# Improving the Performance of Wireless Systems Through Distributed Fault Diagnosis

by

**Anmol Sheth**

B.E, University of Pune (India), 2001

M.S., University of Colorado, 2005

A thesis submitted to the

Faculty of the Graduate School of the

University of Colorado in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

Department of Computer Science

2007

UMI Number: 3273723



ProQuest Information and Learning Company
300 North Zeeb Road
P.O. Box 1346
Ann Arbor, MI 48106-1346

This thesis entitled:
Improving the Performance of Wireless Systems Through Distributed Fault Diagnosis
written by Anmol Sheth
has been approved for the Department of Computer Science

Prof. Richard Han

Prof. Dirk Grunwald

Prof. Douglas Sicker

Prof. Shivakant Mishra

Prof. Regan Zane

Date

The final copy of this thesis has been examined by the signatories, and we find that both the content and the form meet acceptable presentation standards of scholarly work in the above mentioned discipline.

Sheth, Anmol (Ph.D., Computer Science)

Improving the Performance of Wireless Systems Through Distributed Fault Diagnosis

Thesis directed by Prof. Richard Han

The reduction in cost and ease of installation of 802.11 based wireless LAN (WLAN) hardware has resulted in a surge of large scale deployments. Furthermore, with the flexibility of open source WLAN device drivers, there is growing interest to extend the basic WLAN technology to outdoor mesh and long distance networks. However, due to the unreliable nature of wireless links and chaotic deployments, users of such networks frequently encounter degraded performance and lack of coverage. Existing approaches that aim to diagnose these problems are inefficient because they are unable to distinguish among the root causes of performance degradation. In this thesis, through detailed measurement analysis of WLAN deployments, I have identified the primary sources of performance degradation in indoor WLAN deployments and outdoor mesh networks. I have designed and implemented fine-grained detection algorithms that are capable of distinguishing between root sources of performance anomalies at the depth of the physical layer using stock WLAN hardware. Unlike the stock 802.11 protocol, which always resorts to rate fallback as a remedy, I have developed targeted loss recovery mechanisms to mitigate the fault and improve the performance of the network. In addition to software based recovery mechanisms, this thesis also explores the use of smart antennas in mitigating the loss. An important property that emerges from my system is that diagnostic observations are combined from multiple sources over multiple time instances for improved detection accuracy and network performance.

## Dedication

I dedicate this to mom.

## Acknowledgements

I would like to first thank my advisor Prof Richard Han for all the support that he has provided me with during my entire time at University of Colorado. Having started graduate school at the same time as my advisor started as a tenure track faculty member, I was lucky to experience and participate in driving a research agenda from scratch with my advisor. The primary quality that I acquired along this process from him is the ability to take risks and perform exciting research. I would also like to thank Prof. Dirk Grunwald for being a great role model. His approach to research and solving problems has been instrumental in my dissertation.

I am also thankful to the many students in the Systems lab. The long discussions over coffee helped make the six years of my dissertation period memorable and enjoyable.

I am also indebted to my family and wife for supporting me during the last six years. My wife was with me for the last one year of my dissertation, which has been the most productive and fun year of my dissertation period.

Last but not the least, I feel that I have been lucky to have worked with people like Prof. Eric Brewer, Dr. Chandramohan Thekkath and Dr. Pravin Bhagwat during my internships. These researchers have been a role model for me through out my dissertation period. They have been very influential in grooming me as a researcher.

# Contents

**Chapter**

# Tables

**Table**

# Figures

**Figure**

# Chapter 1

# Introduction

In recent years, there has been a surge in 802.11 (WiFi) based Wireless Local Area Network (WLAN) deployments. This sudden surge in WLAN deployments is also coupled with the changing nature of deployments. Deployments of 802.11 networks done when the technology was first introduced (first generation deployments) were typically small, planned and had a limited set of applications using these networks. However, existing WiFi network deployments (second generation deployments) have very different characteristics. Second generation deployments are large, often consisting of 100-200 access points [23]. They are also unplanned and chaotically deployed, and the set of applications using these networks range from QoS specific applications like VoIP and video streaming to interactive Internet gaming. Second generation WiFi deployments are also extending WiFi to outdoor urban mesh deployments [20] and stretching WiFi links over 100-200 kms [52].

The four main factors that have led to this accelerated growth of second generation WLAN deployments are: low cost, low power, easy configuration and portability. The low cost and low power consumption of the chipsets have led to the rapid adoption of WiFi by embedded devices like laptops, digital cameras and smart phones. The small form factor and almost zero configuration of the access points have led to dense deployments in most residential areas [12].

As a result of this changing nature of the second generation deployments, there

have been several measurement based studies that have been performed to understand the "real world" behavior of these deployments [39, 32, 59, 40, 62]. These studies highlight the poor and unpredictable performance of 802.11. The authors in [59, 40] study the performance of 802.11 in a conference setting, where a large number of clients are using the wireless network. The authors observed both short term as well as long term variability in link quality and performance degradation under heavy usage of the wireless network. The authors in [10] report severe performance degradation when WiFi is extended to outdoor multihop mesh networks. The authors attribute inter-link interference and multipath to be the main sources of packet loss in such networks. The authors in [62] report highly asymmetric and unreliable links when WiFi is stretched over distances of 100-200 kms.

To mitigate the performance degradation observed in these second generation deployments, it is important to accurately detect the root source of the performance degradation. Only by detecting the root source can efficient remedies be applied to mitigate the performance degradation. Existing approaches that aim to diagnose these problems are inefficient because they troubleshoot at too high a level and too coarse a granularity. Furthermore, existing tools are unable to distinguish among the root causes of degradation as they lack distributed observations of the network. Even the stock 802.11 MAC protocol cannot distinguish between the root causes. For example, the default 802.11 remediation for any performance degradation fault is to initiate rate fallback. In some cases, this can rapidly degrade the network performance because an inappropriate remedy is applied and stations are forced to lower data rates, leading to poor network performance [33, 30]. In other cases, this rate fallback helps only to partially circumvent the problem. With complete knowledge about the root cause of the fault, efficient remediation procedures can be performed that improve the performance of the network.

This thesis explicitly addresses the lack of diagnosis and troubleshooting tools and

algorithms to mitigate the performance degradation observed in large scale distributed wireless systems. This thesis proposes a novel architecture for distributed wireless network monitoring that, unlike existing systems, ensures complete coverage of the network and accurately detects the root source of the performance degradation fault. Fine grained distributed network monitoring is achieved by instrumenting the client side wireless drivers and efficiently aggregating the information from the distributed end-points. This thesis describes in detail the design, implementation and evaluation of the tools and algorithms for two very different wireless networks; indoor 802.11 deployments and outdoor WiFi based Long Distance (WiLD) deployments. For each of these networks, this thesis identifies the primary sources of performance degradation and proposes and implements remedies to mitigate the performance degradation. In addition to the software based tools and algorithms, this thesis also explores how smart antennas could be used to mitigate the sources of performance degradation observed in WiLD networks.

## 1.1 Thesis Statement

The primary research question addressed by this thesis is; "Can the performance of indoor WiFi and outdoor long distance WiFi networks be improved by identifying the root source of performance degradation?" This thesis shows that by extracting fine-grained metrics from the distributed end-points in the network, the root source of performance degradation can be isolated and informed remedies can be applied that significantly improve the performance of the network.

It achieves this by proposing a novel network monitoring framework, detection algorithms, and informed remedies. The network monitoring frame-work collects fine-grained information of the network from the distributed end-points. Based on this fine-grained information, the detection algorithms are capable of isolating the root source of performance degradation with high accuracy. By isolating the root source, targeted remedies are implemented that significantly improve the performance of the network.

## 1.2 Thesis Contributions

The primary contributions of this thesis are:

- This is the first body of work which looks at building a unified framework to be able to detect and troubleshoot underlying performance degradation faults in distributed wireless systems.

- We are the first to quantify the effect of different performance degradation faults in indoor 802.11 deployments and WiLD networks and measure the impact of the performance degradation faults at different layers of the stack.

- I build novel detection algorithms for each physical layer performance degradation fault for indoor 802.11 deployments and test the accuracy of the detection algorithm on a real testbed.

- We are the first to perform a detailed study of the fundamental shortcomings of the stock 802.11 MAC protocol for WiLD networks.

- I analyze three well known causes for channel losses in outdoor long distance wireless networks, namely, external WiFi interference, non-WiFi interference and multipath interference. Among these, we are the first to show that external WiFi interference is the significant source of packet loss and the effect of multipath and non-WiFi interference is not significant.

- Having identified external WiFi interference as the primary source of losses in WiLD networks, I propose two potential remedies to mitigate these losses: (a) adaptive FEC, and (b) bulk acknowledgments. These remedies have been implemented and evaluated on a real testbed.

- In addition to software solutions to mitigate the performance degradation due to external WiFi interference in WiLD networks, I also evaluated the use of

smart antennas (phased array antennas) and designed algorithms to reduce the effect of external interference in WiLD networks.

- All the above mentioned algorithms and tools were built using commodity off-the-shelf hardware, and did not require any modifications to the underlying 802.11 MAC protocol.

## 1.3 Dissertation Outline

The rest of this thesis is organized as follows. Chapter 2 outlines the challenges involved in root source detection of performance degradation faults, and presents the system architecture based on distributed network monitoring. Chapter 3 presents the design, implementation, and evaluation of the fault detection algorithms for indoor 802.11 WLANs. Chapter 4 motivates WiLD networks as a viable solution to provide network connectivity in low user density areas. Chapter 5 identifies the primary sources of packet loss in WiLD networks, and chapter 6 discusses the design and implementation of the remedies to mitigate this loss. Chapter 7 discusses the limitations of static directional antennas and the software based loss recovery mechanisms. It outlines how smart antennas, like phased array antennas, could be used to overcome these limitations. Chapter 8 presents the related work, and I conclude the thesis in chapter 9.

# Chapter 2

# Challenges in root source fault detection

In this chapter, we highlight the challenges involved in accurate diagnosis of the root source of performance degradation in large scale distributed wireless systems. We also present the system architecture based on distributed network monitoring to address these challenges. The primary requirement for accurate diagnosis is to identify and extract metrics that provide maximum insight into the source of the problem. In the following sections we outline the three main challenges involved in identifying and extracting the metrics for accurate diagnosis.

## 2.1 Convergence of performance degradation faults

Figure 2.1 illustrates some of the commonly observed performance degradation faults in wireless networks. The root causes of the faults are shown by solid boxes and the dashed boxes denote the effect of the root cause. As seen in the figure, faults that originate at the PHY layer converge at higher layers of the stack. At the higher layers, all the faults manifest themselves as degraded performance. It is this convergence of the manifestation of the root causes that makes diagnosis and troubleshooting faults in a wireless network a challenging task. Faults like hidden terminals, capture effect, signal strength variations and noise in the network all cause retransmissions at the MAC and degraded throughput at the network layer. Without adequate visibility into the PHY layer, it is not possible to differentiate a retransmission caused due to hidden terminals

and a retransmission caused due to noise in the network.

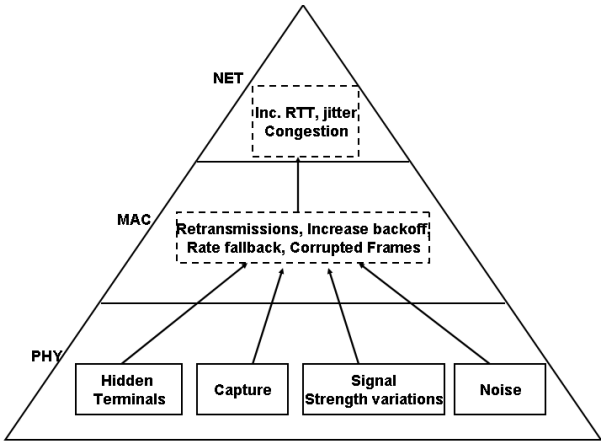


Figure 2.1: Pyramid structure of how faults propagate up the network stack

Hence, to be able to differentiate between the root causes of the performance degradation it is important to monitor across the entire network stack. Cross layer metrics that include the physical layer are required to diagnose the root cause accurately.

## 2.2 Limitations of existing performance diagnostic tools

To address the performance degradation in enterprise scale WiFi deployments, there are a number of open source as well as commercial tools [1, 2] available that perform network planning. With exhaustive site surveys and detailed information about the characteristics of the environment, these tools allow the network administrator to set the frequency channel, power and location of the APs to optimize the performance and coverage of the network.

However, these tools are incomplete because they capture the behavior and organization of the network at a single point in time; wireless networks encounter sufficient time-varying conditions that inexpensive dynamic monitoring is useful. Moreover, many of the problems experienced in a network occur because of the **stations** in the network; most site-planning tools only focus on the placement and performance of access points.

Diagnostic tools that only collect packet statistics at the MAC layer and above will also suffer from a **masking effect** such that a single higher-layer network statistic will aggregate the effects of more fundamental lower-layer causes, thereby masking the individual root causes.

## 2.3 Sniffer placement

Sniffer placement is an important factor for wireless network monitoring, as it determines the coverage of the network. Due to the unreliable nature of the broadcast medium, wireless traces are inherently lossy; hence, a sub-optimal placement of these sniffers could leave parts of the network un-monitored.

Existing wireless network monitoring work has mainly focused on performance monitoring and security monitoring to detect rogue APs [13]. As we will show, a key requirement of diagnosing root cause faults at the physical layer is that along with adequate coverage, multiple sniffer observations are required. Existing work only focuses on placement of sniffers to ensure complete coverage of the wireless network.

A number of traffic measurement studies have been done that collect traffic statistics by monitoring the traffic flowing on the wired end of the network by using tools like SNMP and syslog (AP system logs) [32, 14]. Although these tools provide complete information of the traffic flowing on the wired end of the network, it provides limited visibility into the wireless end of the network. These tools cannot record fine grained information at the MAC and PHY layers and usually only provide aggregate statistics maintained by the AP.

To address the limitations of wired side monitoring, researchers have proposed wireless monitoring based on fixed sniffers. These sniffers are carefully placed relative to the client positions in the wireless network. However, the authors in [69] observe that even with careful placement of wireless sniffers, multiple wireless sniffer traces are required to be merged so as to account for data missed by one or more sniffers.

Furthermore, often client locations are not known **a priori** or these may change over time, requiring sniffers locations to be changed frequently.

An additional constraint is that most faults are **localized** in the network. Hence, sniffers should be collocated with the client stations. For example, only the client station that is close to a microwave oven is subject to noise/interference, but even the closest sniffer to the client may not be able to sense the noise in the network.

## 2.4 System Architecture

Given the above mentioned constraints and limitations of existing wireless network monitoring infrastructure, in this section we outline the system architecture required to diagnose the root source of performance degradation in distributed wireless systems. Our approach is to perform client-end distributed monitoring of the network to ensure complete coverage of the network. As we shall see in the next few chapters, recording network performance metrics at the client and aggregating this information for diagnosis results in correctly identifying the root source of the performance degradation. To summarize, the main design goals are:

- Complete coverage of the wireless network is required. Along with complete coverage, multiple observations are required to improve the accuracy of the fault detection.
- The monitoring framework should be compliant to the stock 802.11 MAC protocol.
- The monitoring framework should be lightweight, and should not overwhelm the network.

The two main challenges involved in extracting fine-grained diagnosis information from the physical layer are:

- FCC regulations do not allow hardware chipset manufacturers to distribute open source versions of the 802.11 physical layer implementation
- To make the system easily deployable, it is important to have a completely software solution with no additional hardware

To extract fine grained information, we propose instrumenting the client side driver to collect information about the underlying physical layer. To collect information about the physical layer we have instrumented the Atheros based Madwifi driver [6]. This client side monitoring ensures complete coverage of the entire network and does not require careful site surveys before the system is deployed. However, there are no constraints in our design that require sniffers to be implemented only in the client. In general, the sniffers are allowed to be placed anywhere in the network, with the recognition that non-client-side placement of sniffers results in a suboptimal picture of the network.

Commodity 802.11 hardware typically divides up the functionality of the 802.11 MAC between the hardware/firmware on the card and the driver running on the host system. This means that the flexibility of such systems varies greatly between manufacturers. To extract information from the physical layer, we have used the Atheros AR5212 chipsets and the open-source **Madwifi** driver [6]. Atheros uses a "hardware abstraction layer" or HAL to provide a common hardware interface for operating systems. The HAL is written in the machine code of the computer hosting the wireless card, and abstracts common functionality across different individual chipsets. Although the HAL is distributed in binary-only format and not extensively documented, there have been attempts to produce an "open-source" HAL.

Using the information from the driver documentation and the "open-source" HAL, we have instrumented the stock Atheros based 802.11 driver to log fine-grained information for each frame received and transmitted. In addition to capturing all the frames on

the link, the effect of external WiFi interference is evaluated by capturing and logging frames transmitted by external WiFi sources. This is achieved by creating a virtual network interface set in "monitor mode" on the same channel as the primary interface. This technique is equivalent to using two physical network interfaces, one being the primary and the other a passive monitor. The Atheros driver is also modified to pass up frames with CRC and PHY errors to measure the precise Packet Error Rate (PER) and Bit Error Rate (BER) for controlled experiments.

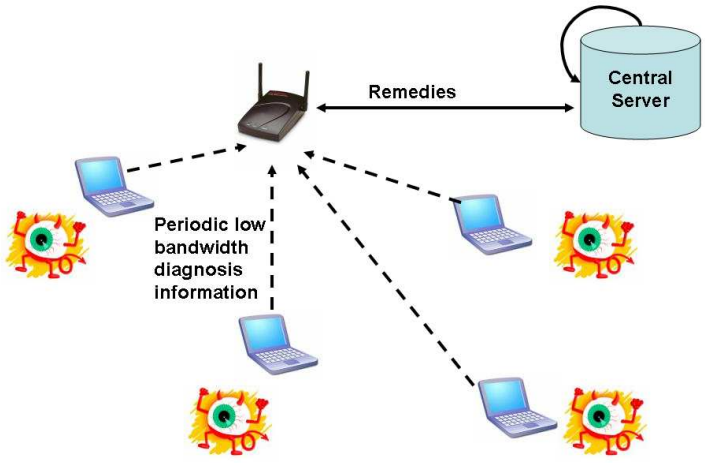


Figure 2.2: Aggregation of distributed observations in indoor 802.11 WLAN networks

The fine-grained diagnostic information collected from the distributed end-points in the network is either aggregated and processed at a central point (access point or a central server on the Internet), or the information is used by the distributed end-points to diagnose the root cause. For example, the diagnosis of hidden terminals and capture effect in indoor 802.11 WLAN deployment requires the aggregation of packet transmission timestamps from the distributed clients in the network. As illustrated in Figure 2.2, the end-points in the network transmit this information to the central server via the AP. The central server aligns the packet transmission timestamps and detects

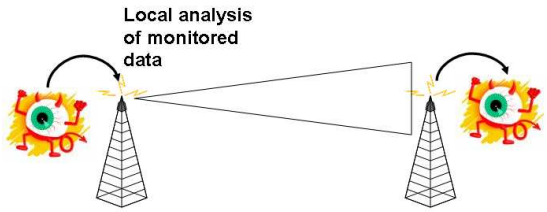


Figure 2.3: Local data analysis in WiLD networks

concurrent transmissions in the network and proposes informed remedies to mitigate the performance degradation. On the other hand, detection of WiFi interference in WiLD networks requires analysis of only locally monitored information as illustrated in Figure 2.3.

To summarize, we collect the following information for every frame by instrumenting the client side wireless driver: complete 802.11 MAC header and IP payload, received signal strength, data rate used to transmit the frame, timestamp of the frame, frames containing PHY and CRC errors, and the noise floor immediately after the frame is received.

# Chapter 3

## Fault diagnosis in indoor 802.11 deployments

Performance degradation in 802.11 WLANs arises from a variety of common sources, including 802.11-based interference, non-802.11 interference [10, 39], RF effects like hidden terminals and the capture effect [19, 34], heterogeneity, and limitations of the 802.11 MAC itself [33, 30]. In this chapter we present the design, implementation, and evaluation of the detection algorithms for commonly observed problems/faults in indoor WLAN deployments. These detection algorithms are able to distinguish between root causes of performance degradation at the granularity of PHY layer phenomena. As will be demonstrated later, gaining a more precise understanding of the root causes of an anomaly at the depth of the PHY layer enables more informed remediation.

In particular, we devise detection algorithms that detect hidden terminals in the network and differentiate that activity from terminals experiencing capture effect. We also devise algorithms that detect noise due to non-802.11 devices and detect anomalous signal strength variations at the AP and determine if those signal variations are caused by environmental conditions or actions by the access point. An important property that emerges from the system is that diagnostic observations are combined from multiple distributed sources over multiple time instances for improved accuracy and efficiency.

Each fault is artificially replicated on a testbed and the performance degradation caused by the fault is analyzed at the physical layer, link layer and at the network layer. Based on the analysis of the fault at the different layers of the network stack, we propose

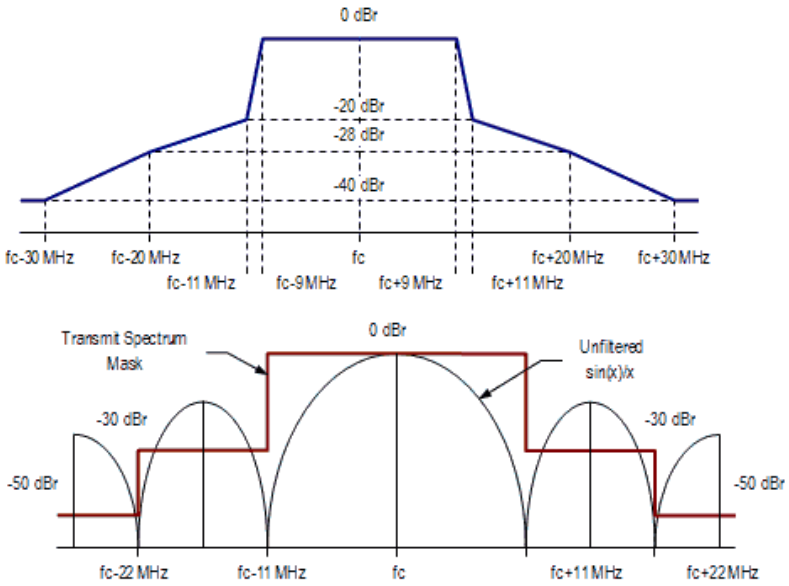


Figure 3.1: Spectral mask for OFDM and DSSS modulation

detection algorithms using information from the underlying physical layer, and finally measure the effectiveness of the detection algorithm.

## 3.1 Noise or non-802.11 interference

Recent metropolitan area WiFi monitoring studies [12] have shown that dense deployments of 802.11 networks consisting of 10-15 APs in range of each other are common. Not only are these deployments dense, but it is also common to have mixed-mode 802.11b and 802.11g deployments. Additionally, the 802.11 protocol operates in the unlicensed 2.4GHz shared spectrum. The signal spillage due to neighboring APs and other devices sharing the spectrum make noise/non-WiFi interference a significant source of performance degradation.

The 802.11 protocol specifies a listen-before-talk based CSMA channel access mechanism, wherein a transmitter would defer transmission until the channel has been sensed free for a duration of (50 $\mu$sec). The noise floor threshold against which the sampled noise floor is compared against ranges from -85 dBm to -90 dBm.

Figure 3.1 shows the transmit spectral mask for OFDM and spread spectrum (DSSS) encoding as specified by the IEEE 802.11b and 802.11g specification respectively. The spectral mask includes the energy that is radiated into the adjacent channels of the spectrum. The figure shows that OFDM has much higher energy radiation as compared to spread spectrum. Even at 22 MHz from the center frequency ($F_c + 22\ MHz$), the energy radiation of a spread spectrum transmission is -50 dBr and that of OFDM is -30 dBr. Thus, assuming the typical transmit power of an AP of 20 dBm and a path loss of -60 dBm for a dense deployment of APs, an OFDM transmission on channel 6 (2.437 GHz) would radiate approximately -70 dBm power in channel 11 (2.452 GHz). This radiated energy that cannot be decoded is sensed as noise by the transmitter on channel 11, and causes the transmitter to defer until the transmission of the OFDM transmission on channel 6 is completed. Hence, even with careful planning of the network, the wide spectral mask of OFDM transmissions causes interference in non-overlapping channels. This noise level increases with multiple access points operating in mixed mode and interfering with each other.

An alternate source of noise/interference in the network is the energy radiated by the non-802.11 devices like microwave ovens and cordless phones, which also operate in 2.4 GHz ISM band. Most non-802.11 devices like microwave ovens and cordless phones operating in the ISM band do not follow a channel access protocol. Due to the lack of a common channel access protocol, there is significant interference caused by these devices.

To overcome the above problems, it is important to be able to sense/measure the level of noise/interference on the channel. With detailed knowledge of the noise level, adaptive channel selection algorithms could be implemented which could reduce the degradation in performance.

### 3.1.1 Impact of noise at MAC and NET layer

To measure the impact of noise/non-802.11 interference on the network stack I conducted controlled experiments using the Agilent 4438C signal generator as a calibrated noise source. A frequency modulated signal, similar to the interference caused by microwave ovens and cordless phones, was generated. The experimental setup consisted of node A associated with the AP. The signal generator was only connected to node A using a RF splitter. The other port of the RF splitter was connected to the sniffer, which logged the timestamp of the frames received and transmitted by node A.

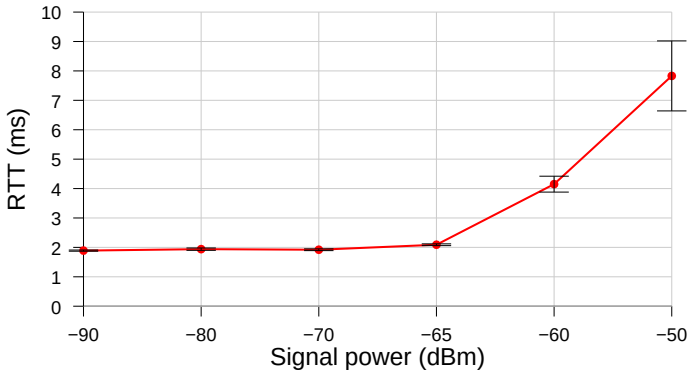


Figure 3.2: Measured RTT increases as the power of the signal generator is increased. Payload is 768 bytes (Bars show 95% confidence interval).

The round trip time (RTT) at the network layer was measured. The power of the signal generator was increased from -90 dBm to -50 dBm and the packet payload was increased from 256 bytes to 1024 bytes in steps of 256 bytes. For each setting of the power and payload size, 1000 frames were transmitted by station A; the experiments were repeated 10 times. The graphs show the mean and the 95% confidence intervals. Figure 3.2 shows the increase in the round trip time as the power of the signal generator is increased. The RTT does not change until the signal power is around -65 dBm. However, beyond -65 dBm, there is a sharp increase in the RTT. Beyond -50 dBm there was 100% packet loss.

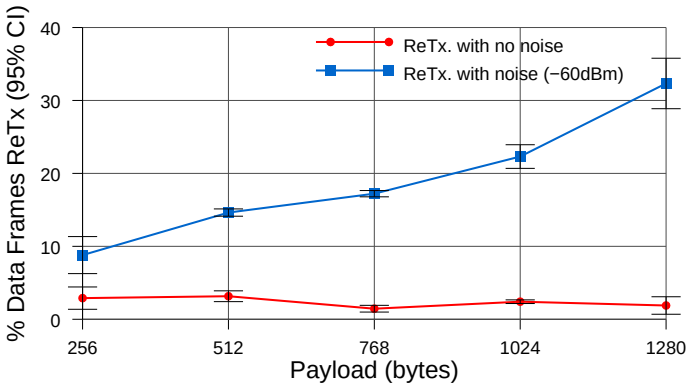


Figure 3.3: Percentage of data frames re-transmitted by node. Signal power set at -60 dBm.

Looking deeper at the MAC layer traces we observed that there are two main reasons which contribute to the increased RTT at the network layer: channel interference and excessive backoff at the MAC layer.

Due to the interference on the channel, a significant percentage of the frames are corrupted and have to be retransmitted at the MAC layer. At a power level of -60 dBm, around 20-30% of the frames received at the MAC layer are corrupted and have to be retransmitted. Figure 3.3 plots the percentage of data frames that are retransmitted for different payload sizes at a power level of -60 dBm. As the payload of the MAC frame is increased, the frame is more likely to be corrupted. Hence at the maximum payload of 1280 bytes, the percentage of retransmissions increases to around 35%. Figure 3.3 also shows the percentage of frames being retransmitted in the absence of any noise in the network.

$$T_{Backoff} + T_{Busy} = T_2 - DATA - Preamble - DIFS - T_1 \tag{3.1}$$

Figure 3.4 provides a high level overview of the 802.11 DCF protocol. The transmitting station needs to sense the medium to be free for a DIFS interval (50 $\mu$sec) and then select a random backoff ($T_{Backoff}$) before initiating transmission. The random

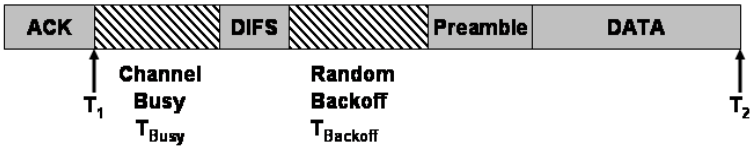


Figure 3.4: DCF mechanism of 802.11 protocol

backoff is chosen from a collision window which is exponentially doubled on an ACK timeout and set to minimum on successfully receiving an ACK. $T_{Busy}$ is the time interval spent sensing the medium to become free. We analyzed the packet trace collected at the MAC layer by the monitoring station to calculate precisely the amount of time the stations spends in backoff and busy sensing ($T_{Backoff} + T_{Busy}$). This is calculated by measuring the amount of time spent after receiving an ACK and the initiation of the next data frame from station A. Equation 3.1 shows the backoff calculation.

| Power (dBm) | Mean ($\mu$sec) | Std.Dev. ($\mu$sec) |
|---|---|---|
| -90 | 96.28 | 160.17 |
| -80 | 96.71 | 168.60 |
| -70 | 105.37 | 224.88 |
| -65 | 212.60 | 876.19 |
| -60 | 286.35 | 716.97 |
| -50 | 960.28 | 1978.08 |

Table 3.1: Mean and std. dev. of the time spent in backoff and busy sensing the medium

Table 3.1 shows the mean and the standard deviation of the amount of time spent in backoff and busy sensing the medium for different power levels of the signal generator. The payload was fixed at 768 bytes. Clearly, at higher transmit power levels, a significant amount of the RTT is spent in backoff and busy sensing. While increasing the contention window reduces **contention** between cooperating stations, it does not reduce **interference** from a noise source.

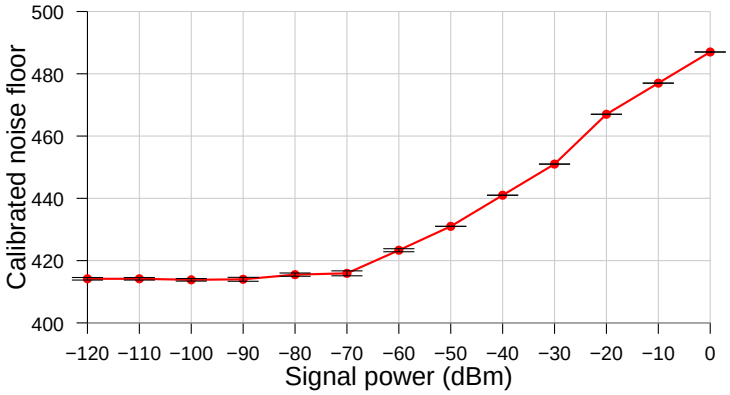


Figure 3.5: Mean and 95% conf. interval noise floor calibration for the Atheros chipset.

### 3.1.2 Impact of Noise at PHY layer and Detection Algorithm

Figure 3.5 shows the mean and 95% confidence interval of the noise floor values reported by the Hardware Abstraction Layer (HAL) for different power levels of the input signal generator.[1] We observe that the noise floor does not change until around -70 dBm; beyond -70 dBm there is a linear increase in the calibrated noise floor. The maximum standard deviation that was observed at any power level was 0.6 units and the average standard deviation across all the power levels was 0.2 units.

Since the Atheros chipset has precise noise detection, the detection algorithm is simplified. We change the sampling rate of the noise floor from 30 sec to a configurable interval (EPOCH_INTERVAL). Every EPOCH_INTERVAL the noise floor is sampled and transmitted to the central server. There is negligible overhead incurred in sampling the noise floor as it involves a single register read operation. The central server maintains a sliding window average of the mean and monitors the noise floor sampled by the sniffer to detect a change in mean.

Figure 3.6 shows the noise floor sampled once every 5 mins by a sniffer over a period of 5 days in a typical residential setting. On an average, there were 8 APs in

[1] The units on the Y-axis are specific to the Atheros chipset and their meaning is not known.

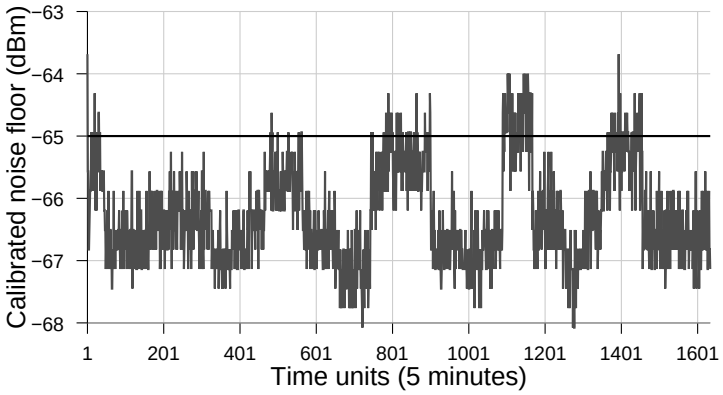


Figure 3.6: Noise floor sampled every 5 mins for a period of 5 days in a residential environment. The detection threshold is set at -65 dBm

range of each other on channel 6. The residential setting is representative of a collection of unplanned networks with APs installed by the home owners. The sniffer's frequency channel was set at 2.437 GHz (Channel 6). Based on the beacon frames recorded by the sniffer, on average there were 8 APs in range of the sniffer operating on the same channel. The graph shows a long term increase and decrease in the sampled noise floor across the five days. As seen in Figure 3.3, the RTT begins to increase only beyond -65 dBm. Hence, we set the noise floor threshold to -65 dBm and trigger a fault when the sliding window average is above the threshold. Figure 3.6 shows that the noise floor often increases above the threshold for long time periods. By detecting the increase in noise floor, the client can either switch the channel of the AP or associate with an alternate AP.

## 3.2 Hidden terminals and capture effect

In this chapter we study the impact of capture effect and hidden terminals at the different layers of the network stack and present detection algorithms to differentiate between the two.

### 3.2.1 Overview of hidden terminals and capture effect

| | Tx. | 1M | 2M | 5.5M | 11M |
|---|---|---|---|---|---|
| Cisco 350 | 20 | -94 | -91 | -89 | -85 |
| Orinoco Gold | 15 | -94 | -91 | -87 | -82 |
| Dlink DWLG650 | 15 | -89 | -86 | -85 | -82 |
| Compaq WL110 | 15 | -94 | -91 | -87 | -82 |
| Linksys WPC11 | 18 | -91 | -89 | -85 | -82 |
| Linksys WPC55AG | 17 | -93 | -91 | -88 | -86 |

Table 3.2: Transmit power and receive sensitivity in dBm. Uniformly, receivers are less susceptible to noise when using the slower data rates and there is significant variance between different receivers. Transmission power can reach as high as 300 mW (25 dBm).

Table 3.2 lists the specifications of the transmit power and receive sensitivity of a heterogeneous collection of common 802.11 client adapters. We observe that each client adapter has a different transmit power and data rate sensitivity. Heterogeneous transmit power leads to **asymmetric transmission ranges**, which exacerbates the problem of the hidden terminals and capture effect in the network.

Figure 3.7 illustrates the difference between hidden terminals and capture effect. Station C is isolated by an RF barrier from stations A and B, resulting in the classical "hidden terminal" problem. Stations A and B cannot sense transmissions by station C; simultaneous transmissions by C and B would cause corrupted packets at the access point, AP. However, not all simultaneous transmissions lead to corruption. For example, due to aspects of the 802.11 media acquisition, stations A and B may simultaneously transmit; however, the transmission from station B is likely to "capture" the AP receiver, leading to a successful reception. The standard remedy for hidden terminals would be to have station C use the RTS/CTS mechanism when communicating with the access point. This would inform stations A and B that the media is busy. Likewise, the

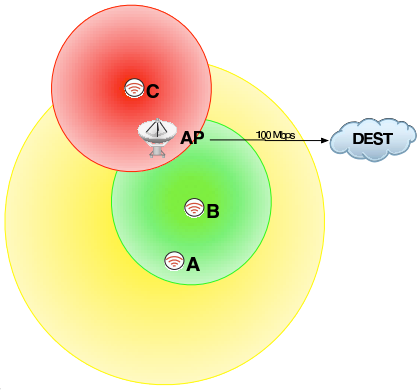


Figure 3.7: Network organizations leading to hidden terminals and capture effect

problem of the "capture effect" can be remedied either by having station A increase its transmit power or by adjusting the media acquisition mechanisms.

In the above description, it is important to note that both capture effect and hidden terminals are caused due to concurrent transmissions and collisions at the receiver. However, the important differentiation between the two is that the transmitting stations that are causing capture at the receiver are not necessarily hidden from each other.

### 3.2.2 Experimental setup

A key question that needs to be answered is **why would two stations that are in range of each other transmit concurrently**. There are two key features that cause the above anomaly. One, the 802.11 protocol sets the contention window to CWmin on receiving a successful ACK and a backoff interval is selected from this contention window. And second, there is high variability in the time required to sense the channel. The 802.11 specification states that the total Clear Channel Assessment (CCA) time including the turnaround time is 25 $\mu$sec (Section 15.4.8.4 of the IEEE 802.11 specification [8]).

Consider the two competing stations A and B as shown in Figure 3.7 that are in

range of each other and have their contention window set to the minimum CWmin. If station B initiates transmission at time $T_B$ and the backoff timer of station A expires within the interval $T_B + 25\ \mu sec$, station A would not have correctly sensed the medium to be busy and would initiate the transmission causing a collision at the receiver. The 25 $\mu sec$ interval is calculated as 15 $\mu$sec for energy detection time and 10 $\mu$sec for the Tx-Rx turnaround time. Variability in both the energy detection time as well as the turnaround time for different chipsets would affect the time difference between the collisions.

In the case of hidden terminals in the network, the nodes are not in range of each other and hence can collide at any point in a transmission.

To measure the impact of capture effect and hidden terminals at the different layers of the network stack, we artificially set up the faults on a testbed similar to the layout shown in Figure 3.7. To set up capture effect, node B was placed closer to the AP as compared to node A. The SNR of node B at the AP was measured to be -50 dBm and that of node A was measured to be -65 dBm. Rate fallback was turned off and the rate at both the stations was fixed.

To measure the impact of hidden terminals, we set up asymmetric hidden terminals. In this case, the transmit power of node C was attenuated such that it had a perfect link to the AP, but it was hidden from node A. We term this example of hidden terminals as **asymmetric hidden terminals**, in which only a single station is hidden from the other. Due to the asymmetric transmission ranges and heterogeneity of client interface specifications, we observed that asymmetric hidden terminals are more common as compared to the classic example of hidden terminals where both stations are hidden from each other.

Note that for the capture effect, node B has a higher SNR at the AP as compared to node A. For the asymmetric hidden terminal example, the transmit power at node C was attenuated such that it is hidden from node A, and hence node A has a higher

| Metric | Capture (11 Mbps) | Capture (5.5 Mbps) | Hidden Terminal (11 Mbps) | Hidden Terminal (5.5 Mbps) |
|---|---|---|---|---|
| Degradation in goodput $1-\frac{Goodput\ with\ anomaly}{Goodput\ with\ no\ anomaly}$ | 0.03 | 0.05 | 0.39 | 0.48 |
| Avg. transmission per data frame $\frac{Total\ frames\ Tx.}{No.\ of\ unique\ frames\ Tx}$ | 1.3<br>$A$(1.42)<br>$B$(1.18) | 1.56<br>$A$(1.87)<br>$B$(1.25) | 1.97<br>$B$(1.57)<br>$C$(2.37) | 2.06<br>$B$(1.78)<br>$C$(2.34) |
| % of data frame that collided | 5.3 | 5.9 | 40.46 | 41.19 |
| % of data frame collisions after preamble | 2.29 | 2.44 | 13.36 | 19.89 |

Table 3.3: Metrics extracted from trace collected for TCP stream tests

SNR at the AP as compared to node C.

Table 3.3 provides a summary of the experimental results comparing the performance degradation caused by capture effect and hidden terminals. The experimental setup consisted of two nodes (either A and B or A and C) generating TCP traffic to the destination node connected on the 100 Mbps Ethernet backbone. Netperf was used as a traffic generator and the payload of the TCP packets was varied from 256 bytes to 1024 bytes in steps of 256 bytes. The experiments were performed with the rate fallback disabled as well as the data rate fixed at 5.5 Mbps and 11 Mbps. For each payload size the experiments were carried out 10 times. The results shown in the table are averages over all the payload sizes.

From the table we observe that in a network consisting of only two nodes, capture effect leads to approximately 5-6% of frames colliding and hidden terminals result in 40-42% of frame colliding. By increasing the number of nodes in the network, the number of collisions would increase, hence further degrading the performance of the network. In [34], the authors present an analytical model to measure the overhead of 802.11 due to collisions and contention in the presence of capture effect. Based on the model and the default 802.11 DCF parameters, the authors conclude that the throughput achieved

of the stock 802.11 protocol is sub-optimal beyond 3 to 4 nodes in the network. This sub-optimality is due to capture effect and time spent in backoff.

To measure the impact at the network layer, we measure the degradation in goodput caused by the anomaly. Capture effect only causes about 3-5% degradation in goodput. However, hidden terminals have a significant effect on the overall performance of the network. This degradation in performance is aggravated at lower data rates because transmissions are longer at lower data rates, increasing the probability of collision. We see approximately 9% drop in performance for the hidden terminal anomaly by changing the data rate from 11 Mbps to 5.5 Mbps. Looking closer at the packet traces, we observe an increase in the retransmissions at the MAC layer. As a metric to measure the number of retransmissions at the MAC layer, we compute the ratio between the total number of data frames transmitted by a station (including retransmissions) and the number of unique frames transmitted. The number of unique frames transmitted are calculated by computing the difference between the start and end sequence number of the trace collected at the MAC layer. We observe a sharp increase in the number of retransmissions at the MAC layer in the hidden terminal case. Along with the increase in retransmission, there is also unfairness involved. For the capture effect, node A (which has a lower SNR at the AP) has a higher number of retransmissions as compared to node B. For the hidden terminal anomaly, node C (which is the low power node and hidden from node A) has a much higher retransmission ratio.

### 3.2.3 Distribution of overlap between colliding frames

As discussed above, both hidden terminals and capture effect are caused due to concurrent transmissions by the stations. Table 3.3 shows the percentage of data frames that collide at the AP due to capture effect and hidden terminals. Based on the analysis at the start of this section, our hypothesis is that collisions due to capture effect should only occur during the first 25-40 $\mu$sec, whereas collisions due to hidden

terminals should not be restricted to this time interval. To test the above hypothesis, we measure the time difference between the start of the two concurrently transmitted frames. To account for variability in the firmware, we extend the interval to the first 100 $\mu$sec.

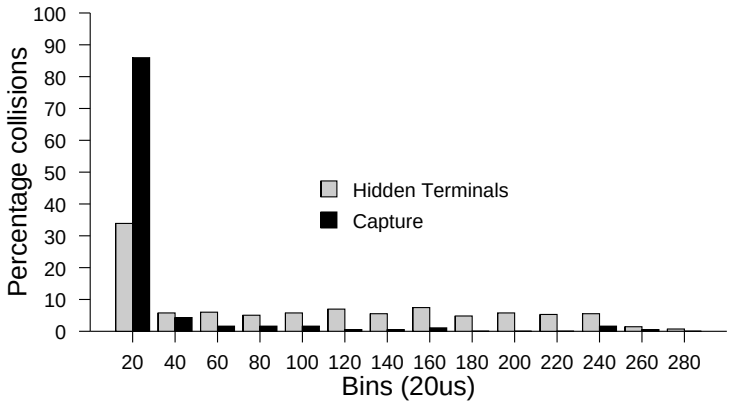


Figure 3.8: Histogram of time difference between the start times of colliding frames

Figure 3.8 shows a histogram of the difference between the start times of the concurrent frames for hidden terminals and capture effect. The payload was 256 bytes and the data rate was set at 5.5 Mbps. From the histogram we observe that for capture effect 85% of the concurrent transmissions have a time difference less than 20 $\mu$sec, whereas for hidden terminals this is not the case. The distribution for hidden terminals has a heavier tail due to collisions occurring during the entire length of the packet as compared to only during the first 25 $\mu$sec. Table 3.3 also states that for capture effect only 2% of the frames transmitted collided after the 100 $\mu$sec interval, whereas for hidden terminals a larger percentage of the frames are transmitted after the 100 $\mu$sec interval.

### 3.2.4 Detection Algorithm

As discussed in section 3.2.3, the primary difference between capture effect and hidden terminals in the network is the extent of overlap between the concurrently transmitted frames. Our network monitoring infrastructure is capable of measuring fine-grained information of every frame transmission from distributed end-points in the network. The detection algorithm leverages off these distributed observations. By recording the timestamps of every frame transmission and ordering them at a central server, the system can accurately distinguish between the two performance degradation anomalies.

Due to the limitations of the current driver design, we use a secondary radio set in monitor mode to record the timestamps of the frames that are transmitted by the primary radio which is associated with the access point. Along with the timestamp, the sequence number, the size of the MAC frame, the transmit data rate and the destination MAC address are also recorded. We assume that the secondary radio knows the length of the preamble used by the primary radio. We present the algorithm first for a simple case where there are only two users associated with the AP. $T_{end1}$ being the timestamp of the data frame at the end of the transmission from station 1 and $T_{end2}$ being the same for frame 2. Using the information about the length of the frame, rate and preamble length we calculate the time at which the frame was transmitted $T_{start1}$ and $T_{start2}$ as $T_{start1} = T_{end1} - (Length_1 * 8/DataRate_1) - Preamble_1$, and similarly for $T_{start2}$. Thus based on the start and end times of the two adjacent data frames we can check whether these are concurrent transmissions or not by using the following simple check; if $T_{start1} \leq T_{start2} \leq T_{end1}$ then frame 2 was transmitted $(T_{start2} - T_{start1})\mu$sec after frame 1 was initiated and vice versa.

The detection algorithm is executed at a central server that maintains a sliding window buffer for the record frames that are received from the the clients in the network.

**Algorithm 1** Detection algorithm for hidden terminals and capture effect

1: For each record **i** in the buffer, calculate the start time $T_{start\_i}$
2: Sort the buffer list based on the start times
3: For every pair of adjacent data frames, check for concurrent transmission
4: If overlap, record the time difference and MAC addresses of the frames under collision
5: If more than 10% of frames collide beyond the 100 $\mu$sec time interval, then hidden terminals, else capture effect

The buffer size is scaled with the number of clients in the network, and is set so as to accommodate 1000 data frames records per client. Since the algorithm is executed periodically, there could be a case where the central server has not received data records from a client in the network. The buffer of records maintained by the central server helps to maintain history information, such that client records that are transmitted after the algorithm is run can still be used in the next iteration of the algorithm. A limitation of the above algorithm is that we are only able to detect collisions between clients that are associated with the same access point, i.e., frames that have the same destination MAC address.

### 3.2.5 Detection Accuracy

Detecting concurrent transmissions requires recording the timestamps of transmitted frames and a global time synchronization protocol across the distributed clients in the network. To synchronize the clocks across the distributed radios, we use the time synchronization protocol specified by the 802.11 protocol [8] and implemented by the Atheros driver. As part of the protocol, the AP embeds a 64-bit micro second granularity time stamp in every beacon frame, and the nodes associated with the AP adjust their local clock based on this broadcasted timestamp. To measure the accuracy of the time synchronization protocol, we measured the error in the timestamps recorded by the distributed clients in our testbed. We measured an error of $\pm 4\mu sec$. This error is sufficient to accurately detect concurrent transmissions. The measured error in the

802.11 time synchronization protocol is consistent with the results presented in [43].

## 3.3 Long term signal strength variations of AP

In this section the impact of long term signal strength variations of the AP and its impact at the MAC and network layer are studied. In the next subsection we present the details of the detection algorithm. The detection algorithm is based on detecting correlated increase/decrease in signal strengths observed at distributed client stations.

Table 3.2 lists the transmit power and receive sensitivity of some of the commonly available WiFi network interfaces from traces collected from residential settings and university campus networks. The table lists the receive sensitivity of the four data rates supported by the 802.11b MAC protocol. With each WiFi interface having a different receive sensitivity, variations in signal strength at the AP would lead to rate diversity in the network. Since the 802.11 media access control mechanism promotes "station fairness", rate diversity leads to a large portion of the medium being used by the lower data rate stations in the network, and hence effectively slowing down the higher data rate clients in the network [33, 30]. Another observation we make from the traces is that the SNR of the clients at the AP vary over a wide range, and a large percentage of the clients that are associated with the AP are in the -70 dBm and -85 dBm range. Thus with clients located at the edge of the communication range, variations in signal strength at the client would cause the network interface to automatically start probing the network for alternate APs.

In this thesis I only focus on detecting long term variations in signal strength of the access points. Although, transient variations due to multipath and fading does lead to rate diversity, the effective long term performance is not degraded. A large number of factors could lead to long term variations in signal strength. These factors could be an obstruction placed on the AP, fading due to a large object placed near the AP, change in transmit power of the AP, antenna of the AP changed/damaged, etc.

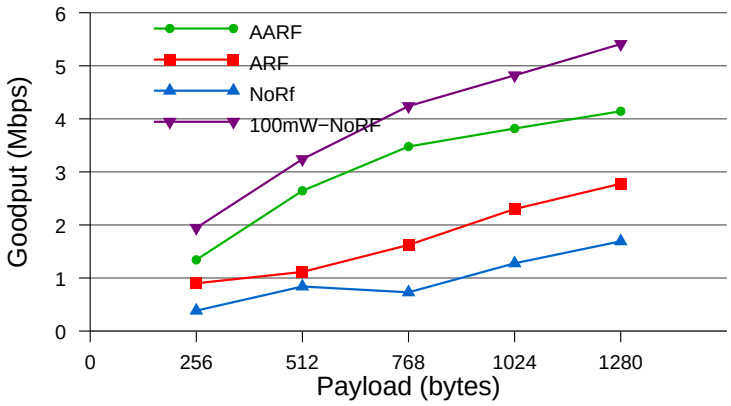


Figure 3.9: Comparison of rate fallback algorithms

To measure the impact of signal strength variations at the network layer, we set up the testbed such that node A is associated with the AP and is at the fringe of the transmission rage of the AP at 5mW. Node A is initiating TCP traffic through the AP to the destination node located on a 100 Mbps Ethernet. The TCP payload was increased from 256 bytes to 1280 bytes in steps of 256 bytes. Two different rate fallback algorithms were tested; Auto Rate Fallback (ARF) and Adaptive Auto Rate Fallback (AARF) .Note that unlike previous experiments, in this experiment the link was not being saturated. This explains the increase in throughput as the payload size increases.

Figure 3.9 shows the degradation in performance as the transmit power of the AP is dropped from 100 mW to 5 mW. When the AP is operating at 100 mW (legend 100 mW-NoRF), node A is well within range of the AP and can communicate at the maximum 11 Mbps with the AP. However, when the transmit power of the AP is stepped down to 5 mW, the client is on the fringe of the transmission range of the AP. In absence of rate fallback (legend NoRF) i.e. the client data rate fixed at 11 Mbps, a large percentage of the ACK from the AP are lost, causing the client to retransmit a large percentage of the data frames. This is because the ACKs are transmitted at the same rate at which the AP received the data frame. Hence, due to the drop in

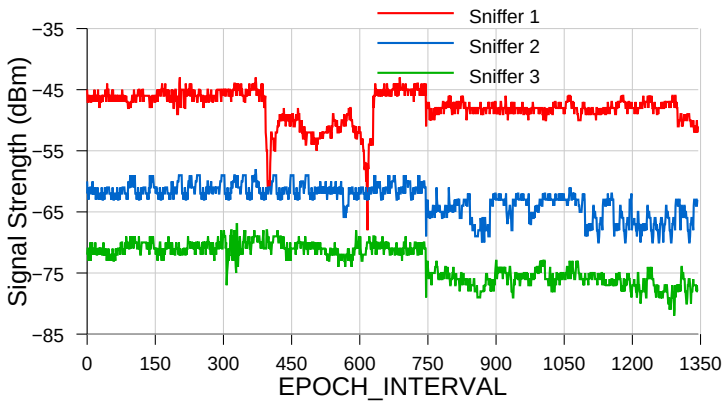


Figure 3.10: Correlated Sensor Observations

transmit power ACKs transmitted at 11 Mbps are not decoded by the client, causing retransmissions. With rate fallback enabled (legend ARF and AARF), the throughput at the client improves. Data frames transmitted at the lower rate are ACKed by the AP at the same rate, and hence the client can receive the ACKs. This decreases the percentage of frames that are being retransmitted.

### 3.3.1 Detection Algorithm

In this section we present the details of the algorithm to diagnose anomalous signal strength variations. The algorithm is based on detecting correlated increase or decrease in signal strength of the AP at distributed sniffer locations.

Figure 3.10 shows a time slice of the signal strength observations of an AP measured at three distributed sniffers. From the trace we observe that between time interval 350-600, sniffer 1 measured a drop in signal strength, whereas sniffer 2 and 3 do not measure the same drop in signal strength. This drop in signal strength was caused due to localized fading at sniffer 1. However at time 750, all the sniffers measure a concurrent drop of 3 dB in signal strength. This was caused due to a power control event at the AP, and hence is observed by the distributed sniffers. Thus, signal strength

variations observed at a single sniffer is not sufficient to differentiate between localized events like fading and global events like change in transmit power at the AP. To reduce the number of false positives caused due to localized events, multiple distributed sniffer observations are required to detect anomalous variations in signal strength at the AP.

Based on extensive experiments carried out in a typical office environment, we observed that three distributed sensors observations are sufficient to detect correlated changes in signal strength. The standard Pearson's Product Moment correlation coefficient ($\rho$) is used as the statistical tool to detect concurrent changes in signal strength. The correlation coefficient can take on any values in the range $-1 \leq \rho \leq +1$. A $\rho$ close to +1 indicates that a concurrent drop/increase in signal strength was observed simultaneously at all sensors, and a $\rho$ close to -1 indicates that there was a drop at one sensor and an increase in signal strength at the other. Any variation in signal strength of the AP would result in a high positive $\rho$.

Each sniffer computes the average of 10 consecutive beacon frames over a period of EPOCH_INTERVAL and transmits the information to a central server. On receiving this time series data from multiple sniffers in the network, the central server computes an intersection of the time series data received from a subset of these sniffers. The intersection of the time series ensures that $\rho$ is computed over observations of the same set of beacon frames. The server calculates the pair-wise $\rho$ between each pair, over a sliding window of size 20.

With three sniffers in the network, there are three pairs of $\rho$. To reduce the chance of a spurious peak in the correlation coefficient due to identical multipath observed at a pair of sniffer, the average of the correlation coefficients from the different sniffers is computed.

Averaging the correlation coefficients has the same effect as a low pass filter which filters out spurious peaks in the correlation coefficient. Thus, only correlated signal strength variations that are globally observed across all the distributed sensors

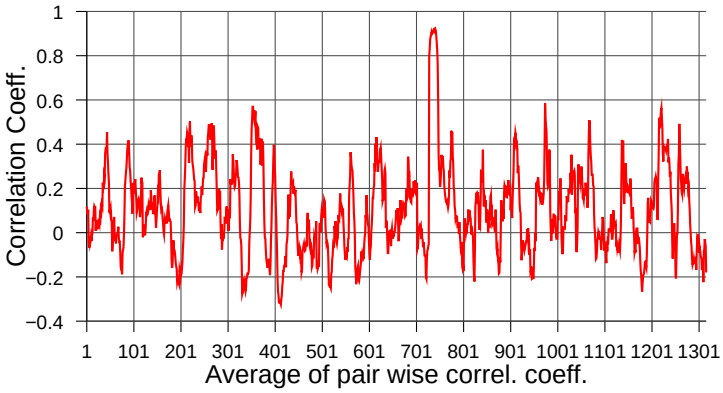


Figure 3.11: Averaged correlation coefficient. Averaging eliminates the spurious peaks and magnifies only the peak that is observed across all the pairs of sniffers

would be magnified and all the other spurious peaks are suppressed. Figure 3.11 shows the average correlation coefficient computed for the signal strength trace in Figure 3.10. As seen from Figure 3.11, at about the same time at which the transmit power of the AP was reduced, the average correlation coefficient rails high and approaches 1. We use a fixed threshold of 0.8 to detect an anomalous signal strength increase/decrease at the AP.

The final part of the algorithm is the selection of sniffers to compute the pairwise correlation coefficients. If the selected sniffers are collocated next to each other, it is possible that each sniffer observes the same local variations in signal strength, and hence falsely triggering an anomalous signal strength fault. Hence, it is desirable to select sniffers such that they are spatially spread out over the network. As compared to random selection of sniffers from the network, our system sorts the clients based on the average SNR that is reported by them. This sorted list is divided into N equal sublists (N = number of sniffers) and a sniffer is randomly selected from each sublist. This cluster-sampling of sniffer stations reduces the possibility of selecting sniffers that are co-located next to each other, and hence reduces the possibility of a false positive.

### 3.3.2 Detection Accuracy

To test the accuracy of the detection algorithm presented above, controlled experiments were carried out in a typical office environment as well as an open lobby. Distributed sniffers were deployed to measure the signal strength of the beacons transmitted by the AP. Using the six different power levels of the Cisco Aironet AP's (20, 17, 15, 13, 7 and 0 dBm) the transmit power was changed once every 5 minutes in steps of 2, 3, 4, 5, 6, 8, and 10 dBm. Correlating the signal strength observations at the distributed sniffers, the false positives and negatives of the detection algorithm was measured. Table 3.4 shows the true positives, false positives and false negatives as a percentage of the total number of events triggered by the detector. The detection threshold was increased from 0.6 to 0.9. For sake of brevity, the table shows the results for two low power changes (2 and 3 dBm), and two high power changes (6, 7 dBm).

| Power (dBm) | Threshold | True Positive | False Positive | False Negative |
|---|---|---|---|---|
| 2 | 0.6 | 16 | 12 | 72 |
| | 0.7 | 11 | 7 | 82 |
| | 0.8 | 0 | 0 | 100 |
| | 0.9 | 0 | 0 | 100 |
| 3 | 0.6 | 50.3 | 10 | 39.7 |
| | 0.7 | 33.33 | 4 | 62.67 |
| | 0.8 | 16 | 0 | 84 |
| | 0.9 | 4.5 | 0 | 95.5 |
| 6 | 0.6 | 100 | 0 | 0 |
| | 0.7 | 100 | 0 | 0 |
| | 0.8 | 100 | 0 | 0 |
| | 0.9 | 51 | 0 | 49 |
| 7 | 0.6 | 100 | 0 | 0 |
| | 0.7 | 100 | 0 | 0 |
| | 0.8 | 100 | 0 | 0 |
| | 0.9 | 83.33 | 0 | 16.67 |

Table 3.4: Detection accuracy (percentage) of signal strength variations at the AP. A correlation threshold of 0.8 is selected.

A high true positive rate indicates that the algorithm is able to correctly detect

the faults in the network, and correct the fault by applying the remedy. A high false positive rate indicates that there are spurious alerts generated. This could cause the network to become unstable by having the clients to constantly switch between APs. A high false negative rate indicates that the detection algorithm is unable to detect the underlying faults in the network. As discussed above, not detecting the signal strength variations of the AP leads to performance degradation.

A number of key observations from the above table are made. First, the detector is able to detect changes in transmit power only when the change is greater than the variations in signal strength due to multipath and fading. Hence, the detection accuracy is low for low power changes of 2 and 3 dBm. However, for higher power changes the detection accuracy increases. Second, the correlation coefficient is dependent on the magnitude of change in power. For example, as the change in power is increased, the number of true positives detected at a threshold of 0.9 also increases. Third, the smallest threshold that results in no false positives and negatives is 0.8. Hence, the detection threshold is set to 0.8 by trading off detection of small changes in power to accuracy of detection.

## 3.4 Remedies for network problems

Table 3.5 provides a summary of the remediation performed by the stock 802.11 drivers and the remedies proposed after diagnosing the root cause of the performance degradation. Due to the lack of visibility of the underlying physical layer, existing 802.11 drivers perform rate fallback as the default remedy for every fault. Existing 802.11 drivers trigger rate fallback when an excessive number of retries are caused at the MAC layer. However, applying rate fallback as the default remedy to troubleshoot every fault leads to significant degradation of the performance of the network[2] .

[2] The 802.11 protocol has support for avoiding hidden terminals using RTS/CTS, but this is not an adaptive mechanism in the standard.

| Anomaly | Existing 802.11 remediation | Informed remediation |
|---|---|---|
| Hidden terminals | Rate fallback | Increase transmit power or enable collision avoidance |
| Capture effect | Rate fallback | Increase or decrease tx. power |
| Noise | Rate fallback | Switch channel or associate with alternate AP |
| Signal strength variations | Rate fallback | Associate with alternate AP |

Table 3.5: Faults converge to degraded performance due to rate fallback. Table also shows the existing 802.11 based remediation and informed remediation based on root cause analysis.

To measure the impact of 802.11 based remediation, a planned network was monitored for a day. The network consisted of 4 APs with an average of 14 clients associated with each AP. The MAC and the IP headers were captured by a single sniffer, and the traces were analyzed offline. From the traces we observed that there is a significant amount of rate diversity as well as rate fluctuation in the network (see Figure 3.12). Most clients are operating at the minimum 1 Mbps data rate, causing unfairness to the higher data rate clients. Analyzing the data rate of the client transmissions, we observed that on an average only 15% of the data frames are transmitted at the highest possible rate. By averaging the received signal strength readings of the beacon frames over a time interval of 15 mins, we observed that there are large (10-15 dB) short term as well as long term variations in signal strength at the AP. This variation in signal strengths leads to exacerbating the rate diversity in the network.

### 3.4.1 Joint Optimization

In this section I propose that diagnosis of the root cause of the fault leads to informed efficient remediation as compared to the default rate fallback based remediation currently performed. By having stations and access points in the network mutually

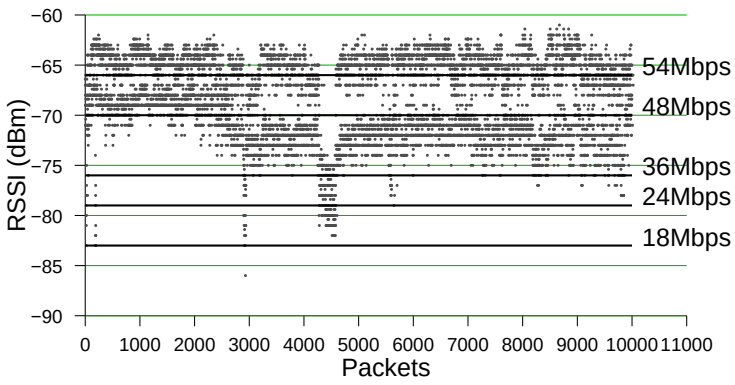


Figure 3.12: Variability in signal strength in an open lobby

share their observations of network events, the system can coordinate these observations to jointly optimize network performance. These are **joint** optimizations because they involve changes at multiple stations or are commanded by the AP. We address these issues in the order presented in Table 3.5.

**Hidden Terminals**: The default 802.11 based remediation of rate fallback does not solve the hidden terminal problem. Instead of rate adaptation, I propose simply stepping up the transmit power of the hidden terminals. This allows them to maintain the data rate and avoid collisions at the receiver. Alternatively, the hidden terminal could initiate RTS/CTS mechanisms, but that comes at a cost of extra control messages for every data frame. The "hidden" node can not independently sense this condition – it must be informed of the problem and likely remedy by the AP.

**Capture Effect**: The simultaneous transmissions are caused due to the 25-30 $\mu$sec interval required to sense the channel. Although this delay cannot be eliminated, we can ensure that the SNR of the frames being received at the receiver is approximately the same, and hence eliminating the unfairness caused due to capture. The default 802.11 based remediation of rate fallback does not reduce the unfairness and further degrades the performance of the network due to rate diversity in the network. As a

remedy, I propose stepping up the transmit power of the node with the lower SNR to avoid the performance degradation and unfairness due to capture effect.

**Noise:** Having detected a rise in the noise floor, instead of invoking rate fallback, I propose that switching the frequency channel to an alternate less noisy channel is a smarter remedy. If other AP's are nearby, the affected station can re-associate to another AP using a different channel. Alternatively, the affected station can request that the AP change the frequency using a new MAC mechanism.

**Signal Strength:** Signal strength variations of the AP transmissions causes excessive rate adaptations performed by the clients in the network, and enabling rate fallback at the client interface does not remedy the problem. On detecting excessive retransmissions due to signal strength variations, I propose disassociating with the AP to an alternate AP or requesting that the AP increase signal strength using a new MAC mechanism.

### 3.4.2 Implementing Remedies

Using the testbed, our initial experiments show that these individual improvements improve performance. However, we do not have a full system evaluation that combines all mechanisms. Implementing some of the remedies requires modifying the 802.11 MAC protocol to implement new control messages. Initial implementations of the remedies are done using the SoftMAC framework [49] which had been developed to implement a control mechanism to complement our existing detection system.

The above listed remedies are some of the simplest remedies that could be performed to tolerate faults in the network. The remedies proposed require no explicit federation between the multiple AP's in the network. As part of future directions I plan to propose remedies that involve coordination between multiple AP's and clients in the network to troubleshoot the fault. I also plan to deploy the complete system in a production network consisting of a large number of APs and heterogeneous clients and

measure the performance improvement achieved by diagnosing the faults and applying efficient remedies.

## 3.5 Summary of indoor 802.11 detection algorithms

To summarize, in this chapter I have presented the design, implementation and evaluation of detection algorithms for the most commonly observed faults in indoor 802.11 deployments. A key feature of the system is that, unlike existing systems, it is able to distinguish between the root causes of performance degradation. This is enabled by collecting fine-grained observations from the distributed end-points in the network. Efficient metric aggregation is performed at the distributed end-points to prevent an overload on the system by transmitting raw packet level details.

A key feature of the system is that the metrics used at the PHY layer to diagnose the faults are independent of each other; this means that each detection algorithm is uniquely attributed to a unique PHY layer metric, and each PHY layer metric triggers a single detection algorithm. For example, the presence of hidden terminals in the network would not lead to a long term increase in the noise floor or change the signal strength at the AP. Hence, each fault can be independently diagnosed and does not depend on the presence/absence of the others. No specific ordering is required to detect the faults, and the faults could be diagnosed in parallel.

I have implemented the algorithms and remedies on a small scale testbed. The stock Atheros driver was modified to collect fine-grained diagnostic information from the PHY layer. Based on the information collected, threshold based detection algorithms were implemented to detect noise/interference, hidden and capture effect and signal strength variations at the AP. Noise in the network is diagnosed by detecting an increase in the noise floor. Based on the calibration of the Atheros chipset noise floor register, we set the interference detection threshold to -65 dBm. Hidden terminals/capture effect are diagnosed by detecting concurrent transmissions by the clients in the network. By

measuring the overlap between two simultaneously transmitted frames, we are able to differentiate between hidden terminals and capture effect. Signal strength variations at the AP are diagnosed by detecting concurrent changes in signal strength recorded at the distributed sniffers. Based on experiments carried out in a typical office environment and open lobby, the algorithm is able to accurately detect signal strength changes greater than 4 dBm using a correlation threshold of 0.8.

The main contribution of this work is that it takes the first step towards building truly self-healing wireless networks and provides detailed information for troubleshooting faults originating at the physical layer. Although the list of faults presented in this chapter are not exhaustive, the completed system addresses the most commonly observed faults in indoor 802.11 deployments that are addressed by the research community. As part of future work, we plan to extend the list of faults and categorize the faults and detection techniques.

# Chapter 4

## Motivating WiFi based Long Distance Networks

Many developing regions around the world are in dire need for low-cost connectivity solutions to provide network coverage. These regions have low telephone penetration rates (roughly 2% in Africa) [38], and rural areas with their low user density cannot support the cost of cellular basestations or fiber (unlike urban areas). Figure 4.1 shows the percentage of population having access to the Internet across the world. As seen from the map, a majority of the world has less than 10% of the population online.

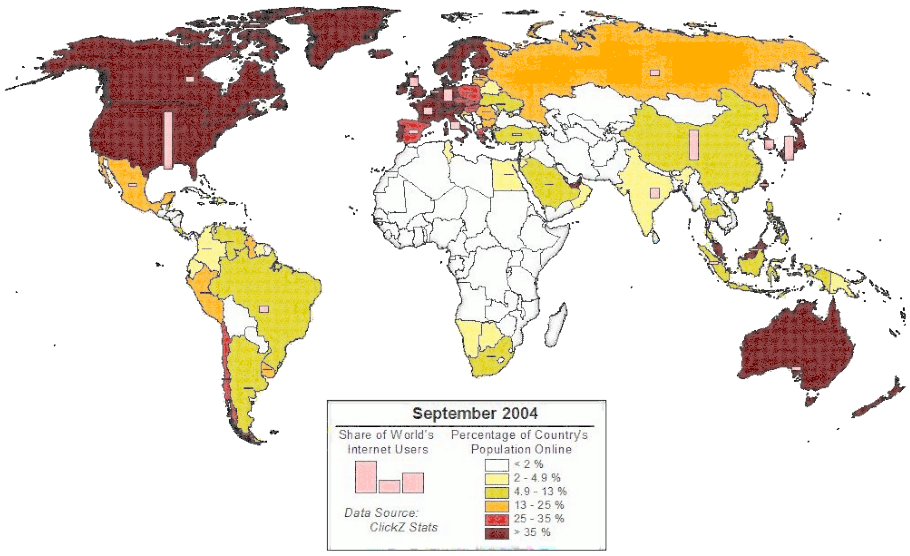


Figure 4.1: World map showing percentage of population online

The evolution of providing network connectivity in these regions of the world is taking quite an alternate route from the traditional networks we observe in the indus-

trialized world. Many large cities in East Africa now have a large number of towers supporting a wide range of different long-range wireless technologies such as microwave, WiFi, WiMax and other commercial wireless broadband solutions. African countries see better opportunity in wireless options for regions that have low penetration of fiber and other wire-line connectivity solutions; many of these countries have higher cellphone penetration rates than fixed-line penetration [38].

The primary reasons for the boom in the use of wireless networks in developing countries are:

**Lower cost:** In developing countries, wire-line connectivity solutions are not economically viable in low-user density areas [48]. Satellite links, a common mode of Internet connectivity in much of Africa, are also very expensive and not widely affordable (typically US$2,000 per month for 1 Mbps). Establishing wireless distribution networks (microwave, WiMax, WiFi-based or CDMA450) to extend coverage within a region requires a much lower capital investment. This allows for decentralized rapid evolution of such networks by local entrepreneurs. Among different wireless options today, WiFi-based networks are **currently** much more economically viable than WiMax, CDMA450 and microwave.

**Ease of deployment:** Wireless networks are relatively easy and quick to deploy, particularly in cases where we do not need new towers. Networks in unlicensed spectrum are preferred because they can be set up by grass-roots organizations as needed, avoiding dependence on a telecom carrier. This is particularly important for rural areas, which are less enticing to carriers due to the low income generation potential,

**Intranet usage:** Providing network access does not necessarily have to be associated with Internet access. In many developing regions, basic local communications infrastructure is absent. A wireless network within a city or a district can enable a wide range of applications including telephony, essential services and health care. For example, we have deployed an intranet network in southern India between hospitals and

rural vision centers that supports rural telemedicine.

Despite such a phenomenal growth in the adoption of wireless networks in developing regions, there have been very few research efforts that take a concerted view towards analyzing how to build such networks. The primary difference between urban environments in developed countries with a majority of regions in the developing world (with the exception of highly populated cities) is the **density of users**. We argue that prior work on wireless mesh networks [10] is best suited for urban environments with high user densities. At lower user densities, the type of wireless network best suited to provide coverage is significantly different from the mesh networking model; such a network would consist of nodes with directional antennas and point-to-point wireless links.

## 4.1 Point-to-Point WiFi based Long Distance Networks

In this section, we begin by contrasting low user density (rural and semi-urban) and high user density environments (urban) and make the case for point-to-point long distance wireless networks using directional antennas in low-density environments. We do so by pinpointing why other well-known wireless technologies (VSATs, cellular, mesh networks) are not economically viable in low-density environments.

| Characteristic | High User Density | Low User Density |
|---|---|---|
| Connectivity requirements | Full coverage required | Islands connected to each other |
| End Devices | Individual, mobile, low power budget and non-LOS | Shared, fixed, high power and LOS |
| Topology | Star-topology or mesh network | Point-to-point with end points within the network |
| Applications | Mainly Internet access | Internet as well as peer-to-peer Intranet access |

Figure 4.2: Characteristics of Low Density and High Density networks

Figure 4.2 lists some of the fundamental differences between providing wireless connectivity in high user density and low user density environments. These differences mainly stem from the constraints of providing **low cost** wireless connectivity with small per-user cost and minimum or no recurring cost.

In low density environments people are usually clustered around small localities (e.g. villages), with large distances among these clusters. Even within villages the user density is low compared to urban areas. In addition, the typically lower incomes lead users to share computer terminals (e.g. Internet kiosks) to amortize the relatively high cost of the devices and network connection.

Satellite networks provide fantastic coverage, but are very expensive. VSAT equipment installation costs over US$10,000 with a recurring monthly cost of over US$2,000 for a 1 Mbps link. In low user-density regions, VSAT is affordable only for businesses or wealthy users.

Networks with a base-station model such as WiMAX, and cellular networks like GPRS and CDMA, have an asymmetric design philosophy where expensive base stations are amortized by large number of cheap clients over many users. In low-density regions, such base stations simply do not cover enough users to be economically viable. The expectation that cellular solves the connectivity problem for developing regions is thus somewhat of a myth: cellular success in developing countries is an urban phenomenon, with a few exceptions. Bangladesh has good rural coverage because it is actually a very high density country, and base stations that cover roads and rail lines also cover many villages. China has dictated good coverage as policy, despite the economic issues. Other countries either subsidize rural users through taxation, much like the US universal access tax, or require some rural coverage as part of spectrum allocation. Thus, many cellular providers incur losses in low user-density regions and partially recoup these losses by either charging very high usage rates or imposing a universal service charge on all users. In its intended deployment model, with expensive basestations covering many users,

WiMax also shares the shortcomings of other cellular technologies.

Finally, 802.11 mesh networks [10], also assume high user density. Moreover, mesh networks suffer from two basic problems when scaled to larger areas. First, as the network grows, an increase in the number of APs with omni-directional antennas leads to increased interference in overlapping cells. Second, the use of low-gain omni-directional antennas increases the hop length, and as a result throughput decreases. Bicket et al. [20] show that in Roofnet, longer routes (traversing multiple wireless hops) are disproportionately slower mainly due to inter-hop collisions.

Thus, we argue that for low density of users, approaches that provide full coverage are not feasible. The alternative would be to cover only those few places where connectivity is required, by employing long-distance point-to-point wireless links. Such links can rely on WiFi, point-to-point WiMax, or other technologies that support long-distance links offering reasonable bandwidths. In choosing such a technology, the most important factors are cost and configurability. An interesting case are environments that have a mix of low and high user density regions. Here, a combined approach where the mesh network is augmented by point-to-point links as required can also be considered ([28]).

## 4.2 Hardware Selection

Until now, for practical and cost-related reasons, we have chosen to examine the possibility of using WiFi-based Long Distance (WiLD) links. WiFi cards are cheap and highly available, enjoying economies of scale. In our existing WiLD deployments, the cost of a WiLD link is approximately $800 (excludes the cost of tower) with no recurring cost. Because they operate in unlicensed spectrum, WiLD links are easy to deploy and experiment with, and spectrum license costs are eliminated. Manufacturers of WiFi chipsets (e.g. Atheros) often support open-source drivers, allowing us to completely subvert the stock 802.11 MAC protocol and tailor the protocol to meet our needs.

An alternative would be to use point-to-point WiMax links; such links would have a few important advantages over WiFi: configurable channel spectrum width (and consequently datarate), better modulation (especially for non-line of sight scenarios); operation in licensed spectrum would permit higher transmit power, and thus longer distances and better signal strengths. However, existing commercial WiMax products are only tailored for cellular providers and do not support point-to-point mode of operation. Existing WiMax hardware is more expensive than WiFi (about $10,000 for basestations), and the high spectrum license costs in most countries dissuade grass-root style deployments. Currently it is also very difficult to obtain licenses for experimental deployment and we are not aware of open-source drivers for WiMax basestations and clients (Wavesat offers a mini-PCI based WiMax client development kit [66]).

## 4.3 Summary

This chapter proposes the rethinking of the research agenda to provide network connectivity in low density developing regions of the world. The chapter highlights the primary differences in providing network connectivity in dense urban areas and low density rural areas. It also points out the limitations of existing solutions and proposes the adoption of WiFi as the underlying radio technology to provide point-to-point network connectivity. The rest of the thesis focuses on the use of WiLD links as the currently preferred solution; however, research investigating long-distance point-to-point wireless networking should be (for the most part) agnostic to the specific underlying wireless technology being used, allowing for other solutions to be used as they become available.

# Chapter 5

## Identifying sources of packet loss in WiFi based Long Distance Networks

Unlike indoor 802.11 WLANs, the root sources of performance degradation have not been characterized in outdoor WiLD networks. This chapter presents a detailed measurement based study and identifies the root sources of performance degradation in outdoor WiLD networks. Having identified the root sources, Chapter 6 presents software based remedies to mitigate the performance degradation. Chapter 7 addresses the limitations of the software based remedies and discusses smart antenna based remedies to mitigate the loss.

WiFi-based Long Distance (WiLD) networks [27, 37] are emerging as a low-cost connectivity solution and are increasingly being deployed in developing regions in both urban[1] and rural settings. The primary cost gains arise from the use of very high-volume off-the-shelf 802.11 wireless cards, of which over 140M were made in 2005. These links exploit unlicensed spectrum, and are low power and lightweight, leading to additional cost savings [21].

Many outdoor short-range WiFi-based networks are being deployed as multihop mesh networks in urban areas( [10, 17, 60]). Roofnet [10], for example, is a 38-node network deployed within a small area (∼6 sq. km), where the median link length is 0.5 km, the longest is 2.5 km, and most links are not line-of-sight (LOS). Each node has

[1] In urban regions in Africa, satellite-based Internet providers use WiLD networks as a distribution network to reach out to the end-users within the region.

one radio with an omnidirectional antenna (8dBi gain, 20-degree beam height).

In contrast, WiLD networks use multihop point-to-point links, where each link can be as long as 10–100 km. To achieve such long distances, each node uses high-gain directional antennas (24dBi, 8 degree beam-width). The two endpoints of each link have direct LOS, in addition to the high-gain antennas, which ensures strong received signal strength. Additionally, in multihop settings, nodes have one radio per fixed point-to-point link to each neighbor, which can operate on different channels as needed.

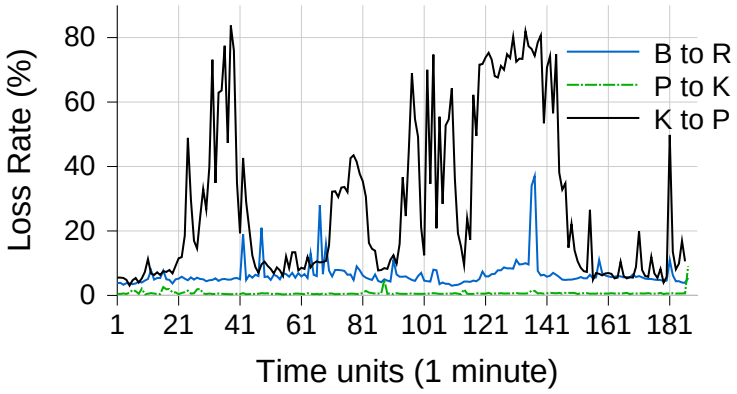


Figure 5.1: Packet loss variation over a period of about 3 hours

Despite the promise of low-cost connectivity, the performance of WiLD networks in the real world has been abysmal. This poor performance is primarily triggered by the high loss variability observed on WiLD links. Figure 5.1 shows the loss rate measured over two of our links ("K-P" and "B-R") over a period of 3 hours on different days. The points K, P, B and R represent the locations where the WiLD end-points are located. The loss rate was averaged over 30-second intervals for a 1 Mbps unidirectional UDP CBR traffic flow with the MAC-layer ACKs turned off and retries set to zero.

The two main characteristics that we observe are: 1) WiLD links demonstrate high variability of loss rate; and 2) the loss rate can be highly asymmetric across a link. Bursts vary in magnitude as well as duration. For example, on the K to P link,

loss bursts ranged in magnitude from 15–80% and the duration of bursts also varied from a transient high burst to a long burst lasting over 25–30 minutes. In contrast, the reverse path (P to K) had almost 0% loss for the entire duration. In addition to the high variability of the loss rate, there is also a residual loss that is always present and remains constant over long time periods. This residual loss ranges between 0–10% and varies with each link. Although Figure 5.1 shows only two links in our testbed, the above behavior is characteristic of all urban links. In contrast, our rural links consistently show loss rates close to zero with a maximum of less than 2%. We explore these differences further and point out that many WiLD links have one end in an urban area. In addition, the losses shown here are only those due to the channel; the 802.11 protocol itself also induces losses.

## 5.1 Overview of packet loss

This chapter presents a detailed measurement study to analyze the sources of packet loss in WiLD network settings. We categorize the sources of packet loss into two broad categories: (a) **channel losses** induced by the long distance wireless channel; (b) **protocol-induced losses** due to the 802.11 MAC protocol. The study is based on a real-world WiLD network deployment consisting of 6 links with lengths varying from 2–20 km. Unlike existing WiLD deployments [56], the testbed includes both rural and urban links. In addition to the real deployment, we also perform detailed experiments using a wireless channel emulator, which enables repeatable controlled experiments [41].

The three main contributions of this work are:

**Channel loss characterization:** We analyze three well known causes for channel losses in wireless environments, namely, **external WiFi interference**, **non-WiFi interference** and **multipath interference**. Among these, we show that external WiFi interference is the most significant source of packet losses in WiLD environments and the effect of multipath and non-WiFi interference is not significant. This is in contrast

to the results of Roofnet network [10] where the authors observed multipath to be the most significant source of packet loss.

**Protocol-induced losses:** The stock 802.11 MAC protocol is ill-suited for WiLD links due to the breakdown of CSMA over long distances and propagation delays. Here, we pinpoint the fundamental shortcomings of the 802.11 MAC protocol.

**Loss remedies:** Having identified external WiFi interference as the primary source of losses in WiLD links, we discuss four potential remedies to mitigate these losses: (a) frequency channel adaptation; (b) rate adaptation; (c) adaptive FEC and (d) bulk acknowledgements. We evaluate the effectiveness of each of these remedies and also implement and evaluate adaptive FEC and bulk acknowledgements.

The focus of our packet loss characterization study is significantly different from other wireless-based loss measurement studies [10, 59]. The work done by Raman et al. [24] is the only other measurement-based study of WiLD deployments of which we are aware. However, the two studies are orthogonal: we focus on determining the impact of different sources of losses and remedies for loss alleviation, their work focused more on performance analysis of 802.11 network at various layers in the network stack and the effect of other parameters (weather, SNR, payload, datarate) on loss variability. Our work also differs from mesh networks like Roofnet [10] in that WiLD networks, as we show, have very different loss characteristics, with loss much more due to external interference than multipath effects.

## 5.2 Experimental methodology

The variability of loss rate in WiLD links has important implications for the design of loss-sensitive protocols like TCP and delay-sensitive applications like video and voice. To understand WiLD links, three techniques are combined: a WiLD testbed with both urban and rural links, a wireless channel emulator for repeatable controlled experiments, and analytical models for loss variability.

| Link | Distance (km) | Environ. | Antenna height(m) |
|---|---|---|---|
| K-P | 20 | Urban | 50 |
| B-R | 8 | Urban | 30 |
| M-P | 2 | Urban | 40 |
| T-A | 11 | Rural | 20 |
| T-S | 13 | Rural | 25 |
| W-N | 15 | Rural | 20 |

Table 5.1: Some of the urban and rural WiLD link in the testbed

We perform our packet loss characterization measurements on a WiLD network testbed comprising of links in both rural and urban environments. Table 5.1 summarizes some of the urban and rural links in our deployments. The links range from 2–20 km in length. The minimum SNR from all the above links was 25.

The two main characteristic of WiLD links that differentiate them from links in a multi-hop urban mesh deployment [10] are the long distance between the two endpoints, and the use of high-gain directional antenna (24 dBi, 8 degree beam-width). The two endpoints of each link have direct LOS, in addition to the high-gain antennas, which ensures a strong received signal strength. In multihop settings, nodes have one radio per fixed point-to-point link to each neighbor, which can operate on different channels as needed.

In addition to the testbed, we also use a wireless channel emulator (Spirent 5500 [63]) to study each source of packet loss in isolation. The emulator allows us to place the two ends of the link in separate RF-isolated boxes (-80dB) and then simulate in real time the RF channel between them. The Spirent 5500 accurately emulates radio channel characteristics with channel loss, fast and slow fading and delay spreads. This enables us to emulate links of any length or loss profile with repeatable results. We perform tests by connecting the channel emulator to the same radios used in our WiLD deployment.

Using the WiLD testbed and the channel emulator, we explore two categories of loss: **channel losses** induced by the wireless channel and **protocol-induced losses** by the 802.11 MAC protocol. Specifically, for channel-induced losses we investigate: a) External WiFi interference, b) External non-WiFi interference and c) Multipath interference. The absence of any mobility of the end points and high SNR eliminate fading and path loss as possible sources of packet loss. For 802.11 protocol induced losses, we investigate: a) Timeouts due to propagation delay, and b) Breakdown of CSMA over WiLD links.

Our experimental methodology is based on collecting fine-grained information from the MAC and the PHY layers without the use of any extra hardware. All our results are based on using a UDP CBR traffic source. Unless otherwise stated, for all our experiments we turn off MAC-layer ACKs and set the maximum retries limit to zero. This allows us to measure the real channel loss rate in absence of any MAC-layer acknowledgments and retries.

We instrument the stock Atheros based 802.11 driver to log fine-grained information for each frame received and transmitted. In addition to capturing all the frames on the link, to evaluate the effect of external WiFi interference, we also capture and log frames being transmitted by external WiFi sources. This is achieved by creating a virtual network interface set in "monitor mode" on the same channel as the primary interface. This technique is equivalent to using two physical network interfaces, one being the primary and the other a passive monitor. We also modify the Atheros driver to pass up frames with CRC and PHY errors.

By using a controlled traffic source we are able to compare the payloads of the transmitted and received frame. This enables us to analyze the cause of frame loss (PHY error or CRC error) as well as the extent to which the fame is corrupted for the different sources of packet loss.

To summarize, we collect the following information for every frame: complete

802.11 MAC header and IP payload, received signal strength, data rate used to transmit the frame, timestamp of the frame, frames containing PHY and CRC errors, and the noise floor immediately after the frame is received.

## 5.3 External WiFi interference

In this section, we investigate external WiFi interference as a potential source of packet loss in WiLD links. Any WiFi traffic that is not a part of the primary WiLD link is categorized as external WiFi interference. Based on the measurements performed on our WiLD testbed and the wireless channel emulator, we show three key results:

- In the presence of external WiFi interference, the loss rate is strongly correlated with the amount of external traffic received on the same and adjacent channels. In contrast, due to the omni-directional antennas used in the Roofnet deployment [10], there was no such strong correlation observed.
- Packet loss due to external WiFi interference is far more significant in WiLD deployments than local mesh networks.
- The loss due to external WiFi interference depends on the relative power level between the primary and external traffic, their channel separation, and the rate of external interference.

### 5.3.1 Correlation of loss rate and external WiFi traffic

Figure 5.2 shows the loss rate across all (rural and urban) the WiLD links. We observe that the loss rate of the urban links vary across a wide range (4–70%). In contrast, all the rural WiLD links have a very small loss rate. The maximum loss rate observed in all our rural WiLD links was 1.7%.

To study this contrast between the rural and urban links, we collected detailed packet level MAC traces. By parsing the MAC header source and destination fields, we

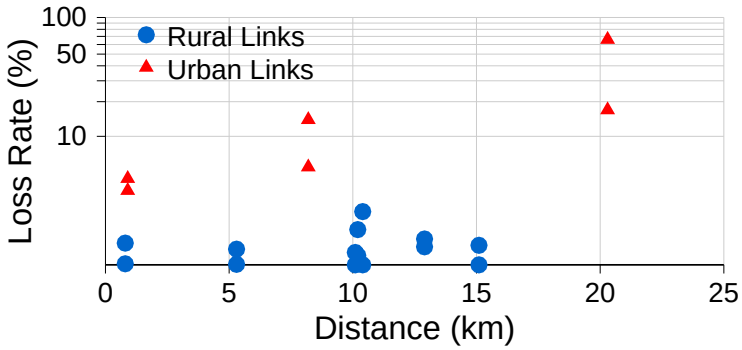


Figure 5.2: Scatter plot of loss rates observed in links deployed in urban and rural areas (note: loss rate is plotted in logscale)

are able to count the number of frames received from external WiFi sources (interference). In the traces collected over all our rural links we do not capture any external WiFi traffic. However, significant amount of external WiFi traffic was captured from the traces collected in the urban WiLD deployment.

Figure 5.3 shows a scatter plot between the loss rate and the absolute number of external WiFi traffic frames received on an urban link (K $\rightarrow$ P) for a period of 6 hours. The figure shows that a subset of the loss rate samples are strongly correlated with the external traffic. For the other subset of the samples, the loss rate increases even when there is no significant increase in WiFi traffic on the same channel.

To investigate this further, we perform a controlled experiment using the wireless channel emulator. To model interference from an external traffic source, along with the primary link traffic we introduce a controlled interference source at the receiver. The traffic rate of the interference source was varied from 0.1 to 1 Mbps and the traffic rate on the primary link was kept fixed at 5 Mbps. The data rate was fixed at 11 Mbps on both links. Figure 5.4 shows a scatter plot of the loss rate and the total number of frames received from the external interference source. From the graph, we observe that for a given loss rate, the amount of external traffic captured by the monitor device

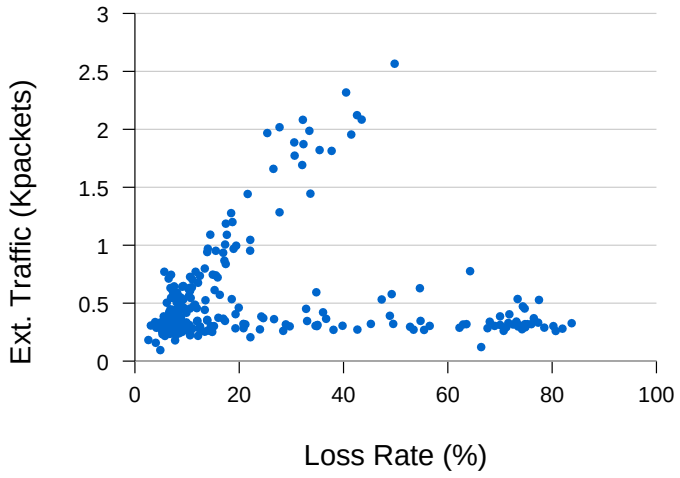


Figure 5.3: Loss rate vs. ext. traffic observed on WiLD link

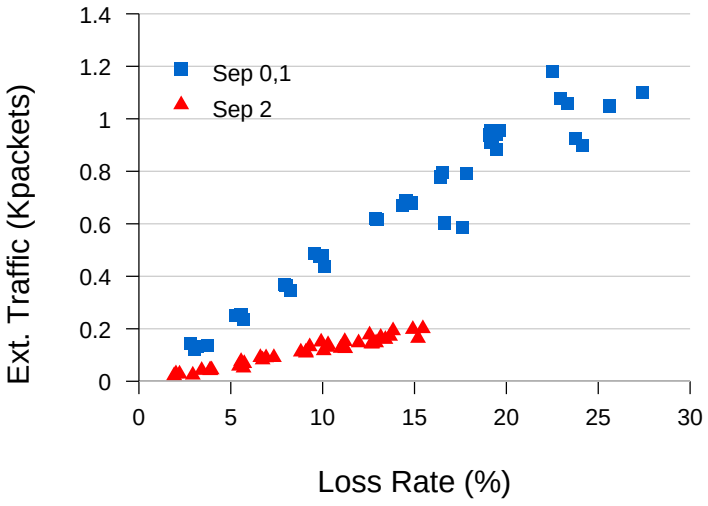


Figure 5.4: Loss rate vs. ext. traffic observed in wireless emulator

depends on the channel separation of the primary and interference source.

The above observed trend is the same as that in Figure 5.3. At a channel separation of 0 and 1, the receiver can receive both the primary link traffic as well as the frames from the interference source. Hence, the loss rate is directly correlated with the

amount of external WiFi traffic captured by the monitor interface. At a channel separation of 2, the receiver is not able to receive the frames from the external interference source. However, the signal spillage of the interference source in the primary channel is sufficient to cause frame corruption. From the traces we observed that almost 100% of the lost frames contained CRC errors.

### 5.3.2 Effect of hidden terminals in WiLD networks

Unlike WiLD deployments, where we have observed significant correlation between loss rate and external interference, it has been observed that there is no significant correlation in outdoor mesh-network deployments (Roofnet [10]). In a mesh-network deployment, an external interference source ($I$) that is within range of the omni-directional transmitter ($Tx$) would be able to sense the medium to be free and backoff its transmission. However in WiLD links, the long distance between the two end points increases the propagation delay, and the antennas used lead to highly directional transmission. These factors in combination exacerbate the **hidden terminal** problem in WiLD networks. Hence in WiLD links, the transmitter and the interference source can erroneously sense the medium to be free leading to collisions whenever they are out of range of each other (because of the directional nature of transmission) or when they cannot sense the medium to be busy in time to backoff (because of the longer propagation delays).

Collisions at the receiver can manifest in two different situations: a) When $I$ doesn't hear $Tx$, and initiates a transmission when the medium is busy with an ongoing packet transmission from $Tx$, and b) When $Tx$ doesn't hear $I$, and causes a collision by interrupting an ongoing packet transmission from $I$.

To isolate the above two cases and measure the performance degradation due to each case, we perform controlled experiments using two WiFi links. We simultaneously send packets from both $Tx$ (512 Kbps CBR traffic) and $I$ (3Mbps traffic), and measure the packet loss rate on the primary link ($Tx \rightarrow Rx$) with MAC-layer ACKs disabled.

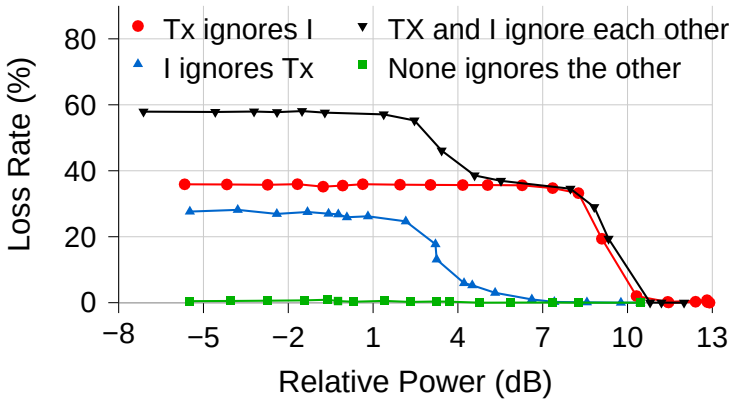


Figure 5.5: Losses due to different hidden terminal effects

To create the situation where $Tx$ cannot hear $I$, we disable the Clear Channel Assessment (CCA) at $Tx$, which simply causes $Tx$ to ignore $I$. We also eliminate propagation delay between $Tx$ and $I$ so that $I$'s CCA works perfectly. We reverse the operations to create the situation in which $I$ cannot hear $Tx$, but $Tx$ hears $I$ perfectly.

We then run four experiments, reflecting the losses in four situations: when $Tx$ can't hear $I$, when $I$ can't hear $Tx$, when neither can hear each other (representative of cases in WiLD networks), and when both $Tx$ and $I$ hear each other (representative of most cases in urban mesh networks).

Figure 5.5 shows the loss rate for each of the above four cases. In the case where $I$ ignores $Tx$, to overcome the interferer completely (achieve 0% loss), packet transmissions from the $Tx$ have to be 7dB stronger than the interfering transmissions. This threshold, at which the primary link is loss free, is much higher (12dB) in the case where $Tx$ ignores $I$. When neither of $Tx$ and $I$ can hear each other, both the above two types of collisions are possible. Hence the loss rate is a summation of the losses generated by the above two collision types. However, when both $Tx$ and $I$ are in range of each other, resembling a mesh-network, losses due to collisions are close to zero. In this case, CSMA ensures that the two transmitters, $Tx$ and $I$, share the medium.

From the above experiment we conclude that the effect of hidden terminals, causing collisions at the receiver, are greatly exacerbated in WiLD networks compared to urban mesh networks.

### 5.3.3 Effect of relative power and rate of external interference

To study the effect of relative power and rate of the external WiFi traffic on the loss of the primary link, we perform two experiments using the wireless channel emulator.

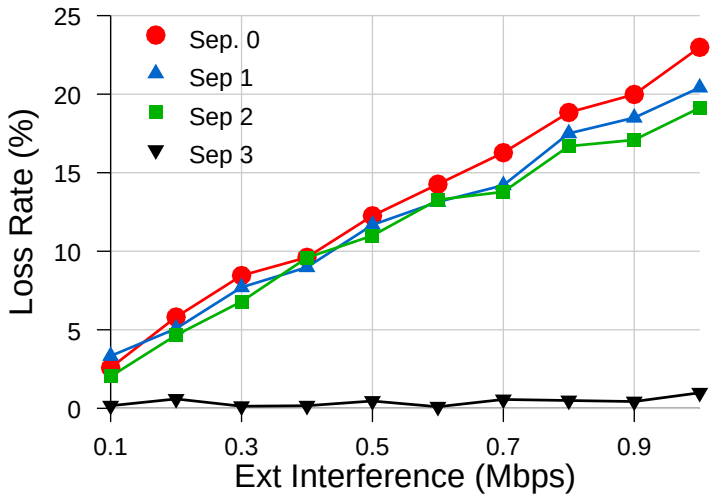


Figure 5.6: Loss rate at different channel separations: Varying interference rate

In the first experiment, we fix the relative power between the interference source and primary WiLD link, and vary the rate of the external interference source. The received signal strength of the interfering source was approximately 6dB higher than the primary link traffic. From Figure 5.6 we observe that for channel separations of 0, 1 and 2, the loss rate increases as the rate of the external interference increases. Also, the loss rate is almost the same for all the above channel separations. However, beyond a channel separation of 2, there is no significant interference from the external WiFi

traffic source and the loss rate is almost zero.

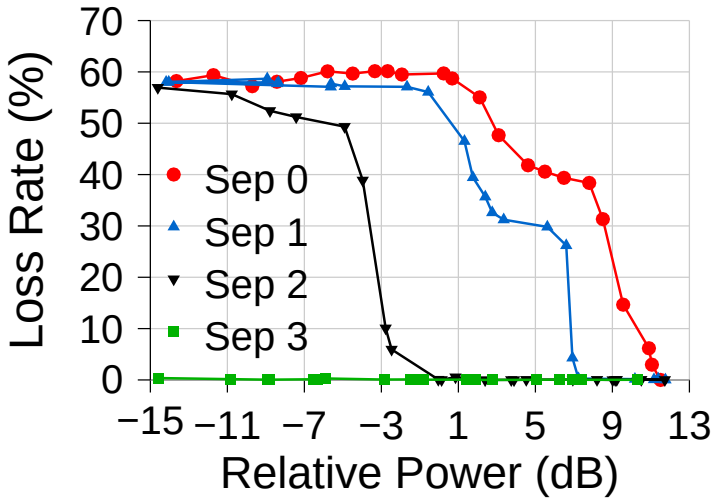


Figure 5.7: Loss rate at different channel separations: Varying interference power

Figure 5.7 shows the variation in loss rate for different relative power levels of the interference source and WiLD link. In this experiment, we maintain the signal strength of the primary WiLD link traffic at 42 dBm and vary the power of the interference source from 34 dBm to 54 dBm (left to right) as shown in the figure. The primary link CBR traffic rate is fixed at 512 Kbps, while the interferer transmits at a rate of 3 Mbps.

We observe that when the interference source is on the same channel, even an interference signal which is 12dB lower than the primary WiLD link could lead to packet loss on the primary WiLD link. When the interference source is significantly higher than the WiLD link(-6dB and beyond), the loss rate is very high ($\geq$50%) for channel separations 0, 1 and 2. This corresponds to the situation where any collision results in the capture of the packet on the primary link. However, within the [-6dB, 6dB] interval the channel separation does matter. At a channel separation of 2, the WiLD link is affected only when the interference source has a higher power. Beyond a channel separation of 2, we do not observe any loss on the primary link.

### 5.3.4 Implications

- We conclude that external WiFi interference is a significant source of packet loss in WiLD networks. Any deployment of WiLD networks in dense urban deployments has to take into account external WiFi interference.

- When calculating the link budget for urban links, it is beneficial to over-provision the received power. A high signal strength could potentially immunize the WiLD link from external WiFi traffic.

- MAC layer adaptation algorithms like adaptive channel switching, rate adaptation, and adaptive link recovery mechanisms (FEC and bulk acknowledgements) could significantly reduce the loss due to external WiFi interference.

## 5.4 Non-WiFi interference

The 802.11b communication protocol operates in the 2.4 GHz shared ISM band. This frequency band is shared with a host of other non-802.11 devices, such as microwave ovens, cordless phones, baby monitors, etc. Most of these non-802.11 devices do not follow a channel-access protocol which could lead to a significant amount of interference caused by these devices.

In Sheth et al. [61], the authors were able to detect and measure non-WiFi interference by sampling the noise floor of the Atheros chipset. The authors observed that in presence of external non-WiFi noise, the noise floor linearly increases with increasing noise. We performed the same experiment on our WiLD testbed, where we sample the noise floor for every packet received. In presence of external noise causing high loss, we would expect the noise floor to be correlated with the loss rate. However, based on extensive measurements carried out on the urban links we do not see any correlation between noise floor and loss rate. In fact, the noise floor remains mostly constant with minor 1–2 dB variations.

In addition to the above test, we also check for wide-band non-WiFi noise. A wide-band noise source would cause interference across the entire 802.11 spectrum. Ideally, this can be measured using a spectrum analyzer and detecting a rise in power across the entire spectrum. However, using a spectrum analyzer is infeasible on the outdoor WiLD links. Thus, to detect wide band noise in our WiLD deployment we synchronize the two ends of a link to rotate across channel 1, 6 and 11 periodically. The sender generates 1 Mbps UDP CBR traffic on each channel and the receiver measures the loss rate on each channel. In presence of any wide-band noise, we would expect to observe a correlation among loss rates across all three channels. However, based on long-term experiments performed on three urban links, we determined that there was no statistically significant correlation, and thus no significant broadband noise.

## 5.5 Multipath interference

Multipath interference is a well known source of packet loss in WiFi networks [10, 26]. It occurs when a RF signal takes different paths from a source to a destination node. Hence, along with the primary line-of-sight signal, the receiver also receives multiple secondary reflections that distort the primary signal. Although it is difficult to measure the exact delay between the primary and secondary paths on our WiLD deployments using commodity off the shelf equipment, based on the experiments using the wireless channel emulator we conclude the following:

- The order-of-magnitude lower delay spreads in WiLD deployments significantly reduce the interference due to multipath in WiLD deployments.
- If WiLD links are deployed in dense urban deployments with non-line-of-sight, multipath interference could lead to significant loss at the higher 802.11b data rates of 5.5 and 11 Mbps.

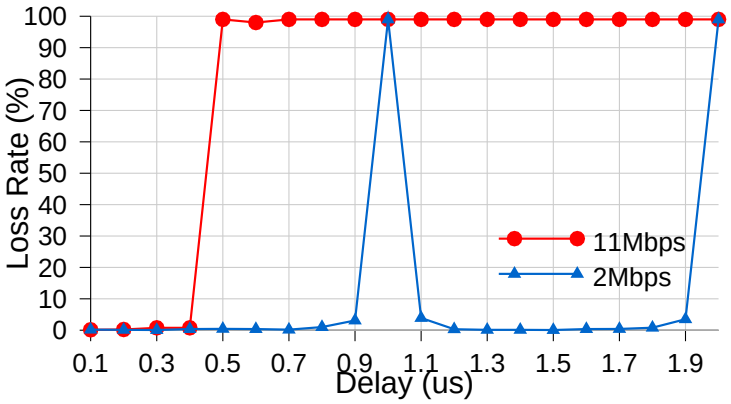


Figure 5.8: Effect of Inter Symbol Interference (ISI) due to multipath: ISI caused due to a single reflection arriving at the receiver

It is important to first quantify the effect due to ISI. If the effect of multipath interference is not significant, we could ignore multipath interference as being one of the significant sources of packet loss in WiLD links. To quantify the effect of ISI, we perform controlled experiments in a wireless channel emulator. Along with the primary line-of-sight path, we introduce a secondary reflected path and vary the relative power and delay between the two paths. Figure 5.8 shows the loss rate as a function of delay for 11 Mbps and 2 Mbps data rate encoding. From figure 5.8 we observe that at 11 Mbps the loss rate remains high for delays greater than 0.5 $\mu$ sec., whereas at 2 Mbps the loss rate only increases when the delay is an integral multiple of the symbol time (1$\mu$sec). Furthermore, the ISI is also a function of the relative power between the primary and the secondary paths. Figure 5.9 shows the loss rate for 11 Mbps with the relative attenuation set at 3dBm, 4.5dBm and 6dBm. As the attenuation of the secondary path is increased, the loss due to ISI reduces.

### 5.5.1 Multipath interference in Roofnet and WiLD deployments

Based on link-level measurements performed in the Roofnet deployment [10], the authors conclude that multipath interference was a significant source of packet loss in

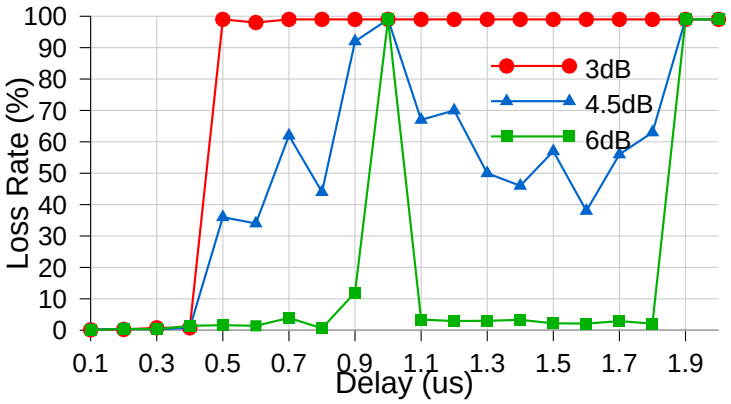


Figure 5.9: Effect of Inter Symbol Interference (ISI) due to multipath: ISI is also a function of power of the reflected ray

the Roofnet deployment. However unlike urban 802.11 mesh deployments, multipath interference is significantly lower in WiLD network deployments due to the order-of-magnitude lower delay spreads. The two factors contributing to lower delay spreads in WiLD networks are the long distance between the two end hosts and the line-of-sight deployment of the nodes. The strong line-of-sight component in WiLD deployments ensures that the attenuation of the primary signal is only due to path loss, and most of the secondary paths are due to reflections from the ground. In comparison to our WiLD deployment, the Roofnet deployment has shorter links and non-LOS deployments. The median link length is 0.5 km and the longest link is 2.5 km, and links are rarely line-of-sight.

Table 5.2 shows the delay between the primary path and secondary path assuming the antenna is mounted at a height of 30 meters and reflection is only from the ground. The two delays are computed for a secondary path reflecting at the quarter point and at the mid-way point between the transmitter and the receiver. Although multipath reflections arriving at the receiver are not constrained to these distances, the table provides the relative difference in delay spreads observed in the Roofnet deployment and WiLD deployment. As the length of the link increases, the primary and the secondary

| Dist. (km) | Delay spread ($\mu$sec) |
|---|---|
| 0.5 | (4.75, 3.59) |
| 1.0 | (2.4, 1.80 ) |
| 2.0 | (1.1, 0.90) |
| 4.0 | (0.6, 0.45) |
| 8.0 | (0.3, 0.22) |
| 16.0 | (0.15, 0.11) |
| 32.0 | (0.07, 0.06) |
| 100.0 | (0.02, 0.01) |

Table 5.2: Delays between a primary and secondary reflection

The delay reduces as the links get longer, and hence reducing the probability of inter-symbol interference.

path travel almost the same distance, and hence the delay between the primary and secondary reflection reduces. As seen from the table, there is an order-of-magnitude difference between the delay in WiLD links and medium range Roofnet links. Aguayo et al. [10] also observed that the RAKE receiver is able to tolerate delay spreads upto 0.3–0.4 $\mu$sec.

| Link | Distance (km) | 5.5 Mbps(%) | 11 Mbps(%) |
|---|---|---|---|
| C-T1 | 5 | 0.052 | 0.17 |
| A1-T2 | 10 | 0.26 | 1.7 |
| A2-V | 15 | 0.5 | 0.59 |

Table 5.3: Loss rates observed in WiLD links deployed in rural areas

To further validate that multipath interference is not a significant source of packet loss, we perform measurements over WiLD links deployed in rural environments. Our hypothesis was that most of the loss in our urban deployment was due to external WiFi interference. Hence, in absence of external interference the WiLd links deployed in the rural areas should not have any loss. Table 5.3 validates our hypothesis, which shows loss rates observed across three such rural links. From the table we observe that the maximum loss rate observed was 1.7% with low variance.

### 5.5.2 Effect of non-line-of-sight dense urban deployments

To study the effect of multipath when WiLD links are deployed in absence of line-of-sight, we perform controlled experiments in the wireless channel emulator. Due to the lack of analytical models, we build an artificial model consisting of 12 reflected paths with the path delay increasing in steps of 0.18 $\mu$sec and the power exponentially decaying. Hence the maximum delay between the primary and the longest secondary path is 2.16$\mu$sec. sec. Figure 5.10 shows the loss rate for payloads of size 768, 1024 and 1280 bytes and the four data rates of 802.11b. From the figure we observe that the lower data rates are resilient to multipath and the length of a frame does not affect the loss rate.

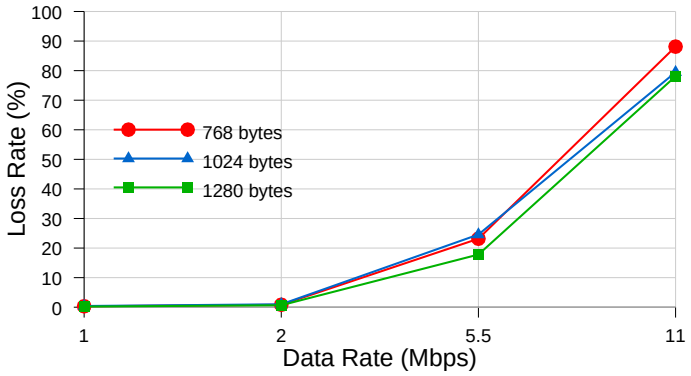


Figure 5.10: Multiple reflections at the receiver. Power is exponentially decaying and delay is increasing linearly in steps of 0.2 us

Looking closer at the packet traces collected at the receiver node, we observed that almost 100% of the errors were caused due to CRC errors. Figure 5.11 shows a histogram of the number of bytes corrupted as a percentage of total number of CRC error frames received. Since the loss rates were almost zero for 1 and 2 Mbps, we present the histogram only for the 5.5 Mbps and 11 Mbps data rates. From the figure we observe that the distribution has a heavy tail and almost 90% of the CRC error frames have less than 10% of the bytes corrupted in the payload. This has implications on the recovery

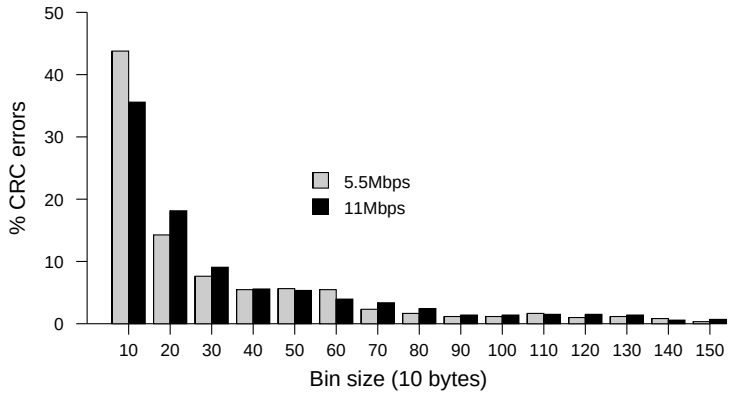


Figure 5.11: Distribution of number of bytes being corrupted

mechanisms in presence of multipath.

### 5.5.3 Implications

- The higher data rates of 11 Mbps and 5.5 Mbps are much more sensitive to multipath interference as compared to the lower data rates of 1 Mbps and 2 Mbps. This has implication on the rate selection algorithm for applications that can trade-off bandwidth for a loss free channel. In presence of significant multipath interference, a rate selection algorithm for such applications could directly move from 11 Mbps to 1 or 2 Mbps instead of stepping down to 5.5 Mbps.

- From 5.11 we observe that 90% of the corrupted frames have less than 10% of bytes corrupted. Also from Figure 5.10 we observe that increasing the length of the payload does not affect the loss rate. Hence, an alternate approach to rate adaptation could be to divide the payload into smaller blocks and encode these blocks to add in sufficient redundancy to tolerate the CRC errors due to multipath interference.

## 5.6 802.11 protocol induced loss

In this section we study the limitations of the standard 802.11 MAC protocol over point-to-point WiLD links. The 2P protocol [56] proposes modifications to the stock 802.11 MAC protocol to enable synchronous send and receive in point-to-multipoint WiLD links. However, in this section we argue that the 802.11 protocol suffers from fundamental limitations that make it unsuitable even for just point-to-point long distance links. The two main limitations of the protocol are the link-layer recovery mechanism and the breakdown of CSMA over long distances.

### 5.6.1 Link layer recovery mechanism

The 802.11 MAC uses a simple stop-and-wait protocol, with each packet independently acknowledged. The receivers are required to send an ACK within a tight time bound (ACKTimeout), or the sender has to retransmit. Because of this, with increasing link distance, the sender has to wait for a longer time for the ACKs to return (proportional to propagation delay). This decreases channel utilization. Also, if the time it takes for the ACK to return exceeds the ACKTimeout parameter, the sender will retransmit unnecessarily.

These problems can be understood by performing a simple experiment using the wireless channel emulator. We configure the emulator to introduce a propagation delay and vary it to emulate links ranging from 0-200 km. We set the ACK timeout value to the maximum possible (746$\mu$s) in Atheros chipsets, corresponding to a distance of 110 km (other cards like Prism 2.5 have lower limits). Figure 5.6.1 shows the performance of the 802.11 stop-and-wait link recovery mechanism over increasing link distance. With the MAC-layer ACKs turned off (No ACKs), we achieve a throughput of 7.6 Mbps for 1440-byte CBR traffic source using the 11 Mbps datarate. When MAC ACKs are enabled, the sender has to wait for an ACK after each transmission, and this leads to decreasing

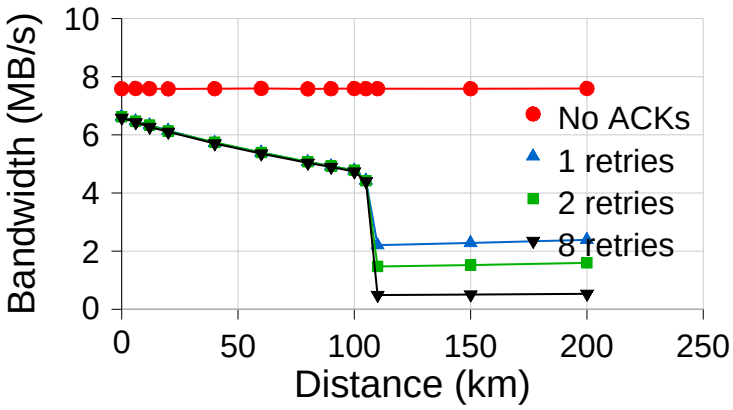


Figure 5.12: Under-utilization due to the 802.11 link recovery mechanism. Traffic is 1440 byte UDP CBR packets at 11Mbps PHY datarate of 802.11b

channel utilization with increasing link distance. After 110 km, the propagation delay exceeds the maximum ACK timeout and the sender always times out before the ACKs can arrive. We notice a sharp decrease in received bandwidth, as the sender retries to send the packet over and over again (even though the packets were most likely received), until the maximum number of retries is reached.

### 5.6.2 The Breakdown of CSMA

The 802.11 MAC protocol was originally designed for short-range indoor broadcast environments where both the transmitter and receiver nodes are in range of each other. The CSMA/CA channel-access mechanism of 802.11 requires each node to sense the medium before transmitting and initiate the transmission only if the channel is idle. However, on longer distance links, it is possible that the two nodes will begin transmission within the window defined by the propagation delay. Thus, in presence of bidirectional traffic there are frequent collisions between the two end points of a WiLD link and the throughput of the link severely degrades as the distance is increased.

### 5.6.3 Implications

- **TDMA based WiLD MAC protocol**: As discussed above, an un-synchronized channel-access mechanism like CSMA causes severe collisions even for plain long distance point-to-point links. This necessitates a MAC protocol that synchronizes the transmissions from both the end points of the link. Although Raman et al. [56] already motivate the need for a TDMA based MAC protocol for point-to-multipoint topologies, we observe that such a synchronized MAC protocol is required even for point-to-point WiLD links.

- **Adaptive link recovery**: An alternate approach that mitigates the under-utilization of the medium due to the large timeouts and propagation delay is to relax the constraint of having only a single un-acknowledged frame. We propose a sliding-window based flow-control approach, in which the receiver acknowledges a set of frames at once (bulk ACKs). This recovery mechanism allows uni-directional traffic flow within any TDMA time slot, and hence avoiding collisions of the data frames and acknowledgments. The MAC protocol for WiLDNet should combine TDMA based slot allocation with adaptive link loss recovery using bulk ACKs.

## 5.7 Summary of packet loss characterization in WiLD networks

In this chapter we presented a detailed study of channel induced (WiFi, non-Wifi, and multipath interference) and protocol induced (timeouts, breakdown of CSMA) losses in WiLD settings. Our main result is that most of the losses arise due to external WiFi interference on same and adjacent channels. This result is in contrast to loss studies of urban mesh networks, where multipath is reported to be the most significant source of loss. We also show that 802.11b protocol limitations make it unsuitable not just for point-to-multipoint links, as claimed in prior work, but also unsuitable for

simple long distance point-to-point links. Given these limitations, the next chapter presents the design and implementation of two very different adaptive link recovery mechanisms.

# Chapter 6

## Designing high performance WiFi based Long Distance Networks

In the previous chapter we identified two main sources of packet loss in WiLD networks; loss due to external WiFi interference and loss induced by the limitations of the stock 802.11 MAC protocol. In this chapter we outline the potential remedies to mitigate the sources of loss in WiLD networks. We motivate the need to re-design the underlying MAC protocol by systematically first evaluating remedies that only require adaptively changing the radio parameters. Specifically, we evaluate adaptive frequency selection and rate adaptation as the potential remedies. We address the limitations of these adaptation techniques, and motivate the need to re-design the MAC protocol to eliminate the loss due to external WiFi interference and the stock 802.11 MAC protocol.

### 6.1 System architecture for fault diagnosis in WiLD networks

In this section we first present an overview of the system architecture to diagnose the faults in WiLD networks. The previous chapter identified the stock 802.11 MAC protocol and external WiFi interference to the primary sources of packet loss in WiLD networks. The protocol induced losses are fixed by changing the channel access mechanism to TDMA instead of CSMA. However, external WiFi interference is not static and could vary significantly with time. This requires an online external WiFi interference detection system that continuously monitors the level of external WiFi interference and adapts the loss recovery mechanisms to mitigate the loss. The system architecture

to monitor the local external WiFi interference is similar to the overall architecture described in section 2.4.

The only difference between the indoor WLAN setup and the outdoor WiLD setup is that since the external WiFi interference is localized to an end-point in the network, sharing local observations of external WiFi activity is not required. The diagnosis data collected at each distributed end-point is locally used to trigger the remedies. In this chapter we explore software based remedies and the next chapter explores smart antenna based remedies. The following two sections describes the software based loss recovery mechanisms in detail.

## 6.2 Radio parameter adaptation

In this section we evaluate adaptive frequency and rate selection as the potential remedies to mitigate the sources of loss in WiLD networks. The primary limitation of the two techniques is that they only provide coarse-grain adaptations, which may not be suitable for QoS specific applications like video streaming.

### 6.2.1 Frequency channel adaptation

A simple solution to mitigate external WiFi interference could be to select an alternate less congested channel and switch to that channel. To motivate this simple technique we perform a channel switching experiment on our WiLD deployment on the K-P link. The source and destination switch between channel 1 and 11 synchronously every 30 seconds. Figure 6.1 shows the variability of loss rate across the two channels for a period of about 2 hours. We can observe that both channel 1 and 11 show bursts that stretch up-to a few minutes. It is important to note that by averaging the loss rate over 30 seconds we are not capturing the transient changes in the channel conditions.

Given the above loss trace across the two channels, table 6.1 compares different channel switching algorithms by the achieved loss rate and the no of channel switches

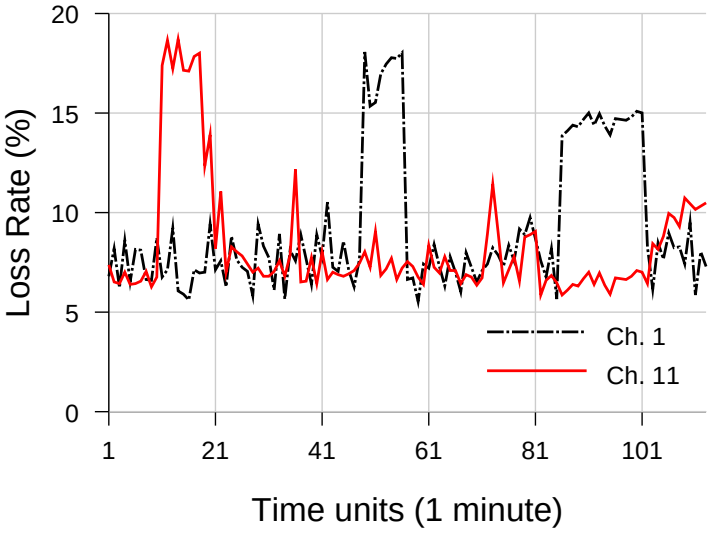


Figure 6.1: Loss variation over time across channel 1 and 11. Loss rate averaged every 1 minute.

required. In the base case (No adapt), where the channel is fixed at either channel 1 or 11, the average loss rate across the entire trace is either 9.2 or 8.3%. If the receiver has complete knowledge of the loss rate on both channels 1 and 11 at the beginning of a time interval (Oracle), then switching to the least lossy channel at any given time achieves the lowest loss rate (at 6.8%); but this comes at a cost of frequent switches of the channel. Adding a small hysteresis of 5% (Oracle 5%) for channel switching reduces the number of switches from 40 to 26 without increasing the average loss rate significantly. In absence of knowledge of loss rates on other channels, we can use the simple approach of jumping to the alternate channel when the loss rate on the current channel exceeds a threshold (e.g. 10% in Change $\geq$ 10%).

Although the reduction in loss rate shown in Table 6.1 by the different algorithms is only of the order of 1-2%, the advantages of channel switching could be significant in presence of long or high-loss bursts.

| | Loss | Switches |
|---|---|---|
| No adapt | (9.2, 8.3) | 0 |
| Lowest rate | 6.8 | 40 |
| Oracle (5%) | 7.006 | 26 |
| Change $\geq$ 10% | 7.76 | 8 |

Table 6.1: Table compares the frequency switching algorithms for the trace in Figure 6.1.

#### 6.2.1.1 Implications of channel switching

Even though adaptive channel switching seems to be a viable solution, large scale WiLD mesh deployments require careful channel assignment to avoid interference between multiple radios mounted on the same tower [56]. Switching the frequency channel on one link could lead to a cascading effect requiring other links to also change their operating channel. Hence, although it could mitigate interference, it is not always possible to switch a frequency channel in a large scale deployment.

### 6.2.2 Rate adaptation

Figure 6.2 shows the variation of loss rates as the relative power of the primary transmitter is increased with respect to that of the interference source for different 802.11 datarates.

We observed that in presence of external WiFi interference, data rate adaptation could either degrade the performance further or cause no effect on the loss rate. From figure 6.2 we see that when the received signal strength of the primary transmitter is higher than that of the interference source (from 0 to 12 dB), there is no difference in the loss rate for different 802.11b datarates. Whereas, when the interferer is stronger than the transmitter, reducing the data rate actually exacerbates the performance. This is because the increased transmission time of the frame increases the probability of a collision with the external traffic.

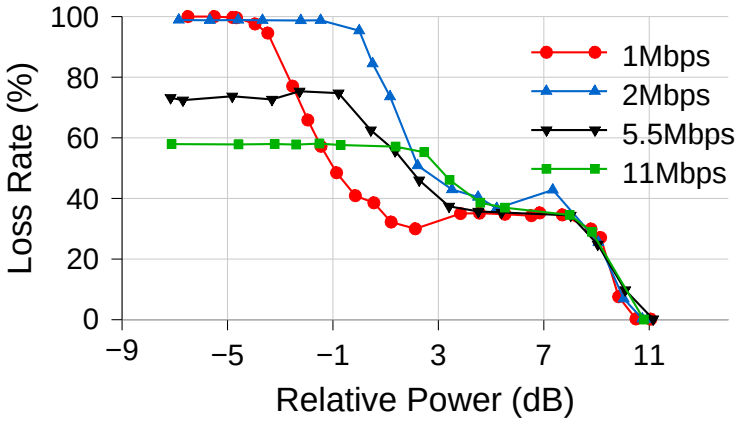


Figure 6.2: Loss rate for 802.11b encoding rates at varying relative power of transmitter compared to interferer. Traffic is 1440 byte UDP CBR packets at 11Mbps PHY datarate of 802.11b.

#### 6.2.2.1 Implications for adaptive rate selection

Most of the 802.11 radios have built in rate-adaptation algorithms which selects a lower rate with resilient encoding on experiencing high loss. However, the above analysis shows that in the presence of loss due to external WiFi interference, it is not worthwhile to adapt the data rate. Rather, we propose using other techniques such as adaptive FEC and link-layer retransmissions to mitigate the loss.

## 6.3 Design of WiLDNet

In this section, we describe the design of WiLDNet and elaborate on how it addresses the 802.11 protocol shortcomings and external WiFi interference to achieve good performance. To address shortcomings of the stock 802.11 MAC protocol, we propose the use of bulk packet acknowledgments. To address the limitation of CSMA over long distances, we build upon the TDMA protocol design of 2P [56]. Additionally, to handle the challenge of high and variable packet losses due to external interference, we design adaptive loss recovery mechanisms that use a combination of FEC and retransmissions

with bulk acknowledgments.

The three main design principles of WiLDNet are:

- The system should not be narrowly focused to a single set of application types. It should be configurable to provide a broad trade-off spectrum across different end-to-end properties like delay, bandwidth, loss, reliability and jitter.
- All mechanisms proposed should be implementable on commodity off the shelf 802.11 cards.
- The design should be lightweight, such that it can be implemented on the resource constrained single board computers (266 MHz CPU and 128 MB memory) used in our testbed.

### 6.3.1 Bulk Acknowledgments

We begin with the simple case of a single WiLD link, with each node having a half duplex radio. In this case, CSMA/CA over long distances is not capable of assessing the state of the channel at the receiver. Given this simple case, at the minimum we require an echo protocol between the sender and the receiver that determines when each node transmits to prevent unsynchronized packet collisions (i.e. both sender and receiver simultaneously transmit). In fact, the echo protocol is the simplest form of a TDMA protocol which is essential in WiLD environments [55]. Hence, from a node's perspective, we divide time into send and receive time slots.

To improve link utilization, we replace the stock 802.11 stop-and-wait protocol with a sliding-window based flow-control approach in which we transmit a **bulk acknowledgment** from the receiver for a window of packets. We generate a bulk acknowledgment as an aggregated acknowledgment for all the packets received within the previous slot. In this way, a sender can rapidly transmit a burst of packets rather than transmit each frame and wait for an acknowledgment for each.

The bulk acknowledgment can be either piggybacked on data packets (sent in the reverse direction), or sent as stand-alone packets if no data packets are available. By piggybacking the acknowledgments, the additional bandwidth usage is minimal . Each bulk ACK contains the sequence number of the last packet received in order and a variable-length bit vector ACK for all packets following the in-order sequence. Here, the sequence number of a packet is locally defined between a pair of end-points of a WiLD link.

Like 802.11, the bulk acknowledgment mechanism is not designed to guarantee perfect reliability. 802.11 has a maximum number of retries for every packet. Similarly, upon receiving a bulk acknowledgment, the sender can choose to advance the sliding window skipping unacknowledged packets depending on the maximum number of retries set. In practice, we can support different retry limits for packets of different flows. The bulk ACK mechanism introduces packet reordering at the link layer, which may not be acceptable for TCP traffic. To handle this, we provide in-order packet delivery at the link layer either for the entire traffic or at a per-flow level.

### 6.3.2 Designing TDMA in lossy environments

To address the inappropriateness of CSMA for WiLD networks, Raman et al. propose 2P [56], a contention-free TDMA based channel access mechanism. 2P eliminates inter-link interference by synchronizing all the packet transmissions at a given node (along all links which operate on the same channel or adjacent overlapping channels). In 2P, a node in transmission mode simultaneously transmits on all its links for a globally known specific period, and then explicitly notifies the end of its transmission period to each of its neighbors using marker packets. A receiving node waits for the marker packets from all its neighbors before switching over to transmission mode. In the event of a loss of marker packets, a receiving node uses a timeout mechanism to switch into the transmission mode.

The design of 2P, while functional, is not well suited for lossy environments. Given that many of the links in our network experience sustained loss-rates over 5–40%, in WiLDNet, we use an implicit synchronization approach. In WiLDNet, we use a simple loose time synchronization mechanism similar to the basic linear time synchronization protocol NTP [47], where during each time slot along each link, the sender acts as the master and the receiver as the slave. Consider a link $(A, B)$ where $A$ is the sender and $B$ is the receiver at a given time. Let $t_{send\_A}$ and $t_{recv\_B}$ denote the start times of the slot as maintained by $A$ and $B$. mutually agreed upon. All the packets sent by A are timestamped with the time difference ($\delta$) between the moment the packet has been sent ($t_1$) and the beginning of the send slot( $t_{send\_A}$). When a packet is received by B at time $t_2$, the beginning of B's receiving slot is adjusted accordingly: $t_{recv\_B} = t_2 - \delta$. In practice, due to software induced variations in propagation time for different packets, the value of $\delta$ as marked in each packet may not be reflective of its true value. To handle this, we use a simple smoothing function $t_{recv\_B} := \alpha * t_{recv\_B} + (1 - \alpha) * (t_2 - \delta)$. As soon as B's receive slot is over, and $t_{send\_B} = t_{recv\_B} + T$ is reached, B starts sending for a period $T$. Hence, an implicit synchronization approach significantly reduces the value of $T_0 - T$ thereby reducing the overall number of idle periods in the network.

### 6.3.3 Adaptive loss recovery

Handling high and variable loss-rates primarily induced by external WiFi interference is a challenging problem. However, to achieve predictable end-to-end performance, it is essential to have an loss recovery mechanism that can hide the loss variability in the underlying channel and provide a bound on the loss-rate perceived by higher level applications along a single link. More specifically, the loss recovery mechanism should provide a loss-rate bound $q$ independent of the underlying link loss-rate.

Achieving such a bound is not easy in our setting due to two factors. First, it is hard to predict the arrival and duration of bursts; also, bursts occur very frequently

in some of our links. Second, the loss distribution that we observed on our links is non-stationary even on long time scales (hourly and daily basis). Hence, it is not easy to use a simple model to capture the channel loss characteristics. In WiLDNet, we can either use retransmissions or FEC to deal with losses (or a combination of both). A retransmission based approach can achieve the loss-bound $q$ with minimal throughput overhead but at the expense of increased delay. However, our FEC approach primarily reduces the perceived loss-rate but cannot achieve arbitrarily low loss-bounds mainly due to the unpredictability of the channel. To achieve arbitrarily low loss rates using only FEC incurs a substantial throughput overhead. FEC incurs additional throughput overhead but does not incur a delay penalty especially since it is used in combination with TDMA on a per-slot basis.

#### 6.3.3.1 Tuning the number of retransmissions

To achieve a loss bound $q$ independent of underlying channel loss rate $p(t)$, we need to tune the number of retransmissions. One can adjust the number of retransmissions $n(t)$ for a channel loss-rate $p(t)$ such that $(1 - p(t))^{n(t)} = q$. Given that our WiLD links support in-order link-layer delivery (or in-order delivery on a per-flow basis, a larger $n(t)$ also means a larger maximum delay, equal to $n(t) * T$ for a slot period $T$. One can set different values of $n(t)$ for different flows. We found that estimating $p(t)$ using an exponentially weighted average is sufficient in our links to achieve the target loss estimate $q$. A purely retransmission based recovery mechanism has minimal throughput overhead as only the lost packets are retransmitted but this comes at a cost of high delay due to the long round trip times over WiLD links.

#### 6.3.3.2 Adaptive FEC-based recovery

Designing a good FEC mechanism in highly variable lossy conditions requires accurate estimation of the underlying channel loss. Underestimating the loss renders

the throughput expended in transmitting FEC packets useless, and overestimating the loss rate leads to throughput wastage. In our environment the loss distribution is non-stationary over large time scales, making it difficult to determine an accurate loss estimator. We experimented with a variety of FEC mechanisms and found that to achieve a target loss-rate $q$ independent of the loss variation, the amount of FEC required is substantially high (often 20–40%) primarily because of the frequent occurrence of bursts.

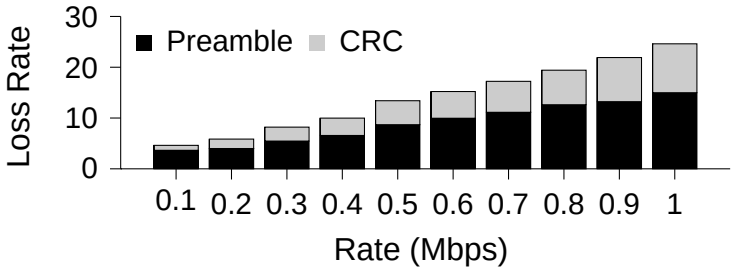


Figure 6.3: Breakdown of channel loss into CRC errors and preamble errors

**Motivating inter-packet FEC:** One can perform two types of FEC: inter-packet FEC (coding across packets) or intra-packet FEC (coding redundant blocks within a packet). In WiLD environments, we found that intra-packet FEC is not beneficial Based on extensive measurements on a wireless channel emulator we observe that in presence of external WiFi interference, the lost packets can be categorized into either CRC errors or preamble errors. A CRC error packet is received by the driver with a check sum error. However, an error in the preamble leads to the entire packet being dropped. This is because, the preamble has information which is used by the underlying firmware to lock onto the transmission of the receiver. Any error in the preamble would cause the firmware to drop the packet completely. Figure 6.3 shows the breakup of the loss rate with increasing external interference. The external interference is increased from 0.1 Mbps to 1 Mbps, and the loss rate is measured. We observe that as

the external WiFi interference increases, the lost packets due to preamble errors also increase. At 1 Mbps of external interference, almost 50-80% of the lost packets are due to preamble errors. Intra-packet FEC can only handle CRC errors but cannot handle the majority of packet losses caused by preamble errors. Hence, we chose to perform only inter-packet FEC.

**Estimating the level of redundancy:** We apply FEC in combination with TDMA. For every time slot of $N$ packets, we add $N - K$ redundant packets for every $K$ packets. We use simple erasure codes based on Vandermonde matrices to generate the redundancy packets. To estimate the redundancy factor, $r = (N - K)/K$, we choose a simple but not perfect estimation policy based on a weighted average of the losses observed in the previous $M$ time slots. Here, we specifically chose a small value of $M = 10$. There are several factors that motivate this simple policy choice. First, predicting the start of a burst is very hard; hence, we do not even attempt to predict it. Second, a small value of $M$, can quickly adapt to both a start of a burst as well as reduce the FEC when a burst subsides. For a time slot of $T = 10ms$, $M = 10$ corresponds to 400 ms to adapt to a change in the loss behavior. Third, due to non-stationary loss distributions, the added reduction that we observed in the perceived loss rate obtained by applying more complicated distribution based estimation approaches [64] is marginal. FEC is best suited for handling residual losses and long bursts. FEC is not suited to handle short bursts especially since the duration of most short bursts (mean duration $=0.3s$) is lower than the time that the weighted average FEC estimation mechanism takes to adapt.

## 6.4 Implementation of WiLDNet

In this section, we describe the implementation details of WiLDNet. The implementation comprises of two parts: (a) driver-level modifications to control or disable features implemented in hardware; (b) a **shim** layer that sits above the 802.11 MAC and

uses the Click [44] modular router software to implement the functionalities described in Section 6.3.3.2.

### 6.4.1 Driver Modifications

The wireless cards we use in the implementation are the high power (200-400 mW) Atheros-based chipsets. WiLDNet requires the following features disabled in the driver:

- Disabling Link-Layer Association: We disable link-layer associations in Atheros chipsets using the Adhoc-demo mode.
- Disabling Link Layer Retransmissions and Automatic ACKs: With the Atheros drivers, we did this by using 802.11 QoS frames with WMM extensions to set the no-ACK policy.
- Disable CSMA: We disable CSMA by turning off the Clear Channel Assessment (CCA) in Atheros chipsets using a proprietary HAL obtained from a vendor. With CCA turned off, the card can send packets without waiting for a clear channel.

### 6.4.2 Software Architecture Modifications

In order to implement single-link and inter-link synchronization using TDMA, the various loss recovery mechanisms, sliding window flow control, and packet reordering for in-order delivery, we use the Click Modular Router [44]. We use Click because it enables us to design a modular system with different functionalities implemented independently by various Click elements. Click simplifies prototyping new ideas and is reasonably efficient for packet processing in software, especially if loaded as a kernel module. Click enables us to intercept and modify link-layer packets exchanged between a wireless interface and the kernel. Using kernel taps, we create fake network interfaces,

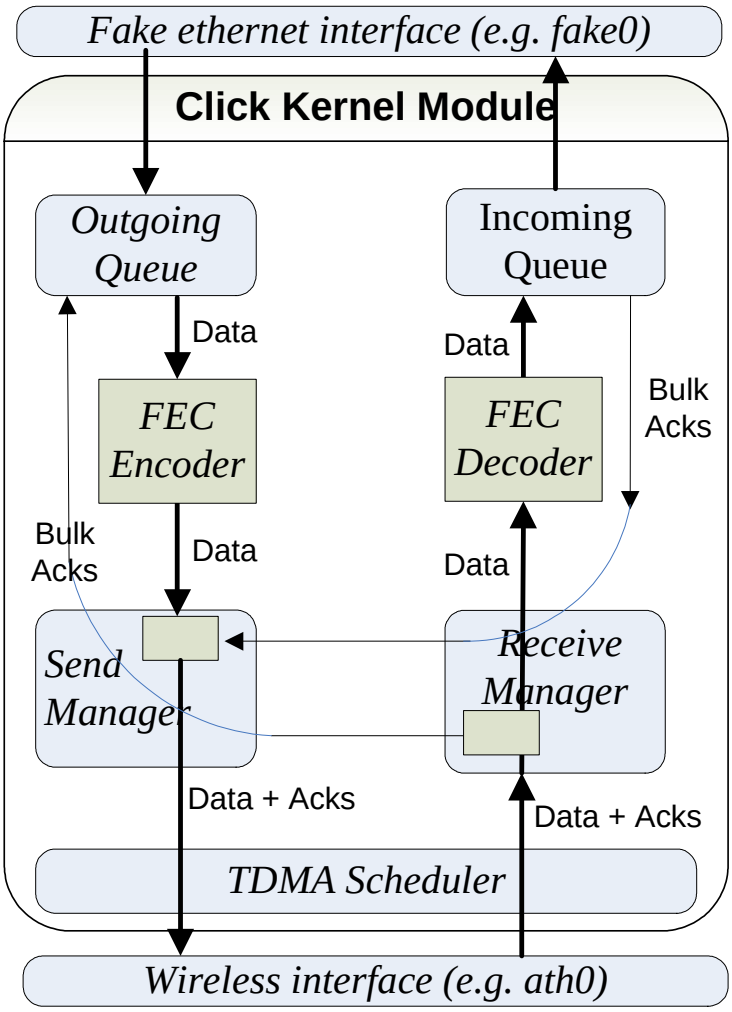


Figure 6.4: Click Module Data Flow

such as **fake0** in Figure 6.4; the kernel communicates with these virtual interfaces as if they would be real Ethernet network interfaces. Click takes the packets sent to a virtual network interface, processes them, and passes them to the corresponding real wireless interface.

Figure 6.4 presents the structure of the Click elements of our layered system, with different functionality (and corresponding packet header processing) at various layers:

- Incoming/Outgoing Queues: The mechanisms supporting sliding window packet flow, bulk acknowledgments, selective retransmission and reordering for in-order

delivery are implemented by the incoming/outgoing queue pair. Packet buffering at the sender is necessary for retransmissions, and buffering at the receiver ensures reordering. To ensure adaptability to various application requirements, in-order delivery and packet retransmission is optional, and the maximum number of retransmission can be set on a per-packet basis.

- FEC Encoder/Decoder: An optional layer is responsible for inter-packet forward error correction encoding and decoding. For our implementation we modify a FEC library [58] that uses erasure codes based on Vandermonde matrices computed over $GF(2^m)$. This FEC method uses a $(K, N)$ scheme, where the first $K$ packets are sent in their original form, and $N-K$ redundant packets are generated, for a total of $N$ packets sent. At the receiver, the reception of any $K$ out of the $N$ packets enables the recovery of the original packets. We choose this scheme because, in loss-less situations, it introduces very low latency: the original K packets can be immediately sent by the encoder (without undergoing encoding), and immediately delivered to the application by the decoder (without undergoing decoding).

- Send and Receive Managers: These elements are responsible for managing the bulk acknowledgments and the encoded data packets. Bulk ACKs are generated by the incoming queue, piggybacked to data packets or sent as stand-alone packets (if there is no data to send), and delivered to the outgoing queue of the peer host, which uses them to delete already delivered packets.

- TDMA Scheduler: This element ensures that packets are being sent only during the designated send slots, and manages packet timestamps as part of the synchronization mechanism.

- TDMA Controller: This element is common for all the interfaces supported by

the click module. It implements synchronization among the wireless cards on the same channel, by enforcing synchronous transmit and receive operation (all the cards on the same channel have a common send slot, followed by a common receive slot).

#### 6.4.2.1 Timing issues

Implementing time synchronization within Click has the disadvantage of being less precise. Since there is packet queuing in the interface itself, there is variability in the time between the moment Click emits a packet and the time the packet is actually sent on the air interface. Thus, the propagation delay between the sending and the receiving click modules on the two hosts is not constant, affecting time slot calculations. Fortunately, this propagation delay is predictable for the first packet in the send slot, when the hardware interface queues are empty. Thus, in our Click implementation, we only timestamp the first packet in a slot, and use it for adjusting the receive slot at the peer. If this packet is lost, the receiver's slot is not adjusted in the current slot, but since the drift is slow this does not have a significant impact.

Another timing complication is related to estimating whether we have time to send a new packet in the current send slot. Since the packets are queued in the interface, the time when the packet leaves Click cannot be used to estimate this. To overcome this aspect, we use the notion of **virtual time**. At the beginning of a send slot, the virtual time $t_v$ is same as current (system) time $t_c$. When we send the first packet, the virtual time becomes $t_v = t_c + duration(packet)$. In general, every time we send a packet, we recompute the virtual time: $t_v = max(t_c, t_v) + duration(packet)$. And every time a packet is sent we check that the virtual time after sending this packet will not exceed the end of the send slot. If the end exceeds the end of the slot, we postpone the packet until the next slot. Although our synchronization scheme works reasonably well, we intend to move this part of the system into the interface firmware for increased

accuracy.

## 6.5 Evaluation of WiLDNet

The main goals of WiLDNet are to increase link utilization and to eliminate the various sources of packet loss observed in a typical multi-hop WiLD deployment, while simultaneously providing flexibility to meet different end-to-end application requirements. Raman et al. [56] show the improvements gained by the 2P-MAC protocol in simulation and in an indoor environment but not in a real deployment in outdoor settings. We believe these are the first actual implementation results of a protocol similar to 2P over an outdoor multi-hop WiLD network deployment.

Our evaluation has two main parts:

- We analyze the ability of WiLDNet to maintain high performance (high link utilization) over long-distance WiLD links. At long distances, WiLDNet demonstrates 2–5x improvements in bidirectional TCP throughput over standard 802.11.
- We evaluate the effectiveness of the two adaptive link recovery mechanisms of WiLDNet: Bulk Acks and FEC.

### 6.5.1 Single Link Without Channel Losses

In this section we demonstrate the ability of WiLDNet to eliminate link under-utilization and packet collisions over a single WiLD link. We compare the performance of WiLDNet with the CSMA (2 retries) base case.

Figure 6.5 shows the performance of WiLDNet over a unidirectional link. The lower unidirectional throughput of WiLDNet, approximately 50% of channel capacity, is due to symmetric slot allocation between the two end points of the link. However, over longer links (>50 km), the TDMA-based channel allocation avoids the under-utilization of the link as experienced by CSMA. Also, beyond 110 km, CSMA begins to retransmit

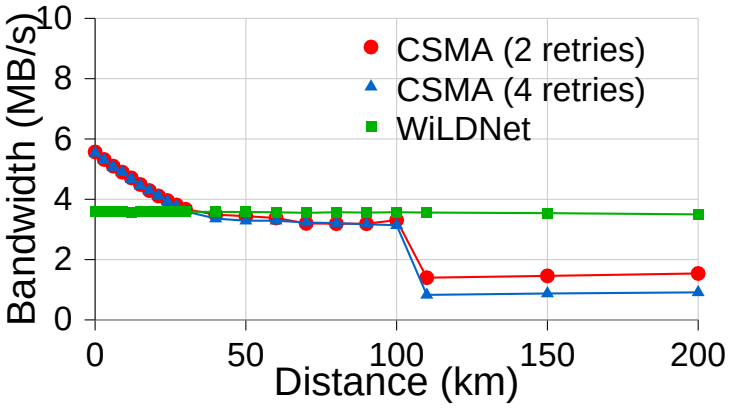


Figure 6.5: Unidirectional TCP performance

repeatedly after timing out waiting for Acks.

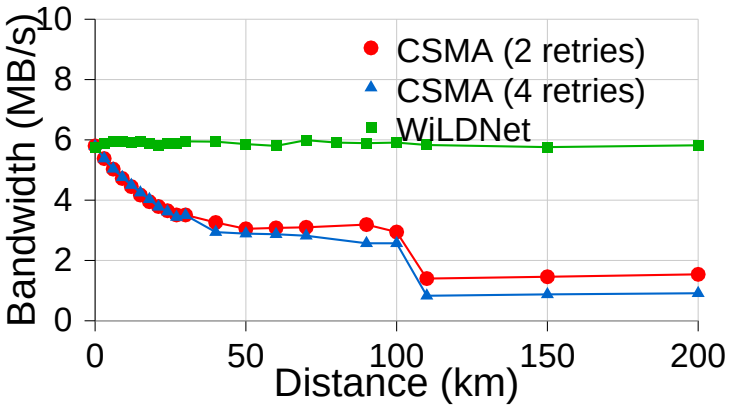


Figure 6.6: Bidirectional TCP performance

Figure 6.6 shows the performance of WiLDNet over a bidirectional TCP flow. In this case, WiLDNet effectively eliminates all collisions occurring in presence of bidirectional traffic. TCP throughput of 6 Mbps is maintained constant and close to the channel capacity in the bidirectional case, at increasing distances.

### 6.5.2 WiLDNet link-recovery mechanisms

Our next set of experiments evaluate WiLDNet's adaptive link recovery mechanisms in conditions closer to the real world, where errors are generated by a combination of collisions and external interference. We evaluate the bulk ACK mechanism as well as the FEC mechanism to recover from loss.

#### 6.5.2.1 Bulk ACK recovery mechanism

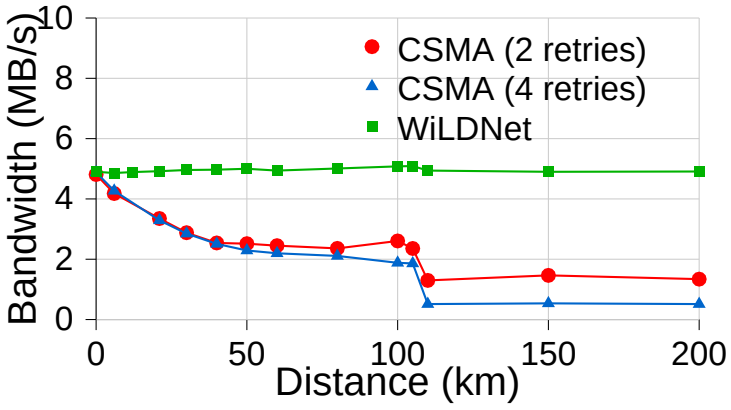


Figure 6.7: Bidirectional TCP with 10% channel loss rate

For our first experiment, presented in Figure 6.7, we uniformly vary the link length on the emulator, and we introduce a 10% error rate through external interference. We again measure TCP bidirectional throughput, and compare the same variations as the ones used in the previous section. Again, WiLDNet performs the best, with throughput unaffected by distance, since the sliding-window retransmissions are not sensitive to propagation delay, as opposed to the stop-and-wait used in 802.11 CSMA. Due to the 10% error, WiLD incurs a constant throughput penalty of approximately 1 Mbps compared to the no-loss case in Figure 6.6.

To compare the bulk ACK recovery mechanism with recovery at a higher layer,

we experimented with a version of the original Berkeley Snoop Protocol [16] that we modified to run on WiLD links. Basically, each WiLD router runs one half of Snoop, the fixed host to mobile host part, for each each outgoing link and integrates all the Snoops on different links into one module.

We compared the performance of standard 802.11 MAC, modified Snoop over standard 802.11 (CSMA), modified Snoop over WiLDNet with no retries, and WildNet with retries enabled. We fixed the distance to 80 km, and varied the channel induced loss rates uniformly from 0 to 50%. Our measurements show that WiLDNet maintains roughly a 2x improvement over all other alternatives, for packet loss rates up to 30%. We also see that the both the Snoop experiments are better than vanilla CSMA, but only at lower error rates (less than 10%). Thus, higher layer recovery mechanisms are better than stock 802.11 protocol, but only at lower error rates.

#### 6.5.2.2 Forward Error Correction (FEC)

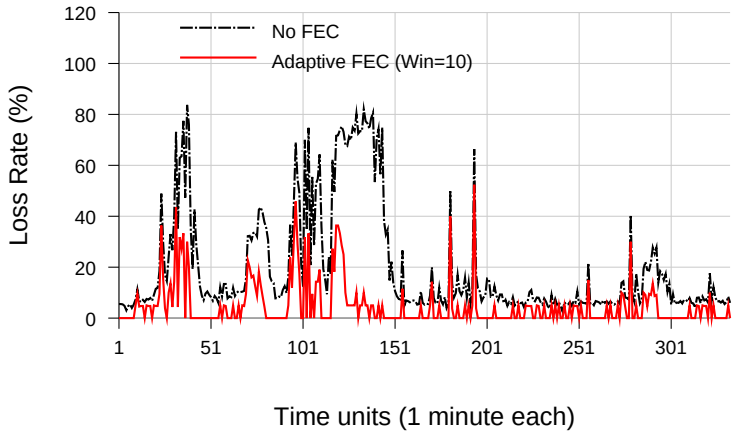


Figure 6.8: Comparison of loss rate observed with and without our adaptive FEC algorithm. Adaptive FEC can significantly reduce the loss rate during periods of long bursts.

Figure 6.8 shows the performance of WiLDNet's FEC algorithm on a real link (M-P from our campus deployment) with highly variable and bursty loss characteristics; the

link was extremely bursty with loss rates as high as 70–80% lasting for 20–30 minutes. Here, we estimate the FEC redundancy factor based on a weighted average of the previous $M = 10$ loss samples. This represents one of the worst case results of our FEC algorithm since this link is the most bursty link in our setup. In this case, the average loss rate across the entire 6 hour period is 19.98%; the FEC algorithm significantly reduced the loss rate to 4.78%. FEC is specifically useful in handling long bursts but is ill-suited to handle short bursts or sudden spikes in loss conditions.

One could alternatively consider a more proactive FEC approach that takes several prior loss samples and protects for the worst case. For this particular trace, such a mechanism reduced the loss-rate to a negligible value but incurred a substantial throughput overhead of 80%. One can consider alternatives to such an approach where we choose the 95th or the 99th percentile. All such approaches incur very high overhead but substantially reduce the perceived loss rate.

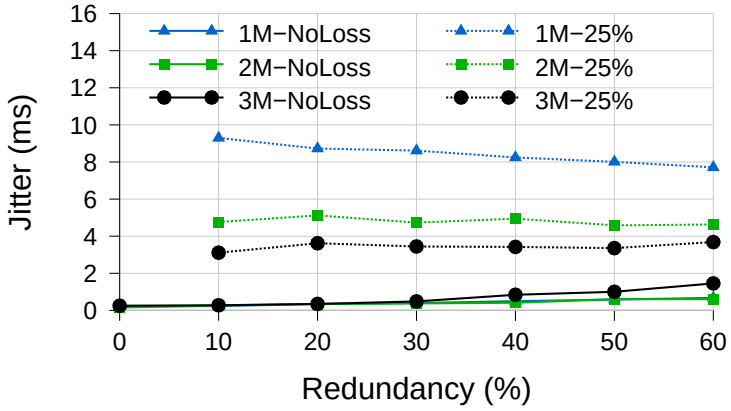


Figure 6.9: Overhead of the encoding and decoding process

Overall, the FEC mechanism we use incurs little jitter penalty. To measure this we performed a simple experiment where we measured the jitter of a flow under two conditions: in the absence of any loss (NoLoss) and in the presence of a 25% loss. Figure 6.9 shows overhead of WiLDNet's FEC implementation. We can see that in the

absence of any loss (NoLoss), when only encoding occurs, the jitter is minimal. However, in the presence of loss, when decoding also takes place, the measured jitter increases. However, the magnitude of the jitter is very small and well within the acceptable limits of many interactive applications (voice or video), and decreases with higher throughputs (since the decoder waits less for redundant packets to arrive).

Moreover, considering the combination of FEC with TDMA, the delay overheads introduced by these methods overlap, since the slots when the host is not actively sending can be used to perform encoding without incurring any additional delay penalties.

### 6.5.3 Tradeoff between bulk ACKs and FEC

One of the main design principles of WiLDNet is to build a system that can be configured to adapt to different application requirements. In this section we present the tradeoffs between delay due to bulk acknowledgments and bandwidth overhead due to adaptive FEC. We observe that WiLDNet can perform in a wide spectrum of the parameter space, and can easily be configured to meet specific application requirements.

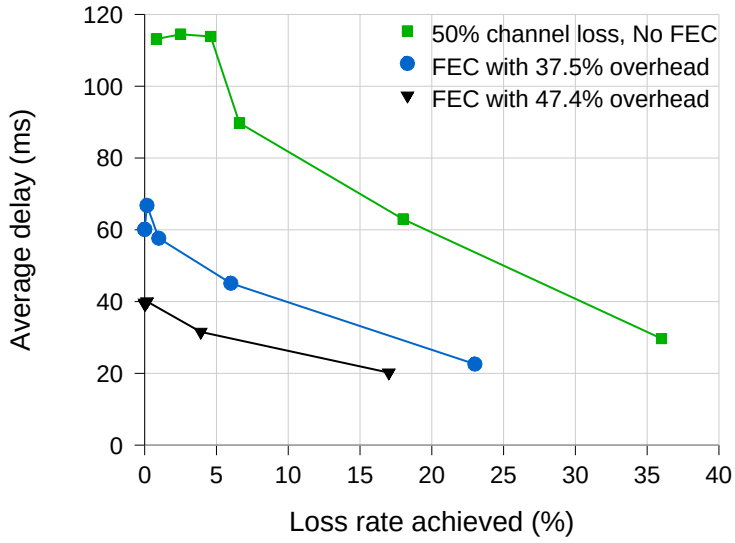


Figure 6.10: Tradeoff between delay and bandwidth

Figure 6.10 shows the tradeoff between delay and bandwidth achieved using a combination of bulk acknowledgments and FEC. For each line in the figure, the number of retries are increased from 0 to 10 in increments of 1 moving from right to left. All the tests are unidirectional UDP tests at 1 Mbps for a fixed slot size of 20ms. In absence of any FEC (50% channel loss, No FEC), the loss rate can be reduced by increasing the number of retries. For example, at a loss rate of 50%, 4 retries are required to reduce the error rate to 5%, which leads to a delay of approximately 110 ms. However by transmitting redundant packets along with the original packets, the delay to recover from the channel loss can be reduced. For example, a FEC redundancy of 37% can be used to reduce the delay to 50ms for a 5% error rate.

This tradeoff has important implications for applications that are more sensitive to delay and jitter (such as real time audio and video) as compared to applications which require high reliability (bulk transfers). For low bandwidth applications like VoIP, we can reduce the delay by increasing the FEC redundancy and reducing the number of retransmissions. For applications that require high throughout (bulk transfers), we can increase the number of retransmissions at the cost of a higher delay.

## 6.6 Summary of WiLDNet design

The commoditization of WiFi (802.11 MAC) hardware has made WiLD networks an extremely cost-effective option for providing network connectivity, especially in rural regions in developing countries. But an important stumbling block in realizing this possibility is the performance problems that these networks observe in real-world deployments. In this chapter, we mitigate the performance degradation caused due to stock 802.11 MAC protocol and due to external WiFi interference. Stock 802.11 protocol induced losses are mitigated by changing the CSMA channel access mechanism to a synchronized TDMA access mechanism. Also, the under-utilization due to long propagation time is overcome by acknowledging a batch of packets as compared to every

single packet. The chapter also discusses the limitations of the stock 802.11 protocol's recovery mechanism (rate and frequency adaptation) to recover from external WiFi interference. Loss induced due to external WiFi interference is mitigated by building adaptive link recovery mechanisms like bulk acknowledgments and FEC. Our design of WiLDNet demonstrates a 2-5x improvement in bidirectional TCP throughput over stock 802.11.

# Chapter 7

## Using smart antennas to overcome WiFi interference

In chapter 5 we identified the stock 802.11 MAC protocol and external WiFi interference to be the primary sources of packet loss in WiLD networks. Chapter 6 presented the software based link layer recovery mechanisms to mitigate these sources of performance degradation. In this chapter we address the limitations of the software based remedies to mitigate the performance degradation in WiLD networks. We explore the use of phased array antennas as a hardware based remedy to mitigate the loss. Similar to the local WiFi interference detection system for WiLD networks, we use the same input to change the antenna beam pattern to mitigate the effect of external WiFi interference. Specifically, we use the electronically steerable phased array antennas to dynamically steer their beam away from WiFi interference sources, thereby mitigating the performance degradation due to WiFi interference.

Although the software based link layer recovery mechanisms significantly reduce the perceived loss, wireless systems built using static directional antennas lack application agility and reliability. The link layer recovery mechanisms are application specific and require manual fine tuning. The static directional antennas are prone to misalignment, making them cumbersome to deploy and manage. A slight misalignment of the static directional antennas could leave the entire network partitioned. To overcome these limitations of the current architecture of static directional WiFi based networks, in this chapter we explore hardware based remedies to mitigate the loss in outdoor

WiLD networks.

Given the capabilities of an antenna that can dynamically steer its transmission beam, in this chapter we present measurement based evaluation of a metric to automatically select the best antenna direction to reduce the packet loss due to WiFi interference. We define the metric as Approximate Signal to Interference ratio (ASIR). ASIR maximizes the difference between the primary link's signal strength and the external interference sources. A high ASIR ensures that in the presence of concurrent transmissions at the receiver, the primary link's transmission captures the external WiFi interference.

An additional advantage of automatically selecting the best antenna state to avoid WiFi interference is that the network can dynamically self-adapt and maintain high performance and avoid network partitioning by adapting to misalignment of the antenna. Such a metric significantly reduces the channel loss rate without using link recovery mechanisms and improves the reliability of the system.

## 7.1 Motivation for phased array antennas

Although directional antennas help in improving the capacity of the network and improve link quality, our experience in building and maintaining large scale networks using static directional antennas has shown that static transmission beams of the antenna limits the recovery mechanisms and reduces the reliability of the network. In this section we outline some of the limitations of building and managing wireless systems using static directional antennas.

- **Cumbersome deployment:** Setting up a wireless system using static directional antennas is a cumbersome task. It requires careful manual alignment of the two antennas. Also, during deployment it is not possible to assess the best direction to point the antenna beam. Current practices use time averaged re-

ceived signal strength as an indicator of the best direction. However, a variation in received signal strength could leave the antenna pointing in a sub-optimal direction resulting in possible degraded performance.

- **High overhead of recovery mechanisms:**The link layer recovery mechanisms discussed in chapter 6 can only mitigate moderate levels of external WiFi interference. In the presence of high external WiFi interference, the recovery mechanisms incur a very high delay and bandwidth overhead. Thus, the static directional antenna have no capability to steer away from high external interference sources.

- **Application specific tuning of recovery mechanisms:** The recovery mechanisms to overcome the protocol and channel induced losses are application specific. The current design of the recovery mechanisms require manual fine tuning of the recovery parameters (number of retries, FEC redundancy, and slot size) to meet particular application requirements. These recovery mechanisms are difficult to configure when multiple network flows require different QoS requirements.

- **Misalignment of antenna:** Once the network is deployed, a slight misalignment of the static directional antenna could interrupt a link and potentially partition the entire network. For example, during periods of high wind in South India the entire network was partitioned due to a misalignment of a single antenna in the network.

- **Scalability limitations:** For the decentralized evolution of a wireless system, it is important that home owners/organizations can set up these antennas without understanding the details of forming a link and maintaining the topology of the network. However, the static beam of the antennas requires careful align-

ment and maintenance of a global topology. Once deployed, it is difficult to reform the topology of the entire network to accommodate new nodes or route around failed nodes in the network.

The above limitations mainly arise due to the static beam pattern of the directional antenna. Our choice of a smart antenna should thus maintain the advantages of a directional antenna and have the additional capability to dynamically steer the transmission beam. Dynamic beam steering allows an endpoint in the network to automatically adapt in the presence of WiFi interference and the changing topology of the network. In the next section we provide an overview of the phased array antenna system used.

## 7.2 Phased array antenna system

In this section we present an overview of the phased array antenna system. The phased array directional antenna system is manufactured by a local Boulder based company called Fidelity Comtech [4]. The antenna system is called Phocus Array antenna. The Phocus Array antenna system consists of eight element phased arrays driven by eight independent T/R (transmit-receive) modules. Various beam patterns are possible by setting the phase-amplitude weights across the eight T/R modules. The phase-amplitude settings used for each beam pattern are stored in flash. The antenna gain for transmit and receive are symmetric.

The software control on the antenna is achieved via serial-line commands from an embedded Single Board Computer (SBC) [7] running off-the-shelf Linux and modified Atheros chipset based Madwifi drivers [6] wireless drivers. The SBC is a low end 100/133 MHz AMD processor, has 64 MB RAM and a compact flash card for persistent storage. The wireless card used is a Atheros chipset based mini-PCI card with an external antenna interface. While a large number of beam patterns are possible, the Phocus

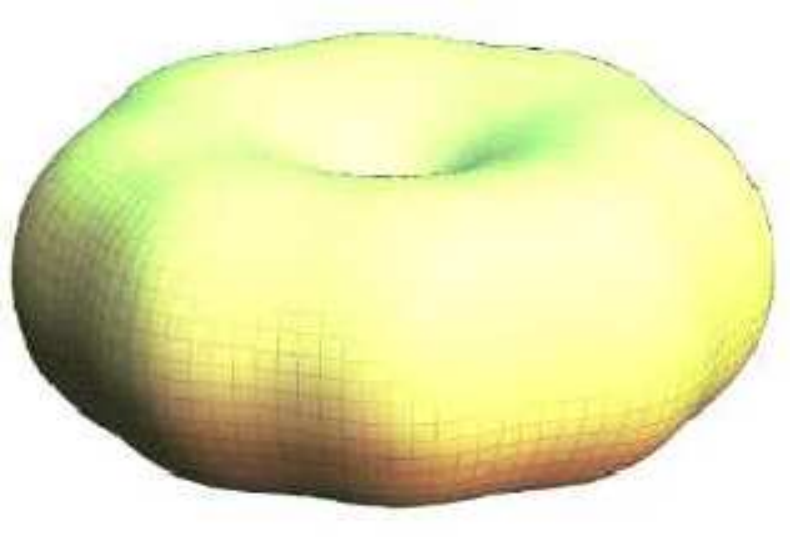

Figure 7.1: Omnidirectional beam pattern

Array is shipped with a default set of 17 patterns. These consist of one omnidirectional beam pattern and 16 directional beam patterns. Each directional beam pattern is 45° wide, and each beam pattern overlaps with the adjacent pattern and is rotated by 22.5°. This achieves a complete 360° circle with the 16 beam patterns. Figure 7.2 shows an illustration of two adjacent overlapping antenna beam patterns. The directional gain for each pattern is approximately 15dBi. As seen from the beam pattern illustration in Figure 7.2, along with the primary main lobe of the directional beam, an antenna pattern also has significant side and back lobes. The time required to switch between beam patterns shown in Figure 7.2 or between the omni-directional beam (Figure 7.1) and directional beam is approximately 100 $\mu$sec. The ratio of the lowest null to the highest peak is approximately 40dB, which allows for selectively "nulling" out external interference sources in the network.

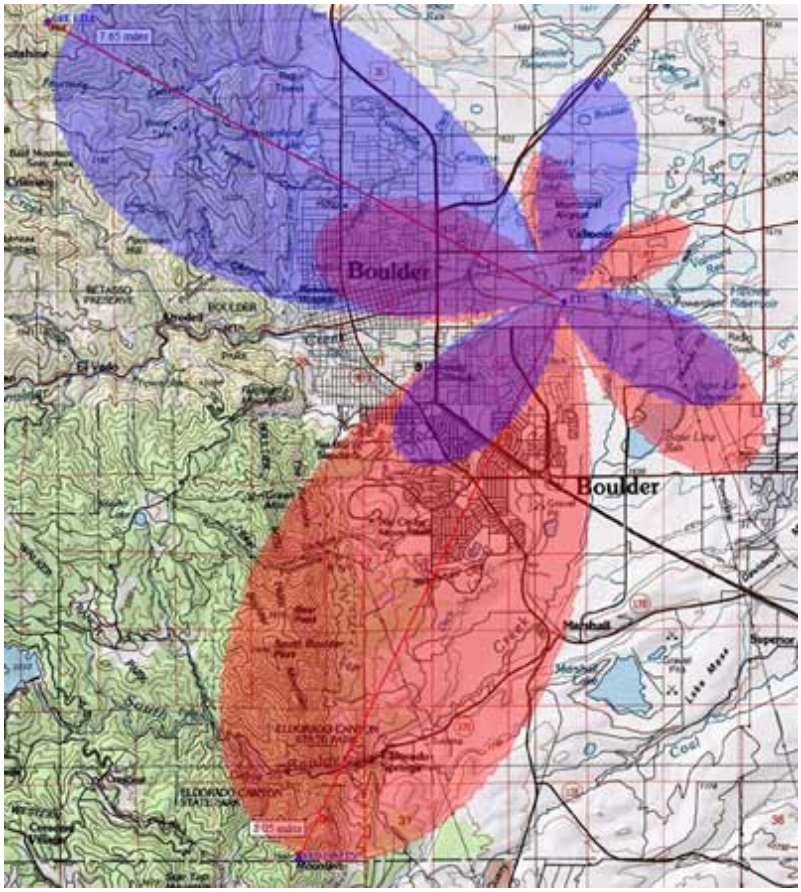


Figure 7.2: Two adjacent antenna beam patterns overlapping with each other

## 7.3 Design principles and challenges in metric evaluation

In this section we outline the challenges involved in designing a metric to automatically adapt the Phocus Antenna beam pattern to mitigate loss in the presence of external WiFi interference sources. The three main guiding principles of our metric design are as follows:

- The metric calculation should not require explicit coordination between the two ends points of a link. For example, measuring loss rate of a link would require the two end points to communicate with each other the number of packets transmitted and received.

- The metric calculation should be done online, and hence not incur a high overhead on the low end single board computers (100/133MHz with 64 MB RAM)
- The metric calculation should not require any additional hardware, and should only use information available from the open source wireless drivers.

Given the above three design constraints, there are a number of challenges involved in designing an efficient metric to measure the extent of WiFi interference and select the best antenna state. These challenges are listed below:

### 7.3.1 Large antenna state space

As described in section 7.2, the Phocus Antenna system consists of 17 distinct states (16 directional and one omni-directional). Hence, for a single link there are 289 distinct state spaces to explore and this state space grows exponentially as the number of nodes in the network increase. Clearly, an exhaustive search through this state space would be wasteful and would not allow online adaptation of the antenna beam orientation. Section 7.5 discusses the mechanisms in avoiding the exhaustive state space exploration based on a **hierarchical state space exploration** technique.

### 7.3.2 Measuring WiFi interference

Due to the easy plug-and-play capability of WiFi hardware, WiFi deployments are typically adhoc. In most residential settings, it is common to observe 20-30 AP's in range of interfering with each other on the same frequency channel. Clearly, in such a chaotic environment it is not easy to estimate the extent of WiFi interference apriori. Any control algorithm that requires an estimate of external WiFi interference needs to estimate the interference dynamically. Another challenge that limits the accurate measurement of WiFi interference is that it is not possible to capture all WiFi traffic. The signal is severely attenuated and may not be decoded at the receiver. Section 7.6

discusses how we approximate the calculation of WiFi interference by designing the ASIR metric.

### 7.3.3 Varying RF conditions

Measuring WiFi interference has the additional complication of varying RF environments. Aguayo et al. in their paper [10] measure the signal strength in an urban mesh network over short and time scales. Their study indicates that there is significant variation in the signal strength and link quality. Additionally, the number of WiFi sources in a network is not static. For example, in a hotspot setting, WiFi devices significantly increase during peak hours of the day and reduce towards the end of the day. These varying RF conditions need to be taken into account while designing the metric. Section 7.6 discusses how our technique is resilient in the face of varying RF conditions.

## 7.4 Experimental setup

In this section we present the overview of our experimental setup. Since WiFi interference is highly localized to the end points in the network, we adopt the same principle of distributed monitoring presented in Chapter 2. Each end point in the network is running a modified Atheros Madwifi driver. The driver is configured to have an additional virtual interface, in addition to the primary interface used for communication with the other end point. The virtual interface is set in monitor mode, and logs all the packets (including packets from external sources), summarizes these packets by extracting the relevant information, and passes this information to the OS kernel. In addition to the radio parameters, information about the antenna state is also captured for every packet received.

Figure 7.3 shows the network topology of the experiments. It consisted of two links, one short (1-3) and one long (1-2). The common endpoint was locate at the

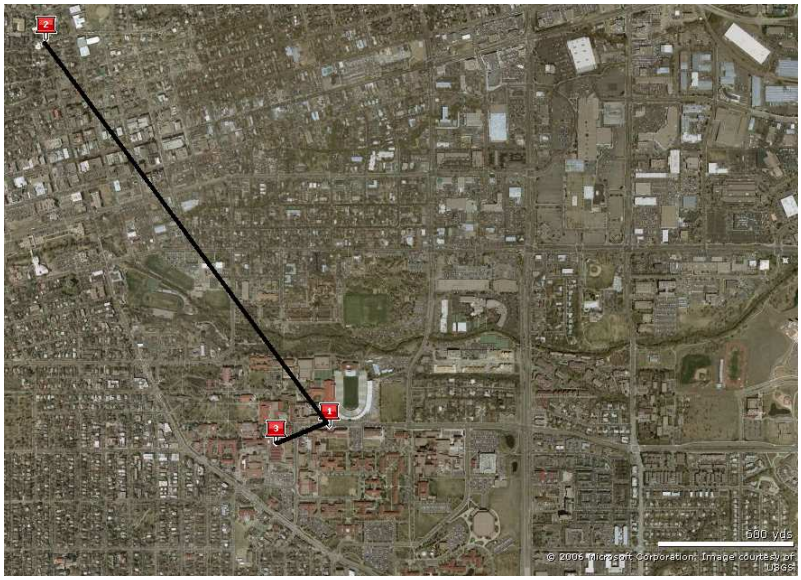


Figure 7.3: Topology of experimental setup

University of Colorado, Engineering tower, and the other two endpoints were located 0.1 miles and 1.2 miles apart. All the endpoint in the network were connected to a wired backhaul, and ssh tunnels were setup to remotely manage these endpoints. All the experiments were remotely configured and setup to run autonomously once started. All the nodes are static and were mounted on short antenna posts.

To run controlled experiments on the network, we use Iperf [5] as the packet generator. UDP CBR traffic was used as the traffic source and packet loss was measured across each test lasting 60 seconds. In addition to the Iperf logs, complete packet logs were also collected and periodically transmitted over the wired backhaul for post-processing. The packet log included all the packets captured over the channel. By post processing the packet traces, we separate out the primary link traffic (1-2 or 1-3) and traffic from external WiFi interference sources.

## 7.5 Avoiding exhaustive state space exploration

As discussed in Section 7.3, each Phocus Array antenna system consists of 17 distinct states, leading to an exponential state space explosion as the network grows. Exhaustively searching through each state is not feasible in such a network. To address this challenge we decompose the state space exploration in two distinct phases. The first phase requires a one time exhaustive search through the state space and identifies the cluster of states that have the link qualities below a threshold (currently set to 20%). This requires a NxN scan of all the states of every link adjacent to a node in the network. This reduces the state space into a cluster of states that have a loss rate below the preset threshold. The adaptation algorithm then dynamically configures the antenna state at the receiver based a greedy maximization of the ASIR metric.

Figure 7.4 shows the result after the first phase NxN exhaustive scan on the H-E link. The graph is a heat map of the measured channel loss rate for the different antenna orientations at the transmitter and receiver. The link layer ACKs were switched off, and hence the loss measured was the raw channel loss. The X-axis of the graph shows the antenna states at the transmitter and the Y-axis shows the antenna states at the receiver. State 0 corresponds to the omni-directional beam pattern. Thus, each box in the above graph corresponds to a combination of antenna states at the transmitter and receiver. The legend of the graph measures the loss rate (LR) for the antenna state combination. For all the experiments, the raw link loss rate below 20% is set as a threshold to select the set of states for the second round of the state space exploration.

From the figure we make the following observations:

- States 4-7 at the transmitter and states 10-13 at the receiver have the lowest loss rate. This cluster is formed when the main high gain lobes of the transmitter and receiver are directly pointing at each other. Also, due to wide beam widths of the main lobe (45°), there are a set of contiguous states that have a low

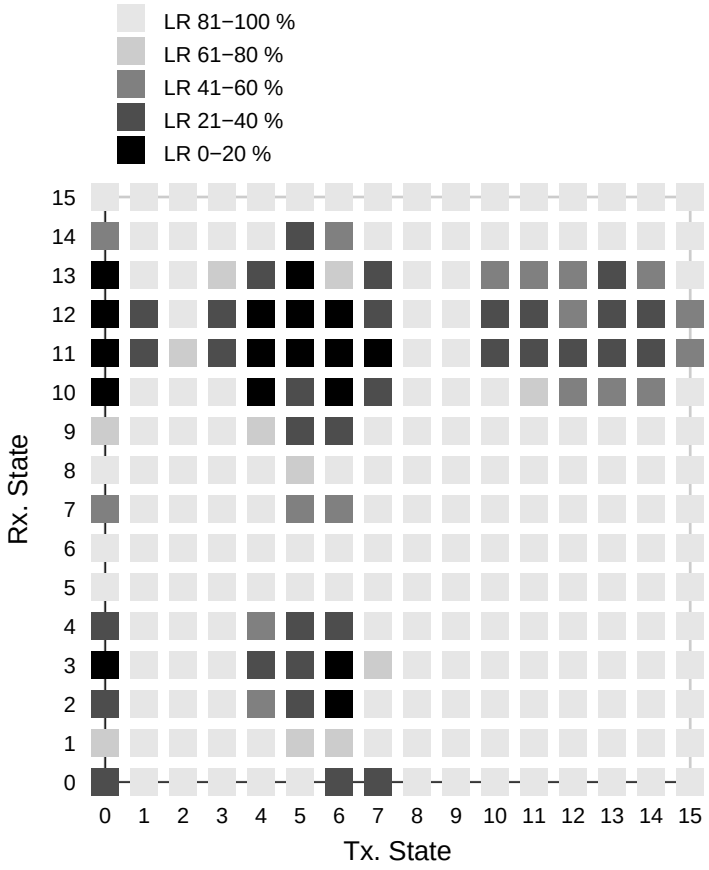


Figure 7.4: Heat map of NxN scan for the H-E link. The hot spots (LR 0-20%) shows the cluster when the main lobes are pointing at each other.

loss rate. This forms the main cluster which also results in the strongest signal strength at the receiver.

- In addition to the main cluster, there are also symmetric clusters to the right and below the main cluster. These clusters occur when the main lobe is oriented with a side lobe. In this case, since the signal strength is weaker as compared to the previous case, the link is lossy and also has a highly variably loss behavior due to external interference.

- When neither the transmitter's nor receiver's main lobes are oriented towards

each other, the heat map is cold. In most of the state space, the signal strength at the receiver is below the noise floor threshold of -95dBm and the receiver is not able to decode the frames.

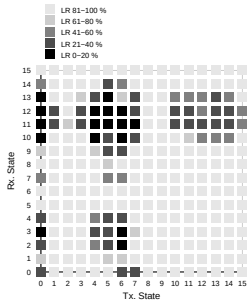


Figure 7.5: NxN scan - day 1

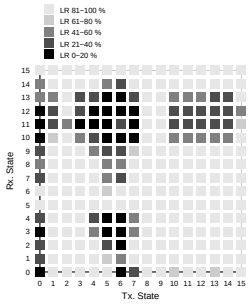


Figure 7.6: NxN scan - day 2

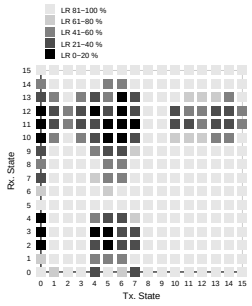


Figure 7.7: NxN scan - day 3

Figure 7.8: NxN scan for the H-E link done over 3 days. The main cluster of states when the antennas are pointing at each other does not change over time.

Figure 7.8 shows the results of the NxN scan done over three consecutive days. We observe that the main cluster of states when the two high gain antenna beams are pointing at each other does not change over time. Thus, the main cluster identified in the first phase of the state space exploration does not change over time. The subsequent phase only needs to adapt within this cluster to reduce the capture effect due to external WiFi interference.

## 7.6 Estimating the extent of WiFi interference

Section 7.5 described the first phase of the state exploration method in which the state space was reduced to a set of states that have a loss rate below a preset threshold. As seen from Figure 7.4, there is a cluster of 16 states having loss rate below 20% when the man lobes of the nodes are directly pointing at each other. Depending on the location of the external WiFi interference source, adapting the receiver antenna

beam pattern within this 16 state cluster could significantly reduce the effect of the interference source.

In this section we explore ASIR as a metric that can estimate the link loss in presence of external WiFi interference. In addition to ASIR, we also explore time averaged received signal strength (RSSI) as an estimation metric. Our initial experimental results show that ASIR is strongly correlated with loss rate induced by the external interference source.

To set up a controlled interference source, we use the shorter 1-3 link as the interfering link. Node 3, the interference source, is broadcasting to node 1. Node 1 is receiving from node 2. UDP unidirectional CBR sources are used at nodes 3 and 2. The interference source, node 3, is rotating through all the 17 states. The primary link nodes, nodes 1 and 1, are rotating through all the 16 states identified by the cluster in Figure 7.4. Thus, for every state in the cluster, node 3 iterates through all its 17 states and the link loss rate was measured. Along with the link loss rate, the complete packet log was also saved.

For the above described experiment, Figure 7.9 shows the best antenna state at the transmitter and receiver for all the states of the interference sources. The best states are determined by measuring the loss rate for each combination of antenna states at the transmitter, receiver and interference source. From the figure we observe that state 10 is the best state at the receiver for almost all orientations of the interference source state. At the transmitter, state 6 is the best state. Although, the transmitter does also toggle occasionally between adjacent states 5 and 7. Thus, for a given location of an interference source, there is clearly a single state (Tx: 5,6 and Rx: 10) in the cluster that provides the lowest channel loss rate. The rest of this section looks at designing a metric to automatically select this state from the cluster.

To understand better why state 10 at the receiver is consistently the best state for all the states in the cluster, we performed measurement analysis of the packet level

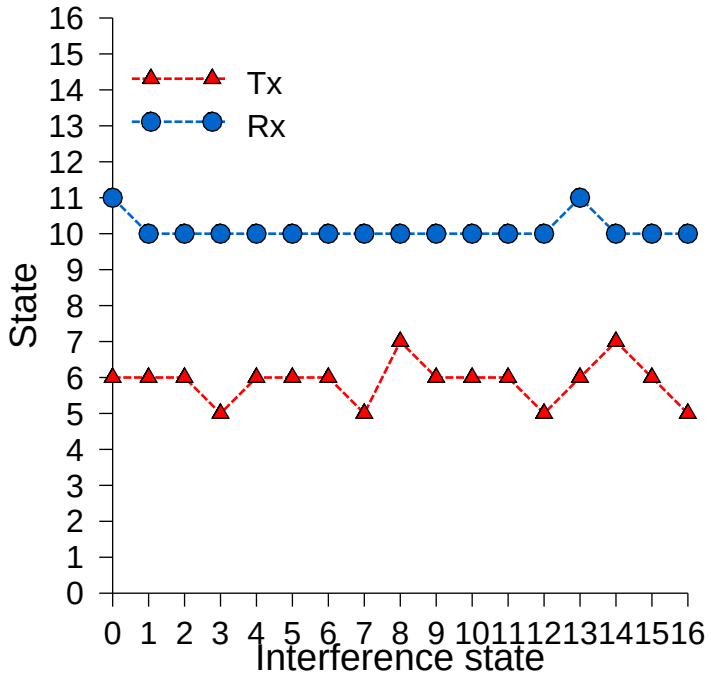


Figure 7.9: Figure shows the best antenna states of the transmitter and receiver (Y-axis) for every state of the interference source (X-axis)

traces collected from the experiment. Figure 7.10 shows the averaged signal strength for every state in the 4x4 cluster. The graph is divided into 4 vertical sections corresponding to the transmitter states 4-7 in the cluster. Within each vertical section, there are 4 independent RSSI groups, which correspond to the receiver states of 10-13. The scale of the Y-axis is the absolute measured RSSI reported by the Atheros based Madwifi driver. From this graph the key observation we make is that the RSSI of the primary link in state (Tx:5, Rx:10) and state (Tx:6, Rx:10) is not the maximum RSSI. In fact, states 11 and 12 at the receiver have a much higher RSSI as compared to state 10. This is counter-intuitive that a state with a lower RSSI is selected instead of a state with higher RSSI.

To explain this behavior, Figure 7.11 shows the RSSI at the receiver E from

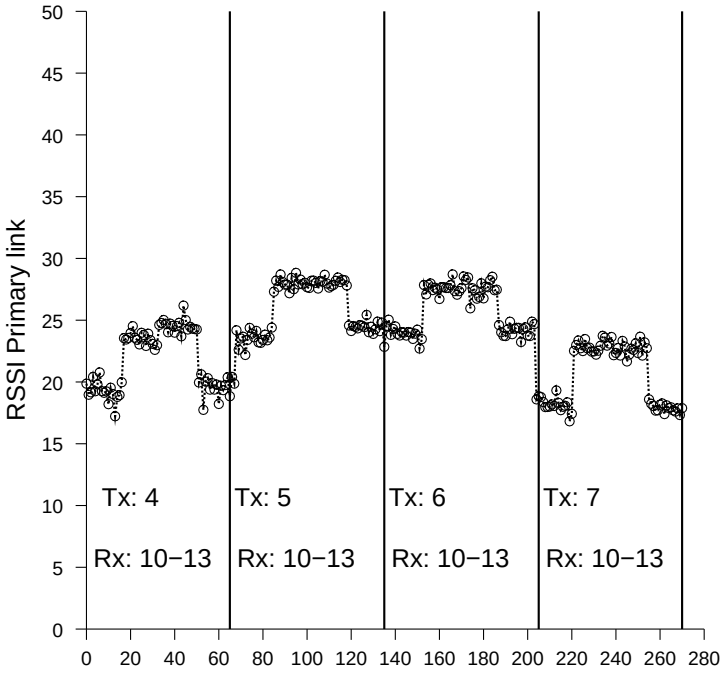


Figure 7.10: Average RSSI of the main link for every state in the 4x4 cluster. Each circle on the graph is for a separate antenna state for the interference source

the interference source M. From the above figure we observe that the RSSI from the interference source is the least in state 10 across all the states. In fact the signal strength from the interference source consistently increases from state 10-13.

From Figure 7.10 and 7.11 we observe that when the receiver is in state 10, the signal strength from the interference source is minimal. In effect, node 1 is steering a null in the direction of interference source (node 3) while maintaining a good link quality with the destination (node 2).

Figure 7.12 shows the lack of correlation of loss rate and signal strength of the primary link. The lack of correlation suggests that selecting the antenna state by simply measuring the primary link's signal strength is not sufficient.

Figure 7.13 shows the correlation of loss rate with the difference between the primary link and interfering link (ASIR). The ASIR metric takes into effect external

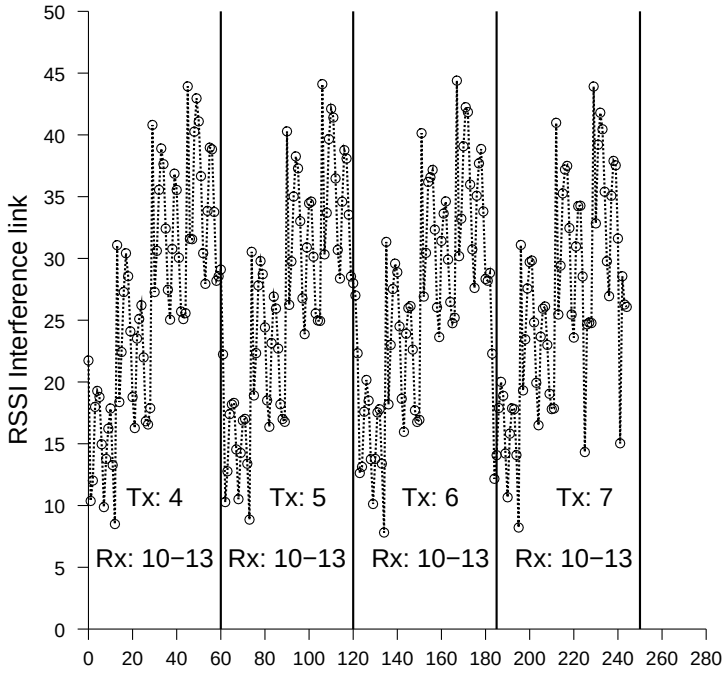


Figure 7.11: Average RSSI of the interference source at the receiver. Each circle on the graph is for a separate antenna state for the interference source

interference sources and shows a strong correlation with the loss rate. When the ASIR is positive ($> 0$), the loss rate is bounded at 40%. Also, the variability increases as the ASIR reduces. When the ASIR is negative ($< 0$ ), the loss rate is high and there is higher variability in the loss rate. Thus the control algorithm selects an antenna state at the receiver by maximizing the ASIR metric. From Figure 7.13 we conclude that metrics that take into account network wide information are more efficient as compared to metrics that limited to the specific link.

## 7.7 Discussion

In this section we briefly discuss some of the limitations of the ASIR metric to mitigate external WiFi interference in WiFi based long distance networks. Our measurements have shown that the ASIR is not an accurate metric of link quality in short

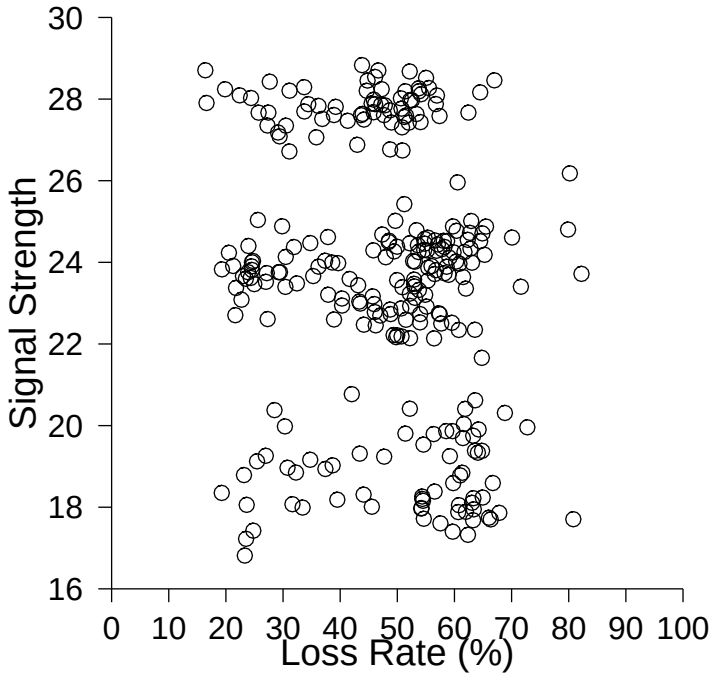


Figure 7.12: Correlation of loss rate and signal strength of the primary link

point-to-point links and in the presence of highly varying RF conditions due to multipath and changing environment. The following two sections discuss these limitations in detail.

### 7.7.1 Short point-to-point links

The primary advantage of using phased array antennas in long distance point-to-point links is that the signal strength sharply falls off outside the primary cluster. This sharp fall off of the signal strength reduces the state space to a small cluster and also maximizes the gain of the link to a few set of antenna states. However in presence of short and non-LOS links, the advantages of directionality are completely lost. The state space is uniformly "hot" and the receiver can always hear the transmitter irrespective of the antenna state. Figure 7.14 shows the heat map for the short (1-3) link. The receiver is node 1 and the transmitter is node 3. From the heat map we make two main

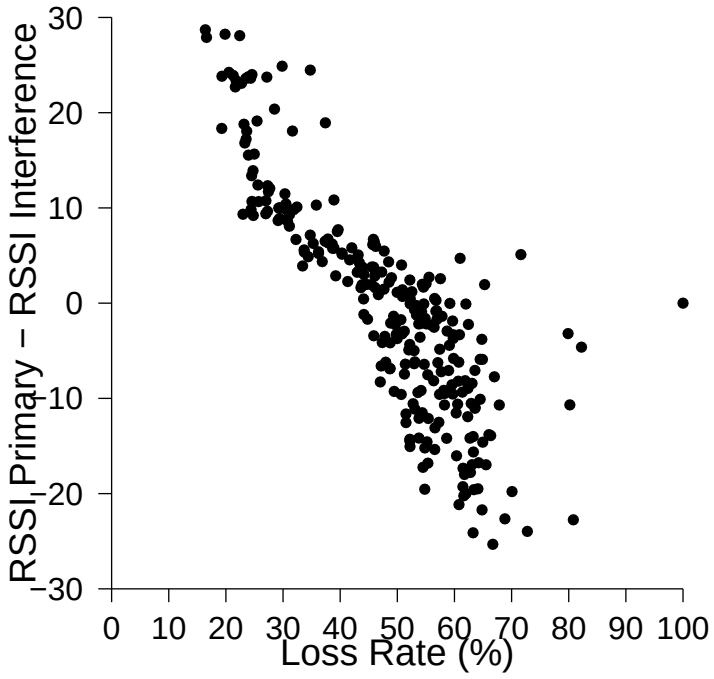


Figure 7.13: Correlation of loss rate and ASIR (RSSI Primary - RSSI Interference)

conclusions; First, due to the short distance between the transmitter and receiver, even the side lobes are in range of each other and there are no gains due to directionality. Second, since the link is non-LOS there is significant loss due to multipath effect. Given such a link, maximizing the ASIR is not beneficial as the directionality is completely lost due to multipath reflections.

### 7.7.2 Dynamic environments

In the presence of dynamic RF environments, it is difficult to assess the amount of external WiFi interference using a metric like ASIR. Since ASIR assumes a fairly static signal strength from the receiver, short time scale variations in signal strength cannot be accounted for. This may lead to the selection of a sub-optimal antenna state between the transmitter and receiver. This problem is exacerbated in short non-LOS

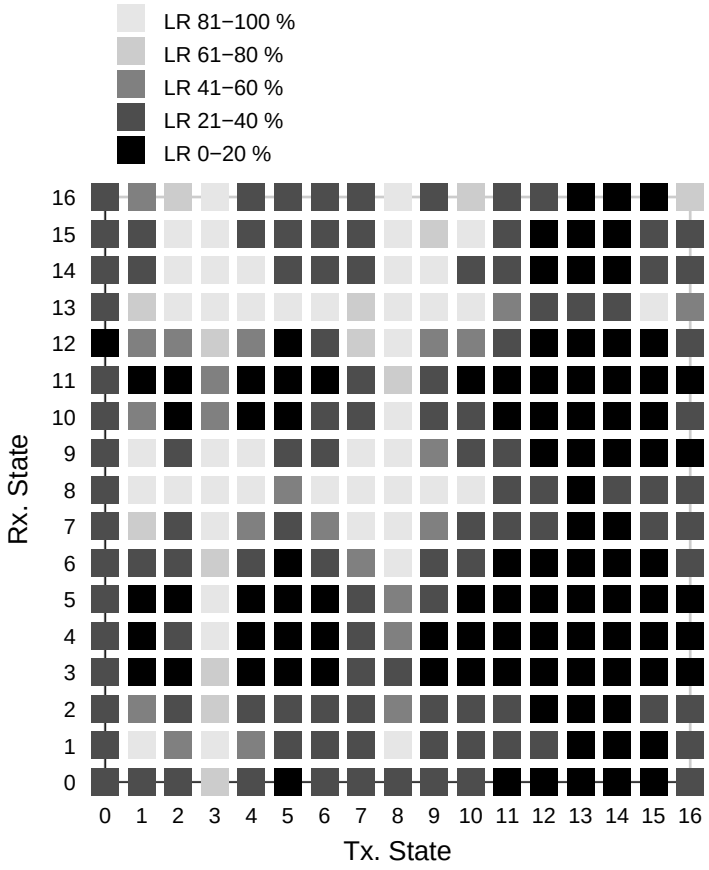


Figure 7.14: Heat map for a short non-LOS link. The heat map is uniformly hot due to multipath reflection.

links that exhibit high variations in signal strength.

## 7.8 Summary of smart antenna based WiFi interference mitigation

In this chapter we discuses the use of phased array antennas as an alternative mechanism to mitigate the performance degradation due to WiFi interference. The design choice of the phased array antennas was driven by its capability to dynamically steer the transmission beam. This basically allows a receiver to dynamically steer away from WiFi interference sources and reduce the capture caused by the external source. This chapter highlights the primary challenges in dealing with phased array antennas

and outlines a hierarchical state decomposition algorithm to deal with the large state space. Our measurement based study has shown that ASIR (RSSI Primary - RSSI Interference) is a good metric for assessing the extent of interference from the WiFi source, and can be used to dynamically steer the antenna beams at the transmitter and receiver nodes. An important observation we make is that the ASIR metric takes into account network wide information as compared to metrics that are limited to the specific link. However, the ASIR metric is not robust in presence of multipath reflections when the links are short and do not have line-of-sight. Further work is required to understand the behavior of such short and non-LOS links.

# Chapter 8

# Related work

In this chapter we highlight some of the related work and altenate techniques for improving performance of wireless systems by root cause fault diagnosis. A large body of work already exists in fault diagnosis and troubleshooting performance degradation faults for the wired networks. Faults on the wired network include IP link failures, Border Gateway Protocol (BGP) misconfiguration [46] and network intrusions and DoS attacks [35]. A wide array of tools [42] and architectures [25] have been proposed that help researchers to extract information from the network to detect these faults.

However, fault diagnosis for wireless networks pose additional new challenges. These systems have to deal with the inherent spatial and temporal nature of wireless propagation. Also, as discussed in this thesis, multiple distributed observations are required to be able to accurately diagnose the root source of the performance degradation fault. The rest of this chapter presents the related work in the following three sections.

## 8.1 Mitigating performance degradation in indoor WiFi networks

The monitoring infrastructure for wireless network is very different as compared to wired networks. For wired networks, a single monitor can capture all the traffic on the network. However in wireless networks, due to the spatio-temporal nature of the RF medium, a single sniffer is not sufficient to capture the state of the entire wireless network. Yeo et al. [69, 70] were the first to explore the feasibility of using multiple

sniffers to deal with the spatial and temporal variability of the wireless link. However, in addition to ensuring complete coverage of the wireless network, our system also requires redundant distributed observations at the PHY layer.

A large number of measurement based studies have been carried out to study the usage pattern of 802.11 based wireless networks [39, 32, 59, 40]. The authors in [59, 40] study the performance of 802.11 in a conference setting, where a large number of clients are using the wireless network. The authors observed both short term as well as long term variability in link quality and performance degradation under heavy usage of the wireless network. The authors also point out that the default 802.11 based remediation of rate fallback exacerbates the problem further, leading to a higher number of retransmissions and dropped frames.

Existing solutions to diagnose faults in wireless networks have limited capability to distinguish between multiple root causes of a fault. [53] proposes an online trace driven simulation tool to diagnose faults in a multi-hop adhoc network. The tool models the network and detects anomalies from normal behavior. However the tool categorizes faults into very broad categories. One of the categories is "random packet dropping", which could arise due to a large number of root causes.

There are a large number of commercial tools [1, 11] available that monitor 802.11 traffic in the network using passive probes. Based on policies defined by the network administrator, a variety of security and performance alerts are generated. Performance alerts are generated for excessive retries, low data rate, frequent handoff of client devices, change of AP parameters, etc. These tools only monitor the 802.11 MAC protocol and do not detect the root cause of the fault originating at the physical layer.

Client side monitoring to diagnose root cause faults has potential to diagnose anomalies for the wired network [51] as well as for wireless networks [9]. In [9], the authors propose an architecture for client side monitoring to detect unauthorized APs, RF holes and performance problems. However, the performance problems are only limited

to detecting whether the fault exists on the wireless network or the wired network.

Problems like hidden terminals [68, 19], capture effect [43], and carrier sensing in the presence of noise/interference in the network [39] have been studied by the research community in isolation. As far as we know, we are the first to present a unified framework which measures the impact of each fault at different layers of the network stack and presents detection algorithms for each of the above faults.

## 8.2 Mitigating performance degradation in outdoor WiLD networks

The use of 802.11 for long distance networking, characterized by directional links and multiple radios per node, raises a new set of technical issues that were first illustrated in [18]. Raman et al. built upon this work in [56, 55] and proposed the 2P MAC protocol. WiLDNet builds upon 2P to make it robust in the face of high loss and reduce the channel under-utilization due to 802.11's stop and wait recovery mechanism. Specifically we modify 2P's implicit synchronization mechanism as well as build in two adaptive bulk ACK based and FEC based link recovery mechanisms.

**WiMax:** An alternative to WiLD networks is WiMax [67, 36] which has been primarily designed for long distance point to multipoint links. WiMax does present many strengths over a WiFi-based approach at both the PHY and the MAC layers such as better physical coding and better handling of multipath effects. However, the two main limitations of WiMax that make it unsuitable are, (a) it is currently very expensive and, (b) WiMax currently is primarily intended for carriers (like cellular). These features of WiMax make it hard to deploy in the "grass roots" style typical for developing regions.

**802.11 MAC modifications:** Several recent efforts have focused on leveraging off-the-shelf 802.11 hardware to design new MAC protocols for different purposes. Overlay MAC Layer(OML) [57] provides a deployable approach towards implementing a TDMA style MAC on top of the 802.11 MAC using loosely-synchronized clocks to pro-

vide applications and competing nodes better control over the allocation of time-slots. SoftMAC [49] is a platform that can be used to build experimental MAC protocols. MultiMAC [29] builds on SoftMac to provide a platform where multiple MAC layers can co-exist in the networking stack and any one can be chosen on a per-packet basis.

**Performance characterization:** The Roofnet project [20, 10] characterizes the causes of packet losses in an urban multi-hop 802.11b network. The authors conclude that the main source of packet loss in their urban mesh deployment was due to multipath interference. The authors observed a weak correlation between factors such as SNR and link distance on loss rates. Also there was no correlation between loss rate and external WiFi interference. However, the authors in [24] point out that external WiFi interference sources are the main source of packet loss in WiLD networks as they exacerbate the hidden terminal problem. Jamieson et al. [39] experimentally evaluate the limitations of carrier-sense with respect to achieving high throughput in multi-hop environments. Garetto et al. [31] show that CSMA performs very badly in multihop wireless networks, and that this is not due to any particular implementation of a CSMA protocol, but is indeed a general coordination problem in multihop wireless networks. In this paper, we study the limitations of CSMA in WiLD network settings.

**Loss recovery mechanisms:** There is a large body of research literature in wireless and wireline networks that have studied the tradeoffs between different forms of loss recovery mechanisms. Many of the classic error control mechanisms are best summarized in the book by Lin and Costello [45]. Of particular interest for this work are the Berkeley Snoop protocol [15] which provides transport-aware link-layer recovery mechanisms in wireless environments, OverQoS [64] which analyzes the FEC/ARQ tradeoff in variable channel conditions and the Vandermonde codes used for reliable multicast in wireless environments [58].

## 8.3 Mitigating performance degradation using smart antennas

The primary limitations of the link recovery mechanisms presented in chapter 6 are that they lack application agility and reliability. These limitations mainly stem from the lack of steerability of the static directional beam. Chapter 7 discussed the use of phased array antennas to implement better remedies. As far as we know, there has been no experimental research performed in understanding how phased array antennas could be used to mitigate WiFi interference.

Most of the research carried out with steerable and directional antennas has been simulation based and limited to the design of MAC protocols for adhoc networks. These MAC protocol propose alternate channel access mechanisms to CSMA to deal with the deafness problem due to directional antennas.

There has been very little experimental research done using phased array antennas. In [22] the authors propose novel mechanisms to enhance the security of WLANs using phased array antennas. In [54], the authors take a holistic view of building complete systems using directional antenna. The authors present an interacting suite of modular network and MAC layer mechanisms for adaptive controlled for steered and switched beam antennas. The authors conclude that using steerable antennas can provide a very significant improvement of performance in adhoc networks. In [50], the authors discuss various schemes to interface stock 802.11 with adaptive beam forming phased array antennas.

The MobiSteer [65] framework investigates the use of steerable phased array antennas to improve performance of 802.11 links in the context of communication between a moving vehicle and roadside APs. Under controlled experiment setting, MobiSteer improves the performance by a factor 2-4. MobiSteer achieves high performance and maintains a high SNR with the stationary roadside AP by adaptive beam steering.

# Chapter 9

# Conclusion

With the rapid adoption of WLAN technology over the last couple of years, there has been a significant shift in the nature of WLAN deployments. The current generation WLAN deployments are large, unplanned and also extend to outdoor environments. Measurement based studies performed to understand the behavior of these deployments under real environments have highlighted the degraded performance observed in these deployments. These new generation WLAN deployments provide inadequate coverage and suffer from unpredictable performance. Existing approaches that aim to diagnose these performance degradation faults are inefficient because they troubleshoot at too high a level, and are unable to distinguish among the root causes of performance degradation.

This thesis addresses the shortcomings of existing approaches, and proposes a novel architecture based on distributed monitoring of the network to gain fine-grained visibility into the root cause of the fault. Informed remedies are proposed and implemeted that significantly improve the performance of the network. This approach to diagnose root source performance degradation faults is applied to indoor 802.11 networks and outdoor WiFi based Long Distance (WiLD) networks.

The thesis presents the design, implementation and evaluation of detection algorithms for the most commonly observed faults in indoor 802.11 deployments. Based on the fine-grained information collected, threshold based detection algorithms are im-

plemented to detect noise/interference, hidden and capture effect and signal strength variations at the AP. Noise in the network is diagnosed by detecting an increase in the noise floor. Hidden terminals/capture effect are diagnosed by detecting concurrent transmissions by the clients in the network. By measuring the overlap between two simultaneously transmitted frames, we are able to differentiate between hidden terminals and capture effect. Signal strength variations at the AP are diagnosed by detecting concurrent changes in signal strength recorded at the distributed end-points. The detection algorithms have high accuracy in identifying the root source of degradation. By identifying the root source, informed remedies are proposed that improve the performance of the network.

The thesis also presents a detailed study and identifies the primary sources of performance degradation in WiLD networks. The main result is that most of the losses arise due to external WiFi interference on same and adjacent channels. This result is in contrast to loss studies of urban mesh networks, where multipath is reported to be the most significant source of loss. The thesis discusses the limitations of the stock 802.11 protocol's recovery mechanism (rate and frequency adaptation) to recover from external WiFi interference. Loss induced due to external WiFi interference is mitigated by building adaptive link recovery mechanisms like bulk acknowledgments and FEC. Our design of WiLDNet demonstrates a 2-5x improvement in bidirectional TCP throughput over stock 802.11.

In addition to the software based recovery mechanisms, this thesis also studies the use of smart antennas to mitigate the loss due to WiFi interference in WiLD settings. This thesis highlights the primary challenges in dealing with phased array antennas and outlines a hierarchical state decomposition algorithm to deal with the large state space. Our measurement based study has shown that ASIR (RSSI Primary - RSSI Interference) is a good metric for assessing the extent of interference from WiFi sources, and can be used to dynamically steer the antenna beams at the transmitter and receiver nodes.

An important observation we make is that the ASIR metric takes into account network wide information as compared to metrics that are limited to the specific link.

To summarize, the primary contribution of my thesis is that it takes the first step towards building truly self-healing wireless networks. Based on detailed measurement based studies, the thesis identifies the primary sources of performance degradation and proposes informed remedies for indoor 802.11 deployments and outdoor WiLD networks.

## Bibliography


[1] AirTight Networks. http://www.airtightnetworks.net.

[2] Aruba. http://www.arubanetworks.com/.

[3] Atheros. http://www.atheros.com.

[4] Fidelity Comtech. http://www.fidelitycomtech.com/.

[5] Iperf. http://dast.nlanr.net/Projects/Iperf/.

[6] Madwifi. http://sourceforge.net/projects/madwifi.

[7] Soekris Engineering. http://www.soekris.com.

[8] Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Spec, IEEE 802.11 Standard. Technical report, Institute of Electrical and Electronics Engineers, Inc.

[9] Atul Adya, Paramvir Bahl, Ranveer Chandra, and Lili Qiu. Architecture and techniques for diagnosing faults in ieee 802.11 infrastructure networks. In MobiCom '04: Proceedings of the 10th annual international conference on Mobile computing and networking, pages 30–44, New York, NY, USA, 2004. ACM Press.

[10] Daniel Aguayo, John Bicket, Sanjit Biswas, Glenn Judd, and Robert Morris. Link-level measurements from an 802.11b mesh network. SIGCOMM Comput. Commun. Rev., 34(4):121–132, 2004.

[11] AirDefense. Wireless lan security and operational support. available from `http://www.airdefense.net`.

[12] Aditya Akella, Glenn Judd, Srinivasan Seshan, and Peter Steenkiste. Self-management in chaotic wireless deployments. In MobiCom '05: Proceedings of the 11th annual international conference on Mobile computing and networking, pages 185–199, New York, NY, USA, 2005. ACM Press.

[13] Paramvir Bahl, Jitendra Padhye, Lenin Ravindranath, Manpreet Singh, Alec Wolman, and Brian Zill. Dair: A framework for troubleshooting enterprise wireless networks using desktop infrastructure. In Proceedings of the Fourth Workshop on Hot Topics in Networks (HotNets-IV). ACM Press, 2005.

[14] Anand Balachandran, Geoffrey M. Voelker, Paramvir Bahl, and P. Venkat Rangan. Characterizing user behavior and network performance in a public wireless lan. SIGMETRICS Perform. Eval. Rev., 30(1):195–205, 2002.

[15] Hari Balakrishan. Challenges to Reliable Data Transport over Heterogeneous Wireless Networks. PhD thesis, University of California at Berkeley, August 1998.

[16] Hari Balakrishnan, Srinivasan Seshan, Elan Amir, and Randy Katz. Improving TCP/IP Performance over Wireless Networks. In ACM MOBICOM, November 1995.

[17] Bay area wireless users group. http://www.bawug.org.

[18] Pravin Bhagwat, Bhaskaran Raman, and Dheeraj Sanghi. Turning 802.11 Inside-out. ACM SIGCOMM CCR, 34:33–38, January 2004.

[19] Vaduvur Bharghavan, Alan J. Demers, Scott Shenker, and Lixia Zhang. MACAW: A media access protocol for wireless LAN's. In SIGCOMM, pages 212–225, 1994.

[20] John Bicket, Daniel Aguayo, Sanjit Biswas, and Robert Morris. Architecture and evaluation of an unplanned 802.11b mesh network. In MobiCom '05: Proceedings of the 11th annual international conference on Mobile computing and networking, pages 31–42, New York, NY, USA, 2005. ACM Press.

[21] Eric Brewer. Technology Insights for Rural Connectivity. Wireless Communication and Development: A Global Perspective, October 2005.

[22] Joseph Carey and Dirk Grunwald. Enhancing WLAN Security with Smart Antennas: A Physical Layer Response for Information Assurance. In IEEE Vehicular Technology Conference (VTC) 2004, 2004.

[23] Ranveer Chandra, Venkata N. Padmanabhan, and Ming Zhang. Wifiprofiler: co-operative diagnosis in wireless lans. In MobiSys 2006: Proceedings of the 4th international conference on Mobile systems, applications and services, pages 205–219, New York, NY, USA, 2006. ACM Press.

[24] Kameswari Chebrolu, Bhaskaran Raman, and Sayandeep Sen. Long-Distance 802.11b Links: Performance Measurements and Experience. In ACM MOBICOM, 2006.

[25] David D. Clark, Craig Partridge, J. Christopher Ramming, and John T. Wroclawski. A knowledge plane for the internet. In SIGCOMM '03: Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications, pages 3–10, New York, NY, USA, 2003. ACM Press.

[26] Martin V. Clark, K.K. Leung, B. McNair, and Zoran Kostic. Outdoor IEEE 802.11 Cellular Networks: Radio Link Performance. IEEE ICC, 2002.

[27] Connecting Rural Communities with WiFi. http://www.crc.net.nz/index.php.

[28] Dharamsala Wireless-Mesh Community Network. http://www.tibtec.org.

[29] Christian Doerr, Michael Neufeld, Jeff Filfield, Troy Weingart, Douglas C. Sicker, and Dirk Grunwald. MultiMAC - An Adaptive MAC Framework for Dynamic Radio Networking. In IEEE DySPAN, November 2005.

[30] Joseph Dunn, Michael Neufeld, Anmol Sheth, Dirk Grunwald, and John Bennett. A practical cross-layer mechanism for fairness in 802.11 networks. In Proceedings BROADNETS 2004, pages 355–364, Oct 2004.

[31] Michele Garetto, Theodoros Salonidis, and Edward Knightly. Modeling Per-Flow Throughput and Capturing Starvation in CSMA Multi-hop Wireless Networks. In IEEE INFOCOM 2006, April 2006.

[32] Tristan Henderson, David Kotz, and Ilya Abyzov. The changing usage of a mature campus-wide wireless network. In MobiCom '04: Proceedings of the 10th annual international conference on Mobile computing and networking, pages 187–201, New York, NY, USA, 2004. ACM Press.

[33] M. Heusse, F. Rousseau, G. Berger-Sabbatel, and A. Duda. Performance anomaly of 802.11b. In Proceedings of IEEE INFOCOM 2003, San Francisco, USA, March-April 2003.

[34] Martin Heusse, Franck Rousseau, Romaric Guillier, and Andrzej Duda. Idle sense: an optimal access method for high throughput and fairness in rate diverse wireless lans. In SIGCOMM '05: Proceedings of the 2005 conference on Applications, technologies, architectures, and protocols for computer communications, pages 121–132, New York, NY, USA, 2005. ACM Press.

[35] Alefiya Hussain, John Heidemann, and Christos Papadopoulos. A framework for classifying denial of service attacks. In Proceedings of the ACM SIGCOMM Conference, pages 99–110, Karlsruhe, Germany, August 2003. ACM.

[36] IEEE 802.16. IEEE 802.16 WirelessMAN Standard for Wireless Metropolitan Area Networks.

[37] IIT Kanpur. Digital Gangetic Plains. http://www.iitk.ac.in/mladgp/.

[38] International Telecommunications Union. World Telecommunications/ICT Development Report. 2006. http://www.itu.int/ITU-D/ict/publications/wtdr_06/.

[39] Kyle Jamieson, Bret Hull, Allen Miu, and Hari Balakrishnan. Understanding the real-world performance of carrier sense. In E-WIND '05: Proceeding of the 2005 ACM SIGCOMM workshop on Experimental approaches to wireless network design and analysis, pages 52–57, New York, NY, USA, 2005. ACM Press.

[40] Amit P. Jardosh, Krishna N. Ramachandran, Kevin C. Almeroth, and Elizabeth M. Belding-Royer. Understanding link-layer behavior in highly congested ieee 802.11b wireless networks. In E-WIND '05: Proceeding of the 2005 ACM SIGCOMM workshop on Experimental approaches to wireless network design and analysis, pages 11–16, New York, NY, USA, 2005. ACM Press.

[41] Glenn Judd and Peter Steenkiste. Using Emulation to Understand and Improve Wireless Networks and Applications. In NSDI, 2005.

[42] Srikanth Kandula, Dina Katabi, and Jean-Philippe Vasseur. Shrink: A Tool for Failure Diagnosis in IP Networks. In ACM SIGCOMM Workshop on mining network data (MineNet-05), Philadelphia, PA, August 2005.

[43] A. Kochut, A. Vasan, A. Shankar, and A. Agrawala. Sniffing out the correct physical layer capture model in 802.11b. 2th IEEE International Conference on Network Protocols (ICNP), pages 252–261, 2004.

[44] Eddie Kohler, Robert Morris, Benjie Chen, John Jannotti, and M. Frans Kaashoek. The click modular router. ACM Transactions on Computer Systems, 18(3):263–297, August 2000.

[45] Shu Lin and Daniel Costello. Error Control Coding: Fundamentals and Applications. Prentice Hall, 1983.

[46] Ratul Mahajan, David Wetherall, and Tom Anderson. Understanding bgp misconfiguration. In SIGCOMM '02: Proceedings of the 2002 conference on Applications, technologies, architectures, and protocols for computer communications, pages 3–16, New York, NY, USA, 2002. ACM Press.

[47] David L. Mills. Internet Time Synchronization: The Network Time Protocol. In Global States and Time in Distributed Systems, IEEE Computer Society Press. 1994.

[48] Shridhar Mubaraq Mishra, John Hwang, Dick Filippini, Tom Du, Reza Moazzami, and Lakshminarayanan Subramanian. Economic Analysis of Networking Technologies for Rural Developing Regions. Workshop on Internet and Network Economics, 2005.

[49] Michael Neufeld, Jeff Fifield, Christian Doerr, Anmol Sheth, and Dirk Grunwald. SoftMAC - Flexible Wireless Research Platform. In HotNets-IV, November 2005.

[50] Michael Neufeld and Dirk Grunwald. Using phase array antennas with the 802.11 mac protocol. In BROADNETS '04: Proceedings of the First International Conference on Broadband Networks (BROADNETS'04), pages 733–735, Washington, DC, USA, 2004. IEEE Computer Society.

[51] V. N. Padmanabhan, S. Ramabhadran, and J. Padhye. Netprofiler: Profiling wide-area networks using peer cooperation. In Proceedings of the Fourth International Workshop on Peer-to-Peer Systems (IPTPS), New York, NY, USA, 2005.

[52] Rabin Patra, Sergiu Nedevschi, Sonesh Surana, Anmol Sheth, Lakshminarayanan Subramanian, and Eric Brewer. Wildnet: Design and implementation of high performance wifi based long distance networks. In NSDI 2007: Proceedings of the 4th USENIX Symposium on Networked Systems Design and Implementation. USENIX, 2007.

[53] Lili Qiu, Paramvir Bahl, Ananth Rao, and Lidong Zhou. Troubleshooting multihop wireless networks. SIGMETRICS Perform. Eval. Rev., 33(1):380–381, 2005.

[54] Santivanez Wiggins R. Ramanathan, J. Redi and Polit. Ad hoc networking with directional antennas: A complete system solution,. In IEEE Journal on Selected Areas in Communications: Special Issue on Wireless Ad Hoc Networks, March 2005.

[55] Bhaskaran Raman and Kameswari Chebrolu. Revisiting MAC Design for an 802.11-based Mesh Network. In HotNets-III, November 2004.

[56] Bhaskaran Raman and Kameswari Chebrolu. Design and Evaluation of a new MAC Protocol for Long-Distance 802.11 Mesh Networks. In ACM MOBICOM, August 2005.

[57] Ananth Rao and Ion Stoica. An Overlay MAC layer for 802.11 Networks. In MOBISYS, Seattle,WA, USA, June 2005.

[58] Luigi Rizzo. Effective erasure codes for reliable computer communication protocols. ACM Computer Communication Review, 27(2):24–36, April 1997.

[59] Maya Rodrig, Charles Reis, Ratul Mahajan, David Wetherall, and John Zahorjan. Measurement-based characterization of 802.11 in a hotspot setting. In E-WIND '05: Proceeding of the 2005 ACM SIGCOMM workshop on Experimental approaches to wireless network design and analysis, pages 5–10, New York, NY, USA, 2005. ACM Press.

[60] Seattle wireless. http://www.seattlewireless.net.

[61] Anmol Sheth, Christian Doerr, Dirk Grunwald, Richard Han, and Douglas Sicker. Mojo: a distributed physical layer anomaly detection system for 802.11 wlans. In MobiSys 2006: Proceedings of the 4th international conference on Mobile systems, applications and services, pages 191–204, New York, NY, USA, 2006. ACM Press.

[62] Anmol Sheth, Sergiu Nedevschi, Rabin Patra, Sonesh Surana, Lakshminarayanan Subramanian, and Eric Brewer. Packet loss characterization in wifi-based long distance networks. In IEEE Infocom 2007: Proceedings of the 26th Annual IEEE Conference on Computer Communications. IEEE, 2007.

[63] Spirent Communications. http://www.spirentcom.com.

[64] Lakshminarayanan Subramanian, Ion Stoica, Hari Balakrishnan, and Randy Katz. OverQoS: An Overlay Based Architecture for Enhancing Internet QoS. In USENIX/ACM NSDI, March 2004.

[65] Kannan Dhanasekaran1 Andreas Timm-Giel Vishnu Navda1, Anand Prabhu Subramanian and Samir R. Das. Mobisteer: Using steerable beam directional antenna for vehicular network access. In MobiSys 2007: Proceedings of the 5th international conference on Mobile systems, applications and services, New York, NY, USA, 2007. ACM Press.

[66] Wavesat. WiMax 3.5GHz mini-PCI Reference Design Kit. http://www.wavesat.com/products/mini-pci.html.

[67] WiMAX forum. http://www.wimaxforum.org.

[68] Kaixin Xu, M. Gerla, and Sang Bae. How effective is the ieee 802.11 rts/cts handshake in ad hoc networks? In Proceedings of the Global Telecommunications Conference, GLOBECOM, pages 72–76. IEEE, 2002.

[69] Jihwang Yeo, Moustafa Youssef, and Ashok Agrawala. A framework for wireless lan monitoring and its applications. In WiSe '04: Proceedings of the 2004 ACM workshop on Wireless security, pages 70–79. ACM Press, 2004.

[70] Jihwang Yeo, Moustafa Youssef, Tristan Henderson, and Ashok Agrawala. An accurate technique for measuring the wireless side of wireless networks. In WiTMeMo '05: Papers presented at the 2005 workshop on Wireless traffic measurements and modeling, pages 13–18, Berkeley, CA, USA, 2005. USENIX Association.