

# **Practical Interference Mitigation for Wi-Fi Systems**

Georgios Nikolaidis

A dissertation submitted in partial fulfillment  
of the requirements for the degree of  
**Doctor of Philosophy**  
of  
**University College London.**

Department of Computer Science  
University College London

2016



I, Georgios Nikolaidis, confirm that the work presented in this thesis is my own. Where information has been derived from other sources, I confirm that this has been indicated in the work.



# Abstract

Wi-Fi’s popularity is also its Achilles’ heel since in the dense deployments of multiple Wi-Fi networks typical in urban environments, concurrent transmissions interfere. The advent of networked devices with multiple antennas allows new ways to improve Wi-Fi’s performance: a host can *align* the phases of the signals either received at or transmitted from its antennas so as to either maximize the power of the signal of interest through *beamforming* or minimize the power of interference through *nulling*. Theory predicts that these techniques should enable concurrent transmissions by proximal sender-receiver pairs, thus improving capacity.

Yet practical challenges remain. Hardware platform limitations can prevent precise measurement of the wireless channel, or limit the accuracy of beamforming and nulling. The interaction between nulling and Wi-Fi’s *OFDM* modulation, which transmits tranches of a packet’s bits on distinct subcarriers, is subtle and can sacrifice the capacity gain expected from nulling. And in deployments where Wi-Fi networks are independently administered, APs must efficiently share channel measurements and coordinate their transmissions to null effectively.

In this thesis, I design and experimentally evaluate beamforming and nulling techniques for use in Wi-Fi networks that address the aforementioned practical challenges. My contributions include:

- *Cone of Silence* (CoS): a system that allows a Wi-Fi AP equipped with a phased-array antenna but only a *single* 802.11g radio to mitigate interference from senders other than its intended one, thus boosting throughput;
- *Cooperative Power Allocation* (COPA): a system that efficiently shares channel measurements and coordinates transmissions between independent APs, and cooperatively allocates power so as to render received power across OFDM subcarriers flat at each AP's receiver, thus boosting throughput;
- *Power Allocation for Distributed MIMO* (PADM): a system that leverages intelligent power allocation to mitigate inter-stream interference in distributed MIMO wireless networks, thus boosting throughput.

# Acknowledgements

Επιστήμη ποιητική ευδαιμονίας.

---

Plato

The safest general characterization of the European philosophical tradition is that it consists of a series of footnotes to Plato.

---

A.N. Whitehead

It always used to strike me as odd that the name for the terminal academic degree is “*Doctor of Philosophy*”, independently of the field of study. What I found out during this long process is that a Ph.D., though a contribution to science, is more importantly a lesson of epistemology, hence philosophy. Considering this, I could not have been any luckier than to have Brad Karp as advisor and teacher. His intellect, rigor, knowledge, integrity, empathy, intuition and humor have made me a better scientist, philosopher and person.

Kyle Jamieson always kept his door open, gave me his guidance through the flurry of work in the wireless area and made this work better in immeasurable ways.

Mark Handley has been an inspiration since the start of my research career. His ability to reach the heart of a problem never ceased to amaze me. More importantly, his tenacious belief in me, even during the darkest hours, has been pivotal for reaching the end.

Jie Xiong was always helpful and gave me the morning shifts. Adam Greenhalgh, Nikola Gvozdiev, Petr Marchenko, Konstantinos Nikitopoulos, Piers O'Hanlon, Costin Raiciu, Lynne Salameh, Socratis Varakliotis, Astrit Zhushi have all been great colleagues, more importantly friends and I thank them for going through several iterations of my talks and papers.

My parents, Evanthia and Ioannis, have given me constant love and support, but more than that, instilled me with love for knowledge and appreciation for scholarship. My brothers, Nikos and Apostolis, have been a source of inspiration since I can remember myself to this day. I could not have been any more fortunate with my family.

Carly has been my partner from the first day of this endeavor. She put up with the long hours, lack of holidays, working weekends, frustrations, disappointments and me. This is one of the many reasons I admire her. Without her encouragement, love, companionship and relentless backing the following pages would be blank.

# Contents

<b>1</b>	<b>Introduction</b>	<b>23</b>
1.1	Background . . . . .	25
1.2	Problem Statement . . . . .	29
1.3	Contribution . . . . .	36
<b>2</b>	<b>Background and Related Work</b>	<b>41</b>
2.1	Exclusive Resource Allocation . . . . .	42
2.2	Carrier Sense and Collision Avoidance . . . . .	44
2.3	Interference Cancellation . . . . .	46
2.4	Antenna Directionality . . . . .	50
2.5	Switched Beam Antennas . . . . .	52
2.6	Adaptive Beamforming . . . . .	54
2.7	Interference Alignment and Interference Nulling . . . . .	57
2.8	Distributed MIMO . . . . .	61
2.9	Frequency Division . . . . .	64
2.10	Power Allocation . . . . .	67
2.11	Bit Loading and Subcarrier Switch-Off . . . . .	69
2.12	Conclusion . . . . .	71
<b>3</b>	<b>Cone of Silence (CoS)</b>	<b>75</b>

3.1	Design . . . . .	80
3.1.1	Use Scenario . . . . .	81
3.1.2	Hardware Platform . . . . .	82
3.1.3	Measuring the Channel: SamplePhase . . . . .	84
3.1.4	The SamplePhase estimator . . . . .	87
3.1.5	Nulling Interferers: Silencer . . . . .	92
3.2	Evaluation . . . . .	95
3.2.1	Beamforming with SamplePhase . . . . .	99
3.2.2	Nulling Interferers with Silencer . . . . .	102
3.2.3	Longevity of Interference-Nulling Patterns . . . . .	106
3.2.4	Nulling Multiple Interferers . . . . .	107
3.3	Discussion . . . . .	109
<b>4</b>	<b>Cooperative Power Allocation (COPA)</b>	<b>111</b>
4.1	Problem . . . . .	114
4.1.1	OFDM and Narrow-Band Fading . . . . .	116
4.1.2	Nulling in Practice: Residual Interference . . . . .	118
4.2	Design . . . . .	121
4.2.1	Coordination Protocol . . . . .	121
4.2.2	Per-subcarrier Power Allocation . . . . .	130
4.2.3	Predicting the Best Strategy . . . . .	136
4.2.4	Overconstrained Nulling . . . . .	139
4.2.5	Incentive Compatibility . . . . .	141
4.3	Evaluation . . . . .	141
4.3.1	Experimental Methodology . . . . .	143
4.3.2	Single Antenna Scenario . . . . .	148
4.3.3	Constrained Nulling Scenario . . . . .	151
4.3.4	Nulling with Weaker Interference . . . . .	153

4.3.5	Overconstrained Scenario . . . . .	155
4.3.6	Multiple Decoders . . . . .	157
4.3.7	Contributions of Features of EquiSINR . . . . .	158
4.4	Discussion . . . . .	160
<b>5</b>	<b>Power Allocation for Distributed MIMO (PADM)</b>	<b>161</b>
5.1	Problem . . . . .	162
5.1.1	Distributed MIMO System Principles . . . . .	165
5.1.2	Scaling Wireless Throughput . . . . .	167
5.2	Design . . . . .	171
5.3	Evaluation . . . . .	173
5.3.1	Experimental Setup . . . . .	174
5.3.2	Overall Throughput Improvement . . . . .	177
5.3.3	Choosing the Optimum Number of Clients . . . . .	179
5.3.4	Improvement at the Maximum . . . . .	181
5.3.5	Throughput Variability Across Streams . . . . .	182
5.3.6	Variability Across Topologies and Scheduling . . . . .	186
5.4	Discussion . . . . .	189
<b>6</b>	<b>Conclusion</b>	<b>191</b>
6.1	Future Work and Closing Remarks . . . . .	196
<b>Appendices</b>		<b>198</b>
<b>A</b>	<b>Mathematical Supplement</b>	<b>199</b>
<b>Bibliography</b>		<b>200</b>



# List of Figures

1.1	Theoretical UDP throughput when sending 1500 byte packets for the bitrates used in 802.11n. Higher bitrates require higher SINR in order to achieve non-zero throughput. Lines with the same color use the same modulation; line style denotes coding rate. . . . .	27
1.2	If host A cannot detect host C's carrier, it might try to send a packet to host B while host C is transmitting to host D. B will be unable to decode A's transmission . . . . .	29
1.3	Standard 802.11 chipsets do not give access to per subcarrier channel state information. The Intel 5300 provides this information and allows setting phase and amplitude for pairs of subcarriers while SDR—and potentially future chipsets—provide data for each subcarrier and allow full manipulation of the per-subcarrier phase and magnitude of the transmitted signal. . . . .	33
2.1	In TDMA, a centralized controller splits time into slots and assigns them to users. If a slot goes unused by its user, no other user can use it. . . . .	43

2.2	Signal 1+2 is the sum of Signal 1 and Signal 2. A host can decode Signal 2, subtract it from Signal 1+2 and retrieve Signal 1. . . . .	46
2.3	The time offset between frames in successive collisions is usually different because of the random exponential back-off mechanism. . . . .	48
2.4	Radiation pattern of a directional antenna. The thick line shows the gain in dB compared to the maximum seen in the largest lobe (at 67.5°) for all the angles on the azimuth plane. . . . .	51
2.5	In an indoor environment, a signal from host S can arrive at its destination R from multiple directions as above where the signal is received from paths P1, P2, and P3. R can maximize the strength of the received signal by choosing an appropriate pattern such as the one depicted. If the array uses a switched beam scheme, the depicted pattern will not be available and R might be forced to choose some other suboptimal one. Furthermore, interference from host I arrives from paths P4 and P5. Placing nulls in these directions would significantly reduce interference. . . . .	55
2.6	The first two versions of the signal will result in a combination that will be stronger while signals $a$ and $c$ would cancel each other out. $\delta$ is the phase difference of the two signals, i.e., the time difference between them. . . . .	58
3.1	Typical dense 802.11 deployment in apartments or offices. AP1, AP2, and AP3 are access points; C1, C2, and C3 are clients. Solid arrows denote desired transmissions; dashed arrows denote unintended interference. . . . .	76

3.2	Example of multipath propagation between a sender $S$ , interferer $I$ , and a receiving directional antenna. Solid arrows represent components from $S$ ; dashed from $I$ . Boundaries are reflective walls; the triangle represents a reflective object.	78
3.3	<i>Left:</i> Phocus phased array 802.11b/g antenna with eight elements. <i>Right:</i> simplified array model. . . . .	83
3.4	Empirical power measurements ( $P_{kl}$ ) of RSS for three representative element pairs for an AP-remote node link when pairs of AP antenna elements transmit simultaneously with varying phase difference ( $\delta$ ) to a remote node (“Empirical” points). The SamplePhase and Lakshmanan <i>et al.</i> estimates of the peak of the sinusoid appear as vertical lines. . . . .	89
3.5	Empirical data with the theoretical fit superimposed. . . . .	91
3.6	By projecting the received signal $\mathbf{y}$ onto the vector subspace orthogonal to an interferer’s channel $\mathbf{h}_i$ , Silencer (shown here in $\mathbb{R}^2$ for ease of exposition) nulls the signal from the interfering client. . . . .	93
3.7	Impact of Silencer on antenna gain pattern: note that these empirical figures show gain <i>vs.</i> direction, but do not show how the antenna manipulates the phase of the received signals. <i>Left:</i> an MRC gain pattern maximizing SNR for a sender. <i>Center:</i> an MRC gain pattern maximizing SNR for an interferer. <i>Right:</i> the resulting Silencer gain cancelling the interferer and beamforming toward the sender. . . . .	93
3.8	The indoor office environment and wireless network topology for the experiments in CoS. Filled dots represent nodes with a single omnidirectional antenna and hollow dots represent nodes with a phased-array antenna. . . . .	96

3.9 Empirically measured throughput for six different antenna gain patterns across six different testbed links (labeled by sender identifier and access point identifier: <i>cf.</i> Figure 3.8 on p. 96). In the absence of interference, SamplePhase offers the greatest throughput because of the accuracy of its channel measurement method. . . . .	99
3.10 RSS distributions of packets drawn from fixed bitrate (54 Mbit/s) experiments on testbed link 2A. . . . .	101
3.11 Bitrates chosen by SampleRate during a representative 30-second interval at link 6A ( <i>cf.</i> Figure 3.9). SamplePhase chooses significantly higher bit rates than other methods, choosing the top two bitrates (48 and 54 Mbps) 80% of the time. . . . .	102
3.12 Empirically measured throughput in the presence of one interferer, for six different antenna gain patterns across six different testbed links (labeled by sender identifier, access point identifier and interferer identifier: <i>cf.</i> Figure 3.8 on p. 96). By nulling, Silencer yields the best end-to-end throughput in the presence of interference. . . . .	105
3.13 Longevity of a Silencer interference-nulling pattern. We run the SamplePhase measurement method only once, at the beginning of this time series. Then, as the wireless channel changes, we measure throughput in the presence of a continuous interferer. We see that the Silencer pattern's throughput does not degrade significantly for approximately 45 minutes. . . . .	107

3.14 Empirically measured throughput in the presence of two concurrent interferers (for a total of three concurrent transmissions), over multiple testbed links (labeled by sender identifier, access point identifier and both interferers' identifiers: <i>cf.</i> Figure 3.8 on p. 96). Adding channel measurements for each additional interferer and incorporating that interferer into Silencer's nulling algorithm yields cumulative additional benefit. . . . .	108
4.1 AP1 and AP2 transmit concurrently to clients C1 and C2, respectively; each AP interferes with the other AP's transmission to that other AP's client. . . . .	115
4.2 Received power from a single send antenna at two different receive antennas, by OFDM subcarrier index. . . . .	116
4.3 End-to-end effect of nulling on SINR, SNR and INR; 30 indoor office topologies; two four-antenna APs sending to two 2-antenna clients. Error bars denote one standard deviation. .	119
4.4 Per-subcarrier effects of nulling; two four-antenna APs sending to two two-antenna clients. . . . .	120
4.5 A timeline of COPA MAC operation. ITS REQ frames include CSI from Follower AP to both clients, while ITS ACK frames include the precoding matrix for the Follower AP. .	124
4.6 MAC-layer ACKs in the case of sequential and concurrent transmission. . . . .	126
4.7 Formats of the different ITS frames. "FC" is the Frame Control Field, "D" is the Duration, "MAC Ad" Is a MAC address, "Payload" contains CSIs, and FCS is the Frame Check Sequence. . . . .	127

4.8	Example of EquiSNR. . . . .	133
4.9	COPA's <i>EquiSINR</i> iterative power allocation. . . . .	136
4.10	BER per subcarrier without coding when using the same nulling precoding matrix and bitrate (39 Mbps) with COPA and without ("NoPA"). Vertical bars denote subcarriers that COPA drops. . . . .	137
4.11	Overview of COPA . . . . .	138
4.12	AP and client locations in the 7th floor of the UCL CS building.	143
4.13	Signal power from each client's own AP plotted against interfering signal power from the other AP; each point is one receiver. . . . .	145
4.14	Throughput CDF (across topologies) for two single-antenna AP, single-antenna client pairs. . . . .	149
4.15	Throughput CDF (across topologies) for two four-antenna AP, two-antenna client pairs. . . . .	152
4.16	Throughput CDF (across topologies) for two four-antenna AP, two-antenna client pairs, when interference is 10 dB <i>weaker</i> than empirically measured in our testbed. . . . .	154
4.17	Throughput CDF (across topologies) for two three-antenna AP, two-antenna client pairs. . . . .	156
4.18	Potential percentage improvements from using per-stream error control coding with CSMA and COPA. . . . .	158
4.19	Throughput CDF (across topologies) for two four-antenna AP, two-antenna client pairs with concurrent transmissions and both APs nulling. . . . .	159
5.1	Enterprise network with multiple APs. Each AP transmits to a single host. . . . .	163

5.2	Distributed MIMO system. All APs transmit to all hosts concurrently. . . . .	164
5.3	Receivers need at least one symbol time's worth of overlap to decode a transmission. . . . .	166
5.4	Floorplan of our system. The blue dots in the periphery show transmitting antenna pairs and the red dots indicate client positions. . . . .	168
5.5	Aggregate throughput of 20 antenna Distributed MIMO system with varying number of transmitted streams. Mean of 10 topologies with the errorbars showing $\pm 1$ standard deviation. . . . .	169
5.6	SNR (top) and SINR (bottom) for the same stream as the number of concurrently transmitted streams increases. The SNR values are normalized for transmit power. . . . .	170
5.7	Standard deviation of SINR within the same stream for 50 topologies as the number of transmitted streams increases. . .	171
5.8	BER per subcarrier without coding when using the same nulling precoding matrix and bitrate (26 Mbps) with PADM and without ("Standard"). Vertical bars denote subcarriers that PADM drops. . . . .	172
5.9	Carrier frequency estimation error. . . . .	175
5.10	Aggregate throughput for Distributed MIMO systems with different number of transmit antennas for varying number of transmitted streams. . . . .	178
5.11	Throughput improvement when using PADM versus standard power allocation. . . . .	179

5.12 Number of clients (top) and value of $b$ (bottom) at maximum aggregate throughput as a function of number of antennas of the virtual AP. . . . .	180
5.13 Improvement when using PADM with the optimal number of streams against standard power allocation with optimal number of streams (top) and $b = 2$ (bottom). . . . .	181
5.14 Mean of standard deviation of throughput across streams for standard power allocation (top) and PADM (bottom). . . . .	183
5.15 Mean of maximum of throughput across streams for standard power allocation (top) and PADM (bottom). . . . .	184
5.16 Mean of minimum of throughput across streams for standard power allocation (top) and PADM (bottom). . . . .	185
5.17 Aggregate throughput distribution across 100 random client topologies for the same 10-antenna virtual AP. . . . .	186
5.18 Aggregate throughput when using the best out of a set number of permutations. Notice that y-values start at 200Mbps. . .	188
5.19 Aggregate throughput improvement when using the best out of a set number of splits. . . . .	189

# List of Tables

1.1	Comparison scenarios in terms of degree of centralization and hardware capability. . . . .	35
2.1	Comparison of selected systems and methods on lower-level goals set for Scenario A. . . . .	72
2.2	Comparison of selected systems and methods on lower-level goals set for Scenario B. . . . .	73
2.3	Comparison of selected systems and methods on lower-level goals set for Scenario C. . . . .	74
3.1	Mean and standard deviation of absolute error for Sample- Phase and L. <i>et al.</i> methods. . . . .	90
3.2	Summary of experimental results for the proposed tech- niques, and the corresponding conclusion or performance improvement. . . . .	98
3.3	Single-interferer experiments: power levels used at sender and interferer. . . . .	104
4.1	Throughput costs incurred by MAC overhead. . . . .	128
4.2	Summary of experimental results for the proposed tech- niques, and the corresponding conclusion or performance improvement. . . . .	142



## Chapter 1

# Introduction

It was in the early 80s that John Cage, an employee of Sun Microsystems, coined the phrase "*the network is the computer*". In light of the past two decades, this has proven to be an emblematic quote, stating a vision that came to be a reality. Since the 90s, the combination of personal computers and the Internet has spurred numerous applications and novel uses.

More recently, there has been a shift in the technologies used in computer networking. Although *IEEE 802.3* [1], commercially known as *Ethernet*, still predominates in *Local Area Networks* (LANs), its wireless equivalent continues to gain popularity. *IEEE 802.11* [2], commercially known as *Wi-Fi*, is used for the same purpose of interconnecting devices at a local level. Its advantages over Ethernet are:

- Lack of need for wiring which, especially in a domestic environment, might be troublesome and undesirable for aesthetic reasons.
- Low infrastructure cost. End-hosts need a wireless access card, with a cost comparable to that of an Ethernet card, and the whole network is interconnected with the use of *Access Points* (APs), the cost of which is

also relatively low and comparable to that of wire-based equipment. However, there is no cost for cables and their installation.

- Flexibility of placement for mobile and portable devices since these do not need to be tethered with a cable.

These advantages have led 802.11 to be the most prevalent networking technique in domestic but also in commercial environments [3]. Furthermore, recent developments, such as the surge in sales of smartphones [4], the penetration of the market by portable Wi-Fi gateways that connect to the Internet via 3rd and 4th generation cellular networks [5] and the offloading of services to local networks by mobile operators [6], pave the way to an even more widespread use of 802.11 networks.

Unfortunately, the allocated bands for Wi-Fi are rather limited, so neighboring networks often operate in overlapping frequencies, which raises the problem of interference. To mitigate interference, 802.11 uses *Carrier Sense Multiple Access with Collision Avoidance* (CSMA/CA) [7], the goal of which is to allow only one host to transmit at a time. Nonetheless, two hosts could potentially transmit concurrently provided that a sender's signal at its intended receiver is sufficiently stronger than any interference and noise. Doing so can result in a significant increase to aggregate throughput—in the best case, a doubling, as two transmissions can proceed concurrently, each at full speed, rather than serially over double the time.

Newer physical layer methods such as *beamforming* and *nulling* take advantage of the multiple antennas on Wi-Fi hosts to increase the strength of the signal of interest or decrease the strength of the interference at a receiver. The research community viewed these techniques with great interest as means to enable proximate hosts to transmit concurrently without

interfering with each other at their intended receivers, and thus increase aggregate throughput. Indeed, prior work has shown experimental results suggestive of significant throughput improvements achieved through beamforming and nulling. In this thesis, we demonstrate that these early results insufficiently account for the real-world behavior of Wi-Fi systems—*e.g.*, by considering narrower frequency bands than those employed by Wi-Fi systems—and as a result, significantly overpredict the throughput improvements achievable through straightforward application of nulling. In Chapter 4, we experimentally demonstrate that nulling often *underperforms* CSMA/CA, identify the physical-layer phenomena unaccounted for by the community to date that are responsible, and propose techniques that address these phenomena, and thus allow nulling to achieve its hoped-for throughput improvement.

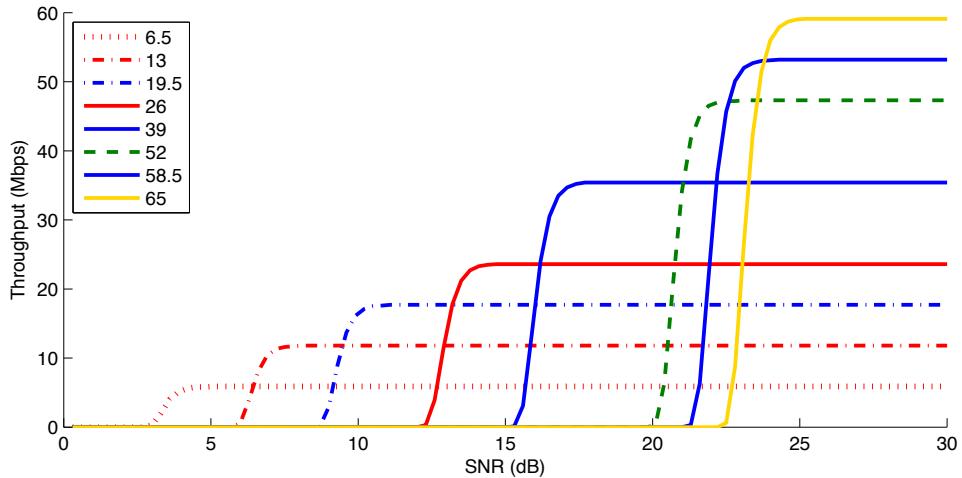
More generally as we will see throughout this thesis, profitably applying interference nulling requires taking into consideration the practical aspects of a system’s design and implementation, since hardware might provide only limited functionality or its inherent limitations can cause nulling to not realize its full potential.

## 1.1 Background

Wireless communications are based on the transmission of electromagnetic waves. A sender codes the information it needs to transmit by modulating electromagnetic waves which propagate through free space or by reflections on objects in the environment and reach a receiver. The receiver extracts the transmitted information from the signal’s amplitude, frequency, and phase.

There are several obstacles to this process. An electromagnetic wave faces attenuation of its amplitude as it travels through space. Hardware and the environment add noise to the signal, distorting it. Furthermore, a signal reaches its destination through several different paths because of reflections in the environment, a phenomenon called *multipath propagation* [10]. These multiple versions of the same signal differ from one another but are received additively at the destination. Another level of complexity arises because all transmissions share the same medium, i.e., free space. One way to mitigate this problem is by assigning different frequency bands to different users. Such a solution does not truly solve the problem though, because repeated division of the available bandwidth results in narrow channels that cannot support high bitrates. It also makes channel assignment a complicated process.

A receiver's ability to accurately decode a sender's transmission is dependent upon the strength of the received signal but also on the noise level, which is measured by the *Signal-to-Noise Ratio* (SNR). Another important factor is interference, which, when combined with the signal strength and the noise level, produces the *Signal-to-Interference-plus-Noise ratio* (SINR). The lower the SINR, the more difficult it is for a receiver to decode a signal, leading to a higher *bit-error rate* (BER). Depending on the parameters of the coding scheme used, such as redundancy, and the modulation, there is an SINR threshold under which the transmission is not decodable. In the case of 802.11, there are several coding and modulation schemes available. Each requires different SINR to achieve the same BER. Because of framing, there is an SINR threshold above which a scheme can achieve an acceptable packet delivery rate, and thus useful throughput. Figure 1.1 shows the coding-modulation schemes used in 802.11n and the corresponding



**Figure 1.1:** Theoretical UDP throughput when sending 1500 byte packets for the bitrates used in 802.11n. Higher bitrates require higher SINR in order to achieve non-zero throughput. Lines with the same color use the same modulation; line style denotes coding rate.

throughputs they achieve under different SNRs. As we can see, schemes that provide higher bitrates also require higher SNR in order to achieve non-zero throughput.

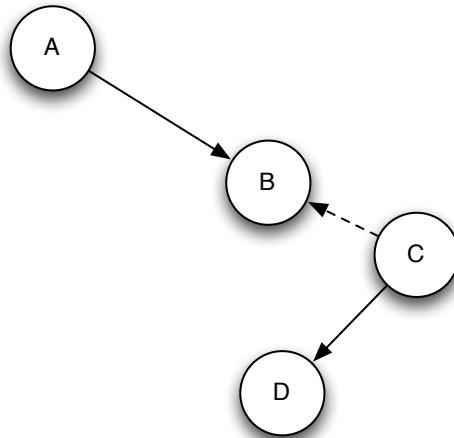
All versions of 802.11, with the exception of 802.11b [2], use a scheme called *Orthogonal Frequency Division Multiplexing* (OFDM) whereby hosts subdivide their channels into smaller parallel ones called subcarriers and transmit information on each of them separately. This scheme is well suited for systems where multipath propagation is strong, as is the case in indoor environments, and allows better use of the available spectrum in the case of wideband channels, where different frequencies may have different propagation characteristics. This is crucial, especially for 802.11n/ac [2], where very wide channels are used.

The standard way of sharing the medium in 802.11 networks is based on CSMA. When a host wants to transmit a packet, it first senses the medium and tries to decide whether it is in use. If not, it proceeds with its trans-

mission. This strategy, though, has its drawbacks: it might be the case that another host is transmitting even if a host cannot detect that transmission, resulting in two concurrent packets at the destination. Figure 1.2 shows a topology in which hosts A and C are too widely spaced for either to detect one another's carrier. Host A will try to transmit a packet to host B even if host C is transmitting to host D. If the SINR of host A's transmission at B is not high enough, the packet will not be decodable, resulting in a retransmission. This situation is known as the *hidden terminal problem* [11, 12]. If a collision occurs, the transmitting hosts will not receive acknowledgements for their frames and will attempt to retransmit. To avoid synchronization of the retransmissions, and thus a new collision, each host chooses a random backoff time within a contention window. If a collision happens nevertheless, the hosts repeat the process but they increase the width of the contention window. This process, called *binary exponential backoff* [2], incurs a significant overhead, though, since hosts waste time idling during the backoff period, resulting in even greater cost for collisions.

Since the adoption of 802.11n [2], commodity hardware employs *Multiple Input Multiple Output* (MIMO) [10], a method that increases the number of concurrent transmissions by using multiple transmit and receive antennas. MIMO hosts send and receive multiple frames concurrently, each forming a different stream of information. Each transmit antenna sends a linear combination of all streams that is disentangled at the receiver provided that the receiver has at least as many antennas as streams transmitted.

Applying the same technique at either the receiver or the transmitter can minimize the strength of interference from unwanted transmissions but such approaches again require sufficiently many available antennas. This



**Figure 1.2:** If host A cannot detect host C's carrier, it might try to send a packet to host B while host C is transmitting to host D. B will be unable to decode A's transmission

interference mitigating mode of use is not provisioned in the 802.11 standard where instead, transmitters take turns using CSMA.

## 1.2 Problem Statement

The aforementioned popularity of wireless networks has led to an increase in the density of their deployment. Even as early as 2005, Akella *et al.* showed several cases of more than four interfering access points in major US cities [13]. This is not surprising, if we take into account that most access points have an advertised transmission radius of 100 meters. Although there are eleven channels in the 2.4 GHz band, they occupy only 60 MHz allowing for just three concurrent non-overlapping users. The adoption of the 5 GHz band somewhat alleviated this bandwidth scarcity by adding one band of eight and one of eleven non-overlapping 20 MHz channels (figures are for the European Union according to [14]). However, the need for increased throughput has led to the provision for concurrent use of four

or eight of those channels by a single AP in 802.11ac [2], thus offsetting the increase in bandwidth from the point of view of spectrum availability for hosts within different proximal networks.

As a result, multiple proximate transmitters often have to use the same part of the spectrum but doing so concurrently causes interference, and as a result leads to undecodable frames and throughput degradation. As we will see throughout this thesis, although there are several methods to mitigate interference, both on the medium access and the physical layer, these often are either too conservative, leaving resources underused, or do not achieve the expected throughput improvement when they are applied in way that does not take into consideration the specific characteristics of the system they are applied to. Additionally, some techniques either require much more complex, and hence expensive hardware than that currently in use, or can only be applied in systems that have special features.

For instance, many dense deployments are not centrally managed. Although neighboring networks may have an incentive for some level of co-operation, the opportunity to cooperate is limited since different entities administer them and it is unrealistic to assume that such networks are connected by a wired backplane through which their operators are willing to exchange information. The use of CSMA/CA mitigates interference but at great cost, since it implicitly assumes that *any* interference will be *completely* destructive, rendering all concurrent transmissions undecodable. As mentioned in the previous section, though, the decodability of a signal depends on the relative strength of the signal of interest in relation to those of interference and noise, which means that if the SINRs of transmitted streams are sufficiently high, some level of concurrency is possible.

On the other hand, large commercial deployments like the ones used for workplaces or large public venues face similar difficulties. In those scenarios, an organization employs several APs in close proximity to one another and reduces interference using a combination of low transmission power, “tilled” channel usage—so that neighboring APs do not use the same frequencies—and careful planning during installation. Nonetheless, using CSMA/CA will still lead to a reduction in throughput since even low amounts of received power will make APs defer [15], while reducing their transmit power too much will cause the emergence of *black spots* [16], *i.e.*, areas without connectivity. Furthermore, such strategies usually come with great financial cost since they require a lot of equipment, experienced installation staff, lengthy measurements, and may be cumbersome to maintain.

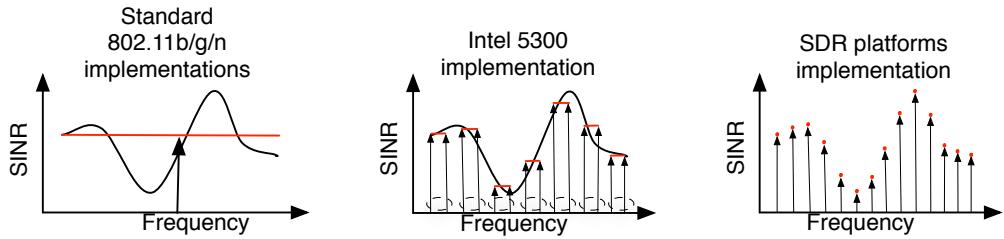
Beyond the medium access layer interference mitigation techniques, there are physical-layer methods that focus on increasing SINR, either on the transmitter or the receiver side and have been the focus of much recent research because they allow some level of concurrency, and thus higher aggregate throughput. Interference nulling, for instance, enables a transmitter with multiple antennas to align its transmissions from each transmit antenna so that they cancel each other out at the antennas of unintended receivers. In the case of large deployments, *Distributed* or *Cooperative MIMO* pools several access points which are interconnected by a fast wired network and are tightly coordinated to form a large virtual AP that acts a single transmitter. Unfortunately, phenomena like channel noise, hardware imperfections, and inaccurate channel estimates make these methods less than perfect, leading to residual interference, which can be detrimental to throughput if it is not accounted for.

This residual interference exacerbates further the natural variability of SINR of the wireless channel over its bandwidth. Although the use of OFDM allows each part of the used spectrum to be treated as an independent channel, the reality is that the fate of the transmitted information is correlated across those channels, since they all use the same bitrate in 802.11-compliant networks.<sup>1</sup> The result is that subcarriers that face adverse conditions negatively affect overall throughput, and since the latest trend is to use wider parts of the spectrum concurrently, the expected variability of SINR of the worst versus the best subcarriers will increase. Furthermore, as we will see in more detail in Chapters 4 and 5, using physical-layer techniques like beamforming or nulling worsen the problem of SINR variability across subcarriers even further.

One way that has long been used to maximize capacity under varying SINRs is subcarrier power allocation and more specifically waterfilling [7]. Nonetheless, users care about throughput in systems that use a single, practical modulation scheme and one coding rate across all their subcarriers. Waterfilling, by focusing on capacity maximization without taking these aspects into account, makes the problem of SINR variability worse. This is because it increases the spread of SINRs by allocating more power to better subcarriers. At the same time, it is not well suited for the typical SINR ranges of 802.11 which are relatively high (typically between 5 and 30 dB). Furthermore, changes in transmission on one stream, such as power redistribution across subcarriers, manifest as changes in interference at others. Such changes need to be accounted for, bringing up again the need for some level of cooperation but also for algorithms that converge to stable solutions.

---

<sup>1</sup>This is not a fundamental limitation but it keeps the cost of equipment low.



**Figure 1.3:** Standard 802.11 chipsets do not give access to per subcarrier channel state information. The Intel 5300 provides this information and allows setting phase and amplitude for pairs of subcarriers while SDR—and potentially future chipsets—provide data for each subcarrier and allow full manipulation of the per-subcarrier phase and magnitude of the transmitted signal.

Another way that platform capabilities affect the solution space of interference mitigation is shown in Figure 1.3. 802.11 chipsets that do not support MIMO only provide a single reading of the total power of the received signal and do not allow the alteration of the phase and amplitude of the transmitted signal. Both of these are crucial for methods such as beamforming that leverage multiple antennas to increase the strength of the signal of interest, or other ones like nulling that reduce the strength of an interfering signal to unintended receivers. Some of the more recent chipsets, like the Intel 5300 [17], give amplitude and phase readings for pairs of OFDM subcarriers and allow some crude control of the transmitted signal for each transmitted stream. There are also *Software Defined Radio* (SDR) devices that, among other things, give readings for each subcarrier and allow the full manipulation of amplitude and phase of each subcarrier of the transmitted signal but also of the coding, framing, and *Medium Access Control* (MAC) layer. Although SDR is only used for research and development, it allows the exploration of potential designs for future commercial 802.11 chipsets and how best to use features like per-subcarrier settings for phase

and amplitude and multiple coders/decoders that are not that far off from today’s designs.

The high-level goal of this work is the improvement of throughput for an end-user of *typical* 802.11 networks under the presence of interference—where “typical” is used to denote an infrastructure network with either one AP serving several clients (as is the case in domestic deployments) or several APs serving several clients (as is the case in commercial settings). At the same time, each system needs to have a set of desirable properties that depend on the use case, and the functionality and characteristics of the hardware used. In later chapters, we will examine three such scenarios. In Scenario A, examined in Chapter 3, APs with a single Tx/Rx chain connected to a phased array antenna try to decode a wanted signal while there are other unwanted, concurrent transmissions. In Chapter 4, we will examine Scenario B, where two neighboring APs that belong to different networks that do not share a common infrastructure, each try to transmit to their own host. Finally, in Chapter 5, we will investigate Scenario C, in which several APs, all belonging to the same organization, and connected with a fast, wired backplane try to serve multiple hosts. The systems for use in each of these scenarios must also achieve a set of further lower-level goals, such as:

- No need for centralized controller (Scenarios A, B).
- No need for wired backplane (Scenarios A, B).
- Efficient distributed coordination of concurrent transmissions by APs (Scenario B).
- Suitability for use with single modulation/coding rate on all OFDM subcarriers (Scenarios B, C).

	Centralization	Hardware capability
Scenario A	Low	Low
Scenario B	Low	Medium
Scenario C	High	High

**Table 1.1:** Comparison scenarios in terms of degree of centralization and hardware capability.

- Computational tractability within coherence time at acceptable hardware cost (Scenarios A, B, C).
- Suitability for use with single Tx/Rx chain and phased array (Scenario A).
- Use of the same subcarrier by multiple APs (Scenarios A, B, C).
- Technique suitable for application at receiver (Scenario A).
- Technique suitable for application at transmitter (Scenarios B, C).

The three aforementioned scenarios allow us to explore a wide range of use cases in two main dimensions, as we can see in Table 1.1. The first dimension is the degree of centralization of a system. Scenarios A and B examine the decentralized case, where transmitters may use a MAC-layer protocol to loosely coordinate, but they are not tightly synchronized. On the other hand, in Scenario C all transmitting hosts are managed by the same authority, they are interconnected with a wired backplane, and they are tightly synchronized.

The second dimension is hardware capability. In Scenario A we use hosts with rather limited capabilities. Our APs have a single receive/transmit chain which provides a single, crude metric of channel quality. Moreover, hosts can apply a single setting to attenuation and phase shift over the whole spectrum of the used channel for each element of their phased array antenna. On the other hand, hosts in Scenario B have a separate transceiver for each of their antennas. Furthermore, they provide channel state in-

formation, and can apply attenuation and phase shift separately to each subcarrier. This additional functionality allows us to perform beamforming and nulling with much greater accuracy. Finally, Scenario C uses APs with similar receive/transmit chains as the ones in used Scenario B, but now these APs are also connected with a fast wired backplane, and can be calibrated and synchronized very tightly with one another, enabling us to combine all APs' antennas into a single “virtual AP” with many antennas, which in turn allows us to achieve even greater throughput improvement.

### **1.3 Contribution**

Packet delivery rate, and consequently throughput, in 802.11 networks depend on the SNR of the received signal. A high SNR is generally desirable because it results in successful decoding of transmissions at higher bitrates. Moreover, modern 802.11 device drivers implement automatic bitrate adaptation, increasing the physical-layer transmission rate whenever they can maintain a low packet loss rate. Hence an increased SNR may lead to an increase in throughput. SNR, though, does not tell the whole story. As we describe in Section 1.2, host contention for the same channel leads to interference, which can negatively affect the performance of wireless networks. As a result, in order to maximize the throughput on an 802.11 link, it is necessary to increase SINR over the decoding threshold for high bitrates. We can achieve this goal in two ways: either by increasing the power of the signal of interest or by reducing the interference. At the same time, there is one more interesting dimension in the problem: that of frequency. Especially in the case of the indoor environment, we cannot assume that all parts of the spectrum experience the same conditions,

and bad subcarriers might have a disproportionately detrimental effect on overall throughput.

As mentioned earlier, CSMA/CA is too crude a tool to deal with contention in today’s dense urban deployments. It is our hypothesis that disseminating information about interference between networks and allowing for some level of concurrency can lead to better spectral utilization and increased overall throughput. On the other hand, assuming a very high level of cooperation between home networks is unrealistic. Domestic users may loosely cooperate—as they do in CSMA/CA—but lack the infrastructure, *e.g.* very fast, direct wired backplanes, and the incentive, because of security concerns, for instance, to share the content of transmitted packets with one another. Of course, in the case of several APs all managed by the same authority and interconnected with a wired network, the potential solution space is richer. Nonetheless, such deployments have limitations as well. Chapter 2 presents the current literature and state of the art, where we examine several methods that aim to mitigate interference under varying conditions, but also discuss their limitations when applied on practical systems.

Many wireless hosts have a single 802.11 transceiver that gives a single channel quality metric for the incoming signal, usually in the form of a *Received Signal Strength Indicator* (RSSI). Although, as we will see in the next chapter, multiple antennas can help in the mitigation of interference, using them effectively requires knowing the relative strengths and phases of the signals arriving on each receive antenna. In Chapter 3, we present *SamplePhase*, a method that allows single-transceiver hosts connected to a phased array antenna to robustly take all necessary measurements for the characterization of the channel between a transmitter and each constituent

antenna of our array on the receiver. Based on that, we show a receiver architecture that improves throughput of a single 802.11g transmission up to  $2\times$  compared to a traditional omni-directional antenna. We then extend our system to deal with concurrent interfering transmissions with *Silencer*, which improves throughput under interference, whether that interference is on the same channel as our receiver or even on a partly overlapping one. Our end-to-end throughput measurements show an improvement between 1.6 times and 17.5 times, depending on the topology. We also show results where Silencer nulls two concurrent interferers and improves throughput up to 3.1 times.

Of course having hosts with multiple transceivers, each of which can measure and manipulate each subcarrier independently increases the design space significantly. Transmitters with such capabilities can align the transmissions from each of their antennas in such a way that they cancel each other out at the antennas of unintended receivers. If transmitters in neighboring networks apply this technique, they can increase their aggregate throughput since, if each nulls its transmission toward unintended receivers, they can all transmit concurrently. Unfortunately, as we will see in Chapter 4, channel noise, hardware imperfections, and inaccurate channel measurements can significantly hinder the performance of such an approach. In fact, our measurements show that for the majority of topologies in an indoor office environment, simply transmitting concurrently while applying nulling actually decreases throughput compared to using CSMA. The reason for this is the presence of residual interference and the variability it causes to SINRs across subcarriers at receivers. This variability is further exacerbated by beamforming.

Subcarrier power allocation can reduce this variability, provided that it is well suited to the modulations used in 802.11 systems. Subcarrier selection can also help significantly. To those ends, we present a MAC layer protocol that allows transmitters to identify opportunities for cooperation and choose the optimal strategy for transmission, simultaneous or consecutive. Of course, changing the power distribution between subcarriers at one host manifests as a change in interference at neighboring ones. We have developed an iterative algorithm that allows transmitters to select the subcarriers they want to use and allocate their transmit power across them, such that the transmitters converge to a final allocation. We implemented our system, *Cooperative Power Allocation* (COPA), on a software radio platform and tested it in a variety of topologies and host configurations. Our measurements show that COPA performs well across all the different scenarios tested, yielding 10-20% throughput improvement on average compared to the best of nulling and CSMA. Furthermore, we also offer an incentive-compatible variant of COPA, in which participating hosts do not transmit concurrently unless they individually benefit in throughput.

Finally, in the case of several APs all managed by the same authority and interconnected with a fast, wired backplane, Distributed MIMO [18] is an exciting new development that can significantly increase aggregate throughput. Nonetheless, as we will see in Chapter 5, channel noise and hardware inaccuracies coupled with relatively correlated channels between transmitting and receiving antenna pairs can have a detrimental effect not only on the overall SINR of a stream, but also on inter-subcarrier SINR variability. Using the principles of subcarrier selection and power allocation, we present *Power Allocation for Distributed MIMO* (PADM), a system that can alleviate the detrimental impact on throughput of subcarriers with partic-

ularly low SINR, while potentially freeing up resources that can be reused by other concurrent transmissions. As we will see, PADM can increase throughput by about 16% for a 20-antenna Distributed MIMO system, while reducing throughput variability between concurrently transmitted streams. Furthermore, we will see that careful selection of the clients transmitted to concurrently can yield a further throughput increase of about 20% for the case of a 10-antenna virtual AP.

Overall, this thesis contributes a systematic exploration of practical effects that limit the efficacy of multi-antenna interference mitigation techniques in Wi-Fi systems, and proposes and evaluates mechanisms that address these effects, and thus improve throughput, allowing these techniques better to realize their potential.

## Chapter 2

# Background and Related Work

Given the widespread adoption of wireless networking and the central role of interference in limiting capacity, the research community has devoted considerable effort to interference mitigation techniques. Having multiple hosts transmit concurrently on the same medium works poorly since receivers typically cannot decode a signal correctly when it overlaps with another one. This problem was identified at the very early stages of building wireless systems and early designs resorted to solutions where a central controller exclusively grants a resource to a single user at a time, whether this resource is time, frequency, or space. As we will see in Section 2.1, exclusive allocation methods are effective but often wasteful, since, users are pre-assigned a resource that they might not fully use when the offered load is light. They also tend to rely on centralized control for authoritative resource allocation.

To overcome the rigidity of this approach, methods like CSMA/CA, which we examine in Section 2.2, and OFDMA [10], which we examine in Section 2.9, aim for a similar result but with a more de-centralized approach that makes them more suitable for unplanned network deployments. Al-

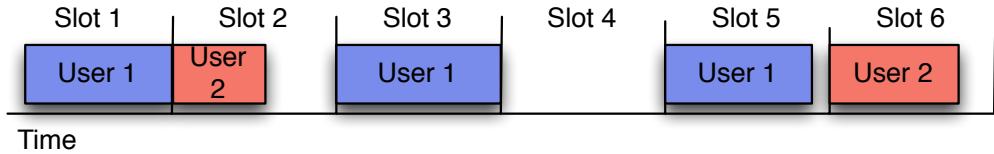
though the distributed approach of these exclusive allocation methods incurs an overhead, they allocate resources more adaptively under changing conditions, and so allow more flexible network deployments. For instance, 802.11 benefits from CSMA/CA's flexibility.

Besides Layer 2 (or MAC layer) approaches, lately there has been active interest in systems that mitigate interference at the physical layer. In this case, several users are allowed to transmit concurrently on the same part of the spectrum. Some methods, like interference cancellation [10] which we examine in Section 2.3, try to extract multiple transmissions from the same received signal, while others, as we will see later in Sections 2.5 through 2.8, take advantage of multiple antennas at the transmitters or receivers and combine several signals to transmit or recover information. For instance, interference nulling, a method that the systems described in later chapters use extensively, allows hosts to minimize the power of unwanted transmissions at unintended receivers thus enabling concurrent transmissions by neighboring hosts.

Finally, as we will see in Section 2.10, transmitters use power allocation to decide how to allocate their limited power budget over multiple available channels, as is the case with subcarriers in OFDM systems. Although not all systems employ power allocation as we will see in later chapters, a suitably interference-aware form of power allocation proves useful for mitigating interference.

## 2.1 Exclusive Resource Allocation

Cellular networks have traditionally applied multiple interference mitigation techniques. Although mobile telephony operates in a predefined sec-



**Figure 2.1:** In TDMA, a centralized controller splits time into slots and assigns them to users. If a slot goes unused by its user, no other user can use it.

tion of the spectrum, mobile carriers are assigned unique subsections of that spectrum by governments, so that multiple networks can be active concurrently in the same locations. Carriers subsequently divide their available spectrum even further, so that several hosts can be served at the same time [7, 10]. Mobile phone base stations split the geographic area around them into *cells* and service each with a different channel frequency from their assigned spectrum by using *sectored antennas* [7, 10]. Sectored antennas steer all radiated power in a single direction using reflectors. Splitting the available bandwidth into narrower bands is known as *FDMA* and also finds use in 802.11 networks [19]. Since different cells operate in different parts of the spectrum, they do not interfere with one another [7, 10].

Besides the use of FDMA in cellular networks at the cell level, mobile hosts within the same cell access the medium using *TDMA*, illustrated in Figure 2.1. Under TDMA, at any given moment only one host transmits, and thus hosts take turns using the channel. Cellular telephony standards specify TDMA sharing orchestrated by the base station [7, 10]. Specifically, the base station divides time into time slots and assigns them to its clients. Every client is allowed to send a frame only during the time slot it has been assigned and the whole system stays synchronized using beacon frames transmitted by the base station. The *Point Coordination Function* is a TDMA

MAC defined in the 802.11 standard [19] but is not widely implemented or used because of its poor robustness against hidden terminals [20].

There are key differences between cellular and 802.11 network deployments. Firstly, there is a limited and government-controlled number of cellular networks and the assignment of spectrum to them is also handled by a central authority. In contrast, anyone can deploy an 802.11 network, even though the available spectrum is limited to only three non-overlapping channels in the 2.4 GHz band and 19 in the 5 GHz band [14]. Secondly, today's 802.11 access points use omni-directional antennas and do not sectorize space around them as cellular base stations do.

## 2.2 Carrier Sense and Collision Avoidance

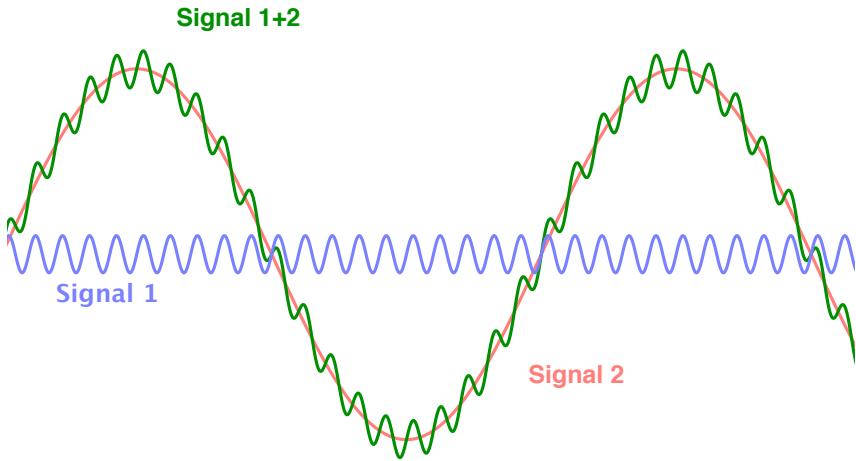
The standard mechanism for interference mitigation in 802.11 is CSMA/CA [2], where before a host transmits, it senses the medium for ongoing activity and only proceeds with its own transmission if the medium is free. Otherwise, it waits for the ongoing transmission to finish, then waits for an additional, random amount of time bounded within a window defined by the specification. If, nonetheless, a collision occurs, hosts increase the size of that window using an exponential backoff scheme until they are successful with their transmission.

Karn [11] and Bharghavan *et al.* [12] noted in the early 90's that "*the relevant contention for the medium does not take place at the sender but at the receiver*". In the case of a wired network, the sender can have almost perfect knowledge of the channel by sensing the medium. This is because the cable that connects hosts is designed to propagate electromagnetic waves well between any two participating stations in a LAN. On the other hand,

free space transmissions propagate towards both participating hosts in a WLAN but also towards hosts that are members of different WLANs. Additionally, electromagnetic propagation in free space attenuates far more severely than over a cable. The result is that a transmitter might not be able to tell that its transmission might interfere with other ongoing ones and proceed, leading to a collision. What is truly important is whether the channel is free or occupied *at the receiver*, since decoding is adversely affected by interference.

Karn's proposal for resolving contention in wireless networks is *collision avoidance*. This scheme implements a *virtual carrier sensing* mechanism by allowing the sender to be informed of the state of the channel at the receiver. Before a host sends a packet, it transmits a very short control frame called a *Request-to-send* (RTS). If that frame is delivered intact to the receiver and a possibly interfering transmission is not detected, the receiver sends back a control frame called a *Clear-to-send* (CTS). If that exchange of control frames concludes successfully, the sender is allowed to transmit the intended data frame. Furthermore, other hosts that overhear the CTS frame are not allowed to send packets for an amount of time specified within the CTS frame.

Although collision avoidance without the use of the RTS/CTS has been shown to be robust [21], it allows only one transmitter to be active at any given moment. This of course can be a very limiting factor especially in the case of dense deployments [13] where several hosts are within range of one another since these hosts would have to time-share the medium, and so achieve only a fraction of the throughput they would if they were the sole users of the medium. Nonetheless, as we will see in Chapter 4, CSMA/CA may in some settings outperform physical-layer techniques intended to in-



**Figure 2.2:** Signal 1+2 is the sum of Signal 1 and Signal 2. A host can decode Signal 2, subtract it from Signal 1+2 and retrieve Signal 1.

crease concurrency, such as those we discuss in Section 2.7. It thus remains an important weapon in the Wi-Fi resource allocation arsenal, and one we employ in the systems we propose in this thesis.

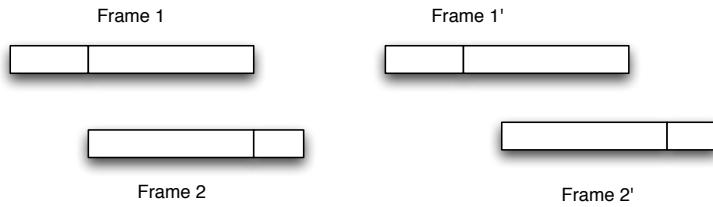
## 2.3 Interference Cancellation

A more recent development in dealing with interference has been *interference cancellation* [22, 23, 24]. The theoretical basis of this technique is that interference, unlike noise, is not a stochastic process but a deterministic one. A consequence of this is the possibility of decoding an interfering signal. As we can see in Figure 2.2, a useful transmission could be heavily affected by an interfering one. The signal at the receiver will be the sum of these two transmissions. If the interfering signal is decodable, it can be subtracted from the sum and a decodable reconstruction of the useful transmission will be available.

It is clear from the above definition that interference cancellation is a multi-step process. Whenever a collision occurs, the receiver first retrieves the easiest to decode piece of information. It later subtracts a reconstruction of that piece, hence *canceling* its contribution, from the original signal and then proceeds with the decoding of another concurrent transmission. The first practical implementation of a successive interference cancellation scheme was demonstrated by Viterbi in 1990 [22] and since then many similar techniques have demonstrated good results [23, 25, 9, 26]. There is also *parallel* interference cancellation, which tries to isolate several signals in parallel for reduced latency and better performance under situations where several signals of comparable power are received [23].

Most prior work in interference cancellation comes from the field of mobile communications. Prior work in interference cancellation in WLANs is based on software-defined radios. In one of the first such systems, Halperin *et al.* use the ZigBee physical layer (IEEE 802.15.4) and software radios to build a system with standard successive interference cancellation, where the strongest decodable signal is decoded first and later subtracted from the actual received signal to make possible the decoding of other transmissions [24]. Their evaluation uses the BPSK modulation scheme, in which it is fairly easy to identify two concurrently transmitted symbols. As the authors point out, their system would face significant challenges under different modulation schemes, where the required difference in received power would need to be even greater than the 4 dB that they assume in their evaluation.

Gollakota *et al.*'s ZigZag implements a sub-frame version of interference cancellation [25]. The system exploits that two colliding packets are likely to collide again—since senders will try to retransmit—but they will collide



**Figure 2.3:** The time offset between frames in successive collisions is usually different because of the random exponential backoff mechanism.

with a different time offset, because of the randomization of the back-off mechanism of 802.11. The receiver can use these different versions of the overlapping packets to decode both of them, block-by-block. As we can see in Figure 2.3, the areas that are affected by the collisions are different. By subtracting the collision-free part of Frame 1' from the sum of 1 and 2, the interference to the first section of Frame 2 is cancelled. This interference-free section can be used to cancel the interference in a section of Frame 1'. By iterating this process, both packets can be decoded. A drawback of the system is that it is meant to work for a scenario where two hosts are transmitting frames to the same AP. When they do that for the first time both packets collide and the hosts repeat their transmissions. When each of the two colliding packets has a different destination, though, it is likely that one of the two transmissions will succeed and will not be repeated, and thus the source host will proceed with the transmission of the next packet. Since the two collisions will include different packets, these will not be decodable by ZigZag.

Interference cancellation is a component in the design of IAC (Interference Alignment and Cancellation) [9]. The difference in this case, though, is that several receiving APs are used at the same time. If a receiver manages to decode a packet, it uses a wired LAN to send the packet to the rest of the receivers. These in turn can use a reconstructed version of the now-

known packet to cancel the interference it causes to other transmissions. This scheme of course requires that all the hosts involved cooperate and exchange network packets through a wired LAN, which is not typical in urban home deployments.

SAM also uses interference cancellation as a component of a more complex scheme focused on the uplink [26]. Here, the whole process is bootstrapped with what the authors term *interference nullifying*, essentially a *zero-forcing* receive filter. After that step, they apply successive interference cancellation, subtracting the frame recovered with interference nullifying from the total received signal.

**Discussion:** Interference cancellation is a useful method for increasing the capacity of a wireless system since it allows a wireless host to receive two concurrent transmissions. Unfortunately, interference cancellation requires separation in the power levels of the concurrent transmissions, preventing its use in cases where signals are received with comparable strength. Also, receivers have to tune their amplifiers to a level suitable for the incoming signal. If the second (in time) concurrent transmission is of higher power than the first, it will saturate the receive amplifier, rendering both signals undecodable.

Even when that is not the case, commodity wireless hosts distort received signals because of noise in both their transmit and receive chains, described by the *Error Vector Magnitude* (EVM) of their components. When a receiver tunes its amplifiers to a level suitable for the incoming signal, the noise level experienced depends on those levels—it is typically about 30 dB below the peak power level of the signal [27]. This means that a second signal, much weaker than the stronger one, would experience a disproportionately higher noise level than the one it would experience if it was the only incom-

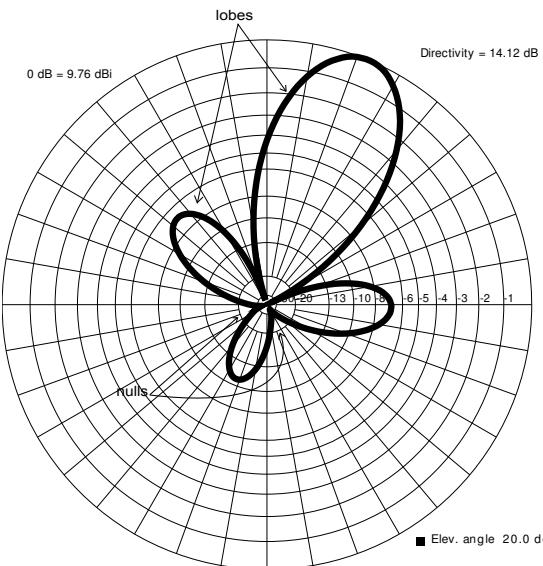
ing transmission. These conflicting phenomena create a “Goldilocks” zone for interference cancellation to be possible, where the difference in power between the different transmissions has to be within a specific range that gets tighter as we increase the complexity of the modulation used.

Moreover, whenever interference cancellation fails, it is always the weakest signal that is not decodable. That might not be problematic when both transmitting hosts are trying to communicate with the same receiver but in cases where the interfering transmission is stronger than the useful one, failure of interference cancellation leads to a waste of considerable computational power without any gain. Finally, all the systems we examined here have achieved results with only two concurrent transmissions and do not investigate the case of multiple coalescing packets or multi-stream transmissions. These weaknesses lead us to not use interference cancellation in any of the system designs we propose and evaluate in this thesis.

## 2.4 Antenna Directionality

The simplest kind of antenna is the dipole. Dipoles radiate a toroidal electromagnetic field that is omnidirectional on its azimuth. The way that an antenna emits electromagnetic radiation is equivalent to its capacity to receive it – which is called *gain*. This means that a dipole offers better reception on its azimuth plane and little reception immediately above or below it.

Other types of antennas offer different gain patterns heavily dependent on their geometry and operating frequency. Often, they show particularly high gain in one or more directions corresponding to their *lobes*. Figure 2.4 shows the radiation pattern of a highly directional antenna that emits



**Figure 2.4:** Radiation pattern of a directional antenna. The thick line shows the gain in dB compared to the maximum seen in the largest lobe (at  $67.5^\circ$ ) for all the angles on the azimuth plane.

most of its radiation in one direction and some of its radiation in others, while in some directions it does not emit any. Antennas with such characteristics are well suited to transmit and receive signals in those directions but not in others. In fact, they might be completely incapable of reception and transmission in the directions around them called *nulls*, where zero or nearly zero gain is applied.

**Discussion:** Directional antennas are used for point-to-point communication between fixed hosts because they have *fixed* characteristics. This makes them a poor fit for 802.11 networks—the focus of this thesis—where hosts cannot be assumed to be permanently fixed and in fact might even be mobile.

## 2.5 Switched Beam Antennas

Another class of antenna can alter its radiation pattern altered under programmatic control. One way to do this is simply by moving the antenna, for example, by rotating it. Another way is by modifying the characteristics of the antenna, either by changing the attributes of its power supply or its geometry. The facility for changing the radiation pattern can be very advantageous in cases where directionality is beneficial in several different directions.

A sub-category of switched beam antennas is that of *phased array antennas*. These are composed from several basic elements – typically dipoles – usually set up in a line or on a circle. Their radiation pattern is adjusted by selecting appropriate values for the electric current of each element – more specifically, amplitude and phase. Because the current can be altered rapidly, the radiation pattern of a phased array can be configured rapidly.

Most of the applications of directional antennas have been in the context of outdoor communications where the directionality of the antenna can be leveraged in order to achieve sufficiently high SNRs over long distances. In the case of *MobiSteer* [28], the authors devise a mechanism that allows wireless hosts in a car to select the best directional pattern from a pre-selected set in order to improve the SNR of wireless access with static omni-directional APs. Ramachandran *et al.* expand on this design and use a larger set of patterns that includes ones that radiate in two or three directions and combines directionality with diversity [29].

DIRC examines the use of directional antennas in the indoor environment [30]. Liu *et al.* use switched beam antennas and devise a MAC

mechanism to allow concurrent transmissions. Their system works in an enterprise WLAN context, where several APs maximize total throughput by transmitting concurrently. In order to do this they build a *conflict graph* of the existing APs. A central controller schedules frame transmissions in such a manner that concurrent transmissions do not interfere with each other.

Before DIRC, Choudhury *et al.* had also focused on a MAC protocol for networks of hosts with directional antennas [31]. Their system uses a reservation algorithm based on RTS/CTS and allows concurrent transmissions in the context of multi-hop wireless networks. This work focuses on the case of transmission of multiple frames along several links and how different links along the same path can be active at the same time.

Finally, Sen *et al.* use switched beam antennas to deliver multicast transmissions [32]. Their system is based on the idea that some of the hosts serviced by an AP might have worse channel conditions than others. In order to be able to deliver the same rate of information to all the clients, they first transmit a frame omnidirectionally and then retransmit directionally to hosts with poor wireless links to the AP. Using a directional pattern results in higher SNR and consequently lower BER, resulting in the same throughput as that achieved for hosts with good links when the AP sends omnidirectionally.

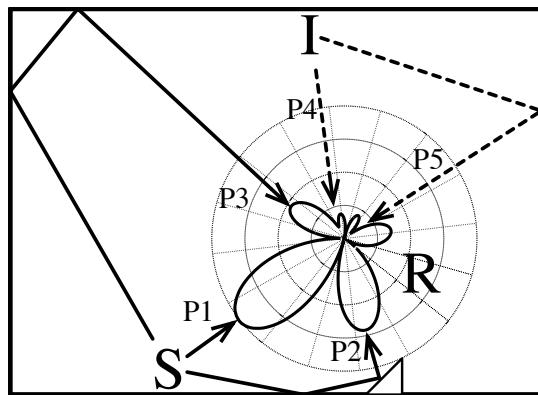
Halperin *et al.* [33] use a highly directional antenna called a horn antenna to create point-to-point links by rotating the antennas of the transmitter and the receiver. They use this scheme to transfer data between hosts in a datacenter so as to avoid hotspots. Their system uses a scheduler that arranges the transmissions between end hosts, but these transmissions have to be long enough to amortize the cost of rotating the antennas to establish

a link. Zhou *et al.* [34] extend this model and add reflectors to the walls and the ceiling of the datacenter to create more complex wireless topologies and avoid interference.

**Discussion:** Switched beam antennas can adapt to channel conditions more quickly than directional antennas with a fixed pattern. Being able to switch between patterns lets a host to change the direction in which it focuses its energy, allowing the best available pattern to be chosen depending on the host it tries to communicate with. This approach still does not utilize, though, the full potential of multiple antennas. Most of the work cited above uses phased array antennas, the radiation characteristics of which can be controlled far more precisely than explored in prior work. Pre-selecting a small set of radiation patterns and then switching between them or mechanically moving highly directional antennas might yield better results than omnidirectional transmission but these preconfigured fixed-beam patterns may differ significantly from the best possible beam shape that the hardware used can achieve.

## 2.6 Adaptive Beamforming

Static, preconfigured beam patterns may perform especially poorly in indoor environments where transmissions are affected by due to *multipath propagation*, as is the case in Figure 2.5. In this example, there is significant energy arriving at  $R$  from paths  $P_2$  and  $P_3$  besides the main path  $P_1$ . Vendor-supplied patterns typically focus all their energy in a single direction and thus do not gather potentially valuable power arriving from different directions along other paths. Furthermore, interference from host  $I$  arriving from paths  $P_4$  and  $P_5$  can be nulled if nulls are placed at the appropriate angles. It is statistically improbable for a phased array to have



**Figure 2.5:** In an indoor environment, a signal from host S can arrive at its destination R from multiple directions as above where the signal is received from paths P1, P2, and P3. R can maximize the strength of the received signal by choosing an appropriate pattern such as the one depicted. If the array uses a switched beam scheme, the depicted pattern will not be available and R might be forced to choose some other suboptimal one. Furthermore, interference from host I arrives from paths P4 and P5. Placing nulls in these directions would significantly reduce interference.

optimal patterns preconfigured for every possible scenario. Hence, if it uses a switched beam scheme, it will be forced to choose a suboptimal one.

Thompson *et al.* make the case for antennas that can dynamically adapt their radiation pattern for use in CDMA mobile networks [35]. They argue that phased arrays can significantly increase spatial reuse. Their system models the channel and later combines the signals received by each individual element with methods such as *Maximal Ratio Combining* (MRC) and *Selection Diversity* [10], which we shall see in more depth in Chapter 3.

Lakshmanan *et al.* propose a method for the calculation of optimal patterns that tries to maximize SNR – but does not take into account any interference – using simple power measurements [36]. Their scheme leverages power measurements taken by commodity hardware to estimate the

channel between a host and each one of the elements of a phased array. After modeling the channel, they proceed with MRC in order to maximize the power of the desired signal. Since BER, and consequently successful packet reception, depends on SINR, maximizing the signal of the intended signal alone might not necessarily lead to an increase in throughput if it is accompanied by an increase in the power of interfering signals.

Park *et al.* in [37] describe a system that leverages the multiple antennas of 802.11n hosts in order to transmit concurrently with legacy hosts. In their system, when an 802.11b/g link is active, an 802.11n sender beamforms, placing a null towards the legacy receiver while sending concurrently a frame of its own. In addition, while receiving, the 802.11n receiver places a null towards the legacy sender in order to cancel out its interfering transmission. Similarly, Mundarath *et al.* in [38] perform transmit and receive beamforming, but their system requires the separation of the available bandwidth into data and control channels. The control channel is used to exchange RTS/CTS control packets between the participating hosts as well as for calculating the attenuation and phase effects of the channel.

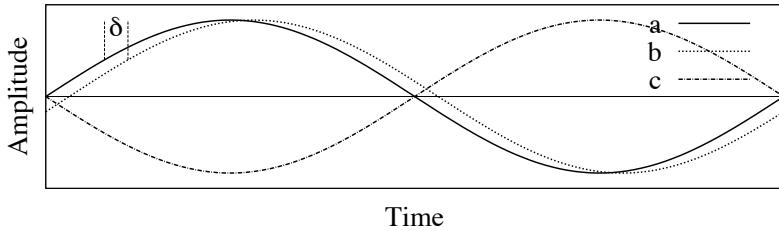
Aryafar *et al.*, on the other hand, leverage the power of *Software Defined Radio* (SDR) and build a system that does adaptive transmit beamforming [8]. Their system is designed to transmit frames to multiple hosts at the same time, with the use of *Zero-Forcing Beamforming*. This approach can also maximize SINR at a host when the sources of interference are known. This system only emphasizes physical layer reception and transmission of packets, without an evaluation under realistic usage scenarios. Parts of the signal processing are done in an offline manner and their system does not use 802.11.

Precisely adapting the radiation patterns of receivers and transmitters to time-varying channel conditions is also important for the methods we examine in the next section, and overall an important part of the designs we present in Chapters 3, 4 and 5.

## 2.7 Interference Alignment and Interference Nulling

Modern wireless systems use multiple antennas. 802.11n, for example, employs *Multiple Input – Multiple Output* (MIMO) principles to send several streams of information concurrently by leveraging multiple antennas on both the sender and the receiver. As we will see in more depth in Chapter 3, the same signal received by different antennas is subject to different attenuation and propagation length, which in turn leads to a different phase. If the different versions of the signal are *aligned* in a particular way, their combination can lead to an amplified or attenuated version of the original transmission. Figure 2.6 shows three versions of the same signal. If a host receives versions *a* and *b* the result will be a stronger signal that is easier to decode. If, on the other hand, the host receives versions *a* and *c*, their combination will have near-zero amplitude and the original transmission will be impossible to decode. Both cases are useful. Aligned signals boost SINR at the receiver while interfering signals whose multiple versions are out-of-phase will be *nulled*, yielding an interference-free frame.

The alignment properties for signals can be extended to multiple dimensions and multiple different signals as we shall see in next chapters. The multiple dimensions derive from the use of multiple antennas and can be used to align signals in such a way as to not interfere with each other. The



**Figure 2.6:** The first two versions of the signal will result in a combination that will be stronger while signals  $a$  and  $c$  would cancel each other out.  $\delta$  is the phase difference of the two signals, i.e., the time difference between them.

idea of structuring multiple transmissions in a way that they do not interfere with each other by aligning signals was first introduced by Maddah-Ali *et al.* [39, 40], while Jafar and Shamai use interference alignment to compute the number of degrees of freedom of the MIMO cross-interference channel from a theoretical perspective [41]. The same principle has been used at the receiver side before that, as part of the V-BLAST architecture [10] which is used for decoding multiple concurrent streams of information. More specifically, transmitters can align multiple versions of the same signal in such a way that their sum is zero at the antenna(s) of unintended receiver(s), a process called *interference nulling*. To do so they attenuate and calibrate the phase of the transmissions from each of their antennas, a process called *precoding*. Alternatively, in interference alignment, transmitting hosts can align several unwanted transmissions so that they appear as a single interfering transmission at an unintended receiver. If that unintended receiver has multiple receive antennas, it can recover a useful transmission by filtering out this compound interference.

A crucial aspect of MIMO techniques on the transmitter side is the number of available antennas known as the *Degrees of Freedom* (DoF) of the trans-

mitter. For each additional transmit antenna a host can do one of three things [10]:

- Send one additional stream of information to a receiver, provided that this receiver has at least as many antennas as the total number of streams it receives. The transmitter can send streams to multiple hosts at any given moment.
- Align or null its transmissions towards a receive antenna. Typically that antenna belongs to an unintended receiver and the transmitter would normally need to null towards all the antennas of that host to ensure that it doesn't interfere with any useful concurrent transmissions.
- Increase the power of existing streams at their destination due to increased transmit power as well as diversity gains because of the additional paths to the receiving antennas.

In IAC [9], Gollakota *et al.* show an implementation of interference alignment. In their system, several hosts send multiple transmissions at once. They achieve this by carefully choosing the phase of their transmissions so that each host receives a number of frames equal to the number it can decode while the rest will align in a non-interfering – but undecodable – way at their destination. A central administrator ensures that all hosts in the system are time-synchronized and that each packet will either be decodable at its destination, or if not, that it will be decodable at some other host and will be sent to its destination using a wired backplane. This is possible because the system works in the context of an enterprise environment where several APs cooperate, are centrally managed, and have wired connections to a common LAN.

Tan *et al.* [26] propose a different approach to interference alignment that they call *interference nullifying*. Unlike in IAC, where transmitters are responsible for the alignment of their signals, hosts in SAM align frames *at the destination*. They do so by taking advantage of each 802.11 frame's initial *known preamble*. The preamble can be used to estimate the phase difference between versions of the same signal received at different antennas as well as their relative strength, both of which enable nullifying. After retrieving one frame with this method, they proceed with the decoding of the rest by using successive interference cancellation. The difference between their scheme and interference alignment is that in Tan *et al.*'s case the signal alignment takes place at the receiving host during post-processing, while true interference alignment requires concurrently transmitting hosts to cooperate in order to align their transmissions. SAM and IAC both try to mitigate the problem of medium-sharing among hosts on the same network but do not address interfering hosts from other networks.

802.11n+ combines interference alignment and interference nulling [42]. Lin *et al.* take advantage of hosts with multiple receive and transmit antennas and build a decentralized system that can combine these two techniques to increase throughput. In their system, hosts are allowed to transmit even if there are ongoing transmissions, provided that they can properly align or null their signal so that they will not disturb any concurrent reception. Kumar *et al.* [43] on the other hand use a centralized approach and combine alignment and nulling in OpenRF, a system built with commodity hardware. In OpenRF, a centralized controller knows all the hosts that want to transmit in advance and orchestrates them, providing them with the exact precoding values necessary to avoid interference at receivers.

In a more recent development, Adib *et al.* in MoMIMO [44] show a system that can bring the benefits of interference alignment and nulling to single-antenna hosts by using controlled motion. The authors employ sliding antennas and a stochastic hill-climbing algorithm to achieve changes in the phase of transmitted signals. Sliding antennas require physically complex and awkward antenna motion assemblies. At the same time, modern hosts usually have more than one antenna.

**Discussion:** Most of the proposed systems such as IAC and OpenRF are designed for enterprise deployments, which use several APs interconnected with a high bandwidth wired backplane. Home and small business wireless networks generally lack such a wired LAN between APs. Finally, practical systems experience noise in the transmit and receive chains of the participating hosts that is additive to the noise of the wireless channel. This means that methods such as alignment and nulling are imperfect in practice. As we will see in Chapter 4, these imperfections can significantly degrade performance. However, interference nulling and alignment can increase significantly the throughput of wireless networks under interference, and are thus a major focus throughout this thesis.

## 2.8 Distributed MIMO

As we discussed in Section 2.7, having multiple antennas gives a great deal of flexibility as it allows hosts to send and receive multiple streams of information, mitigate interference, and make transmissions stronger at intended receivers. Practical hosts, though, have only a limited number of antennas due to physical limitations. If the antennas of a host are close to one another, their channels are correlated, limiting their usefulness for concurrent transmissions. This phenomenon is called *channel hardening* [45]

*Distributed MIMO* overcomes these antenna limitations by combining several different hosts to create one large *virtual* host. To achieve this, the different hosts must be very tightly coordinated in several respects, including the timing of their transmissions, their exact transmit frequencies and the relative phases between their transceivers – something that requires very precise calibration. Furthermore, all constituent hosts of the virtual AP must share transmitted/received data with one another, since this information is necessary to construct the transmitted signals or retrieve the received ones. This requirement means that hosts must be interconnected with a fast wired backplane in order to be able to exchange the large quantities of data required.

MegaMIMO [18] is one of the first practical implementations of downlink distributed MIMO in the context of 802.11 networks. In order to deal with the problem of phase synchronization, Rahul *et al.* elect one of the transmitting nodes as a leader. At the start of a transmission, the leader transmits a synchronization header which is used by other transmitters to calibrate their frequency, time, and phase offset. Since all transmitters are calibrated to the same reference, they can act a single virtual AP. To deal with intra-packet frequency offset, MegaMIMO keeps a rolling estimate of the offset and calibrates transmitters dynamically.

Balan *et al.* [46] implement a distributed MIMO system in which they also examine additional transmission techniques besides standard zero-forcing such as the use of *Tomlison-Harashima Precoding*, which provides a small capacity increase in high SNR regimes, or *Blind Interference Alignment*, a scheme that provides capacity gains by repeating transmitted symbols when there is no channel state information available at the transmitters, in a similar way to the Alamouti scheme [10].

Yang *et al.* in BigStation [47] examine the practical aspects of building distributed MIMO systems, such as high I/O load and processing power requirements. Their system achieves significant throughput gains with a relatively low mean processing delay, even though it uses commodity PCs and Ethernet switches (but software defined radio front-ends). To do so, they break down the necessary steps for transmitting and receiving and they pipeline and distribute the computation across multiple hosts. Although they achieve significant gains, the delay in their system is still prohibitive (mean value of 860  $\mu$ s) for a real 802.11 deployment, highlighting the practical challenges distributed MIMO deployments face.

GeoSphere is a system that can be used in the context of both MIMO and distributed MIMO systems to improve throughput at the receiver. Niki-topoulos *et al.* [48] use a computationally efficient implementation of a decoding method called a *sphere decoder* to deal with correlated channels and inter-stream interference. Their technique approaches the performance of the optimal receive architecture, the *Maximum Likelihood* decoder, in a practical way.

On the other hand, MIDAS [49] tries to avoid the practical difficulties of synchronizing several transmitters altogether by using a single host with several radio front-ends and instead distributing the antennas in space. Although such a design poses challenges during the installation of an AP, since expensive coaxial cables are necessary, it benefits from the collocation of front-ends on the same host, so they can all share the same frequency and timing offsets. Since their phase offset is always the same, it only need be calibrated once, when the APs boot.

Distributed MIMO is also used in LTE-Advanced [50] but there it solves a slightly different problem. Cellular networks are designed with differ-

ent cells physically separated from one another, so that users are typically within the range of a single cell. If a user, though, is at the boundary between two cells, where typically the power received from either base station is low, the network can employ a feature called *Coordinated Multipoint* (CoMP), in which they use distributed MIMO to transmit to the same user, thus increasing the overall capacity to that user.

**Discussion:** Distributed MIMO is in theory an optimal strategy but it faces several practical limitations. Participating hosts must carry out complex calibration and synchronization while the computational cost of the implementation is high. More importantly, participating hosts must be interconnected by a fast, backplane, wired network and willing to share all transmitted information with one another, a requirement only met by enterprise deployments, since domestic users typically lack both the infrastructure and the willingness to share that data. This requirement also limits the usefulness of the scheme to the AP side of the network, since user laptops and cell phones do not have wired connections to use to cooperate with one another. Distributed MIMO has the potential to linearly increase throughput with the number of antennas but even in cases where the aforementioned issues are not a problem, phenomena like channel hardening and noise limit performance due to inter-stream interference and reduced delivered power at the receivers. In Chapter 5 we will examine in detail a Distributed MIMO system that takes these phenomena into account and mitigates them.

## 2.9 Frequency Division

Another way to avoid interference is to assign different parts of the available bandwidth to different hosts. In cellular systems, for instance, each

cell is assigned a unique set of frequencies in its neighborhood [10], thus mitigating inter-cell interference. This policy is very rigid: it precludes neighboring base stations using the same frequency even in cases where they would not affect each other's performance – for instance, when the interference they cause to each other is very weak, or when a cell doesn't use part of its assigned spectrum because of low user demand. *Inter-Cell Interference Coordination* (ICIC) [51, 52] is a set of techniques used in LTE-Advanced [53] and WiMax [54] that give more flexibility and allow base stations to share the available bandwidth. Of these, most prominent is *Fractional Frequency Reuse* [55], which allows each cell to use all the available bandwidth in its “core”, where its signal to its users is significantly stronger than any interference these users might receive from neighboring cells. At cell boundaries, however, where the signals from multiple base stations are at comparable levels, Fractional Frequency Reuse uses exclusive frequency allocation.

Within their communication range, base stations may use a technique called *Orthogonal Frequency Division Multiple Access* (OFDMA) [7]. Like OFDM, this method entails the division of available bandwidth into sub-channels, which are bundled into blocks, each of which can be assigned to a different user based on channel conditions and traffic demands. A cellular base station typically serves up to hundreds of users concurrently and efficient distribution of the available resources is critical. Huang *et al.* [56] present a theoretical analysis of the problem and prove that it is NP-hard, but also that approximation methods may provide solutions that are far from optimal. This is especially true when the number of users is close to the number of available subcarriers, when users' demands are close to system capacity, when the variance of SINRs across subcarriers is high, or

when subcarriers are highly correlated. Seong *et al.* [57] study the problem in the case where the number of available subcarriers is substantially higher than that of users and approach it as a non-convex optimization which they solve with Lagrangian dual decomposition for either sum-rate maximization or sum-power minimization. Li *et al.* [58] present a semi-distributed system, where a central radio network controller assigns large chunks of the bandwidth to base stations in order to avoid inter-cell interference, and each base station breaks down its assigned bandwidth into subchannels which it can assign independently to individual users.

In the context of 802.11 networks, FARA [59] is a combined frequency-aware rate adaptation and MAC protocol that uses OFDMA in the down-link of a single access point, assigning different users non-overlapping sets of subcarriers based on their respective wireless channels to the access point. FICA [60] builds on FARA, and applies OFDMA to uplink traffic as well. Tan *et al.* split the channel into chunks of 16 subcarriers and allows several hosts to contend for each of them and transmit concurrently. They also provide a mechanism that allows different hosts to synchronize their transmissions within 1  $\mu$ s, but this is looser than the requirement for Distributed MIMO.

WiFi-NC [61] splits a single wideband OFDM channel into multiple narrower subchannels, but unlike in FARA and FICA, these can be accessed independently from each other, without any further synchronization requirement between different transmitters. A problem that arises under such an approach is that when a host transmits within a band, its transmission causes interference to neighboring frequencies, called *out-of-band interference*. Standard 802.11 avoids this problem by spacing adjacent channels out with the use of *guard bands* and also by using hardware filters.

To suppress out-of-band interference between their subchannels, Chintalapudi *et al.* make use of sharp elliptical filter-banks. These, however, require longer durations for OFDM symbols. SampleWidth [62], on the other hand, lets transmitters adjust the widths of their channels to fixed values, allowing them to increase width for increased throughput or decrease it for increased range, lower power consumption, and better overall spectral density.

**Discussion:** Frequency division methods help systems significantly increase their spectral efficiency. Unlike static bandwidth allocations, they allow hosts to adapt their spectral use based on traffic demands and channel conditions. Nonetheless, they effectively apply unchanged the principle of exclusive frequency allocation, but in a more fine-grained manner. As we saw in Section 2.7, though, applying physical layer techniques can allow multiple hosts to concurrently use the *same* frequency, increasing overall capacity. In this respect, frequency division on its own leaves capacity on the table and that is why we do not use it in the systems presented in this work.

## 2.10 Power Allocation

As we saw in Chapter 1, the basic concept behind OFDM is breaking down available bandwidth into narrow subchannels – the subcarriers – which are orthogonal to one another, and thus can carry information independently and in parallel. Wireless hosts, though, have a limited power budget for their transmissions, both because of practical limitations, such as power consumption, and regulatory ones, so that many users can operate equipment within the same band. This brings up the issue of power allocation across the different subcarriers. 802.11 takes a straightforward approach to

this problem – it simply allocates the same amount of power to each subcarrier. Nonetheless, from a capacity maximization perspective for the Gaussian channel, the optimal strategy is called *waterfilling* [10]. Waterfilling allocates more power to subcarriers with higher SINRs and less to subcarriers with lower SINRs. Practical wireless systems, though, use modulations with discrete constellations. This means that applying more power to a subcarrier with high SINR yields diminishing returns after some point beyond which this additional power would be more useful if allocated to subcarriers with low SINRs. Based on that intuition, Lozano *et al.* [63] propose *mercury/waterfilling*, which takes into account both the SINR per subcarrier and the modulation scheme used and provides an optimal power distribution that minimizes mean square error between estimated and transmitted symbols. Liu *et al.* [64] extend waterfilling to OFDMA networks in a system that tries to allocate subcarriers to several users and apply power allocation to them with the objective of transmit power minimization.

Chung *et al.* [65] extend waterfilling to the multi-transmitter scenario with *iterative waterfilling* (IW) which, even though suboptimal, yields capacity improvements. In IW, a centralized controller applies waterfilling power allocation to each of the transmitters separately and then updates the SINR values and repeats the process until the solution converges. Wilson *et al.* [66] also adopt an iterative approach based on the *MaxSINR* algorithm. MaxSINR initially calculates the optimal MMSE filters at the receivers without any transmit beamforming. It then changes the direction of the problem and calculates the MMSE filters at the intended transmitters if these were to receive transmissions from the receivers. It then repeats this process iteratively, changing the direction of the problem in each iteration. Wilson *et al.* combine this technique with power control in order to achieve the

same SINR in the forward and reverse channel which helps the algorithm to converge. Ho *et al.* [67] propose a different approach: in order to allow for a middle ground between beamforming and interference nulling, they use the linear combination of the two solutions. They then apply binary power control iteratively, *i.e.*, if a subcarrier experiences too much interference above the thermal noise floor (which in a practical system would be the noise floor of the receive amplifier), they stop using the subcarrier for that stream and re-apply binary power control.

**Discussion:** Previous power allocation work focuses on the Gaussian channel and is mostly theoretical. By contrast, 802.11 systems use discrete constellations. Furthermore, they apply coding on top of the transmitted symbols using a single coder/decoder, and they use a single coding rate and modulation combination, *i.e.*, bitrate, for all subcarriers. This means that although subcarriers are orthogonal to one another, the fates of the bits they carry are closely tied, since if bad subcarriers introduce more bit errors than the coding layer can fix, all bits in a frame will be lost. Also, the algorithms mentioned solve the problem of power allocation for isolated hosts that do not experience interference, or too CPU-intensive to implement in a real system. As we will see in Chapter 4, taking these considerations into account is important when designing practical subcarrier power allocation systems.

## 2.11 Bit Loading and Subcarrier Switch-Off

Another way to deal with the variability of SNR across subcarriers in OFDM systems is to assign each one an appropriate modulation scheme according to its current channel conditions. This technique is called *Bit Loading*, and it is common practice in wired *Digital Subscriber Lines* (DSL) [68].

In the context of wireless systems, Yin *et al.* [69] provide an efficient algorithm that enables a single transmitter to serve multiple hosts by assigning each subcarrier to a different receiver based on its traffic requirements and bit-load them appropriately. Similarly, Kim *et al.* [70] propose an algorithm based on integer programming that approximates the same problem in an efficient manner, something critical since such an optimization is NP-hard.

A more practical alternative to Bit Loading is *Subcarrier Switch-Off* (SSO), where instead of trying to determine the ideal modulation for each subcarrier, transmitters simply do not use subcarriers that introduce too many bit-errors at all. Nitsche *et al.* show experimentally that choosing the right subcarrier subset for a given bitrate can yield significant throughput improvements for a single OFDM channel. Punal *et al.* compare schemes that perform SSO and power allocation across subcarriers in simulation and find that SSO is more useful than power allocation [71, 72]. They suggest combining the two methods.

**Discussion** Bit Loading and Subcarrier Switch-Off, like subcarrier power allocation, are intended to ameliorate the performance problems arising from variable channel conditions across different subcarriers in OFDM systems. Prior work has shown that they are promising methods, but only in the context of an interference-free channel, where each transmitter can make decisions independently about which subcarriers to use and how to choose transmission power or modulation method. Unfortunately, in the presence of interference this problem becomes more complex, since the decisions made by one transmitter affect those of all of its neighbors. In Chapters 4 and 5 we introduce a method that overcomes this complexity.

## 2.12 Conclusion

Growing wireless capacity demands, widespread, unplanned deployment of networks and wireless hosts, and limited spectrum have spurred research in communication in the presence of interference. As we saw in this chapter, methods that are based on exclusive allocation of resources are leaving resources unused, whether that is done in a centralized or distributed manner. Interference cancellation, although promising, has been adopted in few systems because of its fragility under realistic SINR regimes and modulation schemes used in practice. Interference nulling and alignment, on the other hand, are relatively easy to implement and deploy, although most systems that adopt them use them under centralized control with a wired backplane, *e.g.*, in enterprise settings. Furthermore, many commodity transceivers do not expose to the user functionalities that would allow operations such as channel estimation or precoding, both crucial for those methods. Moreover, phenomena including hardware imperfections, channel noise, poor channel diversity, etc. lead to inaccurate nulling, which can hamper performance in OFDM systems that do not take residual interference into account. Finally, changes in the power allocation across subcarriers by one host manifest as unpredictable changes in interference in neighboring networks.

In Section 1.2, we described three scenarios where interference has a significant impact on performance and we set out a set of lower-level requirements for each one. In Scenario A, an AP tries to decode a wanted signal while there are other unwanted, concurrent transmissions. Although this AP has a phased array antenna, it uses a single transceiver, and does not share any common infrastructure with its neighboring networks. Table 2.1

Lower-level goals for Scenario A	DIRC	IAC	OpenRF	802.11n+	SAM	MegaMIMO	OFDMA	Waterfilling	Mercury / waterfilling	CoS (Chapter 3)
Technique suitable for application at receiver		✓			✓					✓
No need for centralized controller					✓	✓				✓
No need for wired backplane					✓	✓				✓
Computational tractability within coherence time at acceptable hardware cost	✓	✓	✓	✓	✓	✓	✓	✓		✓
Use of the same subcarrier by multiple APs	✓	✓	✓	✓	✓	✓				✓
Suitability for use with single Tx/Rx chain and phased array	✓									✓

**Table 2.1:** Comparison of selected systems and methods on lower-level goals set for Scenario A.

shows how a selection of systems and methods we saw here does against the set of goals we set for Scenario A in the first chapter. As we can see, no pre-existing solution satisfies all our requirements. In Chapter 3, we will examine CoS, a system that allows an AP to decode a wanted signal under the presence of interference while satisfying all the requirements of Scenario A.

Scenario B also involves proximate APs from different networks, but this time each tries to transit to its own client. These APs have multiple receive and transmit chains and their transceivers can apply precoding separately for each subcarrier. Prior work typically focuses on interference nulling in such a case, but subcarrier SINR variability and residual interference have

Lower-level goals for Scenario B	DIRC	IAC	OpenRF	802.11n+	SAM	MegaMIMO	OFDMA	Waterfilling	Mercury / waterfilling	COPA (Chapter 4)
Technique suitable for application at transmitter	✓	✓	✓	✓		✓	✓	✓	✓	✓
No need for centralized controller				✓	✓					✓
No need for wired backplane				✓	✓					✓
Efficient distributed coordination of concurrent transmissions by APs				✓						✓
Suitability for use with single modulation/coding rate on all OFDM subcarriers									✓	✓
Computational tractability within coherence time at acceptable hardware cost	✓	✓	✓	✓	✓	✓	✓	✓		✓
Use of the same subcarrier by multiple APs	✓	✓	✓	✓	✓	✓				✓

**Table 2.2:** Comparison of selected systems and methods on lower-level goals set for Scenario B.

a significant impact on performance. Table 2.2 shows how our selection of prior work does against the requirements of our scenario. The last column shows how COPA does, a system that satisfies all the requirements of Scenario B, which we will examine in detail in Chapter 4.

Finally, if several APs share a common, fast, wired backplane, and they are all managed by the same organization, we can significantly increase the aggregate throughput and the number of hosts served by using Distributed MIMO. Nonetheless, subcarrier SINR variability and inter-stream interference can hurt performance here too. Table 2.3 compares our selection of the

Lower-level goals for Scenario C	DIRC	IAC	OpenRF	802.11n+	SAM	MegaMIMO	OFDMA	Waterfilling	Mercury / waterfilling	PADM (Chapter 5)
Technique suitable for application at transmitter	✓	✓	✓	✓		✓	✓	✓	✓	✓
Suitability for use with single modulation/coding rate on all OFDM subcarriers									✓	✓
Computational tractability within coherence time at acceptable hardware cost	✓	✓	✓	✓	✓	✓	✓	✓		✓
Use of the same subcarrier by multiple APs	✓	✓	✓	✓	✓	✓				✓

**Table 2.3:** Comparison of selected systems and methods on lower-level goals set for Scenario C.

prior work against the goals we set in the first chapter. In the last column we see PADM, a system that satisfies all our requirements for Scenario C, which we will examine in Chapter 5.

## Chapter 3

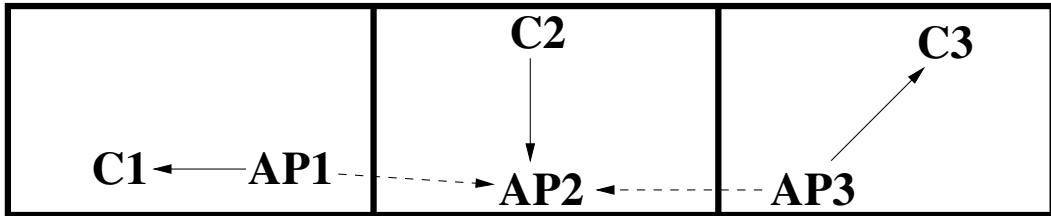
# Cone of Silence (CoS)

Spurred by the availability of low-cost commodity radio hardware and freely usable unlicensed spectrum, users have enthusiastically embraced 802.11 wireless networking in home and office environments. As these networks proliferate rapidly, particularly in populous urban areas, their deployment density increases significantly. Measurements of 802.11 base station deployments in major US cities taken in 2005 already showed thousands of cases in which four or more 802.11 access points mutually interfered [13]. As only three non-overlapping channels are available in the 2.4 GHz band and eleven in the 5 GHz band, while at the same time increased throughput requirements have made channel bundling commonplace, these increasingly dense deployments pose a wireless capacity challenge—physically proximal networks must share finite bandwidth.

Consider a dense deployment of 802.11 networks on overlapping channels, typical in today’s residential and commercial areas, as shown in Figure 3.1. The occupant of each of three apartments (or offices) operates his own

---

The work in this chapter was done in 2009 with what was at the time state-of-the-art commercially available hardware.



**Figure 3.1:** Typical dense 802.11 deployment in apartments or offices. AP1, AP2, and AP3 are access points; C1, C2, and C3 are clients. Solid arrows denote desired transmissions; dashed arrows denote unintended interference.

access point (AP) with a standard omnidirectional antenna, and because of their close proximity, AP1 and AP3 interfere with reception by AP2. In particular, if client C2 and AP1 are hidden from one another, client C2 may transmit to its AP, AP2, and AP1 may simultaneously transmit to its client C1. Consider C2 as the sender, AP2 as the receiver, and AP1 as the interferer. Interference from AP1 will reduce the throughput C2 achieves at AP2. Indeed, more than one interferer may transmit concurrently; *e.g.*, AP3 might transmit to clients of its own, too, further interfering at AP2.

When multiple wireless networks operated by independent, non-cooperating individuals interfere, a receiver in one network derives no benefit from successfully decoding transmissions from an interferer in another network; data from another network are typically of no interest. Moreover, operators of these networks do not centrally coordinate transmit schedules, or share decoding information among nodes. As we saw in Chapter 2, many recent advances in mitigation of interference have targeted environments where one enterprise operator closely coordinates multiple cooperating senders [9, 30], or where concurrent transmissions' contents are all of interest to a single receiver (*i.e.*, where interferers are part of the *same* network) [26]. In contrast, in this chapter, we specifically focus on mitigating interference in the ubiquitous “chaotic,” non-cooperative

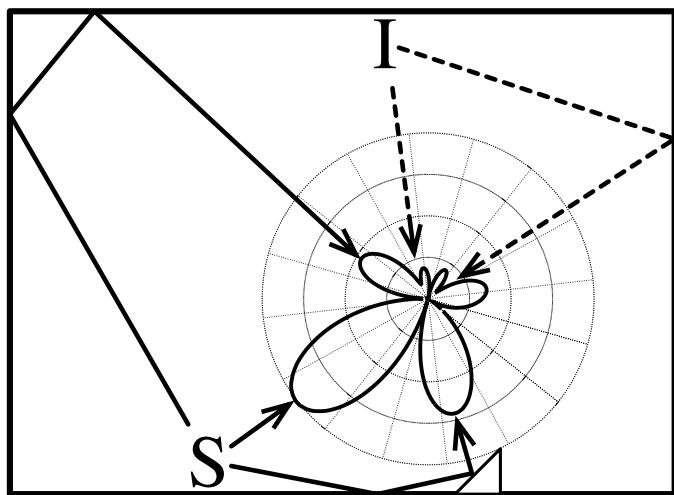
deployments described above. Specifically, we will focus on Scenario A, described in Section 1.2, where an AP tries to decode a useful transmission from its own client, while other clients belonging to neighboring networks transmit concurrently, thus causing interference.

Directional antennas are capable of improving throughput on wireless links in such dense, interference-rich deployments. The throughput achievable on a link depends on how well the receiver can discern a sender of interest’s signal, while distinguishing it from competing background noise and interference from other concurrent senders—on the *signal-to-interference-plus-noise ratio* (SINR). The greater the bitrate at which a packet is transmitted, the greater the SINR with which the receiver must receive the packet in order to decode it successfully. And at a given transmit bitrate, as SINR increases, bit-error rate (BER) decreases, reducing costly link-layer retransmissions.

Extracting the greatest SINR from a receiver’s directional antenna entails solving two distinct problems. First, how can one direct gain toward a sender of interest’s signal, thus improving the strength with which it is received? And second, how can one *avoid* directing gain toward interfering signals from concurrent transmitters, and thus *null* interference? A system may independently address either or both of these problems. Solving either increases SINR, and can thus improve throughput.<sup>1</sup> The relative benefits to SINR of directing gain toward a sender’s signal *vs.* nulling interference depend heavily on the deployment scenario. In an interference-rich environment, nulling is *vital* to achieving the full SINR and throughput

---

<sup>1</sup>Complementary arguments apply when a sender transmits with a directional antenna. In this chapter, we focus on *reception* using directional antennas, though as we discuss in Section 3.3, the techniques described will be useful at senders, too. In Chapters 4 and 5 we will focus on the downlink.



**Figure 3.2:** Example of multipath propagation between a sender  $S$ , interferer  $I$ , and a receiving directional antenna. Solid arrows represent components from  $S$ ; dashed from  $I$ . Boundaries are reflective walls; the triangle represents a reflective object.

improvements a directional antenna can offer. Lakshmanan *et al.* offer a technique for maximizing gain toward an indoor sender of interest [36], but this technique does not explicitly null interferers.

Multipath propagation, commonplace indoors, significantly complicates effective use of directional antennas by causing a sender's signal to arrive at a receiver in multiple components from unpredictable bearings, each with a different phase.<sup>2</sup> Figure 3.2 offers an idealized illustration of this phenomenon in a simple topology, where a receiver equipped with a directional antenna attempts to receive from sender  $S$  while an interferer  $I$  transmits concurrently. The solid arrows indicate multiple components from  $S$  while the dashed arrows indicate multiple components from  $I$ . We see that multiple components arise from reflections of each transmitter's signal (off walls and any of the many other reflective objects in the indoor

---

<sup>2</sup>For simplicity of exposition, we do not discuss the details of phase in this introduction; we describe these details in Section 3.1.

setting) that depend not only on the locations of the sender and receiver, but on the time-varying minutiae of the physical surroundings. In order to maximize SINR successfully, and extract full benefit from directionality, the receiver must configure its antenna such that high-gain *lobes* are directed toward the incident bearings of  $S$ 's signal, while low-gain *nulls* are directed toward the incident bearings of  $I$ 's signal. The heavy line undulating about the receiver represents just such a *gain pattern* for its directional antenna. Radial distance from the center of the pattern to this line indicates the gain of the antenna in dB in that radial direction, and lobes are oriented toward  $S$ 's components, while nulls are oriented toward  $I$ 's.

Software-steerable, *phased array* antennas, such as the one used in this work (described in Section 3.1.2), allow quick shaping of the gain pattern entirely electrically, under software control, *without* any mechanical motion. These antennas provide good directional characteristics, but because of their size, they are only suitable for static hosts such as an AP. While phased array antennas have previously been used successfully indoors with fixed, essentially single-lobe beam shapes [30], such beam shapes cannot maximize SINR as effectively as ones flexibly and dynamically *customized* in accordance with the specific multiple arrival bearings of senders' and interferers' signals.

In this chapter, we present Cone of Silence (CoS), a technique for improving throughput under interference in 802.11 wireless networks. CoS incorporates two main techniques: *SamplePhase*, a method for accurately, robustly, and efficiently deriving a custom pattern for an AP's antenna that maximizes the signal strength received from a specific sender or interferer; and *Silencer*, a method that, given signal-maximizing patterns for a sender and one or more interferers, produces a single pattern that simultaneously nulls

the interferers while maximizing signal strength from the sender of interest, thus maximizing SINR and throughput.

An evaluation of a prototype of CoS on an indoor 802.11b/g testbed demonstrates that CoS can improve a sender’s throughput under interference over that achieved by an omnidirectional receiver by between  $1.6\times$  and  $17\times$ , and that CoS improves receive throughput by nulling one or two concurrent interferers. CoS achieves these substantial performance improvements while offering the following key properties:

- An AP using CoS can null even *uncooperative* interferers from which it can receive packets—CoS does not require APs to schedule transmissions collaboratively, as do previous techniques for mitigating interference with directional antennas [30] or multiple antennas [9].
- Unlike schemes that receive many concurrently transmitted packets, but require processing-intensive full decoding of each one [26, 9], CoS can null multiple concurrent interferers using multiple antennas connected to only *a single commodity 802.11 radio*. To our knowledge, CoS’s Silencer is the first such implementation of a decorrelator [10].

### 3.1 Design

We begin with a brief overview of the use scenario for CoS, followed by a primer on the phased array antenna hardware platform on which CoS is built. Thereafter, we present SamplePhase, an algorithm for measuring the wireless channel between the CoS AP and other radios. We then present the design of Silencer, which builds on SamplePhase to simultaneously steer our AP towards associated clients and null one or more interferers.

### 3.1.1 Use Scenario

Consider again the topology in Figure 3.2. Recall that in order to improve SINR at a receiving AP equipped with a beam-steerable directional antenna, CoS must derive a pattern for the antenna that gathers path components from the sender of interest,  $S$ , while nulling path components from the interferer,  $I$ . CoS goes about that goal in two logical steps:

- First, CoS considers the sender of interest and each interferer individually. For each such remote transmitter, a CoS AP applies the SamplePhase algorithm, described in Subsection 3.1.3, to derive one receive pattern for each remote transmitter that maximizes received signal strength from that remote transmitter alone.
- Second, for each sender of interest, a CoS AP applies the Silencer algorithm, described in Subsection 3.1.5, whose input consists of the receive pattern that maximizes signal strength at the AP from the sender of interest, as well as one pattern for each interferer that does the same for that interferer. Silencer produces one pattern for each sender of interest that nulls all interferers while directing gain to maximize signal strength from the sender of interest.

CoS must determine the identities of interferers. Doing so for 802.11 transmitters who interfere strongly at the AP is not difficult; an AP may simply scan the channels that overlap its own periodically, and record the MAC addresses of any senders that occupy the channel heavily. It is precisely the strongest interferers that stand the greatest chance of reducing a sender of interest's throughput to the AP that will be most easily identified in this fashion. When receiving from *any* sender of interest, CoS *always* nulls toward *all* interferers that it has identified. CoS cannot be certain that an

interferer is sending at any given time, and so may needlessly null that interferer. That choice would be problematic if nulling significantly reduced throughput from the sender of interest as a side effect. In Subsection 3.2.2, we present experimental evidence to argue that for this system, nulling does not do so—in effect, that nulling an interferer tends to be “safe” for the sender of interest.

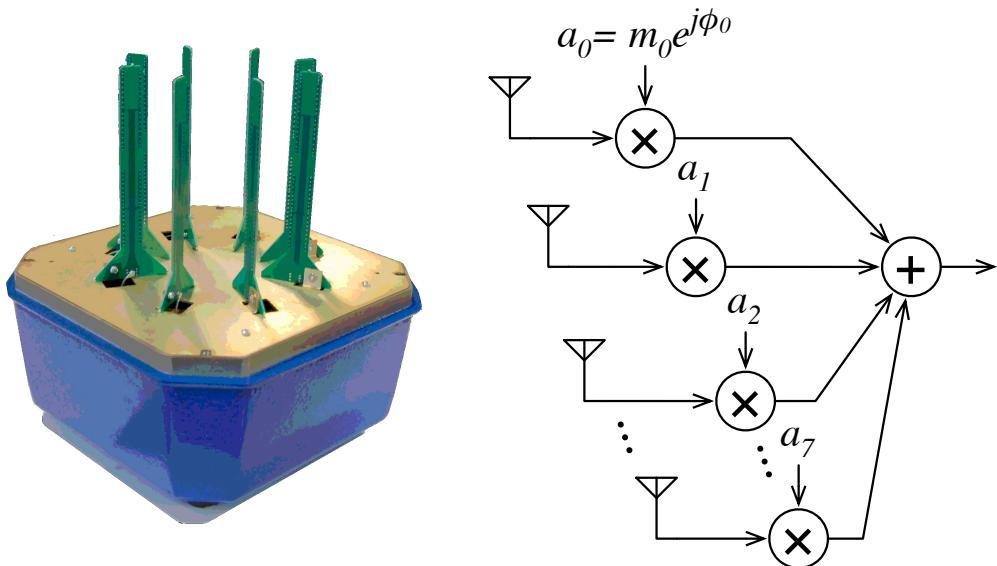
Like any AP, CoS maintains a list of associated clients. CoS stores the Silencer-produced, throughput-maximizing pattern tailored to each associated client, and configures its directional antenna to the appropriate such pattern each time a client transmits to the AP. To do so, however, CoS must have foreknowledge of when each client will transmit.<sup>3</sup> As do others who have proposed interference mitigation techniques for 802.11 networks [9], we envision that an AP would use a TDMA schedule among its own clients to allow it to predict when each client transmits. Because TDMA scheduling among a base station and its clients is a well-understood area in the literature – the original 802.11 specification includes the PCF MAC, a TDMA approach [19] – we leave the building of a TDMA MAC implementation out of scope for this thesis, and focus on nulling interference assuming that a CoS-enabled AP can use TDMA to predict which of its clients will send.

### 3.1.2 Hardware Platform

We have built the CoS prototype atop the Phocus phased array 802.11b/g antenna [73] manufactured by Fidelity Comtech Inc., shown in the left of Figure 3.3. The array consists of eight elements spaced equally on the cir-

---

<sup>3</sup>It is important to note that the AP need *only* predict transmissions from *its own clients* – as explained above, CoS assumes that interferers, who are uncooperative because they are part of other networks, transmit constantly.



**Figure 3.3:** *Left:* Phocus phased array 802.11b/g antenna with eight elements.  
*Right:* simplified array model.

cumference of a circle, each of which consists of four stacked dipoles. A signal processing module (right of Figure 3.3) mixes the signal from or to element  $k$  with a complex gain  $a_k$  allowing adjustment of its phase  $\phi_k$  and magnitude  $m_k$  during reception and transmission. The eight resulting signals are summed and in turn connect to an antenna port of a standard Atheros AR5413 802.11 chipset. The gain and phase applied to each element may be independently controlled in software, the phase in single-degree increments between  $-\pi$  and  $\pi$  radians and the gain in 1% increments between 0% and 100%.

Taken together, the phase shifts and magnitudes configured for all elements define a complete pattern. Changing the array's pattern takes approximately 120  $\mu$ s.

As supplied by the manufacturer, the antenna's software supports only 17 factory-configured "stock" patterns: one omnidirectional, and the re-

maining 16 “high-gain”, each of which consists of a single high-gain lobe approximately 0.75 radians in width, pointing toward one of 16 equally spaced directions about the antenna’s center. The CoS software is not bound by this restriction; it can configure each element independently to any of the supported phase and gain values, yielding a vast variety of possible multi-lobed patterns.

### 3.1.3 Measuring the Channel: SamplePhase

The first challenge we face is measurement of the wireless channel from each client to all of the AP’s antennas. Reflections off walls and other objects found in the typical indoor home or office mean that indoor wireless networks operate in the presence of strong multipath reflections, wherein transmissions arrive from multiple directions at the AP.

In order to produce the best end-to-end performance, we argue that a channel measurement algorithm should satisfy the following objectives:

- *Performance*: The algorithm should produce measurements that result in the best throughput.
- *Efficiency*: The overhead of the algorithm should be as low as possible without sacrificing performance.
- *Reliability*: The algorithm should meet the above two objectives consistently, with low performance variance, even in challenging wireless environments.

In prior work, Lakshmanan *et al.* [36] have proposed a channel measurement algorithm that we improve upon here. We experimentally compare the two algorithms in Subsection 3.2.1 and briefly touch upon efficiency differences between the two algorithms.

**The SamplePhase algorithm.** To reduce complexity, our approach leverages received signal strength (RSS) readings measured at a client from packets sent by the AP, and takes advantage of the reciprocity of the wireless channel. In our prototype, the CoS AP sends measurement probes to all remote nodes, which record RSS measurements and return them wirelessly to the AP. In a production deployment, we envision that the CoS AP would send 802.11 null data frames to its clients to elicit ACKs, and measure RSS on these returning ACKs (in fact, the Phocus array software already implements this functionality as shipped). For interferers, CoS could simply measure RSS on frames received by the AP – whether these are ACKs elicited by the interferer’s own CoS AP or opportunistically overheard data or control frames. In the latter case, our AP needs to know in advance that a certain host will transmit a frame in order to use the appropriate pattern during reception. This is true for frames such as link layer ACKs.

Consider an approximation of the wireless channel between the  $k^{\text{th}}$  AP element and the client as a single complex number of a certain phase  $\theta_k$  and magnitude  $\sqrt{P_k}$ .<sup>4</sup> Then based on RSS readings, SamplePhase outputs a set of eight channel measurements:  $\sqrt{P_1}e^{j\theta_{r1}}, \sqrt{P_2}e^{j\theta_{r2}}, \dots, \sqrt{P_8}e^{j\theta_{r8}}$ , where  $r$  is the index of a reference element in the array, and  $\theta_{kl} = \theta_l - \theta_k$  is the phase of element  $k$  relative to element  $l$ . SamplePhase only measures the relative differences between the channel phases  $\theta_k$ , since these determine beam shape.

---

<sup>4</sup>The 802.11b/g wireless channel is 20 MHz wide and so in fact cannot be completely characterized by a single complex value. This limits the performance of any MIMO method that is not applied separately for each subcarrier in the case of channels that vary significantly over their different frequencies. The availability of only a “summary” complex value averaged over the entire 20 MHz channel is a platform limitation – the Atheros AR5413 in the Phocus Array AP only reports RSSI for the whole channel on average, and the analog phase shifters of the Phocus Array apply the same phase shift across the whole 20 MHz channel.

SamplePhase measures the individual channel magnitudes from each individual element  $\sqrt{P_k}$  ( $k = 1 \dots 8$ ) directly. It transmits 25 contiguous bursts of three *probe* packets each from each individual element of the AP to the remote node to which the channel is being measured. The bursts are interleaved across elements, so that a period of interference impacts just a small number of packets from any particular element.

SamplePhase's phase measurements are based on the following observation about the wireless channel. Suppose the phase of the channel from element  $k$  to the remote node is some value  $\theta_k$ , and the phase of the channel from element  $l$  to the remote node is another value  $\theta_l$ . Further, suppose that the AP transmits data with a phase difference  $\delta$  between elements  $k$  and  $l$  that we choose and program into the AP. Then, by the principle of superposition (Lakshmanan *et al.* provide a detailed derivation [36]), the power of the elements' combined transmissions at the client is

$$P_{kl}(\delta) = P_k + P_l + 2\sqrt{P_k P_l} \cos(\theta_l - \theta_k + \delta). \quad (3.1)$$

Rearranging the above, we find the following:

$$\cos(\theta_{kl} + \delta) = \frac{P_{kl}(\delta) - (P_k + P_l)}{2\sqrt{P_k P_l}} \quad (3.2)$$

The above suggests the following way of estimating  $\theta_{kl}$ : using empirically measured values of  $P_k$ ,  $P_l$ , and  $P_{kl}(\delta)$ , sample the expression on the right-hand side of Equation 3.2 at one or more values of  $\delta$ . Then, the best estimator of  $\theta_{kl}$  will minimize the sum of squared errors between the empirically measured values from the right-hand side of Equation 3.2 and computed values from the left-hand side of the same equation. Sampling multiple evenly spaced values of  $\delta$  removes any phase ambiguity with high likeli-

hood, as explained in Section 3.1.4. For simplicity and to overcome practical limitations on the number of antenna patterns that the Fidelity Comtech array can store at once, SamplePhase uses four evenly spaced values of  $\delta$  for its phase measurements.

### 3.1.4 The SamplePhase estimator

Starting from Equation 3.2, define

$$\zeta(\delta) = \frac{P_{kl}(\delta) - (P_k + P_l)}{2\sqrt{P_k P_l}}. \quad (3.3)$$

For a given base element, we program the AP to transmit with two elements active for  $\delta = \{-\frac{\pi}{2}, 0, \frac{\pi}{2}, \pi\}$ , and compute values for  $\zeta(-\frac{\pi}{2})$ ,  $\zeta(0)$ ,  $\zeta(\frac{\pi}{2})$ ,  $\zeta(\pi)$  directly from multiple empirical RSS measurements.

We now explain how to compute the SamplePhase estimator  $\hat{\theta}_{kl}$  for the true channel phase difference between elements  $k$  and  $l$ ,  $\theta_{kl}$ . Define the measured error at angle  $\delta$ ,  $\epsilon(\delta)$ , as follows:

$$\epsilon(\delta) = \zeta(\delta) - \cos(\theta_{kl} + \delta). \quad (3.4)$$

Using the least-squares method, we fit the measured values  $\zeta(\cdot)$  to a sinusoidal function. Let  $S(\hat{\theta}_{kl})$  be the sum of the squared errors  $\epsilon(\cdot)$ , as a function of the SamplePhase estimator:

$$S(\hat{\theta}_{kl}) = \epsilon^2\left(-\frac{\pi}{2}\right) + \epsilon^2(0) + \epsilon^2\left(\frac{\pi}{2}\right) + \epsilon^2(\pi). \quad (3.5)$$

Using Equation 3.4 to expand Equation 3.5 and applying trigonometric identities, we find

$$\begin{aligned} S(\hat{\theta}_{kl}) &= \zeta^2\left(-\frac{\pi}{2}\right) + \zeta^2(0) + \zeta^2\left(\frac{\pi}{2}\right) + \zeta^2(\pi) \\ &\quad + 2 - 2 \left[ \zeta\left(-\frac{\pi}{2}\right) \sin(\hat{\theta}_{kl}) + \zeta(0) \cos(\hat{\theta}_{kl}) \right. \\ &\quad \left. - \zeta\left(\frac{\pi}{2}\right) \sin(\hat{\theta}_{kl}) - \zeta(\pi) \cos(\hat{\theta}_{kl}) \right]. \end{aligned} \quad (3.6)$$

In order to minimize  $S$ , we seek the zeros of  $S'$ :

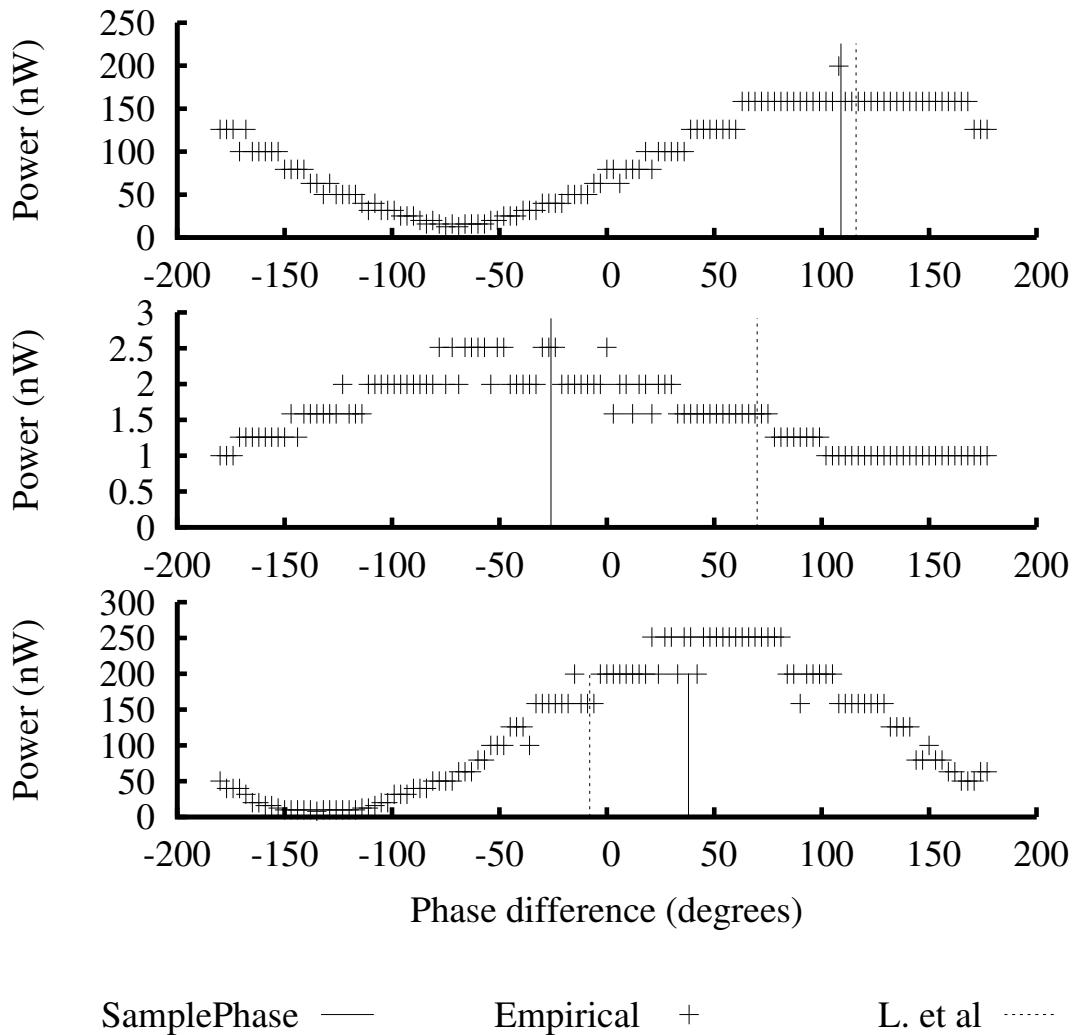
$$\begin{aligned} S'(\hat{\theta}_{kl}) &= -2\zeta\left(-\frac{\pi}{2}\right) \cos(\hat{\theta}_{kl}) + 2\zeta(0) \sin(\hat{\theta}_{kl}) \\ &\quad + 2\zeta\left(\frac{\pi}{2}\right) \cos(\hat{\theta}_{kl}) - 2\zeta(\pi) \sin(\hat{\theta}_{kl}) \\ &= 0 \\ \therefore \hat{\theta}_{kl} &= \arctan\left(\frac{\zeta\left(-\frac{\pi}{2}\right) - \zeta\left(\frac{\pi}{2}\right)}{\zeta(0) - \zeta(\pi)}\right) \end{aligned} \quad (3.7)$$

where  $\arctan$  is generalized to have exactly one root in  $[0, 2\pi)$  given the numerator and denominator in Equation 3.7.

From the above derivation, we can see that it does not matter which angles are used evaluate  $\zeta$ , so long as these angles are spaced evenly, with a  $\frac{\pi}{2}$  difference between them.

Now, suppose we have  $n$  quadruplets of evenly spaced measurements starting at an arbitrary phase  $\phi$ . Using further trigonometry, Equation 3.7 fully generalizes to Equation 3.8.

$$\hat{\theta}_{kl} = \arctan\left\{ \frac{\sum_{i=1}^n \left[ \left( \zeta\left(\frac{2\pi i}{n} + \phi\right) - \zeta\left(\frac{2\pi i}{n} + \pi + \phi\right) \right) \cos\left(\frac{2\pi i}{n} + \phi\right) - \left( \zeta\left(\frac{2\pi i}{n} + \phi\right) - \zeta\left(\frac{2\pi i}{n} + \pi + \phi\right) \right) \sin\left(\frac{2\pi i}{n} + \phi\right) \right]}{\sum_{i=1}^n \left[ \left( \zeta\left(\frac{2\pi i}{n} + \phi\right) - \zeta\left(\frac{2\pi i}{n} + \pi + \phi\right) \right) \cos\left(\frac{2\pi i}{n} + \phi\right) + \left( \zeta\left(\frac{2\pi i}{n} - \frac{\pi}{2} + \phi\right) - \zeta\left(\frac{2\pi i}{n} + \pi + \phi\right) \right) \sin\left(\frac{2\pi i}{n} + \phi\right) \right]} \right\} \quad (3.8)$$



**Figure 3.4:** Empirical power measurements ( $P_{kl}$ ) of RSS for three representative element pairs for an AP-remote node link when pairs of AP antenna elements transmit simultaneously with varying phase difference ( $\delta$ ) to a remote node (“Empirical” points). The SamplePhase and Lakshmanan *et al.* estimates of the peak of the sinusoid appear as vertical lines.

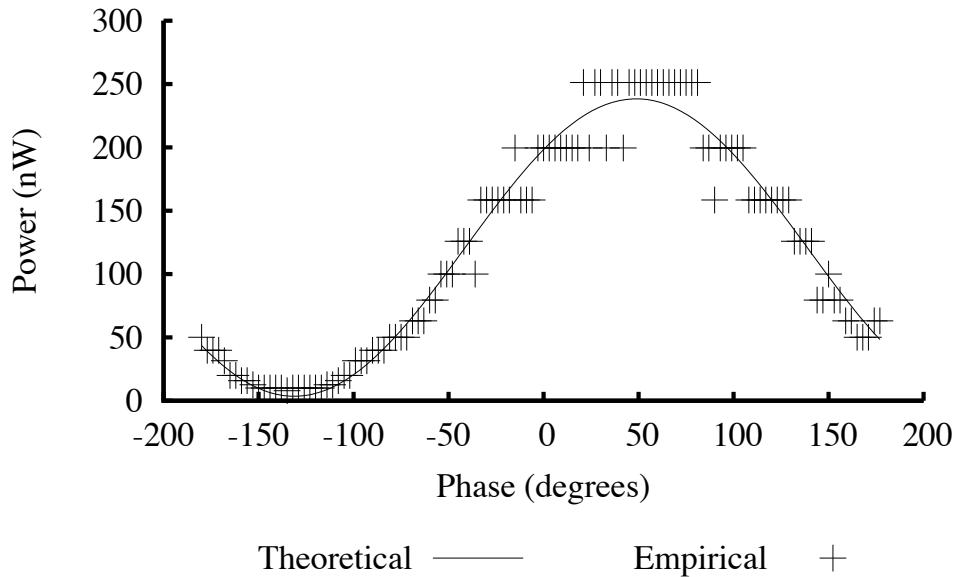
**SamplePhase microbenchmarks.** Figure 3.4 shows example measurements of  $\sqrt{P_{kl}(\delta)}$  for three representative element pairs on an AP-client link in our testbed (described in Section 3.2), as we vary  $\delta$  between  $-\pi$  and  $\pi$  radians. We see the expected sinusoidal relationship, and note that estimating  $\theta_{kl}$  by using measurements of  $P_{kl}$  near the peak or trough of the

sinusoid may decrease accuracy, because of the quantization of the sample data and the decreased slope of the sinusoid near those points. On the same plots, the vertical lines represent the peaks of the sinusoid as predicted by the estimators of  $\theta_{kl}$  produced by the SamplePhase and the Lakshmanan *et al.* methods. We see that by using multiple sample points, SamplePhase finds the peaks of the sinusoidal data better than the prior method for these representative element pairs.

	Mean	Standard deviation
SamplePhase	10	11
L. et al	26	26.2

**Table 3.1:** Mean and standard deviation of absolute error for SamplePhase and L. *et al.* methods.

Table 3.1 lists the mean and the standard deviation of the absolute error for both methods. To derive these results, we fit the empirical data to the sine wave whose phase minimizes the sum of squared errors (Figure 3.5). Note that this fitting to a sine wave uses many more samples than both SamplePhase and Lakshmanan *et al.*'s method. We measured the absolute error of each of the two methods by computing the absolute difference between each method's estimate and the peak of the best-fit sine wave. Then looking across all links in our testbed (shown in Figure 3.8 on p. 96), and at all array element pairs on every testbed link, we computed the average absolute error and standard deviation of the absolute error for both SamplePhase and the Lakshmanan *et al.* method. The results are shown in Table 3.1. In this table, we see that SamplePhase offers both a lower absolute mean error and a lower standard deviation of absolute error. We conclude that SamplePhase more accurately finds the antenna phase that maximizes the signal strength when two elements send together.



**Figure 3.5:** Empirical data with the theoretical fit superimposed.

In Section 3.2, we show that SamplePhase furthermore finds better overall beamforming and interference-canceling patterns than the prior method, as determined by the end-to-end metric of throughput.

**Beamforming toward a client.** Once the AP has measured the channel between itself and a client, it can beamform its transmissions to or receptions from that client by weighting the  $k$ th element's input by the channel measurement to the  $k$ th element,  $\sqrt{P_k}e^{j\theta_{kr}}$ , where  $r$  is the reference element chosen during SamplePhase's measurement. This results in co-phasing the signals from all antennas so that they align and constructively interfere. The combination of co-phasing and weighting proportional to  $\sqrt{P_k}$  maximizes signal-to-noise ratio (SNR) at the receiver and is known as maximal ratio combining (MRC) in the literature [10]. MRC does not maximize signal to interference plus noise ratio (SINR), however, and so interfering transmissions will impact a beamforming AP's throughput, as we show in

Section 3.2. We therefore seek a way to null interfering clients and maximize SINR.

### 3.1.5 Nulling Interferers: Silencer

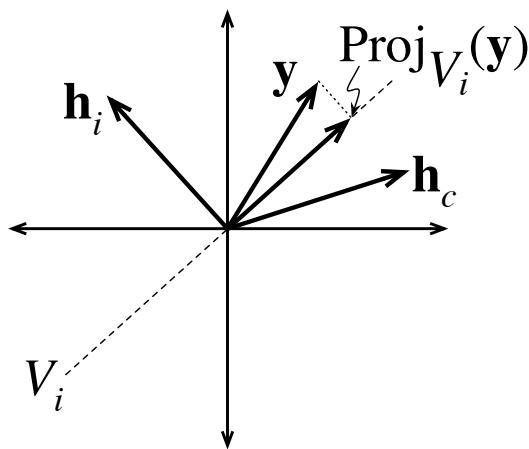
Silencer is an implementation of a decorrelator [10], a computational structure that allows distinct signals to be received concurrently. What distinguishes Silencer from other decorrelator implementations is that Silencer can recover a signal from a sender of interest while nulling other concurrently received signals *without* decoding these other signals.

Using channel measurements from the methods in Subsection 3.1.3, we can represent the channels to clients as vectors in an eight-dimensional space (since our AP has eight elements):

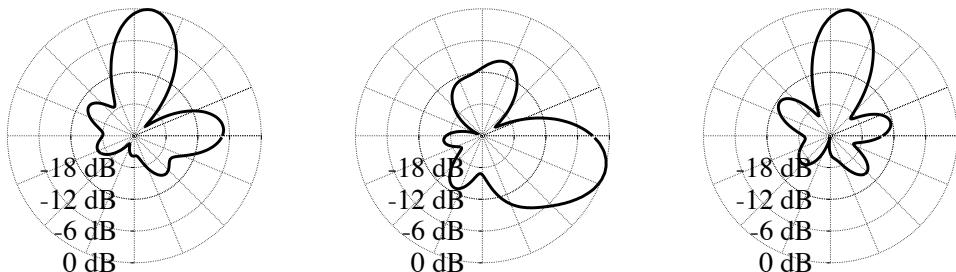
$$\mathbf{h}_c = \begin{bmatrix} \sqrt{P_1}e^{j\theta_1} \\ \sqrt{P_2}e^{j\theta_2} \\ \vdots \\ \sqrt{P_8}e^{j\theta_8} \end{bmatrix} \quad (3.9)$$

where the measurements for  $\mathbf{h}_c$  are taken at client  $c$ . To null interferer  $i$  (either another AP or an interfering client), the AP measures the channel  $\mathbf{h}_i$  between itself and the interferer, and using the Gram-Schmidt algorithm [74], computes a basis in  $\mathbb{C}^8$  for the vector space orthogonal to  $h_i$  (indicated by  $V_i$  in Figure 3.6). Then, Silencer projects the received signal  $\mathbf{y}$  onto  $V_i$  (indicated by  $\text{Proj}_{V_i}(\mathbf{y})$  in Figure 3.6).

After the interference nulling step, Silencer directs gain in the direction of the intended client's channel  $\mathbf{h}_c$  (in the  $V_i$  vector subspace). If we represent projection onto  $V_i$  with the  $8 \times 8$  complex matrix  $\mathbf{Q}_i$ , the overall operation on the received signal  $\mathbf{y}$  is therefore  $(\mathbf{Q}_i \mathbf{h}_c)^* \mathbf{Q}_i \mathbf{y}$ . We program the AP



**Figure 3.6:** By projecting the received signal  $\mathbf{y}$  onto the vector subspace orthogonal to an interferer's channel  $\mathbf{h}_i$ , Silencer (shown here in  $\mathbb{R}^2$  for ease of exposition) nulls the signal from the interfering client.



**Figure 3.7:** Impact of Silencer on antenna gain pattern: note that these empirical figures show gain *vs.* direction, but do not show how the antenna manipulates the phase of the received signals. *Left:* an MRC gain pattern maximizing SNR for a sender. *Center:* an MRC gain pattern maximizing SNR for an interferer. *Right:* the resulting Silencer gain cancelling the interferer and beamforming toward the sender.

with the eight-element, complex-valued vector  $\mathbf{Q}_i^* \mathbf{Q}_i \mathbf{h}_c$  to implement this operation.

**Generalization to multiple interferers.** Silencer easily generalizes to multiple interferers  $i_1$  to  $i_7$ , each with a different channel estimate  $h_{i_1} \dots h_{i_7}$ , by using the Gram-Schmidt process to construct a vector subspace orthogonal

to the span of all the interference vectors. In Subsection 3.2.4, we present experimental results nulling up to two out of three simultaneously transmitting senders.

**Practical limitations in nulling interferers.** Although CoS can theoretically entirely remove interference from up to seven simultaneous interferers, several practical design issues limit real-world system performance:

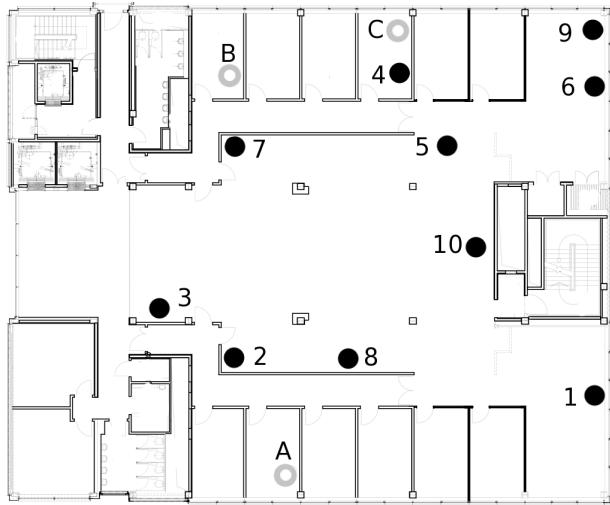
- *Hearing the interferer.* In order to compute  $\mathbf{h}_i$ , CoS needs to receive a sufficient number of packets from interferer  $i$ . This precludes nulling the most distant interferers, since our commodity hardware detects packets only down to  $-94$  dBm.
- *Estimating the channel to the interferer.* The accuracy of the channel estimation algorithm will impact the degree to which CoS can beamform towards clients and null interferers. For OFDM modulations, since CoS measures and beamforms across all OFDM subcarriers simultaneously, it does not capture inter-subcarrier differences in the 20 MHz channel. This is a practical tradeoff: measuring inter-subcarrier differences would require a software-defined radio, or a PHY interface that returns per-subcarrier RSS readings. CoS sacrifices some performance for the simplicity of running on a 2009-era commodity hardware platform and the speed of only requiring three  $\mathbb{C}^8$  matrix multiplications and a Gram-Schmidt iteration in order to compute a beamforming pattern that nulls a new interferer.
- *Adapting when an interferer ceases sending.* When CoS nulls an interferer that has since ceased transmission, it sacrifices some amount of signal power that would have improved the overall SINR had it not nulled that interferer. In Subsection 3.2.2 we quantify the throughput

impact of nulling towards an interferer despite that interferer's not transmitting.

- *The degree of similarity between the client's channel and the sender's channel.* The more orthogonal the sender of interest's channel is to each of the interferers' channels, the less of the sender's signal Silencer will null along with those of the interferers. Fortunately, since CoS uses eight antennas and complex-valued channel vectors, there are many more degrees of freedom than the two shown in Figure 3.6. We examine how well CoS can null interferers end-to-end in Section 3.2.
- *The time between channel estimation and interference cancellation.* One important concern is how much time the beam-steering and interference-canceling patterns we derive correspond to the channel's behavior, because their longevity, together with the time needed to measure the channel, determine CoS's overhead. In Subsection 3.2.3 we measure pattern lifetime.

## 3.2 Evaluation

There are several key performance questions surrounding indoor interference nulling. First, by how much does SamplePhase increase received signal power and throughput on a single link? Next, how well does Silencer null interference and allow that same link to function in the presence of an interferer? For both of the preceding questions, how long do the patterns derived last? Finally, how many simultaneous interferers can CoS null? In this section, we answer these questions using experiments in a typical indoor office environment, on the 13-node testbed on the 7th floor of the UCL CS building, shown in Figure 3.8.



**Figure 3.8:** The indoor office environment and wireless network topology for the experiments in CoS. Filled dots represent nodes with a single omnidirectional antenna and hollow dots represent nodes with a phased-array antenna.

**Experimental setup.** Our testbed consists of three Phocus phased-array antenna nodes on which we run CoS and 10 Soekris nodes each equipped with a single omnidirectional antenna. All nodes use Atheros 5413 WiFi cards and the madwifi driver under Linux. Soekris nodes use madwifi v0.9.4, whereas the phased arrays use madwifi v0.9.2.1, including patches from the OpenWRT project (for back-ported bug fixes) and Fidelity Comtech (for antenna phase and gain control functionality). To explore many different topologies, we use Soekris nodes to transmit as either senders or interferers. Notionally, these may be thought of as omnidirectional APs.

**Methodology.** Our experiments run in channels one (2.412 MHz) and six (2.437 MHz) of the 2.4 GHz ISM band. Using a WiSpy<sup>5</sup> dBx spectrum analyzer to monitor the entire 2.4 GHz spectrum, we measured the noise floor of the network at  $-94$  dBm throughout our experiments. We also verified

---

<sup>5</sup><http://metageek.net>

the presence of light background traffic from one other network being received at an average  $-92$  dBm (measured at the middle of the testbed in Figure 3.8) on channel six and the occasional presence of background traffic on channel one.

Senders in our throughput experiments send greedy unidirectional flows of 1500-byte UDP packets. For the auto bitrate experiments, we enable bitrate selection at senders using the madwifi implementation of the SampleRate algorithm. Channel measurements between AP and clients take roughly 2 s and then experiments proceed by measuring the throughput of each of the patterns evaluated over 30 seconds.

Unless otherwise stated below, when we compare different antenna patterns, we normalize total antenna gain, running SamplePhase and using its total radiated power as a reference power level, and scaling the Silencer, directional, and omnidirectional patterns to emit the same total power. We label the latter two “Scaled Highgain” and “Scaled Omni,” respectively. To put our performance into perspective, we also compare against omnidirectional (“Omni”) and stock high gain (“Highgain”) patterns with the highest antenna gain configurable by the user: 2.15 dBi for omnidirectional<sup>6</sup>, and 15 dBi with a  $43^\circ$  beam width for high gain. Unless otherwise stated, senders and interferers in the experiments below transmit at full power (18 dBm).

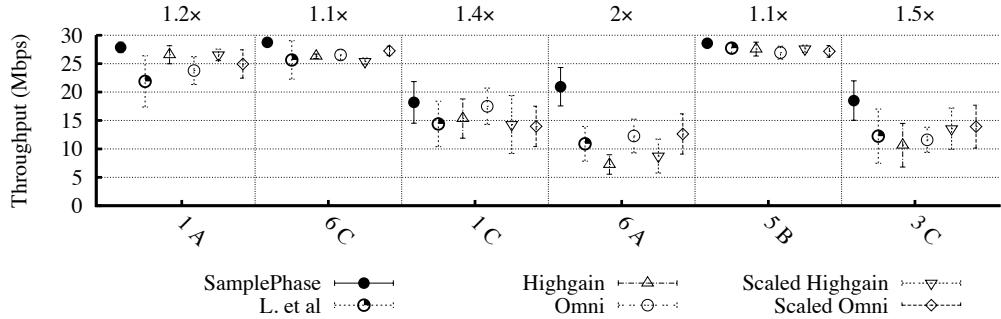
Table 3.2 gives a roadmap for the key experiments we present in this section, and the performance gains they achieve.

---

<sup>6</sup>The vendor does not provide a figure for gain relative to an isotropic antenna. The figure above is based on the assumption that the antenna in omnidirectional mode acts as a half-wavelength dipole.

<b>Experiment</b>	<b>Conclusion or performance improvement</b>	<b>Discussed in</b>
SamplePhase throughput	Over many links sending one at a time in our testbed, SamplePhase improves throughput over an omnidirectional pattern by 1.5–89%, improves throughput over the best directional pattern by 6.4–219%, and improves throughput over the Lakshmanan <i>et al.</i> method by 4.3–124%.	§3.2.1
Silencer throughput	Over many testbed links with an interferer placed near the receiver of each, CoS improves throughput over an omnidirectional pattern by 40–1013% and improves throughput over the best stock directional pattern by 31–222%. Silencer can also null traffic on adjacent WiFi channels.	§3.2.2
Longevity of patterns	In a busy indoor office environment, CoS patterns work effectively for on the order of 10 hours during quiet times and two hours during busy times.	§3.2.3
Nulling many interferers	Over many testbed links, Silencer can null two simultaneous interferers (a total of three concurrent transmissions), achieving throughput gains of $3.1 \times$ over the best omnidirectional pattern with the same interference. Nulling distinct interferers yields additive throughput gains.	§3.2.4

**Table 3.2:** Summary of experimental results for the proposed techniques, and the corresponding conclusion or performance improvement.



**Figure 3.9:** Empirically measured throughput for six different antenna gain patterns across six different testbed links (labeled by sender identifier and access point identifier: *cf.* Figure 3.8 on p. 96). In the absence of interference, SamplePhase offers the greatest throughput because of the accuracy of its channel measurement method.

### 3.2.1 Beamforming with SamplePhase

We first examine how well SamplePhase improves throughput at the receiver compared to the Lakshmanan *et al.* method and simple omnidirectional patterns. We also determine whether SamplePhase’s measurements of the channel yield any throughput improvement as compared with the highest-throughput pattern among the manufacturer’s fixed high-gain patterns.

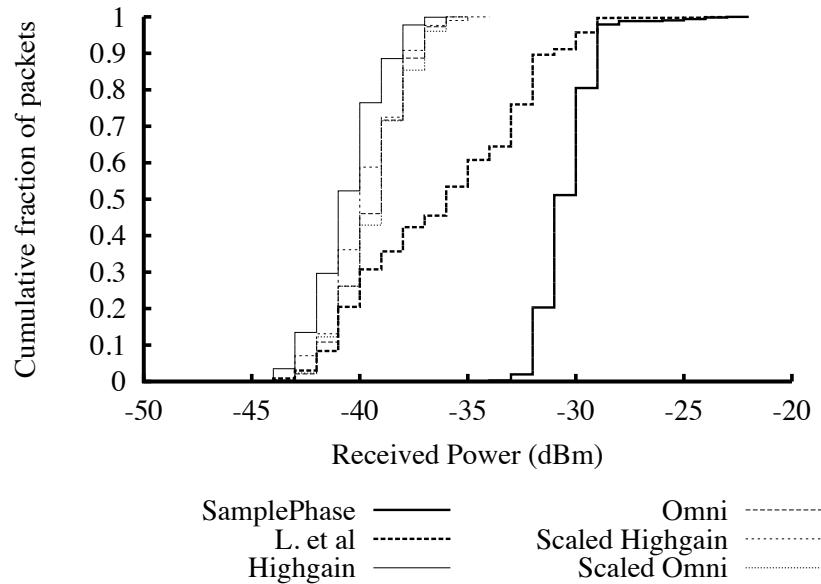
We determine which high-gain pattern among the 16 possible such patterns offers the greatest throughput by empirical measurement. To do so, we interleave two iterations through the 16 directional patterns, each spaced equally around the  $360^\circ$  circle at  $22.5^\circ$  increments. We test each pattern for 30 seconds, completing the experiment in 16 minutes. We pick the stock high gain pattern that yields the best overall average throughput, and compare SamplePhase against this pattern, both at full array power (27 dBm) and scaled on a per-link basis to use the same total power as the SamplePhase pattern.

In a topology with AP  $A$  receiving and a single sender  $S$  sending, we compare receive throughputs at the AP using a SamplePhase-derived pattern steered toward  $S$ , the best high-gain pattern for  $S$ , the scaled best high-gain pattern for  $S$ , an omni pattern, and a scaled omni pattern.

For many different two-node  $(S, A)$  topologies, we measure receive throughput at  $A$  for the above patterns. Each measurement we report in Figure 3.9 is the mean of 10 one-minute measurements with error bars representing 95% confidence intervals. We re-run the optimization at the start of each one-minute measurement interval.

Figure 3.9 presents the main SamplePhase throughput result, in which we compare patterns generated by the SamplePhase and the Lakshmanan *et al.* measurement methods, described in Subsection 3.1.3, with the high gain and omnidirectional antenna patterns described above. From the figure, we see that in the absence of interference, SamplePhase offers the greatest throughput.

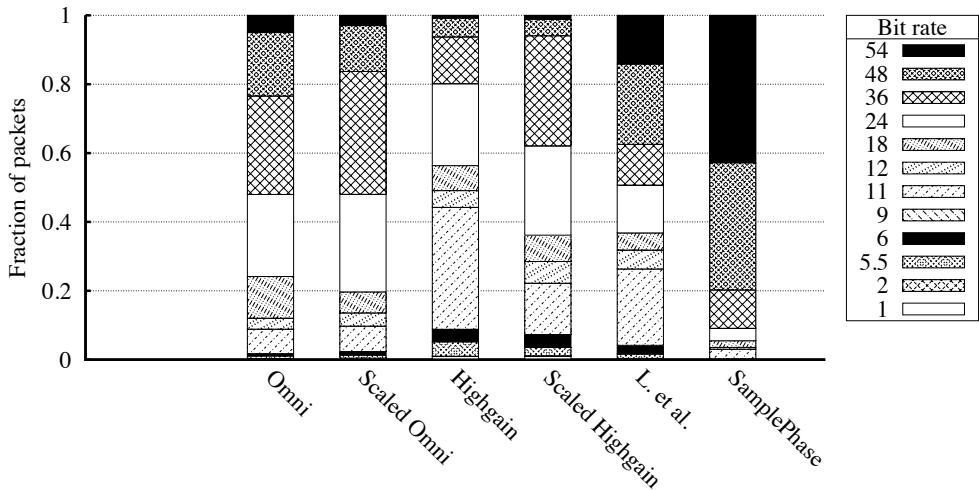
Two factors explain this throughput improvement. The first is that SamplePhase derives patterns that maximize RSS better than other methods. In Subsection 3.1.3, we present micro-benchmarks that show that SamplePhase derives patterns that maximize element-pairwise RSS better than competing methods. In order to show that SamplePhase's patterns improve RSS over other antenna patterns, we ran a fixed bitrate micro-benchmark over a link with high packet delivery rate at that bitrate. This micro-benchmark eliminates a sampling bias favoring strongly received packets that we would introduce if we examined RSS readings from the auto bitrate experiments in this section. In Figure 3.10 we see that these pairwise gains in RSS translate into improved RSS for the pattern as a whole, which



**Figure 3.10:** RSS distributions of packets drawn from fixed bitrate (54 Mbit/s) experiments on testbed link 2A.

increases the SNR of received packets, making successful decoding more likely.

The second phenomenon that explains the throughput improvement in Figure 3.9 is that as a result of incurring fewer bit errors, SampleRate, the bitrate adaptation scheme, chooses to use higher rates when using SamplePhase-derived patterns. For a representative link in our testbed, we examined SampleRate’s data structures over a 30-second representative period in the middle of our throughput experiment. Figure 3.11 shows the fraction of packets that SampleRate chooses to send at each bit rate over this link. From Figure 3.9, we see that with the SamplePhase pattern, SampleRate chooses the top bitrate (54 Mbps) for slightly more than 42% of all packets it sends and either of the top two highest (54 and 48 Mbps) for a total 80% of all packets sent. None of the other measurement methods chooses the top two bitrates for more than 40% of all packets sent. Omnidirec-



**Figure 3.11:** Bitrates chosen by SampleRate during a representative 30-second interval at link 6A (cf. Figure 3.9). SamplePhase chooses significantly higher bit rates than other methods, choosing the top two bitrates (48 and 54 Mbps) 80% of the time.

tional patterns use 24 Mbps and lower bitrates for half of all packets. This means that the performance of SamplePhase is mostly limited by the maximum available rate and if higher bitrates were available, the performance improvement could be greater.

### 3.2.2 Nulling Interferers with Silencer

802.11 networks operate at relatively high SNRs, so the cause of poor performance is often interference. In this experiment, we test how well Silencer nulls a single interferer, an expected common case in a light to moderately loaded network. In a topology with AP *A* receiving, a sender *S* sending, and an interferer *I* interfering, we measure the improvement in receive throughput at the AP using a Silencer pattern and compare it to that of a SamplePhase-derived pattern steered toward *S*, the best high-gain pattern for *S*, the scaled best high-gain pattern for *S*, an omni pattern, and a scaled omni pattern.

For several three-node ( $S, I, A$ ) topologies, we measure receive throughput at  $A$  for the above patterns. Each measurement we report in Figure 3.12 is the mean of 10 one-minute measurements with error bars representing 95% confidence intervals. We re-run the optimization for each one-minute measurement interval. The limited physical extent of our testbed makes it difficult to place sender-interferer pairs such that neither senses the other's carrier. Therefore, we use the `TX_STOMP` register of the Atheros chipset to run experiments with carrier sense turned off at both  $S$  and  $I$ . Doing so yields more flexibility to try different power levels (shown in Table 3.3) at the sender and interferer to more broadly explore how Silencer performs at the AP. Because the path between sender and interferer is distinct and independent of the sender-AP and interferer-AP paths, it is reasonable to turn off carrier sense in order to emulate topologies in which the sender and interferer are mutually hidden terminals. Note that Silencer can also be beneficial in cases where the two transmitting hosts are within one another's range. In such scenarios, using Silencer could potentially allow APs to mitigate interference to such a degree that two concurrently transmitting hosts would achieve higher aggregate throughput compared to using CSMA/CA, but we do not examine this scenario here.

In all experiments, the interferer sends broadcast packets at 54 Mbps. We can see that using Silencer outperforms every other strategy, yielding a mean throughput of 11.5 Mbps across topologies. This is an improvement of between 1.7 and 17.5 times over using a standard omnidirectional pattern. For the same topologies, using SamplePhase is the second best option resulting in 9 Mbps on average. On many occasions using a high-gain stock pattern performs similarly to SamplePhase (for instance in topology "7C1") and it generally outperforms using the omnidirectional pattern. For topol-

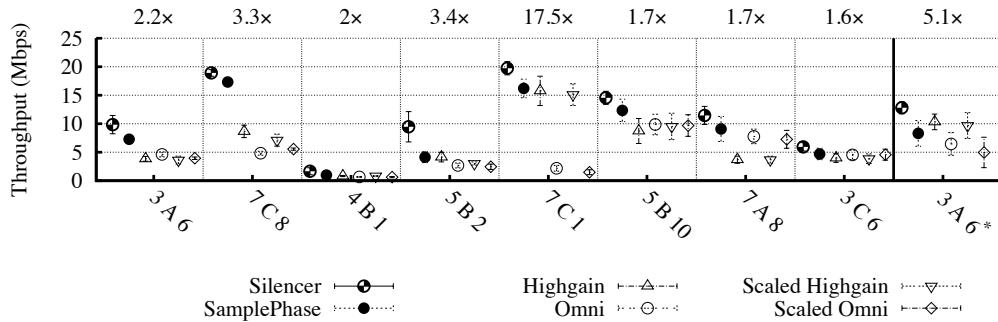
Topology	Sender (dBm)	Interferer (dBm)
3 A 6	18	18
7 C 8	18	18
4 B 1	12	18
5 B 2	12	18
7 C 1	12	18
5 B 10	12	6
7 A 8	18	6
3 C 6	18	18
3 A 6*	12	18

**Table 3.3:** Single-interferer experiments: power levels used at sender and interferer.

ogy “7A8”, though, this is not the case. This may happen if the side lobes of the high-gain pattern are steered in the direction of interference.

On the final link (labeled “3A6\*”) in Figure 3.12, we test the ability of Silencer to null interference on an *adjacent* channel. For these data points, we tune the interferer’s radio to WiFi channel 7, leaving the sender on channel 6. We allow SamplePhase to send measurement probes on channel 7 to the interferer, and on channel 6 to the sender. We then offer the resulting patterns as inputs to Silencer in the usual way. Once Silencer has generated a pattern that nulls the interferer, we tune the AP to channel 6. We note that Silencer still effectively nulls interference and increases the throughput of the link, despite combining patterns generated by SamplePhase on adjacent channels.

We performed all of our experiments with a unidirectional UDP workload. Using UDP allowed us to analyze changes in throughput without confounding effects introduced by congestion control. Congestion-controlled TCP is more loss-averse, since a single packet loss leads to halving of the sender’s congestion window. As such, we expect TCP would benefit even more from the use of Silencer.



**Figure 3.12:** Empirically measured throughput in the presence of one interferer, for six different antenna gain patterns across six different testbed links (labeled by sender identifier, access point identifier and interferer identifier: cf. Figure 3.8 on p. 96). By nulling, Silencer yields the best end-to-end throughput in the presence of interference.

**Penalty associated with nulling interferers.** Because CoS *always* nulls interferers of which it is aware, but interferers do not send during every packet time, there may be an opportunity cost associated with “needless nulling” (of an interferer not sending). That is, nulling an interferer may collaterally also reduce the signal strength from the sender of interest. To evaluate whether such an effect noticeably reduces throughput, we performed an experiment in which we compared the throughput achieved by a pattern derived with SamplePhase with that of a pattern derived using Silencer. In the latter case, we used the same sender as in the former, with the addition of an interferer in order to calculate the Silencer-derived pattern. We then measured the throughput achieved by the same sender in the absence of interference when receiving using these two patterns. The result was a 0.1% reduction in throughput for Silencer.

Although this result suggests that coarse-grained nulling does not achieve significantly lower throughput compared to nulling, in Chapter 4 we will see that when we can null or beamform individually on each subcarrier, this cost becomes substantial.

CoS APs have eight-element antennas, hence eight degrees of freedom, and nulling a single host consumes only one of those degrees. If the number of concurrently nulled hosts increases to three or four—thus consuming half of the available degrees of freedom—we expect that the cost of nulling will start to become considerable.<sup>7</sup> The pervasive use of 802.11 networks means that there could be numerous proximate networks operating in the same frequency band.

Moreover, many clients are devices such as sensors, which transmit only intermittently. Under such conditions, it is hard for an AP to select a fixed set of devices to null permanently. Yet, always nulling as many hosts as possible—seven in the case of the Phocus Array—may hurt performance considerably. Such use cases would further require a MAC-layer protocol that would limit the number of concurrently transmitting hosts and enable their identification in advance.

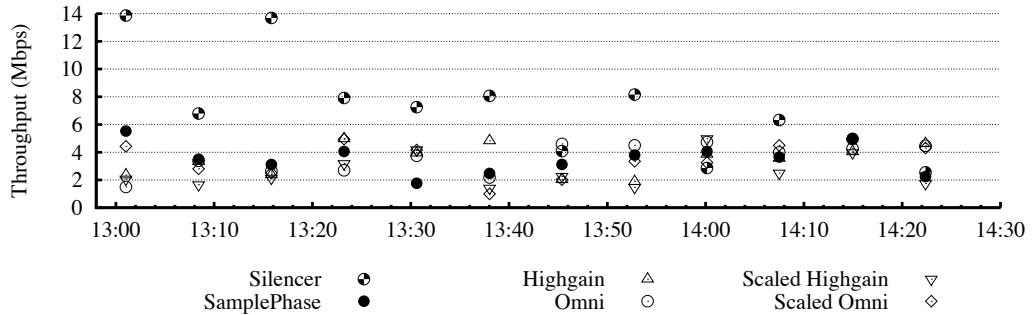
### 3.2.3 Longevity of Interference-Nulling Patterns

In this experiment, we ask the following question: How long can we reasonably expect to be able to use a throughput-maximizing pattern before changes in the channel cause the pattern’s performance to degrade significantly? In other words, how do received power and throughput evolve over time when a CoS-enabled AP continues to receive after deriving beam patterns using SamplePhase and Silencer?

For *one link* on which Silencer provides a throughput gain over omni and high gain patterns, we receive using both the Silencer-derived and high gain directional patterns. In Figure 3.13 we plot a time series of the thirty-second moving average of throughput. We interleave throughput measure-

---

<sup>7</sup>See Chapter 5 for a more in-depth discussion of this phenomenon.

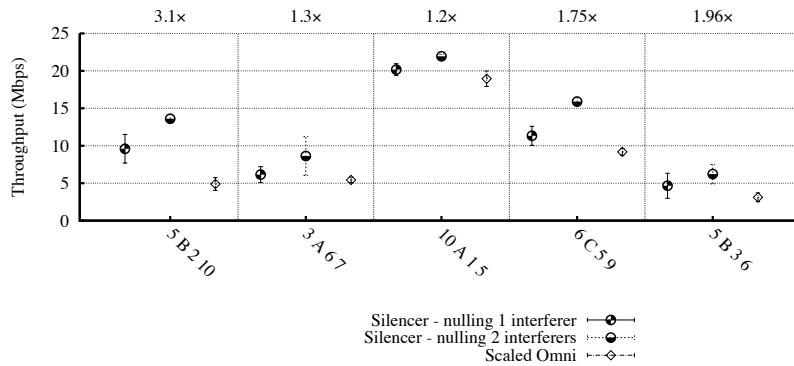


**Figure 3.13:** Longevity of a Silencer interference-nulling pattern. We run the SamplePhase measurement method only once, at the beginning of this time series. Then, as the wireless channel changes, we measure throughput in the presence of a continuous interferer. We see that the Silencer pattern’s throughput does not degrade significantly for approximately 45 minutes.

ments of Silencer, a SamplePhase pattern that does not attempt to null, the high gain directional pattern that achieves the highest throughput on the link, the omnidirectional pattern and power-normalized versions of the last two. Our experiment took place during a weekday afternoon, between 13:00 and 14:30, with normal activity for the environment, e.g., people moving around, and our hosts were static. We can see that in this experiment, the pattern derived by Silencer outperforms all the others consistently for about 45 minutes, after which its performance degrades roughly to theirs. This should be true for all static scenarios during periods where the reflectors in the environment do not change.

### 3.2.4 Nulling Multiple Interferers

Since multiple simultaneous interferers are common in densely deployed wireless networks, Silencer’s performance when more than one interferer sends concurrently is important. We address this question experimentally.



**Figure 3.14:** Empirically measured throughput in the presence of two concurrent interferers (for a total of three concurrent transmissions), over multiple testbed links (labeled by sender identifier, access point identifier and both interferers' identifiers: *cf.* Figure 3.8 on p. 96). Adding channel measurements for each additional interferer and incorporating that interferer into Silencer's nulling algorithm yields cumulative additional benefit.

For five links in our testbed, we set up two interferers and a sender, for a total of three senders, each of which transmits packets simultaneously, as fast as possible, with carrier sense disabled as in the single-interferer experiments in Subsection 3.2.2. Over all links evaluated in this experiment, the sender transmits at 18 dBm power and the interferers transmit at 12 dBm. We compare the throughputs of a CoS-enabled AP in omnidirectional mode, Silencer nulling one of the interferers, and Silencer nulling both interferers. The results of this experiment appear in Figure 3.14. We see that Silencer nulling one interferer while not taking into account the other offers substantial throughput gains over an omnidirectional pattern, doubling throughput on some links. Furthermore, when Silencer measures the channel to both interferers and nulls both, it achieves additional substantial throughput gains over Silencer nulling just one interferer.

### 3.3 Discussion

The broad adoption of 802.11 wireless networks forces an ever-increasing population of co-located users to share finite unlicensed spectrum. Much of this sharing occurs among distinct networks deployed in uncoordinated fashion in close proximity. Capacity-enhancing proposals for enterprise wireless networks, in which a large organization wishes to increase capacity in a network of many centrally controlled APs, have profited greatly from multi-antenna techniques in recent years, but assume cooperation among APs.

In this chapter, we have presented Cone of Silence (CoS), a system that leverages multiple antennas to increase receive throughput at an AP in the presence of interference from uncooperative nearby senders. CoS adopts long-known techniques for multi-antenna systems, maximum ratio combining (MRC) and the decorrelator, but applies them in the novel context of an 802.11 receiver with a single, commodity radio. CoS’s decorrelator, Silencer, nulls multiple interferers without decoding their signals, while maximizing signal strength from a sender of interest. Our experimental prototype CoS 802.11b/g access point equipped with an 8-element phased array antenna demonstrates a throughput improvement under interference of between  $1.6\times$  and  $17.5\times$  over that achievable with an omnidirectional antenna, and achieves consistently higher throughput than beamforming toward the sender alone. SamplePhase and Silencer achieve throughput gains by allowing automatic bitrate adaptation on clients to choose higher bitrates for more packets; they increase SINR at the AP, improving the reliability of higher bitrates.

This work demonstrates a clear throughput improvement when using CoS, both in the absence of interference and when hidden terminals interfere. As discussed in Subsection 3.2.2, CoS could potentially improve throughput even in topologies where the two transmitting hosts are within range of one another. In such a case though, it would be necessary to turn off CSMA/CA and instruct the hosts to transmit concurrently. In the next chapter we will see a system that allows neighboring hosts to transmit concurrently and do so in a distributed fashion.

Newer wireless devices allow hosts to manipulate the wireless signal differently for each subcarrier. Doing so allows for much more precise solutions when using MIMO techniques such as beamforming and nulling. In the next chapter we will see how we can take advantage of such functionality while building a transmit architecture suitable for dense wireless deployments.

## **Chapter 4**

# **Cooperative Power Allocation (COPA)**

Deployments of Wi-Fi wireless LANs in homes and offices have proliferated so widely that it is now commonplace for several such networks to operate in close proximity. These dense, uncoordinated deployments often interfere significantly with one another. As we saw in Chapter 2, the research community has explored interference mitigation approaches that centrally control wireless senders' concurrent transmissions. One such approach centrally instructs separate access points (APs) run by the same organization to use their respective antennas to cancel the interference each AP might deliver to other APs' clients [9, 43, 30, 75]. Another such approach is distributed MIMO, which pools the antennas of APs run by the same organization into one large "virtual" AP that concurrently sends multiple streams of data to multiple clients [18]. These approaches show great promise in the enterprise setting where a single entity controls all APs and (in the case of distributed MIMO) connects them via a gigabit wired backplane.

Wireless LANs in separate homes and offices, however, are typically owned and administered by different parties, where there is no common controller that can orchestrate their transmissions centrally, nor any gigabit-speed wired backplane interconnecting their APs. How can one mitigate the interference uncoordinated, densely deployed APs cause to one another's clients, and thus improve aggregate throughput? One well-known tool for interference mitigation is *nulling*, where an AP uses multiple antennas to cause multiple instances of its transmitted signal to cancel one another at another AP's client. In this chapter, we explore *selfish cooperation*, where two APs run by different parties coordinate over the wireless medium to send concurrently while nulling toward one another's clients. The APs are selfish in that they may only decide to cooperate when neither suffers a reduction in throughput when they send concurrently. We demonstrate that while nulling can significantly reduce interference, the practical capacity improvement nulling alone achieves in realistic indoor environments is limited because nulling overlooks—and indeed, elevates the importance of—the complementary problem of power allocation.

We present *Cooperative Power Allocation* (COPA), a system in which two APs alert one another of the clients to which they intend to send. Each AP explicitly considers the relative strengths of both APs' transmissions at both clients, then nulls toward the other's client, cooperating with its counterpart to allocate power to each narrow sub-band, or *subcarrier*, within the Wi-Fi channel. The APs choose power allocations that yield a signal to interference plus noise ratio (*SINR*) at each client that is most conducive to high aggregate throughput. As we discuss in Section 4.1, subcarrier-granularity power allocation is important to throughput because frame reception fails when a Wi-Fi receiver cannot correctly decode the data on just

a few of the dozens of subcarriers that comprise the Wi-Fi channel. Thus, even when receivers hear some subcarriers with very high SINR, and others with low SINR, the sender has no choice but to send with a (lower bit-rate) modulation and code that protect the data on the subcarriers with poor SINR, sacrificing available wireless capacity.

Unfortunately, as we illustrate experimentally in Section 4.1.2, nulling exacerbates the variability in per-subcarrier SINR at receivers. COPA is thus an important complement to nulling: COPA APs cooperatively allocate transmit power so as to explicitly avoid low-SINR subcarriers at their receivers, and thus improve aggregate throughput. Furthermore, a cooperative system must sometimes offer an incentive for different, possibly selfish users to cooperate. COPA supports two modes of power allocation: one that aims to achieve the greatest aggregate throughput, possibly at the expense of one AP’s throughput; and another that aims to improve aggregate throughput subject to the constraint that no AP suffer reduced throughput. In Section 4.3, we experimentally explore the price COPA pays in aggregate capacity to achieve fairness.

Our main contributions in this chapter are to elucidate the real-world performance of nulling in the setting where interfering APs are not centrally controlled and do not share a gigabit backplane, and materially improve it. Our salient findings include:

- In 83% of topologies in an office environment in which two 4-antenna Wi-Fi APs send to two 2-antenna clients, nulling *underperforms* CSMA. On these topologies, in which the variability of SINR across subcarriers introduced by naïve nulling forces the APs to transmit at lower bit-rates, COPA improves nulling’s throughput by a mean of 64%, such

that the throughput of COPA’s approach to nulling *exceeds* CSMA’s in 76% of the same topologies.

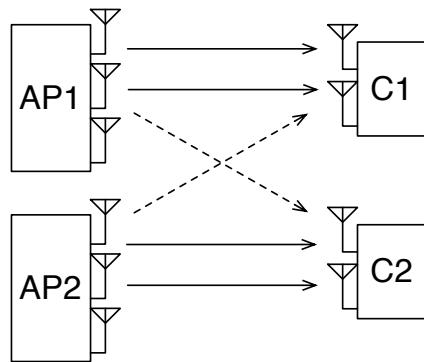
- In the remaining 17% of the same topologies in which naïve nulling outperforms CSMA in throughput, it does so by a median of 12%. On these topologies, in which cross-interference between each AP and the other AP’s client is relatively weak, but naïve nulling still introduces throughput-limiting variability of SINR across subcarriers, COPA improves nulling’s throughput improvement over CSMA to a median of 45%.

## 4.1 Problem

Consider the Wi-Fi deployment shown in Figure 4.1 on the facing page, where two independently operated Wi-Fi LANs are in proximity (*e.g.*, in adjacent offices or apartments). Because of the proximity of the two LANs, if AP1 transmits to C1 while AP2 transmits to C2 concurrently, each will likely interfere with the other’s transmission.

The throughput of AP1’s transmission to its intended receiver C1 is determined by the *signal-to-interference-plus-noise ratio* (SINR), where *signal* denotes the received power of AP1’s signal at C1, *interference* denotes the received power of AP2’s signal at C1, and *noise* denotes the *noise floor*, or the power level of background RF noise in the environment. Thus, from C1’s perspective, stronger received interference from AP2 or a weaker received signal from AP1 reduces SINR at C1, thus reducing throughput.

In a scenario like this, prior work focuses on two major techniques to mitigate throughput reduction: *carrier sense* and *nulling*. Carrier sense (CS) attempts to eliminate interference by avoiding concurrent transmission: a



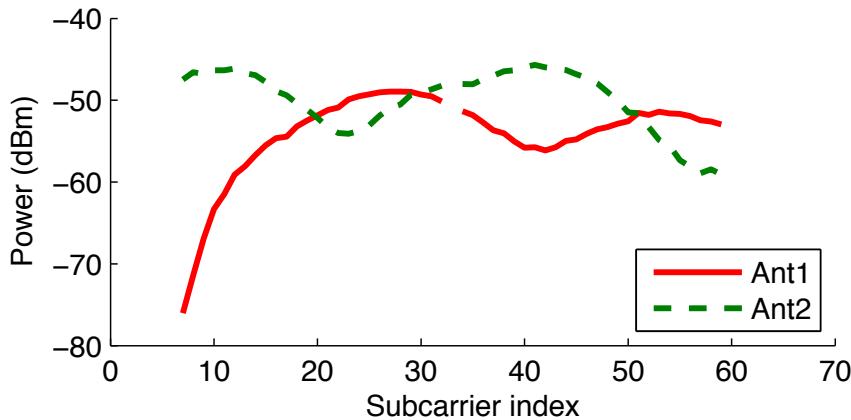
**Figure 4.1:** AP1 and AP2 transmit concurrently to clients C1 and C2, respectively; each AP interferes with the other AP’s transmission to that other AP’s client.

sender defers if it hears another sender transmitting. CS results in sequential transmissions, *e.g.*, in two greedy senders’ taking turns transmitting on average. Given collisions and medium acquisition overheads, the aggregate throughput of CS is bounded above by that of a perfectly scheduled Time Division Multiple Access (TDMA) scheme. Nulling, on the other hand, entails concurrent transmissions by the two senders, and attempts to cancel out interference at the unintended receiver. A sender with multiple antennas can cancel its own transmission by sending one instance of its signal from one antenna and a phase-shifted instance of the same signal from a second antenna. If it chooses the phase shift so that the two signals arrive perfectly out-of-phase at the antenna of the unintended receiver, they will sum to zero.

But CS and nulling share a common deficiency: Wi-Fi senders using these techniques today allocate power equally across the 20 MHz or 40 MHz frequency band of a transmission,<sup>1</sup> and they do so in a fashion oblivious to the detailed interference effects each sender causes to the other’s receiver.

---

<sup>1</sup>Platform limitations confine our results to 20 MHz.



**Figure 4.2:** Received power from a single send antenna at two different receive antennas, by OFDM subcarrier index.

To see why this matters, let us examine in greater detail how a 20 MHz Wi-Fi channel behaves in practice.

#### 4.1.1 OFDM and Narrow-Band Fading

CS and TDMA both assume that concurrent transmissions by two APs will decrease SINR sufficiently that neither client can decode correctly. However, modern Wi-Fi senders first redundantly code outbound frames, then break the wideband wireless channel into some number of *subcarriers*, sending distinct “slices” of the redundantly-coded frame’s bits on each subcarrier—a scheme known as Orthogonal Frequency-Division Multiplexing (OFDM). The SINR of each individual OFDM subcarrier determines the bit-error rate with which a receiver can decode information sent on that particular subcarrier. To limit hardware complexity, the Wi-Fi standard constrains a sender to use the same OFDM modulation on all subcarriers [2].

Indoors, it is expected that many OFDM subcarriers will suffer from *narrow-band fading* because of multipath propagation effects. Some sub-

carrier frequencies are affected more than others by reflection, scattering, and shadowing [10]. In addition, the fading pattern often drastically differs at different locations separated by a distance of just 12.5 cm (one radio wavelength), as Figure 4.2 from our testbed shows. The sender here allocates equal power to each subcarrier, resulting in the received power shown in the figure.

There is a power budget for the entire 20 MHz channel that state-of-the-art Wi-Fi OFDM senders divide equally between subcarriers, but other allocations are possible. We observe that in the presence of narrow-band fading, the ability to unequally allocate power across subcarriers creates the opportunity for two APs to send concurrently to their respective clients. For example, suppose AP1’s transmission on some subcarrier propagates strongly to its client C1, but much more weakly to AP2’s client C2. If AP1 and AP2 transmit concurrently, AP1’s transmission may cause an insignificant SINR drop on that subcarrier at C2. Even if AP2’s transmission on that subcarrier causes a throughput-reducing SINR drop at C1, AP1 could potentially alleviate this by increasing the power it sends on that subcarrier.

To take advantage of such opportunities, the two APs would need to make power allocation decisions cooperatively: if both APs knew the per-subcarrier signal strength of each AP at both clients, they could maximize aggregate throughput by choosing how much power to transmit on each subcarrier. Today’s Wi-Fi networks and OFDMA system design proposals [59, 60] ignore other senders when they allocate power; instead we propose *cooperative power allocation*. This approach is possible even for single-antenna APs, as we will describe in Section 4.3.2.

Prior approaches to power allocation do not fit the scenario of cooperating Wi-Fi APs. The technique of *waterfilling* maximizes achieved link capacity for idealized radios that transmit Gaussian signals [10], but performs poorly for practical radios like those used in Wi-Fi, which transmit discrete constellations [63]. The related technique of *mercury/waterfilling* [63] optimally distributes power among subcarriers for discrete constellations. Mercury-waterfilling takes as a given that all subcarriers of a given set will be used, and minimizes the bit-error rate of a transmission, without considering eschewing weakly received subcarriers outright. Nor does it address power allocation under dynamically changing interference, such as results when two concurrently sending APs perform power allocation while sending toward their own clients and nulling toward each others' clients.

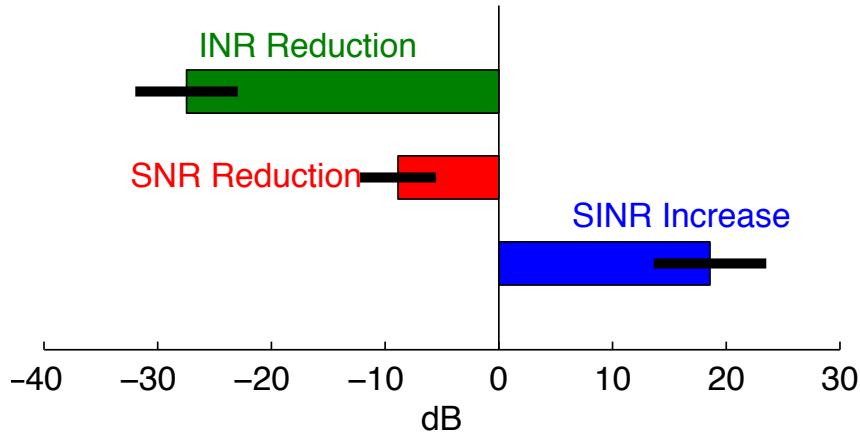
#### 4.1.2 Nulling in Practice: Residual Interference

In principle, when one AP nulls toward another's client, the nulling should eliminate *all* interference from the first AP at the client's antenna. In practice, however, there is residual interference left over after nulling. Furthermore, senders null on a per-subcarrier basis, and so efficacy may vary significantly from subcarrier to subcarrier, even though averaged across subcarriers, nulling reduces interference well.

**Nulling viewed on average.** To evaluate the efficacy of nulling we transmit concurrently from two four-antenna APs to two two-antenna clients in an indoor office environment. Each AP sends two MIMO streams to its own client and nulls toward each of the two antennas of the other client.<sup>2</sup>

---

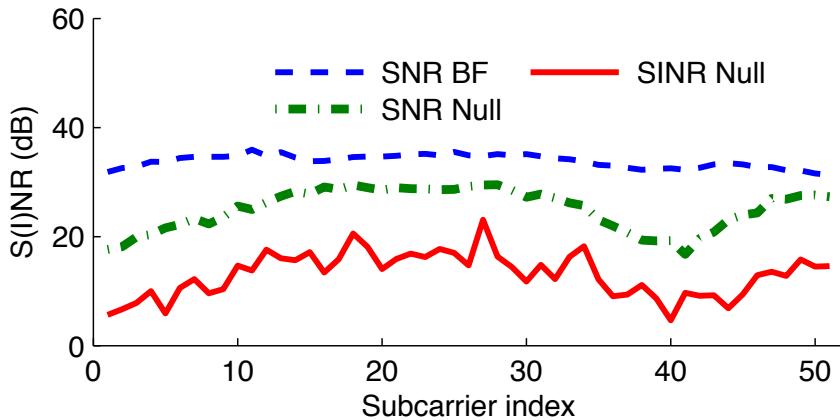
<sup>2</sup>Our APs and clients are Rice WARP v2s communicating over a 20 MHz channel in the 2.4 GHz band, using OFDM and Wi-Fi's 802.11n high-throughput bit-rates; full details of our experimental setup are in Section 4.3.



**Figure 4.3:** End-to-end effect of nulling on SINR, SNR and INR; 30 indoor office topologies; two four-antenna APs sending to two 2-antenna clients. Error bars denote one standard deviation.

How much does nulling improve SINR at C1 and C2? In our testbed, depicted in Figure 4.12 on page 143, we take measurements at C1 and C2 in 30 different indoor office topologies, each consisting of different placements of two four-antenna APs and two clients. Our results in Figure 4.3 show a reduction in interference (“INR reduction” in the figure) averaging 27 dB. Although nulling reduces interference significantly, the reduction does not generally exceed 30 dB. Furthermore, nulling may also reduce the signal an AP delivers to its own client. We term this “collateral damage,” as two signals intended to cancel at the other AP’s client may also partially cancel at the intended receiver. Experimentally, we see that the cost of nulling the signal of interest (“SNR reduction” in the figure) averages  $-8$  dB, offsetting the reduction in interference for a net 18 dB SINR improvement on average, with SINR improvement generally no better than 23 dB.

**Nulling viewed per subcarrier.** We now examine the effect of nulling on individual subcarriers using client C1 in one of our testbed topologies as an illustrative example. The “SNR BF” curve in Figure 4.4 shows the SNR of



**Figure 4.4:** Per-subcarrier effects of nulling; two four-antenna APs sending to two two-antenna clients.

each of AP1's subcarriers at C1 when only AP1 sends—this is the baseline, when AP1 has complete freedom to adjust the phase of its transmissions, thus beamforming towards C1. The “SNR Null” curve shows the result of AP1 nulling towards C2 while sending to C1. Not only has mean power decreased, but the SNR is more variable. The primary reason is that in order to null, AP1 can no longer fully align the phase of its transmissions as received at C1. The effect resembles that of narrowband fading: we see an increased SNR variance across subcarriers. Lastly, the “SINR Null” curve shows the SINR C1 experiences when AP1 and AP2 send concurrently, but null towards each other's clients. In addition to the effect of imperfectly aligned phase for “SNR Null,” incomplete nulling has further reduced the mean SINR and introduced further variance across subcarriers. Several noise sources conspire to cause imperfect nulling, including receiver noise when measuring the channel state in order to calculate the nulling phase, and transmitter imperfections and noise when sending the nulled signal.

The consequence of an increase in SINR variance across subcarriers is often reduced throughput. Some prior work, such as IAC [9] and Aryafar *et al.* [8], performs nulling using only one subcarrier, so this variability is not observed. Wi-Fi uses 52 subcarriers in every packet transmission [2], and employs only a single modulation and convolutional code across all of them. When the SINR varies widely across subcarriers, a few low-SINR subcarriers can cause the decoding of an *entire packet* to fail, and thus cause the sender to reduce bit-rate. As Wi-Fi does not cooperatively allocate power to subcarriers by taking SINR at all receivers into account, it cannot reallocate power from good subcarriers to weak ones to “save” them from having catastrophically low received SINR.

## 4.2 Design

In this section, we describe the design of COPA, starting with a description of how COPA APs learn about each other’s transmissions (Subsection 4.2.1), followed by a description of how each COPA AP chooses the amount of power to allocate to each OFDM subcarrier (Subsection 4.2.2). A description of the system’s overall design, including how both the above design elements integrate with transmit nulling concludes the section (Subsection 4.2.3).

### 4.2.1 Coordination Protocol

Since COPA APs don’t necessarily belong to the same administrative domain, nor are even connected by a high speed LAN, they must coordinate over the air in order to:

1. Inform each other about opportunities for concurrency,

2. Disseminate information about the channels (*channel state information*, or *CSI*) between APs and clients, and
3. Exchange information about the choices each AP makes.

COPA strives to accomplish these goals with a minimum of protocol overhead. The basic mechanism we choose is the use of *control* messages transmitted using an omnidirectional spatial profile. We term our control message the *Intention-to-Send* or *ITS*.

## Learning CSI

When a COPA AP overhears frames from nearby clients or other APs, it measures the channel from those senders, and caches the resulting CSI in a table indexed by sender address. Since the wireless channel is reciprocal between sender and receiver,<sup>3</sup> COPA APs learn CSI information to nearby clients by overhearing their recent transmissions, as shown in Step ① of Figure 4.5.

How recent must clients' transmissions be in order to ensure that COPA APs use accurate CSI information? CSI does not need to be refreshed at the start of every 4 ms 802.11 *transmit opportunity* (the time granularity of medium acquisition), but instead once every *coherence time*, the amount of time for which the wireless channel remains mostly constant, given the speed of motion of nearby mobile clients, objects, and people. The coherence time is given by  $t_c = \frac{m \cdot \lambda}{v}$ , where  $\lambda$  is the wavelength of the carrier frequency,  $v$  is the speed of the host, and  $m$  is a parameter that characterizes the physical environment [10]. A conservative value for  $m$  is 0.25 [10, 76], which results in coherence times of 28 ms for a speed of 4 km/h and 112 ms

---

<sup>3</sup>Except for radio front-end differences, which modern Wi-Fi radios calibrate away before operation [2].

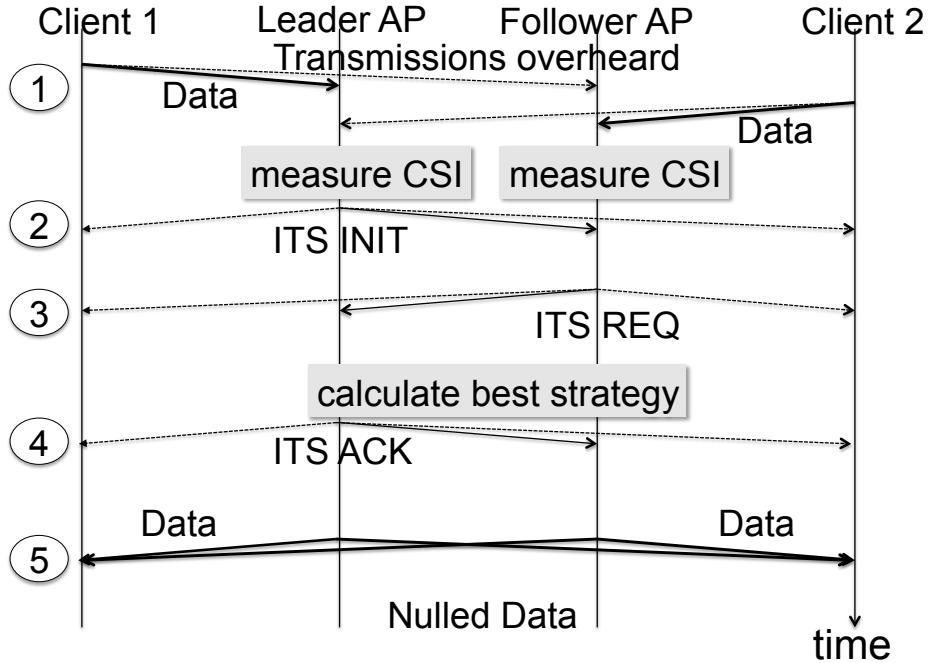
for a speed of 1 km/h. In our experimental evaluation (§4.3), we measure CSI once every 30 ms—sufficiently often for COPA to work in an environment with a coherence time of that short a duration.

COPA compresses CSI information using adaptive delta modulation across subcarriers' amplitude and phase (separately), and compressing the result using a variant of the Lempel-Ziv lossless data compression algorithm. This yields a compression ratio of two on average for the channels in our testbed (Section 4.3).

Finally, to avoid interference between different OFDM carriers, concurrent transmissions need to be synchronized in time within a cyclic prefix (800 ns) [9]. This is possible in today's Wi-Fi medium access control protocol by senders timing their transmissions off the end of ongoing transmissions [77], and COPA leverages the same mechanism in its medium access control protocol, which we now describe.

## The ITS Exchange

When traffic from the wired backhaul arrives at the COPA AP for downstream transmission to a client, the AP first checks whether there are any ongoing COPA transmissions. In doing so the AP obeys the status quo Wi-Fi carrier sense deference rule of waiting until ongoing transmissions complete and then competing for the medium by means of a bounded exponential backoff [2]. Once the medium becomes clear, all APs ready to send traffic to clients then contend to send an ITS INIT control frame as shown in Step ② of Figure 4.5. ITS INIT expresses an AP's intent to send to a specified receiver, and the AP that wins the contention is elected *Leader AP*. The ITS INIT contains both the Leader AP's identity and the identity



**Figure 4.5:** A timeline of COPA MAC operation. ITS REQ frames include CSI from Follower AP to both clients, while ITS ACK frames include the precoding matrix for the Follower AP.

of the client to which the leader AP is about to send (Client 1 in our example).

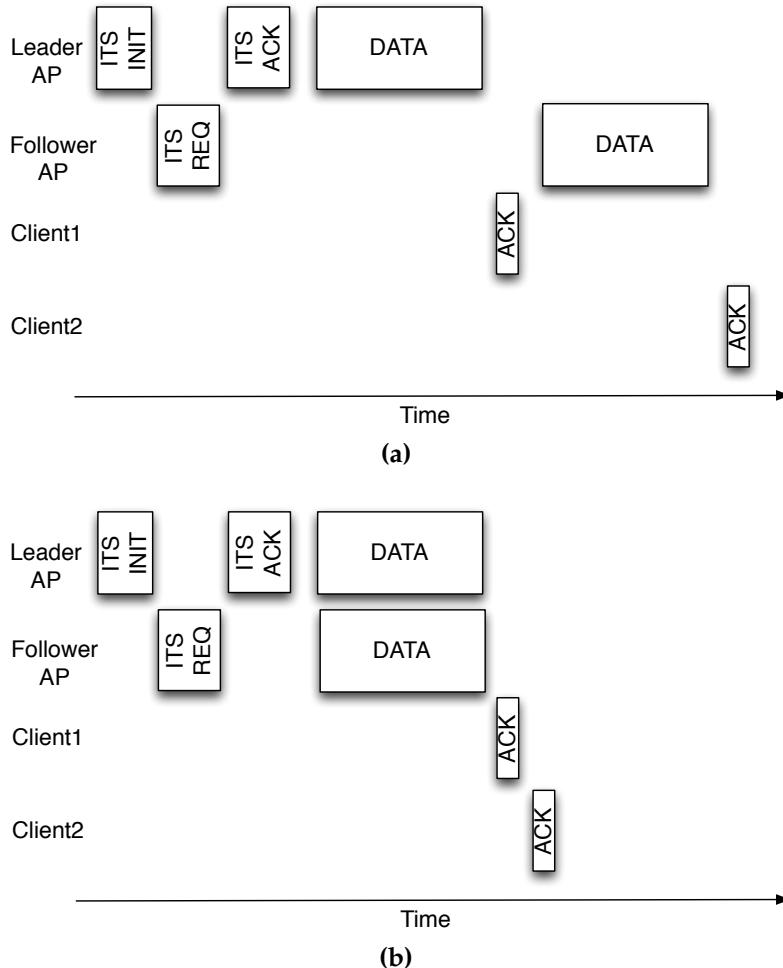
After the ITS INIT frame, any other APs that have traffic to send to their clients then cancel the transmission of their ITS INIT frames and contend to send a frame that indicates that the AP (which we term a *Follower*) requests participation in the next transmission opportunity with the Leader. This frame is called the ITS request (ITS REQ) frame and is shown in Step ③ of our example. The ITS REQ frame contains the identities of the Leader, Follower, and both clients, and the CSI from Follower to Client 1 and Client 2.

Once the Leader receives an ITS REQ frame from a Follower, it estimates the best joint strategy for both APs. This computation is described in the next two subsections (4.2.2, 4.2.3). Once it has made its choices, the Leader AP sends an ITS ACK (Step ④) as a broadcast, received by Follower AP and both clients. In the first case, if one of the two APs wins the initial contention, the other does not send an ITS REQ back for the rest of the coherence time. This means that the first AP has the opportunity to either engage in an ITS exchange with some third AP, or alternatively transmit on its own. On the other hand, if the two APs decided to transmit concurrently, the ITS ACK also includes the precoding matrix that the Follower should use for a concurrent transmission. Finally both APs transmit: concurrently if the calculation shows that to be the best strategy, or sequentially if no good concurrent solution is available.

After the two APs finish their transmission, each client acknowledges the correct reception of its data frame by sending a MAC-layer acknowledgement (ACK). As we can see in Figure 4.6a, if the two APs transmitted sequentially, each client transmits its ACK within a *Short InterFrame Space* (SIFS) (10  $\mu$ s) after receiving its intended data frame. If on the other hand the two APs transmitted concurrently, the two clients stagger their ACKs to avoid a collision, a scheme also proposed in IAC [9]. In this case, illustrated in Figure 4.6b, Client 1, which is the client of the Leader AP, sends its ACK first. Then Follower AP's client, Client 2, follows with its own acknowledgement.

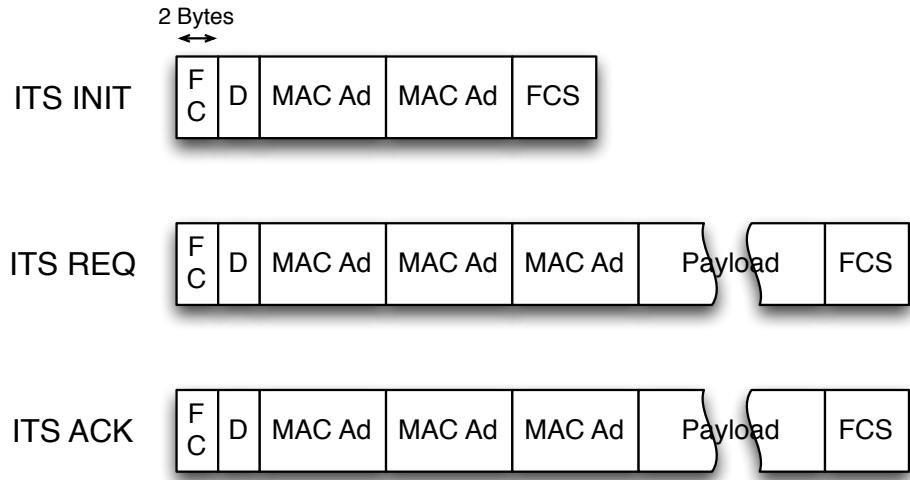
## The ITS Frame Structure and Overhead

Figure 4.7 shows the frame format for the different types of frames used in the ITS exchange. All frames contain the two-byte Frame Control field



**Figure 4.6:** MAC-layer ACKs in the case of sequential and concurrent transmission.

present in every 802.11 frame [2], which carries control information about the frame such as protocol version, frame type, etc., the two-byte Duration field and the four-byte Frame Check Sequence, which is used for error detection based on the CRC-32 algorithm. The ITS INIT frame additionally contains the MAC address of the leader AP and the MAC address of its intended receiver. The ITS REQ frame on the other hand contains the MAC addresses of the leader AP, the follower AP and the follower AP's



**Figure 4.7:** Formats of the different ITS frames. “FC” is the Frame Control Field, “D” is the Duration, “MAC Ad” Is a MAC address, “Payload” contains CSIs, and FCS is the Frame Check Sequence.

intended receiver, while finally the ITS ACK contains the MAC addresses of the leader and follower APs.

ITS REQ and ITS ACK also include a “Payload” field that contains channel state information. This CSI is two bytes for each of the 52 used subcarriers (one byte for the I component and one Q component) which means that each Tx-Rx antenna combination yields 104 bytes of information. If, for instance, we are dealing with 4-antenna transmitters and 2-antenna receivers, the total size of the CSI would be 1664 bytes. To reduce the payload size we first encode the amplitudes and phases for each subcarrier using differential encoding and then apply the lossless Lempel-Ziv-Markov chain compression algorithm. Doing so for the different topologies used in Section 4.3 yields a compression ratio of about 2.05 on average. If further savings are deemed necessary, it is possible to reduce the bits used per subcarrier from 16 to 12—the highest resolution used by the 802.11 standard [2]—to save another 25%. Another way to reduce the size of the payload is to send CSI

Coherence time (ms)	COPA Conc (%)	COPA Seq (%)	CSMA CTS (%)	CSMA RTS/CTS (%)
4	9.3	7.7	2.7	3.7
30	5.1	3.5	2.7	3.7
1000	4.5	2.8	2.7	3.7

**Table 4.1:** Throughput costs incurred by MAC overhead.

for just one out of two or four successive subcarriers and use interpolation to recover the CSI of the others. This method—which is also part of the 802.11 specification [2]—takes advantage of the fairly similar channel conditions faced by successive subcarriers. Of course, using a lossy compression method like subcarrier grouping or coarser quantization reduces the resolution of CSI, which will degrade nulling. In all our experiments we assumed transmitting the full set of CSI *i.e.*, 16 bits for each of the 52 subcarriers.

The ITS exchange adds overhead beyond that of CSMA. The magnitude of this overhead depends on how often COPA must disseminate CSI (which in turn depends on the coherence time of the environment), and on whether COPA decides on sequential or concurrent transmission. Table 4.1 analytically compares the throughput costs incurred by COPA’s ITS exchange in the sequential and concurrent transmission cases, *vs.* by CSMA’s CTS-to-self and CSMA’s RTS/CTS, for different coherence times. These values are for the scenario of two 4-antenna transmitter, 2-antenna receiver pairs, for the topologies that we test in Section 4.3. They take into account a mean medium acquisition time of 67.5  $\mu$ s, which assumes no collisions during the random backoff stage of Collision Avoidance, all inter-frame spaces such as SIFS and DIFS, preambles, ACKs, etc [2].

**Discussion: Other MAC Considerations.** Since COPA’s ITS exchange relies on probabilistic contention between senders, collisions are possible,

which result in garbled ITS frames. In these cases, senders follow the standard bounded exponential backoff and retry their transmissions. This also means that the hosts participating in a transmission cycle are determined randomly. Note that some host combinations may yield higher throughput than others (*e.g.*, hosts that cause less interference to one another). However, COPA uses an egalitarian approach to medium acquisition that gives all hosts the chance to contend for the next transmission cycle in a fashion agnostic to the throughput achievable by hosts, as is the case in “stock” CSMA/CA.

In the experimental evaluation in this chapter, we limit ourselves to cases with two sending APs. If more APs use COPA, only two will participate in a transmission cycle. At the end of each cycle, all APs will contend for access and subsequently a new AP pair will transmit. When there are more than two senders present, the fairness of COPA’s ITS mechanism merits consideration. When two COPA senders elect to send sequentially after an ITS exchange, they implicitly win two consecutive contention rounds. To avoid unfairness to other senders who may be present, after two COPA senders send sequentially, they should defer to other senders in the immediately following contention round by using a modified contention window of  $[aCW_{min} + 1, 2 \cdot aCW_{min} + 1]$ , rather than the default of  $[0, aCW_{min}]$ . We expect this deference to improve fairness; we leave an evaluation of this modification to future work.<sup>4</sup>

All ITS control packets contain the *Duration* field indicating the duration on the wireless medium of the data each AP will send to its client. To

---

<sup>4</sup>We expect this change either to negligibly reduce throughput or possibly even improve it, owing to a possible decrease in collisions. Note that this modification only takes place after a full ITS exchange, since for the rest of the coherence time the two hosts do not engage in further ITS exchanges.

nearby nodes not participating in the coordinated transmission, ITS control packets thus function in the same way as 802.11's RTS/CTS exchange: other nodes defer for the duration of the coordinated transmission, even if overhearing only one side of the ITS exchange.

One could argue that adding new MAC-layer functionality adds opportunity for attack by an adversarial host. For instance, a host sending an ITS REQ could include false channel state information, in an effort to improve its own throughput. The 802.11 MAC is already replete simpler attacks of this kind though, (*e.g.*, always choosing a contention window of zero) that a host could easily employ in order to monopolize the wireless medium.

#### **4.2.2 Per-subcarrier Power Allocation**

Given precoding matrices and CSI, COPA calculates the expected SINR at both clients on every OFDM subcarrier. This depends on how much power each AP sends on each subcarrier. COPA's goal, however, is not necessarily to maximize average SINR, but instead to maximize throughput. Current Wi-Fi standards constrain us to using a single decoder at the receiver [2], so subcarriers with a very poor SINR may cause a high bit error rate (BER) at the receiver, even if most of the subcarriers have good SINR.

To prevent bad subcarriers from causing bit errors, COPA simply drops them. It does so by indicating in the preamble of the A-MPDU to the receiver with a bit-mask which subcarriers to attempt to decode. Dropping subcarriers mitigates decoding errors, but also frees power to be allocated to other subcarriers (improving the bitrate achievable on them), and allows a concurrent sender to use the wireless capacity on an otherwise disused subcarrier interference-free.

Although it is not possible for Wi-Fi hardware to radiate zero power on a subcarrier—the typical carrier leakage from adjacent subcarriers for the Maxim 2829 transciever is  $-27$  dB [27], similar to what we experimentally observed—dropping a subcarrier leads to a reduction in interference. This in combination with nulling by the other user of the subcarrier so that it doesn't cause the AGC of the original receiver to choose very low values, thus not amplifying sufficiently its used subcarriers, allows the allocation of the available frequencies to whichever host can benefit aggregate throughput the most.

Furthermore, although standard 802.11 applies the same transmit power to every subcarrier used, as we saw in Section 2.10, alternative power allocations can improve throughput. The most well-known power allocation method is *waterfilling* [10], which maximizes the aggregate Shannon capacity of a system that uses a Gaussian constellation. According to this, subcarriers that have higher SINRs are given more transmit power since they can deliver symbols more reliably. But practical systems like 802.11 use *discrete* constellations, such as BPSK and 64-QAM. For such systems, as we saw in Figure 1.1, increasing subchannel SINR above some level has no effect in achievable throughput, while applying more power to weaker subcarriers could bring them above the necessary threshold for reliable communication. Based on this intuition, Lozano *et al.* propose *mercury/waterfilling* [63], which computes the optimal power allocation for systems that use discrete constellations.

Unfortunately, mercury/waterfilling has a rather high computational cost because it requires numerical integration, making it impractical for use in a real-time wireless implementation. Using the same intuition we instead propose *EquiSNR*. For a single AP transmitting to a single client without

interference from a concurrent sender, the procedure is shown in Algorithm 4.1.

**Algorithm 4.1:** Power allocation for one MIMO stream.

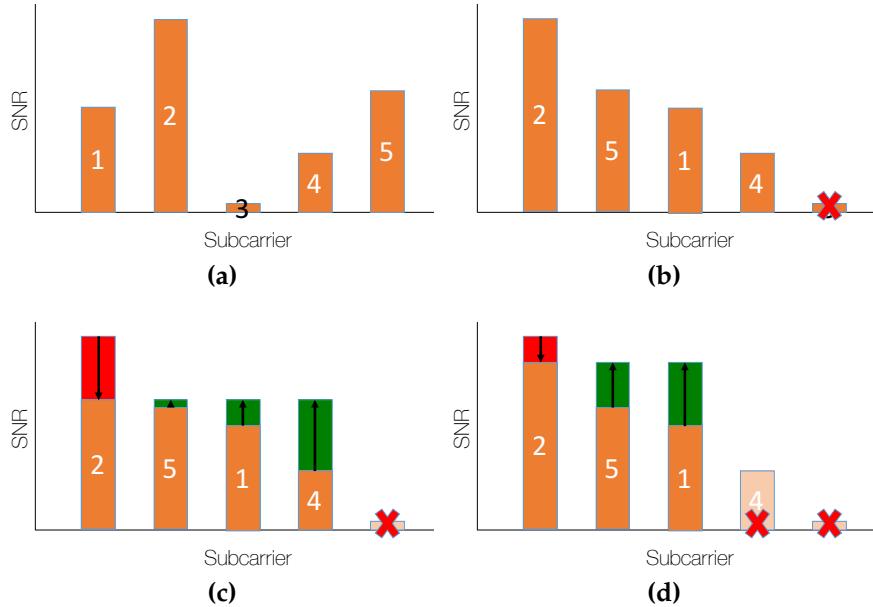
```

1 Sort the subcarriers into order of increasing SNR
2 for  $i$  in range(0, NUM_SUBCARRIERS) do
3     Allocate no power to the first  $i$  subcarriers
4     Allocate power to remaining subcarriers so as to equalize their SNR
5     Try all 802.11 modulations and pick the one that maximizes
        throughput using Effective BER
6     Calculate throughput given modulation and number of subcarriers
        used
7 end
8 Use number of subcarriers that maximizes throughput

```

Figure 4.8 illustrates an example of the application of the EquiSNR algorithm. Figure 4.8a shows the SNRs across the different subcarriers used by our system without any power allocation. After sorting the subcarriers by SNR, EquiSNR drops subcarrier “3”, which is the weakest one (Figure 4.8b), and reallocates the transmit power in order to equalize the SNRs on the remaining ones. In this case, it reduces the transmit power of subcarrier “2” and increases it for all other used subcarriers (Figure 4.8c). It then cycles through all bitrates used by 802.11 and calculates the expected throughput for each one, based on its Effective BER [78] and the number of subcarriers used. After it picks the bitrate that yields the highest throughput it continues by dropping subcarrier “4”, which is the next weakest subcarrier (Figure 4.8d), reallocates the transmit power, and re-calculates the expected throughput. This process repeats until we try all possible numbers of subcarriers and we are left with a single one.

EquiSNR tries to equalize the SNR at the subcarriers it hasn’t discarded, in an effort to increase the SNR of subcarriers with bad channel conditions



**Figure 4.8:** Example of EquiSNR.

while not wasting excess power on subcarriers that are already well above the decoding threshold for the bitrate used. Of course, trying to equalize the SNRs without dropping any subcarriers could have a potentially catastrophic effect—we could end up sacrificing too much of our transmit power to improve SNR for a subcarrier while worsening SNR for all others. As we will see in Section 4.3, though, the combination of subcarrier selection and power equalization can work well. The great advantage of EquiSNR over mercury/waterfilling is its low computational complexity, since it only requires a single division for each subcarrier—in order to find the inverse of its amplitude—instead of numerically calculating an integral, making it a good candidate for a real-time system.

## Concurrent Subcarrier Power Allocation

When two APs transmit concurrently to two clients, whether performing nulling or not, the task becomes much more complicated. For each AP we

can calculate the subcarriers used and power allocation for those subcarriers, as above. However, we must then take into account the interference caused by this choice of power allocation. To illustrate the point consider the following scenario:

We've decided AP2 won't use a subcarrier because of interference from AP1. This reduces interference at Client 1, so AP1 can reduce the power it uses on that subcarrier to improve others. Since AP1 reduces the power used, the SINR at Client 2 improves, so we now decide AP2 can use that subcarrier after all. To get acceptable SINR at Client 1, AP1 must now increase the power on that subcarrier, reducing power on others to compensate.

Since total power is limited, any change in power allocation to one subcarrier requires adjusting all others, changing interference on all subcarriers at the client of the other AP. It is clear that an optimal solution requires considering all possible power allocations and combinations of used subcarriers between the two APs, which becomes impractical.

To achieve an acceptable heuristic solution, we use the process shown in Figure 4.9. Based on the precoding matrix and CSI, we first calculate a power allocation solution *independently* for each MIMO stream between AP1 and Client 1 and between AP2 and Client 2. This initial power allocation assumes per-subcarrier interference based on the other sender's allocating its transmit power equally across all subcarriers. After this initial allocation, we compute the revised interference each stream causes to all the other streams. We feed this interference matrix back into the allocation algorithm and recompute which subcarriers should be used and how

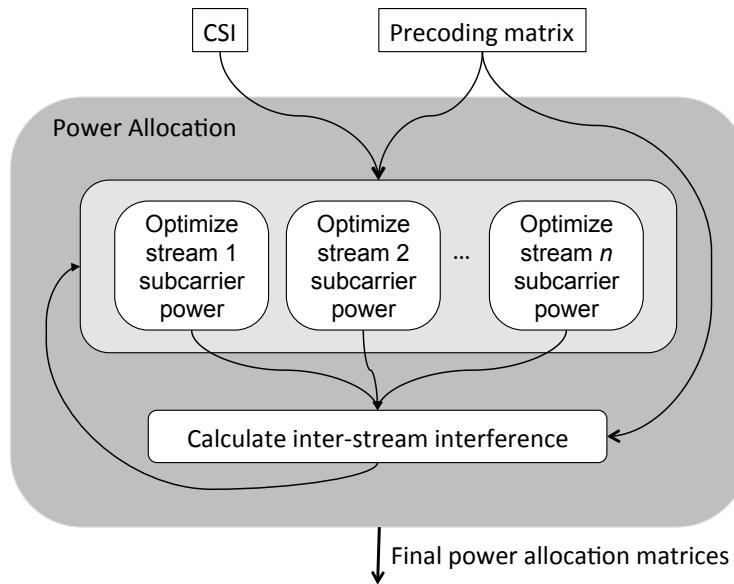
to allocate power among them. The process iterates until it converges or an iteration limit is reached.

Iterative algorithms may face difficulty converging to a good solution when there is instability between iterations. A simple example is the case of the single subcarrier described earlier. During our power allocation process, in order to increase stability, each stream is not allowed to change the number of subcarriers it uses between successive iterations without bound, but must keep at least half of the previously used subcarriers. For instance, if a stream was occupying subcarriers ranked 1 to 40 during the previous iteration and now wants to use only those ranked 1 through 10, it will be forced to keep those ranked 1 through 20. It can drop more during the next iteration. If between successive iterations there are no changes in the subcarriers selected by each AP or if we reach a preset number of iterations—ten in our experiments—the algorithm terminates. Furthermore, since there is no guarantee of reaching a global maximum, and because of independent allocations to each stream, the algorithm may occasionally diverge from the best solution, in which case it reports the best solution previously found.

This algorithm resembles Algorithm 4.1 (EquiSNR), but as it equalizes SINR rather than SNR to take account of this inference, we term it *EquiSINR*.

### EquiSINR Power Allocation: Example

To get some intuition about the behavior of COPA, we can examine what happens on the subcarrier level for a single stream. Figure 4.10 shows the BER per subcarrier for a single stream for a pair of four-antenna AP, two-antenna client networks, with concurrent transmissions, *i.e.*, when we use a

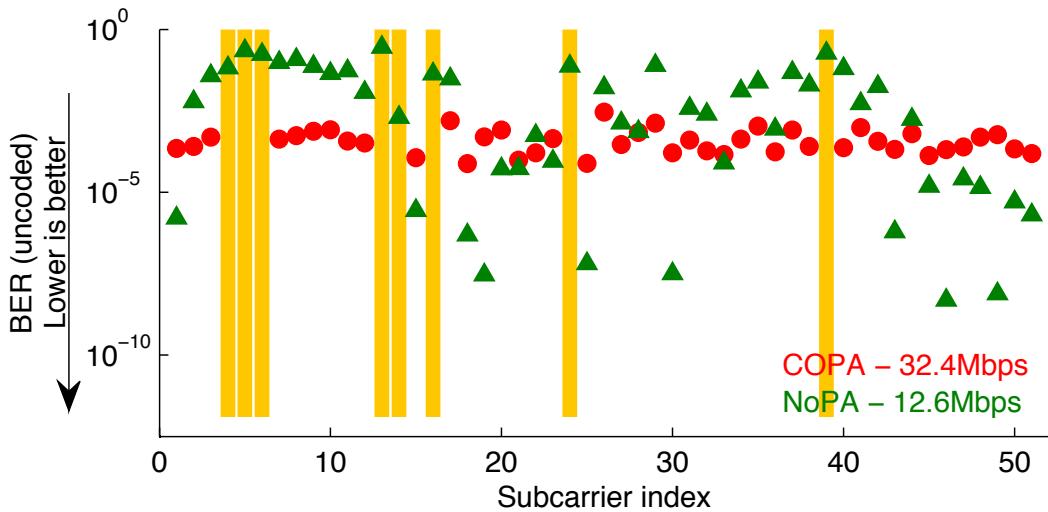


**Figure 4.9:** COPA's *EquiSINR* iterative power allocation.

nulling precoding matrix, and the same bitrate both for COPA and for the no-power allocation case. The throughput value for no power allocation (“NoPA”) is the one achieved when using its optimal bitrate (13.5 Mbps). Although NoPA offers a better BER than COPA on several subcarriers, it exhibits great BER variation across subcarriers. By contrast, COPA offers lower overall BER variation and drops particularly bad subcarriers. As a result, although COPA drops 8 subcarriers, it achieves higher throughput because it manages to use a higher bitrate (*39 vs. 13.5 Mbps*) on the remaining ones, leading to a significant throughput increase.

#### 4.2.3 Predicting the Best Strategy

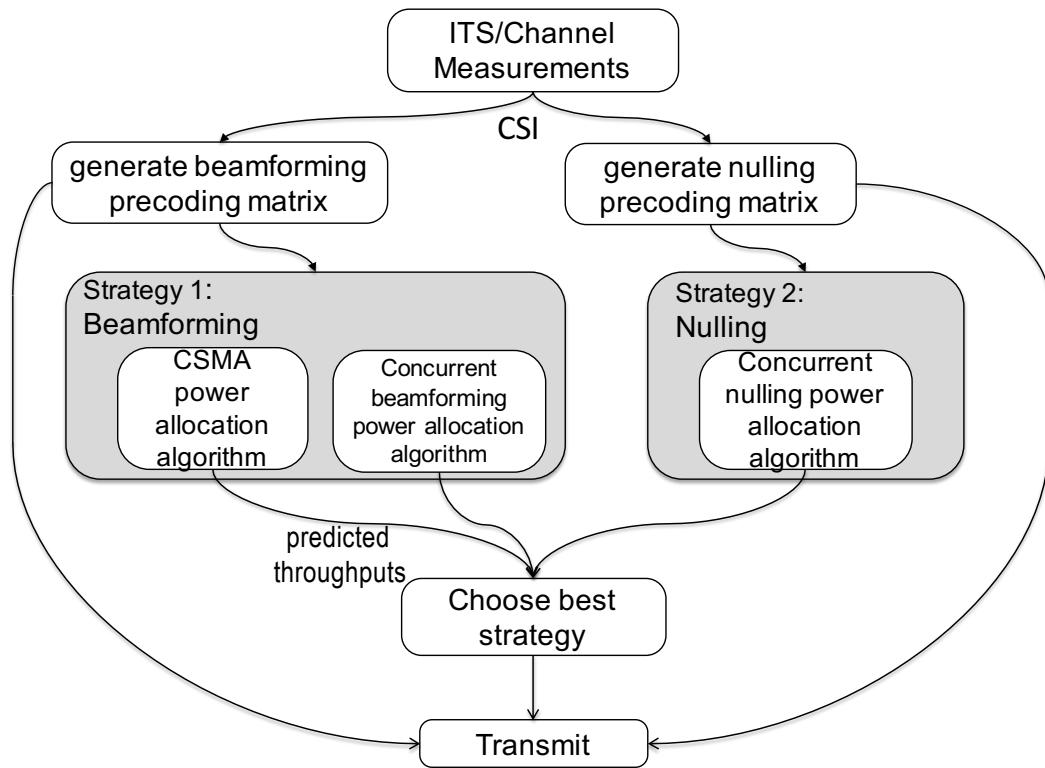
The overall architecture for each AP’s implementation of COPA is shown in Figure 4.11. After receiving the CSI, the leader AP calculates four pre-coding matrices: two “transmit beamforming” ones (one for itself and one for the follower AP) that maximize power at the intended receiver, and are calculated using the *Singular Value Decomposition* (SVD) of the appropriate



**Figure 4.10:** BER per subcarrier without coding when using the same nulling precoding matrix and bitrate (39 Mbps) with COPA and without (“NoPA”). Vertical bars denote subcarriers that COPA drops.

channel, and two “nulling” ones (again one for itself and one for the follower) that use a combination of nullspace projection and the SVD to null interference at the unintended receiver while maximizing power at each AP’s own client.

Returning to Figure 4.11, we see that COPA applies the EquiSINR power allocation and subcarrier selection algorithm and calculates the throughputs achieved by multiple medium access strategies. First of these is *COPA-SEQ*, in which transmitters use the transmit beamforming precoding matrices and transmit sequentially. In our evaluation, COPA-SEQ always beats stock 802.11n without power allocation, which is expected since the latter serves as its starting point. We can also use the same transmit beamforming precoding matrices for a concurrent strategy. This non-nulled concurrent solution only performs EquiSINR power allocation and subcarrier selection to mitigate interference. For single-antenna cases, where nulling is not possible, this is the only concurrent strategy we consider. But even when



**Figure 4.11:** Overview of COPA

nulling is possible, if the cross-interference is very weak, nulling may unnecessarily waste transmit power without yielding any useful reduction in interference. Finally, another concurrent strategy is to use EquiSINR with the nulling (or possibly alignment) precoding matrices. This subsumes traditional nulling, which serves as the starting point for the first iteration of EquiSINR.

We note here again that additional power allocation schemes are possible. In our experiments we also apply mercury/waterfilling (see §4.1.1) instead of equalizing SINR in Step 4 of Algorithm 4.1.

Once all these power allocation matrices have been generated, we calculate the effective BER and hence the aggregate throughput the two APs would achieve if they chose each scheme, after Halperin [78]. Again as in

Figure 4.11, COPA then transmits with the throughput-maximizing strategy.

Ideally, we could tell in advance which strategy would win without computing different solutions and comparing their throughput. In practice, this is not so easy: if cross-interference is relatively weak, nulling with power allocation always beats CSMA. If interference is very weak indeed, concurrent sending without nulling can even beat nulling, as it sends more power to the intended recipient. In the interest of correctly choosing the scheme that offers the greatest throughput, COPA therefore explicitly computes the expected throughput of each scheme and chooses the predicted winner, taking channel conditions into account.

#### 4.2.4 Overconstrained Nulling

Nulling requires that a transmitter have enough antennas to phase-align its transmissions so that they cancel each other at an unintended receiver. For example, if each AP has four antennas and each client has two antennas, each AP can generate two MIMO streams to its own client and still null its transmission at both the other client's antennas.

Sometimes, though, the problem is overconstrained. If, for example, two APs have three antennas each, and their clients each have two antennas, then the APs are faced with a stark choice: either send two MIMO streams each but don't null, resulting in significant interference, or send only one MIMO stream each and null that stream at the other AP's client. The problem arises from the fact that if a receiver receives more concurrent streams than the number of its antennas, it cannot use a Minimum Mean Square Error receive filter [10] to disentangle them, which can make the intended streams undecodable.

We have investigated whether in this scenario it is possible to send two streams while *partially* nulling by optimizing the precoding matrix for aggregate throughput. The short answer is that in many cases it is, but this optimization is extremely computationally expensive, requiring tens of seconds of compute time for calculating the precoding matrix for a single subcarrier. Instead, because our APs cooperate, we can adopt a more effective and much cheaper solution: simply shut down a receive antenna.

When the second AP responds to the first AP's ITS, it already knows whether the problem is overconstrained. In its responding ITS, it indicates that it wishes to participate in a concurrent transmission (if that is what COPA decides is the best strategy), but with reduced rank so that the problem is no longer overconstrained. It chooses whichever of its client's antennas have the best expected SINR, and indicates that the other antennas should be shut down for this transmission. By doing so, the receiver does not have to deal with the potentially high levels of interference that will inevitably end up in the shut-down antennas and that it cannot deal with using MMSE filtering, since it doesn't have enough degrees of freedom.

In the case of two three-antenna APs sending to two two-antenna clients, this approach would result in AP1's sending two MIMO streams to client 1 while nulling to client 2's remaining antenna, whereas AP2 sends one MIMO stream to client 2 and nulls to both of client 1's antennas. In principle this may give up to a 50% throughput improvement over CSMA.

Although this solution is asymmetric—one client gets more throughput than the other—randomness in the DCF results in each AP's sending the first ITS about half the time, so on average the asymmetry cancels out. This simple solution works well, as we show in the evaluation.

### 4.2.5 Incentive Compatibility

Our goal so far has been to maximize aggregate throughput. This clears any transmission backlog fastest, but it may result in one receiver's getting lower throughput than it would if its AP had not cooperated. In general, we would like a solution to be *incentive-compatible*, in the sense that no client ever loses out if its AP cooperates using COPA. In this way, APs always have an incentive to cooperate.

A simple tweak to COPA makes it incentive-compatible: simply revert to sequential transmission with power allocation and subcarrier selection if concurrent transmission would reduce either client's throughput. This is done by including this additional constraint when deciding on the "best strategy" in Figure 4.11. We evaluate this variant alongside that without this constraint next in Section 4.3.

## 4.3 Evaluation

COPA is motivated by experiments that show nulling does not perform well in our environment. But as Figure 4.3 shows, nulling does significantly reduce interference, and thus increases SINR. However, it also introduces an increase in the SINR variability across subcarriers (Figure 4.4), resulting in bit errors on weak subcarriers, thus seriously limiting throughput.

In this section we experimentally demonstrate that in many scenarios, summarized in Table 4.2, leveraging cooperative subcarrier selection and power allocation rescues nulling's performance and allows APs to transmit concurrently. Despite this, we will see that our nulling/power-allocation solution does not always work well, and so a system must consider multiple possible algorithms including sequential transmissions.

Experiment	Conclusion or performance improvement	Discussed in
Single antenna scenario	COPA benefits single antenna hosts primarily by improving their performance when transmitting without interference, improving throughput by 15% over CSMA.	§4.3.2
Constrained nulling scenario	Two four-antenna APs transmit concurrently to their two-antenna clients while nulling their unintended receiver, offering a mean improvement of 54% compared to vanilla nulling.	§4.3.3
Nulling with weaker interference	When interference is lower, vanilla nulling is more effective but COPA still outperforms it by 36%.	§4.3.4
Overconstrained nulling scenario	When APs do not have enough transmit antennas to null unintended receivers, clients shut down some receive antennas to make nulling possible. COPA outperforms vanilla nulling by 39%.	§4.3.5
Nulling with multiple decoders	Using multiple encoding and decoding chains allows each subcarrier to use a different bitrate, giving COPA a further 5% throughput improvement.	§4.3.6
Contributions of features of EquiSINR	Equalizing SINR without switching off subcarriers provides only modest improvement, while most of the benefits in EquiSINR come from the first two iterations.	§4.3.7

**Table 4.2:** Summary of experimental results for the proposed techniques, and the corresponding conclusion or performance improvement.



**Figure 4.12:** AP and client locations in the 7th floor of the UCL CS building.

### 4.3.1 Experimental Methodology

We ran experiments throughout our lab, which includes open-plan floor space as well as offices and corridors. Figure 4.12 shows the host locations used throughout our experiments.

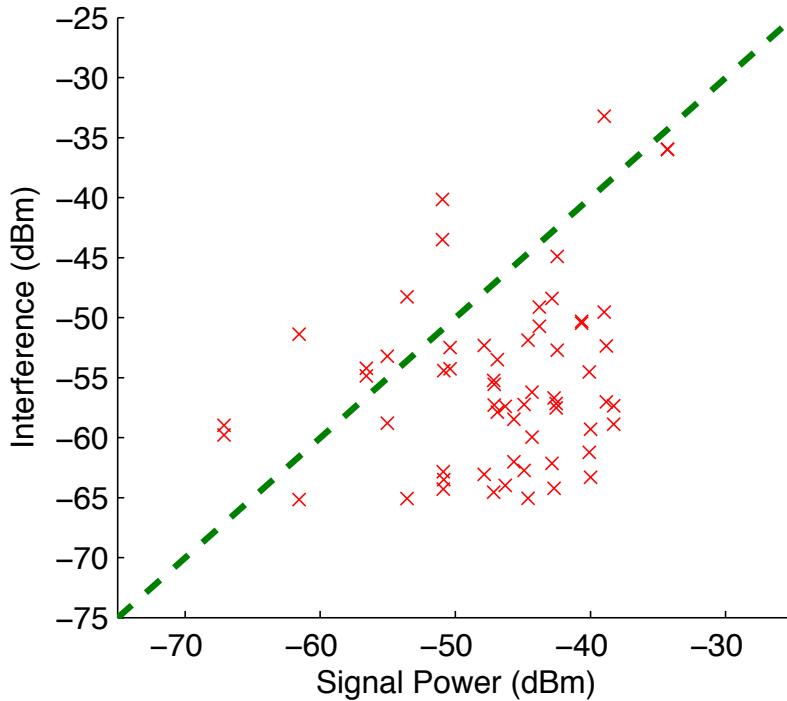
We use the WARP version 2 platform for both senders and receivers running in the 2.4 GHz band using 20 MHz channels and 15 dBm of maximum transmit power (commodity hardware typically uses between 10 and 20 dBm). All four nodes are connected to a PC that runs a modified version of the WARPLab framework that is used to calculate the precoding matrices and power allocations for our experiments. Our WARPs are mounted on trolleys so we can move them around the building to test many different topologies.

We choose topologies that include both short and long links, and constructed the scenarios so that usually the signal of interest was more powerful than the interfering signal. The rationale for this is that hosts are normally (but not always) closer to their own AP than to an interfering AP. In a few topologies we deliberately positioned the receiver so its direct line of sight was blocked by a metal filing cabinet. In Figure 4.13, each receiver in each topology appears as a point. We plot SNR at a client from its own AP against the SNR of the signal from the interfering AP (effectively the INR). Clients below the  $x = y$  line receive a stronger intended signal than interfering signal. It can be seen from this plot that our topologies include a fairly wide range of signal and interference strengths, though there are few really poor channels because of the nature of the building. We separately examine channels with weaker interference in Section 4.3.4.

In each topology we ran the following scenarios:

- Two single-antenna APs transmitting to two single-antenna clients.
- Two four-antenna APs transmitting to two two-antenna clients. We refer to this as the “constrained case”, where four MIMO streams and full nulling should be possible.
- Two three-antenna APs transmitting to two two-antenna clients. We refer to this as the “overconstrained case”, where there are not enough transmit antennas to both send four MIMO streams and fully null.

WARP doesn’t straightforwardly support synchronizing transmission by two independent four-antenna senders. Therefore, in our experiments involving concurrent transmission, each sending host transmits on its own to the two receivers, which record the samples they observe and then combine them. Normally, each transmission is scaled by the receiver’s AGC



**Figure 4.13:** Signal power from each client’s own AP plotted against interfering signal power from the other AP; each point is one receiver.

so that it fills the dynamic range of the ADC in the transceiver. Before combining the two signals, we undo this scaling by dividing each signal’s samples by the gain applied by the AGC, in floating point to avoid losing precision. We do so both because of the WARP’s limited synchronization capabilities and because doing so allows us to take more accurate interference measurements, which on many occasions would otherwise have been lost below the noise introduced by the variable-gain amplifier.<sup>5</sup>

A full COPA deployment would need to synchronize two AP’s transmissions. Rahul *et al.* [77] have demonstrated a method that can synchronize two sending hosts within 20 ns, which is shorter than the sampling inter-

---

<sup>5</sup>Although this process results in reduced quantization noise for the weaker transmission than what it would normally experience, it doesn’t affect our results. If the interfering transmission is the weaker one, we do not decode it. If the weaker transmission is the useful one, it will in any case be undecodable because of its low SINR.

val of a typical 802.11 transceiver. At the start of a reception, receivers use AGC to set the correct amplifier gain and Schmidl-Cox for synchronization. With the two transmitting hosts synchronized within tens of nanoseconds, both of these methods work correctly. The potential phase offsets between the interfering and useful transmissions at the receiver are irrelevant, since it needn't decode the interfering signal. One potential difficulty of combining the two transmissions is the addition of *Additive White Gaussian Noise* (AWGN) [10] twice, but as the noise level of the interfering transmission is typically significantly lower than that of the useful one, doing so only underestimates the performance of our system.

To send multiple streams, hosts use the singular value decomposition of the channel and to null we project onto the appropriate nullspace.<sup>6</sup> On the receiving side, hosts use a Minimum Mean Square Error filter to maximize the received power without amplifying noise.

We use the measured SINRs to calculate the uncoded BER [78] for each 802.11n modulation, from which we in turn calculate the coded BER for 802.11n's different coding rates [10]. From the frame error rates, we predict throughput achieved using 802.11n's standard 4 ms transmit opportunity duration.

Our results include the appropriate MAC overhead for each scheme—ITS for concurrent schemes, CTS-to-self for CSMA, preamble, and ACK. During the ITS exchange, we include the transmission of CSI and precoding matrices once every 30 ms. Our experiments assume greedy unidirectional flows without congestion control. This means that transmitting hosts always have enough data to send to their respected receivers in order to cover the full transmit opportunity duration at the bitrate chosen by COPA.

---

<sup>6</sup>See Appendix A for more information.

If that is not the case, a sender could either choose a lower bitrate or only send for part of the transmit opportunity. Either of these two strategies would limit the throughput benefits of COPA when transmitting concurrently. However, available load can be taken into account when hosts decide their transmission strategy. They can proceed with sequential transmissions if one of the two transmitters does not have enough data to send. Another approach would be to assign to the non-backlogged transmitter just enough subcarriers so that it can fill the transmit opportunity duration, while leaving the rest of the subcarriers for the sole use of the other transmitter, thus allowing it to potentially achieve a higher bitrate. We leave the examination of these strategies to future work.

We also assume that each AP already has knowledge of the channel between itself and its own clients, which would normally be measured implicitly from previous transmissions of data or control frames. If (fresh) CSI for a client is not available, its AP can either use the *No Data Packet* (NDP) mechanism or probe for a frame with staggered preambles to acquire it, as is done in standard 802.11n/ac. Doing so would require the exchange of two short packets (20–30  $\mu$ s) adding about 0.2% of overhead in an environment with a 30 ms coherence time; COPA and vanilla 802.11n would incur the same such overhead. Finally, clients send their ACKs at the highest possible bitrate which can ensure a ACK frame error rate of less than 0.1%, in order to ensure their correct reception, since lost ACKs result in retransmissions.

Selectively using subcarriers could problematically increase the *Peak to Average Power Ratio* (PAPR) [79]. In our experiments hosts only drop a few

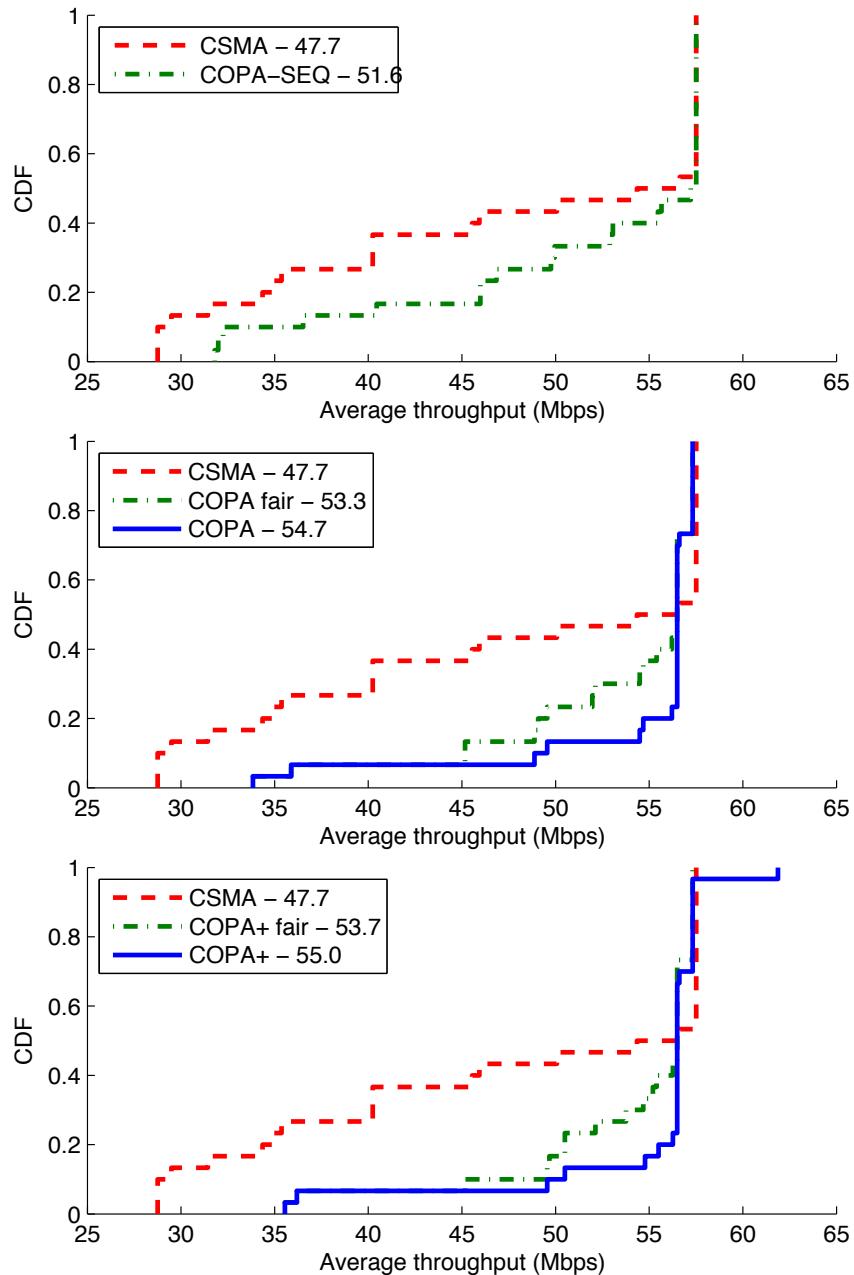
subcarriers; there are enough remaining and they have enough entropy from data scrambling that we do not observe any such problem.

**Limitations of our methodology.** The WARP platform incurs a latency cost when downloading samples received and uploading samples to send. In our experiments, there is thus a 2–3 second delay between measuring the CSI and using the resulting precoding matrix and power allocation for concurrent transmissions. However, measurements of our indoor testbed environment (omitted in the interest of brevity) show that the channel coherence time in our experiments exceeds the WARP’s latency penalty, and so the results we present are representative of a system operating at line speed. As noted above, our experimental evaluation of the throughput achieved by COPA, however, includes the overhead of COPA’s ITS exchanges (see Table 4.1) in an environment with a much shorter 30 ms coherence time—we do so to account for COPA’s costs in a more dynamic environment.

Finally, we cannot measure the performance of mercury/waterfilling in live experiments, as computing its concurrent power allocation solution requires tens of seconds (typically 30–50 s) in four-stream scenarios. For these results, we instead use an emulated channel.

### 4.3.2 Single Antenna Scenario

Although we don’t anticipate concurrent sending to work well very often with single-antenna APs, we investigate this scenario in order to understand the effect and limitations of power allocation and subcarrier selection. In Figure 4.14, we present three CDF graphs showing the effects of the different strategies in 30 topologies—the top graph shows non-concurrent variants, the middle shows practical COPA variants (including concurrent



**Figure 4.14:** Throughput CDF (across topologies) for two single-antenna AP, single-antenna client pairs.

and sequential transmission), and the lower graph shows the best (but impractical) COPA variants.

In just under half the topologies regular CSMA (*i.e.*, no concurrent senders) can achieve throughput of 57.5 Mbps—the maximum achievable rate when transmitting at 65 Mbps with a 4 ms transmit opportunity. The mean throughput though is 47.7 Mbps because the remaining receivers fare fairly poorly. The COPA-SEQ curve shows the effect of EquiSINR power allocation and subcarrier selection, without concurrent senders. COPA-SEQ achieves a mean throughput of 51.6 Mbps. We have investigated whether this improvement comes from subcarrier selection or from power allocation: either one by itself gives about 60-70% of the improvement, but both are needed together for the full benefits to be seen. CSMA plus mercury/waterfilling, for example, gives 50.1 Mbps in this scenario.

In the middle graph we can see the improvement over CSMA when we allow COPA to do concurrent transmission. The “COPA Fair” curve shows how COPA performs when we restrict it to be incentive-compatible as we saw in Section 4.2.5, whereas “COPA” just aims for maximum total throughput.

“COPA Fair” achieves 3% more than COPA-SEQ; the improvement arises from selection of concurrent transmission in some topologies, even though nulling is not possible with a single antenna. In about 15% of the topologies there is a slight drop in throughput when using “COPA Fair” because of the increased MAC overhead.

The “COPA” curve shows how COPA’s non-incentive-compatible version performs. Aggregate throughput now rises to 55 Mbps, and concurrent transmission is possible in more cases, even though without nulling COPA does not greatly improve throughput. Here COPA has selected what is essentially a form of OFDMA, with some subcarriers being used by only one AP at a time. In these few cases each subcarrier is used by the AP

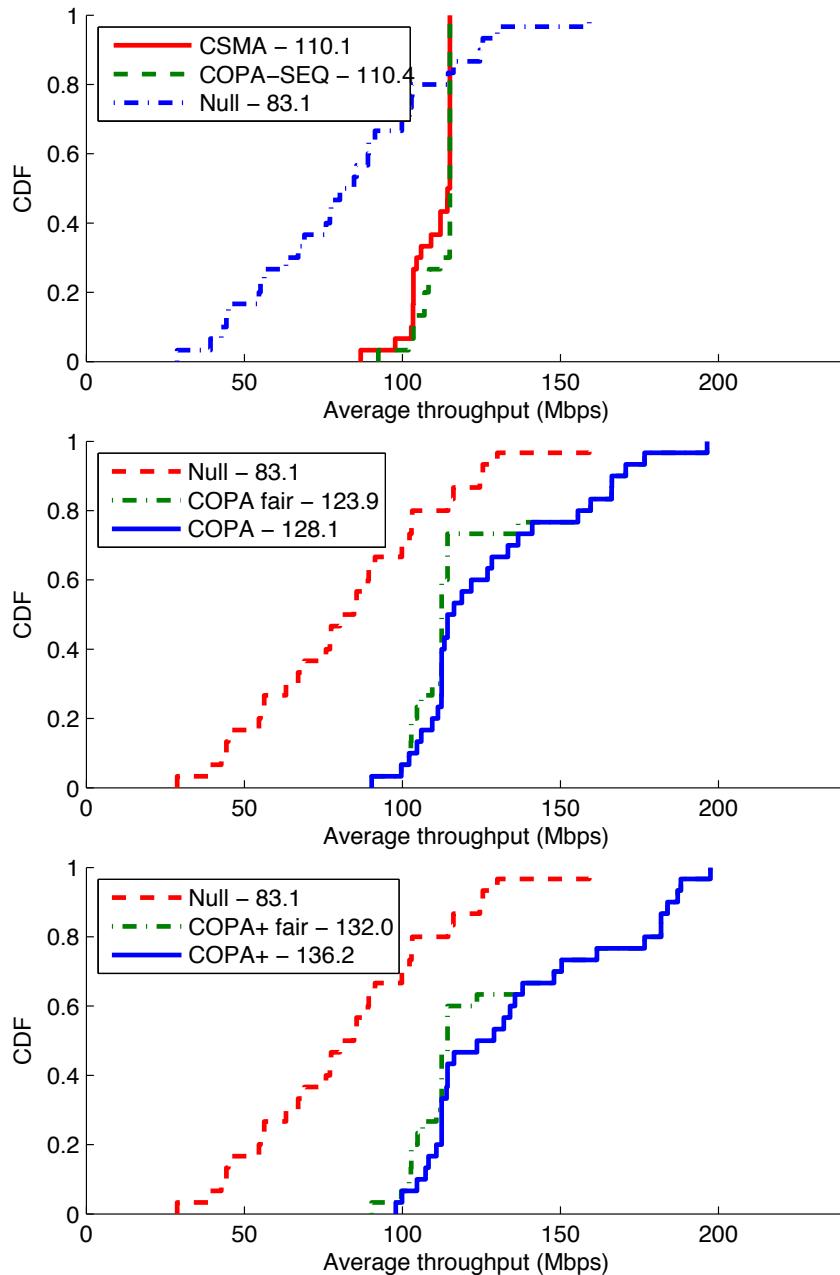
that can best make use of it, despite unfairness. The difference between the “COPA” and “COPA Fair” curves is the price of fairness. In this single-antenna case, COPA often allocates some subcarriers exclusively to one sender.

The “COPA+” curves in the lower graph in Figure 4.14 show what happens when we include iterated mercury/waterfilling (including subcarrier selection) among the strategies COPA can select. These curves are trace-driven emulation based on real CSI measurements because of the high processing time of this algorithm. Although COPA+ is computationally impractical, the curves illustrate the additional gains that a more optimal power allocation scheme might achieve. At the top right of the graph we see true concurrent transmission occurring in two topologies, with the same subcarriers successfully used concurrently by both APs.

### 4.3.3 Constrained Nulling Scenario

In Figure 4.15 we show the performance achieved by two four-antenna APs transmitting to two two-antenna clients. In this scenario there are enough degrees of freedom that it ought to be possible to perform concurrent transmission of two MIMO streams from each AP to its client while nulling at the other client. Again, the top graph shows non-concurrent variants, the middle shows COPA with concurrent transmission, and the lower graph shows the impractical COPA+ results. We also show regular nulling without power allocation or subcarrier selection as a baseline in all graphs.

When we first obtained these results we were surprised by how poorly nulling performs with OFDM. Nulling only outperforms CSMA in 17% of our topologies, and even then, not by much. We examined the cause in Figure 4.4. Nulling works well for some subcarriers but not so well for



**Figure 4.15:** Throughput CDF (across topologies) for two four-antenna AP, two-antenna client pairs.

others. This, added to the variability in SNR when a transmitter tries to null an unintended receiver, leads to high SINR variability between subcarriers.

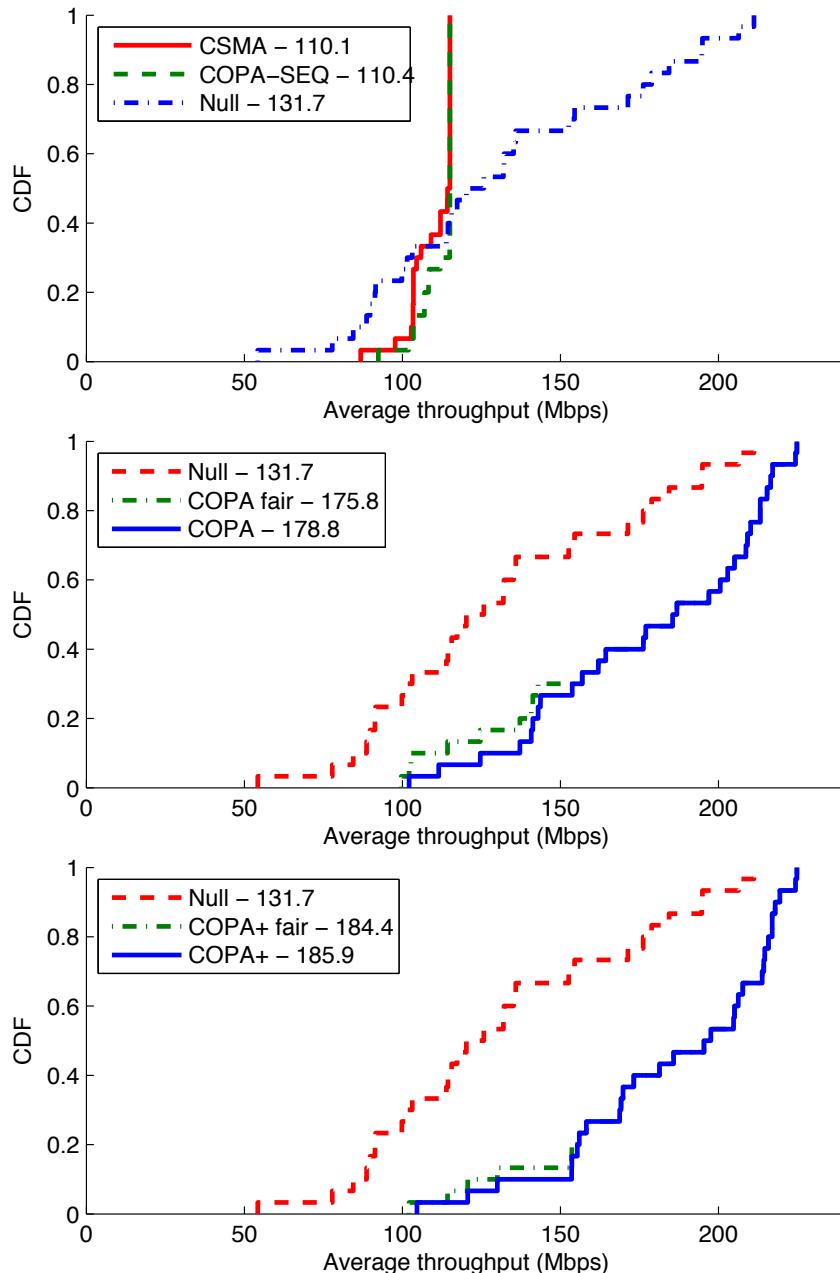
Because 802.11 hosts use a single decoder, weaker subcarriers dominate BER, dragging down the achievable bitrate.

In this scenario, CSMA achieves two full MIMO streams in 60% of the topologies, with only slightly reduced throughput for most of the remainder, as the power budget with four antennas is 4x higher than in the previous scenario.

The “COPA” curve in the middle graph shows how much more effective nulling is when combined with power allocation and subcarrier selection. There is a mean of 54% improvement over vanilla nulling. If we use the incentive-compatible “COPA fair” variant that figure drops to 48%; the 6% difference between the two is the cost of fairness. In approximately 30% of cases “COPA” selects COPA-SEQ because no better concurrent solution was found. In all the remaining cases a concurrent nulled solution is chosen. The mean throughput improvements don’t tell the whole story though—nulling has higher variance than CSMA. Even though COPA improves things significantly, the variance is still high. Sometimes COPA gives negligible improvement over CSMA, but when it uses concurrent transmissions the gains are substantial. Finally, the COPA+ curves in the lower graph indicate that were it not for prohibitive computational cost, using iterated mercury/waterfilling might yield a further increase of about 10%.

#### 4.3.4 Nulling with Weaker Interference

Other researchers have reported nulling works better in their environments than we find it to in ours. We speculate that their building construction or choice of interferer location may result in lower cross-interference than we observe. To test this hypothesis we took the traces from all our 4x2



**Figure 4.16:** Throughput CDF (across topologies) for two four-antenna AP, two-antenna client pairs, when interference is 10 dB weaker than empirically measured in our testbed.

topologies, reduced the interference strength by 10 dB, left the signal of interest unchanged, and ran emulated experiments. This results in emulated

topologies where APs can still hear each other well enough to exchange ITS packets, but where before nulling, interference is on average about 20 dB below the intended signal.<sup>7</sup>

Results are shown in Figure 4.16: with weaker interference, vanilla nulling works relatively well, beating CSMA in 65% of topologies. However, COPA greatly increases throughput. With weak interference, COPA almost never needs to fall back to COPA-SEQ. There is little difference between “COPA” and “COPA Fair” because both clients normally win from running COPA. Even when they don’t both win, the weaker client doesn’t suffer greatly. COPA beats CSMA by 62% and beats vanilla nulling by 36%. The win is biggest for the receivers that do worst with nulling, with many receivers getting more than 50% better throughput with COPA than with vanilla nulling. COPA+ does even better, beating vanilla nulling by 41%, but is unlikely to be practical.

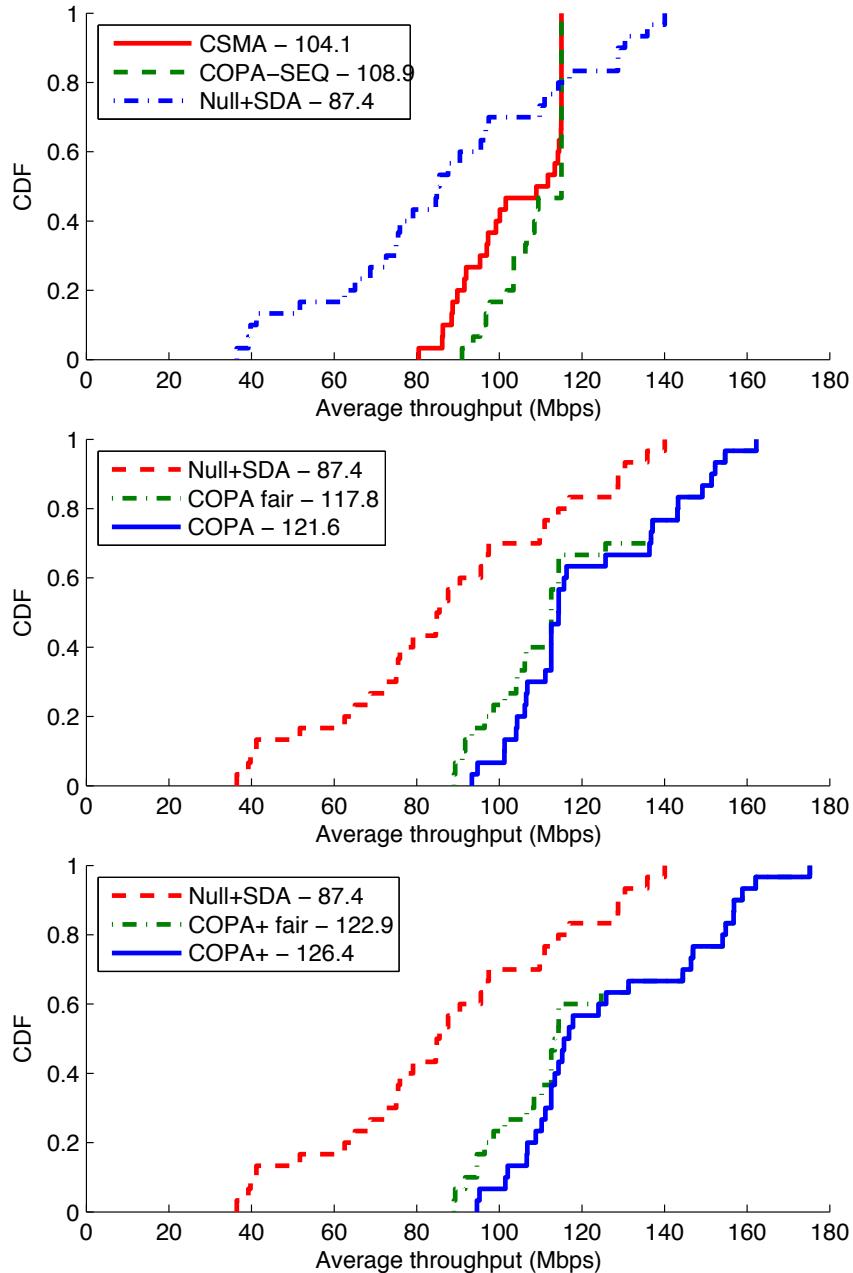
### 4.3.5 Overconstrained Scenario

Between the single-antenna case and the full 4x2 case lie overconstrained scenarios, where some measure of nulling is possible, but there are not enough degrees of freedom to null completely. To explore this region, we examined the case where two three-antenna APs each have a two-antenna client. In this scenario we can normally only use CSMA to send two streams at a time, since the transmitters do not have enough antennas to null towards their unintended receiver.

We have already seen that even when there are enough degrees of freedom, nulling is of limited effectiveness when using OFDM. To improve the starting condition for COPA’s concurrent strategies, we have one of the two

---

<sup>7</sup>To see this, take Figure 4.13 and move each point down by 10 dBm.



**Figure 4.17:** Throughput CDF (across topologies) for two three-antenna AP, two-antenna client pairs.

APs tell its receiver to shut down a receive antenna, so that sender is no longer overconstrained. Both APs then have enough degrees of freedom

to proceed with their transmission while nulling towards their unintended receiver.

The “Null+SDA” curve in Figure 4.17 shows the effect of shutting down an antenna (SDA) and then performing what is otherwise vanilla nulling. This provides some benefit to clients with good channels but by itself doesn’t approach CSMA’s throughput. COPA does significantly better. “COPA Fair” beats CSMA by 13% and “COPA” beats CSMA by 17%. Around 40% of topologies can choose concurrent strategies, and those that do gain between 20% and 40% over CSMA. Unsurprisingly, COPA+ does even better, with around 60% of topologies benefiting from concurrent strategies.

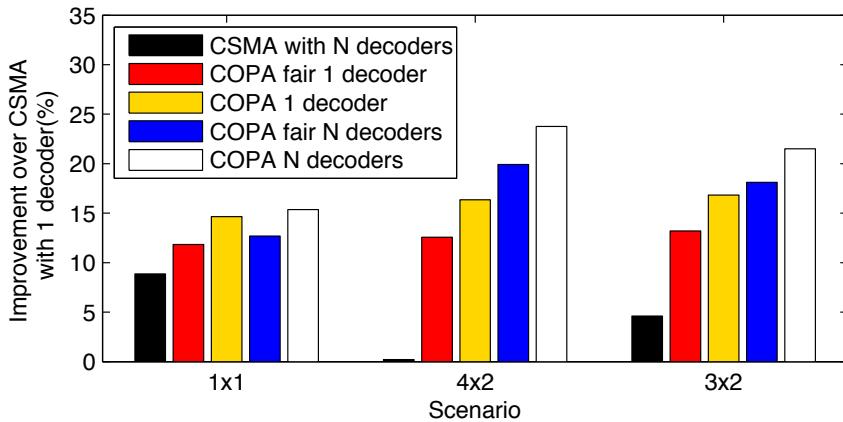
#### 4.3.6 Multiple Decoders

As we saw earlier, the variability of SINR across subcarriers results in diminished throughput. By dropping or allocating more power to subcarriers that experience more adverse channel conditions, COPA helps recover throughput that would otherwise be sacrificed. Although current hardware does not support it, a further improvement can come by selecting the bitrate for each subcarrier independently. To do so, both AP and client would need to use multiple encoders and decoders for each stream (one for each coding rate supported). However, this approach would allow us to adapt better to the SINR of each subcarrier, and accommodate greater variation variation across subcarriers.

In Figure 4.18 we examine the potential benefit of using one decoder per coding rate in our topologies.<sup>8</sup> The figures give the improvement of each scheme over what CSMA would achieve with one decoder. In the single-

---

<sup>8</sup>802.11 uses four coding rates and current 802.11ac transceivers usually have two decoders, each one used for different streams [2].

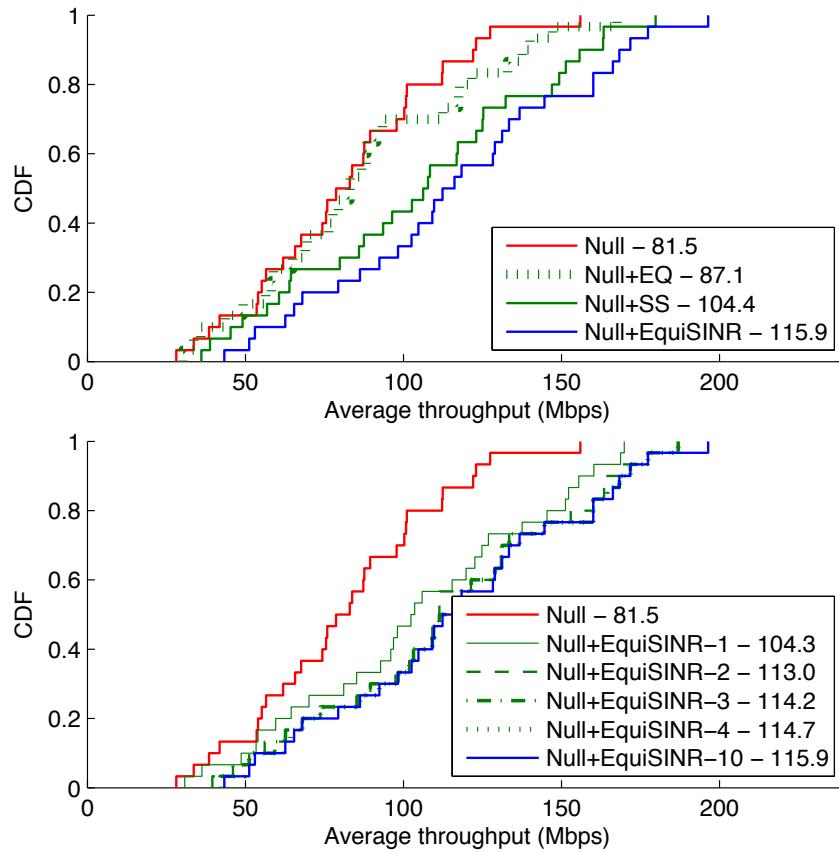


**Figure 4.18:** Potential percentage improvements from using per-stream error control coding with CSMA and COPA.

antenna case, using multiple decoders improves CSMA performance, but fails to greatly improve COPA performance, as nulling isn't possible. In the four- and three-antenna AP cases, on the other hand, CSMA doesn't benefit significantly as it is already running at the top bitrate. Using one decoder per channel increases throughput over baseline "COPA" and "COPA Fair" by about a further 10% with four-antenna APs and by about 5% with three-antenna APs. These gains are modest: even with a single decoder COPA, has already realized most of the potential gains.

### 4.3.7 Contributions of Features of EquiSINR

In Figure 4.19 we show the CDF of the aggregate throughput for the two four-antenna AP, two-antenna client scenario when using different features of the EquiSINR algorithm. The "Null" curve shows the baseline scenario, when both transmitters are sending two streams to their respective clients while nulling the unintended ones. The "Null+EQ" curve shows what happens when we iteratively allocate power in order to equalize SINR across subcarriers and across receivers without doing any subcarrier selection (*i.e.*, without dropping any subcarriers), while the "Null+SS" curve shows the



**Figure 4.19:** Throughput CDF (across topologies) for two four-antenna AP, two-antenna client pairs with concurrent transmissions and both APs nulling.

complementary choice. Finally, the “Null+EquiSINR” shows what happens when we use the EquiSINR algorithm with the same precoding matrices. As we can see, simply applying SINR equalization provides relatively modest gains, since by doing so we often dedicate too much transmission power in order to reduce BER on just a few subcarriers. Using iterative subcarrier selection, on the other hand, is more effective, since we abandon subcarriers that introduce too many bit errors at the desired bitrate, and yields about 28% throughput improvement. Finally, combining the two methods increases gains even further to 42%.

Another important aspect of the EquiSINR algorithm is its iterative nature. In the bottom graph of Figure 4.19, there are several curves marked as “Null+EquiSINR-X” where “X” is the maximum number of iterations used if the algorithm has not converged earlier. As we can see, the greatest gains come after the 1<sup>st</sup> iteration, which is in effect the “non-cooperative” version of EquiSINR, since in this case each transmitter hasn’t yet taken into account the power allocation and subcarrier selection decisions of the other concurrent transmitter. Nonetheless, increasing the number of iterations still yields significant, though diminishing, returns, increasing gains by 50%, from 28% to 42% for the 10<sup>th</sup> iteration.

## 4.4 Discussion

We have presented COPA, an approach for mitigating interference between loosely cooperating Wi-Fi APs and clients that leverages per-subcarrier power allocation, interference nulling, and multi-stream transmission to improve throughput. Our experiments show that interference nulling increases the variability of SINR across subcarriers at receivers, but COPA’s cooperative power allocation mitigates this effect, materially boosting throughput.

The techniques that comprise COPA should deliver throughout improvements in interfering OFDM networks operated independently without central control, but also in those deployed in enterprise settings, where inter-stream interference can still exact a toll in throughput. In the next chapter, we explore the extent to which COPA-like techniques can benefit enterprise Distributed MIMO systems.

## Chapter 5

# Power Allocation for Distributed MIMO (PADM)

Wireless networking is a popular choice not just in the home and small-business environment, but in the enterprise world as well. Developments such as *Bring Your Own Device* (BYOD) and *hot-desking* mean that employees increasingly use mobile and portable devices and are less willing to be tethered to an Ethernet cable. Similarly, venues like malls, stadiums, concert and conference halls often offer Wi-Fi connectivity to their customers, who primarily use their mobile handsets to access the Internet.

Such wireless deployments have characteristics that differentiate them from the scenarios we considered in the previous chapters. Unlike domestic and SME networks, which are typically installed in a chaotic and unplanned manner, these networks are usually carefully planned and deployed in such a way as to maximize the number of users they can serve while maintaining low installation and maintenance costs. Another important differentiator is the higher user density they must deal with. To cope with this density of users, network operators typically deploy multiple APs and either

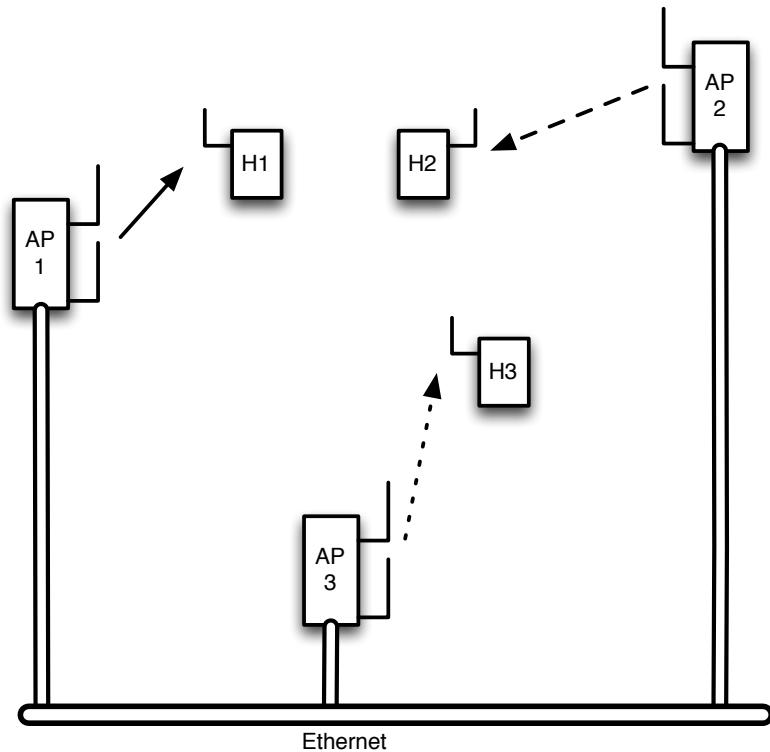
restrict their transmit power or set them to different frequency channels to avoid inter-AP interference and maximize concurrency.

Recent work [18, 80, 46] has proposed *Distributed* or *Cooperative MIMO* as a way to increase aggregate throughput in enterprise deployments. Such systems take advantage of all APs' management by the same authority, which means that they all have access to all data frames transmitted. Synchronizing all APs' transmissions creates a large *virtual AP* that uses the standard multiuser MIMO techniques to transmit multiple streams of information to wireless clients.

Increasing the number of concurrently served clients beyond some point yields diminishing returns because of *channel hardening* [45]. The additional channels created by adding antennas start to show increasing correlation with existing ones, and inaccuracies introduced by hardware and channel noise lead to an increase of *inter-stream* interference with each new added stream. In this chapter, we show that these phenomena can be significantly mitigated through the use of subcarrier power allocation and selection.

## 5.1 Problem

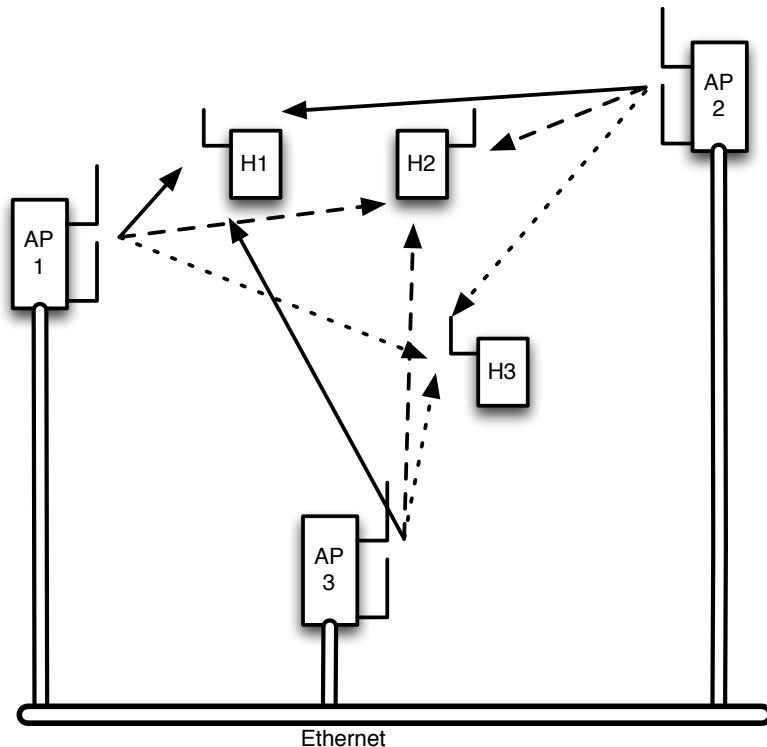
Enterprise wireless deployments are designed to serve moderate (*e.g.*, workplace) to high (*e.g.*, conference venue) host densities, unlike domestic and SME deployments, where there are fewer users, and hence fewer devices. To deal with the increased density, such venues are equipped with several APs. In Figure 5.1, we can see an example enterprise network. In this case, we have 3 APs and 3 clients. Each client is associated with a single



**Figure 5.1:** Enterprise network with multiple APs. Each AP transmits to a single host.

AP, which uses beamforming whenever it transmits a packet to maximize the received power at its client.

Unfortunately this approach comes with inefficiencies. If the three APs can sense each other's transmissions, CSMA/CA will lead to a single active transmission at any given moment. Network operators could set the transmission power of each AP at a relatively low level but doing so will lead to dark spots close to the periphery of APs' ranges. Alternatively, we could set the APs to operate on different frequency bands, essentially emulating cell deployment in cellular networks. However, the available bandwidth for 802.11 networks is limited and even though further parts of the spectrum became usable with the introduction of 802.11n, increased throughput demands have led to developments such as *channel bonding* [2], where



**Figure 5.2:** Distributed MIMO system. All APs transmit to all hosts concurrently.

hosts can bundle together several channels and treat the combination as a single one, leading to a small number of non-overlapping combined channels. Furthermore, these new parts of the spectrum are in the 5 GHz band, which has much poorer propagation characteristics [81], and are thus often unsuitable for use in large spaces with complex geometries, as is often the case in enterprise environments. Also, cellular-type deployment is often costly in planning and maintenance. Shortfalls in either of these areas will inadvertently lead to problems like inter-AP interference, poor coverage, and/or unused resources.

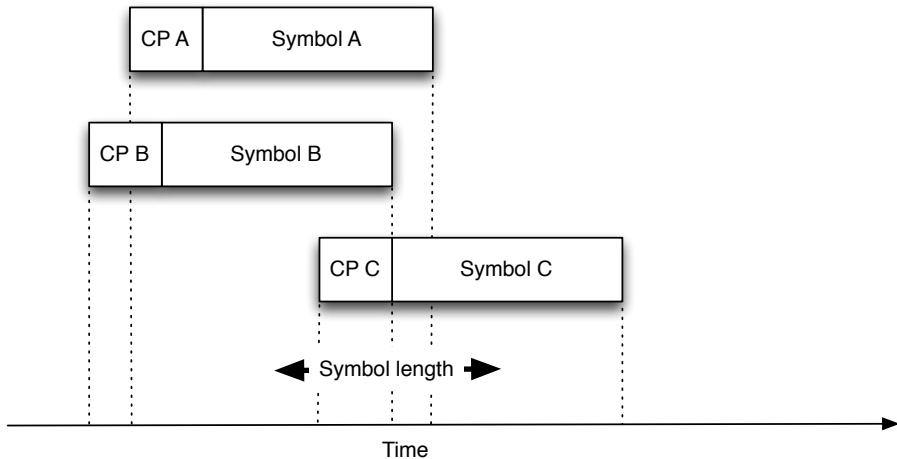
Distributed MIMO systems take a different approach to scaling multiple wireless transmissions: instead of trying to isolate different transmissions in time, space, and/or frequency, they combine them in a way analogous

to the scaling of multiple SISO systems to a single MIMO one. Going back to our example, as we can see in Figure 5.2, we can combine the three two-antenna APs and create a single six-antenna AP which can use zero-forcing to transmit to all three hosts concurrently, without any cross-AP interference.

### 5.1.1 Distributed MIMO System Principles

When building a Distributed MIMO system, we try to create an infrastructure that resembles a single large AP with multiple antennas. Achieving this entails fulfilling three important requirements. First, all APs need access to all the *data packets* that will be transmitted as well as to all the *channel state information* between receiving and transmitting antennas. In the single AP case all these data are available to all transmit chains—one per antenna. In the multiple AP case, satisfying this requirement is relatively easy: modern enterprise deployments usually connect APs to a Gigabit Ethernet backbone fast enough to allow the exchange of all the necessary information.

Second, all APs must synchronize the *timing* of their transmissions. When a receiver tries to decode a symbol, it must convert it to from the time domain to the frequency domain, using the *Discrete Fourier Transform* (DFT) [74]. The DFT, though, must be applied to successive samples that all have the same frequency content. In 802.11, each transmitted symbol is preceded by a cyclic prefix, which is a repetition of the last quarter of its samples in the time domain. By the properties of the DFT, the resulting combination of cyclic prefix and symbol has the same frequency content, so a receiver can choose *any* symbol length's worth of samples over the combination to perform the DFT.



**Figure 5.3:** Receivers need at least one symbol time's worth of overlap to decode a transmission.

The same holds true when we have transmissions from multiple antennas. If, though, there are significant time offsets between the different transmissions, finding a symbol length's worth of samples with the same frequency content might not be possible. In Figure 5.3 we can see one such example. The timing offsets between transmissions A, B, and C are such that they overlap for less than a symbol length. In this case, the combined symbol would not be decodable at the receiver. When all transmissions are from the same AP this is not an issue, since all transmit chains share the same clock, and are fully synchronized. For the multiple AP case, Rahul *et al.* have demonstrated a system, that achieves synchronization with 95th percentile error of less than 20 ns, much less than the 800 ns length of a cyclic prefix [77], so using this scheme suffices to achieve the necessary time synchronization.

The third important requirement for Distributed MIMO systems is carrier frequency synchronization. Since different APs use different oscillators, they have different carrier frequencies. Furthermore, those frequencies show a small drift over time, due to hardware imperfections and temper-

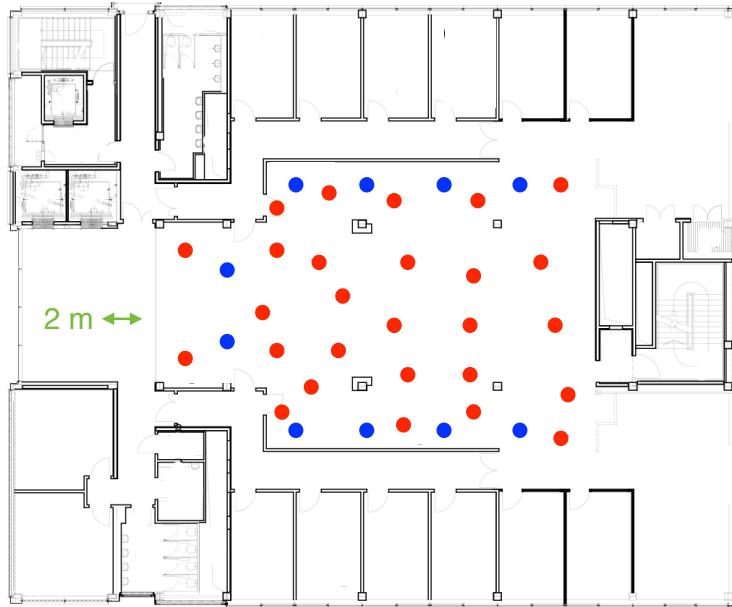
ature fluctuations. Offsets in carrier frequency between different transmitters impact OFDM systems in two important ways. First, CFO causes loss of orthogonality between the subcarriers of an OFDM symbol, leading to *Inter-Carrier Interference* (ICI). The effect of ICI can be significant but in a system like 802.11, which has an inter-carrier spacing of 312.5 kHz, that happens for CFO values above 1 kHz [80]. Second, when using a MIMO technique (*e.g.*, zero-forcing), it is necessary that the relative phases between transmissions from different antennas remain constant for the whole duration of the transmitted frame. This is because CFOs cause a relative rotation of the transmitted symbols in the IQ domain. The existence of a carrier frequency offset between a transmitter and a receiver is normal and easily dealt with at the receiver but when we have multiple transmitters, there are multiple frequency offsets. To overcome this obstacle, Rahul *et al.* propose electing one of the constituent APs of the distributed MIMO system as a leader and having all the rest adjust their carrier frequency to the same value as that of the leader [18]. This way, the receiver can easily compensate for a single carrier frequency offset and the relative rotation of symbols transmitted by different APs remains constant for the duration of a frame.

### 5.1.2 Scaling Wireless Throughput

To study the behavior of Distributed MIMO systems, we implemented one in our lab. Figure 5.4 shows the floor plan of our workspace. The blue dots indicate the placement of pairs of antennas that are members of our virtual AP—ten locations which amounts to 20 antennas in total—while the red dots indicate 22 client positions.<sup>1</sup>

---

<sup>1</sup>Our system is described in detail in Section 5.3.1.

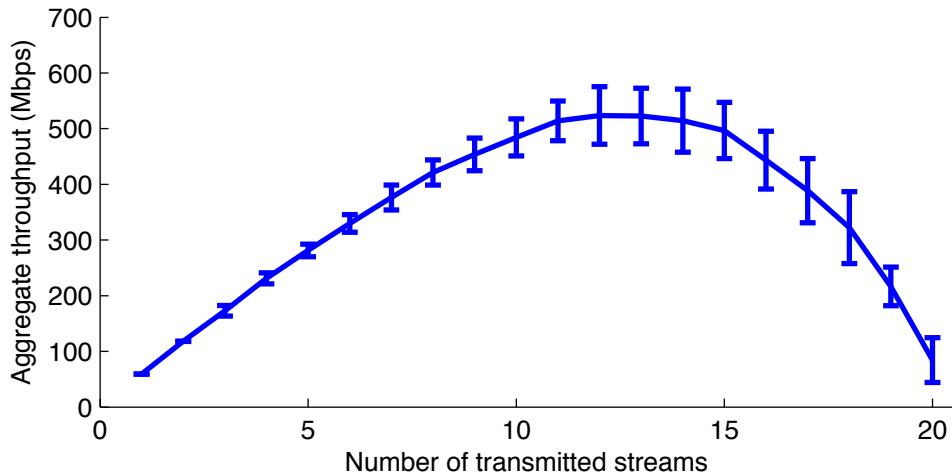


**Figure 5.4:** Floorplan of our system. The blue dots in the periphery show transmitting antenna pairs and the red dots indicate client positions.

Since this Distributed MIMO system is composed of up to 20 antennas, it can transmit up to 20 concurrent streams of information. Ideally, we would like each transmitted stream to achieve the maximum possible throughput and the total throughput of our system to increase linearly as we add receivers.

Unfortunately, in systems with a large number of antennas, as is the case with Distributed and Massive MIMO, increasing the number of antennas beyond some point yields diminishing returns.

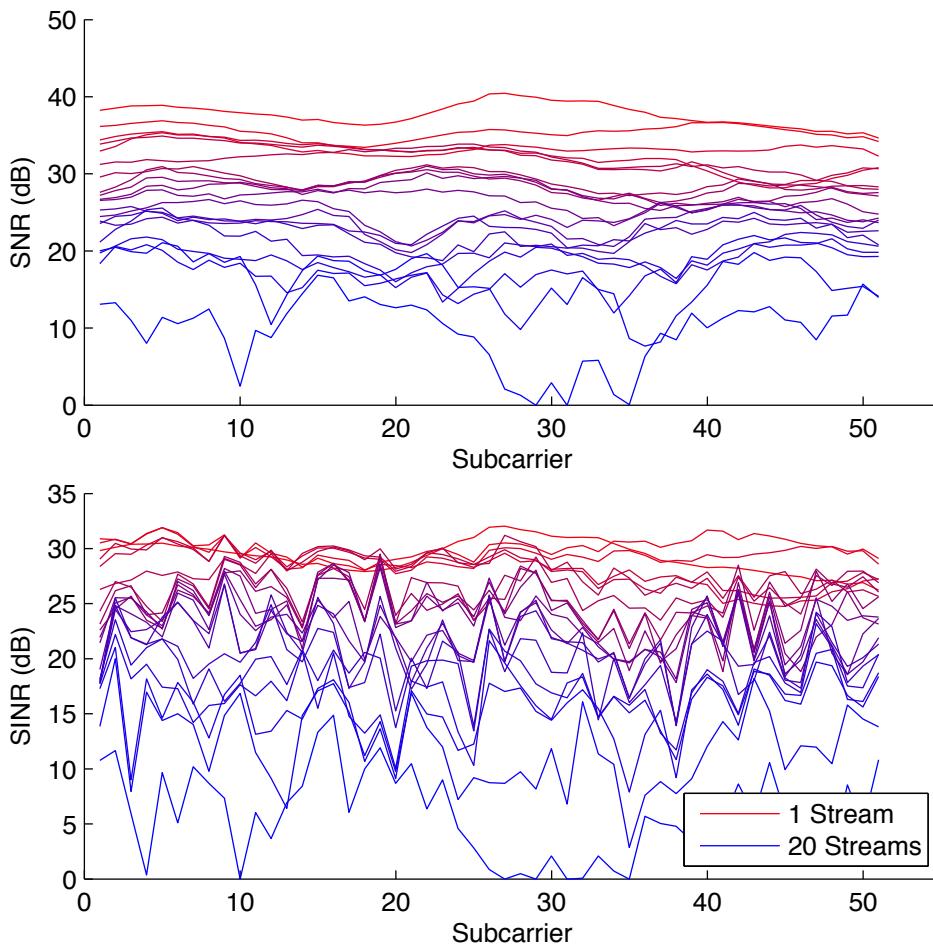
Figure 5.5 shows the aggregate throughput of our system when all 20 antennas are used, for varying numbers of transmitted streams. We initially chose a random permutation of 20 client positions and increased the number of hosts served. Each point represents the mean of 10 such permutations. As we can see, increasing the number of served clients yields a linear increase in throughput for up to about 8 clients. After that point,



**Figure 5.5:** Aggregate throughput of 20 antenna Distributed MIMO system with varying number of transmitted streams. Mean of 10 topologies with the errorbars showing  $\pm 1$  standard deviation.

throughput keeps increasing, albeit with diminishing returns, up to 13 concurrent transmissions. Beyond that point, each additional stream leads to a decrease of aggregate throughput. This behavior is because of a phenomenon called *channel hardening* [45, 47]. As the number of antennas in the system increases, the variance of the resulting channels increases as well. This in turn means that in order to orthogonalize our transmissions, we typically have to sacrifice a lot of the received power to at least some of our clients. Furthermore, even if we use orthogonalization techniques like zero-forcing, phenomena such as noise and hardware inaccuracies lead to increased inter-stream interference [82].

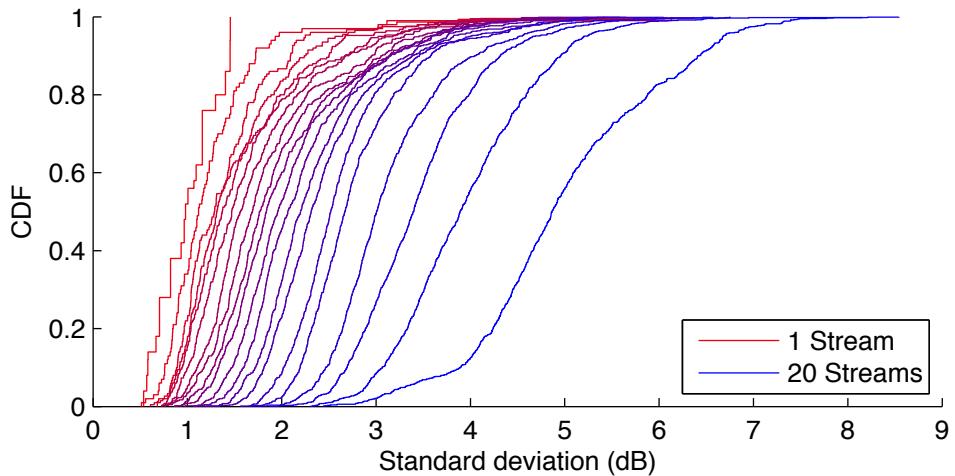
But what is the effect of channel hardening on a *single* stream? Figure 5.6 shows the SNR and SINR of the same received stream when our system uses all its 20 antennas but with increasing number of concurrently transmitted streams. As we can see, increasing the number of transmissions leads to an overall reduction in SNR and SINR, but also to an increase in



**Figure 5.6:** SNR (top) and SINR (bottom) for the same stream as the number of concurrently transmitted streams increases. The SNR values are normalized for transmit power.

the variance of SNR and SINR across subcarriers. It is worth noting that the SNR values in the top graph are normalized by transmit power, so the changes we see are solely due to progressively poorer conditioning and not because we split the total power across more transmitted streams.

Zooming out of a single stream, the macroscopic effect of channel hardening on different streams can be seen in Figure 5.7. Here, each curve shows the cumulative distribution of the standard deviation of SINRs within the same stream, for different topologies as we increase the number of con-



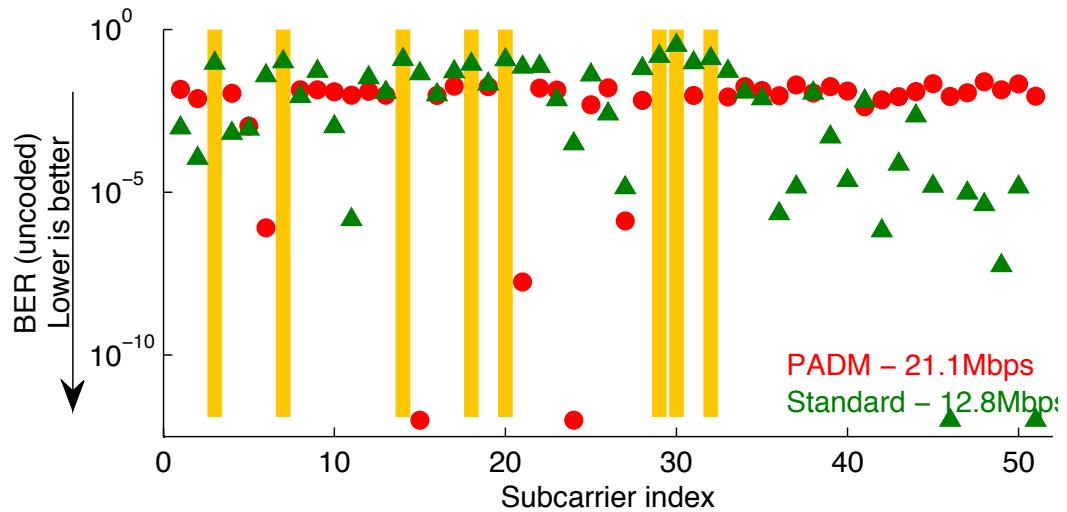
**Figure 5.7:** Standard deviation of SINR within the same stream for 50 topologies as the number of transmitted streams increases.

currently transmitted streams. It is obvious that as the number of streams increases, so does the variability of SINR across the different subcarriers of the same stream.

As we saw in Section 4.1, increased variability of SINR across the subcarriers of a stream can lead to significant reduction in throughput, since 802.11 hosts use a single bitrate across their different subcarriers.

## 5.2 Design

As we saw in the previous section, increasing the number of transmitted streams in a given Distributed MIMO system does not yield a linear increase in aggregate throughput as one might expect. This is due to channel hardening, which also causes an increase in SINR variability across the subcarriers of transmitted streams. This phenomenon resembles the one we observed in the previous chapter, where the combination of multi-stream transmission and residual interference from imperfect nulling causes a similar increase in SINR variability.



**Figure 5.8:** BER per subcarrier without coding when using the same nulling pre-coding matrix and bitrate (26 Mbps) with PADM and without (“Standard”). Vertical bars denote subcarriers that PADM drops.

Since all subcarriers in a given stream use the same modulation and are encoded with a single encoder, a few particularly bad ones can have a disproportionately negative effect on the overall throughput. The EquiSINR algorithm, described in detail in Section 4.2, addresses this issue by dropping particularly bad subcarriers and redistributing the transmitted power across the remaining ones, allowing for higher bitrates and in turn higher throughput. Figure 5.8 illustrates the effect of using PADM on a single stream transmitted by a 20-antenna virtual AP that serves 15 hosts concurrently. As we can see, reducing SINR variability and dropping subcarriers with particularly low SINR allows PADM to transmit using the 26 Mbps bitrate instead of the 13.5 Mbps bitrate used in the baseline case, similarly to what we observed for COPA in Figure 4.10.

Of course, changes in the power allocation in one stream manifest as changes in interference with the remaining ones. To deal with this, EquiSINR works out the per-stream power allocation, combines the results,

calculates the new inter-stream interference, and repeats the process until it converges to a solution.

In *Power Allocation for Distributed MIMO* (PADM), we follow the same process in order to deal with our virtual AP's inter-stream interference. Unlike in COPA, the existence of a fast wired backbone allows us to exchange all the necessary coordination information over the wired backplane connecting the APs, and to employ the lightweight MAC protocol described by Rahul *et al.* [18]. Also, unlike in COPA, considering access strategies is not necessary since it has been shown that Distributed MIMO outperforms standard 802.11 [18, 47, 80].

### 5.3 Evaluation

To evaluate the behavior of a Distributed MIMO system with and without the use of PADM, we built the testbed described in Section 5.3.1. The natural question that arises is: "Does PADM increase the aggregate throughput of a virtual AP and under what conditions?", which we answer in Section 5.3.2. We also examine whether power allocation allows us to increase the number of concurrently served clients (Sections 5.3.3 & 5.3.4). Another interesting question is how much throughput variability exists between concurrently transmitted streams and how that is affected by increasing their number, which we examine in Section 5.3.5. Finally how much aggregate throughput variability exists between different but equally large subsets of clients that are served concurrently by the same virtual AP, and can we take advantage of that variability to increase the mean aggregate throughput by choosing groups of clients to transmit to concurrently (Section 5.3.6)?

### 5.3.1 Experimental Setup

In Figure 5.4, we can see the floorplan of our lab along with the host locations used in our experiments. We used the WARP v2 [83] hardware platform along with a modified version of the WARPLab v7.5 framework, and 20 MHz channels in the 2.4 GHz band. For the transmitting APs, we used five WARP boards which we spread at the edges of our open-plan office. Each WARP board has four antennas, which we divided in two pairs—each pair is marked with a blue dot in Figure 5.4. Antennas in the same pair were placed 25 cm apart and pairs connected to the same WARP board were placed about 4 m apart, in order to obtain the best possible coverage. Due to the limited number of WARP boards available to us, we used a single board as a client and moved it to 22 different locations—indicated with red dots in Figure 5.4—in order to take measurements for several client positions. We then combined these measurements to emulate the behavior of multiple active receivers.

To take channel measurements, our APs transmitted successively to the client, making sure that the first AP to transmit (the *leader*) was the one with the most power delivered to that client. We did so to ensure that our Automatic Gain Control module would choose the correct settings for the receive amplifiers, thus avoiding saturation. Due to platform constraints, we did not implement the tight synchronization described by Rahul *et al.* [77], but our transmitters were loosely synchronized (within tens of microseconds) and we then used the Schmidl-Cox algorithm [84] for timing synchronization on each of the received transmissions. For each of the client positions, we also took measurements between the leader and follower APs and each follower used this measurement to calculate its carrier frequency offset to the leader. These measurements happened in quick suc-

cession ( $\sim 500\mu\text{s}$ ) to make sure that the CFO remains the same between the two [80]. We then used these estimates to correct the CFO of each AP at the client, after the approach taken by MegaMIMO [18].

We initially measured the channel in order to obtain CSI information and then performed one more measurement, which we used in order to emulate an actual transmission by applying the relevant precoding matrices. We then used the coded effective BER of the channel [78, 7] to estimate the throughput of a stream that uses 4 ms long transmit opportunities (TXOP) and 802.11n bitrates.

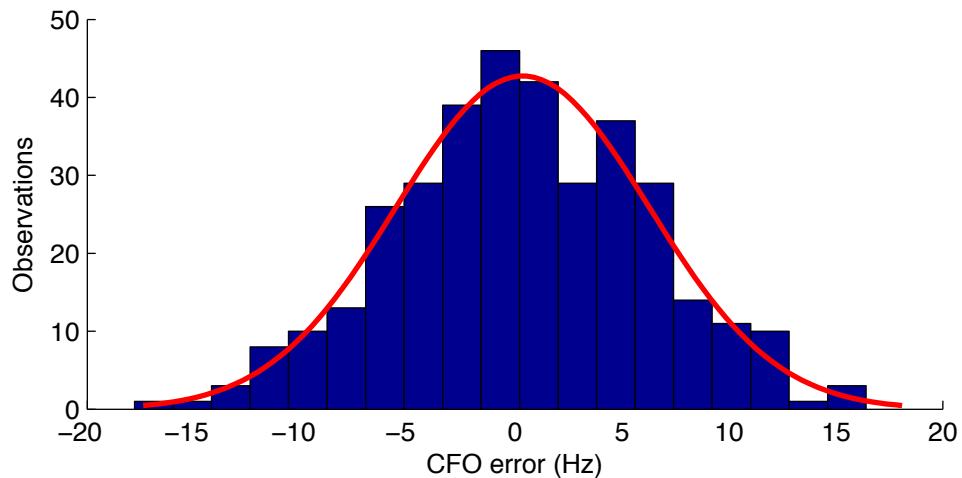


Figure 5.9: Carrier frequency estimation error.

To get a fine-grained CFO estimate, we did a linear fit of the per-subcarrier phase for each of the successive symbols of the 802.11 preamble, weighting each phase by its power, so phases from subcarriers with more power contribute more to the estimate, similarly to the N symbol MRC estimator described by Murphy [80]. We then averaged the estimates for the successive symbols taking into account the time difference between them, since later symbols will show a greater phase offset due to CFO. As a sanity check for our CFO estimator, during the transmission from the leader to

the follower APs, instead of sending just a single 802.11 preamble, we sent a further 40 repetitions of its Long Training Symbol sequence and used all the measurements to compute a ground-truth CFO. In Figure 5.9 we can see the distribution of the carrier frequency estimation error, which follows the normal distribution with a mean of 0.35 Hz and a standard deviation of 5.7 Hz, well below the 50 Hz value that is necessary to avoid distortions according to Murphy [80] and in line with the standard deviation value of 5.4 Hz reported in the same work. This value translates to  $2.1 \text{ ms} * 5.7 \text{ Hz} * 2\pi = 0.075 \text{ rad}$  for a maximum sized packet at the lowest bitrate, or introduction of noise at the level of -29 dB in the case of a virtual AP with two transmitters of equal received power at the client.

<b>Algorithm 5.1:</b> Picking AP antennas and client locations.
---

```

1 Clients: 20 random client positions;
2 AP_antennas: empty;
3 Remaining_antennas: all AP antennas;
4 for Client in Clients do
5   Pick strongest AP antenna from remaining ones;
6   Add AP antenna to AP_antennas;
7   Remove AP antenna from Remaining_antennas;
8 end
9 for i in range(1, 20) do
10   AP_antenna_subset: first i AP antennas from AP_antennas;
11   for j in range(1,i) do
12     Client_subset: first j clients from Clients;
13     Calculate throughput;
14   end
15 end
```

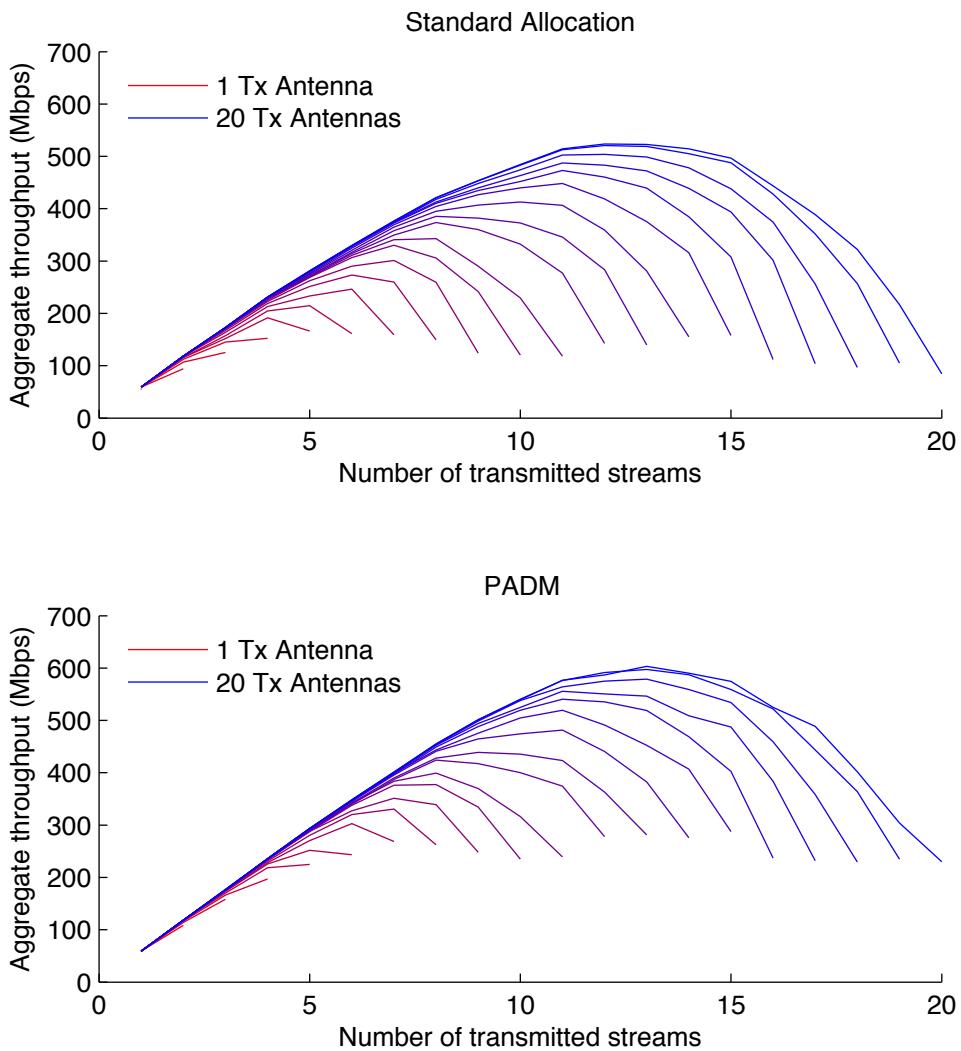
To evaluate the performance of our Distributed MIMO system, we used our testbed under different AP and client configurations. To pick these different topologies we used the procedure outlined in Algorithm 5.1. We did so to ensure that whenever we added an extra antenna to our virtual AP, that

antenna could deliver some amount of power to at least one of our clients, thus avoiding adding AP antennas that cannot contribute capacity-wise to our system and only exacerbating the effects of channel hardening. Since we had five 4-antenna APs available, we could create virtual APs that used 1 to 20 antennas, and consequently could serve 1 to 20 clients. We initially chose random, ordered permutations of 20 client positions. For each permutation, we calculated which AP antenna had the strongest signal to the first client and added it to our virtual AP. Then, from the remaining AP antennas, we picked the one with the strongest signal to the second client. We repeated this process for all remaining clients, thus creating a permutation of the 20 AP antennas. Based on these two ordered permutations of client and AP antenna positions, we calculated the throughput of all valid client-AP antenna combinations.

### 5.3.2 Overall Throughput Improvement

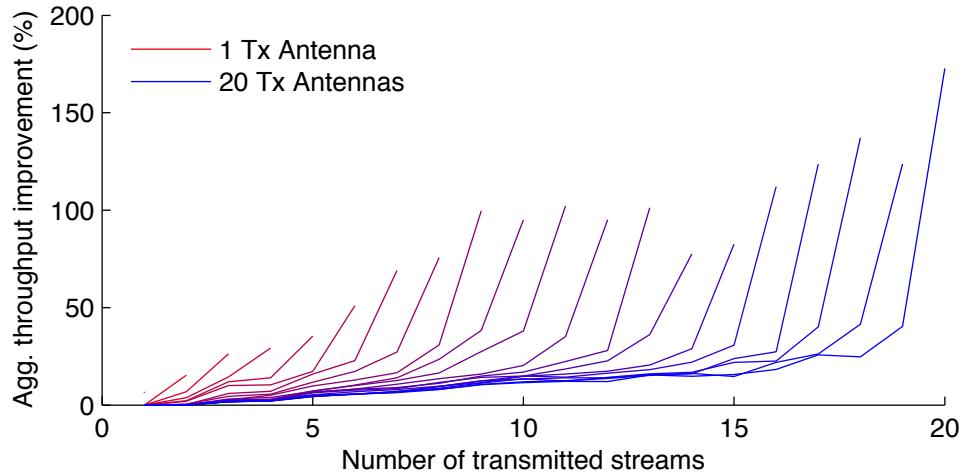
As we saw in Section 5.1, increasing the number of transmitted streams for a given number of transmit antennas increases SINR variability, which in turn can have an adverse effect on the aggregate throughput of a Distributed MIMO system. This phenomenon is similar to the one observed in Chapter 4, and thus seems ripe for mitigation using the EquiSINR algorithm.

At the top of Figure 5.10, we can see the performance of our Distributed MIMO system using zero-forcing with equal power allocation across subcarriers. Each curve shows the performance of a Distributed MIMO system with a set number of antennas, as we vary the number of transmitted streams, with each point showing the mean of the aggregate throughput across 10 topologies. We can see that as long as the ratio of transmitted



**Figure 5.10:** Aggregate throughput for Distributed MIMO systems with different number of transmit antennas for varying number of transmitted streams.

streams to virtual AP antennas  $b$  is kept high—between 2 and 1.5—adding an extra transmitted stream leads to a linear throughput increase, as reported by Rahul *et al.* [18]. As  $b$  approaches 1.5, adding extra streams offers diminishing returns in throughput. When  $b$  falls below 1.5, adding extra streams reduces throughput due to channel hardening, and in accordance to what was observed by Yang *et al.* [47].

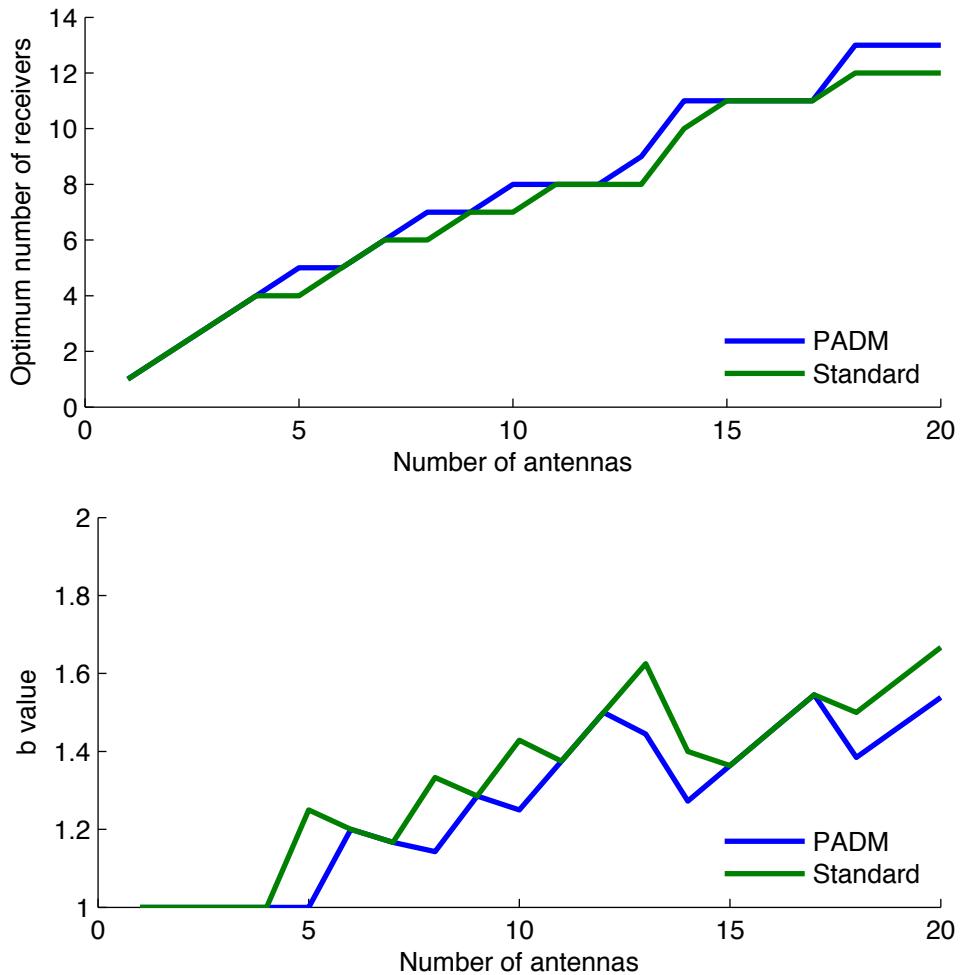


**Figure 5.11:** Throughput improvement when using PADM versus standard power allocation.

The bottom of Figure 5.10 shows the performance of the same topologies when using PADM. We can see that the curves have shifted upwards, indicating a throughput improvement, as becomes clearer in the next figure, Figure 5.11. There we can see the throughput improvement quantified for all the different combinations of transmit antennas and transmitted streams. When  $b$  is between 2 and 1.5, the throughput improvement is more modest, since for such ratios the resulting channels are still relatively uncorrelated, yet significant, since even in these cases there is some channel variation. As  $b$  is reduced further, though, channel hardening starts to dominate the behavior of our system, SINR variability becomes important, and PADM yields significant throughput improvements.

### 5.3.3 Choosing the Optimum Number of Clients

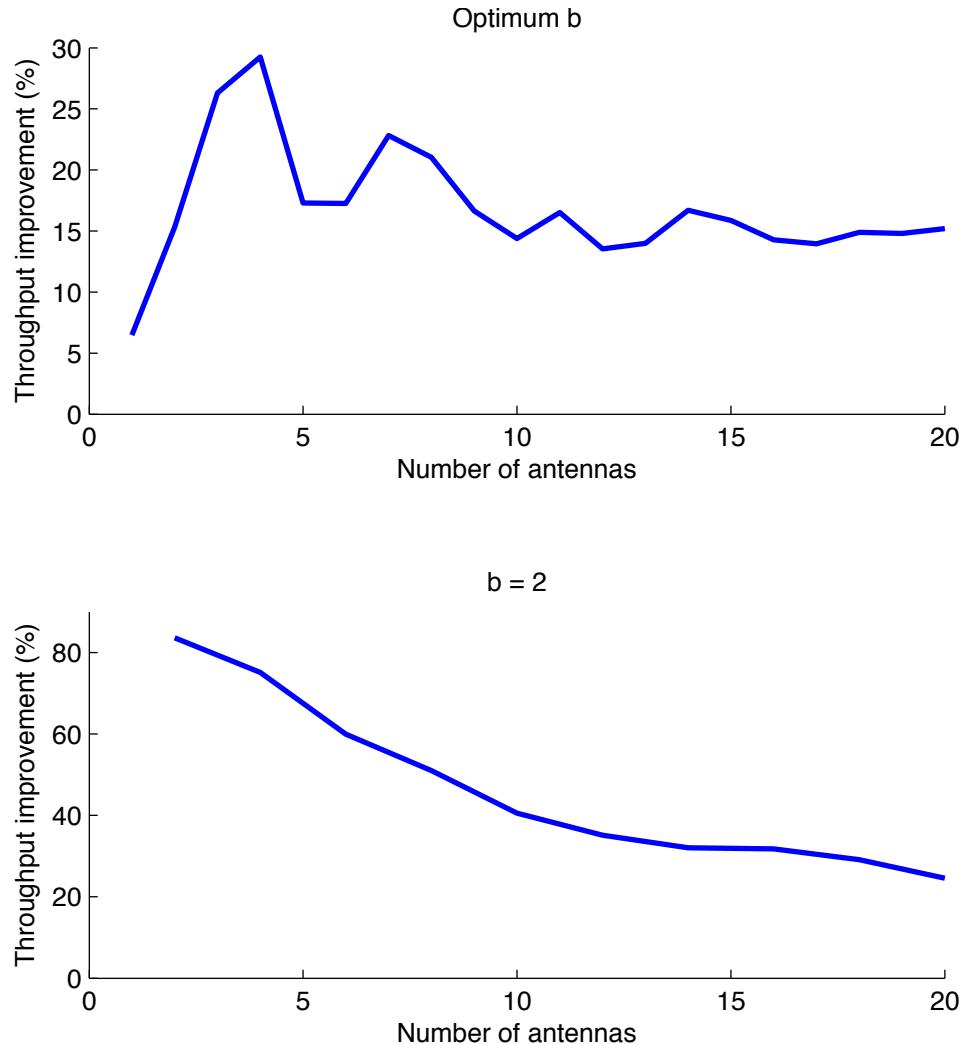
Figure 5.10 shows that for a given number of virtual AP antennas, there exists a number of transmitted streams that maximizes the aggregate throughput. At the top of Figure 5.12 we can see what that number is



**Figure 5.12:** Number of clients (top) and value of  $b$  (bottom) at maximum aggregate throughput as a function of number of antennas of the virtual AP.

in the standard case and when using PADM, while at the bottom we can see the value of  $b$  for the optimum number of transmitted streams. As we can see, for virtual APs with more than 4 antennas, in 50% of the cases PADM allows us to serve one extra client while still increasing aggregate throughput. On the other hand, for very small numbers of antennas, we can add a concurrently transmitted stream for each available antenna in our virtual AP.

### 5.3.4 Improvement at the Maximum



**Figure 5.13:** Improvement when using PADM with the optimal number of streams against standard power allocation with optimal number of streams (top) and  $b = 2$  (bottom).

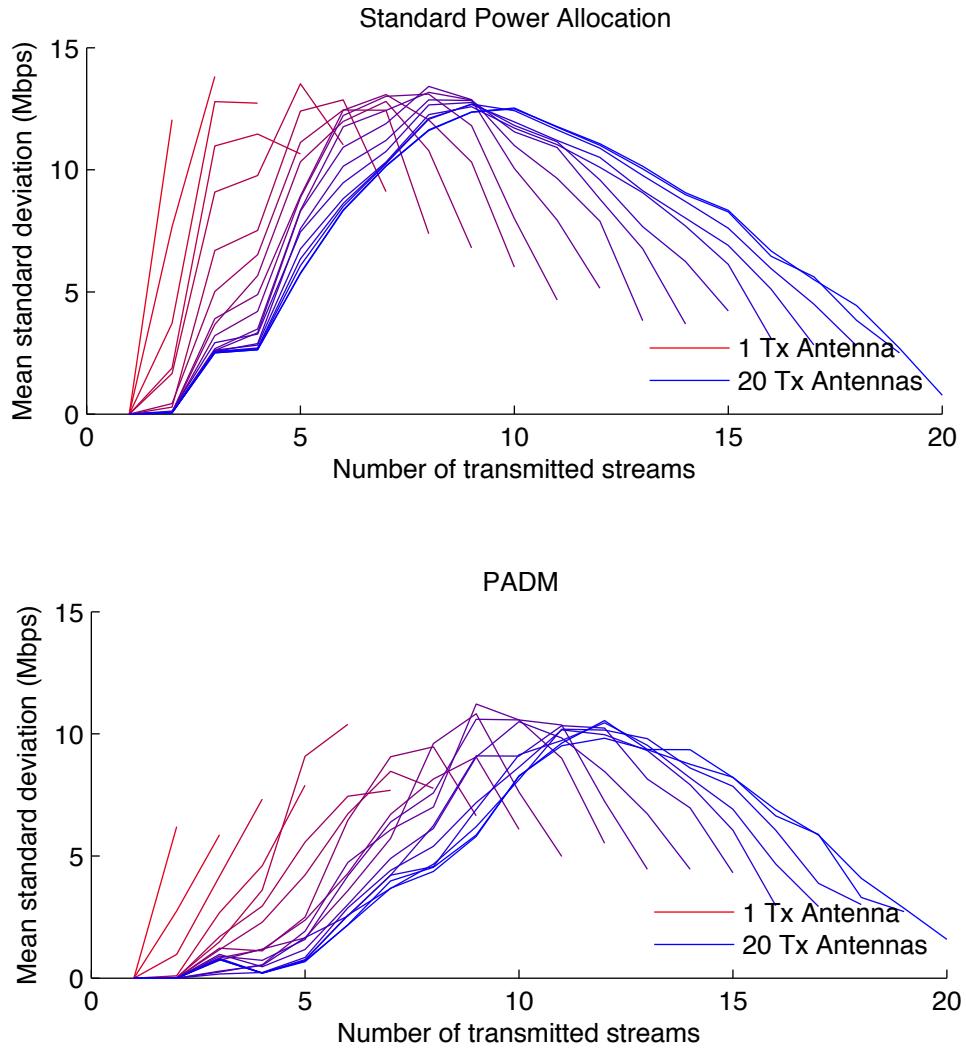
In the previous two subsections, we saw that we can improve the aggregate throughput of a Distributed MIMO system using PADM, but that it is also important to choose the number of concurrently transmitted streams correctly. One could be conservative and choose a value of 2 for  $b$  as Rahul *et al.* imply [18], but in this case we sacrifice throughput in favor of linear

scaling. If, on the other hand, we try to be more aggressive and operate closer to  $b = 1.5$ , we can serve more clients concurrently and maximize aggregate throughput, but we should be careful as increasing the number of transmitted streams beyond that point can reduce throughput.

At the top of Figure 5.13, we can see the improvement in aggregate throughput when using PADM over standard power allocation, while we transmit the optimum number of streams for each scheme. We can see that the greatest improvement is for three and four-antenna virtual APs. In these cases, there is a lot of room for improvement because the optimal value for  $b$  when using standard power allocation is 1, which means that our virtual AP uses all the available degrees of freedom and as a result the transmitted streams show higher SINR variability. As the number of antennas increases, the improvement stabilizes around 16%. At the bottom of the same figure we can see the throughput improvement when we use PADM and the optimum number of transmitted streams versus using standard power allocation and  $b = 2$ . In this case, for small numbers of antennas we get a significant throughput improvement because as we saw, using PADM allows us to operate close to  $b = 1$ . As the number of antennas grows, the improvement converges to 30%, since by using PADM we can transmit 30% more streams for the same number of transmit antennas while still maintaining a high throughput for each of the transmitted streams.

### **5.3.5 Throughput Variability Across Streams**

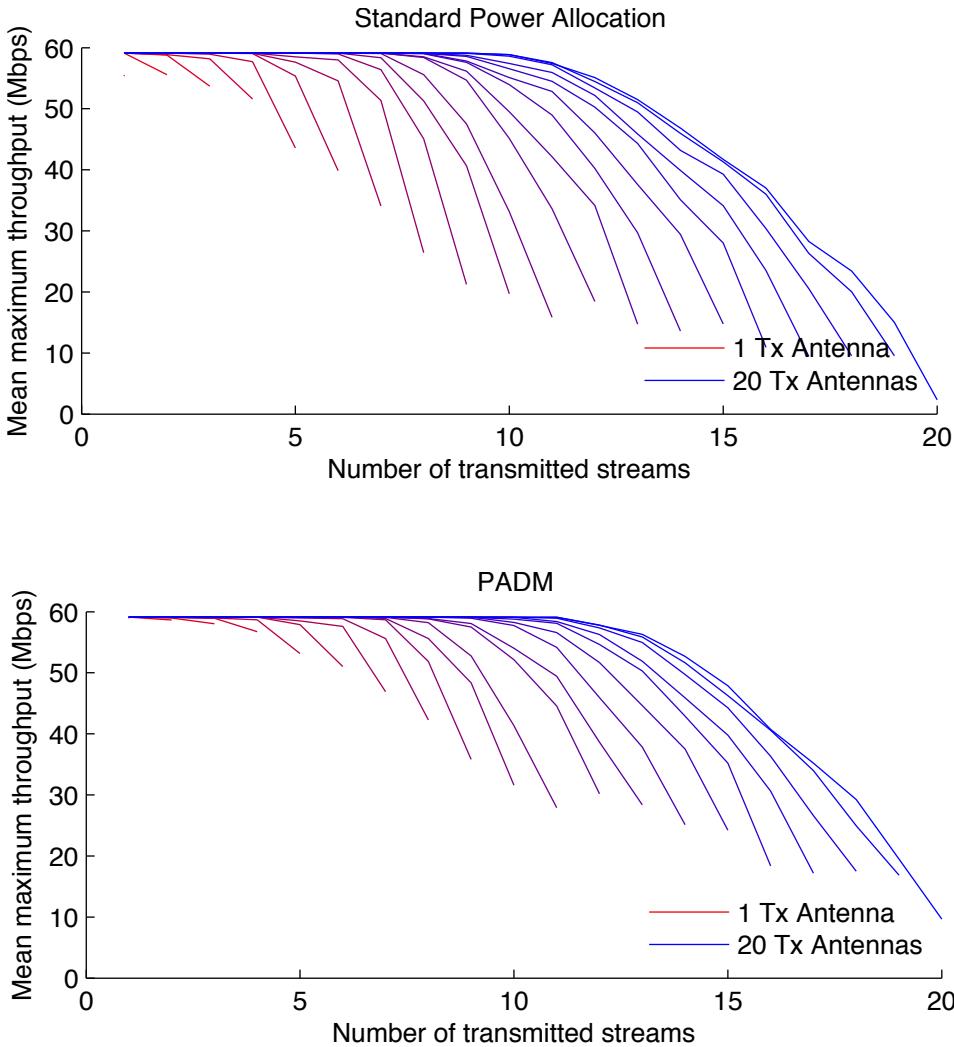
As we saw earlier, increasing the number of streams transmitted by a virtual AP increases the SINR variability and the correlation between channels, resulting in variation across the throughputs achieved by each of the concurrently transmitted streams. The evolution of this variation can be



**Figure 5.14:** Mean of standard deviation of throughput across streams for standard power allocation (top) and PADM (bottom).

seen at the top of Figure 5.14, which shows the mean standard deviation across ten different topologies for virtual APs with different number of antennas as we vary the number of transmitted streams.

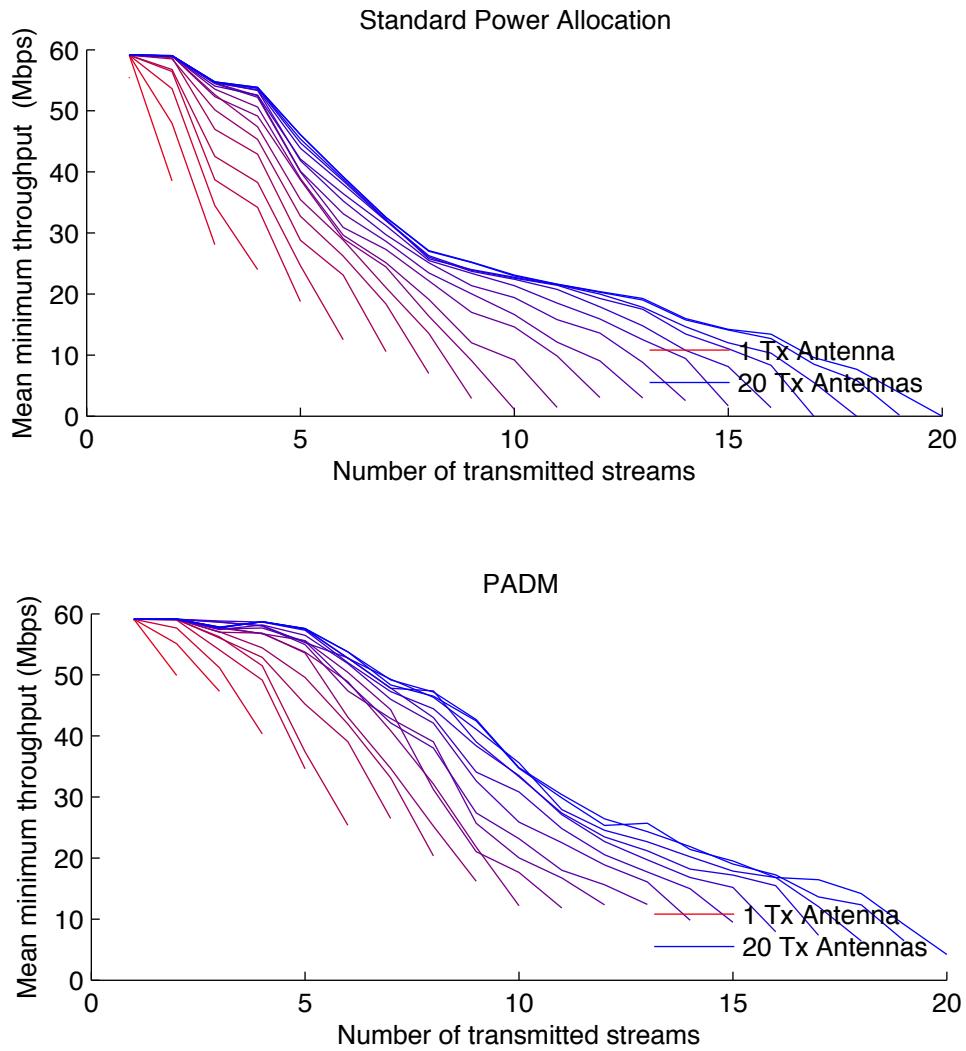
As we can see, the shape of the standard deviation follows a bell-like curve. Initially the standard deviation increases. We can get a better intuition about why this happens by looking at Figure 5.15, which shows the mean



**Figure 5.15:** Mean of maximum of throughput across streams for standard power allocation (top) and PADM (bottom).

of the maximum throughput stream, and Figure 5.16, which shows the mean of the minimum throughput stream. In this first phase, the highest throughput stream always achieves the maximum possible throughput while adding extra transmitted streams results in a decrease in the throughput of the stream with the minimum throughput. Beyond some point, though, inter-stream interference starts to affect even the maximum throughput stream, while the minimum throughput streams start to settle

to either one of the lowest bitrates more resilient to interference or degrade to zero throughput. At that point the standard deviation begins to decrease.

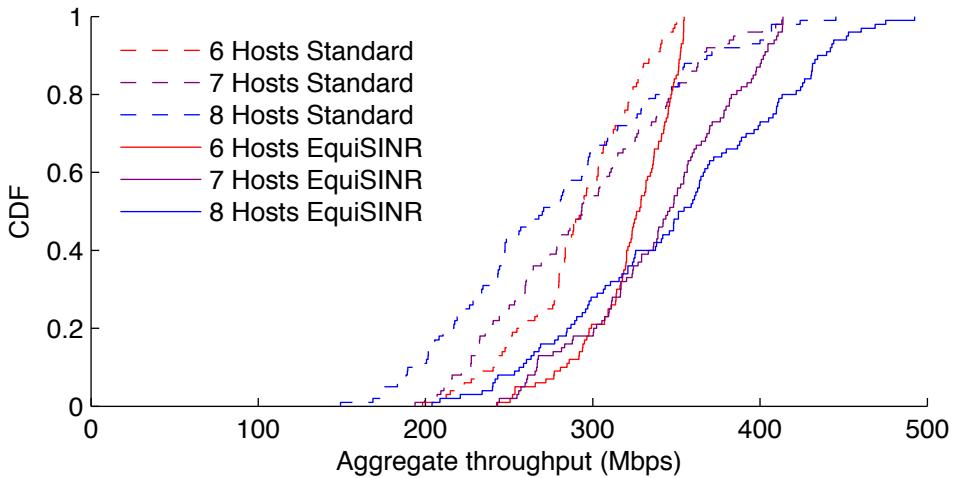


**Figure 5.16:** Mean of minimum of throughput across streams for standard power allocation (top) and PADM (bottom).

Using PADM allows us to delay the degradation in throughput for the maximum-throughput stream, while increasing the throughput of the minimum-throughput one, resulting in reduced overall standard deviation

and a shift of its maximum towards higher numbers of concurrently transmitted streams.

### 5.3.6 Variability Across Topologies and Scheduling

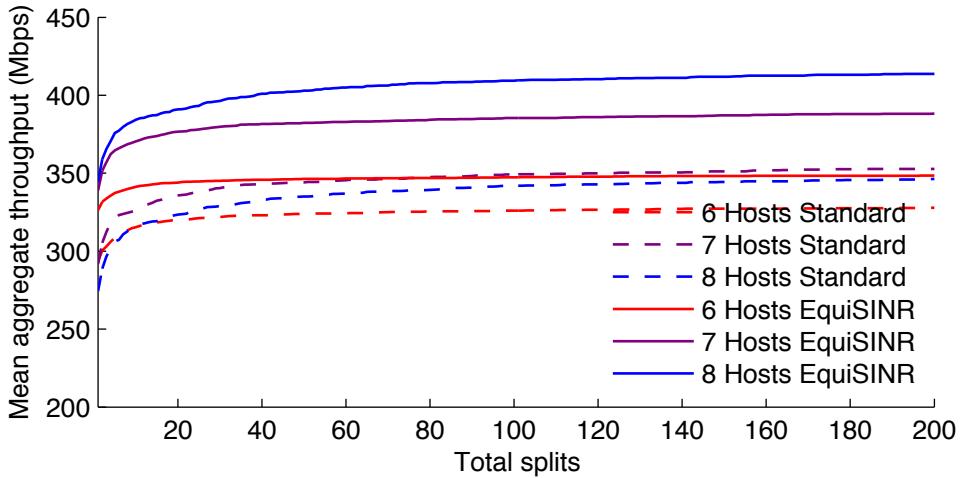


**Figure 5.17:** Aggregate throughput distribution across 100 random client topologies for the same 10-antenna virtual AP.

Since the achieved throughput of a Distributed MIMO system depends on the channels between transmit and receive antennas and the severity of channel hardening, we would expect that the same virtual AP would achieve different aggregate throughputs depending on the exact sets of hosts it serves. Indeed, in Figure 5.5, we can see that the standard deviation of the aggregate throughput for our 20-antenna virtual AP increases and becomes significant as the  $b$  value of our system falls below 2. To get a better sense of the distribution of aggregate throughputs across different topologies, we created a virtual AP with ten transmit antennas—one antenna for each blue dot in Figure 5.4, so that we get the best possible coverage of our office—and chose 100 random distinct subsets of either six, seven, or eight clients. We chose these values because these represent the neighborhood where PADM achieves maximum aggregate throughput ac-

cording to Figure 5.10. We then calculated the aggregate throughput of our system both with standard and with PADM power allocation, and plotted their CDFs in Figure 5.17. As expected, using PADM yields significant throughput improvements across the board. Interestingly, as the number of served hosts increases, the range of observed values increases too. As we increase the number of clients, topologies with low correlation add one more stream with little or no effect on the existing ones, while in topologies with highly correlated channels the throughput of the original streams is adversely affected, leading to a divergence between “good” and “bad” topologies.

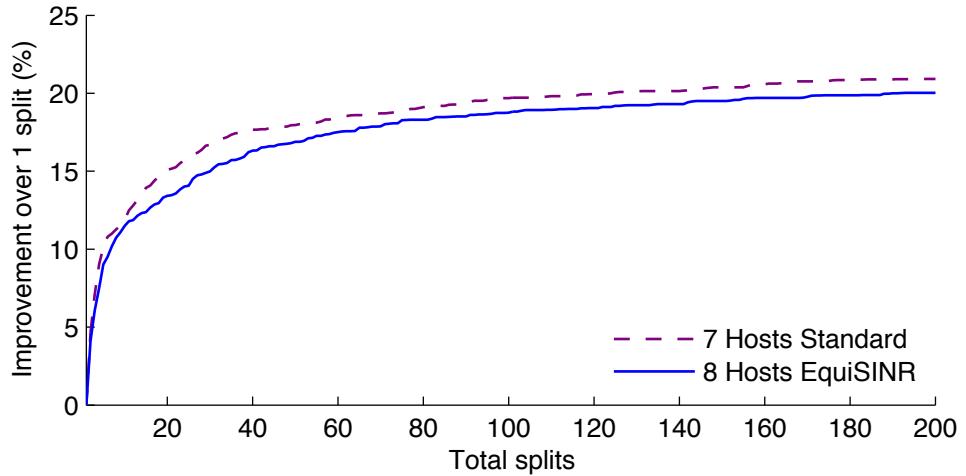
Of course, a ten-antenna virtual AP used in an office environment like ours would probably have more than 8 hosts associated with it at any given moment, and consequently have packets for several hosts in its queue. If the total destinations are fewer than the optimal value of 7 or 8 (depending on whether we use PADM or not), our transmit strategy is obvious: transmit to all destinations concurrently. If the number of destinations is between the optimal value and just below two times the optimal value, the optimal transmit strategy is also straightforward: split the hosts in two groups and transmit consecutively. As the number of destinations reaches twice that of the optimal, though, we reach an interesting point where although splitting the hosts in two groups is still optimal, the shapes of the CDFs in Figure 5.17 raise an interesting question: does *how* we split hosts into groups make a difference? Specifically, since “good” and “bad” topologies exist, given a set of hosts, can we split them in two groups that are on average better than other random splits, yielding higher total throughput, in the spirit of “The power of two choices” [85] or IAC [9]?



**Figure 5.18:** Aggregate throughput when using the best out of a set number of permutations. Notice that y-values start at 200Mbps.

To answer this question we assumed that our ten-antenna virtual AP had packets for 12, 14, or 16 client topologies in its queue. We split those clients randomly into two groups and calculated the average aggregate throughput of our system, assuming consecutive transmissions. We repeated the random split 199 more times and repeated the experiment for 100 topologies (*i.e.*, distinct combinations of client positions). We then calculated the average throughput when using the best out of the 200 random splits and plotted the results in Figure 5.18. As we can see, increasing the number of splits we consider can yield a handsome increase in aggregate throughput.

In Figure 5.19 we can see the improvement expressed as a percentage for the optimal number of concurrently served clients under each scheme. As we can see, taking the best of two choices yields a 5% throughput improvement for standard power allocation and a 4% improvement for PADM. This gain can be increased all the way to 20% when choosing the best out of 200



**Figure 5.19:** Aggregate throughput improvement when using the best out of a set number of splits.

random splits, although after about 50 the gain reaches 17%, and beyond that point the returns diminish.

## 5.4 Discussion

Distributed MIMO is a promising technique that can help wireless systems scale their delivered throughput with the number of installed APs, allowing them to serve more clients concurrently. As we saw in Section 5.1, though, for a given number of antennas, increasing the number of transmitted streams does not result in a linear increase in aggregate throughput. Increased channel correlation results in a reduction of the total received power as well as an increase of the SINR variation across the subcarriers of the same stream, limiting the achieved throughput.

PADM allows us to abandon subcarriers that have disproportionately negative effect on the throughput of a stream, and reallocates excess power from subcarriers that don't need it to ones that can significantly benefit from it. PADM thus allows us use of higher modulation and coding rate combina-

tions, and results in an overall throughput improvement. At the same time, remedying SINR variation also results in a decrease in the throughput variation *across streams*, allowing PADM to deliver balanced performance to the receiving hosts. Another important dimension of the problem is choosing the optimal number of concurrently transmitted streams, and finally, under heavy loads, correct groupings of clients for concurrent transmission can yield a further increase in throughput.

## Chapter 6

# Conclusion

The prevalence and importance of wireless networking are clear. From workstations and personal laptops to mobile handsets and tiny sensors, devices benefit immensely in terms of usability from a lack of tethering. At the same time, the sheer volume of connected hosts combined with the throughput demands of applications pose a hard problem, since all these devices need to communicate over the same limited spectrum. Furthermore, future demand is expected to increase, as people already envision wireless communications for traditionally wired applications, like datacenter communications and video transmission from a set-top box to a television. Even though there have been significant performance advances over the years, new applications always emerge eager to grab every sliver of throughput available.

Interference mitigation has been an active research area, but as we saw in Chapter 2, much of the prior work did not take into account important aspects of typical wireless systems. As we then saw in Section 4.3 of Chapter 4, such naive applications of these techniques can lead to surprising results, such as interference nulling's yielding less throughput than

CSMA/CA. The contribution of this work revolves around applying theoretical principles to mitigate interference, but doing so in such a way as to take into account the limitations of practical systems.

In Chapter 3 we presented *Cone of Silence*, a system that allows multi-antenna devices with simple attenuators and phase shifters and a single wireless transceiver to take advantage of the increased diversity due to their multiple antennas and improve receive throughput, both when in a quiet setting, but most importantly, in the presence of wireless interference. *SamplePhase*, our measurement algorithm, provides robust channel estimates that can be used for receive beamforming and yields up to 89% throughput improvement over omnidirectional and up to 219% over static directional reception. *Silencer* builds upon SamplePhase, and allows receivers to improve their throughput when there are (potentially multiple) concurrent transmissions, even in adjacent channels, and provides improvement of up to 1013% over omnidirectional and up to 222% over static directional reception. Of course, the type of APs used in CoS would be considered outdated by today's standards and given the option one should use more capable hardware, like that used in COPA. Nonetheless, early generations of wireless technologies use simpler hardware: current 802.11ad hardware in the 60 GHz band for instance does not use separate receiver/transmitter chains for each antenna due to space and cost constraints [86]. The techniques used in CoS could be useful in early hardware platforms for future wireless systems.

Multiple antennas are not only beneficial to receivers but also to transmitters, as they allow them to increase the power they deliver to their intended receivers and to reduce it at unintended ones, using transmit beamforming and nulling, respectively. In Chapter 4, we applied these techniques

atop Software Defined Radios that allow us to fully manipulate the waveform of the transmitted signal. This additional capability is particularly interesting in systems that use Orthogonal Frequency Division Multiplexing. Such systems divide the spectrum available to them into orthogonal subchannels, called subcarriers, and transmit information on each of them in parallel. SDR allows us to manipulate the signal sent or received on each individual subcarrier. This is important because different subcarriers can experience substantially different propagation conditions, while at the same time, data sent on all subcarriers are sent at the same bitrate by the same encoder and decoder chain. Furthermore, phenomena like strong correlation of the channels between different antenna pairs and channel noise, but also importantly, the accuracy limits of practical, commercially viable hardware, result in imperfect beamforming and nulling, yielding residual interference and reduction of the total power at the intended receiver due to overall power constraints. These in turn exacerbate the variability of SINR of different subcarriers, often resulting in a few of them having a disproportionately negative effect on overall throughput.

*Cooperative Power Allocation* takes these phenomena into account and allows neighboring transmitters to identify opportunities to increase their throughput. It does so by allowing them to cooperate in the allocation of transmit power across subcarriers, but also by letting them selectively drop particularly bad ones. Our results show that while “naive” nulling underperforms traditional 802.11n in 83% of the topologies we tried, nulling with COPA outperforms 802.11n in 76% of the same topologies. At the same time, COPA does not insist on concurrent transmissions in topologies where such a strategy does not perform well, and falls back to CSMA-like operation that avoids interference altogether. Doing so allows an in-

crease of aggregate throughput of 17% in the  $4 \times 2$  antenna scenario. COPA also offers a variant that instead of emphasizing aggregate performance, allows hosts to avoid transmitting concurrently if such a strategy has an adverse effect on their own throughput. In those cases, COPA reverts to a CSMA-like access strategy more often, but still yields a 19% throughput improvement on average.

In Chapters 3 and 4 we concentrated on small, “chaotic” wireless deployments that are typical in urban domestic and SME environments. Nonetheless, large wireless networks are used in a corporate setting as well. In this case, multiple APs provide wireless connectivity to multiple hosts, and *Distributed or Cooperative MIMO* provides a promising prospect for scaling aggregate throughput. In Distributed MIMO, several APs interconnected with a fast, wired backbone network are tightly coordinated to form a *virtual AP* that can act as a traditional AP that uses multi-user MIMO techniques to concurrently transmit several streams of information to multiple clients.

Unfortunately, Distributed MIMO systems are not immune to interference—this time in the form of *inter-stream interference*—and subcarrier variability. As we saw in Section 5.1, increasing the number of concurrently served clients for a given virtual AP leads to an increase in SINR variability across the subcarriers of the transmitted streams. PADM applies subcarrier selection and power allocation to such systems, and by reducing variability and dropping particularly bad subcarriers, allows streams to deliver 16-30% higher throughput. We also show that further gains can be obtained by correctly selecting which streams should be transmitted concurrently. Such an approach can yield a further 20% throughput improvement in the case of a ten-antenna virtual AP. Although previous evaluations of Distributed

MIMO systems generally have assumed that all transmitters within range are part of the *same* virtual AP, that need not be the case. That is, there will exist settings where there is uncontrolled interference. In such scenarios, it may be beneficial to combine PADM with COPA, allowing multiple Distributed MIMO (or standard) networks to coexist in proximity.

All of our experiments took place on the 7th floor of UCL’s Malet Place Engineering Building. Although we took great care to use topologies that varied as much as possible, results in buildings of different construction or between buildings may look different. In those cases, the major differentiating factor would be a possibly greater attenuation of signals between hosts in different networks, resulting in lower cross-interference. As we saw in Section 4.3.4, though, lower cross-interference changes results quantitatively but not qualitatively. On the other hand, in all our experiments the hosts were static. Applying our methods to mobile scenarios is not straightforward. CoS requires a measurement phase on the order of seconds, while COPA and PADM assume a coherence time at the scale of tens of milliseconds. The channel between fast-moving hosts, though, changes quickly. The measurement and MAC overheads of our systems may be prohibitively high in such settings. In our work we have already shown methods for decreasing the size of Channel State Information, such as compression, and hinted at others, such as quantization. Ideas from cellular networks, such as mobility modeling and trajectory projection [87], may prove useful in modeling the way the channel changes over time, and thus reduce the necessity for frequent dissemination of CSI.

Finally, we assumed in our experiments that transmitters always have a backlog of data to send to their receivers. Recent developments such as the *Internet-of-Things* (IoT) create environments with many highly intermittent,

low-bitrate sources/destinations of traffic. Such scenarios need a different approach to resource allocation, possibly with several hosts being served concurrently, each one assigned just enough resources (*e.g.*, spectrum or time) to cover its low bitrate requirements.

## 6.1 Future Work and Closing Remarks

The main theme of this work is building systems that bring theoretical ideas to life while taking into account the limitations posed by the use of practical hardware in realistic environments. One could ask whether as hardware improves over time, the techniques described in the previous chapters will continue to be needed. I believe they will remain of practical value for some time. For instance, while standard 802.11 equipment uses transmit power amplifiers with 3% EVM, some newer 802.11ac hosts that implement the optional 256-QAM modulation need to use improved power amplifiers with 1.5% EVM [2]. For such hosts, the residual interference in Chapter 4 or the inter-stream interference in Chapter 5 would be reduced by about 3 dB. Nonetheless, although things would surely improve for legacy modulations, when using 256-QAM, the aforementioned effects would be just as detrimental as they are now for 64-QAM.

One the other hand, more capable hardware, in the guise of more gates, and hence processing power, can help dealing with analog imperfections. Optimizing concurrent multi-stream transmissions both in terms of power allocation, the precoding matrices used, and the way that the available spectrum is allocated to clients is an NP-hard problem as we saw in Chapter 2. Methods such as non-linear optimization combined with transmit techniques such as Dirty Paper Coding [88], and receive structures such as Maximum Likelihood filters [10], all of which are computationally ex-

pensive, can yield further performance improvements. Also, as we saw in Section 5.3.6, scheduling decisions when serving multiple clients can bring significant throughput improvements. Doing so with exhaustive search is of course computationally expensive, but figuring out new, more efficient analytical methods along with faster 802.11 chipsets might make such scheduling tractable. Moreover, in Section 4.3 we saw other ways that more logic gates can help, such as through the use of multiple encoders and decoders or the adoption of more computationally intensive power allocation schemes, like mercury/waterfilling.

On a different front, the prospect of taking into account the specific application demands of each host opens another rich research area. For instance, a client that is in the middle of a Voice-Over-IP call has strict delay requirements but low throughput demands. To such a client, a transmitter could choose to send frames often but without allocating many resources, namely subcarriers. Therefore, the unused capacity could be used for increasing the throughput of other streams or to serve one more client with similarly low throughput demands. Another interesting possibility is combining COPA with full duplex radios [89]. Doing so would dramatically reduce the overhead of the MAC layer and could potentially even allow us to use COPA under high-mobility conditions, since clients could instantly inform transmitters about the latest channel state information.

Of course, future work should be grounded in the systems approach that underlies this thesis. Real-world deployments often differ significantly from theoretical models. Furthermore, commodity hardware is always used at the limit of its capabilities—in fact, this exact choice makes it economically viable—which naturally requires taking the contour of this limit into account. Doing so will allow us to provide users with the capacity they

need to create new applications, some of whose transformational potential we may not yet anticipate.

## Appendix A

# Mathematical Supplement

Let's consider a multi-antenna host (Host T) that transmits a signal towards an intended receiver (Host A) while trying to null its transmission towards an unintended receiver (Host B). Let  $H_A$  be the matrix describing the channel between T and A, and  $H_B$  the matrix describing the channel matrix between T and B.

To null towards B, T calculates the *null space* or *kernel* of  $H_B$ :

$$N_B = \text{Null}(H_B) \quad (\text{A.1})$$

where  $H_B N_B = 0$ .

The equivalent signal space between T and A after T nulls B is  $H_A N_B$ . T then calculates the Singular Value Decomposition of the equivalent signal space:

$$U \Sigma V^* = \text{SVD}(H_A N_B) \quad (\text{A.2})$$

To beamform towards A, T uses  $P = N_B V$  as the precoding matrix, so the received signal at A is:

$$Y_A = H_A P = H_A N_B V = U \Sigma V^* V = U \Sigma \quad (\text{A.3})$$

while the received signal at B is:

$$Y_B = H_B P = H_B N_B V = 0V = 0 \quad (\text{A.4})$$

# Bibliography

- [1] IEEE. Ethernet Specification. <https://standards.ieee.org/about/get/802/802.3.html>.
- [2] IEEE. 802.11 Specification. <http://standards.ieee.org/about/get/802/802.11.html>.
- [3] iPass. WiFi Growth Map. <https://www.ipass.com/wifi-growth-map/>.
- [4] Business Insider. Another record quarter for smartphone sales. <http://www.businessinsider.com/another-record-quarter-for-smartphone-sales-2013-55>.
- [5] Business Wire. Emergence of next-generation wireless networks will significantly augment the global mobile hotspot market until 2020, says Technavio. <http://www.businesswire.com/news/home/20160428005035/en/Emergence-Next-generation-Wireless-Networks-Significantly-Augment-Global>.
- [6] 3GPP. Enhanced Generic Access Networks (EGAN) study. <http://www.3gpp.org/ftp/Specs/html-info/43902.htm>.
- [7] A. Goldsmith. *Wireless Communications*. Cambridge University Press, 1st edition, 2005.

- [8] Ehsan Aryafar, Narendra Anand, Theodoros Salonidis, and Edward W. Knightly. Design and experimental evaluation of multi-user beamforming in wireless LANs. In *Proceedings of the Sixteenth Annual International Conference on Mobile Computing and Networking*, MobiCom '10, pages 197–208, New York, NY, USA, 2010. ACM.
- [9] S. Gollakota, S. Perli, and D Katabi. Interference alignment and cancellation. In *SIGCOMM*, 2009.
- [10] D. Tse and P. Viswanath. *Fundamentals of Wireless Communication*. Cambridge University Press, 2005.
- [11] Phil Karn. MACA - a new channel access method for packet radio. In *ARRL/CRRRL Amateur radio 9th computer networking conference*, volume 140, pages 134–140, 1990.
- [12] Vaduvur Bharghavan, Alan Demers, Scott Shenker, and Lixia Zhang. MACAW: a media access protocol for wireless LANs. In *ACM SIGCOMM Computer Communication Review*, volume 24, pages 212–225. ACM, 1994.
- [13] A Akella, G Judd, S Seshan, and P Steenkiste. Self-management in chaotic wireless deployments. In *MobiCom*, August 2005.
- [14] Broadband Radio Access Networks (BRAN) - 5 GHz high performance RLAN - Harmonized EN covering the essential requirements of article 3.2 of the R and TTE Directive, 2012.
- [15] Yihong Zhou and Scott M Nettles. Balancing the hidden and exposed node problems with power control in CSMA/CA-based wireless networks. In *Wireless Communications and Networking Conference, 2005 IEEE*, volume 2, pages 683–688. IEEE, 2005.

- [16] George T Karetos, Sofoklis A Kyriazakos, Evangelos Groustiotis, Felicita Di Giandomenico, and Ivan Mura. A hierarchical radio resource management framework for integrating WLANs in cellular networking environments. *Wireless Communications, IEEE*, 12(6):11–17, 2005.
- [17] Daniel Halperin, Wenjun Hu, Anmol Sheth, and David Wetherall. Tool release: gathering 802.11 n traces with channel state information. *ACM SIGCOMM Computer Communication Review*, 41(1):53–53, 2011.
- [18] H Rahul, S Kumar, and D Katabi. JMB: Scaling wireless capacity with user demand. In *SIGCOMM*, 2012.
- [19] IEEE Standard 802.11-2007: Wireless LAN MAC and PHY Specifications, June 2007.
- [20] Guido R Hiertz, Dee Denteneer, Philips Lothar Stibor, Yunpeng Zang, Xavier Pérez Costa, and Bernhard Walke. The IEEE 802.11 universe. *Communications Magazine, IEEE*, 48(1):62–70, 2010.
- [21] Micah Z Brodsky and Robert T Morris. *In defense of wireless carrier sense*, volume 39. ACM, 2009.
- [22] Andrew J Viterbi. Very low rate convolution codes for maximum theoretical performance of spread-spectrum multiple-access channels. *Selected Areas in Communications, IEEE Journal on*, 8(4):641–649, 1990.
- [23] Jeffrey G Andrews. Interference cancellation for cellular systems: a contemporary overview. *Wireless Communications, IEEE*, 12(2):19–29, 2005.
- [24] D Halperin, T Anderson, and D Wetherall. Taking the sting out of carrier sense: Interference cancellation for wireless LANs. In *MobiCom*, September 2008.

- [25] Shyamnath Gollakota and Dina Katabi. *Zigzag decoding: combating hidden terminals in wireless networks*, volume 38. ACM, 2008.
- [26] K Tan, H Liu, J Fang, W Wang, J Zhang, and G Voelker. SAM: Enabling practical spatial multiple access in wireless LAN. In *Proc. of the ACM MobiCom Conf.*, pages 49–60, Beijing, China, September 2009.
- [27] Maxim 2829 datasheet. <http://datasheets.maximintegrated.com/en/ds/MAX2828-MAX2829.pdf>.
- [28] V Navda, A Subramanian, K Dhanasekaran, A Timm-Giel, and S Das. Mobisteer: Using steerable beam directional antenna for vehicular network access. In *MobiSys*, June 2007.
- [29] Kishore Ramachandran, Ravi Kokku, Karthikeyan Sundaresan, Marco Gruteser, and Sampath Rangarajan. R2D2: regulating beam shape and rate as directionality meets diversity. In *Proceedings of the 7th international conference on Mobile systems, applications, and services*, pages 235–248. ACM, 2009.
- [30] X Liu, A Sheth, M Kaminsky, K Papagiannaki, S Seshan, and P Steenkiste. DIRC: Increasing indoor wireless capacity using directional antennas. In *SIGCOMM*, August 2009.
- [31] R Choudhury, X Yang, R Ramanathan, and N Vaidya. Using directional antennas for medium access control in ad hoc networks. In *MobiCom*, September 2002.
- [32] S Sen, J Xiong, R Ghosh, and R Choudhury. Link layer multicasting with smart antennas: No client left behind. In *ICNP*, October 2008.
- [33] Daniel Halperin, Srikanth Kandula, Jitendra Padhye, Paramvir Bahl, and David Wetherall. Augmenting data center networks with multi-

- gigabit wireless links. In *ACM SIGCOMM Computer Communication Review*, volume 41, pages 38–49. ACM, 2011.
- [34] Xia Zhou, Zengbin Zhang, Yibo Zhu, Yubo Li, Saipriya Kumar, Amin Vahdat, Ben Y Zhao, and Haitao Zheng. Mirror mirror on the ceiling: Flexible wireless links for data centers. *ACM SIGCOMM Computer Communication Review*, 42(4):443–454, 2012.
- [35] John S Thompson, Peter M Grant, and Bernard Mulgrew. Smart antenna arrays for CDMA systems. *Personal Communications, IEEE*, 3(5):16–25, 1996.
- [36] Sriram Lakshmanan, Karthikeyan Sundaresan, Sampath Rangarajan, and Raghupathy Sivakumar. Practical beamforming based on RSSI measurements using off-the-shelf wireless clients. In *IMC*, November 2009.
- [37] Minyoung Park, L Lily Yang, and Soon-Hyeok Choi. Opportunistic interference cancellation for MIMO systems. In *Personal, Indoor and Mobile Radio Communications, 2008. PIMRC 2008. IEEE 19th International Symposium on*, pages 1–6. IEEE, 2008.
- [38] JC Mundarath, Parameswaran Ramanathan, and Barry D Van Veen. Nullhoc: a mac protocol for adaptive antenna array based wireless ad hoc networks in multipath environments. In *Global Telecommunications Conference, 2004. GLOBECOM’04. IEEE*, volume 5, pages 2765–2769. IEEE, 2004.
- [39] M Maddah-Ali, A Motahari, and A Khandani. Signaling over MIMO multi-base systems: Combination of multi-access and broadcast schemes. In *ISIT*, 2006.

- [40] M Maddah-Ali, A Motahari, and A Khandani. Communication over MIMO X channels: Interference alignment, decomposition, and performance analysis. *IEEE Trans. on Information Theory*, 54(8):3457–3470, August 2008.
- [41] S Jafar and S Shamai. Degrees of freedom of the MIMO X channel. *IEEE Trans. on Information Theory*, 54(1):151–170, January 2008.
- [42] K Lin, S Gollakota, and D Katabi. Random access heterogeneous MIMO networks. In *SIGCOMM*, 2011.
- [43] S Kumar, D Cifuentes, S Gollakota, and D Katabi. Bringing cross-layer MIMO to today’s wireless LANs. In *SIGCOMM*, 2013.
- [44] Fadel Adib, Swarun Kumar, Omid Aryan, Shyamnath Gollakota, and Dina Katabi. Interference alignment by motion. In *Proceedings of the 19th annual international conference on Mobile computing & networking*, pages 279–290. ACM, 2013.
- [45] Bertrand M Hochwald, Thomas L Marzetta, and Vahid Tarokh. Multiple-antenna channel hardening and its implications for rate feedback and scheduling. *Information Theory, IEEE Transactions on*, 50(9):1893–1909, 2004.
- [46] Horia Vlad Balan, Ryan Rogalin, Antonios Michaloliakos, Konstantinos Psounis, and Giuseppe Caire. Achieving high data rates in a distributed MIMO system. In *Proceedings of the 18th annual international conference on Mobile computing and networking*, pages 41–52. ACM, 2012.
- [47] Q Yang, X Li, H Yao, et al. BigStation: Enabling scalable real-time signal processing in large MU-MIMO systems. In *SIGCOMM*, 2013.

- [48] K Nikitopoulos, J Zhou, B Congdon, and K Jamieson. Geosphere: Consistently turning MIMO capacity into throughput. In *SIGCOMM*, 2014.
- [49] J Xiong, K Sundaresan, K Jamieson, A Khojastepour, and S Rangarajan. MIDAS: Empowering 802.11ac networks with multiple-input distributed antenna systems. In *CoNEXT*, 2014.
- [50] Daewon Lee, Hanbyul Seo, Bruno Clerckx, Eric Hardouin, David Mazzarese, Satoshi Nagata, and Krishna Sayana. Coordinated multipoint transmission and reception in LTE-advanced: deployment scenarios and operational challenges. *Communications Magazine, IEEE*, 50(2):148–155, 2012.
- [51] C. Kosta, B. Hunt, AU. Quddus, and R. Tafazolli. On interference avoidance through inter-cell interference coordination (ICIC) based on OFDMA mobile systems. *Communications Surveys Tutorials, IEEE*, 15(3):973–995, March 2013.
- [52] N Himayat, S Talwar, A Rao, and R Soni. Interference management for 4G cellular standards. *IEEE Communications Magazine*, August 2010.
- [53] 3GPP Technical Specification 36.211 version 11.5.0 Release 11: Evolved Universal Terrestrial Radio Access (E-UTRA) Physical channels and Modulation.
- [54] IEEE Standard for Air Interface for Broadband Wireless Access Systems 802.16-2012.
- [55] Thomas Novlan, Jeffrey G Andrews, Illsoo Sohn, Radha Krishna Ganti, and Arunabha Ghosh. Comparison of fractional frequency reuse approaches in the OFDMA cellular downlink. In *Global Telecom-*

- munications Conference (GLOBECOM 2010), 2010 IEEE*, pages 1–5. IEEE, 2010.
- [56] P-H Huang, Y Gai, B Krishnamachari, and A Sridharan. Subcarrier allocation in multiuser OFDM systems: Complexity and approximability. In *WCNC*, 2010.
  - [57] K Seong, M Mohseni, and J Cioffi. Optimal resource allocation for OFDMA downlink systems. In *IEEE Intn'l Symp. on Information Theory*, 2006.
  - [58] G Li and H Liu. Downlink radio resource allocation for multi-cell OFDMA system. *IEEE Trans. on Wireless Comms.*, 5(12):3451–3459, December 2006.
  - [59] H Rahul, F Edalat, D Katabi, and C Sodini. Frequency-aware rate adaptation and MAC protocols. In *MobiCom*, 2009.
  - [60] K Tan, J Fang, Y Zhang, et al. Fine-grained channel access in wireless LAN. In *SIGCOMM*, 2010.
  - [61] K Chintalapudi, B Radunovic, V Balan, et al. WiFi-NC: Wi-Fi over narrow channels. In *NSDI*, 2012.
  - [62] R Chandra, R Mahajan, T Moscibroda, et al. A case for adapting channel width in wireless networks. In *SIGCOMM*, 2008.
  - [63] A Lozano, A Tulino, and S Verdú. Optimum power allocation for parallel gaussian channels with arbitrary input distributions. *IEEE Trans. on Information Theory*, 52(7):3033–3051, July 2006.
  - [64] Y-F Liu and Y-H Dai. On the complexity of joint subcarrier and power allocation for multi-user OFDMA systems. *IEEE Trans. on Signal Processing*, 62(3):583–596, February 2014.

- [65] Seong Taek Chung and John M Cioffi. Rate and power control in a two-user multicarrier channel with no coordination: the optimal scheme versus a suboptimal method. *Communications, IEEE Transactions on*, 51(11):1768–1772, 2003.
- [66] Campbell Wilson and Venugopal V Veeravalli. A convergent version of the Max SINR algorithm for the MIMO interference channel. *Wireless Communications, IEEE Transactions on*, 12(6):2952–2961, 2013.
- [67] Zuleita KM Ho, Mariam Kaynia, and David Gesbert. Distributed power control and beamforming on MIMO interference channels. In *Wireless Conference (EW), 2010 European*, pages 654–660. IEEE, 2010.
- [68] ITUT Recommendation. G. 992.1: Asymmetric digital subscriber line (ADSL) transceivers. *ITU-T, June*, 1999.
- [69] Hujun Yin and Hui Liu. An efficient multiuser loading algorithm for OFDM-based broadband wireless systems. In *Global Telecommunications Conference, 2000. GLOBECOM'00. IEEE*, volume 1, pages 103–107. IEEE, 2000.
- [70] Inhyoung Kim, Hae Leem Lee, Beomsup Kim, and Yong H Lee. On the use of linear programming for dynamic subchannel and bit allocation in multiuser OFDM. In *Global Telecommunications Conference, 2001. GLOBECOM'01. IEEE*, volume 6, pages 3648–3652. IEEE, 2001.
- [71] Oscar Punal and James Gross. Combined subcarrier switch off and power loading for 80 MHz bandwidth WLANs. In *Local & Metropolitan Area Networks (LANMAN), 2011 18th IEEE Workshop on*, pages 1–6. IEEE, 2011.

- [72] Oscar Punal, Humberto Escudero, and James Gross. Power loading: Candidate for future WLANs? In *World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2012 IEEE International Symposium on a*, pages 1–4. IEEE, 2012.
- [73] Fidelity Comtech, Inc. Phocus Array 3100X v2.1 data sheet.  
<http://www.fidelity-comtech.com/PDFs/DS%20Phocus%203100x%20v2.1%20v4%20Final.pdf>.
- [74] Gilbert Strang and Wellesley-Cambridge Press. *Introduction to linear algebra*, volume 3. Wellesley-Cambridge Press Wellesley, MA, 1993.
- [75] X Liu, A Sheth, M Kaminsky, K Papagiannaki, S Seshan, and P Steenkiste. Pushing the envelope of indoor wireless spatial reuse using directional access points and clients. In *MobiCom*, 2010.
- [76] Hakyung Jung, Kideok Cho, Yanghee Choi, et al. React: Rate adaptation using coherence time in 802.11 WLANs. *Computer Communications*, 34(11):1316–1327, 2011.
- [77] Hariharan Rahul, Haitham Hassanieh, and Dina Katabi. SourceSync: a distributed wireless architecture for exploiting sender diversity. *ACM SIGCOMM Computer Communication Review*, 41(4):171–182, 2011.
- [78] Daniel Halperin, Wenjun Hu, Anmol Sheth, and David Wetherall. Predictable 802.11 packet delivery from wireless channel measurements. In *SIGCOMM*, 2010.
- [79] Patrick Vandenameele. *Space Division Multiple Access for Wireless Local Area Networks*. KAP, 2001.
- [80] Patrick Murphy. *Design, Implementation and Characterization of a Cooperative Communications System*. PhD thesis, Rice University, 2010.

- [81] C.A. Balanis. *Antenna theory: analysis and design*. Harper & Row series in electrical engineering. Wiley, 1982.
- [82] Mirette Sadek, Alireza Tarighat, and Ali H Sayed. A leakage-based precoding scheme for downlink multi-user MIMO channels. *Wireless Communications, IEEE Transactions on*, 6(5):1711–1721, 2007.
- [83] WARP Project. <http://warpproject.org>.
- [84] Timothy M Schmidl and Donald C Cox. Robust frequency and timing synchronization for OFDM. *Communications, IEEE Transactions on*, 45(12):1613–1621, 1997.
- [85] Michael Mitzenmacher. The power of two choices in randomized load balancing. *Parallel and Distributed Systems, IEEE Transactions on*, 12(10):1094–1104, 2001.
- [86] Joan Palacios, Danilo De Donno, Domenico Giustiniano, and Joerg Widmer. Speeding up mmWave beam training through low-complexity hybrid transceivers. 2016.
- [87] Pubudu N Pathirana, Andrey V Savkin, and Sanjay Jha. Mobility modelling and trajectory prediction for cellular networks with mobile base stations. In *Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing*, pages 213–221. ACM, 2003.
- [88] Max HM Costa. Writing on dirty paper (corresp.). *Information Theory, IEEE Transactions on*, 29(3):439–441, 1983.
- [89] Mayank Jain, Jung Il Choi, Taemin Kim, Dinesh Bharadia, Siddharth Seth, Kannan Srinivasan, Philip Levis, Sachin Katti, and Prasun Sinha. Practical, real-time, full duplex wireless. In *Proceedings of the 17th an-*

*nual international conference on Mobile computing and networking*, pages 301–312. ACM, 2011.