformer becomes optimal, significantly outperforming zeroforcing. We see that Kinitos recognizes this, almost always choosing conjugate, and performs within 97.5% of the oracle.

8.3.6 Kinitos Adapts to Varying Scenarios

In the previous scenarios we found that a naive scenario always performed within 10% of the oracle, however it is important to note that in each scenario it was a different naive system; the naive systems are unable to adapt to different scenarios. By simulating a trace that has varying mobility and SNR, we see that none of the naive systems perform well, whereas Kinitos is able to easily adapt. Kinitos performs within 98.7% of the oracle, despite drastically fluctuating SNR, from -3 to 35 dB, and drastically fluctuating mobility, from 0 users mobile to all 16.

8.4 Related Work

While there have been a number of reported works that detect mobility or mitigate the impact of mobility on wireless systems, to the best of our knowledge Kinitos is the first reported work to directly detect mobility in massive-MIMO systems and use it to optimize system parameters such as beamformer selection and interval. Mobility detection is well known in cellular systems for handovers [122], and is even defined in the LTE standard [123], however these typically focus on long time scales, on the order of seconds, and have little to do with beamforming. Mobility has also been extensively studied in small scale wireless systems for rate adaption [124], mitigating the impact on beamforming [94, 125], or even improving performance through spatial diversity [120]. Wireless systems have also been used to detect mobility in the environment through tomographic techniques, e.g., [126, 127], but this has little to do with optimizing massive-MIMO beamforming systems.

8.5 Discussion

Mobility fundamentally limits the system capacity of many-antenna MU-MIMO systems, and, as we have shown, it is critical for MU-MIMO systems to dynamically change system parameters to optimize the real-world achievable rate. Based on our system model and extensive real-world channel traces we devise a novel *mobility-aware* MU-MIMO system that is capable of performing within 95% of an oracle system across a plethora of realistic scenarios.

Chapter 9

Concluding Remarks

Many-antenna MU-MIMO has the potential to drastically improve the performance of wireless systems, however scaling up the number of antennas and simultaneous users creates significant challenges to system design. In particular, traditional hardware design, computational architecture, and control channels can not be efficiently used in next-generation many-antenna systems. Moreover, mobility presents a fundamental limit to the performance of MU-MIMO systems, and this challenge is exacerbated as the number of base station antennas and spatial streams increases.

This thesis presents clean-slate solutions that address these system challenges, which combined create the first practical massive-MIMO base station implementation which is capable of achieving the full potential of many-antenna MU-MIMO in real-world deployments. We modeled the theoretical system capacity limits of many-antenna MU-MIMO in mobile channels, then implemented a novel many-antenna MU-MIMO specific hardware platform which supports distributing the realtime MU-MIMO across individual radio modules, enabling the physical hardware system and computational capabilities to scale to 100s or even 1000s of antennas in practice. We designed and implemented a novel control channel which enables the required power per-antenna at the base station to be reduced proportionally to the square of the number of antennas, enabling very power-, size-, and cost- efficient base stations. Leveraging this many-antenna platform and control channel we conducted extensive channel measurements and analysis to characterize the real-world performance of systems with up to 104 antennas operating from UHF to 5.8 GHz. We demonstrate that mobility is an enormous challenge for many-antenna MU-MIMO, and devise a mobility-aware MU-MIMO system that is capable of achieving within 5% of optimal performance across various realistic mobile scenarios.

As a key candidate technology for 5G and next-generation wireless systems, it is highly likely that many-antenna and massive-MIMO systems will be deployed within the next five years. We believe the techniques and characterization we present in the thesis will be crucial for guiding the adoption of practical and efficient many-antenna systems. Indeed, work is already underway to commercialize many of the innovations presented here [2].

Bibliography

[1] N. Anand, R. E. Guerra, and E. W. Knightly, "The case for uhf-band mu-mimo," in *Proceedings of the 20th annual international conference on Mobile computing and networking*, pp. 29–40, ACM, 2014.

[2] Skylark Wireless LLC. `http://skylarkwireless.com`.

[3] Cisco Systems, Inc., "Cisco visual networking index: Global mobile data traffic forecast update, 20162021 white paper," 2017.

[4] Federal Communications Commission (FCC), "Spectrum analysis: Options for broadcast spectrum," 2010.

[5] Federal Communications Commission (FCC), "Connecting america: The national broadband plan," 2010.

[6] 3GPP, "RP-141831: Study on Elevation Beamforming/Full-Dimension (FD) MIMO for LTE." `ftp://ftp.3gpp.org/tsg_ran/TSG_RAN/TSGR_66/Docs/RP-141831.zip`, 2014.

[7] Qualcomm, "Leading the world to 5G." `https://www.qualcomm.com/media/documents/files/qualcomm-5g-vision-presentation.pdf`, 2016.

[8] K. Mallinson, "The path to 5G: as much evolution as revolution." `http://www.3gpp.org/news-events/3gpp-news/1774-5g_wiseharbour`, 2016.

[9] Y. Kim, H. Ji, J. Lee, Y.-H. Nam, B. L. Ng, I. Tzanidis, Y. Li, and J. Zhang, “Full dimension mimo (fd-mimo): The next evolution of mimo in lte systems,” *IEEE Wireless Communications*, vol. 21, no. 2, pp. 26–33, 2014.

[10] N. Choubey and A. Panah, “Introducing Facebook’s new terrestrial connectivity systems - Terragraph and Project ARIES.” `https://code.facebook.com/posts/1072680049445290`, 2016.

[11] S. Malkowsky, J. Vieira, L. Liu, P. Harris, K. Nieman, N. Kundargi, I. Wong, F. Tufvesson, V. Öwall, and O. Edfors, “The world’s first real-time testbed for massive mimo: Design, implementation, and validation,” *arXiv preprint arXiv:1701.01161*, 2016.

[12] I. B. Collings, H. Suzuki, and D. Robertson, “Ngara broadband access system for rural and regional areas,” *Telecommunications Journal of Australia*, vol. 62, no. 1, 2012.

[13] C. Shepard, H. Yu, N. Anand, E. Li, T. Marzetta, R. Yang, and L. Zhong, “Argos: Practical many-antenna base stations,” in *Proc. Ann. Int. Conf. Mobile Computing & Networking (MobiCom)*, 2012.

[14] C. Shepard, “Argos: Many-Antenna MU-MIMO.” `http://argos.rice.edu`.

[15] G. G. Raleigh and J. M. Cioffi, “Spatio-temporal coding for wireless communication,” *IEEE Transactions on communications*, vol. 46, no. 3, pp. 357–366, 1998.

[16] G. J. Foschini, “Layered space-time architecture for wireless communication in a fading environment when using multi-element antennas,” *Bell labs technical journal*, vol. 1, no. 2, pp. 41–59, 1996.

[17] P. W. Wolniansky, G. J. Foschini, G. Golden, and R. A. Valenzuela, "V-blast: An architecture for realizing very high data rates over the rich-scattering wireless channel," in *Signals, Systems, and Electronics, 1998. ISSSE 98. 1998 URSI International Symposium on*, pp. 295–300, IEEE, 1998.

[18] T. Marzetta, "Noncooperative cellular wireless with unlimited numbers of base station antennas," *IEEE Trans. on Wireless Communications*, 2010.

[19] P. Rudnick, "Digital Beamforming in the Frequency Domain," *Acoustical Society of America Journal*, vol. 46, p. 1089, 1969.

[20] F. Rusek, D. Persson, B. K. Lau, E. G. Larsson, T. L. Marzetta, O. Edfors, and F. Tufvesson, "Scaling up MIMO: Opportunities and challenges with very large arrays," *IEEE Signal Processing Magazine*, vol. 30, no. 1, pp. 40–60, 2013.

[21] E. Larsson, O. Edfors, F. Tufvesson, and T. Marzetta, "Massive MIMO for next generation wireless systems," *IEEE Communications Magazine*, 2014.

[22] L. Lu, G. Y. Li, A. L. Swindlehurst, A. Ashikhmin, and R. Zhang, "An overview of massive MIMO: Benefits and challenges," *IEEE Journal on Selected Topics in Signal Processing*, vol. 8, pp. 742–758, October 2014.

[23] "Samsung takes first 5G steps." `http://www.computerworld.com/article/2497385/data-center/samsung-takes-first-5g-steps-with-advanced-antenna.html`.

[24] F. Boccardi, R. W. Heath, A. Lozano, T. L. Marzetta, and P. Popovski, "Five disruptive technology directions for 5G," *IEEE Communications Magazine*, vol. 52, pp. 74–80, February 2014.

[25] J. G. Andrews, S. Buzzi, W. Choi, S. V. Hanly, A. Lozano, A. C. K. Soong, and J. C. Zhang, "What will 5G be?," *IEEE Journal on Selected Areas in Communications*, vol. 32, pp. 1065–1082, June 2014.

[26] Q. Yang, X. Li, H. Yao, J. Fang, K. Tan, W. Hu, J. Zhang, and Y. Zhang, "BigStation: Enabling scalable real-time signal processing in large MU-MIMO systems," in *Proc. ACM SIGCOMM*, 2013.

[27] Y. Li, Y. Xm, M. Dong, G. Xu, J. C. Zhang, Y. Kim, and J. Lee, "Implementation of full-dimensional MIMO (FD-MIMO) in LTE," in *Proc. IEEE Asilomar Conference*, pp. 998–1003, 2013.

[28] J. Vieira, S. Malkowsky, K. Nieman, Z. Miers, N. Kundargi, L. Liu, I. Wong, V. Owall, O. Edfors, and F. Tufvesson, "A flexible 100-antenna testbed for massive MIMO," in *IEEE GLOBECOM Wrkshp. Massive MIMO: from theory to practice*, 2014.

[29] Nutaq, "TitanMIMO." `http://nutaq.com/en/products/titanmimo`.

[30] S. Perlman and A. Forenza, "pCell: Wireless reinvented." `http://www.rearden.com/artemis/An-Introduction-to-pCell-White-Paper-150224.pdf`, 2014.

[31] Xilinx, "Xilinx and BEEcube announce highly scalable prototyping platform for 5G massive MIMO antenna systems." `http://press.xilinx.com/2015-02-25-Xilinx-and-BEEcube-Announce-Highly-Scalable-Prototyping-Platform-for-5G-Massive-MIMO-Antenna-Systems`.

[32] C. Shepard, N. Anand, and L. Zhong, "Practical Performance of MU-MIMO

Precoding in Many-antenna Base Stations," in *Proc. ACM CellNet Workshop*, 2013.

[33] E. Aryafar, N. Anand, T. Salonidis, and E. Knightly, "Design and experimental evaluation of multi-user beamforming in Wireless LANs," in *Proc. Ann. Int. Conf. Mobile Computing & Networking (MobiCom)*, 2010.

[34] H. Yang and T. Marzetta, "Performance of conjugate and zero-forcing beamforming in large-scale antenna systems," *IEEE Journal of Selected Areas in Communications*, 2013.

[35] H. Ngo, *Performance Bounds for Very Large Multiuser MIMO Systems.* PhD thesis, Linköping University, The Institute of Technology, 2012.

[36] Altera, "Floating-Point Megafunctions User Guide," Nov. 2011. Available at: `www.altera.com/literature/ug/ug_altfp_mfug.pdf`.

[37] Netgear, "PROSAFE 52-Port Gigabit Stackable Switch." Available at: `www.netgear.com/business/products/switches/stackable-smart-switches/GS752TXS.aspx#two`.

[38] "InfiniBand." `www.infinibandta.org`.

[39] Rice University Wireless Open Access Research Platform. `http://warpproject.org`.

[40] E. Research, "Universal Software Radio Peripheral." `https://www.ettus.com/`.

[41] K. Tan, J. Zhang, J. Fang, H. Liu, Y. Ye, S. Wang, Y. Zhang, H. Wu, W. Wang, and G. M. Voelker, "Sora: High Performance Software Radio Using General Purpose Multi-core Processors," in *Proc. USENIX NSDI*, 2009.

[42] C. Shepard, H. Yu, N. Anand, L. Li, T. Marzetta, Y. Yang, and L. Zhong, “Argos: Practical base stations with large-scale multi-user beamforming,” in *ACM MobiCom*, 2012.

[43] OpenAirInterface: 5G software alliance for democratising wireless innovation. `http://www.openairinterface.org/`.

[44] N. Nikaein, R. Knopp, F. Kaltenberger, L. Gauthier, C. Bonnet, D. Nussbaum, and R. Ghaddab, “Demo: Openairinterface: An open lte network in a pc,” in *Proceedings of the 20th Annual International Conference on Mobile Computing and Networking*, MobiCom ’14, (New York, NY, USA), pp. 305–308, ACM, 2014.

[45] The SoapySDR Project. `https://github.com/pothosware/SoapySDR/wiki`.

[46] C. Shepard, J. Ding, R. E. Guerra, and L. Zhong, “Understanding real many-antenna mu-mimo channels,” in *Signals, Systems and Computers, 2016 50th Asilomar Conference on*, pp. 461–467, IEEE, 2016.

[47] G. Radio, “GNU Radio: The Free & Open Software Radio Ecosystem.” `https://www.gnuradio.org/`.

[48] N. Zilberman, Y. Audzevich, G. A. Covington, and A. W. Moore, “Netfpga sume: Toward 100 gbps as research commodity,” *IEEE Micro*, vol. 34, no. 5, pp. 32–41, 2014.

[49] C. Shepard, R. Doost-Mohammady, J. Ding, R. E. Guerra, and L. Zhong, “Argosnet: A multi-cell many-antenna mu-mimo platform,” in *Signals, Systems and Computers, 2017 51st Asilomar Conference on*, IEEE, 2017.

[50] C. Shepard, R. Doost-Mohammady, R. E. Guerra, and L. Zhong, "ArgosV3: An efficient many-antenna platform," in *Extended Demonstration Abstract in Proc. ACM MobiCom*, 2017.

[51] A. Ashikhmin and T. Marzetta, "Pilot contamination precoding in multi-cell large scale antenna systems," in *Proc. IEEE Int. Symp. Information Theory (ISIT)*, pp. 1137–1141, 2012.

[52] J. G. Andrews, W. Choi, and R. W. Heath Jr, "Overcoming interference in spatial multiplexing mimo cellular networks," *IEEE Wireless Communications*, vol. 14, no. 6, p. 95, 2007.

[53] D. Lee, H. Seo, B. Clerckx, E. Hardouin, D. Mazzarese, S. Nagata, and K. Sayana, "Coordinated multipoint transmission and reception in lte-advanced: deployment scenarios and operational challenges," *IEEE Communications Magazine*, vol. 50, no. 2, 2012.

[54] Symmetricom, "Timing and Synchronization for LTE-TDD and LTE-Advanced Mobile Networks," tech. rep., `www.aventasinc.com`, 2013.

[55] C. Shepard, H. Yu, and L. Zhong, "ArgosV2: A flexible many-antenna research platform," in *Extended Demonstration Abstract in Proc. ACM MobiCom*, 2013.

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

[57] X. Gao, F. Tufvesson, O. Edfors, and F. Rusek, "Measured propagation characteristics for very-large mimo at 2.6 ghz," in *Signals, Systems and Computers (ASILOMAR), 2012 Conference Record of the Forty Sixth Asilomar Conference on*, pp. 295–299, IEEE, 2012.

[58] "Bristol and bt collaborate on massive mimo trials for 5g wireless." http://www.bris.ac.uk/news/2017/february/massive-mimo-trials.html, 2017.

[59] Samsung and Nokia Networks, "New SID proposal: Study on elevation beamforming/full-dimension (FD) MIMO for LTE," *http://www.3gpp.org/ftp/tsg_ran/tsg_ran/TSGR_65/Docs/RP-141644.zip*, September 2014.

[60] A. Forenza, S. Perlman, F. Saibi, M. Di Dio, R. van der Laan, and G. Caire, "Achieving large multiplexing gain in distributed antenna systems via cooperation with pcell technology," in *Signals, Systems and Computers, 2015 49th Asilomar Conference on*, pp. 286–293, IEEE, 2015.

[61] R. Rogalin, O. Y. Bursalioglu, H. C. Papadopoulos, G. Caire, A. F. Molisch, A. Michaloliakos, H. V. Balan, and K. Psounis, "Scalable synchronization and reciprocity calibration for distributed multiuser MIMO," *IEEE Transactions on Wireless Communications*, pp. 1815–1831, 2014.

[62] "Samsung takes first 5G steps." http://www.computerworld.com/article/2497385/data-center/samsung-takes-first-5g-steps-with-advanced-antenna.html.

[63] P. Zetterberg, "Experimental investigation of TDD reciprocity-based zero-forcing transmit precoding," *EURASIP Journal on Advances in Signal Processing*, vol. 2011, p. 5, 2011.

[64] K. Nishimori, K. Cho, Y. Takatori, and T. Hori, "Automatic calibration method using transmitting signals of an adaptive array for TDD systems," *IEEE Transactions on Vehicular Technology*, vol. 50, no. 6, pp. 1636–1640, 2001.

[65] K. Nishimori, T. Hiraguri, T. Ogawa, and H. Yamada, “Throughput performance on IEEE 802.11ac based massive MIMO considering calibration errors,” in *Intl. Symp. Antennas and Propagation (ISAP)*, Dec 2014.

[66] E. Perahia and R. Stacey, *Next Generation Wireless LANs: 802.11 n and 802.11 ac*. Cambridge university press, 2013.

[67] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*. Cambridge University Press, 2005.

[68] Cisco Systems, Inc., “Cisco wireless mesh access points, design and deployment guide, release 7.3,” 2012.

[69] E. Magistretti, O. Gurewitz, and E. W. Knightly, “802.11ec: collision avoidance without control messages,” in *Proc. Ann. Int. Conf. Mobile Computing & Networking (MobiCom)*, 2012.

[70] F. Khan, *LTE for 4G mobile broadband: air interface technologies and performance*. Cambridge University Press, 2009.

[71] C. Dick and F. Harris, “FPGA implementation of an OFDM PHY,” in *Proc. IEEE Asilomar Conference*, November 2003.

[72] F. Khan, *LTE for 4G mobile broadband: air interface technologies and performance*. Cambridge University Press, 2009.

[73] A. Touray, “LTE: Random access.” `http://www.lmk.lnt.de/fileadmin/Lehre/Seminar09/Ausarbeitungen/Ausarbeitung_Touray.pdf`, 2009.

[74] P. Murphy and A. Sabharwal, “Design, implementation, and characterization of a cooperative communications system,” *IEEE Transactions on Vehicular*

Technology, vol. 60, pp. 2534–2544, July 2011.

[75] Y. G. Kim and N. Beaulieu, "On MIMO beamforming systems using quantized feedback," *IEEE Transactions on Communications*, vol. 58, pp. 820–827, March 2010.

[76] D. V. Sarwate and M. B. Pursley, "Crosscorrelation properties of pseudorandom and related sequences," *Proceedings of the IEEE*, vol. 68, no. 5, pp. 593–619, 1980.

[77] R. L. Frank, "Polyphase codes with good nonperiodic correlation properties," *IEEE Transactions on Information Theory*, vol. 9, no. 1, pp. 43–45, 1963.

[78] D. Chu, "Polyphase codes with good periodic correlation properties (corresp.)," *IEEE Transactions on information theory*, pp. 531–532, 1972.

[79] Q. Wang, C. Mehlfuhrer, and M. Rupp, "Carrier frequency synchronization in the downlink of 3gpp lte," in *IEEE Int. Synp. Personal Indoor and Mobile Radio Communications (PIMRC)*, pp. 939–944, 2010.

[80] E. Larsson, O. Edfors, F. Tufvesson, and T. L. Marzetta, "Massive MIMO for next generation wireless systems," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 186–195, 2014.

[81] M. Karlsson and E. G. Larsson, "On the operation of massive mimo with and without transmitter csi," in *IEEE Int. Wrkshp. Signal Processing Advances in Wireless Communications (SPAWC)*, pp. 1–5, 2014.

[82] T. Nitsche, A. B. Flores, E. W. Knightly, and J. Widmer, "Steering with eyes closed: mm-wave beam steering without in-band measurement," in *Proc. IEEE*

INFOCOM, 2015.

[83] X. Zhang and K. G. Shin, "E-Mili: energy-minimizing idle listening in wireless networks," *IEEE Transactions on Mobile Computing*, vol. 11, no. 9, pp. 1441–1454, 2012.

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

[85] F. Tufvesson, O. Edfors, and M. Faulkner, "Time and frequency synchronization for OFDM using PN-sequence preambles," in *Proc. IEEE VTC*, vol. 4, pp. 2203–2207, 1999.

[86] H. Balan, R. Rogalin, A. Michaloliakos, K. Psounis, and G. Caire, "AirSync: Enabling distributed multiuser MIMO with full spatial multiplexing," *IEEE/ACM Transactions on Networking*, vol. 21, pp. 1681–1695, Dec 2013.

[87] "Common public radio interface." `http://www.cpri.info/spec.html`.

[88] J. Serrano, M. Lipinski, T. Wlostowski, E. Gousiou, E. van der Bij, M. Cattin, and G. Daniluk, "The white rabbit project." `http://www.ohwr.org/projects/white-rabbit`, 2013.

[89] C. Shepard, A. Javed, and L. Zhong, "Control channel design for many-antenna mu-mimo," in *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, pp. 578–591, ACM, 2015.

[90] B. Hassibi and B. M. Hochwald, "How much training is needed in multiple-antenna wireless links?," *IEEE Transactions on Information Theory*, vol. 49, no. 4, pp. 951–963, 2003.

[91] The HDF Group, "HDF5." `https://www.hdfgroup.org/HDF5/`.

[92] H5PY, "HDF5 for Python." `http://www.h5py.org/`.

[93] Cinetics CineMoco System. `https://cinetics.com/kit/cinemoco_system`.

[94] R. E. Guerra, N. Anand, C. Shepard, and E. W. Knightly, "Opportunistic channel estimation for implicit 802.11af MU-MIMO," in *Proc. of First International Conf. in Networking Science & Practice (ITC)*, 2016.

[95] Scientific Computing Tools for Python. `http://www.numpy.org/`.

[96] Measured Doppler Frequency in Indoor Office Environment , 2009.

[97] J. Wallace, M. A. Jensen, A. L. Swindlehurst, and B. D. Jeffs, "Experimental characterization of the mimo wireless channel: Data acquisition and analysis," *Wireless Communications, IEEE Transactions on*, vol. 2, no. 2, pp. 335–343, 2003.

[98] C. Zhong, M. R. McKay, T. Ratnarajah, and K.-K. Wong, "Distribution of the Demmel condition number of wishart matrices," *IEEE Transactions on Communications*, 2011.

[99] A. A. Sani, L. Zhong, and A. Sabharwal, "Directional antenna diversity for mobile devices: characterizations and solutions," in *Proc. Ann. Int. Conf. Mobile Computing & Networking (MobiCom)*, 2010.

[100] N. Anand, J. Lee, S.-J. Lee, and E. W. Knightly, “Mode and user selection for multi-user mimo wlans without csi,” in *Computer Communications (INFOCOM), 2015 IEEE Conference on*, pp. 451–459, IEEE, 2015.

[101] A. Svensson, “An introduction to adaptive qam modulation schemes for known and predicted channels,” *Proceedings of the IEEE*, vol. 95, no. 12, pp. 2322–2336, 2007.

[102] A. Shokrollahi, “Raptor codes,” *IEEE transactions on information theory*, vol. 52, no. 6, pp. 2551–2567, 2006.

[103] Measured Channel Variation and Coherence Time in NTT Lab. `https://mentor.ieee.org/802.11/dcn/10/11-10-0087-00-00ac-measured-channel-variation-and-coherence-time-in-ntt-lab.ppt`, 2010.

[104] X. Xie, X. Zhang, and K. Sundaresan, “Adaptive feedback compression for MIMO networks,” in *Proc. Ann. Int. Conf. Mobile Computing & Networking (MobiCom)*, 2013.

[105] E. Everett, C. Shepard, L. Zhong, and A. Sabharwal, “Softnull: Many-antenna full-duplex wireless via digital beamforming,” *IEEE Transactions on Wireless Communications*, vol. 15, no. 12, pp. 8077–8092, 2016.

[106] P. Murphy, C. Shepard, L. Zhong, C. Dick, and A. Sabharwal, “Fpgas help characterize massive-mimo channels,” *XceLL Journal*, vol. 89, pp. 18–25, 2014.

[107] Coherence Time Measurement for TGac Channel Model, 2009.

[108] Effect of SDMA in 802.11ac. `https://mentor.ieee.org/802.11/dcn/09/11-09-0303-01-00ac-effect-of-sdma-in-802-11ac.pdf`, 2009.

[109] S. Yoo, S. Kim, Y. Son, J. Yi, and S. Choi, "Practical antenna selection for wlan ap," in *Proc. IEEE INFOCOM*, 2016.

[110] H. Hofstetter, I. Viering, and W. Utschick, "Evaluation of suburban measurements by eigenvalue statistics," in *Proceedings of the 4th COST*, vol. 273, Citeseer, 2002.

[111] X. Gao, F. Tufvesson, O. Edfors, and F. Rusek, "Measured propagation characteristics for very-large mimo at 2.6 ghz," *IEEE Asilomar Conference on Signals, Systems, and Computers*, 2012.

[112] X. Gao, O. Edfors, F. Rusek, and F. Tufvesson, "Massive mimo performance evaluation based on measured propagation data," *Wireless Communications, IEEE Transactions on*, vol. 14, no. 7, pp. 3899–3911, 2015.

[113] J. Hoydis, C. Hoek, T. Wild, and S. Ten Brink, "Channel measurements for large antenna arrays," in *Wireless Communication Systems (ISWCS), 2012 International Symposium on*, pp. 811–815, IEEE, 2012.

[114] À. O. Martínez, E. De Carvalho, and J. Ø. Nielsen, "Towards very large aperture massive mimo: A measurement based study," in *2014 IEEE Globecom Workshops (GC Wkshps)*, pp. 281–286, IEEE, 2014.

[115] J. Flordelis, X. Gao, G. Dahman, F. Rusek, O. Edfors, and F. Tufvesson, "Spatial separation of closely-spaced users in measured massive multi-user MIMO channels," in *Proc. IEEE Int. Conf. Communications (ICC)*, 2015.

[116] T. Ji, C. Zhou, S. Zhou, and Y. Yao, "Low complex user selection strategies for multi-user mimo downlink scenario," in *Wireless Communications and Networking Conference, 2007. WCNC 2007. IEEE*, pp. 1532–1537, IEEE, 2007.

[117] J. Mao, J. Gao, Y. Liu, and G. Xie, “Simplified semi-orthogonal user selection for mu-mimo systems with zfbf,” *IEEE Wireless Communications Letters*, vol. 1, no. 1, pp. 42–45, 2012.

[118] W.-L. Shen, K. C.-J. Lin, S. Gollakota, and M.-S. Chen, “Rate adaptation for 802.11 multiuser mimo networks,” *IEEE Transactions on Mobile Computing*, vol. 13, no. 1, pp. 35–47, 2014.

[119] J. R. Walton, J. W. Ketchum, S. J. Howard, and M. Wallace, “Adaptive rate control for ofdm communication system,” Jan. 16 2007. US Patent 7,164,649.

[120] M. Grossglauser and D. Tse, “Mobility increases the capacity of ad-hoc wireless networks,” in *INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 3, pp. 1360–1369, IEEE, 2001.

[121] B. M. Hochwald, T. L. Marzetta, and V. Tarokh, “Multiple-antenna channel hardening and its implications for rate feedback and scheduling,” *IEEE transactions on Information Theory*, vol. 50, no. 9, pp. 1893–1909, 2004.

[122] T. Sohn, A. Varshavsky, A. LaMarca, M. Chen, T. Choudhury, I. Smith, S. Consolvo, J. Hightower, W. Griswold, and E. De Lara, “Mobility detection using everyday gsm traces,” *UbiComp 2006: Ubiquitous Computing*, pp. 212–224, 2006.

[123] 3rd Generation Partnership Project (3GPP), “Technical specification group radio access network; evolved universal terrestrial radio access (e-utra); user equipment (ue) procedures in idle mode (release 12),” 2015.

[124] J. Camp and E. Knightly, "Modulation rate adaptation in urban and vehicular environments: cross-layer implementation and experimental evaluation," *IEEE/ACM Transactions on Networking*, vol. 18, no. 6, pp. 1949–1962, 2010.

[125] O. Bejarano, R. P. F. Hoefel, and E. W. Knightly, "Resilient multi-user beamforming wlans: Mobility, interference, and imperfect csi," in *Computer Communications, IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on*, pp. 1–9, IEEE, 2016.

[126] F. Adib and D. Katabi, *See through walls with WiFi!* ACM, 2013.

[127] Y. Wang, J. Liu, Y. Chen, M. Gruteser, J. Yang, and H. Liu, "E-eyes: device-free location-oriented activity identification using fine-grained wifi signatures," in *Proceedings of the 20th annual international conference on Mobile computing and networking*, pp. 617–628, ACM, 2014.

[128] C. Shepard, L. Zhong, H. Yu, O. Bejarano, E. Knightly, and L. E. Li, "Argos: Practical massive-mimo." `https://mentor.ieee.org/802.11/dcn/13/11-13-1440-00-0hew-argos-practical-massive-mimo.pptx`, 2013. doc.: IEEE 802.11-13/1440r0.