Practical Performance of MU-MIMO Precoding in Many-Antenna Base Stations

Clayton Shepard*, Narendra Anand*, and Lin Zhong
Rice University, Houston, TX
{cws, nanand, lzhong}@rice.edu
* Equal Contribution

ABSTRACT

Many-antenna base stations promise manyfold spectral capacity increases in theory. However, our recent experimental work has shown a significant performance gap between the traditional MU-MIMO linear precoding method, *zero-forcing*, and the method proposed for many-antenna base stations, *conjugate*. Thus, a critical question in the field of many-antenna base stations is: Under what scenarios, if any, does conjugate precoding outperform zero-forcing in real systems?

Towards answering this question, we leverage our experience in building many-antenna base stations to derive a model for the performance of linear precoders in real-world systems. We isolate the primary factors which discrepantly affect these linear precoders, then capture their complex interactions in an analytical model. By combining our real-world capacity results with this analytical model, we find new insight in to the tradeoffs between conjugate and zero-forcing precoding. Our results suggest that conjugate will outperform zero-forcing when there are many simultaneous users, the users have high mobility, or the implementation employs less-capable hardware. We find that our model is not only useful for guiding the hardware design of base stations, but can also facilitate dynamically switching to the optimal linear precoding algorithm in realtime, through *adaptive precoding*.

Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design—*Wireless Communication*

Keywords

Large-scale Antenna Systems (LSAS), Many-Antenna, Massive MIMO, Multi-User MIMO, Beamforming, Linear Precoding, Conjugate, Zero-forcing

1. INTRODUCTION

Recent work has proposed using many-antenna base stations to vastly improve spectral capacity in cellular networks by serving tens of users simultaneously. However, traditional linear precoding techniques do not scale up well with the number of antennas. For example, the predominant multi-user multiple input multiple output (MU-MIMO) linear precoding technique, *zero-forcing*, leverages a pseudo-inverse of the channel matrix to nullify 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 increases it approaches optimal [5]. 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 [8].

Unfortunately, our recent experimental work has shown that even with a substantial number of base station antennas that conjugate performs significantly worse than zero-forcing. For example, it only achieves 45% capacity with 64 base station antennas [8]. However these results only indicate the channel capacity *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 conjugate precoding outperform zero-forcing 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 capacity of linear precoding techniques in realtime systems. Using results from our implementation of a many-antenna base station, we leverage this model to identify and investigate the tradeoff points at which conjugate can outperform zero-forcing. We find that in a low-end, cost-effective, base station conjugate outperforms zero-forcing 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,

CellNet'13, June 25, 2013, Taipei, Taiwan
Copyright 2013 ACM 978-1-4503-2074-0/13/06 ...$15.00.

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 capacity, which we call *adaptive precoding*.

The rest of this paper is organized as follows: We provide a brief background in Section 2. In Section 3 we discuss the factors which affect performance, then use them to build a performance model in Section 4. We leverage this model to predict tradeoff points between the precoding techniques, which we present with other results in Section 5. In Section 6 we discuss future work, followed by a brief overview of related work in 7, then conclude in Section 8.

2. BACKGROUND

There are many forms of MU-MIMO; we focus on *linear precoding* since other methods are computationally infeasible in practice, or do not take advantage of the potential capacity gains from many-antenna systems. Let $\mathbf{s}$ denote a $K \times 1$ vector representing the data-bearing symbols to K users. Linear precoding creates a downlink transmission vector $\mathbf{s}'$ for M antennas, by multiplying the original data vector $\mathbf{s}$ by a $M \times K$ matrix $\mathbf{W}$: $\mathbf{s}' = \mathbf{W} \cdot \mathbf{s}$. In the uplink the data symbols from the K terminals can be recovered similarly, by performing $\mathbf{s} = \mathbf{W^T} \cdot \mathbf{s}'$.

The beamforming weights, $\mathbf{W}$, are computed according to the precoding algorithm; in this work we analyze the two predominant algorithms: *conjugate* and *zero-forcing*. Conjugate uses beamforming weights which are the complex conjugate of the channel matrix, $\mathbf{H}$, $\mathbf{W}_{conj} = c\mathbf{H}^*$, where $\mathbf{H}^*$, which maximizes the SNR to each user, regardless of interference. Zero-forcing calculates the beamweights as a pseudo-inverse of the channel matrix, $\mathbf{W}_{zf} = c\mathbf{H}^* \left(\mathbf{H}^{\mathrm{T}}\mathbf{H}^*\right)^{-1}$, which forces inter-user interference to zero.

For more detailed background, we suggest [5, 8, 3].

3. PERFORMANCE FACTORS

The factors which affect the performance of base stations employing linear precoding can be classified as either environmental or by 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 zero-forcing precoding, and characterize them in real-world systems.

3.1 Environmental Factors

3.1.1 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 [2]. Previous work based on LTE channel models often use approximately 1 ms coherence times [5].

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 spectrum to obtain accurate CSI. In LTE models the coherence bandwidth is 210 kHz, as described in further detail in [5].

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 capacity. More importantly, however, this overhead is fixed with respect to channel coherence time. Thus, as channel coherence is reduced, the relative capacity loss grows. Since conjugate and zero-forcing have drastically different computational overheads they behave differently as coherence time varies.

3.1.2 Precoder Spectral Efficiency

Zero-forcing and conjugate provide vastly different spectral efficiencies during actual data transmissions [8]. 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 zero-forcing varies significantly with SNR, as further explored in [9, 8]. However, zero-forcing 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 zero-forcing, but performs much better at low SNRs, as shown nicely in [7]. While the relative performance to conjugate still varies with SNR, it is not as drastic.

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 capacity.

3.2 Design Factors

3.2.1 Number of Antennas

The number antennas, both on the base station or with each additional user, drastically affects the 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 zero-forcing, 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 grow at different rates with the number of base station antennas and users. Since zero-forcing has polynomial unparallelizable complexity, it suffers more 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.

3.2.2 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 capacity 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 zero-forcing 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, zero-forcing 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, zero-forcing 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 zero-forcing 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).

Additionally, 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 [1] 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 40 Gbps. Along with inversion latency, data transport latency significantly detracts from the performance of zero-forcing 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 [6]. 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 [4]. For the round trip transportation of 191 15x15 matrices, this system incurs a latency of approximately 70 μs.

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 capacity	bps/hz

Table 1: Parameters. Upper set are model inputs categorized by environment and design. Lower set are model variables.

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. Practical communication systems use wide channel bandwidths in order to increase capacity. Unfortunately, as mentioned above, the frequency response of this channel is not flat, thus CSI estimation and precoding computation has to be repeated at regular intervals across the band. Thus, the number of inverses and amount of data transport required 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. 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.

4.1 Parameters

A list of model parameters, sorted by their category, environment or design, is shown in Table 1. If a value is specific to a precoding technique it is denoted with a $_{ZF}$ or $_{C}$ for zero-forcing and conjugate, respectively.

4.2 Model Derivation

The goal of this model is to find the real-world achieved capacity of a linear precoding system when given the channel coherence, number of base station antennas, number of users, hardware capability, precoder spectral capacity, 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:

$$\Theta = \theta \cdot \gamma \cdot K \tag{1}$$

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

$$\gamma = \frac{C_t - E - P}{C_t} \tag{2}$$

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:

$$E = \frac{K}{C_b} \tag{3}$$

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

$$P_{ZF} = 2 \cdot \left(\frac{M \cdot K \cdot \frac{B}{C_b} \cdot N_b}{S} + L \right) + \frac{B}{C_b} \cdot T_{\text{-}1} \tag{4}$$

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.

4.3 Complete Model

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

$$\Theta_C = \frac{C_t - \frac{K}{C_b}}{C_t} \cdot \theta_C \cdot K \tag{5}$$

And for zero-forcing is:

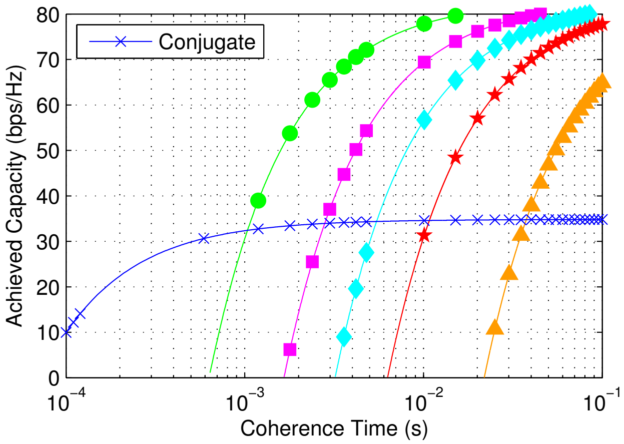
$$\Theta_{ZF} = \frac{C_t - \frac{K}{C_b} - \left(2 \cdot \left(\frac{M \cdot K \cdot \frac{B}{C_b} \cdot N_b}{S} + L \right) + \frac{B}{C_b} \cdot T_{\text{-}1} \right)}{C_t} \cdot \theta_{ZF} \cdot K \tag{6}$$

5. 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 zero-forcing cross, as they highlight the conditions when it is important to consider the tradeoffs between the two precoding techniques.

5.1 Simulation Methodology

Using the performance model described in Section 4, we input a range of realistic parameter values and analyze their impact on performance. As defined in Table 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



	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 1: Zero-forcing and conjugate performance comparison for different hardware configurations in a M=64, K=15 system.

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 [8] we collect the real-world spectral efficiency, θ, achieved by conjugate and zero-forcing 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_{-1}, on 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 1 [6, 4]. We assume that processing is local, and thus propagation delay is negligible.

5.2 Results

The main factors which affect the performance tradeoffs between conjugate and zero-forcing 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 zero-forcing.

5.2.1 Coherence Time and Hardware Capability

We first look at the achieved capacity of conjugate and zero-forcing with regard to coherence time. Figure 1 shows that while serving 15 users simultaneously, conjugate beamforming outperforms zero-forcing 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 zero-forcing dominates its capacity.

However, we can also see in Figure 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.

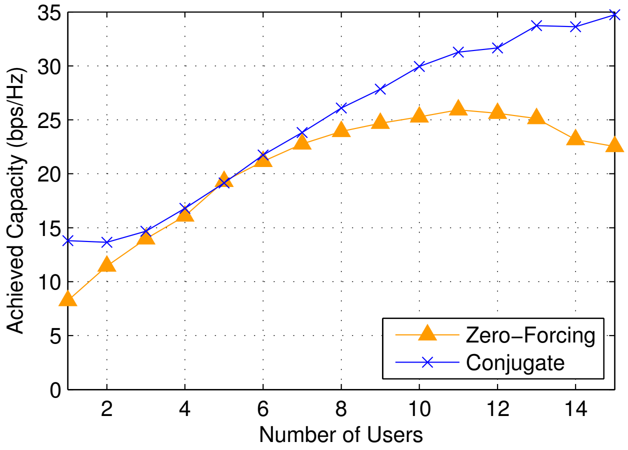


Figure 2: Zero-forcing and conjugate performance comparison for number of terminals and fixed coherence time of 30 ms with low-end hardware.

5.2.2 Number of Users

Finally, we note that as the number of users grows, the performance of zero-forcing quickly degrades under the constraint of low coherence times, as the overhead from data-transport and processing dominate its capacity. Figure 2 demonstrates a scenario where conjugate begins to outperform zero-forcing with more users; with 4-6 users their performance is equivalent, but as the number of users grows to 15, zero-forcing achieves only 65% the capacity of conjugate. This also demonstrates the criticality of choosing the optimal number of users to serve, as the capacity of zero-forcing 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 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.

5.3 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 zero-forcing 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 zero-forcing system for an environment with very short coherence time.

Adaptive Precoding. 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 zero-forcing through adaptive precoding. Since users exhibit widely varying mobility, their coherence time may drop below the threshold where zero-forcing 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.

6. DISCUSSION AND FUTURE WORK

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, 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.

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 [9], Yang et al. analyze the radiated power and computational requirements of conjugate and zero-forcing 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 zero-forcing. 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.

8. CONCLUDING REMARKS

Many-antenna base stations show enormous potential in multiplying the spectral capacity of wireless systems. However it is imperative to discover and understand at the real-world factors which affect their performance in order design systems which achieve their potential capacity gain. We have analyzed and described the critical system factors which discrepantly affect the performance of the two predominant linear precoders envisioned for many-antenna beamforming. Contrary to some existing theoretical theoretical analysis, our results indicates that conjugate beamforming likely outperforms zero-forcing in many realistic scenarios. Our robust model can not only be used to help guide system design and provisioning, but also indicates that base stations can greatly benefit from adaptive precoding, enabling them to dynamically switch to the optimal precoding technique as the users and environment vary.

Acknowlegements

This work was funded in part by NSF grants CRI 0751173, MRI 0923479, NetSE 101283, MRI 1126478 and CNS 1218700. Clayton Shepard was supported by an NDSEG fellowship. We thank Ashutosh Sabharwal, Edward Knightly, Chris Hunter, and Patrick Murphy for their input and support.

References

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

[2] E. Aryafar, N. Anand, T. Salonidis, and E. Knightly. Design and experimental evaluation of multi-user beamforming in Wireless LANs. In *Proc. ACM MobiCom*, 2010.

[3] F. Fernandes, A. Ashikhmin, and T.L. Marzetta. Inter-cell interference in noncooperative TDD large scale antenna systems. *IEEE Journal on Selected Areas in Communications*, 2013.

[4] InfiniBand. Available at: www.infinibandta.org.

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

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

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

[8] C. Shepard, H. Yu, N. Anand, E. Li, T. Marzetta, R. Yang, and L. Zhong. Argos: Practical many-antenna base stations. In *Proc. ACM MobiCom*, 2012.

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