MAC-PHY Cross-layer Techniques for Simultaneous Multiuser Communication in Wireless Networks

by

Dola Saha

M.S., University of Colorado Boulder, 2008

B.Tech., University of Kalyani, India, 2002

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

2013

This thesis entitled:
MAC-PHY Cross-layer Techniques for Simultaneous Multiuser Communication in Wireless Networks
written by Dola Saha
has been approved for the Department of Computer Science

_____
Prof. Dirk Grunwald

_____
Prof. Douglas Sicker

_____
Prof. Timothy X Brown

_____
Prof. Shivakant Mishra

_____
Prof. Kenneth R. Baker

_____
Prof. Dipankar Raychaudhuri

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.

Saha, Dola (Ph.D., Computer Science)

MAC-PHY Cross-layer Techniques for Simultaneous Multiuser Communication in Wireless Networks

Thesis directed by Prof. Dirk Grunwald and Prof. Douglas Sicker

Layering in wireless networks provide clear abstractions to how various resources are managed for a particular communication link. However, the unpredictability of the wireless channel presents great challenge to these clear abstractions. Often, optimizations in these layers are not transparent to others. This creates a necessity to violate the modular approach and share crosslayer information to modify each layer's functionalities, which eventually improves the overall performance of the network.

In this thesis, novel MAC-PHY crosslayer protocols have been designed, implemented and evaluated. These protocols provide unprecedented gain in various aspects of a wireless network, by facilitating simultaneous multiuser communication. By harnessing the untapped potential of the various signal processing subsystems in the physical layer, these protocols are able to increase network throughput, make certain group communications faster and enable covert communication. Using reconfigurable hardware to expose physical layer information, improvement is achieved in higher layers. Furthermore, it is also important to modify the physical layer based on the feedback from higher layers. The two-way handshaking changes the conventional modular approach and allows implementation of simultaneous communication in wireless domain.

To make the crosslayer techniques practical, this thesis presents clear implementation steps to embed these concepts as an extension to common wireless network protocols and evaluate those using practical experiments and radio measurements. Through these techniques we are able to show practical benefits from a mutable radio and a crosslayer approach to protocol design for next generation, high bandwidth wireless networks.

Dedication

To Aveek and Adri.

Acknowledgements

First of all, I would like to thank my PhD advisors Prof. Dirk Grunwald and Prof. Douglas Sicker for their support, advice and guidance throughout my graduate studies.

Would like to thank my research collaborator, Aveek Dutta for his invaluable support in helping me with prototyping my research concepts, which makes my research even more valuable due to its practicality.

Also, would like to thank Prof. Dipankar Raychaudhuri and Ivan Seskar of Rutgers University for their support and cooperation over the years.

Thank you to all the members of my thesis committee, Prof. Tim Brown, Prof. Shivakant Mishra and Prof. Ken Baker for their insightful comments that has helped me enrich my research.

And lastly, a big thank you to all my family members, specially my parents for believing in me and my friends for helping me endure the emotional roller coaster ride of completing a doctorate degree.

Contents

Chapter

Tables

Table

Figures

Figure

Chapter 1

Introduction

Studies [2] have predicted exponential growth of mobile devices in the recent future that will lead to starvation of the shared wireless communication medium. Unfortunately, the current network infrastructure is not capable of handling the high demand of the applications, like high definition video streaming, online gaming and file sharing, which will constitute the majority of data traffic for next generation wireless networks. Not only do these nodes have high bandwidth requirements, they also have to be reliable at the same time. Catering to these new generations of system performance specifications requires a collective effort in many frontiers of network engineering: whether it is designing high capacity waveform at the physical layer or optimization at the MAC and routing layers.

Advances in networking have been accelerated by the use of abstractions, such as "layering", and the ability to apply those abstractions across multiple communication media. Wireless communication provides the greatest challenge to these clean abstractions because of its inherent nature of unpredictability in the presence of lossy communication medium. Wireless networking and its limitations have been studied for years, but the increasing number of users and demand in bandwidth create a necessity to improve the network performance by re-thinking several aspects of the network design. Furthermore, the tendency to emulate wired network protocols in the wireless domain, has led to artificial limitations in wireless networks. The layered architecture of the network has led researchers to focus on improvements in specific areas without considering its impact on others. This has often led to limited gains in network performance. Therefore, it is imperative to re-think the approach towards solving these problems for the next generation networks.

As wireless networks get denser and require higher bandwidth, efficient sharing of the wireless

medium is required. Benefits from contention based (CSMA) or time (TDMA) and frequency based (FDMA) sharing of the wireless medium is limited by the ability to avoid interference from other users in the network. Noise and interference are fundamental aspects of communication, and are exceptionally important for wireless communication because it is more difficult to contain propagation without waveguides such as wires and fibers. Therefore, sharing is best achieved by employing some form of simultaneous multiuser communication method. Multiuser communication using multiple access technique is quite prevalent in modern wireless communication protocols. Typically, these techniques require *orthogonal channels* to mitigate interference from simultaneous access of the wireless channel by multiple users.

The primary enabler for this type of multiuser communication is the underlying radio PHY or more precisely, the waveform used for the communication. These waveforms can range from single-carrier to multiple-carrier waveforms and from underlay (below noise floor) to overlay (above noise floor) waveforms. Single carrier communication, like CDMA focuses on decoding the strongest signal while discarding anything else as noise or interference. Network protocol designers, both at the physical and the network layers, have long considered interference in wireless protocols as a problem to be avoided. Avoiding interference or noise is a design choice that limits the scope of simultaneous multiuser communication. Multi-carrier waveforms, like OFDM, on the other hand, utilizes the inherent orthogonality of the carriers to facilitate simultaneous multiuser communication while mitigating interference among parallel communication channels making it a good choice for designing protocols for this type of channel sharing.

Methods of implementing multiuser techniques is not limited by the underlying waveforms but can be extended to higher layers in the protocol stack. Concepts of sharing higher layer primitives like MAC layer packet entities have been found in the literature as well. These techniques are typically independent of the underlying waveform or the PHY layer. Hence, they provide greater flexibility in terms of its application across different types of wireless protocols. However, the drawback of these techniques residing within a protocol layer is that they cannot utilize information from multiple layers to improve network performance. Hence, it is intuitive to re-think the simultaneous multiuser communication from a cross-layer point of view to extract information that otherwise would have remained unutilized.

The benefit from all these multiuser techniques can be measured in many ways. Whether it is gain

in the aggregate network throughput or reduced overhead in network management and control or in some cases benefits such as making a communication secret. Therefore, it is important to consider the methods to innovate simultaneous multiuser protocols and measure the multifarious benefits in various aspects of wireless networking.

Motivated by these fundamental problems, in this thesis, we present a collection of novel protocol design techniques with the following characteristics: a) provide a MAC-PHY crosslayer approach for protocol design, b) efficient sharing of the wireless medium by employing simultaneous multiuser communications and c) meet bandwidth demand requirements by improving aggregate throughput and lowering latency in the network. In order to incorporate these characteristics to network protocols, the hardware that is used as a network interface will have to have significant new features and capabilities. Conventional hardware implementing the MAC or the PHY are fixed-function with limited tuning parameters. In the light of these new techniques, we need a highly programmable hardware that would allow fine-grained control over the various aspects of the MAC and PHY that is beyond the conventional tuning parameters, like transmitter power and data-rate. Hence, in this thesis, we embrace the flexibility provided by Software Defined Radio (SDR) to implement and evaluate these novel techniques.

Physical (PHY) layer designing has been a challenge to researchers for years, as this requires knowledge of communication theory, understanding the functioning of higher layers, as well as the knowledge for hardware development. ASIC was the sole choice of hardware platform for a long time. However, the improvements in communication theory and the intelligence expected in physical layer have increased the demand for a flexible physical layer system, which would interact with higher layers for signal processing. Recent popularity of SDRs allowed us to modify the physical layer as per the requirement of the protocol designer to harness various PHY related information that was unavailable until now. The superior flexibility of SDRs allow protocol designer to innovate cross-layer designs by thinking of the MAC and PHY as one entity operating in unison. As the intelligence of the PHY is improved, it creates a demand for re-thinking all the higher layer protocols, which has to handle the extra gain received, and has to react accordingly. A new avenue has opened to the upper layers including both MAC and transport layers, where researchers are redesigning efficient protocols to fully utilize the intelligence of the physical layer. To successfully com-

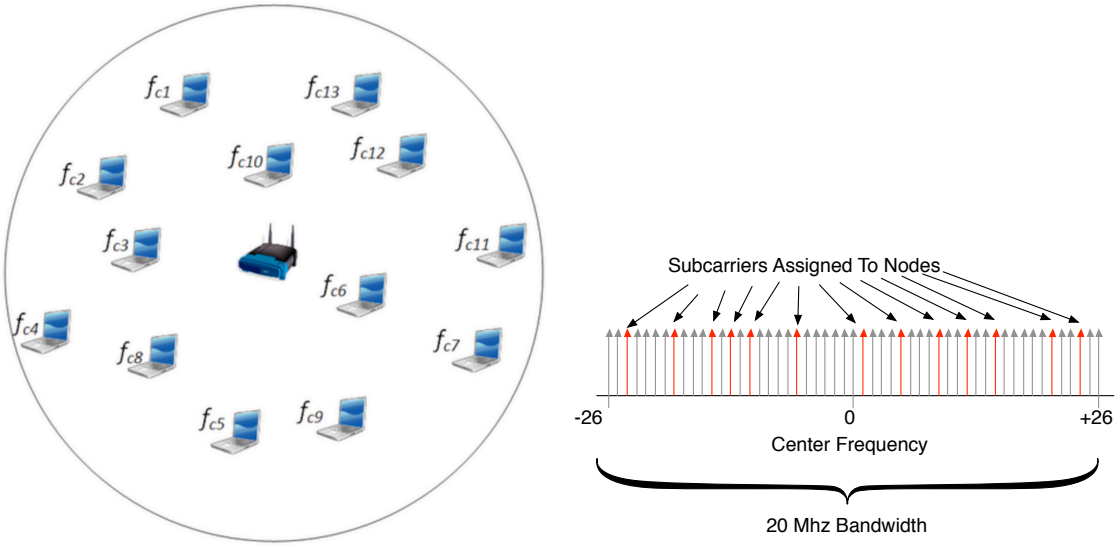
plete this procedure, MAC and transport layers may need to interact with the physical layer on demand and instantaneously for decision making in signal processing.

Propagation characteristics in wireless medium are difficult to model due to its dependency on the surrounding environment, both structural and in the presence of other sources of transmission. Also, sometimes it becomes very hard to understand the implications of a new protocol unless it is implemented in a system level. Hence, in this thesis, we have verified the theoretical concepts by system level implementation in SDR prototypes and evaluated those by practical experiments and measurements from testbeds. The goal of this thesis is to redesign various aspects of wireless networking to meet the demands of next generation wireless networks, while keeping the system backward compatible as much as possible for seamless migration to the next generation wireless networks.

In this thesis, we present six novel ideas, which implements various MAC-PHY crosslayer protocols for simultaneous multiuser communication for next generation wireless networks. This work is a positive step towards cross-layer implementation that has been evaluated by testbed experiments. We describe the different protocols briefly.

1.0.1 SMACK

In this research, we demonstrate how simultaneous transmission can be used to implement a **reliable broadcast** for an infrastructure and peer-to-peer network, using a prototype SDR. We utilize the orthogonality of the subcarriers in the Orthogonal Frequency Division Multiplexing (OFDM) waveform for simultaneous communication. Multiple nodes use pre-assigned subcarriers to transmit orthogonal tones, which are detected at the receiver by Fourier Transform. Thus, the presence of a tone at the receiver indicate an acknowledgment of reception of a broadcast or a multicast. Figure 1.1 shows the frequency assignment in SMACK. Either all of the subcarriers can be assigned, as in a dense network, or a sparse assignment is also possible, that results in a non-contiguous OFDM. The key property that makes this technique successful is the orthogonality of the OFDM subcarriers that makes the simultaneous acknowledgments inherently free of interference. This technique exploits **simultaneous transmission** to reduce the cost of reliable multicast by **orders of magnitude**, and can also be extended to common group communication primitives, such as



(a) Subcarrier assignment in a network

(b) Non-contiguous OFDM transmission

Figure 1.1: Schematic illustration of SMACKs using OFDM

anycast, broadcast, leader election and others. To validate the system, we have implemented the technique using a prototype SDR and evaluated the protocol in a testbed with various subcarrier assignments. We've shown that by using, rather than fighting against, the properties of the wireless physical media, we can develop robust signaling primitives that are both practical and allow innovative algorithms. This work has been extended in parallel polling in a centralized network and as an active radar in vehicular networking. We discuss the protocol details along with implementation and evaluation in 5.

1.0.2 PAMAC

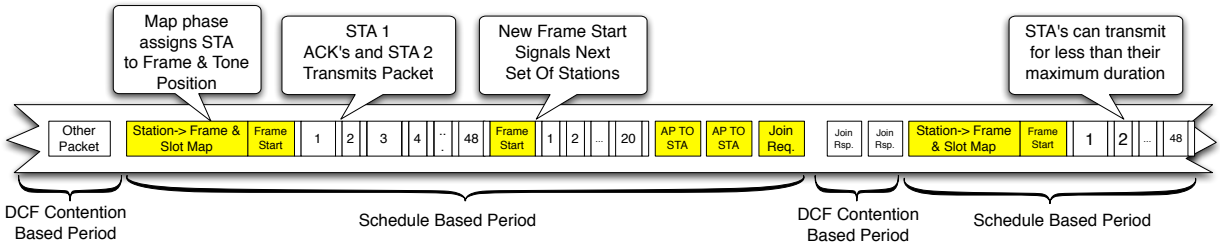


Figure 1.2: PAMAC Scheduling

PAMC or PHY Aided MAC is an intelligent application of the OFDM subcarriers to lower latency in scheduling. This technique utilizes the simultaneous transmission technique presented in SMACK, to have clients signal the Access Point (AP) whether they have packets to send. By employing energy detection of the the orthogonal tones over multiple time slots, the AP gets the following information: a) which stations have packets to send and b) whether the traffic load is high, medium or low. Then, the AP schedules clients efficiently while wasting little of the spectrum on signaling overhead. Figure 1.2 shows the time sequence of various stages in the protocol. After the AP receives simultaneous tones from users defining their respective queue lengths, it sends out a schedule in form of a *start frame*. The client nodes follow with uplink traffic one after the other based on the schedule. The proposed protocol of parallel polling is a) fast, since no packet transmission is required for polling responses and all clients respond concurrently; b) reliable, as the poll response is contention free and c) scalable. We demonstrate the feasibility of implementing such a system using a FPGA based prototype SDR. We then show how the MAC protocol can scale using the QualNet network simulator and compare the performance to a contention based protocol. The results and protocol design is discussed in chapter 4.

1.0.3 Active Radar

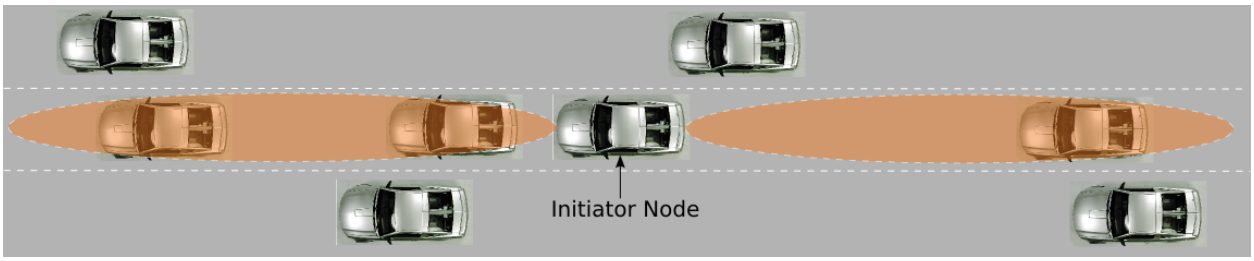


Figure 1.3: Vehicles in Highway

In this application, we also utilize simultaneous transmission techniques to design a collision detection system in vehicular network. We present a cooperative technology that uses multicarrier wireless communication to detect and disseminate the possibility of a collision. Using precise timing and synchronization, we can detect the distance of each of the vehicles, their current velocity and current acceleration or

deceleration conditions, and can raise an alarm if there is a sudden change in these states. Using simultaneous, multi-party acknowledgments, we can rapidly gather this information about a number of vehicles in an efficient manner. Figure 1.3 shows a typical highway traffic where the *initiator* node broadcasts a message and all the vehicles able to decode the message, sends an ACK using randomly chosen OFDM subcarriers. By estimating the time-of-arrival of these individual tones the *initiator* is able to estimate the relative distances of the vehicles. Repeating this cycles provides estimate of velocity and acceleration which is detailed in chapter 6.

1.0.4 GRaTIS

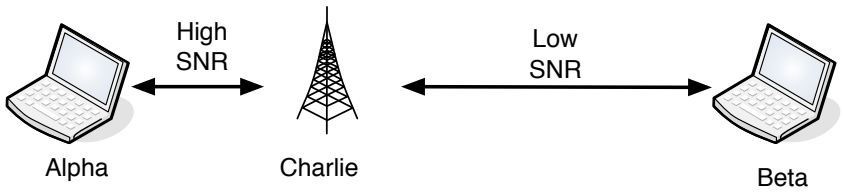


Figure 1.4: Example of spatial rate diversity and potential use case for GRaTIS

This research exploits the SNR diversity of the nodes to combine packets intended for two receivers and transmit those together in the time normally needed to transmit one of the packets. Figure 1.4 shows a typical scenario where the SNR diversity can be utilized for simultaneous communication. This technique increases the aggregate throughput precisely when it is most needed - when the network is busy and suffers from rate unfairness. We reinterpret the constellations already available for conventional wireless links and provide group rates, which result in higher network throughput with no hardware changes, both at the transmitter and the receiver. We have implemented the protocol in SDR and experimental results show several use of GRaTIS that yields unforeseen gains in throughput in wireless networks. Applying GRaTIS on real-time packet trace analysis reveals that even with a few simple combinations, we can gain significant airtime. Through our analysis we show that GRaTIS provides better error performance than other contemporary simultaneous packet transmission techniques, making it a suitable candidate for emerging wireless networks. Chapter 7 explains the protocol in greater detail.

1.0.5 Dirty Constellation

This research leverages the variability in the wireless channel and hardware conditions to encode a covert channel in the physical layer of common wireless communication protocols. Covert channels are a form of multiuser communication because the adversary will have to receive a meaningful message while the covert message is delivered to another user. A typical setup is shown in figure 1.5 with two mobile users and a common transmitter. Hidden messages rely on a very low probability of detection to be successful. The cover traffic in this method is the baseband modulation constellation. Packet sharing techniques and pre-distortion of the modulated symbols of a decoy packet allows the transmission of a secondary covert message while making it statistically undetectable to an adversary. We demonstrate the technique by implementing it in hardware, on top of an 802.11a/g PHY layer, using a SDR and analyze the undetectability of the scheme through a variety of common radio measurements and statistical tests. Higher datarate, very low probability of detection coupled with ease of implementation within existing protocol stacks make Dirty Constellation a very successful method to implement covert channels at the physical layer. In chapter 8, we discuss the protocol in more details and evaluate it by extensive measurements and analysis.

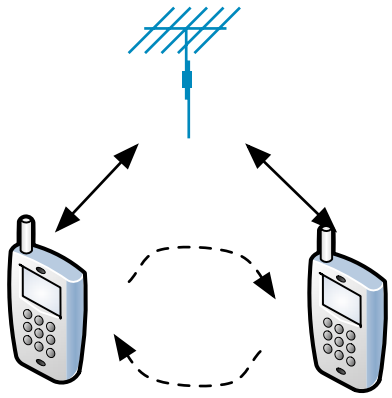


Figure 1.5: Undetected Side-Channel Communication

1.1 Thesis Contributions

This thesis addresses the challenges faced in next generation wireless networks by providing efficient, practical solutions with tractable implementation using SDRs. Through our research and subsequent evaluations, we have been able to show that by jointly designing the MAC and PHY we are able to harness unprecedented gains in several aspects of networking. Thus, this thesis is an example of PHY enabled MAC layer design that utilizes the PHY layer to implement novel MAC layer protocols. The thesis utilizes the presence of *n*oise in wireless medium to design concurrent communication between multiple users improving aggregate throughput of the network, without hampering single link performance, while keeping the system backward compatible. Therefore, the contributions of this thesis are as follows:

- We have utilized the PHY layer beyond just a method to generate waveform, by making it more accessible by the MAC layer. By focusing on the capabilities of the DSP subsystems we are able to innovate efficient MAC-PHY crosslayer protocols for high bandwidth wireless networks.
- We have designed MAC layer protocols that carefully control the PHY to implement simultaneous multiuser communication in wireless networks.
- We have applied this generic technique across multiple dimensions of wireless network: Reliable broadcast, centralized polling, simultaneous packet transmission to multiple users, vehicular safety and lastly, covert communication.
- To show the practicality of these protocols we have implemented these on a SDR prototype platform as a plausible extension to the 802.11a/g and and other OFDM based PHY layers.
- We have evaluated the performance of each of the techniques by extensive theoretical and experimental analysis and in some cases through rigorous simulations.

1.2 Thesis Organization

The thesis is structured in the following way. Chapter 2 provides a theoretical background of OFDM and its properties that are crucial to implement some of the crosslayer techniques in this thesis. In chap-

ter 3 we discuss the hardware and software components of the chosen SDR prototype, which is used for evaluating the various protocols. Then, we present the protocol design, implementation and evaluation of the simultaneous acknowledgments scheme in chapter 5. The various extrapolations of this physical layer signaling mechanism in higher layers is described in chapter 4 and chapter 6. This is followed by the multiuser techniques at the MAC layer: GRaTIS in chapter 7 and Dirty Constellations in chapter 8. Finally we conclude in chapter 9.

Chapter 2

Theoretical Background

The technique of using multiple bit streams for transmission was used more than 50 years ago in voice data transmission [3] over wires. After that, use of multicarrier communication have become more familiar to researchers in other communication domains. Orthogonal Frequency Division Multiplexing (OFDM) [4] is a special type of Multicarrier Modulation (MCM), where the data stream is divided into several bit stream and the modulated subcarriers are spaced closely, although overlapping in such a manner that they do not interfere with each other. In older Frequency Division Multiplex (FDM) systems, a band-pass filter was used to separate the intended frequencies and demodulation was done using standard procedure. However, in OFDM, these steep bandpass filters are not used to separate the subcarriers. Instead, using FFT, the time domain signal is converted to frequency domain, which regenerates the subcarrier information. With successive studies after the first patent in 1970, researchers came to a point where it seemed that multicarrier modulation [5] is actually feasible in the wireless domain.

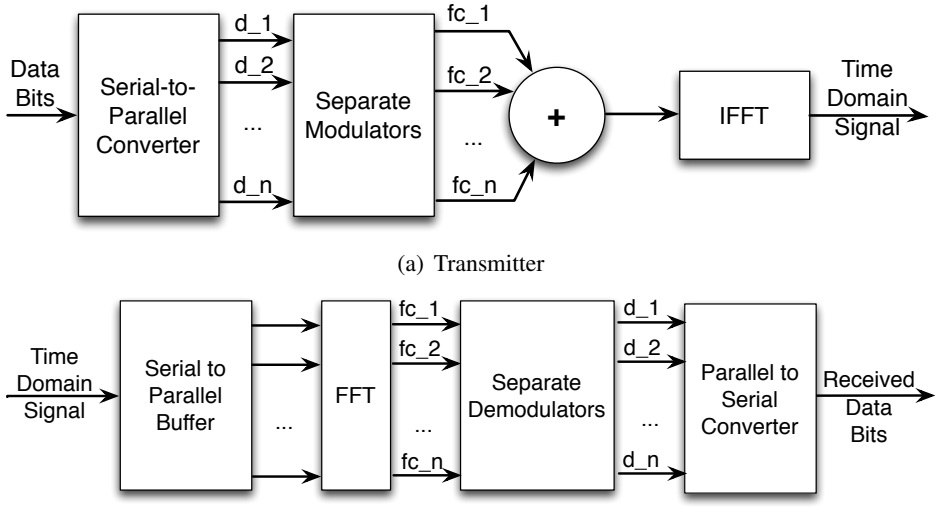
The fact that the component sinusoids of an OFDM signal can be easily aggregated to form time domain signals as in eq. 2.1 empowers us to use any part of the spectrum by suitably selecting the the spectral coefficients x(k).

$$X(n) = \sum_{i=0}^{N-1} x(k) \sin(\frac{2\pi kn}{N}) - j \sum_{i=0}^{N-1} x(k) \cos(\frac{2\pi kn}{N}) \tag{2.1}$$

Here, $X(n)$ is the value of the signal at time n which is composed of frequencies denoted by $2\pi kn/N$, k is the index of frequency over N spectral components which divides the available bandwidth with equal spacing and $x(k)$ gives the value of the spectrum at k^{th} frequency.

This leads to the notion of non-contiguous OFDM (NC-OFDM) which can degenerate to even a single frequency or **tone**. A Fourier transform of such an NC-OFDM signal reveals the spectral energy and can be detected using fairly simple methods. The simplicity of OFDM and ease of implementation of such a system has led us to innovate the newer protocols and signalling methods described in this thesis.

Figure 2.1 shows the basic transmitter and receiver design of a multicarrier transceiver. The data for transmission is divided into parallel bit streams, which are modulated separately, such that each of them have a different carrier frequency. Then, the frequency domain signal is converted to time domain by IFFT, which ensures that closely spaced subcarriers overlapping in spectra are mutually orthogonal and they do not interfere with each other. At the receiver side, the time domain signal is buffered first to feed into the FFT block, which converts the time domain signal to its frequency domain components. Once the subcarriers are separated, they can be demodulated and the serial bit streams can be converted to a single data stream by a Parallel-to-Serial converter.



(a) Transmitter

(b) Receiver

Figure 2.1: Basic Multicarrier Transceiver

2.1 Challenges in OFDM

There are however a few challenges of using Orthogonal Frequency Division Multiplexing (OFDM) in wireless domain, which we discuss in details.

2.1.1 Packet Detection and Synchronization

Wireless is a distributed domain and there does not exist any synchronized clock for all the transceivers. But, to receive the data correctly, it is of absolute importance to start the FFT at a precise sample. To detect the start of a packet, two common mechanisms [6] are used. One is correlation with stored samples, and the second one is correlation of previously received samples. After the start of the packet is detected, the signal is processed as shown in figure 2.1.

Correlation determines the degree of similarity between two signals. If the signals are identical, then the correlation coefficient is 1 and if they are completely different, the correlation coefficient is 0. This property is utilized by the packet detection block to detect the start of the data symbols, which is preceded by a known sequence of repetitive preamble symbols. The receiver performs an autocorrelation with a delayed version of the signal, where the delay is equal to the symbol period. The repititive nature of the preamble and the delayed autocorrelation function provides us a similarity factor which helps to identify a valid packet. The receiver also performs a cross-correlation between the received signal and the locally stored copy of the expected sequence at the begining of the packet. In 802.11a/g, both the techniques are used to detect a valid packet and determine the exact start of data symbols, by utilizing two sets of predefined preamble symbols.

2.1.2 Channel Estimation and Equalization

Channel fading is a common issue in wireless communication, which distorts the signal in both amplitude and phase. In multicarrier communication, the channel is defined by as a set of subcarriers, and we notice that each subcarrier fading is different from other. But, subcarriers can fade in a linear or non-linear fashion. To cope with this kind of fading in multicarrier modulation, researchers utilized some subcarriers *as pilots* to capture channel state information. These pilot subcarriers are inserted at regular intervals with a

known amplitude and phase. Figure 2.2 shows four pilot subcarriers inserted in a 52 subcarrier system used in 802.11a/g technology at 2.4GHz. The receiver estimates the channel state information from the pilots and calculates the inter-pilot channel state by using linear or non-linear interpolation methods. These estimates are used to equalize the data carriers in order to reinstate their modulation levels required for successful decoding. Figure 2.3 shows the transmitted OFDM signal captured by a signal analyzer, which contains the four pilot subcarriers and 48 data subcarriers, as shown in figure 2.2. All the subcarriers overlap in the frequency domain, although the orthogonality of each of the subcarriers is maintained.

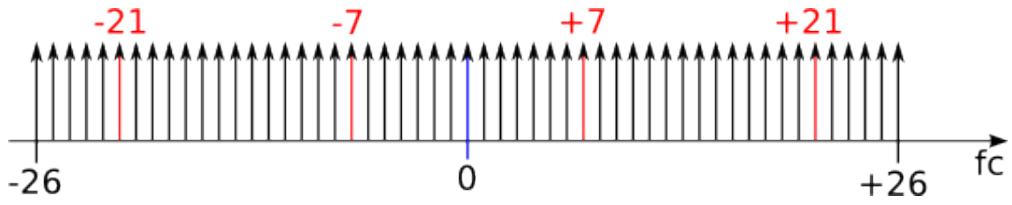


Figure 2.2: Subcarrier Frequencies in OFDM (Pilots are inserted at Subcarriers -21, -7, +7, +21; Rest are Data Subcarriers)

Figure 2.4 shows the signal before and after equalization of a Quadrature Phase Shift Keying (QPSK) modulated data signal. The unequalized data, captured over the air, is distorted to such an extent that it is difficult to estimate each subcarrier. However, a linear equalization results in four distinct blobs showing four different possible states for QPSK. Hence, equalization has to be done before the signal is passed to the demodulation block. In more practical systems, we have another equalizer block in figure 2.1 at the receiver.

2.1.3 Multipath and Inter-Symbol Interference

Multipath propagation is a key feature in wireless communication, where the signal propagates in multiple paths and all the reflected signals reach the receiver after some delay. Such multipath propagation can overlap constructively or destructively at the receiver. Destructive interference between two or more reflected signals can creat deep spectral nulls in the frequency passband of received radio signals. In this scenario, a single carrier communication will not be able to effectively recover the information encoded in the subcarrier. However, with the equalization technique described in §2.1.2, OFDM recovers the informa-

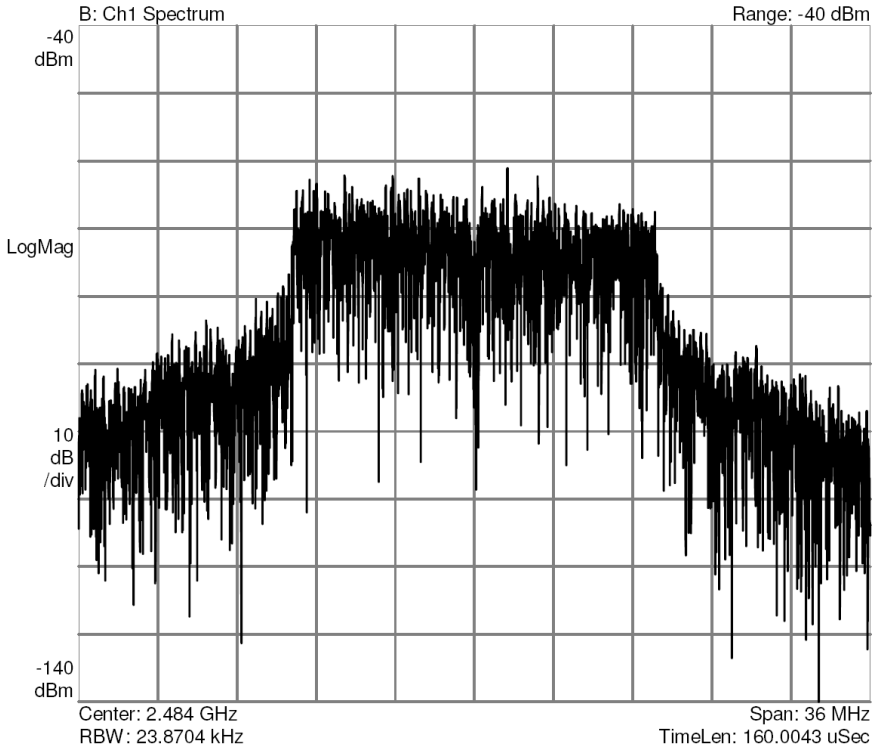
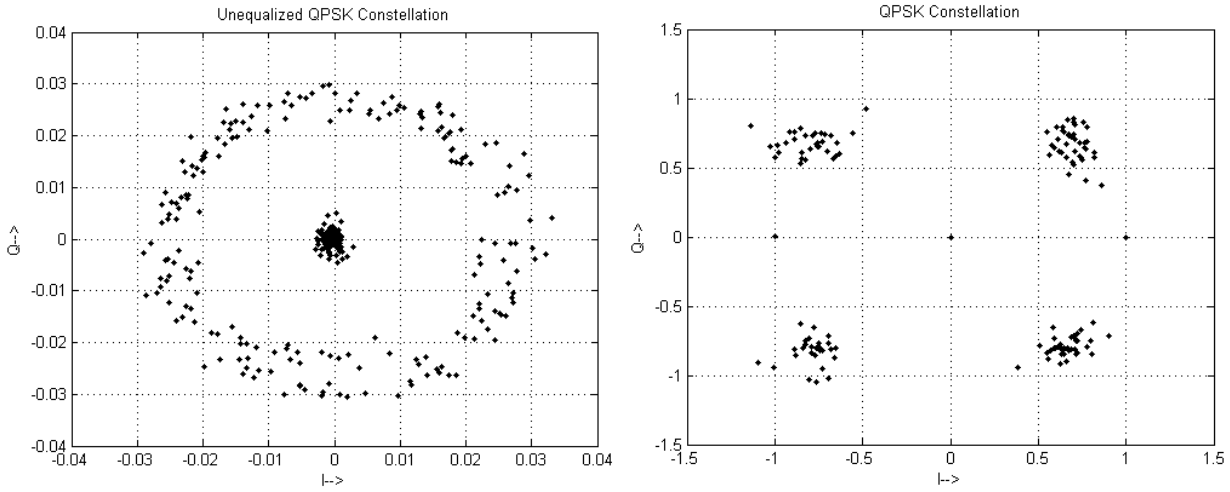


Figure 2.3: Orthogonal Frequency Division Multiplexed Data Received with Signal Analyzer in 2.4GHz, containing 48 Data Subcarriers and 4 Pilot Subcarriers



(a) Unequalized QPSK Constellation

(b) Equalized QPSK Constellation

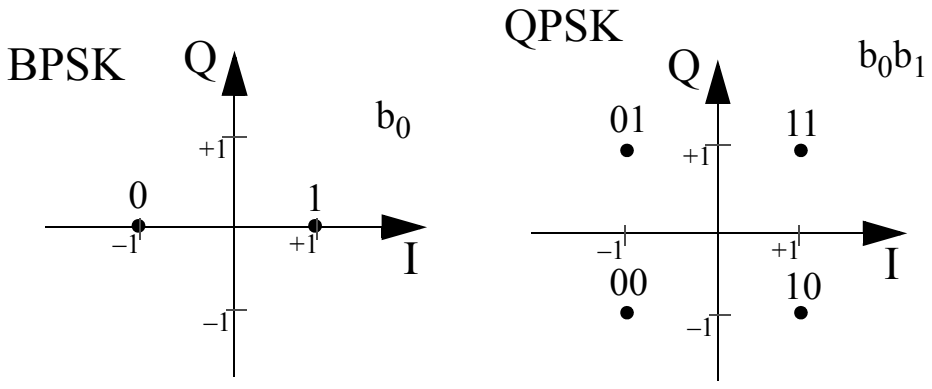
Figure 2.4: Equalization of QPSK

tion encoded in the faded subcarriers.

Signal processing becomes complex when multipath effects cause contiguous symbols to interfere with each other. This phenomenon is known as Inter-Symbol Interference (ISI), where the signal of previous symbol is added up as a noise in the time domain. ISI can be reduced by inserting a guard interval in between two symbols. This guard interval is such chosen that the reflected signals due to multipath will fade such that its power is not enough to interfere with the next symbol. In an OFDM symbol cyclic prefix is transmitted during this guard interval. It is a repetition of some of the samples from the end of the symbol at the beginning. The purpose is to allow multipath to settle before the main data arrives at the receiver. The receiver is normally arranged to decode the signal after it has settled because this is when the frequencies become orthogonal to one another. In 802.11 based OFDM in 2.4GHz band, the effective transmission of one symbol is done in $3.2\mu s$, and the cyclic prefix is $0.8\mu s$, constituting a symbol duration of $4\mu s$.

2.1.4 Modulation and Demodulation

The modulation is performed at the transmitter side after the interleaving. The binary data is grouped into one or more bits, depending on the modulation type (BPSK, QPSK, 16-QAM, 64-QAM). Each of these groups is then converted into a complex number representing constellation points. Figure 2.5 shows the encoding of BPSK and QPSK modulations.



(a) Mapping of BPSK Modulation

(b) Mapping of QPSK Modulation

Figure 2.5: Constellation Bit Encoding

The output value, d, is formed by multiplying the resulting $(I + jQ)$ value by a normalization factor K_{MOD}, depending on the type of modulation. $d = (I + jQ) \times K_{MOD}$. The purpose of the normalization factor is to achieve the same average power for all mappings.

The demodulation is performed at the receiver side after the equalization. The equalizer de-rotates the constellation and restores the constellation to its original configuration. The demodulator is a maximum likelihood baseband demodulator, which performs threshold test as per the symbol energy. Decision boundaries are given by the perpendicular bisector of the line joining the two constellation points. This gives the optimum decoder as it minimizes the Euclidean distance between received signal and the nearest constellation point.

The modulation rate information is encoded in the RATE field of the SIGNAL Symbol in 802.11a/g packet, which is the first symbol in the packet. Once this symbol is demodulated, and the modulation rate information is decoded, the rest of the symbols are decoded using the modulation rate specified in the SIGNAL symbol.

2.2 OFDM as Facilitator

Even though the technology is prevalent for approximately 20 years and standards like IEEE 802.11a/g, IEEE 802.16 and LTE have embraced OFDM/OFDMA modulation techniques, we do not notice any intelligent use of the technology rather than simply using it as a medium of transmission at higher data rates. In this thesis, we will focus on how to efficiently use the orthogonal subcarriers in higher layers. We also show that simple modifications in modulation and demodulation can yield unprecedented gains in network performance. Also, covert communication can be achieved by similar modification in modulation and demodulation blocks of the physical layer. For these techniques to implement and realize in a system, we argue that reconfigurability of hardware is of utmost importance.

Chapter 3

Reconfigurable Hardware

The use of software defined radio [7] (SDR) and its evolution for use in cognitive radio [8] (CR) has been studied in detail for many years. The systems are becoming common enough, and the programming interfaces capable enough, that many different research groups are beginning to explore how software defined radios can be exploited to improve performance or enhance user experience.

There are many underlying motivations behind the advancements in SDRs. One of the first, and most ambitious, goal was to solve a persistent problem for the military. Military forces operate in many different regions, each of which has regulatory oversight of spectrum allocation and different communication standards. Developing and deploying radios that could operate across the different ranges of spectrum and implement the different policies needed for deployment in different regions was a difficult task. It would be much easier to have a single radio into which software could be "poured" into the radio to have its behavior conform to the specific locale. Similar motivations exist today - for example, two competing standards for wide-area cellular technology, LTE and WiMAX, both use similar frequencies, waveforms and technologies – although the main difference between the technologies occurs above the physical layer, the physical interfaces are similar enough that it is compelling to develop a single radio platform that could handle both standards, including any future variants of the standards.

Standards are another motivation for the the mutable lower layers enabled by software radio. Standards take considerable time to be finalized – a software or mutable platform lets vendors develop and deploy products prior to a fixed standard, and then address any changes once that standard is finalized. Software radios also allow companies to innovate beyond the practice of current standards, and use experimental de-

ployments to assess the value of those extensions. It is also conceivable that a fully software radio might allow vendors to side-step standardization before deployment. For example, many Internet services and development models (**e.g.,** REST [9]) are tried out by practical deployments before broad acceptance. Due to the lack of programability, the barrier to innovation in wireless networks is much higher.

Although the concept of software defined radios has been around for a long time [10], the evolving technology in digital signal processing and architecture has made it more practical. There are four elements needed to enable software radio for high-bandwidth wireless communications: sufficient I/O throughput to transport data from the Analog to Digital (A/D) or Digital to Analog (D/A) devices and the CPU, sufficient digital signal processing throughput, a suitable software development environment and a flexible RF "front end".

3.1 Hardware

The A/D-D/A devices and the radio "front end" translate between the analog RF signals and the digital world of samples. There are many challenges to a purely software implementation of a modern high-speed wireless signal, similar to that of WiFi [11]. The radio interface generates or consumes $\approx$ 160MB/s of data – for a 20MHz waveform, 40 Msamples per second are needed, and each sample is typically two 16-bit values (representing the In-phase(I) and quadrature(Q) components); until the last two years, this is beyond the ability of most common I/O interfaces. Moreover, the computation needed exceeded the processor throughput of most general-purpose computers in 2007. The challenges for a "narrowband" wireless system are much lower, and Vanu [12] describes such a system.

Many current systems are implemented using **field programmable gate arrays**. For example, Amiri [13] describes a mesh network organization built using custom FPGA boards. Our own group has developed a modular transceiver design using FPGAs and two different design flows - either a traditional "monolithic" implementation [14] and a more modular "system on chip" organization [15]. This latter organization uses FPGAs to implement a "sea of components" that are interconnected by a dynamically routed on-chip network, allowing flexibility while producing a specific radio configuration.

A similar approach has been advocated by researchers [16] using the Intel Exoskeleton framework [17]

to augment an existing CPU with specific "signal processing components" interconnected by a routable network. The benefit, and limitation, of that approach is that the signal processing blocks are pre-configured, and although the existing blocks can be used in different configurations, blocks cannot be added or changed. This is a limitation because any new signal processing algorithm would need to use the existing blocks; it is a benefit because the chip density and power for such a custom design is typically a 10-fold improvement over FPGA designs. Current designs using FPGAs for signal processing are probably not realistic for low-cost handset designs because of power and circuit density.

An alternative step is to design a CPU that can handle different software radio tasks using distinct instructions optimized for those tasks. Woh et. al. [18] illustrates the challenges a basic architecture of SDR will face when we move to 4G networks. The authors considered the SODA architecture [19] as the basis of their enhancement. This architecture has SIMD support for parallel multiple operations on the same data. The major algorithms that a high-speed network like the WiMax, LTE, and 3GPP standards are comprised of are a) Fast Fourier Transform (FFT)/ Inverse Fast Fourier Transform(IFFT), b) Space Time Block Coding (STBC), c) Vertical BLAST (Bell Laboratories Layered Space-Time) or V-BLAST and d) Low Density Parity Check (LDPC). The FFT/IFFT routines are required for any multicarrier communication, to change the time domain signal to frequency domain and **vice versa**. Since more subcarriers are expected in higher data rate networks, the FFT size can go up to a width of 2048 or more. Space-time coding, STBC, is an encoder/decoder, which is used in MIMO transmission, for encoding two copies of the same data and transmitting it in two time slots. The encoding requires conjugate and negation calculation. Both the encoding and decoding process can be run in parallel and can be computed in a SIMD machine with FFT. V-BLAST is another encoder/decoder technique used for MIMO systems, where spatial multiplexing is obtained by transmitting independent data streams over multiple antennas. It also requires conjugating and negating a block of data. It is an iterative process at the receiver, and consumes computational cycles for implementation. 4G systems are expected to use LDPC codes and also Turbo codes, which are already used in 3G system. LDPC codes are channel encoders, meant for converting the data into a meaningful codeword before transmission to compensate for errors. LDPC also exhibits data level parallelism, which can be implemented in SIMD processor. The last design alternative used for some software radios is an array of small general

"tiled processors" connected by an on-chip network, such as the picoArray [20]. Such designs have been used to build an 802.16 base station.

Each of these design alternatives (general purpose CPU, reconfigurable FPGA, reconfigurable SOC, SIMD processor or tiled array) have advantages and disadvantages. A more fundamental question is **are these capabilities needed?** We believe the answer is "yes". The current trends [21, 22, 23, 24] in software radio innovation require extensive reconfigurability of the hardware and continuous communication from the MAC layer for decision making in signal processing. In legacy hardware, the system would process all the signals similarly and would generate a packet and hand it over to MAC, which then determines what to do with the received packet. At the transmitter side, carrier sensing [23] is done in some subset of subcarriers based on the decision from a statistical model maintained in the MAC layer. The physical layer reconfiguration is required depending on the current environment of the network and the received signal (whether it is a collision or not) and requires constant feedback from the MAC layer for signal processing, which cannot be pre-determined or predicted beforehand and programed in an ASIC. This creates a demand for a reconfigurable, higher layer controlled physical layer to act to the environment and to the received signal; this requires the capabilities of one of the platforms described above.

3.2 Software

Many on-going projects [21, 22] use the GNURadio [25] framework for signal processing at the receiver side. GNURadio provides a platform of extreme reconfigurability as all the signal processing is done in software code. The software stack is built on the USRP (Universal Software Radio Platform), from Ettus Research, connected to the computer with the USB port, but can be extended to other platforms. The radio front-end up to A/D or D/A conversion is done in hardware, and the rest of the signal processing blocks for both transmitter and receiver are in software. Since a desktop computer is used to compute the signal processing, the processing is serial and takes more time than real demand of the network.

More elaborate (and complete) software environments exist; the SCA (software communication architecture) is a complex framework involving CORBA-based services to configure the radio. An open source version of this software, OSSIE [26], is available, and demonstration of this system also uses the GNU

USRP and general purpose CPU's.

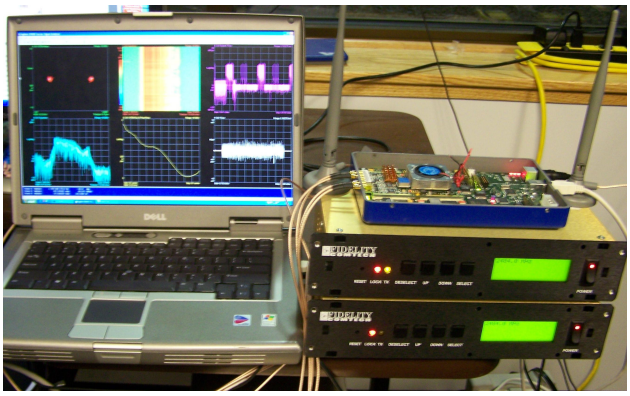
The GNURadio framework is mainly appropriate for "non-real time" work because it uses a general-purpose computer. Authors in [23] and [27] use their custom built flexible hardware platform, both of which are FPGA based. One example is the Wireless Open-Access Research Platform (WARP), built on Xilinx Virtex-II Pro FPGA board, where the MAC protocol is written in C and runs on PowerPC cores, and the PHY is implemented in FPGA.

This programming model, which combines C and hardware design languages like Verilog or VHDL, is very challenging for most computer science network researchers. Software, and particularly software accessible by standard networking researchers remains a major challenge for software radio.

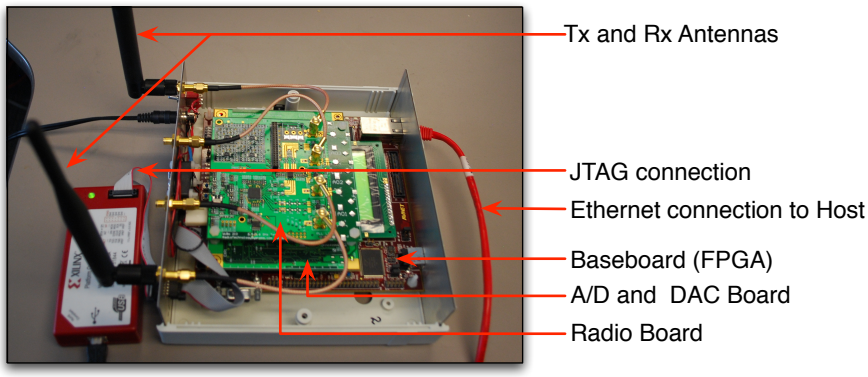
3.3 Reconfigurable Hardware as a Facilitator

All these research efforts constitute a substantial part of the current trends in wireless networks, which require significant communication between physical layer and upper layer protocols with easy reconfigurability of the physical layer hardware platform. The reconfigurable hardware utilized in this work are built on our previous work [11, 14, 28]. The baseband design of the transmitter is partly implemented in software and partly in hardware, whereas the receiver is implemented completely in hardware. All the configuration information for the PHY layer can be sent to the hardware using a packet format, which is decoded in the hardware to transmit or receive a packet.

We used two variations of hardware platforms for this thesis. Both the radios have the same baseband design, as described in our previous work [11, 14, 28]. Figure 3.1 shows the two platforms. Version 1 is the SDR developed on Xilinx ExtremeDSP development kit IV manufactured by Nallatech, with a custom front-end radio responsible for up/down conversion to/from the 2.4GHz ISM band, as shown in figure 3.1(a). The ExtremeDSP board includes either a Virtex IV or a Virtex II FPGA equipped with a PCI/USB interface and two sets of A/D and D/A converters. Gain control is also a part of the radio that can be controlled manually. Version 2 is the SDR, where Xilinx Virtex5-LX50 FPGA acts as the motherboard for the baseband OFDM design and ADC/DAC and RF daughter cards. These daughter cards are programmable from the host computer through Ethernet interface. Also, the Gigabit Ethernet is used to send control and data packets to



(a) Version 1

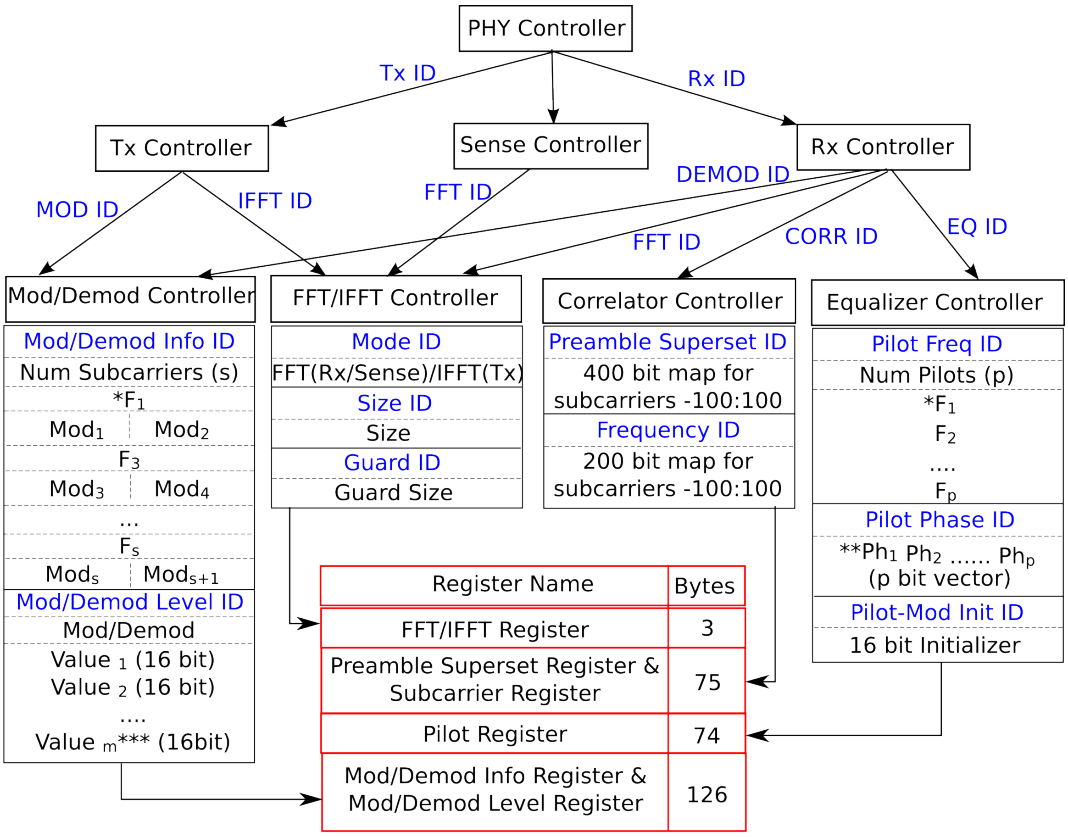


(b) Version 2

Figure 3.1: Software Defined Radio platforms used in this thesis

and from the radio board. The detailed design and modifications required in each work will be explained in corresponding chapters.

Figure 3.2 shows a hierarchical design of the control path for the OFDM based SDCR transceiver. The PHY Controller is the interface between the MAC or the Cognitive Engine and the underlying hardware. This controller helps to modify the physical layer parameters for the hardware.



** F_i = Subcarrier Index (-100:100).*
*** Ph_i = Phase of Pilot at subcarrier F_i. Values can be 0(Phase=0) or 1(Phase=π).*
**** The value of m depends on Modulation Type. m=1(BPSK), 2(QPSK), 4(16QAM) or 6(64QAM).*

Figure 3.2: PHY Controller

Chapter 4

PAMAC - PHY Aided MAC

Wireless networks are inherently contention based systems. A considerable amount of time is lost in contending for the medium, retransmissions, collision etc. Rodrig **et al** [29] have shown that only 40% of the transmission time is used for actual information transfer. Most of the reasons for this low utilization of the wireless medium involve exchange of co-ordination packets between the AP and the client. Substantial time is also spent in decoding these co-ordination or control packets. Evidently, there is a requirement to improve the signalling mechanism in wireless networks while it is equally imperative to reduce contention in the network.

In a time-division multiplexing network, an AP has the primary control of the media and assigns time slots for client transmissions. This method ensures contention free data transmission and also requires proper signalling and information exchange between the two parties, usually using some form of broadcast messages and the subsequent acknowledgements from the clients. The unreliability of the wireless medium has always made reliable broadcasting a challenge. Much work has already been done on improving the Distributed Co-ordination Function in 802.11 MAC to improve fairness and throughtput by modifying back-off algorithms or contention window [30, 31, 32].

In this research, we show that we can use the same multicarrier communication methods as used to implement high data-rate transmission mechanism to build very low-cost signaling methods that make time-division networking very practical. In this work we intend to go beyond the capabilities of a conventional software based MAC to a faster, reliable MAC design by including PHY layer functionality and use them for exchanging MAC layer information.

Multi-user communication requires some form of **orthogonal channel** for modulation that allows multiple parties to communicate simultaneously. Orthogonal Frequency Division Multiplexing (OFDM) is effective form of multicarrier modulation that forms the basis of the 802.11a/g PHY. OFDM is a mechanism that splits the available spectrum into a number of orthogonal non-interfering **subchannels**. Being orthogonal, each of the subcarriers can be treated as information carrying medium without significant interference with another subcarrier. Under OFDM, different nodes can also communicate on different **subcarriers** over the same medium and at same carrier frequency. The ability to be able to distinguish multiple simultaneously transmitted signals is a challenge in communication protocols. Therefore such signals typically need to be fairly simple.

For example, think of asking a room full of people if they ate breakfast that day. Individuals could respond using voice, but humans have a hard time distinguishing all the streams of information. However, if people raise their hands instead, it's immediately clear who has and has not eaten breakfast. However, it's hard to get complex information, such as **what** someone had for breakfast, since the set of possible responses are so large.

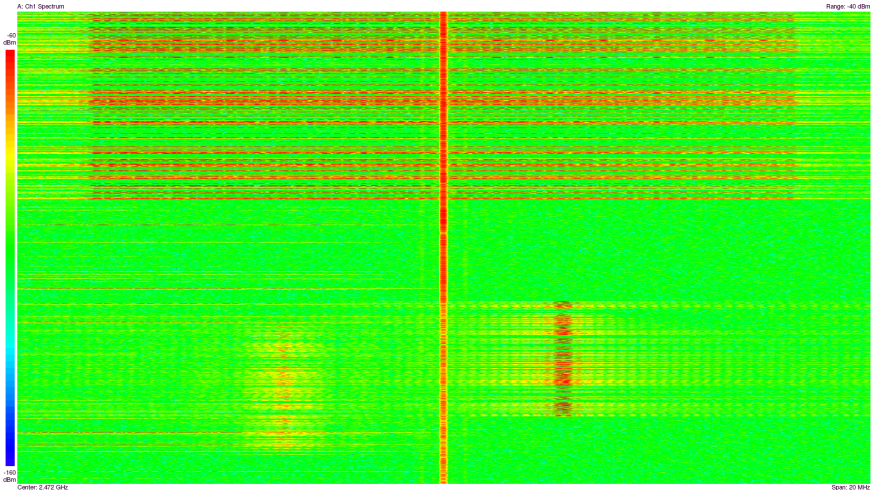


Figure 4.1: Waterfall Plot Using Three Prototype Radio Platforms

Figure 4.1 illustrates how radios can simultaneously "raise their hands" using orthogonal communication frequencies. This waterfall plot shows energy at different frequencies (horizontal axis) over time (vertical axis). The wideband energy at the top of the figure is a broadcast packet asking nodes to respond

if some condition is met; the two bands near the bottom of the plot are responses from two nodes. Later, we show that although the two nodes start transmitting at slightly different times and at different energy levels, it's easy to determine that two specific nodes have responded based on timing constraints.

Simultaneous transmissions, such as those shown in Figure 4.1 can be an advantage in a number of network applications that call for multiple nodes to participate and also use simple information, like route requests, leader election, network management and other operations involving broadcast or multicast messages. Not only does simultaneous transmissions make the message exchange faster, its also allows such exchanges to be reliable by rapidly singling acknowledgements.

This work examines how such **physical signaling** can be used to improve the basic performance of MAC protocol. With no interfering APs around, the protocol can provide contention-free high throughput to a large number of users. Thus, exploiting simultaneous multiuser communication will make the uplink and downlink scheduling faster as it eliminates signaling by packets. As a comparison to 802.11 networks we find that in order to schedule for $'n'$ users we need $'n'$ responses where as in our protocol using OFDM based PHY layer signalling the response is of the order of $\lceil n/53 \rceil$ if we are using 53 subcarriers per OFDM symbol. This is the primary focus of this work.

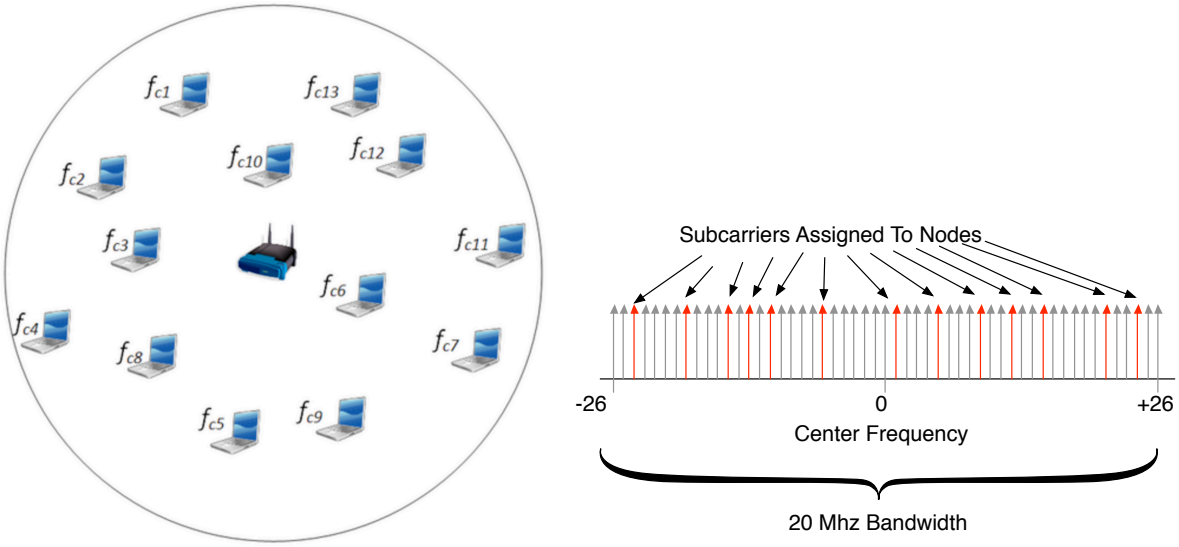
In this work we emphasize that with a smart PHY layer which supports non-contiguous OFDM transmissions we build a more efficient, fast and variable time-division multiplexing MAC layer. We first demonstrate the feasibility of using physical layer signalling to exchange MAC layer information between nodes, addressing problems of near-far effects and time synchronization and coordination. We then describe the MAC layer protocol that uses the simultaneous signaling made possible by the programmable radio, and the demonstrate its efficacy using and scalability using QualNet simulations.

The rest of the chapter is organized as follows. In §4.1 we show the practicality of physical layer signalling to build novel protocols. This followed by a MAC protocol design using this signalling method in §4.2. §4.3 describes the simulation infrastructure we used to evaluate the protocol under large scale conditions. Future work and conclusion is given in §4.4 and §4.5 respectively.

4.1 Demonstrating Implementation Feasibility

In this section, we first establish that it is both feasible and practical to use both physical layer signaling and simultaneous communication.

Later, in §4.2, we focus on building an efficient MAC protocol by speeding **group communication** using **simultaneous transmission and reception**. Some group communication are caused by network protocols or applications, such as broadcast or multicast packets. However, the bulk of group communication in many wireless networks is used for coordinating media access **via** a contention based access protocol. An alternative mechanism is a time-division access protocol, similar to the Point Co-ordination Function (PCF) in the 802.11 MAC. Even these time-division protocols require exchange of control packets and signaling between multiple nodes.



(a) Subcarrier assignment in a network

(b) Non-contiguous OFDM transmission

Figure 4.2: Schematic illustration of ACKs using OFDM

We will show that such overhead can be reduced significantly if we allow all the clients to communicate with the AP simultaneously. Figure 4.2(a) shows a typical infrastructure based network. Assume the AP assigns each of the clients a unique subcarrier index as shown in Figure 4.2(b) which will be used by the client to signal information. For this implementation we have chosen the OFDM based physical layer for

802.11a/g as the underlying signaling.

Figure 4.2(b) shows a schematic illustration of the properties of the OFDM waveform that are needed. A given bandwidth, such as the 2.4Ghz band used by 802.11g, is subdivided in a number of **subcarriers** around a center frequency; that center frequency is the "channel" to which an 802.11 radio is set. Some subcarriers are lost to guard bands (to prevent interference with adjacent channels) and some are lost for other purposes such as pilot tones, used to improve reception. In 802.11g, 53 subcarriers remain for data modulation. Normally, a single transmitter modulates all subcarriers to send high bandwidth data; we still use that scheme when transmitting data, but when stations signal the AP, they use simultaneous transmission.

4.1.1 Encoding The Signals

Clearly, stations can use their individual OFDM subcarriers to transmit a single bit of information, such as "I have packets to send". However, it's also possible to send multiple bits of information. For example, to implement a time-division or polling access protocol, an access point might need the clients to indicate:

(1) Who has packets to send?

(2) Approximately how many packets do you have to send?

Knowing the approximate queue length might let the AP implement various "fairness" methods by assigning different time slots to different clients based on their queue length.

In such a scheme, the clients would all receive a single broadcast packet that effectively poses the question above. The clients might respond with one of four states (EMPTY, LOW, MEDIUM and HIGH queue). We encode these four answers by sending a BPSK modulated "1" or a "0" in their assigned subcarriers spread over four subsequent OFDM symbols. Thus the encoding in Table 4.1 can be done to signal any of the four cases,

Since sending a one is essentially sending a single tone with frequency dependent on the subcarrier index, the receiver can detect the presence of a tone by computing the Fourier transform of the composite

Table 4.1: Signaling Scheme for AP

State	1^{st} **&** 2^{nd} **symbol**	3^{rd} **&** 4^{th} **symbol**
NO packets to send	0	0
YES, LOW Priority	0	1
YES, MEDIUM Priority	1	0
YES, HIGH Priority	1	1

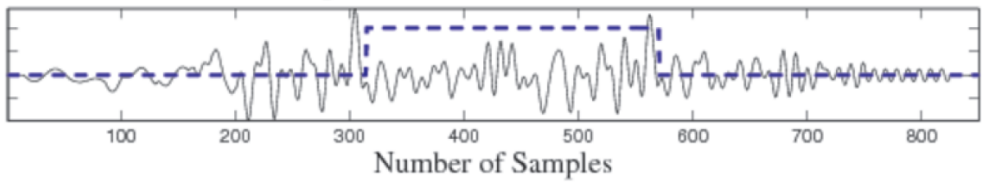
received signal. The only challenge is that different users will have different transmission times, and subcarriers at the receiver will have different times of arrival. Thus, selecting a suitable FFT window is key to detecting significant power in the subcarriers.

4.1.2 Detecting The Signals

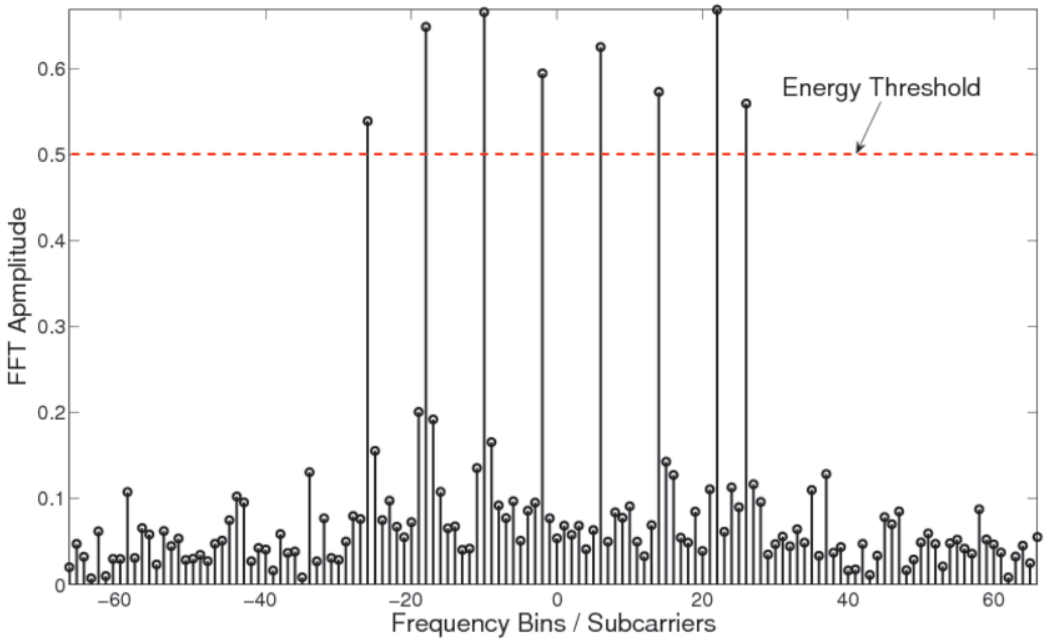
Figure 4.3(a) shows a composite waveform consisting of tones of different frequencies. The blue dotted line marks the optimum FFT window at the receiver. Figure 4.3(b) shows the magnitude of the Fourier transform of the composite waveform, revealing that 8 clients have actually transmitted tones, while other subcarriers remain idle. Thus, a suitable threshold is required to detect energy on the individual subcarriers. Variation in signal energy will result in relative distances of the clients from the AP and attenuation due to frequency selective fading and multipath effect. It's easy to selection a threshold below which packets could not be received. That means that if the energy on a subcarrier is below the threshold, that client is too far away or the channel induces sufficient fading to render normal data transmission (**e.g.** 802.11 data packets) impossible. Therefore chances of incorrectly rejecting a (meaningful) valid signal is very small.

Selecting the FFT window is key to successfully detecting the energy of each tone from the clients. Due to near-far effects and the different processing power of the clients tones from different nodes that will reach AP at different time. In a typical infrastructure network we assume that distance from the AP to the farthest node is $\approx 300m$, which implies a round trip delay of about $2\mu s$. There may also be delay from the transceiver turnaround time and hardware transfer times. Therefore we can define T as

$$T \geq 2 \times T_{propagation} + T_{r_x latency} + T_{hardware} + T_{t_x latency}.$$



(a) Composite Waveform



(b) FFT Magnitude

Figure 4.3: Fourier Transform of the Composite Waveform

Given that each OFDM symbol has a duration of $4\mu s$, we need to specify the interval T that the AP needs to wait before performing the FFT while ensuring that within that interval all nodes have transmitted their individual tones to signal about their queue status.

Figure. 4.4 shows the relative timing diagram and optimum FFT windows. Given a RTT of $2\mu s$ from the farthest node we start the FFT window after $3\mu s$. This gives us a flexibility of the $1\mu s$ which might occur due to any unforeseen circumstances. The **"brown bar"** marks the optimum FFT window which is $4\mu s$. A key point here is that the FFT window doesn't have start after exactly $3\mu s$. Since tines are coming from different users all we need to ensure that all the tones are present for the entire duration of the FFT window

which is denoted by the interval **"Flexible FFT Window"**. Since the near-far effect will also affect the next two symbol transmission a gap of $4\mu s$ is required before performing the next FFT. Similar to the first FFT window we have an error margin of $1\mu s$ before and after the optimum window. The combined information from these two FFT window provide the AP with all the information it needs to receive two bits of data, as described above. The method can be extended to allow clients to transmit an arbitrary number of bits.

The precision of clock synchronization needed for this method is actually less than for normal 802.11g data payloads. Periodically, the AP synchronizes the clocks on the clients to maintain the $1\mu s$ clock accuracy needed in the protocol. Unlike single user OFDM transmission, strict receiver timing synchronization is not required since no demodulation is required – we are simply detecting "energy in the channel". Also, since these are unique single frequency tones, the OFDM subcarrier are transmitted without any PLCP header or any identifier which saves bandwidth and makes detection faster at the AP. This makes implementation fairly simple and straightforward, and the technique should be able to be implemented on commodity 802.11 hardware.

To understand how much more efficient it is to use physical layer signaling, consider the costs of transmitting a message using the 802.11g PHY that's the basis for our extension. A normal message requires a 20μs preamble to be transmitted and then, at best assuming the 54Mb/s modulation rate, each 48×6 bits takes one OFDM symbol time (4μs) to transmit. Thus, a 64 byte message, which can't actually even contain the Ethernet addresses in a standard 802.11g packet would take at least $20 + 4 \times 3$ or 32μseconds. After a 2μsecond "SIFS" period, clients would normally respond using a similar message format. Thus, a response to a standard 802.11g packet would take another $\approx 32\mu$s. By comparison, using physical layer signaling **all 53** clients can provide two bits of information (such as queue length) within four OFDM symbol periods, or a total of 16μs, or only half the time for a **single** station to respond using standard messages.

4.1.3 Hardware Implementation

To demonstrate that the challenges to using simultaneous reception to implement our protocol are indeed tractable, we implemented a prototype using a software defined radio platform. The basic design involves an OFDM transceiver on a Virtex-IV FPGA along with a custom front-end radio [33]. The plat-

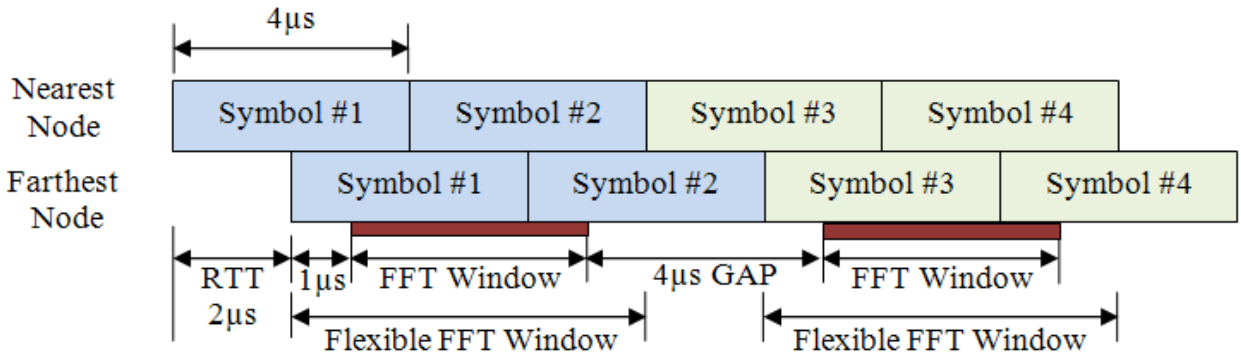


Figure 4.4: Signal Timing Diagram

form is capable of transmitting and receiving generic 802.11g frames as given in the 802.11 physical layer specification [34].

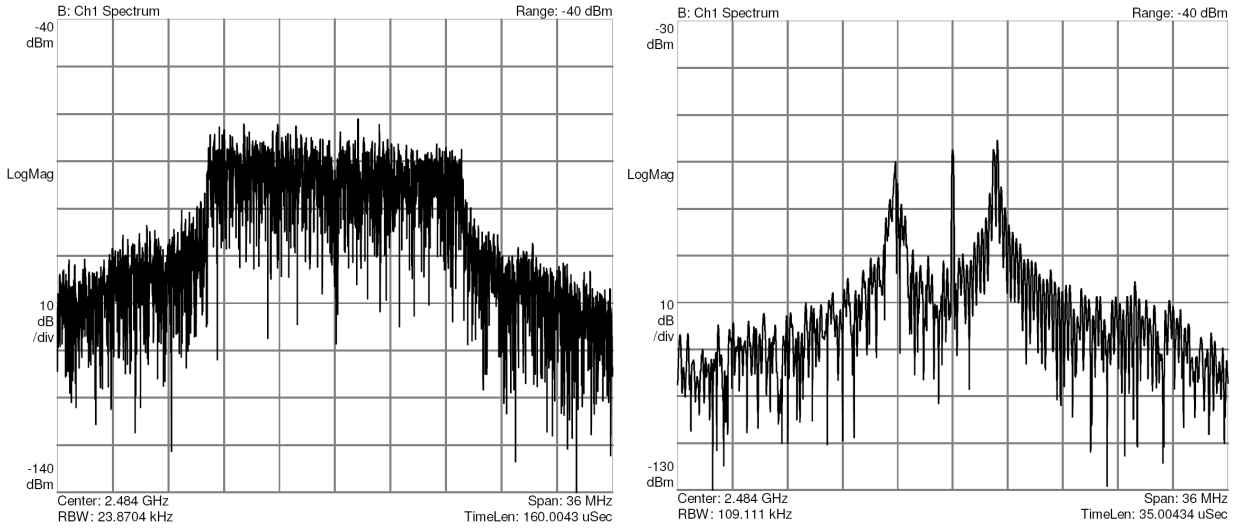
Implementing the protocol described in §4.1 requires transmission of **non-contiguous OFDM symbols**, where none but one of the subcarriers is used to transmit the information. This requires some changes in the transmitter design. The transmitter design has been detailed in [11] which employs a hybrid design allowing sufficient reconfigurability to perform such non-contiguous transmissions. The protocol requires the involvement of a reconfigurable transceiver as well as a MAC layer that controls the hardware to perform the required tasks.

The AP (implemented using our hardware) receives a composite additive signal from all the clients and, depending upon the number of users, the number of distinct frequency components in the signal will vary. Observing the magnitude of the Fourier transform we can detect high energy subcarriers and thus identify the corresponding client.

Given that you can demonstrate almost anything using a Matlab simulation, we felt it was important to demonstrate the protocol using three prototype hardware nodes. One of the radios was used to transmit one broadcast packet using the standard 802.11a/g PHY specification. The receivers decoded the broadcast packet and prepare the ACK packet with information on their pre-assigned subcarrier and transmit. The receivers were placed at two widely-varying distances from the transmitter to highlight the impact of near-far differences in clients. We used a vector-signal analyzer to capture the physical data.

Figure 4.5(b) shows the spectrum for the broadcast packet, which uses all the subcarriers for trans-

mitting broadcast information, and Figure 4.5(a) is the spectrum for the composite response signal from the two client nodes. Node 1 is transmitting in subcarrier $+8$ and node 2 is transmitting the ACK using -8. These same signals were shown earlier in the waterfall plot in Figure 4.1. Figure 4.1 showed that the node transmitting in subcarrier $+8$ has a higher signal power (closer to the AP) compared to the one transmitting using subcarrier -8 (further from AP). Also the different time of arrival of the ACKs show the near-far placement of nodes and subsequent attenuation. The waterfall plot also shows the broadcast packet at the top of the graph – that packet is transmitted first using the full spectrum available.



(a) Spectrum of Broadcast

(b) Spectrum of ACKs

Figure 4.5: Spectrum of the Broadcast Packet and Response Using Three Radios

4.2 Efficient MAC Protocol Using PHY Signaling

Much of the overhead of wireless networks arises from the network signaling for media access. One example is the 802.11 distributed coordination function (DCF) protocol, which has been reported to have up to 60% overhead due to the media access overheads and retransmissions due to errors [29].

In this section, we describe Physical Assisted MAC (PAMAC), a MAC protocol that is compatible with the 802.11 DCF phase, is modelled after the PCF (point coordination facility) protocol but uses physical layer signaling to further reduce signaling overhead. In §4.3, we compare the performance of PAMAC to

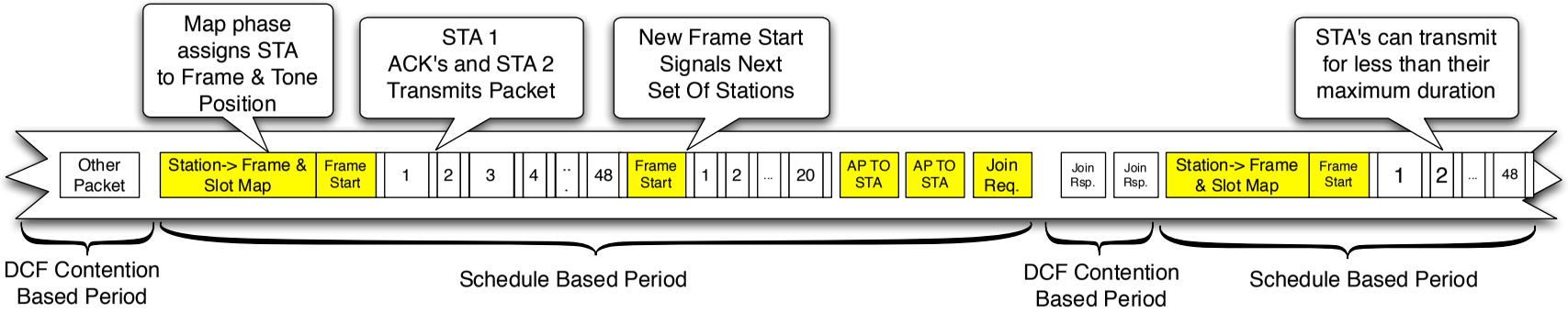


Figure 4.6: Schematic Time Series Showing Protocol Operation - Darkened packets indicate packets sent by the access point and all non-filled packets are send by different stations.

a conventional 802.11 DCF MAC; we were unable to compare the performance of PAMAC to the 802.11 PCF MAC due to limitations in our simulator.

The full description of PAMAC would entail state transition charts similar to those used to describe 802.11 DCF operations. Rather than use such charts, Figure 4.6 shows a sample time sequence for PAMAC operation. The time line shows messages transmitted over time from left to right; darkened messages are transmitted by the AP and all other messages arise from stations. As described in §5.1.1, an access point using PAMAC must assign stations to specific "tones" or subcarriers used for physical signaling. Thus, the initial operation of the network would involve an advertisement for stations to join the schedule-based phase of the network operation; such a message may be occasionally repeated to allow nodes the choice to join or leave the scheduled phase. The "Join Request" message elicits "Join Response" messages from stations. Once a pre-specified join period has expired, the AP will start a schedule based operation period.

The AP will first transmit a "traffic map" frame that assigns stations to specific "frames" and "tones" within those frames. In a small network, there would usually be a single frame; additional frames are used if the number of stations exceeds the number of subcarriers available for simultaneous signaling (**e.g.** 53 subcarriers for 802.11g). The "traffic map" frame is only sent when stations leave or enter the scheduled network operation.

Prior to sending "uplink" traffic from the stations to the AP, the AP sends a "frame start" message. Following the frame start, the stations reply with a tone sequence indicating if they have any messages to upload; all stations assigned a subcarrier or tone for that frame with a packet to transmit would respond simultaneously, as described in §5.1.1. The AP then transmits a single tone subcarrier indicating which station should transmit. Stations transmit for a fixed duration, possibly sending multiple messages during that period. If a station finishes before the end of the fixed duration, it can transmit a single tone on its subcarrier to indicate that it has finished transmitting; the AP will then indicate the subcarrier or tone ID of the next station that will be allowed to transmit. There are no hidden terminal problems since all transmitter wait to transmit until they are told to start, and the AP is in charge of designating which station should transmit.

When stations transmit to the AP, they use the standard 802.11g PHY layer encoding, including the

preamble sequence for clock recovery, PCLP header and standard MAC header. It would be possible to reduce the size of the MAC header since the AP knows which station is transmitting, but we did not model this in our simulations. The station **must** include the overhead of the timing recovery preamble because that information is needed to synchronize the clock phases to allow high rate modulations to be decoded.

Following the "uplink" phase, the AP transmits packets to stations using standard 802.11 packet encodings (**e.g.** also prepends a preamble, PLCP, **etc**), but transmits them in a continuous stream where stations ACK the packet and the AP then transmits the next downlink packet without releasing the media by starting transmission prior to the end of the SIFS interval.

The ordering of uplink and downlink phases can be flexible; since the AP is in charge of scheduling transmitters, it can allow as many or as few to transmit packets as desired. It can also indicate that a station can use multiple transmission slots by repeatedly indicating that the station should transmit. Likewise, the AP can conduct multiple "uplink" phases followed by a single "downlink" phase or intersperse the individual phases. Clearly, different orderings would impact packet delay, fairness and other MAC characteristics.

In the implementation we use for our protocol comparison and evaluation, the AP allows all up-link traffic to be transmitted and then transmits all queued downlink traffic. This may slightly increase packet jitter or delay, but we have not studied the impact of varying that policy.

4.3 Result And Analysis Of Simulation Study

To evaluate the performance of the proposed protocol, we implemented an OFDMA based transceiver in QualNet [35], operating at $2.4GHz$, largely matching the capabilities and characteristics of our hardware platform. We compared the performance of the proposed protocol with the conventional IEEE 802.11a based MAC protocol provided by QualNet. We assume that the AP is in the middle of the scenario and all the clients are randomly distributed within a radius of $150m$. Thus, the AP is within the transmission zone of all clients, but all clients are not within the transmission zone of each other. Many similar random scenarios were used; a later example will illustrate the layout.

Our protocol is referred to as 'PAMAC', while the IEEE 802.11a based MAC protocol is refereed to as '802.11' throughout rest of the chapter. Table 4.2 shows the parameters used for simulation.

Table 4.2: General Simulation Parameters

Seeds	10
Packet Size (VoIP)	$120bytes$ (G.711 codec) $10bytes$ (G.729A)
Packet Arrival Interval	$15ms$ (G.711 codec) $10ms$ (G.729A)
Physical Layer Data Rate	$36Mbps$
Simulation Time	$120secs$
Pathloss Model	Two-Ray
Application Layer	CBR
Transport Layer	UDP
Mobility	None

We evaluated our protocol for VoIP application, which requires low but constant bit rate for efficient quality of voice service. Since VoIP packets tend to be fairly small, this workload is representative of workloads that incur considerable signaling overhead; it is also an increasingly important protocol as the number of cellular phones using 802.11 to improve quality increases. We used G.711 codecs, which generate $120bytes$ of data packet at the application layer, at a constant interval of $15ms$. All VoIP calls were full duplex sessions between a client and the AP. No client initiated multiple sessions.

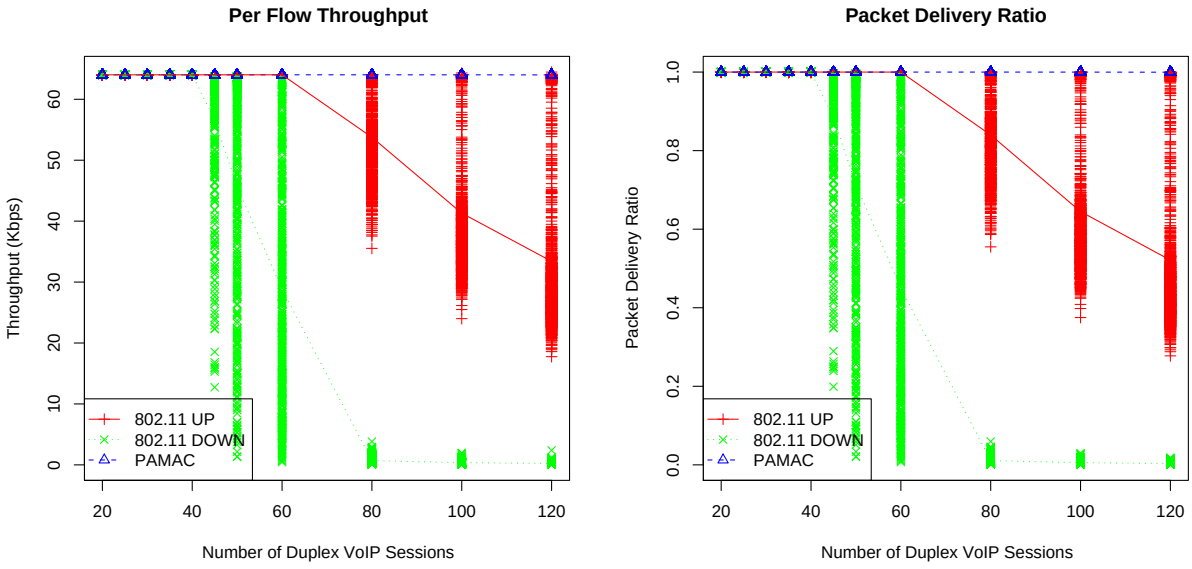
Figure 4.7 shows the performance of 'PAMAC' compared to standard '802.11', with increasing number of VoIP Sessions. PAMAC successfully caters efficient service to 120 clients, with almost no packet loss. The average end-to-end delay is significantly low, less than $20ms$, and the jitter in delay also remains low even at 120 duplex sessions. The end-to-end delays of the 802.11 protocol are much higher for larger numbers of stations due to queue overflow and we do not show that in the graph. We are interested to know that our protocol does fairly efficient communication even with 120 duplex sessions and call quality is maintained. However, the standard 802.11 MAC gets saturated beyond 40 concurrent sessions. The uplink and downlink flows in 802.11 show distinctively different behavior. As network gets saturated, the AP builds a large queue of packets to send to the stations. Since the AP is using DCF, it does not get enough access to the medium depending on the cumulative traffic that is accumulated in its queue for all the clients. Hence, downlink sessions suffer more than the uplink ones. Other MAC protocols, such as Idle Sense [36] can resolve such unfairness, but they don't remove the overhead of contention.

For reliable communication, with more nodes participating in the network, we studied the effect of using RTS/CTS, but we do not include those results because they largely duplicate the results when **not** using RTS/CTS. We observed that the overall performance of '802.11' deteriorates due to the use of RTS/CTS, and the network gets saturated even with 40 sessions, which is due to the fact of control packets consuming the bandwidth rather than data transmissions. The PAMAC protocol does not need to use RTS/CTS for uplink or downlink traffic; stations only transmit when they are told to do so, and thus hidden terminal issues can not arise. Likewise, when the AP sends downlink traffic, all stations overhear the traffic and thus do not transmit.

We also evaluated our protocol with another VOIP codec,G.729A, which has a lower bandwidth requirement. Figure 4.8 shows that when using this codec, PAMAC outperforms '802.11' also with very low number of VoIP sessions. With 45 clients, uplink sessions maintain QoS, while the downlink sessions suffer with the 802.11 contention based method. With $36Mbps$ physical layer bandwidth, 802.11 can only handle 30 duplex sessions, which is $(16Kbps \times 30) = 480Kbps$ of application layer data. Again, this limited use of the wireless spectrum is due to the poor coordination of the wireless medium; increased signaling can improve the media utilization and our physical layer signaling makes such signaling very inexpensive.

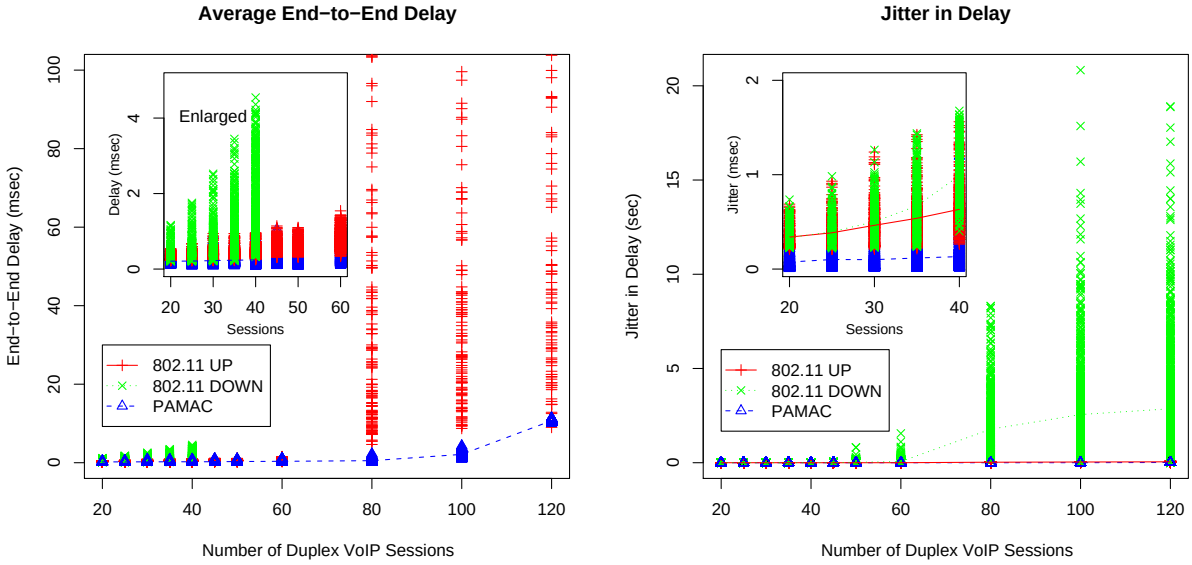
We also studied whether the variation in throughput of 802.11 clients is due to any spatial distribution of the clients. Figure 4.9 shows the topology of an example scenario, where the AP is located in the middle of the network, and the clients are distributed around it. The radius of circles representing the stations are drawn proportional to the throughput of the client, which indicates that the throughput has no clear relation to the spatial distribution of clients. The performance depends on the chance that a node gets to access the medium and transmits packets. In contention based protocols, carrier sensing and back-off leads to poor and unfair access to the medium. The PAMAC method allows both controlled media access and low overhead to achieve that controlled access.

Figure 4.10 shows how PMAC improves bandwidth utilization with compared to 802.11. The plot is a snapshot in time of $1msec$ duration. Node 1 has been designated as the AP and the other nodes are stations in the network. The **'red'** colored instances are either tones from the clients to the AP saying that they have packets to send or they are tones sent by AP to signal the clients to send data. Every signal tone



(a) Throughput

(b) PDR

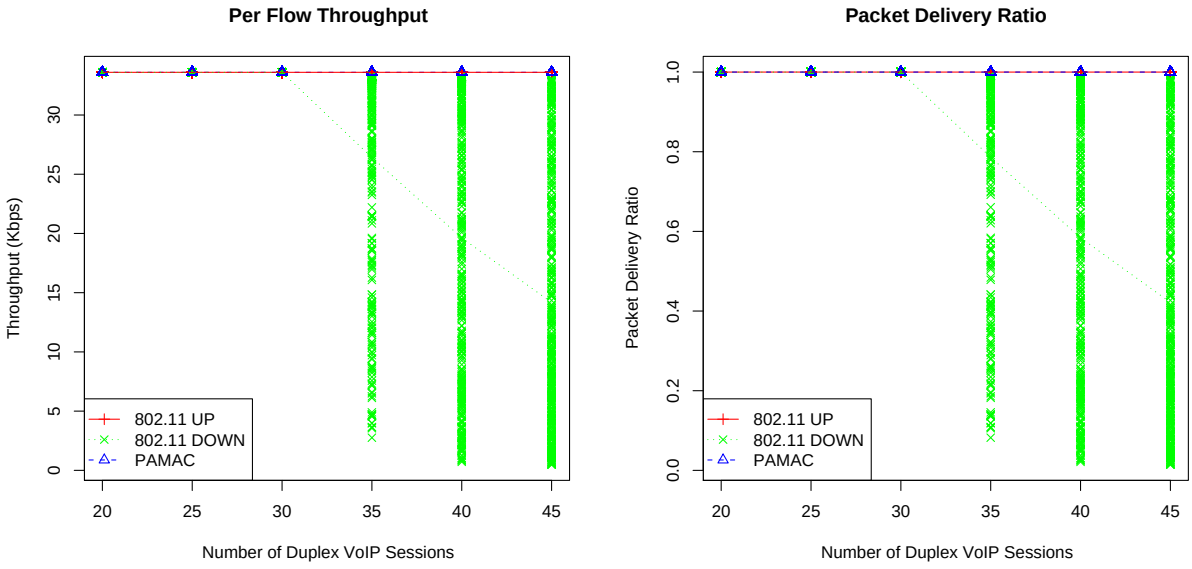


(c) Delay

(d) Jitter

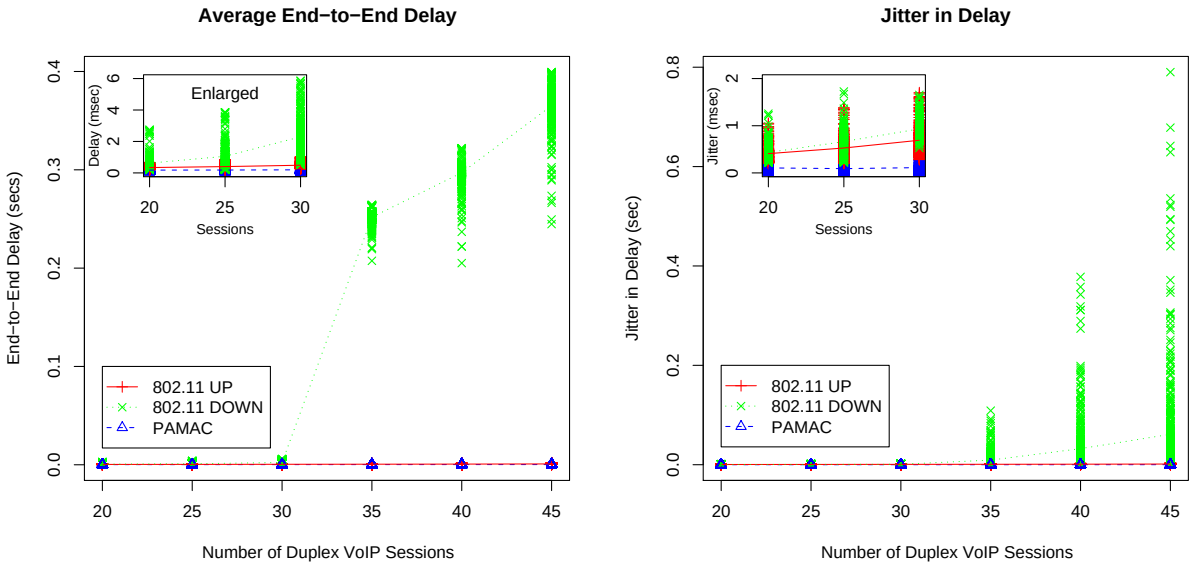
Figure 4.7: Protocol Performance - G.711

from the AP is followed by a packet transmission by the client denoted by **blue** timelines. Also, it is evident from the plot that all the tone signals from the client happen simultaneously as defined in the protocol. The **green** colored line denote the broadcast packets from the AP asking the clients about their queue position.



(a) Throughput

(b) PDR



(c) Delay

(d) Jitter

Figure 4.8: Protocol Performance - G.729

On the contrary in 802.11 case, the shorter duration **green** signals are either ACKs or RTS/CTS. After contending for the medium data packets from the clients are transmitted, shown in **blue**. The irregular arrangement of the arrows across time shows the contention period and significantly reduced utilization of

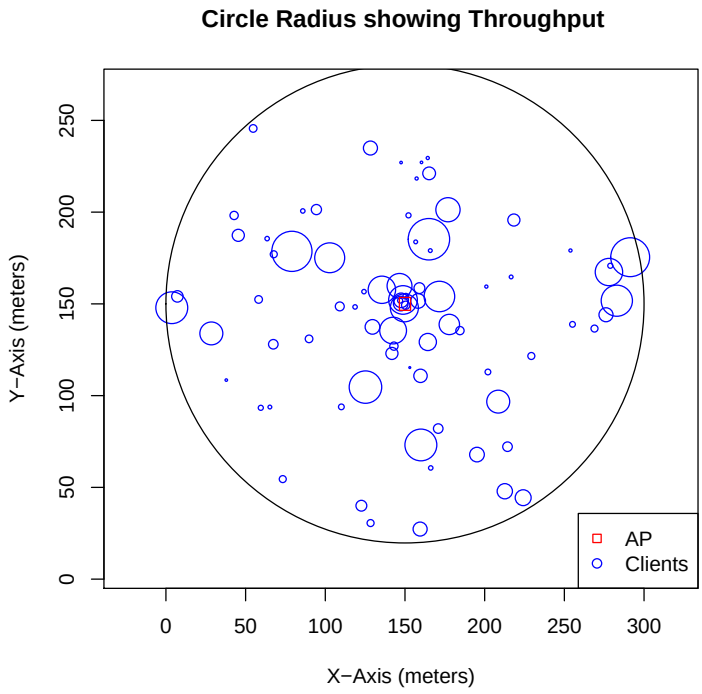
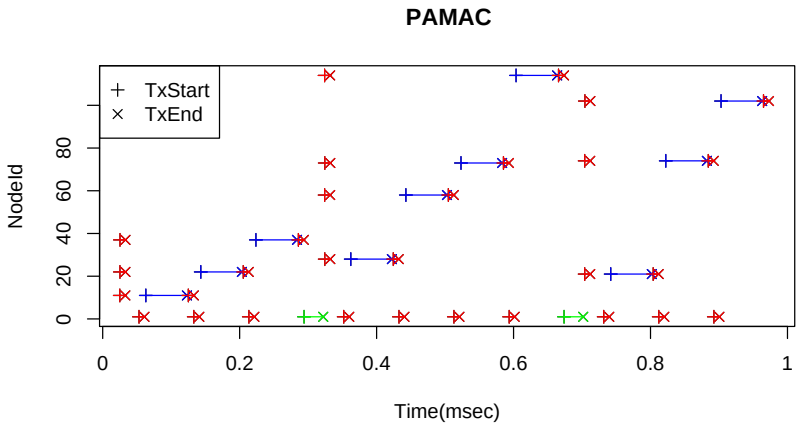


Figure 4.9: Throughput of Clients

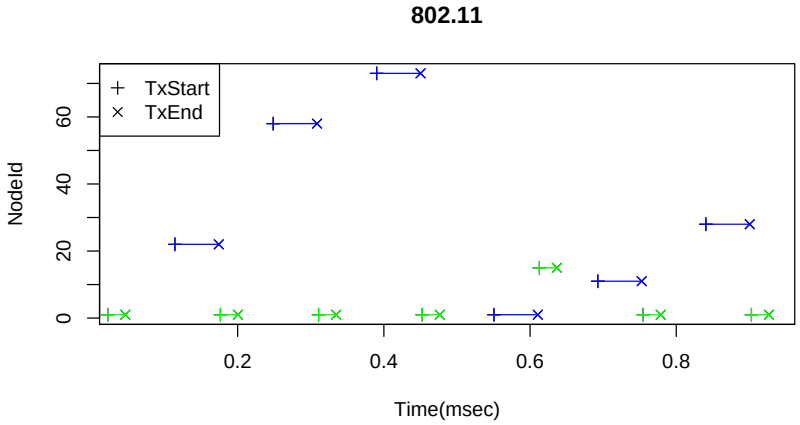
the wireless medium.

4.4 Future Work

The protocol we describe may present challenges in co-existing with already deployed CSMA/CA based 802.11 networks, but there are many scenarios where it is not required to co-exist with 802.11 networks, **e.g.** - a large conference room with more than 100 users or an enterprise wireless network deployment deployment that assigns neighboring APs to non-overlapping channels. We believe that, like other extensions to common protocols such as 802.11n, it's possible for the protocol to sense when non-compliant APs and stations may interfere with the signaling and revert to traditional packet based signaling; however, we have not verified this by simulation. This type of signaling mechanism can be used to simultaneously acknowledge broadcast and multicast packets, thus reducing overhead by making the process parallel. Also, this idea of smart signaling can be extrpolated to higher layer protocols like TCP ACKs that requires efficient cross-layer design and timing constraints which needs to be explored.



(a) PAMAC



(b) 802.11

Figure 4.10: Bandwidth Utilization with Time

4.5 Conclusion

We've shown that by using, rather than fighting against, the properties of the wireless physical media, we can develop robust signaling primitives that are both practical and allow innovative algorithms. We used a signalling method (OFDM) that is easy to understand and visualize, but the general technique is amenable to other methods of orthogonal signaling, such as CDMA or combined methods such as coded OFDM.

The critical insight is that we can combine the results from multiple clients using simultanous recep-

tion in an efficient manner. We can use this mechanism to both make specific network functions, such as broadcasts, reliable, but can also use the primitives to implement higher level group communication and signaling protocols. As long as the queries require simple "yes/no" answers, there are a number of robust mechanisms to combine the signals.

The question remains of how such functionality could be exposed to client and host operating systems, particularly since similar techniques are difficult to implement on non-broadcast networks (**i.e.** most traditional networks).

Chapter 5

SMACK - A SMart ACKnowledgment Scheme for Broadcast Messages in Wireless Networks

Noise and interference are fundamental aspects of communications, and are exceptionally important for wireless communications because it's more difficult to contain propagation without waveguides such as wires and fibers. Avoiding interference or noise is a fundamental design objective that limits the scope of simultaneous multi-user communication. Conventional single carrier communication focuses on decoding the strongest signal while discarding anything else as noise or interference.

Multi-user communication requires some form of **orthogonal channel** for modulation that allows multiple parties to communicate simultaneously. There are a number of ways to implement orthogonal channels - code division multiple-access (CDMA) has been adopted as a very reliable multiple access techniques by using specially designed **codes** with strong cross-correlation properties. With spatial frequency reuse, frequencies are allocated in a way such that signals from far away communicating pairs will be so strongly attenuated that they won't interfere in local communication. Time division multiplexing, or taking turns using a channel, is another method.

In this research, we focus on using **orthogonal frequency division modulation** (OFDM) to provide distinct orthogonal signals. OFDM is a mechanism that splits the available spectrum into a number of orthogonal non-interfering **subchannels**. Being orthogonal, each of the subcarriers can be treated as an information carrying medium without significant interference with another subcarrier. Variants of the OFDM waveform are used in a number of current wireless (and wired) physical layers, including the $802.11a/g$. Under OFDM, different nodes can also communicate on different **subcarriers**, as used in WiMax, which

employs "scalable OFDMA" where users use different subcarriers or set of subcarriers to transmit data over the same medium and at the same carrier frequency.

The ability to distinguish multiple simultaneous transmissions requires either the signal structure to be fairly simple or the decoding/detection mechanism to be complex. In this work we focus on a set of network primitives that calls for a very simple answer typically in binary; in the form of **yes or no**. Empowered by subcarrier transmission using OFDM we can either transmit a 1 or a 0 to convey these binary answers. Not only is this form of signaling simple, the detection of such a multiuser communication can be accomplished using spectrum sensing and energy detection. Simultaneous transmissions can be an advantage in a number of network applications that call for multiple nodes to participate and also use simple information. Examples include route requests, leader election, network management and other operations involving broadcast or multicast messages. Not only does simultaneous transmission make the message exchange faster, it also allows such exchanges to be reliable.

To demonstrate that the complexity in implementing this form of multiuser communication is indeed tractable, we implemented the protocol in a prototype hardware platform. Using FPGA based Software Defined Radios (SDR) we demonstrate the ability to detect multiple tone transmissions using Fourier transform and energy detection. The contributions of this research work are:

- We describe the practical constraints on using simultaneous communication for a wireless mesh network.
- We describe how simultaneous reception can be used to greatly improve protocol performance.
- We demonstrate the practicality of the system using a Software Defined Radio implementation of our protocol.

The rest of the chapter is organized as follows. §5.1 explains the protocol functionality and its efficiency and §5.2 describes the robustness of the protocol. In §5.3 we present the challenges and issues involved in implementing the protocol using SDRs. This is followed by the actual hardware implementation and design aspects in §5.4. To evaluate the hardware and the protocol performance we present a set of experiments in §5.5. The results from the experiments have been analyzed in §5.6. To demonstrate the usefulness

of this physical layer protocol to higher layer protocols we present a few applications in §5.8. Prior work related to this work has been investigated in §5.9. Finally we summarize this chapter in §5.9.1.

5.1 Smart Acknowledgments

In this work, we focus on speeding **group communication** using **simultaneous transmission and reception**. There are many types of group communications, the most common of which is broadcast or multicast. Conventional infrastructure wireless networks (**e.g.**, a standard WiFi network) usually only use broadcast packets to translate wired broadcasts into wireless packets. The standard 802.11 physical layer doesn't provide a method for determining if a broadcast was delivered; thus such broadcasts are typically transmitted at the lowest modulation rate (in an effort to increase the reliability of reception). Since broadcast messages are exchanged without acknowledgment control frames, there is a limited scope for the source or the access point (AP) to reliably ensure the reception of the message at the host nodes.

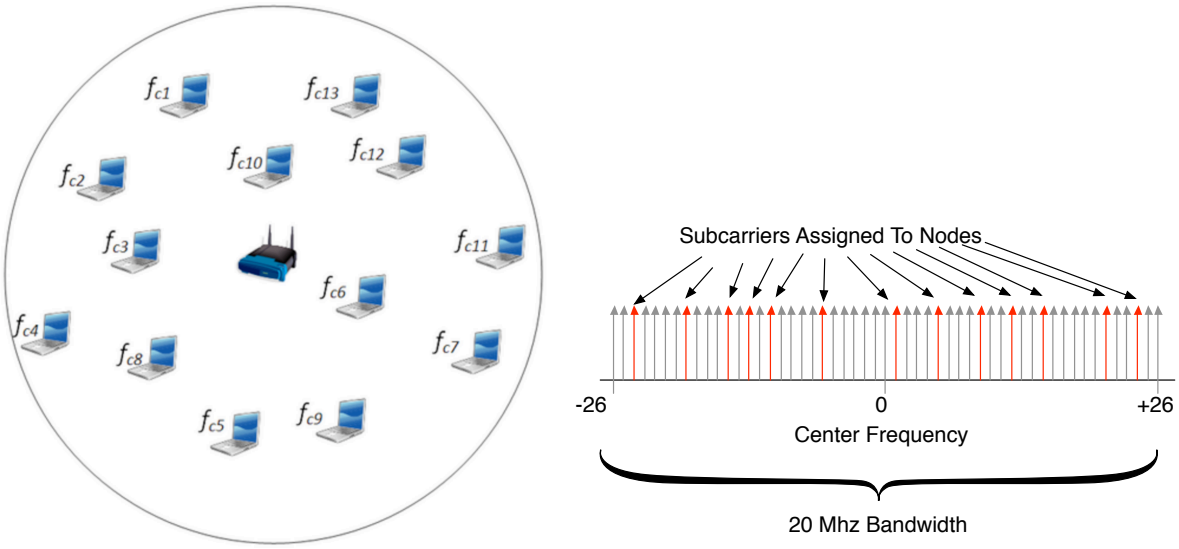
In "ad hoc" networks, broadcast messages are used for many purposes. Typical applications include host discovery, network maintenance, route discovery, etc. For example, wireless protocols such as AODV [37] periodically broadcast a routing table to "neighboring nodes" (meaning those that can hear the message). Nodes also periodically transmit "hello" messages to determine if nodes are still reachable. These messages are typically "unicast" messages, because there is no way to safely determine if they've been received.

Reliable broadcast messages, "hello" link maintenance messages and many other communications share a common pattern: a message is sent and one or more nodes should "vote" on the transmitted message. For reliable broadcasts, the vote is an acknowledgment that "I have received and can decode the message". If a node has not received the message, the sender would retransmit it. Link maintenance messages are almost identical, except that if a formerly "adjacent" node does not receive the message, it is removed from the node neighbors table (with no retransmission). Many other protocols, such as voting protocols, can map to a similar query followed by a yes/no decision from other nodes.

Some of these protocols concerning a single network "link" have an analogous extension to a "network" counterpart. For example, there is considerable work on providing reliable network-wide support for

broadcast packets in wireless networks, as well as distributed leader election.

5.1.1 SMACK - Reliable Link Layer Broadcasts



(a) Subcarrier assignment in a network

(b) Non-contiguous OFDM transmission

Figure 5.1: Schematic illustration of ACKs using OFDM

For any reliable broadcast mechanism to be reliable, there must be a clearly defined set of nodes in the network; Figure 5.1(a) shows a single access point and multiple clients. Each client is assigned a unique "membership number". For our implementation we have chosen the OFDM based physical layer for $802.11a/g$ as the underlying signaling method. Figure 5.1(b) shows a schematic illustration of the properties of the OFDM waveform that are needed. A given bandwidth, such as the 2.4Ghz band used by 802.11g, is subdivided into a number of **subcarriers** around a center frequency; that center frequency is the "channel" to which an 802.11 radio is set.

In 802.11g, 53 subcarriers remain for data modulation. Normally, a single transmitter modulates all subcarriers to send high bandwidth data. In our protocol, since we only need to transmit a "yes" or "no", we assign subcarriers to individual nodes, as illustrated in Figure 5.1(b); different clients are assigned subcarrier bins labeled as $f_{c1}, f_{c2}, \ldots, f_{cn}$ where n depends on the number of users and the number of subcarriers available. The orthogonality of individual subcarriers allows us to use each of them as separate

data carriers for different hosts. Using multicarrier modulation techniques allows the AP to receive ACKs from a greater number of clients in the shortest possible time, dramatically reducing the time to gather reliable acknowledgments for broadcasts. We use the physical layer to combine the responses from the different nodes. Upon receiving a successful broadcast message from the AP the clients use their pre-defined subcarriers to transmit a $'1'$ as an ACK.

To summarize, the protocol has the following steps:

(1) When nodes join the network, the AP assigns each node a unique "membership id", which is a small integer.

(2) An AP sends the broadcast message using conventional PHY specifications for $802.11a/g$.

(3) On receiving the broadcast message all clients decode the message (if possible).

(4) If a client successfully decodes the message, the client then uses the single orthogonal subcarrier specified by the membership identifier to indicate it has received and decoded the message

(5) The AP receives the composite time domain signal of **all** OFDM subcarriers and performs an FFT to obtain the frequency domain representation of the signal. After performing demodulation the individual acknowledgments can be recovered. A one in the n^{th} bit position can be mapped as an ACK from one of the N (number of subcarriers) clients.

Due to the conversion between the time domain and frequency domain, relatively tight timing synchronization is needed for the composite additive signal to be decoded at the AP – in other words, all the responding stations must transmit at about the same time; however, that time synchronization is provided by the broadcast message itself as explained in §5.3.2.

To understand how much more efficient it is to use physical layer signaling, consider the costs of transmitting a message using the 802.11g PHY that is the basis for our extension. A normal message requires a $20\mu s$ preamble to be transmitted and then, at best assuming the $54Mbps$ modulation rate, each 48×6 bits takes one OFDM symbol time ($4\mu s$) to transmit. Thus, a 64 byte message, which can't actually even contain the Ethernet addresses in a standard 802.11g packet would take at least $20 + 4 \times 3$ or 32μseconds. After

a $16\mu s$ "SIFS" period for a 20MHz channel [34], clients would normally respond using a similar message format. Thus, a single response to a standard 802.11g packet would take another $\approx (32 + 16) = 48\mu s$.

By comparison, using physical layer signaling **53** clients can provide a single bit of information within two OFDM symbol periods, or a total of $8\mu s$ (as detailed in §5.3.2), or one-sixth the time for a **single** station to respond using standard messages. This means that using the proposed protocol, the time needed for a single station will be reduced by about an order of magnitude; when the number of potential respondents increases, that time is reduced by two orders of magnitude.

5.2 Robustness of SMACK

5.2.1 Against Varying Signal Power

The reliable broadcast acknowledgment scheme described in §5.1 typically caters to a network of directly reachable nodes. The signal power from these clients may vary widely. Setting a single threshold for all these clients would be difficult if the received signal power of each of the subcarriers at the AP vary in a broad range. Hence, we propose to adjust the transmission power of tone transmitters/clients such that the received power of the subcarriers from different clients at the AP are comparable and within tolerable limits, ensuring that the weaker signal does not get lost due to the high power of the stronger signal. The dynamic transmit power adjustment of the clients can be decided based on existing channel assessment techniques as done in CDMA [38]. The calibration of the transmit power control mechanism based on the channel condition is kept as future work. In this way, we can set a single threshold to detect all the clients in the network, as the received power of the individual subcarriers become similar after adaptive power control. To detect the farthest client, we need to detect its signal. We argue that the weakest client's signal at the AP is not only detectable, but also decodable if a packet is transmitted. Otherwise normal 802.11 communication with that client will not have been possible. In case our proposed protocol fails to detect acknowledgment from the weakest client, the fallback mechanism to retransmit to that particular client will ensure reliable delivery of the broadcast message to the client.

5.2.2 Against Interference

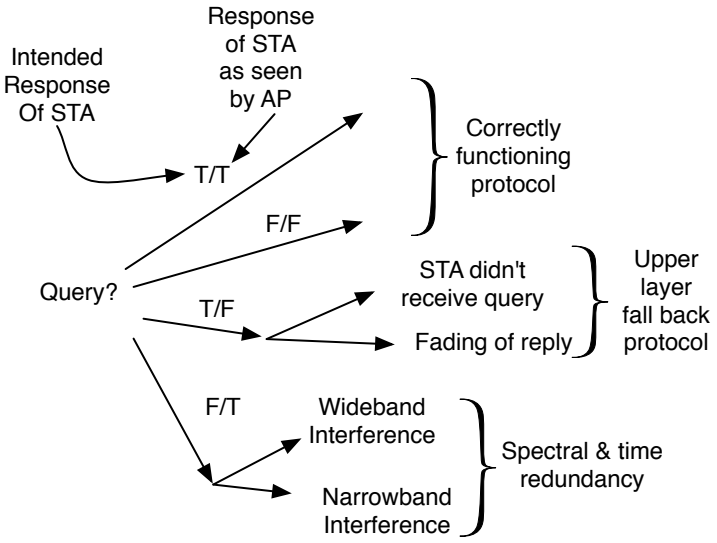


Figure 5.2: Protocol Fallback Decision Tree

A significant contributor that might cause the protocol to degrade are spurious or burst noise in 2.4 GHz ISM band, e.g., Zigbee, Bluetooth devices, microwave oven and interference from hidden terminals.

In order to address such scenarios we present a fallback mechanism of the protocol which involves upper layer intervention in order to make the protocol robust and reliable in the presence of spurious interference. Figure 5.2 shows the possible states of the protocol and the decision making mechanism at the AP. We start by defining the *cause* and *effect* of the protocol's decision branch. *Cause* refers to *the intended responses of the stations/clients* and *effect* is defined as *the response of the stations/clients as detected by the AP*. Both the cause and the effect can have two possible binary states - True or False. Based on all possible combinations of cause and effect we address the error correcting mechanisms or a fallback method.

Branches *True/True* and *False/False* - These two branches exhibit error free functioning of the protocol. If the intended and actual responses match then no error correction is required.

Branch *True/False* - This decision branch can be attributed to instantaneous channel noise between the AP and station. This error can occur in two ways: either the station did not receive the broadcast message or the ACK is attenuated at the AP and fails the threshold test. We refer to the second phenomenon,

where the station transmits the tone but the AP does not recognize it, as a *False Negative*. It is possible that a receiver may simply not hear the query and fail to respond. As with any protocol that assumes the absence of response to be meaningful, some higher level method is needed to insure that such a decision is appropriate or that the protocol should be amended to insure that only **positive** responses are acted on.

Branch *False/True* - Wideband or Narrowband noise can cause the threshold test to falsely trigger and we refer this phenomenon as a *False Positive*. As described in section 5.3.2 the signal detection mechanism operates in a small time window of $4\mu sec$ after the SIFS period. So if there exists any unwanted narrowband or broadband signal within the FFT window that can be taken care of in the following way.

Interference can be of two types - either a narrowband or a broadband. We refer to any interference less than $20MHz$ bandwidth as *narrowband interference*, which essentially corrupts the intended spectrum partially. Zigbee, which operates in a 5MHz bandwidth can be one of the potential narrowband interferers. Hidden terminal clients of another AP using our protocol can also be another potential narrowband interferer. To reduce the errors introduced due to narrowband interference, we can assign each client multiple subcarriers to transmit ACKs. This mechanism will allow the AP to detect a false positive by employing a simple *all or nothing* decision metric. If the AP fails to detect energy in all of the subcarriers assigned to a client, it is regarded as a false positive. Assigning multiple random subcarriers $5MHz$ apart will ensure robustness against interference from Zigbee nodes. Also, we argue that there exist remote possibilities where a hidden terminal client of another AP in our protocol is assigned the exact same combination of subcarriers as one of the intended clients of our AP and respond in the exact same time slot of our FFT window for detection. Hence, we do not address this problem in this work.

We refer to any unwanted signal of equal to or more than $20MHz$ bandwidth as *wideband interference*, which causes false positives in the detection mechanism at the AP. For a long-lived wideband interference we can eliminate the chances of false detection by performing FFT immediately before and after the protocol window of $(8\mu sec + 2\mu sec) = 10\mu sec$ as in Figure 5.4 – if signals are detected prior to or following the intended transmission time, the likely source of those signals would be long-term noise or interference. To detect errors due to wideband interference of duration less than $10\mu sec$, we keep two subcarriers ($+20$ and -20) unassigned to any client. Energy in any of these two subcarriers will detect the

presence of a wideband interference. In this scenario, a rebroadcast after carrier sensing can efficiently solve the problem. However, if the wideband interference is very short lived in the order of $nsec$ (as in UWB), it will not affect the FFT results as the sampling frequency of our system is $12.5nsec$ which is more than the pulse width.

5.3 System Parameters

Normal wireless communication is a point-to-multi-point process involving a single transmitter and one or more receivers; our design inverts that assumption. There are some important challenges in implementing such a protocol.

5.3.1 Threshold

The use of thresholds is very common in signal detection and decoding. From the basic operation of carrier sensing in CSMA/CA to maximum likelihood decoding of baseband modulation to even advanced forms of spectrum sensing in cognitive radio environment, all employ some form of threshold testing to extract information from the received signal. In this implementation we utilize Fourier analysis, which is efficiently implemented in hardware using the Fast Fourier Transform (FFT) algorithm. We use threshold tests to identify the presence of spectral components (**i.e.**, is a station transmitting a tone?).

For a fairly simple signaling mechanism as described in §5.1 we simply need to look at the average signal power to decide on a threshold. Input signal levels are strictly controlled by automatic gain controllers at the receiver front-ends to prevent saturation of the analog to digital converters (ADC). The average received signal strength (RSS) can be measured using eq.5.1, where $r(d)$ refers to the received signal samples and D refers to averaging filter length.

$$R(d) = \sum_{i=0}^{D-1} |r_{d+i}|^2 \tag{5.1}$$

Figure 5.3 shows experimental results from hardware where signal energy is averaged over 128 samples. As long as the envelope of the composite waveform is kept constant the average signal energy does not change much and is always above the average noise floor. Thus we argue that this average RSS can be used

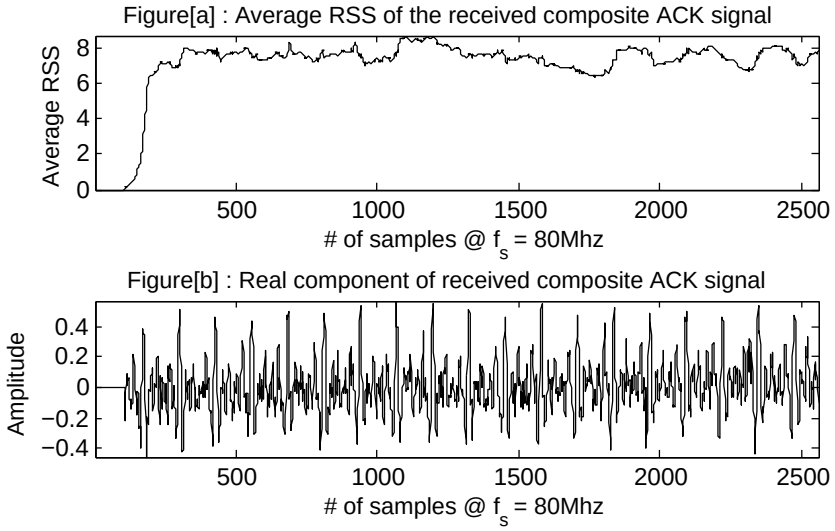


Figure 5.3: Received signal strength

to determine the threshold level and there is no need to change threshold over time as long as the average signal energy is kept fairly constant by suitable gain controller.

5.3.2 Timing Considerations

The effectiveness of using Fourier transform to extract spectral components requires all the subcarriers to be present with sufficient energy within the FFT window. In this implementation (§8.3) we have used a 256-point FFT that corresponds to one OFDM symbol ($3.2\mu s$). Therefore, this window of 256 samples should have all the subcarrier information. Evidently, there is an implicit timing constraint imposed on the broadcast node. This is further worsened due to the near-far effect and the different processing power of the clients nodes causing the tones to reach the AP at different times. Therefore the broadcast node has to estimate a suitable FFT window to successfully receive the ACKs. This time is calculated from, after the last sample transmitted to air interface to the first sample of a valid FFT window, which is given by eq. 5.2.

$$T \geq 2 \times T_{propagation} + T_{r_x latency} + T_{hardware} + T_{t_x latency} \tag{5.2}$$

Assuming a typical distance from the AP to the farthest node in an infrastructure based network to be

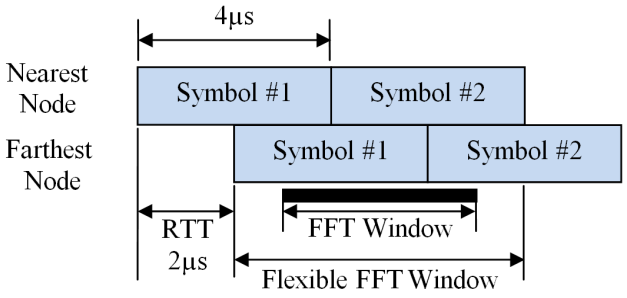


Figure 5.4: FFT timing requirement

$\approx 300m$ results in a round trip delay of about $2\mu s$, together with receive-transmit path latencies and $R_x - T_x$ turnaround time for our hardware ($T_{r_xlatency} + T_{hardware} + T_{t_xlatency}$) allows us to decide on the correct FFT window. Given that each OFDM symbol has a duration of $4\mu s$, we can define a **flexible FFT window** which compensates for all the latencies and propagation delays as given in eq.5.2.

Figure. 5.4 shows the relative timing diagram and optimum FFT windows. Given a RTT of $2\mu s$ from the farthest node we start the FFT window anywhere after $2\mu s$ which gives us enough flexibility against any unforeseen signal delays. The **"black bar"** marks the optimum FFT window of $3.2\mu s$ or 256 sample wide.

Unlike single user OFDM transmission, strict receiver timing synchronization is not required since no demodulation is required despite receiving data from multiple clients – we are simply detecting "energy in the channel". Also, since these are unique single frequency tones, the OFDM subcarriers are transmitted without any PLCP header or any identifiers like pilot tones which saves bandwidth and makes detection faster at the AP. This makes implementation fairly simple and straightforward, and the technique should be able to be implemented on commodity 802.11 hardware.

5.3.3 Frequency offset and Doppler shift

The composite baseband received signal can be represented by

$$r(n) = \sum_{i=0}^{N-1} A_i e^{j2\pi(f_i + \delta f_o + \delta f_{d_i})nT_s} \tag{5.3}$$

where A_i,f_i,δf_o, δf_{d_i} are respectively the resultant amplitude, subcarrier frequency, frequency offset during down-conversion at the receiver and the Doppler shift for the i^{th} subcarrier.

Frequency offset correction is extremely important for normal OFDM based packet transmissions. Any residual frequency from the down-conversion stage may cause a significant change in modulation level, which makes it impossible to decode (demodulate) the signal.

This is precisely the reason why we **do not** demodulate the signal – we simply look for power in the subcarrier (**i.e.**, a "tone"). Since we are not worried about modulation levels, any offset in frequency will not affect the FFT results. Thus we argue that since the subcarrier spacing for our implementation is $312.5KHz$, carrier frequency offsets, which is typically in tens of KHz for the radios used in our experiments, will not cause subcarriers to shift frequency bins.

Doppler spread is the maximum frequency shift between the transmitter and the receiver caused by their relative motion or by any scatterer in the environment. Doppler shift is given by eq. 5.4.

$$f_m = \frac{v f_c}{c} \tag{5.4}$$

where f_m is the maximum frequency shift of the signal transmitted at the carrier frequency of f_c, with a relative velocity of v between the transmitter and the receiver; c being the velocity of light. Using eq.5.4, for a object moving at $5km/hr$ which is a typical human walking speed we have a maximum Doppler shift of approximately $11Hz$. Therefore the Doppler shift is not sufficient to cause spectral leakage onto adjoining subcarriers. Unless the nodes are highly mobile it is very unlikely that the sinusoid envelope will vary to such an extent to cause the threshold test to fail. Neither will it cause the subcarrier to shift frequency bin leading to false detections.

5.4 Implementing SMACK using SDR

To demonstrate simultaneous reception for reliable acknowledgments we implemented a prototype using a SDR platform. The SDR involves an OFDM transceiver on a Virtex-IV FPGA along with a custom front-end radio as shown in Figure 5.5. The design and implementation has been detailed in [11, 14], which deals with all the signal processing algorithms that have been synthesized into fixed point hardware designs. The platform is capable of transmitting and receiving generic 802.11g as given in physical layer specification [34]. The OFDM transceiver components consist of a custom radio front-end responsible for up/down

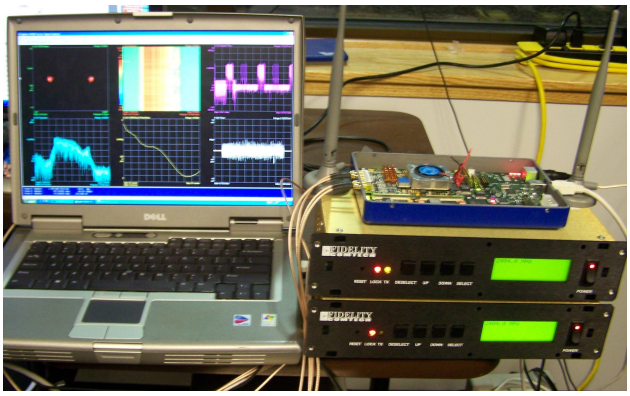


Figure 5.5: Nallatech boards with radios and antennas

conversion to/from the 2.4GHz ISM band and a Xilinx ExtremeDSP development kit IV manufactured by Nallatech. The ExtremeDSP board includes either a Virtex IV or a Virtex II FPGA equipped with a PCI/USB interface and two sets of A/D and D/A converters. Gain control is also a part of the radio that can be controlled by software on the host computer.

Transceiver latency plays an important role in our implementation. It is required to determine the turnaround time for the receiver at the broadcast node. Usually for any practical transceiver, the minimum time that is required for the MAC/PHY to receive the last symbol of a frame at the air interface, process the frame and respond with the first symbol on the air interface of the response frame is of great interest. This includes receiver side PHY layer processing delay + MAC processing delay + Transmitter side processing delay + PCI transfer delay for both Rx and Tx + Front-end radio hardware delay. If we disregard the MAC processing delay and the PCI transfer delay then we can summarize the following:

(1) Receiver side:

Difference between the last symbol received at the air interface to last bit transferred to host = 14.83μsec.

(2) Transmitter side:

Difference between the FIFO read signal to the first analog sample out from the DAC = 11.68 μsec.

(3) Key note: The FFT/IFFT module consumes the bulk of the latency = 7.4 μsec. x 2 (for Tx and Rx) = 14.8μsec.

It is observed that most clock cycles are consumed by the FFT/IFFT unit and other than that the latency is attributed largely to various buffering elements required for proper functioning of the pipeline. in order to further reduce latency we need to use better pipelined cores with faster cycle times. This is purely a limitation of our prototyping hardware, and not of the method – any commercial WiFi chipset is already capable of the processing needed to implement our technique.

The receiver side of the broadcast node comprises of an FFT engine coupled with the energy detection blocks as shown in Figure 5.6. This design can form a part of the standard receiver chain [14] and the mode of operation (depending on if the node is operating as a client of AP) can be easily selected using software controlled registers.

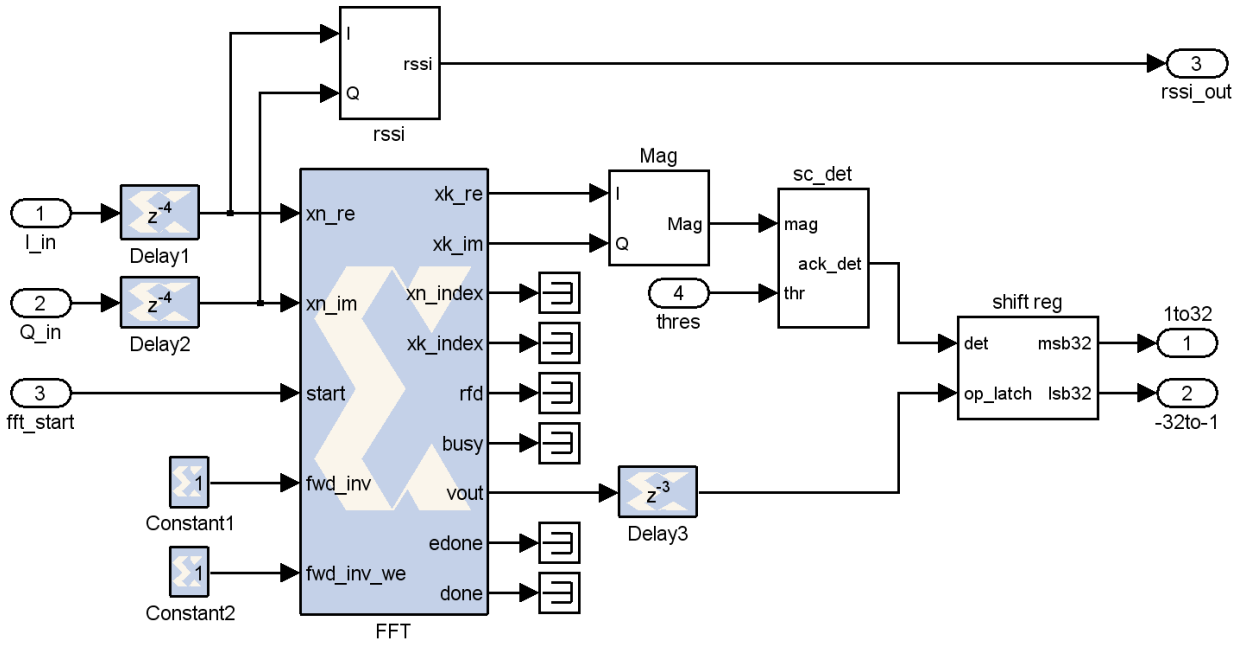


Figure 5.6: Design for the detecting ACK at AP

As explained in §5.3.2, triggering the FFT is a key design challenge. Given our hardware design and its inherent latencies, we find that the total time required for an ACK to reach the AP is $(T_{r_x latency} + T_{t_x latency}) = 26.51\mu s$. Since ACK transmission control logic is done in hardware, no MAC processing delay or PCI/USB data transfer delays are introduced. In order to accommodate any propagation delays and other eventualities we further add a cushion of $2.49\mu s$ to the above latency. Thus we trigger the FFT exactly

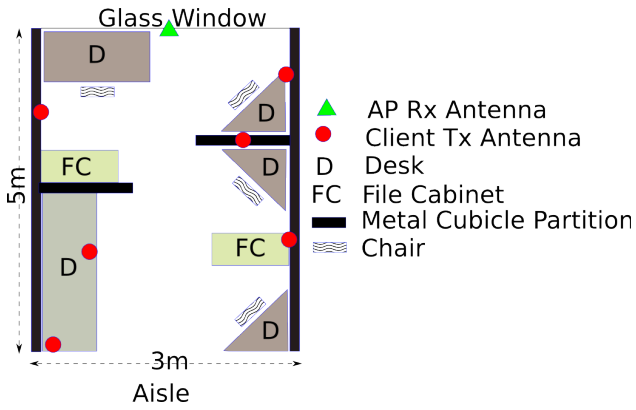


Figure 5.7: Floor-map of experimental setup

$29\mu s$ after the last sample of the broadcast packet transmitted to air interface. This time difference ensures that all the ACK tones from client nodes are available with sufficient energy at the AP to be able to use a simple threshold test to detect them.

The transceiver is operated in the 2.4GHz ISM band with a $20MHz$ bandwidth in order to co-exist with other $802.11a/g$ transmissions. The $20MHz$ spectrum is split into 64 subcarriers including the 0^{th} subcarrier (d.c.). The 0^{th} subcarrier is never used as it will introduce unwanted DC offset at the receiver which has to be removed using suitable algorithms. The output of the energy detector is typically a **bit mask** of 63 subcarriers (excluding the dc subcarrier). This 63 bit mask is read by the MAC layer routine using two software addressable registers. The bit mask for $-ve$ subcarriers are numbered MSB = -32 to LSB = -1 while the $+ve$ subcarriers are numbered MSB = 1 to LSB = 33 (which happens to be always zero as we are using 32 subcarriers). For example, if subcarriers [-26, -16, -6, +6, +11, +16] are being used to transmit ACKs then the bit mask for the $-ve$ frequencies is given by $0x2008020$ and that of the $+ve$ frequency is $0x4210000$. The presence of a '1' in the bit mask indicates that subcarrier index is used to transmit the ACK. Thus a reliable and fairly simple detection of acknowledgment has been accomplished using a software defined radio.

5.5 Experimental Setup

In this section, we describe our experimental setup and methodology to understand how feasible subcarrier detection mechanism is in reality.

For compatibility with existing 802.11 compliant networks, our clients would have to transmit an acknowledgment within the **SIFS** period of the broadcast packet to avoid collision with any other transmissions. However, hidden terminals are not immune to this scenario and may cause collision at the client nodes. However, if the receiver receives the packet and transmits the tone, the chances of collision are very low at the AP. Either a client will transmit a tone due to reception of the broadcast packet, or fail to transmit tones due to the loss of the broadcast packet. Other nodes not participating in the broadcast that are outside the transmission range of the AP will back-off after they sense the broadcast signal transmitted by the AP due to normal carrier sense mechanisms. Thus, coexistence with existing 802.11 networks will not be a problem if stations transmit tones within the SIFS period. The 2.4GHz band is also shared by 802.15 Zigbee nodes as well, but they use similar CSMA/CA sensing mechanism before transmission, which will ensure successful coexistence with our network. Any protocol using a carrier sense media access will similarly be compatible.

Our prototype system, as described in §5.4, cannot transmit the tone within the SIFS period. Hence, setting up experiments in the presence of other 802.11 networks would induce erroneous results in our protocol evaluation. So, we have used $2.484GHz$ as the carrier frequency for our experiments. Closest to the IEEE 802.11 channel 11 ($2.462GHz$), this band of $20MHz$ is free from any transmissions generated by WiFi cards, but has very similar propagation properties to those used by the 802.11 network. This channel is also affected by microwave ovens, and other spurious transmissions generated by different electrical devices (all of which occurred during our prototype evaluation).

Figure 5.7 shows a floor-plan of our indoor setup, with 6 tone transmitters and 1 receiver/detector. The distances between the transmitters and the receiver in our testbed can be extended, and experiments with longer distances and more number of nodes remain as future work.

5.6 Results

To maximally utilize 7 available radios, we decided to show the performance of our protocol in two steps. The first set of experiments demonstrate the eficiency of the subcarrier detection mechanism, as described in §5.6.1. The second set of experiments demonstrate actual transmission of a broadcast packet, followed by tone transmission from two nodes on successful reception of broadcast packet, as detailed in §5.6.2. We used 3 Nallatech Virtex IV PCI boards as 3 client nodes or the tone transmitters. The rest of the 4 boards were Nallatech Virtex II boards equipped with a USB interface. Each of the 6 clients were set in transmit mode, equipped with one radio and a transmitter antenna, continuously transmitting tones in a pre-assigned subcarrier. The detector node is setup in receive mode and repeatedly triggers the detection mechanism to realize the performance of the energy detection scheme. Three of the client nodes were in *line-of-sight* (LOS) of the detector antenna, and the rest were purposefully positioned in *non-line-of-sight* nLOS to introduce sufficient signal distortion. The maximum distance between the transmitter and the receiver antenna was approximately $5m$. Antennas were placed at a height of approximately $2m$ from ground level. All the results shown in Figure 5.9, 5.10, 5.11 and 5.12 are averaged over five individual experiments at different times of the day, each experiment was performed 10,000 times to detect the tones. This is done to show the robustness of the detection mechanism in presence of ambient noise.

It is to be noted that since we are performing signal processing at baseband using digitized samples, units of various parameters are not important because they are represented using fixed-point precision once converted from the analog domain. For baseband processing, absolute values as quantized by the ADC are important and not the true measured values in units of current or voltage. The actual values in units of current or voltage will depend on the number representation in the design and the dynamic range of the ADCs and other electrical components prior to the ADC. Therefore, without loss of generality and integrity, the units of all our variables are to be interpreted as absolute values.

5.6.1 Efficiency of Tone Detection

In this section, we determine the performance of our protocol, which is based on tone detection in different subcarriers. Initially, we aim to show the variation of signal in both time and frequency domain and how the variation affects the selection of threshold. Then, we have chosen three different setups to analyze the effect of spectral leakage around the desired subcarriers. In experiment 1, evenly spaced subcarriers have been assigned to minimize any spectral leakage. In experiment 2, every alternate subcarrier has been chosen to detect the effect of spectral leakage in the intermediate unassigned subcarriers. Experiment 3 has been designed to assign contiguous subcarriers for transmission, such that spectral leak may affect detection at the two extremities of the set of subcarriers.

5.6.1.1 Threshold Selection

To demonstrate the variability of the spectrum over time and its effect on detection percentage, we collected spectrum data in the same indoor setup as shown in Figure 5.7. The receiver gathered $204.8\mu s$ of signal, which indicates data for 64 successive FFT computations, each of duration $3.2\mu s$. In this way, we collected the composite signal at three different times of the day, resulting in $(64 \times 3) = 192$ FFT computations. Figure 5.8 shows the variation of spectrum energy in both frequency and time. There are three regions of signal in time ([1-64], [65-128] and [129-192]), all plotted sequentially. Since coherence time of the channel is more than 64 FFT computations, we do not notice any major change in signal power within a single region. However, individual subcarriers undergo fading at different times of the day, as we move from the region of [1-64] FFT computations to the region of [65-128] computations. Figure 5.8 also shows that there is a considerable amount of variation from $-47.93dBm$ to $-57.26dBm$, in signal power among different subcarriers at the same FFT computation time. However, these variations are not enough to create a problem in selecting a single threshold, $-65dBm$ as shown in the figure. Although individual subcarriers undergo attenuation over time the average signal energy envelope remains almost constant. This helps us in maintaining a steady threshold for tone detection.

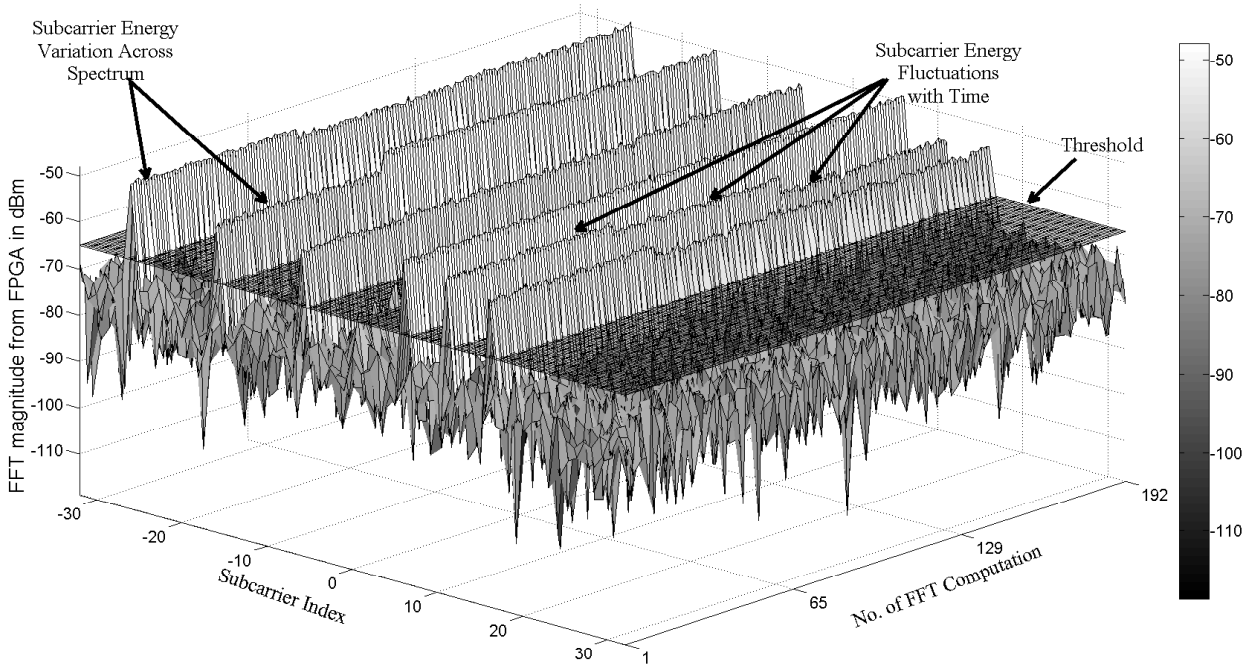
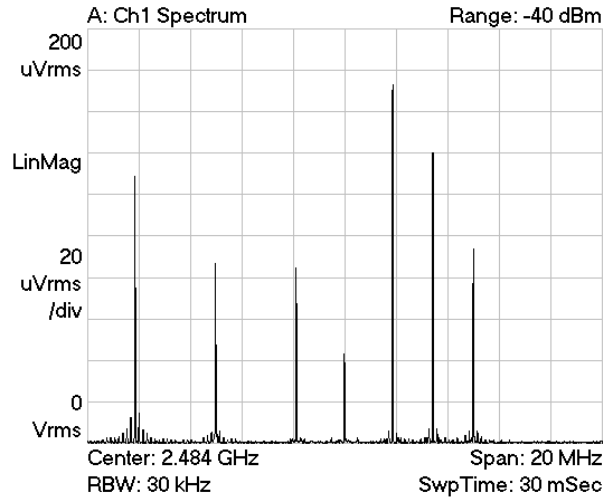


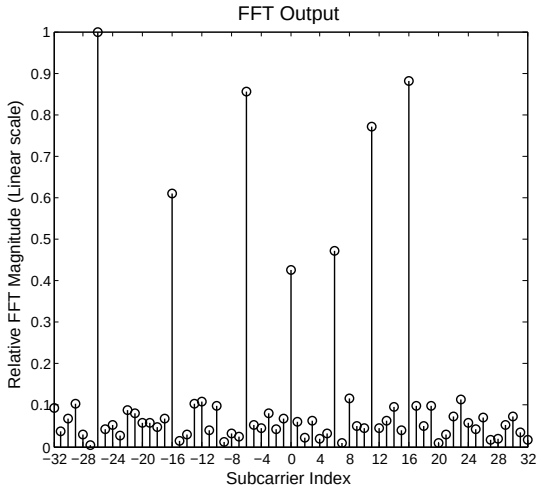
Figure 5.8: Variation of spectrum over time

5.6.1.2 Experiment #1 - Evenly Spaced Subcarriers

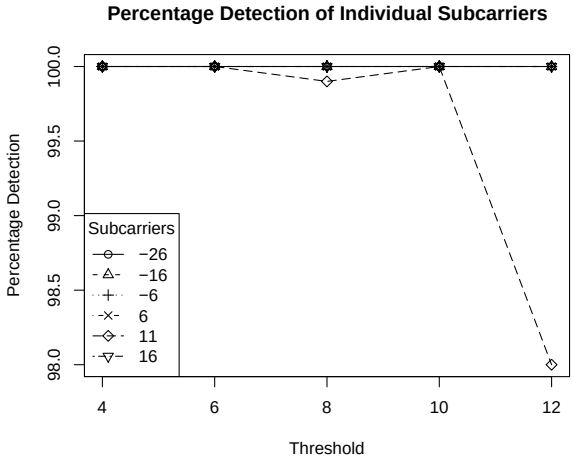
In order to benchmark our system performance we used an Agilent 89600S Vector Signal Analyzer (VSA) to compute the spectral components while we present our computation using the FPGA based FFT engine. For this experiment we have chosen subcarriers [-26, -16, -6, $+6$, $+11$, $+16$] which are widely spaced not to interfere with each other. Figure 5.9(a) and 5.9(b) shows the similarity in the FFT computations by the VSA and our hardware. However it is to be noted that although the measurements are spaced in time and have different subcarrier amplitudes, they provide the same spectral components which have been seen to be consistent over prolonged duration of time. Figure 5.9(c) shows high detection percentage at lower thresholds, while the percentage of detection of heavily attenuated subcarrier $+11$ reduces only 2% at the maximum threshold. We notice that the threshold can be easily chosen from a broad range of 6 to 10.



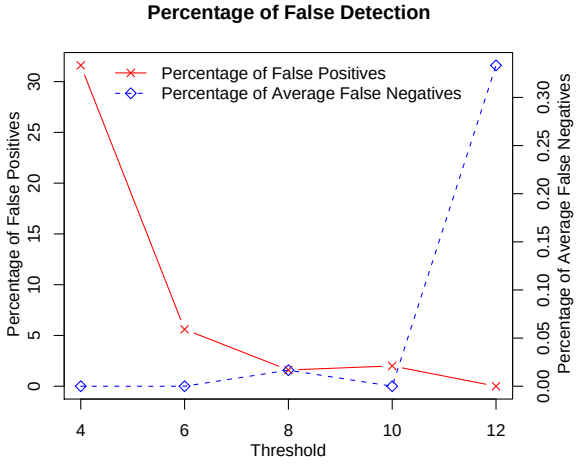
(a) FFT result from VSA



(b) FFT result from FPGA



(c) Detection percentage



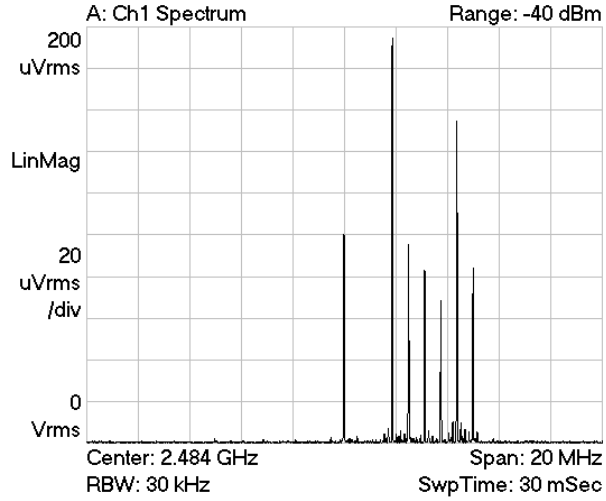
(d) False positive and False negatives

Figure 5.9: Result of Experiment #1 : Clients transmitting in widely spaced subcarriers - [-26, -16,-6,+6,+11,+16]

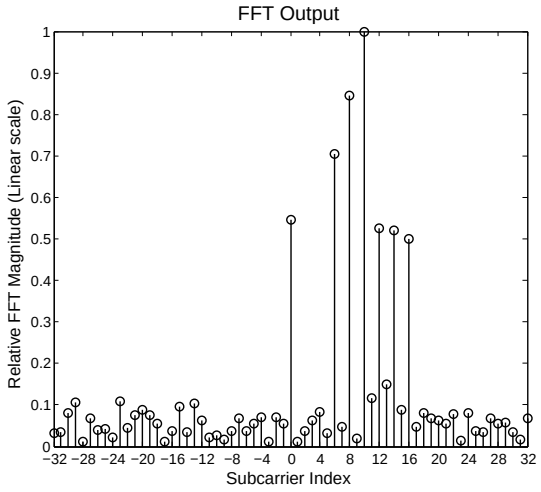
5.6.1.3 Experiment #2 - Closely Spaced Subcarriers

Subcarriers [+6, +8, +10, +12, +14, +16] have been used to demonstrate the effect of spectral leakage of detection percentage. Again, Figure 5.10(a) and 5.10(b) shows identical spectral components. A drop in detection percentage for subcarrier +14 at threshold 8 can be attributed to instantaneous deep fading during the measurement phase. Figure 5.10(d) shows that even at low thresholds the number of false

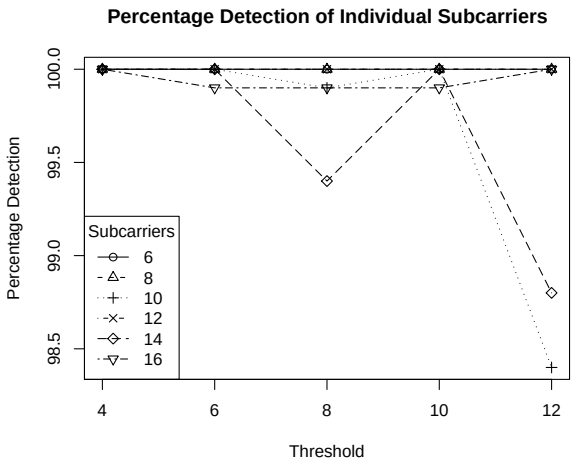
positives are low. This really shows that energy in other subcarriers which forms the noise floor for the threshold test is very low.



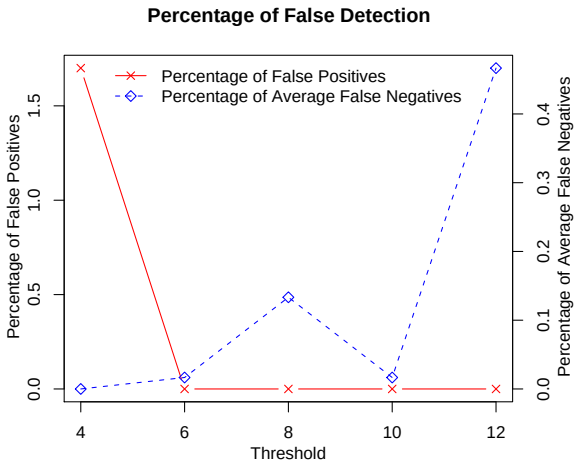
(a) FFT result from VSA



(b) FFT result from Hardware



(c) Detection percentage



(d) False positive and False negatives

Figure 5.10: Result of Experiment #2 : Clients transmitting in closely spaced subcarriers - [+6,+8,+10,+12,+14,+16]

5.6.1.4 Experiment #3 - Contiguous Subcarriers

Transmitting tones on contiguous subcarriers, for example, [+8, +9, +10, +11, +12, +13], is representative of a pathological case. With results in shown in Figure 5.11(c) and 5.11(d) we argue that even

with contiguous subcarriers there is very limited inter-subcarrier interference. The detection percentage and false positives show similar trends to that of experiment #1, which shows that even under the most critical case the spectral components are easily detected by performing simple Fourier transform.

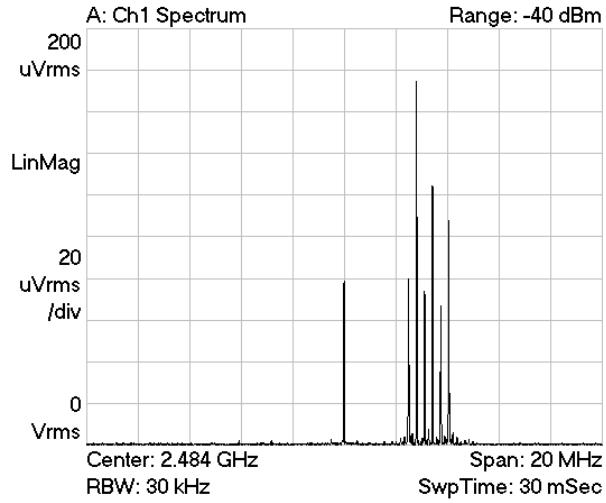
The detection percentage together with the false positives and false negatives in all three experiments show that with our experimental setup and resources, it is not hard to determine an optimal threshold, which is 8 in this case. Threshold testing is applied at the output of the FFT engine, using the absolute value of the FFT result on a linear scale. The threshold values show in the Figure 5.9, 5.10, 5.11 are scaled and adjusted numbers to suit the output signals levels of our fixed point FFT engine. The important thing to note is how the detection mechanism performs with changing threshold, rather than the actual number in the threshold axis.

5.6.2 Complete System Performance

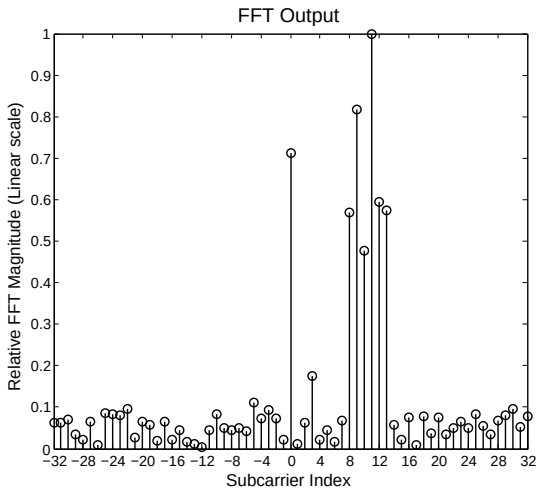
To demonstrate the correctness of the detection mechanism and the timing requirements mentioned in §5.3.2 and §8.3 we setup a testbench using three nodes equipped with our SDRs. One of the nodes is setup as the broadcaster, transmitting broadcast packets at regular intervals using BPSK 1/2 rate modulation, and performing an FFT to detect subcarrier energy after $29\mu s$ as described in §5.3.2. The other two responder nodes placed at 5m line-of-sight from the broadcaster, and are setup to transmit tones at subcarriers $+12$ and -12 respectively. The nodes only transmit the tone if they receive a broadcast packet correctly.

Figure 5.12 shows the overall performance of the complete setup. We notice that with only two subcarriers, the noise floor is very low and percentage of detection is high. The subcarrier -12 has been transmitted at a higher transmit power than subcarrier $+12$. We notice the effect in our results as well. False Detection is calculated per subcarrier, any false detection in positive frequencies has been considered to be the outliers caused by subcarrier $+12$, and vice-versa. Threshold 3 appears to be a low threshold for subcarrier -12, with percentage false positive of 2.5%. We notice detection of both the subcarriers -11 and -13 frequently. Since subcarrier $+12$ has a lower energy, we see that at threshold 12, percentage of detection deteriorates. In this scenario, threshold can be kept anywhere between 5 to 10 for optimum results.

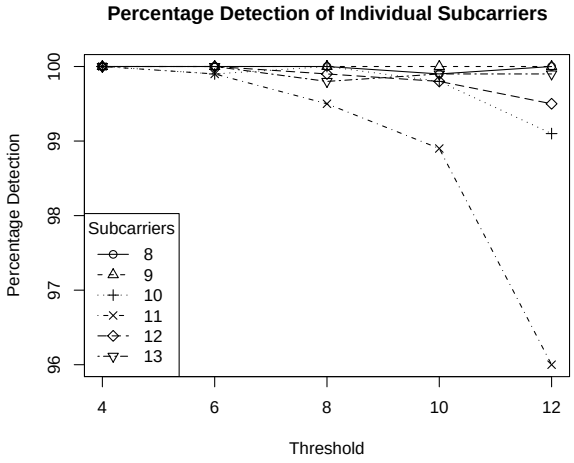
Experimental results in this section not only prove that we can use simple Fourier transform to detect



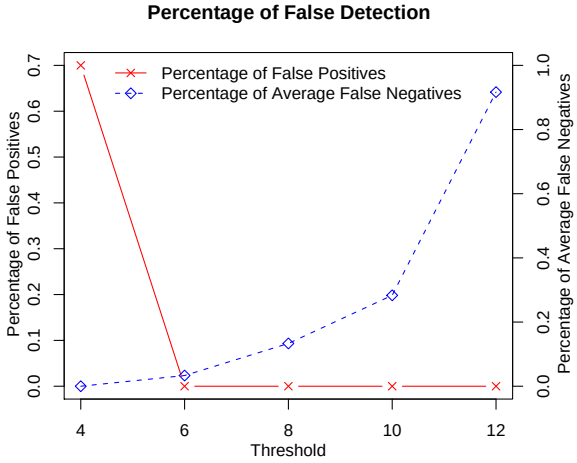
(a) FFT result from VSA



(b) FFT result from Hardware



(c) Detection percentage



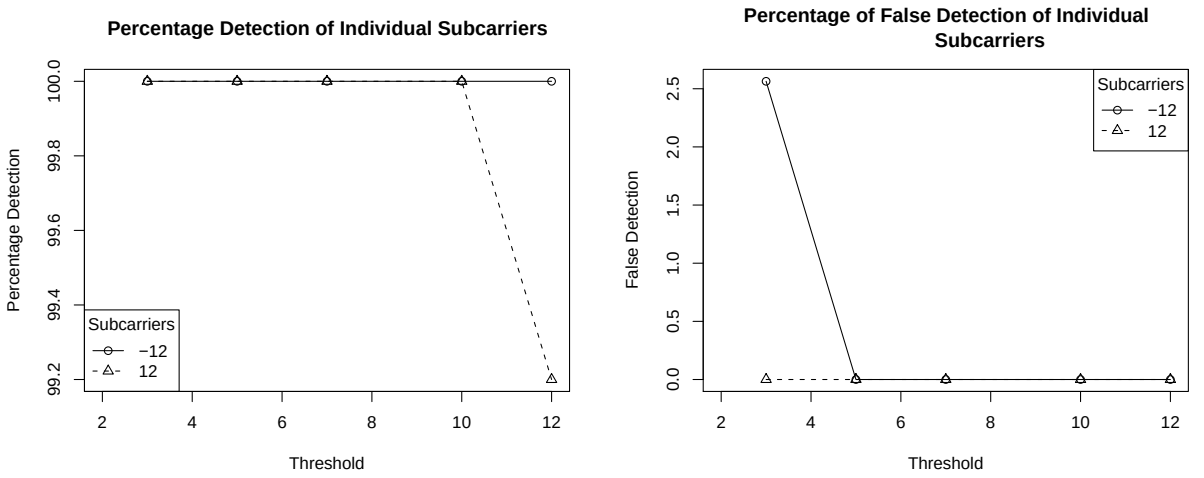
(d) False positive and False negatives

Figure 5.11: Result of Experiment #3 : Clients transmitting in contiguous subcarriers - [+8,+9,+10,+11,+12,+13]

multiple tone transmissions no matter how dense the subcarrier spacing is, but also show that implementing such mechanism using a reconfigurable radio to meet the timing constraints is indeed feasible.

5.7 Discussions

In this section, we discuss the robustness of our scheme to low client SNR and SNR variations across clients. In our experiments, the minimum client SNR is measured to be $15.65dB$ and the maximum as



(a) Percentage detection of individual subcarriers

(b) Percentage of false detection of individual subcarriers

Figure 5.12: Complete system performance with one broadcaster and two responders

$27.07dB$. In networks larger than our testbed, the client SNRs may be lower and span a wider range. Our conjecture is that such scenarios can be addressed as follows: although the maximum and the minimum SNR values will reduce, the power control mechanism, as described in §5.2.1, should be able to keep the variation within the limits of our current measured SNR range. Despite the fact that the minimum SNR from the weakest client will be less than the minimum shown in our experiments, we argue that if a modulated packet can be decoded from that client, which requires both amplitude and phase detection, our detection mechanism will be able to detect the existence of energy in that subcarrier. However, in low SNR regimes, unlike single user OFDM transmission, our multi-user protocol will have different inter-subcarrier interference properties. The effect of such interference in our protocol needs to be evaluated by further experiments.

5.8 Beyond Acknowledgments

In this section, we discuss how simultaneous communication mechanism can be utilized in higher layers to improve various protocols.

5.8.1 Reducing Redundant Rebroadcast

We can extend our single hop ACK mechanism further to reduce redundant rebroadcasts in multihop wireless networks by choosing a remote neighbor for the next broadcast in a network-wide broadcast. We exploit the physical layer signaling to estimate the relative distances of the neighbors, by detecting concurrent ACKs. Due to near-far effect, the signal from the "near" node arrives before that of the "far" node, as shown in Figure 5.4. This can be detected at the physical layer using multiple overlapping FFT's at the beginning of signal reception. Among the set of nodes that respond we can determine which ones are further away (assuming they take the same amount of overhead time to start the ACK transmission). We can exploit that information and select the farthest node for retransmission of broadcast, thus building a reliable broadcast protocol in multihop mesh networks with a minimum number of broadcast packets that mitigates broadcast storm.

5.8.2 Parallel Polling

Concurrent communication mechanisms can be utilized in polling nodes whether they have packets to transmit, and based on the polling results, medium access mechanism can be ascertained. The parallel polling mechanism can be used by the AP [39] to query its clients about their queue length. Based on the responses, the AP can assign variable slots to the clients for uninterrupted transmission. This mechanism is faster than any other polling mechanisms, which require transmission of a series of packets by all the participating nodes to know the responses.

5.9 Related Work

This work presents detection of concurrent transmission as a mechanism to acknowledge broadcast/multicast packets. Mitigating broadcast storms and making the broadcasts reliable are two important issues that are inter-twined and addressed by many researchers in different ways. To reduce redundant broadcasts, the authors in [40] propose several schemes, namely probabilistic, counter-based, distance-based, location-based and cluster-based schemes. However, there is no acknowledgment mechanism to ensure that

each of the neighbors have received the message. To ensure reliable broadcast with permanent probabilistic failures, an asymptotic bound for achievability of broadcast has been deduced in [41].

Acknowledgment is an important phenomenon to report whether a message has been successfully received by the intended receiver. The performance of various response collecting methods, like polling, TDMA and Group Testing, have been compared in [42]. These protocols require transmission of multiple packets transmitted by different nodes, which are distributed temporally. Comparatively, our protocol collects responses simultaneously from multiple nodes within a very short period of time without transmission of any response packets. Demirbas **et al.** [43] proposes **Pollcast** to estimate the number of simultaneous responses of a polling by checking the RSSI; a collision will increase the received signal strength. A variation of similar work has been proposed in **Backcast** [44], where acknowledgment is transmitted without any source address by multiple nodes at the same time. Results show that with fewer nodes, the ACK is decodable and received signal strength approximately indicates the number of concurrent transmissions. Both **Pollcast** and **Backcast** are incapable of detecting the exact neighbor who has transmitted the response. We move a step forward from these mechanisms and not only correctly detect the number of concurrent transmissions, but also detect the exact neighbors that have participated in transmitting the acknowledgment.

The increasing popularity of OFDM in current wireless technologies has convinced us to choose it as the underlying mechanism of communication. It has been embraced by current wireless technologies in IEEE 802.11 WLAN [45] and WiMax [46]. It is also one of the physical layer communication system in IEEE 802.22 WRAN [47]. Simultaneous transmission of tones or simply each node transmitting in a single subcarrier, has the same orthogonal property, but requires less complexity at the receiver to detect. OFDM/OFDMA utilizes the bandwidth by transmitting in a set of subcarriers, which requires pilot tones inserted at regular intervals in frequency domain to capture the channel coefficients and aid equalization [48]. To decode a packet transmitted over a set of subcarriers, it is necessary to equalize the received signal with the help of information received from the pilot tones. Our mechanism uses simple energy detection scheme at the receiver without the hassle of equalization and baseband decoding.

OFDMA has also been introduced in cellular network as a simultaneous communication mechanism, where subcarrier assignment [49, 50] considers a set of contiguous subcarriers. Non-contiguous OFDM [23]

has mostly been popular in the cognitive radio domain, where a transmitter does not have access to a contiguous set of subcarriers for transmission due to presence of primary users. In this scenario, timing synchronization [51] and decoding the signal is a challenge. Although in our case, the signal generated from multiple nodes is a non-contiguous OFDM signal, our protocol only requires energy detection in each of the subcarriers and hence do not encounter the challenges of non-contiguous OFDM communication.

Simultaneous transmissions can also be detected by the multi-user detection scheme in CDMA. To detect CDMA codes transmitted by the clients, the receiver has to perform correlation for all the N clients/codes. The post processing of the signal is time consuming if an elimination process is used, or extremely resource consuming if N parallel correlators are used. To avoid complexity of the problem, researchers [52] use various heuristic methods to obtain a suboptimal solution.

Instances of using simultaneous tone transmissions on OFDM subcarriers for higher layer applications are rare. Energy detection of subcarriers has been utilized by Roman **et al.** [53] in a leader election protocol to eliminate contenders for channel access mechanism. Here, authors use only 8 subcarriers to indicate whether it is contending for the wireless medium. After a few number of contending slots, a winner is decided which gets access to the medium. However we demonstrate the use of the signaling mechanism to address a broader array of network problems. We also address the challenges and scope of implementing such a protocol using reconfigurable hardware, which is the novelty of this work.

Spectrum sensing also forms an integral part in the evolution of cognitive radio based research. Although there are various approaches to find spectrum holes, as given in [54], we still find that the basic operation is a set of threshold tests that ultimately differentiates the **good** signals from the **bad**. Although SNR Wall [55] remains a problem for simple detection mechanism in cognitive domain, our protocol does not suffer from this effect. In cognitive radios, the secondary user should be able to robustly detect the presence of a primary user in the vicinity even from hidden positions where the primary user's signal goes below the 'SNR Wall' and is difficult to be detected by a simple threshold test. In our protocol, the tones are transmitted by clients of an AP, who are all one-hop neighbors. The signal from those clients are high enough that even packets transmitted by the clients can be decoded at the AP. So, the signal energy is not as low as we notice in cognitive radio domain and a simple threshold test, as we suggest, can be used to detect

the tones.

Prior works in a similar domain powered by simple implementable algorithms has led us to explore beyond the boundaries of preset methods and innovate new protocols in the domain of wireless networks.

5.9.1 Conclusion

We've shown that by using, rather than fighting against, the properties of the wireless physical media, we can develop robust signaling primitives that are both practical and allow innovative algorithms. We used a signaling method based on OFDM that is easy to understand and implement using reconfigurable hardware. We have also shown that if the signaling mechanism is kept simple, not only does it makes certain network functions, such as reliable broadcasts faster, but can also use simple detection mechanism to extract the required information. These primitives can also be used to implement higher level group communication and signaling protocols as long as the queries require simple "yes/no" answers. The critical insight is that we can combine the results from multiple clients using simultaneous reception in an efficient manner to aid higher protocols to perform more efficiently.

Chapter 6

Active Radar

In this work, we present a cooperative technology that senses rapid changes in traffic and communicates in the network to minimize traffic hazards, by employing a **software defined radio** for both "cooperative RADAR" and vehicular networking. Our method uses multicarrier wireless communication to **detect** and **disseminate**. Using precise timing and synchronization, we can detect the distance of each of the vehicles, their current velocity and current acceleration or deceleration conditions. Using simultaneous, multi-party acknowledgments, we can rapidly disseminate or determine information about a number of vehicles in an efficient manner.

Vehicle safety can be greatly increased by situational awareness, which is increased by sensing systems and vehicular network disseminating the information. In this work, we seek to combine the benefits of radar systems and vehicular networks, using a novel paradigm enabled by **software defined radios**. Our system depends on the protocol designer being able to mix PHY-layer and MAC-layer signaling. The basic concept is to rely on **simultaneous reception of limited information from multiple parties**. In our system, we use individual subcarriers of an OFDM signal to represent responses, but we could supplement this with any orthogonal signaling (code division, **etc**).

In a crowded highway traffic accident, radar systems will only warn the cars in line-of-sight of vehicles decelerating in the accident. The information will be propagated, though slowly, when each of the cars decelerates. But the time taken by the drivers in vehicles at non-line-of-sight to react to the sudden deceleration may be long enough to lead to disastrous results. Other wireless communication based approaches will generate many communication messages that might overburden the network. By comparison, our protocol

broadcasts a periodic "is there a problem?" message, and individual cars can respond simultaneously by orthogonal signaling to indicate a problem. Moreover, we can use the timing of the response to indicate the **distance** to a respondent, allowing us to take corrective actions. Because multiple cars can respond simultaneously, the response takes little time, allowing many such broadcast packets to be used.

The rest of the chapter is organized as follows: §6.1 presents the protocol design for vehicle collision avoidance. §6.2 we dicuss the hardware implementaion of the protocol using SDR.

6.1 Protocol

In this work, we focus on estimating distance and sudden acceleration or deceleration of vehicles by using **simultaneous transmission and reception** in multicarrier modulation systems. A node periodically transmits a 'Query' message, and adjacent vehicles respond back transmitting a tone in a random subcarrier. If the processing is done at the hardware, the processing time is approximately the same for all the nodes, and the time to respond only depends on the propagation delay. The original querying node now detects energy in each of the subcarriers to estimate distance and acceleration.

6.1.1 Simultaneous Transmission in Multi-carrier Modulation

For this implementation we have chosen the Orthogonal Frequency Division Multiplexing (OFDM) based physical layer for $802.11a/g$ as the underlying signaling, which allows simultaneous transmission in 52 orthogonal subcarriers, which can be detected at the receiver by simple Fourier Transform. We utilize each of the orthogonal subcarriers to transmit a tone, which is detected at the receiver by measuring the energy in the subcarrier. It is not required to demodulate the signal, since we are not transmitting any modulated information in the subcarrier. The receiver node only detects the time of arrival of the signal to determine the distance. Compared to other models, which require periodic transmission of messages, our method just transmits tones of duration of two OFDM symbols, thus reducing the effective time to transmit response to a broadcast packet.

To summarize, the protocol has the following steps:

(1) A node (radio in a vehicle) carrier-senses the channel and transmits a 'Query' packet using the complete bandwidth available to it.

(2) All nodes (radio receivers in other vehicles, which are in the first vehicle's radio range) receiving this 'Query' packet immediately respond by transmitting a tone in one of the subcarriers, chosen randomly.

(3) The initiating node detects the time of arrival of signal in each of the subcarriers and calculates the approximate distance of any car from it. It receives the composite time domain signal of **all** OFDM subcarriers and performs an FFT to obtain the frequency domain representation of the signal. The start of the signal is determined by performing multiple FFTs during reception.

(4) The initiating node again transmits a second 'Query' packet without carrier sensing after SIFS time, so that it gets access to the channel. This time, responders reply using the same subcarrier as chosen in the first time.

(5) The initiator node compares the time of arrival of the new signal with the one at first time and calculates the current condition (constant velocity, acceleration, or deceleration) of the vehicle, which is described in detail in 6.1.2.

If two vehicles choose the same subcarrier to transmit their tone, then the initiator node receives the signal first from the nearer node followed and overlapped by the signal from the farther node. In this case, the initiator remains unaware of the current conditions of the farther node; however, we are typically more interested in nearby conditions.

6.1.2 Estimation of Distance and Acceleration

Figure 6.2 shows the timing diagram of the signals transmitted on air by the initiator node and two responders, node A and node B. T_r is the time required by the responder to respond to the 'Query' packet. T_r includes processing time at the responder and round-trip propagation delay. If the processing is done at the hardware, the processing time is the same for all the responders, and the time to respond back only

depends on the round trip propagation delay. T_rA and T_rB denote the times required to respond by node A and node B respectively. T_rA_1 and T_rA_2 are the response times of A for the first and second queries respectively. These response times are detected by the initiator node and round trip propagation delay is extracted by subtracting the processing time from T_r. If T_rA_1 equals T_rA_2, then relative velocity of the vehicle with respect to the initiator vehicle is constant and no warning is generated. If T_rA_1 is less than T_rA_2, then the relative velocity of the vehicles are changing. But in this case, the responder is moving away from the initiator vehicle. Hence, no alert is generated. However, if T_rA_1 is greater than T_rA_2, then vehicle A is either in front of the initiator vehicle and is applying brakes, or is behind the initiator vehicle, pumping the accelerator. In this scenario, we generate a collision warning.

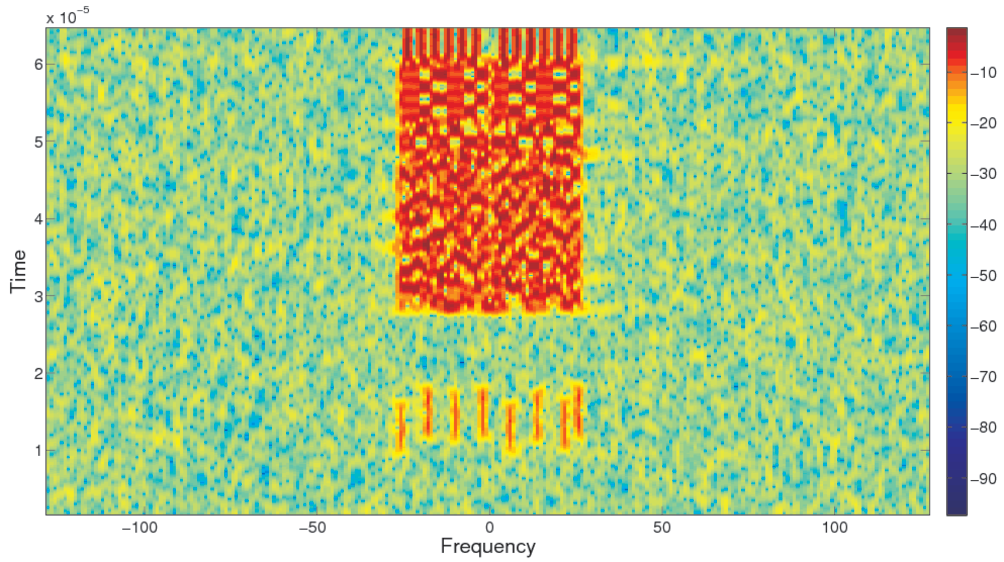


Figure 6.1: Waterfall Plot of 8 nodes transmitting at different times

6.2 Hardware Implementation

To demonstrate tone transmission and simultaneous reception of OFDM, we implemented a prototype using a software defined radio platform, the basic design has been discussed in [11, 14]. The tone transmission is done by selecting one of the subcarriers in the transmitter design. The initiator node receives a composite additive signal from all the neighbors and, depending upon the number of users, the number of

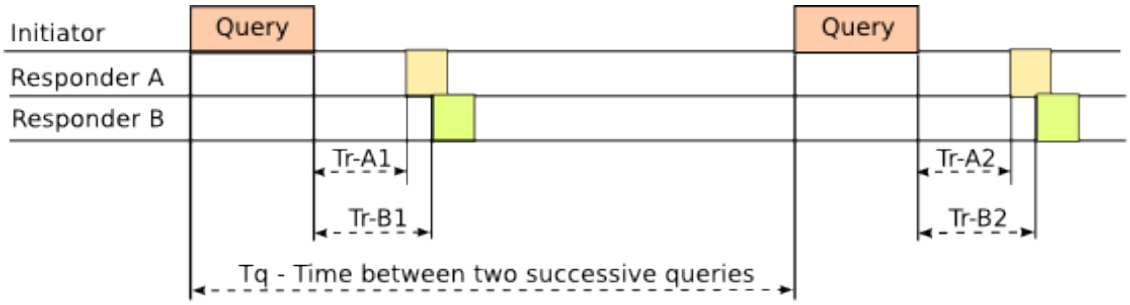


Figure 6.2: Timing Diagram

distinct frequency components in the signal will vary, as shown in Figure 6.3. A simple Fourier transform at the initiator will reveal the tones in the signal. Observing the magnitude of the Fourier transform we can identify high energy subcarriers. However to estimate the time-of-arrival of each response, we need to perform the FFT continuously as a sliding window over the received samples. In our implementation, the sampling time is $12.5ns$, which gives us a detectable distance of $3.75m$ between two cars. We used one radio for transmission of broadcast message, which is received and decoded by two other radios in the vicinity, followed by tone transmissions by those two radios. Figure 6.4 shows that the node transmitting in subcarrier $+8$ has a higher signal power (closer to the initiator node) compared to the one transmitting using subcarrier -8 (farther from initiator node).

6.3 Conclusion

The idea of using multicarrier communication in vehicular safety applications is innovative, and incorporates the benefits of existing applications, while excluding the shortcomings of the current solutions. We implemented a prototype in hardware, and shown results in static scenarios. In the future, we plan to extend our work and show results from moving vehicles.

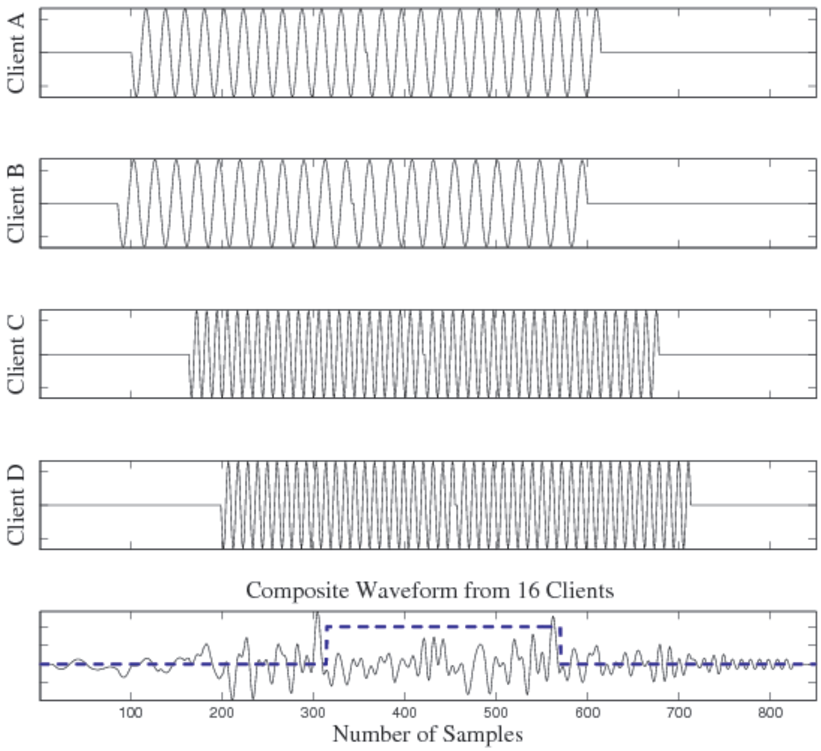


Figure 6.3: Timing Offsets Between Responses and FFT Window

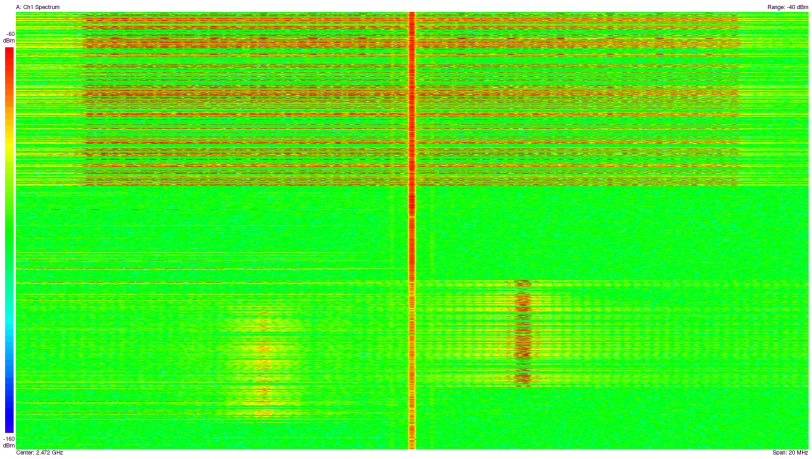


Figure 6.4: Waterfall Plot using Two Prototype Radio Platforms, frequency in X-axis and time in Y-axis, captured by Signal Analyzer

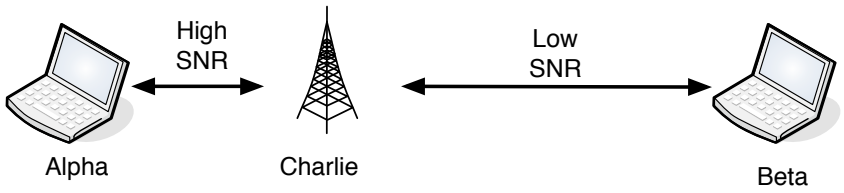
Chapter 7

GRaTIS- Free Bits in the Network

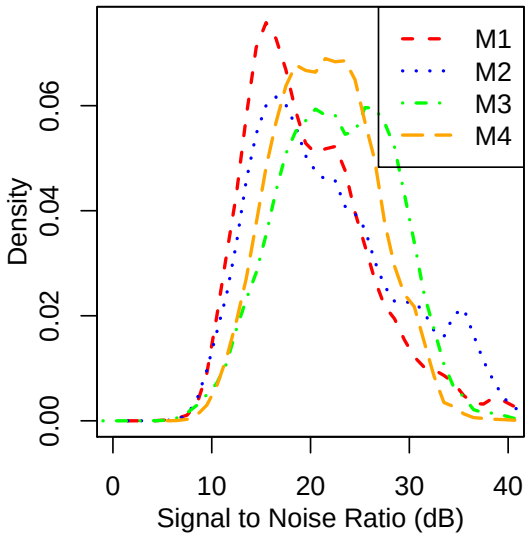
Modern wired, wireless and optical communication systems use different **modulation schemes** to balance data rates against error rates. Each modulation scheme encodes a varying number of bits in a physical representation of the data. Low rate modulations are used in a noisy channel and high-rate modulations are used when the SNR is higher. For example, wireless systems such as 802.11 or WiMAX use the following modulations: BPSK (2 states, 1 bit), QPSK (4 states, 2 bits), 16QAM (16 states, 4 bits) and 64QAM (64 states, 8 bits). These modulates are augmented with different redundancy codes to achieve a fixed number of "transmission rates".

Those transmission rates are robust under varying SNRs. For example, the QPSK-3/4 rate requires an SNR of 8.0dB to deliver 18Mb/s throughput. The 16QAM-$1/2$ rate, delivering 24Mb/s, could be used if the SNR was 12.5dB, however if the SNR falls between 8dB and 12.5dB we get a "better" signal that reduces packet drops but results in little net throughput improvement. In this work, we show that it is possible to exploit the higher SNR of this primary node to encode another message for a second receiver, increasing the aggregate network bandwidth. In doing so we introduce multiple data rates to provide an even gradation of SNR across a group of receivers – we call this Group Rate Transmission with Intertwined Symbols, or GRaTIS. Extending the prior example, consider the network organization in figure 7.1(a). If the link between Charlie and Beta has an SNR of 10dB, and the link between Charlie and Alpha has an SNR of 20.5dB, using GRaTIS we can send an 18Mb/s message to one node **and simultaneously send another** 18**Mb/s message to a second node resulting in an effective throughput of** 36**Mb/s**.

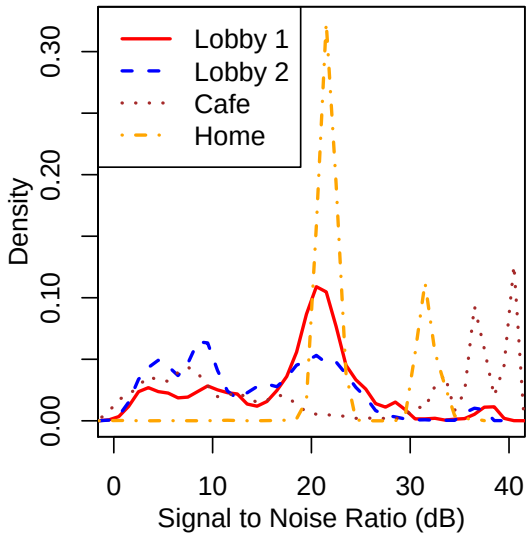
Our method depends on **SNR diversity** between the receivers – in other words, we exploit the fact



(a) Example Of Spatial Rate Diversity



(b) SNR Profile 1



(c) SNR Profile 2

Figure 7.1: Variation of SNR due to spatial diversity in 802.11a/g networks. Profile 1: Measured indoors by 4 packet sniffers at SIGCOMM 2008 [1] Profile 2: Measured indoors in common areas around a university cafe and lobbies and also in home networks.

that most networks have nodes that experience different SNR values. Fortunately, most networks exhibit considerable SNR diversity. Figure 7.1(b) shows the distributions of SNR at a number of locations measured at a SIGCOMM conference in 2008 and figure 7.1(c) shows the same variation, which are measured by us around a university campus. Figure 7.1(c) also shows a SNR profile that is typical to a home network shared by two users with high volume video streaming. The diversity in SNR occurs because of the spatial layout of nodes, room geometries and interference from other sources.

A broader set of related work is discussed in §7.8, but it is useful to summarize how GRaTIS relates to similar ideas. GRaTIS is similar to **hierarchical modulation**, which is used in digital broadcasting, but differs because it is applied to non-broadcast data and combines data targeted for specific nodes. CDMA

networks use **multiuser detection** methods in the uplink and exploit the gains from orthogonal spreading codes and closed loop power control. CDMA systems use successive interference cancellation (SIC), which has been examined in a broader networking context [56]. SIC removes an interfering signal in order to reveal a secondary signal, and is part of the complex signal processing used in CDMA base-stations. By comparison, GRaTIS can address either infrastructure or **ad hoc** traffic (although our analysis emphasizes an infrastructure downlink), is simple to implement and depends on the wireless stations having **diverse** SNR levels, simplifying control. In GRaTIS, the original signal is encoded to be received correctly by the two receivers and doesn't suffer from channel estimation errors and the concerns that have been raised about the broader utility of SIC [57]. SIC is also used in **superposition codes**, which is another method for entwining two messages [58, 59], but superposition coding requires significantly enhanced SNR for both receivers of the message, limiting its utility and it is also difficult to implement. By comparison, GRaTIS requires only a modest increase in SNR that provides the required diversity to achieve network wide gain and is easy to implement on conventional signal processing pipelines.

The benefits of GRaTIS are the larger number of **group rates**, which provide increased opportunities for improving performance but doesn't impact performance when not used; it can be made backward compatible with existing wireless networks; it is easily implemented on conventional wireless signal processing pipelines; and, it complements the gains with advanced rate adaptation techniques [56, 60, 61] and physical layer techniques such as FARA [62] and network coding [63]. Using different analysis techniques we show that this scheme is both practical, profitable and implementable. We summarize the key contributions of this research as follows:

- We reinterpret the constellations already available for conventional wireless links and provide **group rates**, which result in higher network throughput with no hardware changes.
- We perform a standard analysis of packet error rates for this scheme to ascertain the applicability in real networks.
- We implement GRaTIS on a 802.11a/g compatible software defined radio (SDR) prototype to show that the technique is easy to implement and makes use of existing hardware modulation and de-

modulation methods. In the prototype, much of the added processing is handled by simple software, rather than complicated fixed-function hardware and DSP algorithms.

- We use the SDR nodes in a testbed setup to measure the SNR requirement for over-the-air transmission of GRaTIS encoded multiuser data packets.

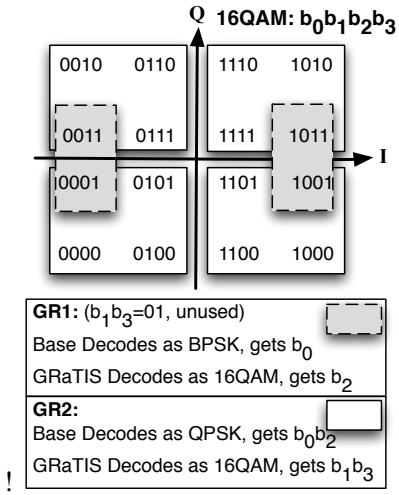
- We apply the results of the testbed to analyze various 802.11a/g traces from SIGCOMM conferences and other small/mid sized wireless data networks to determine how the diverse range of SNR of the receivers can be used to obtain substantial gain in network throughput. This shows the potential performance improvement when GRaTIS is applied on realistic downlink traffic.

- We also show that GRaTIS outperforms a competing method (superposition coding) both in theory and practice. We validate this through experiments conducted in a testbed of SDR nodes using actual over-the-air packet transmissions.

In §7.1, we describe the GRaTIS technique in more detail. In §7.2 we conduct a basic packet-error analysis for the GRaTIS protocol showing under what conditions the technique can be used. We then describe the hardware used to implement the technique in §7.4.1 and the results from our implementation in §7.4.2. §7.5 describes a (potential) gain analysis for GRaTIS using 802.11a/g packet traces. The benefits of sending two messages at once can yield very different results as described in §7.7 and §7.6 and finally in §7.8, we discuss prior work in this domain and place our work among contemporary techniques.

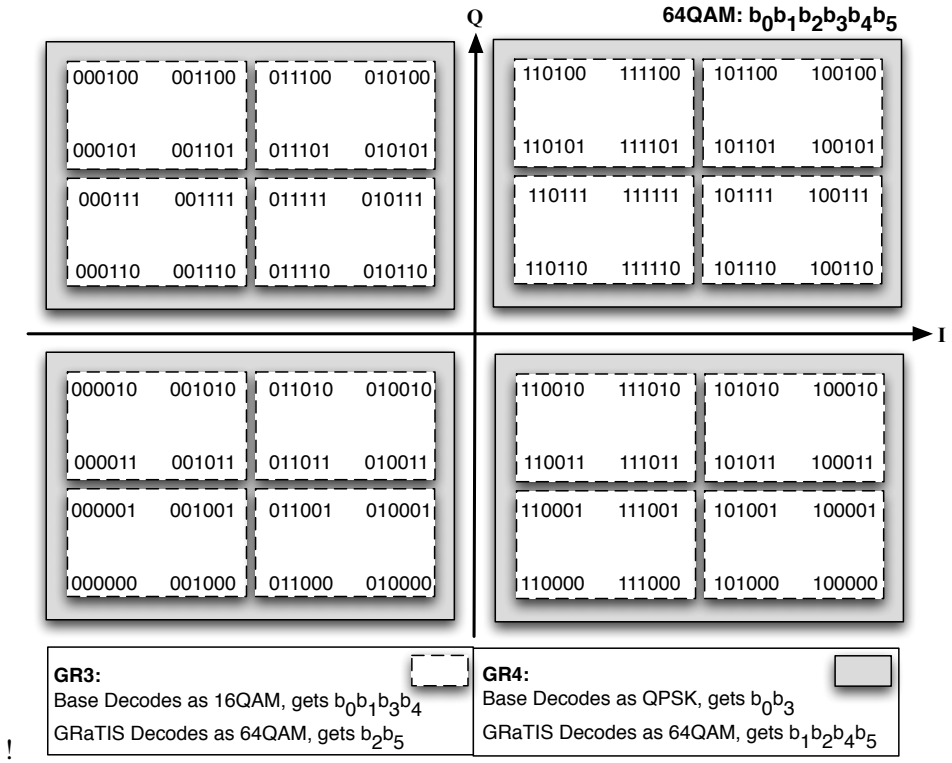
7.1 GRaTIS: Free Bits

In this section, we describe GRaTIS. Discrete steps in the SNR requirements for each data rate forces the rate adaptation algorithm to fall back to a lower rate even if the node is reachable at a SNR higher than the minimum required. In this scenario, we use GRaTIS to identify two "layers" of constellations within the standard constellations available in 802.11a/g. These two layers are used to map two packets of two different users to form one single packet, such that the time required to transmit the merged packet in GRaTIS is less than the time required to transmit two separate packets in the best achievable data rate of 802.11a/g. The two



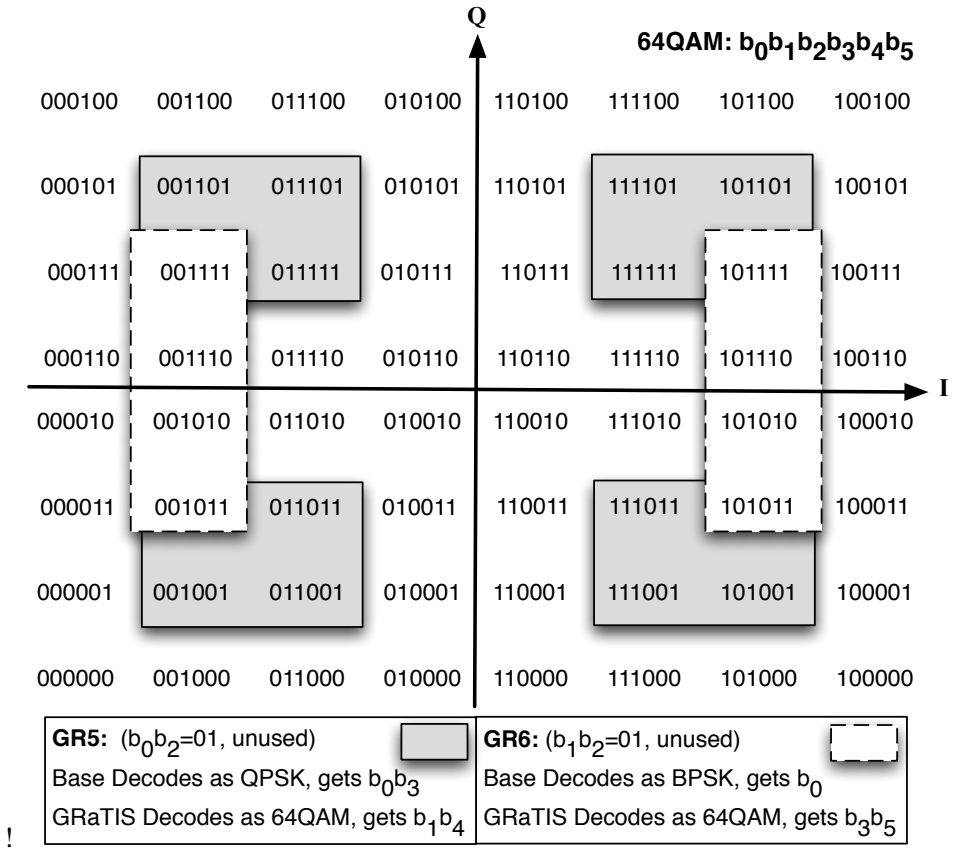
(a) GR1 and GR2

Figure 7.2: Encoding and decoding of GRaTIS derived from standard 802.11a/g constellations. Rectangular regions show the transmitted cluster and the corresponding decision boundary for the base layer.



(b) GR3 and GR4

Figure 7.2: Encoding and decoding of GRaTIS derived from standard 802.11a/g constellations. Rectangular regions show the transmitted cluster and the corresponding decision boundary for the base layer. (continued)



(c) GR5 and GR6

Figure 7.2: Encoding and decoding of GRaTIS derived from standard 802.11a/g constellations. Rectangular regions show the transmitted cluster and the corresponding decision boundary for the base layer. (continued)

layers are designed in such a way that one of the layers can be decoded by a legacy decoder, and we call it the **Base Layer**. The second layer is obtained by extracting a few bits after the legacy demodulation system converts the I/Q samples from the analog domain to the binary domain, and we call it **GRaTIS Layer**. The second packet is transmitted during the transmission of the first packet, without any extra airtime, and comes as free bits to the receiver with a higher SNR – those free bits increase the aggregate throughput of the network.

For example, assume the received SNR of two nodes n_1 and n_2 from a common transmitter are SNR_{n_1} and SNR_{n_2} respectively. Also, there exists a GRaTIS rate, where the SNR requirement for Base and GRaTIS layers are SNR_b and SNR_g respectively, and $SNR_b <= SNR_{n_1}$ and $SNR_g <= SNR_{n_2}$. Consider the best achievable data rate in 802.11a/g are R_{n_1} and R_{n_2}, while that using GRaTIS are R_b and R_g

respectively. The common transmitter uses GRaTIS to transmit x bits of data at rate R_b to node n_1, which takes time t_g (total transmission time using GRaTIS). Since in GRaTIS, the GRaTIS layer is transmitted at the same time along with the Base layer, there is no extra time required to transmit the GRaTIS layer. So, the transmitter also transmits y bits of data using rate R_g within the same time t_g. Therefore the total data rate of this transmission is $\frac{(x+y)}{t_g}$.

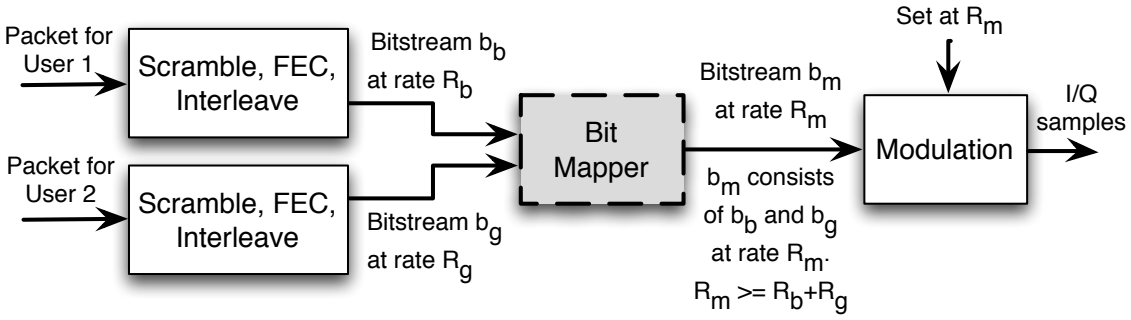
Now we calculate the achievable data rate if the common transmitter uses 802.11a/g to transmit the same packets. The transmitter uses rate R_{n_1} to transmit the x bits of data to node n_1 in time t_1. After this, it transmits y bits of data at rate R_{n_2} in time t_2. The aggregate data rate for these two transmissions is $\frac{(x+y)}{t_1+t_2}$. The pair of rates R_b and R_g are selected as one of the GRaTIS rates, **iff** $\frac{(x+y)}{t_1+t_2} < \frac{(x+y)}{t_g}$. Or in other words, GRaTIS rates are selected only if there is potential gain in aggregate throughput over the legacy system.

We have developed an encoding and decoding technique that requires minimum change to a stand alone 802.11a/g transceiver and relies on identifying clusters that are a subset of the standard set of 802.11a/g constellations (BPSK, QPSK, 16QAM and 64QAM). Depending on the cluster size and how they are split between the two layers, the error performance of the base layer and the GRaTIS layer varies. Typically, the clusters are derived from higher order constellations (16QAM and 64QAM) so that there is a sufficient amount of bits available to encode the packet for the GRaTIS layer.

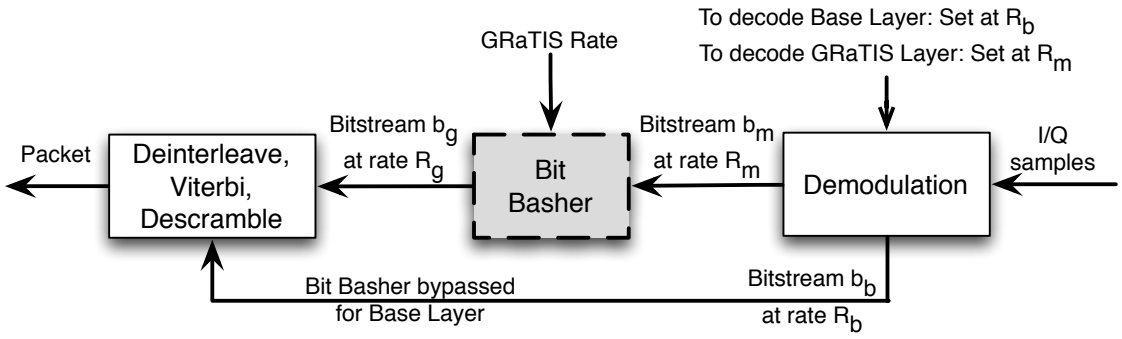
7.1.1 Encoding Packets using GRaTIS

We introduce **six** distinct GRaTIS rates, or methods of combining packets, as shown in figure 7.2, to increase aggregate throughput of the network. The GRaTIS rates are termed **GR1** through **GR6**. We select the combination of layers as one of the GRaTIS rates if the data rate achievable by merging is more than that of two packets transmitted separately. At the transmitter, two packets are encoded independently up to the modulation subsystem as shown in figure 7.3(a). Then the bits of two packets, b_b and b_g, are encoded at rates R_b for the base and R_g for the GRaTIS layer respectively, and combined to form a compound symbol that represents one of the constellation points corresponding to a standard modulation in 802.11a/g, denoted by R_m. This mapping ensures that the compound symbols are mapped only to the I/Q vectors that are part of a selected GRaTIS cluster. In this way, the modulator remains unchanged, as it is fed with the compound

bitstream (b_m), and modulation type (R_m) to which it modulates. Since all the packet merging is done at the bit-level it does not require any change in the signal processing pipeline.



(a) Transmitter Pipeline



(b) Receiver Pipeline

Figure 7.3: Transceiver pipeline for GRaTIS – shaded subsystems show additional processes required for GRaTIS.

The clusters are selected to optimize properties of the I/Q-plane mapping used to represent information in wireless networks. For example, Figure 7.2(a) shows the constellation points used in **GR1** and **GR2**, which are derived from a 16QAM constellation. In **GR1**, resultant cluster points are modulated to carry two bits of useful information, one bit for each layer. To reduce the probability of error in the base layer, the points are chosen such that the vectors in the I-plane are greater than that of BPSK mapping, while the deviations in the vectors of the Q-plane is used to carry another BPSK packet in the GRaTIS layer. Out of the 4 bits available for every constellation point in the cluster, bit b_0 is used to encode the base layer, and bit b_2 contains the GRaTIS layer. The other two bits, b_1 and b_3, remain constant at 0 and 1, respectively to map the compound symbol to the desired cluster. The shaded region shows the transmitted constellation for **GR1**. **GR2** uses all the constellation points of a 16QAM constellation, and provides 2 bits of information

per subcarrier, as is done in QPSK, to each of the two nodes. Bits b_0b_2 and bits b_1b_3 are used to encode the information of base layer and GRaTIS, respectively. As a result of such mapping when a cluster point that is closest to an axis crosses the axes due to channel noise, a symbol error occurs for the base layer, as seen in QPSK modulations, but this event does not incur any error in the GRaTIS layer. Hence, this type of mapping provides some extra error protection to the GRaTIS layer and so, the SNR requirement for GRaTIS layer of **GR2** is less than that of 16QAM.

GR3 and **GR4** utilizes 64QAM constellation to encode the two layers as shown in figure 7.2(b). **GR3** provides a 16QAM data rate to the base layer using bits $b_0b_1b_3b_4$, while **GR4** provides a QPSK data rate to the base layer using bits b_0b_3. The remaining bits are used to encode the data of GRaTIS layer. As in **GR2**, the GRaTIS in these cases have additional protection from error as crossing the base layer boundaries does not introduce any error in the GRaTIS layer. **GR5** and **GR6** uses a cluster derived from a 64QAM constellation as shown in figure 7.2(c). **GR5** uses bits b_0b_3 for the base layer and bits b_1b_4 for the GRaTIS, providing QPSK data rate to both the packets. Bits b_2b_5 are modulated as 1, to generate the desired cluster as shown in dash-dotted lines. **GR6** uses bits b_0 and b_3b_5 to encode the information of the base and GRaTIS layers, respectively, while modulating bits b_1 as 0 and b_2 as 1 leading to the desired cluster points.

7.1.2 Decoding Packets using GRaTIS

GRaTIS has two layers, intended for two receivers. The Base layer is encoded in such a way that its decoding is the same as decoding any generic 802.11a/g packet. Decoding the constellations to information bits is done using pre-defined thresholds called **decision boundaries**. For BPSK, this boundary is the Q-axis, whereas for QPSK there are four such boundaries: the four quadrants of the I/Q plane. The number of decision boundaries increases with the increasing number of points in the constellation. For example, in **GR1**, bit b_0 of 16QAM modulation is encoded as the base layer. So, the constellation points generated in the left side of the Q-plane always yields a value of 0, and the right side of the Q-plane is always decoded as a 1. This phenomenon is the same as BPSK modulation. At the receiver side, the base layer of **GR1** can be simply decoded as a BPSK packet, i.e., any received sample to the right side of the Q-plane is mapped to bit 1, and a point to the left side will yield a bit 0.

Figure 7.3(b) shows the demodulation pipeline for a GRaTIS compatible node. The GRaTIS layer is first decoded as the constellation from which the GRaTIS cluster has been derived (R_m). Then, the bits designated for the GRaTIS layer are extracted using the **GRaTIS rate** information, to form the bit stream for the second packet (b_g) at rate R_g. The rest of the receiver decode pipe remains unchanged. For example, if a packet has been transmitted using **GR2**, then it is decoded as 16QAM packet, to yield 4 bits. To extract the GRaTIS bits, the receiver extracts 2 bits, in this case $b_1 b_3$. In this way, with the knowledge of the proper constellation mapping, we can decode the extra packet while being completely backward compatible with a legacy node, which is unaware of any GRaTIS layer transmission.

7.1.3 Medium Access Control for GRaTIS

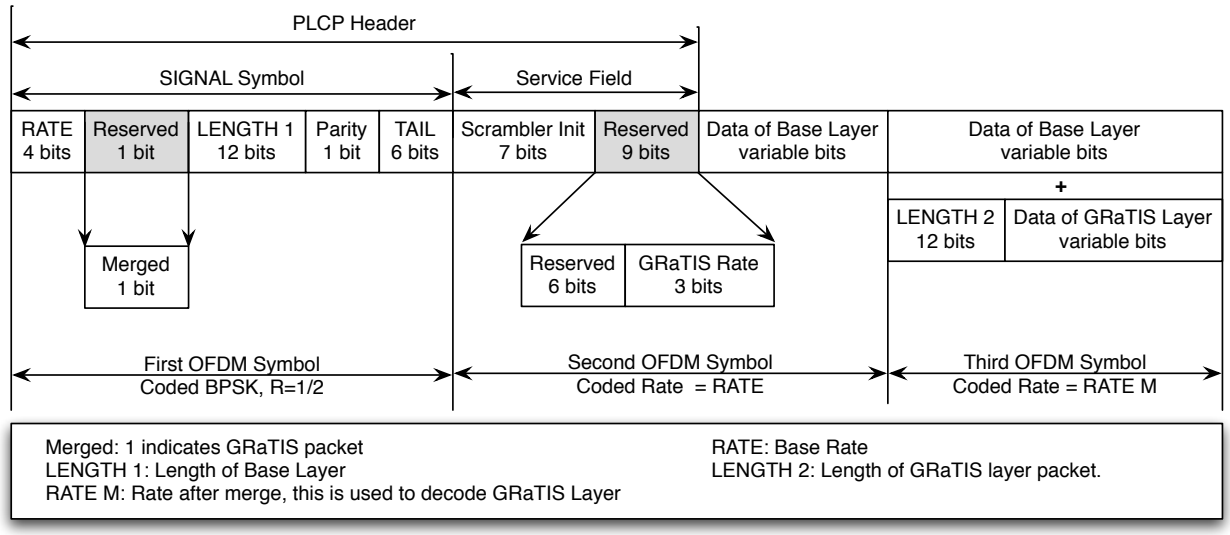


Figure 7.4: Modified 802.11a/g PLCP header – shaded fields indicate modifications to support GRaTIS.

To successfully decode a GRaTIS packet, a node needs to know the encoding information of the packet. We use the reserved bits of the PLCP Header of IEEE 802.11a/g packet to provide the encoding information as shown in figure 7.4. The 1-bit **reserved** field in the **SIGNAL** symbol is used to denote whether the packet contains a GRaTIS layer. The RATE field indicates the modulation rate at which the base layer is encoded. We use 3 bits out of 9 reserved bits in the **Service** field to indicate the rate of the GRaTIS layer encoding, which can have values from 1 to 6. As in any unmerged packet of IEEE 802.11a/g,

the first symbol is modulated in BPSK with $1/2$ rate coding. The second symbol is modulated in the rate specified in the **RATE** field of SIGNAL symbol. In this symbol, GRaTIS layer information is embedded, which is used to demodulate from the third OFDM symbol onwards, which marks the beginning of the data payload. The first 12 bits of the third symbol in the GRaTIS layer carries the length of the GRaTIS packet that is used to decode the second packet.

The encoding procedure ensures that decoding of the base layer is exactly the same as decoding any generic 802.11a/g packet. The demodulation for the GRaTIS layer changes from the third symbol onwards based on the information received in the 'GRaTIS rate' field in the second symbol of the packet, as shown in figure 7.4. Based on the GRaTIS layer decoding capability of the clients, an AP can decide which group rate to use to merge packets. An AP will never merge packets of a GRaTIS incompatible client in the GRaTIS layer.

In the 802.11a/g PHY layer, each message must be individually acknowledged. One major hindrance in using multiuser communications such as GRaTIS is the need for those acknowledgments. For this we rely on a **simultaneous acknowledge mechanism** (SMACK) [64] to gather acknowledgments from multiple recipients of the merged packet. SMACK will reduce the overhead of scheduling multiple acknowledgment packets and will reduce the multi-party acknowledgment time. The AP can schedule acknowledgments for the clients that cannot transmit SMACK.

7.1.4 GRaTIS as a Facilitator

Recent research on accurate channel prediction [60, 61, 65] has enabled finer, more accurate control for the correct data rate for a link. GRaTIS will benefit from these channel estimation techniques, which will essentially help to make correct decision on GRaTIS rate selection, such that there is minimum packet loss and packet re-transmissions are minimized in the network.

In recent years, there has been encouraging work in the physical layer of 802.11a/g, yielding higher aggregate network throughput. GRaTIS is orthogonal to many of these technologies, which can be applied in conjunction with GRaTIS to improve the overall performance of the network. Wireless network coding [63, 66] can be used with GRaTIS in 802.11a/g wireless network to decrease the number of packet transmissions

in a network. A frequency aware rate selection [23] mechanism can be applied for GRaTIS rates to transmit lower data rates in frequency selective fading scenarios.

Although we have implemented GRaTIS in a 802.11a/g network, it can be implemented in any wireless or wired network that uses modulation to encode packets. We have implemented and experimented with the protocol in OFDM-based system, and it can be readily applied to any other single carrier or OFDM based wired or wireless protocol. In this work, we have shown how the merging information can be disseminated in a WiFi-based network. However, to extrapolate this technique in any other protocols, like WiMax or LTE, new methods to disseminate this information is required, and is out of scope of this work. In the cognitive radio domain, non-contiguous OFDM transmissions will be a necessity, where GRaTIS can be efficiently utilized in the subcarriers chosen for cognitive transmission.

7.2 GRaTIS: Rate Analysis

In this section we evaluate the bit error rate (BER) and packet error rate (PER) for various group rates in GRaTIS, in the presence of Additive White Gaussian Noise (AWGN) using the standard techniques used to analyze digital communication performance. At the receiver, the noisy constellation points in the I/Q plane are mapped to corresponding bits by using maximum likelihood (ML) decoding. Constellation points are required to be within an area in the I/Q-plane defined by a modulation dependent decision boundary to ensure error free decoding. The BER for such a scheme is given by eq. 7.1.

$$P_B(E) = \frac{1}{2}\,\mathrm{erfc}\left(\sqrt{\frac{\Delta_E(i,j)}{4N_0}}\right) \tag{7.1}$$

The bit error rate for an arbitrary modulation scheme and ML decoding boundaries is upper bounded by eq. 7.2, where,

$$P_{WUB}(E) = \sum_{j=1}^{M-1}\sum_{i\neq j}\frac{1}{2M}\,\mathrm{erfc}\left(\sqrt{\frac{\Delta_E(i,j)}{4N_0}}\right) \tag{7.2}$$

$$= \sum_{k=1}^{N}\frac{A_d(k)}{2M}\,\mathrm{erfc}\left(\sqrt{\frac{\Delta_E(k)}{4N_0}}\right) \tag{7.3}$$

where,

– $\Delta_E(i,j)$ = Squared Euclidean distance between two distinct constellation points i and j.

– N = Possible different squared Euclidean distances in the *decoded* constellation, where $N \leq M(M-1)/2$.

– M = Total number of *decoded* constellation points.

– N_0 = Additive white noise power.

– $\Delta_E(k)$ = Distinct pairwise Euclidean distance in the *decoded* constellation.

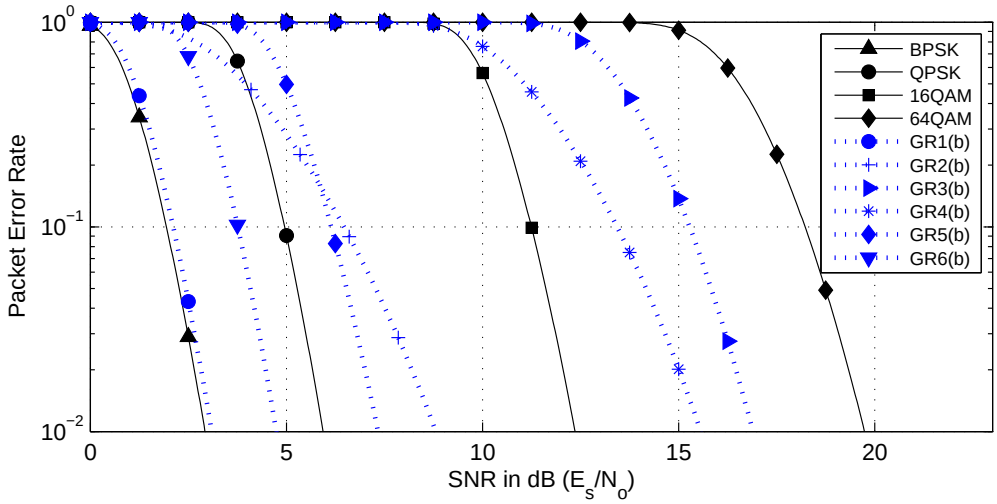
– $A_d(k)$ = Number of signal pairs having squared Euclidean distance of $\Delta_E(k)$.

Using eq. 7.2 we can compute the BER for any constellation and ML decision boundary. An example BER computation for the base layer of **GR2** has been shown in §7.3. The PER for a packet size of 128 bytes after Viterbi decoding is computed using the maximum free distance for a particular coding rate and its corresponding distance spectrum [67, 68, 69, 70]. The theoretical PER plots for the Base layer and GRaTIS layer is shown in figure 7.5 and figure 7.6 respectively.

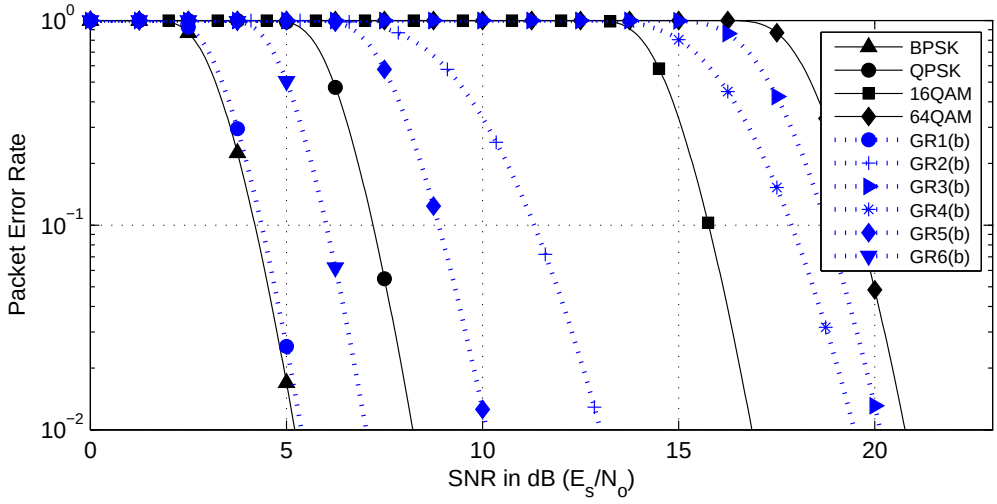
Table 7.1: Throughput and SNR requirements for 802.11a/g and GRaTIS rates

Mod	CR[1]	Data Rate (Mbps)		SNR(dB)			
				Base		GRaTIS	
		Link	Grp	Th[2]	Exp	Th[2]	Exp
BPSK	1/2	6	n/a[3]	3.0	4.5	n/a[3]	n/a[3]
BPSK	3/4	9		5.0	6.0		
QPSK	1/2	12		6.0	7.0		
QPSK	3/4	18		8.0	9.0		
16QAM	1/2	24		12.5	13.0		
16QAM	3/4	36		17.0	18.0		
64QAM	2/3	48		19.5	24.0		
64QAM	3/4	54		21.0	26.0		
GR1	1/2	6	12	3.5	5.0	17.0	17.0
GR6	1/2	6	18	4.5	5.0	18.5	21.0
GR5	1/2	12	24	7.5	7.5	18.5	23.0
GR2	1/2	12	24	9.0	10.5	11.5	15.0
GR4	1/2	12	36	15.0	15.0	19.0	23.0
GR3	1/2	24	36	17.0	16.5	20.0	25.0
GR1	3/4	9	18	5.5	n/i[4]	19.5	n/i[4]
GR6	3/4	9	27	7.0		20.5	
GR5	3/4	18	36	10.0		20.5	
GR2	3/4	18	36	13.0		14.5	
GR4	3/4	18	54	19.5		21.5	
GR3	3/4	36	54	20.0		22.0	

[1]Coding Rate, [2]Theoretical, [3]Not Applicable, [4]Not Implemented



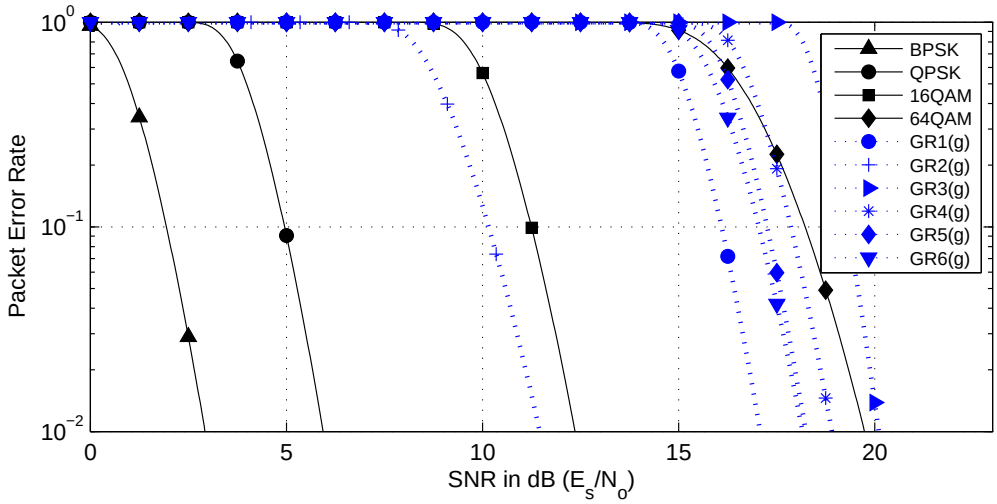
(a) PER for coding rate = $1/2$ (Except $64QAM = 2/3$)



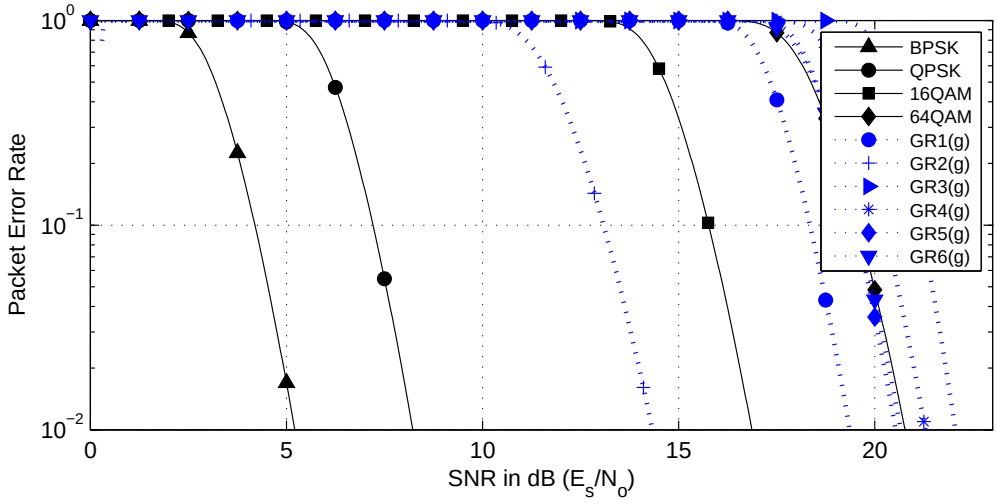
(b) PER for coding rate = $3/4$

Figure 7.5: PER for Base layer compared to legacy 802.11a/g modulations.

In the BER computation we consider that the GRaTIS constellations are mapped using Gray code [45] and encoded using a $1/2$ rate (except for 64QAM, which is encoded using a $2/3$ rate) as well as $3/4$ rate convolution code. This analysis has been done to ascertain the operating range of different group rates and their potential benefits when used to merge packets using GRaTIS. The BER performance for the GRaTIS layer is as important as the base layer because it allows the MAC to identify users with suitable SNR that can decode the bits from the enhanced GRaTIS layer. The BER computation for the GRaTIS layer is similar to that of the base layer and can be easily calculated using eq. 7.2. We discuss the performance of both



(a) PER for coding rate = $1/2$ (Except $64QAM = 2/3$)



(b) PER for coding rate = $3/4$

Figure 7.6: PER for GRaTIS layer compared to legacy 802.11a/g modulations.

the layers using a testbed in §7.4.2. The theoretical and testbed results are listed in Table 7.1, which shows the SNR requirements for a 2% packet error rate (PER) along with the SNR requirement for the 802.11a/g standard rates. The SNR - throughput relationship for various group rates are used as a look-up while downlink packets are being considered to be merged. As shown in Table 7.1, the group rates provide a variety of step-down rates while utilizing SNR diversity in the network to increase the aggregate throughput of the network.

From figure 7.5, we find that the base layer of **GR1** and **GR6** requires a SNR between QPSK and

BPSK. Hence these group rates can be used to modulate signals for nodes whose SNR are lower than that required by QPSK. Similarly, **GR6** and **GR2** offer two step down rates for nodes not reachable with 16QAM but having higher SNR than QPSK. While **GR3** offers similar flexibility by stepping down to an intermediate data rate instead of 16QAM, the benefits from **GR4** can be seen when used in conjunction with the GRaTIS layer: providing a combined data rate equal to that of 64QAM, which none of the two nodes would have been able to achieve with independent packet transmissions.

7.3 Example BER calculation

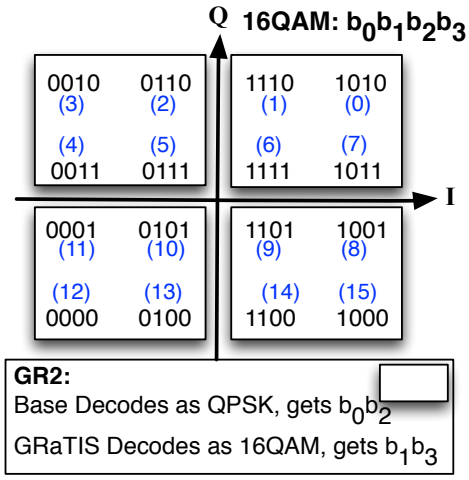


Figure 7.7: Decoding boundaries for *GR2* Base layer.

In this section we present an example of BER calculation for an overlapped modulated packet. In GRaTIS the transmitted constellation is decoded using a different set of decision boundaries, which yields a constellation of smaller size. For example, a transmitted constellation of 16QAM can be decoded as a QPSK as done in **GR2** base layer. In such a case the BER for the base layer will be between QPSK and 16QAM. For a 16QAM constellation the possible constellation point sets are $\{\pm A, \pm A\}$, $\{\pm 3A, \pm A\}$, $\{\pm A, \pm 3A\}$ and $\{\pm 3A, \pm A\}$. A is a modulation dependent parameter given by $\sqrt{\frac{K_b E_b}{K_{mod}}}$, where K_b is the number of bits per symbol, E_b is the energy per bit and K_{mod} is a scaling factor so that all constellation points have unit energy: for $16QAM$ this scaling factor is $\sqrt{10}$ [45]. Thus $A = \sqrt{\frac{K_b E_b}{10}} = \sqrt{\frac{E_s}{10}}$, where E_s is the energy

per symbol.

In figure 7.7, we revisit the constellation diagram and the decision boundaries for *GR2* Base layer. Here, a 16QAM constellation is decoded using the decision boundaries for QPSK and the constellation points are marked $[0 \dots 15]$. We recall from §7.1, that for the base layer a jump from one signal point to another within a quadrant will not cause any bit error since all the points in one quandrant are mapped to one QPSK constellation point. Therefore, we start by identifying the pair-wise squared Euclidean distances between transmitted constellation points that will cause a bit error.

$$\Delta_E(0,3) = 36A^2 \tag{7.4}$$

$$\Delta_E(0,15) = 36A^2 \tag{7.5}$$

$$\Delta_E(1,2) = 4A^2 = \Delta_E(7,8) \tag{7.6}$$

$$\Delta_E(1,14) = 36A^2 = \Delta_E(7,4) \tag{7.7}$$

$$\Delta_E(6,5) = 4A^2 \tag{7.8}$$

$$\Delta_E(6,9) = 4A^2 \tag{7.9}$$

Now, we can compute the conditional error probabilities of the points in the top-right quandrant **viz.** $0, 1, 6, 7$ using eq. 7.1 as follows,

$$\begin{aligned} P_{WUB}(E|\vec{I}=0) &= \frac{1}{2}\operatorname{erfc}\left(\sqrt{\frac{\Delta_E(0,3)}{4N_0}}\right) \\ &\quad + \frac{1}{2}\operatorname{erfc}\left(\sqrt{\frac{\Delta_E(0,15)}{4N_0}}\right) \\ &= \operatorname{erfc}\left(\sqrt{\frac{9A^2}{N_0}}\right) \end{aligned} \tag{7.10}$$

$$
\begin{aligned}
P_{WUB}(E|\vec{I}=1) &= \frac{1}{2}\,\mathrm{erfc}\left(\sqrt{\frac{\Delta_E(1,2)}{4N_0}}\right) \\
&\quad + \frac{1}{2}\,\mathrm{erfc}\left(\sqrt{\frac{\Delta_E(1,14)}{4N_0}}\right) \\
&= \frac{1}{2}\,\mathrm{erfc}\left(\sqrt{\frac{A^2}{N_0}}\right) + \frac{1}{2}\,\mathrm{erfc}\left(\sqrt{\frac{9A^2}{N_0}}\right)
\end{aligned} \tag{7.11}
$$

$$
\begin{aligned}
P_{WUB}(E|\vec{I}=6) &= \frac{1}{2}\,\mathrm{erfc}\left(\sqrt{\frac{\Delta_E(6,5)}{4N_0}}\right) \\
&\quad + \frac{1}{2}\,\mathrm{erfc}\left(\sqrt{\frac{\Delta_E(6,9)}{4N_0}}\right) \\
&= \mathrm{erfc}\left(\sqrt{\frac{A^2}{N_0}}\right)
\end{aligned} \tag{7.12}
$$

Combining eq. 7.10, 7.11 and 7.12 we get,

$$
\begin{aligned}
P_{WUB}(E) &= \frac{4}{16}P_{WUB}(E|\vec{I}=0) + \frac{8}{16}P_{WUB}(E|\vec{I}=1) \\
&\quad + \frac{4}{16}P_{WUB}(E|\vec{I}=6) \\
&= \frac{1}{2}\,\mathrm{erfc}\left(\sqrt{\frac{9A^2}{N_0}}\right) + \frac{1}{2}\,\mathrm{erfc}\left(\sqrt{\frac{A^2}{N_0}}\right)
\end{aligned} \tag{7.13}
$$

Substituting the value of A,

$$
P_{WUB}(E) = \frac{1}{2}\,\mathrm{erfc}\left(\sqrt{\frac{9E_s}{10N_0}}\right) + \frac{1}{2}\,\mathrm{erfc}\left(\sqrt{\frac{E_s}{10N_0}}\right) \tag{7.14}
$$

Similar BER can be obtained for any constellation and any decision boundary. We compute the theoretical PER using these results and plot against increasing symbol to noise ratio (E_s/N_0) in figure 7.5 and figure 7.6.

7.4 Implementation and Evaluation

In this section, we discuss the details of implementing GRaTIS in a reconfigurable radio and the method of evaluation with over-the-air packets, transmitted from the radio in a small testbed scenario in an indoor environment.

7.4.1 Implementing GRaTIS

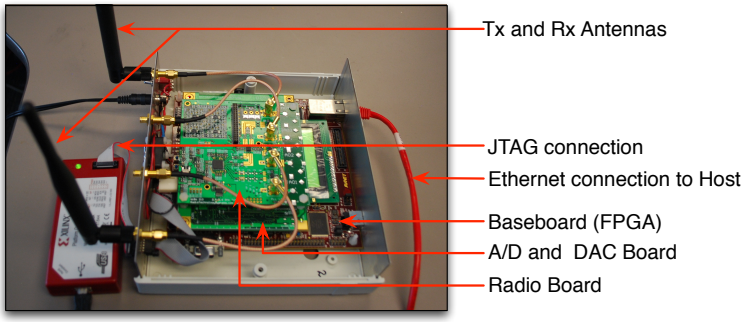
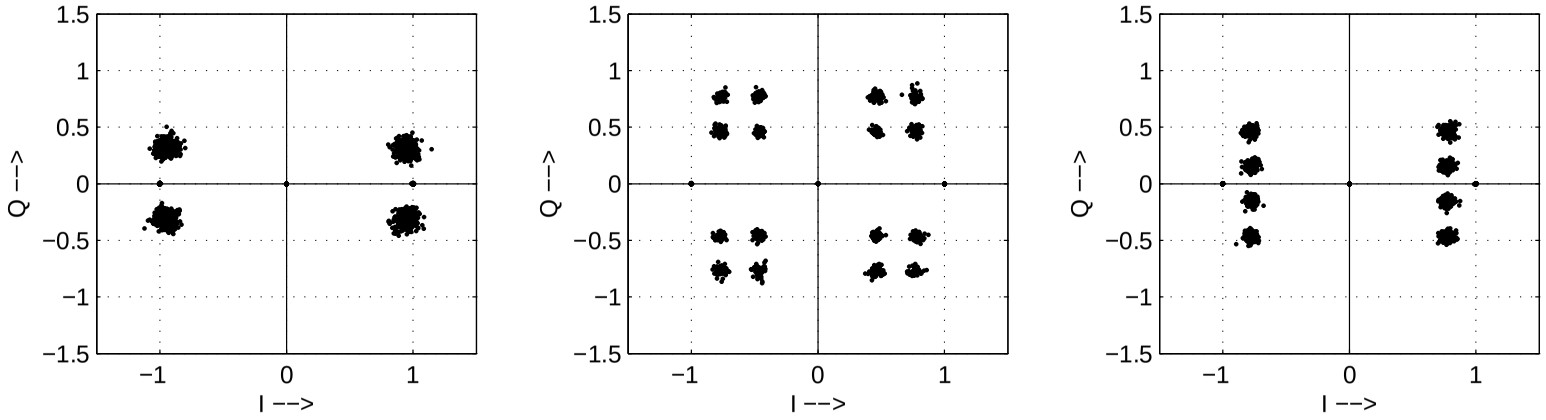


Figure 7.8: SDR platform used to implement GRaTIS.

The very nature of the technique used in GRaTIS requires minimal changes in a traditional 802.11a/g OFDM based radio pipeline. As discussed in §7.1, the encoding and decoding of the GRaTIS constellations can be easily performed in software by mapping the data bits to the target constellation points. If the proper compound symbols are provided to the baseband modulator, it would produce the target GRaTIS constellation without any requirement to change the I/Q vector lengths to achieve a certain error performance. Keeping the baseband modulator unaltered makes GRaTIS backward compatible and easy to implement in commodity hardware.

Although most of the encoding can be done in software, access to the packetization engine of the MAC layer is required. Since, this abstraction layer is not available to us from commodity hardware, we implemented this technique using a prototype hardware based on previous work [11, 14, 28], as shown in figure 7.8. The prototype uses a hybrid Software Defined Radio (SDR) based on a Virtex-5 FPGA that can transmit and receive generic 802.11a/g data packets [45]. Using such a platform we can have access to the packetization layer of the MAC where the data bits from two users are combined to form a symbol in the I/Q plane, which is accomplished by the **bit mapper** unit shown in 8.6(a). Thus, we can selectively transmit only the I/Q samples that correspond to a GRaTIS cluster while others remain unused.

Similarly, the receiver of the base layer can be oblivious of any enhanced layer. Only receivers capable of decoding the GRaTIS layer require additional controls to decode packets. Depending on the GRaTIS rate used to encode the second packet, the demodulator extracts the additional bits: typically termed as



(a) GR1 (b) GR5 (c) GR6

Figure 7.9: Various GRaTIS constellations transmitted using the SDR prototype.

bit slicing or **bit bashing**. Figure 7.3(b) shows the basic structure of the modified demodulator. The **bit basher** unit is responsible for separating the bits of the GRaTIS layer. Many of these controls govern how the required bits are sliced from the combined constellation symbols, which is again a bit level operation that can be easily done by simple software controls to the underlying hardware as a part of the MAC de-packetization layer. Other receiver subsystems prior to the demodulator, **e.g.**, the synchronizer and the equalizer, remain unchanged for implementing GRaTIS. The equalizer always aims to restore the original transmitted constellation, while it is the decoder that decides either to interpret it as a standard packet or a combined packet using GRaTIS. Apart from decoding packets, the receiver also reports the average receive SNR of a packet and also performs MAC CRC checks in the hardware to measure packet loss.

Figure 7.9 shows the various constellations produced by the prototype SDR that support multiuser communication using GRaTIS. Figure 7.9(a) shows the constellation for **GR1**, which is derived from a 16QAM constellation by not using the other constellation points. Similarly, figure 7.9(b) and 7.9(c) shows a modified 64QAM constellation that provides a combined network throughput of 4 bits/OFDM subcarrier (2 for base and 2 bits for GRaTIS layer) and 3 bit/OFDM subcarrier (1 for base and 2 bits for GRaTIS layer) respectively.

As mentioned, the receiver uses the standard OFDM demodulation techniques and largely the same MAC layer. Although the hardware chain is unchanged, the software controlling the binary operations that follow the demodulator do need to be updated; this is usually a software upgrade. This is a distinct advantage over other multiuser decoding techniques that rely heavily on complex signal processing algorithms, involving custom constellations and more general control of the I-Q mapping. Hence, GRaTIS provides an example of harnessing the power of existing resources while innovating new and improved protocols.

7.4.2 GRaTIS: Putting it to Work

In this section, we utilize our hardware prototype transceiver to measure the actual SNR required for satisfactory operation in a typical indoor wireless network. A testbed has been setup with three nodes, one transmitter and two receivers in an indoor laboratory environment. The nodes are placed at a distance of approximately 5m on work desks in enclosed cubicles. Channel quality and SNR variation is obtained by

controlling the transmit power as well the location of the receiver nodes relative to the transmitter. The average throughput and average SNR required for 2% PER across different node arrangements have been computed. The group rates proposed in §7.1, with $1/2$ rate convolution coding for both the layers have been compared to the standard rates available in $802.11a/g$.

The SNR is computed from the digital I/Q samples in the hardware. Power is measured as $|r(n)|^2$, where $r(n)$ is the complex signal samples. P_{s+n} denote the power of the signal and the noise combined, averaged over 5 OFDM symbol period, after a packet is detected and the MAC CRC is received correctly. So, it reflects the average power over the data symbols of the OFDM packet. The noise power, P_n, is time averaged over 5 OFDM symbols after the packet is completely decoded. Both of these values are computed in the hardware and sent to the user along every data packet. The SNR in dB is then computed from these values using $10\log_{10}((P_{s+n} - P_n)/P_n)$.

Figure 7.10 shows the performance of the base and GRaTIS layer of six group rates, along with standard modulations, BPSK, QPSK, 16QAM and 64QAM. For each modulation, we plot the physical layer throughput and mark the minimum SNR required for a PER of 2%. Maintaining acceptable error rates while maximizing throughput is important, as it might lead to unwanted re-transmissions consuming additional airtime and reducing the throughput of the network. These marks are the SNR requirements for each modulation below which it cannot be used reliably. The base and GRaTIS layers are denoted by suffixes '(b)' and '(g)' respectively.

GR1(b) and **GR6(b)** are two group rates that provide BPSK rate, and have an SNR requirement between BPSK and QPSK. Thus, when a node becomes unreachable in QPSK, we can use these two group rates to combine packets, and send a packet to another node reachable at higher SNR: at BPSK rate to node reachable at 17dB or higher using **GR1(g)** and at QPSK rate to a node reachable at 21dB or higher if using **GR6(g)**. Although the GRaTIS layer needs a higher SNR to operate, which could potentially receive a higher rate packet, the collective throughput by combining the two packets is more than throughput obtained by transmission of two individual packets. The gain in throughput can be computed as described in §7.1.

GR2 and **GR5** provides a QPSK data rate to both of the layers, and have SNR requirements in between QPSK and 16QAM. The presence of these two group rates provides more flexibility to choose the

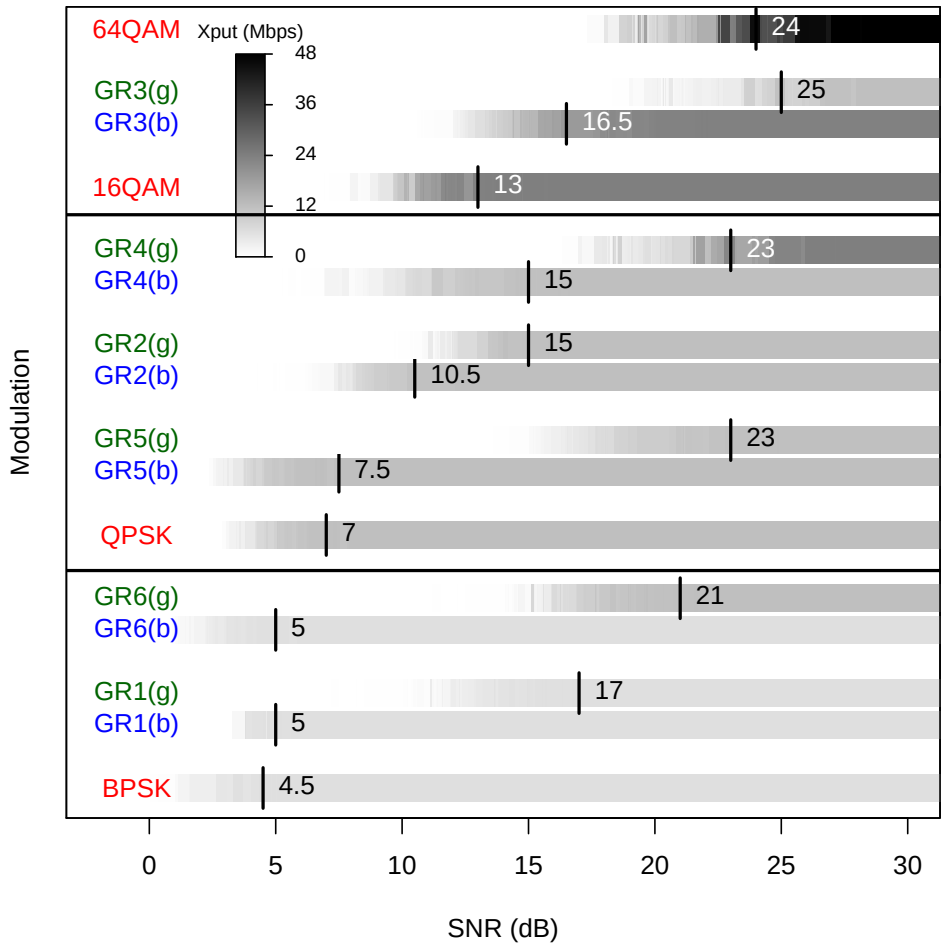


Figure 7.10: Link throughput of GRaTIS((b)-Base, (g)-GRaTIS) and 802.11a/g rates with increasing SNR. The rates are grouped according to the increasing base rate. The numbers against each rate denote the SNR required to decode a 128 byte, $1/2$ rate convolution coded packet with 2% PER.

GRaTIS rate in a wider range of SNR. If SNR of a node is more than 7.5dB but less than 10.5dB, and that of another node is more than 23.5dB, **GR5** can be used effectively, but **GR2** cannot be used in this SNR range. Similarly, if a node has SNR between 10.5dB and 13dB, while another near node has SNR between $15dB$ and 23.5dB, we cannot use **GR5** for the near node, but can successfully encode the packets using **GR2**.

GR3(b) provides a data rate equal to that of 16QAM, and has an SNR requirement in between 16QAM and 64QAM. Evidently, this group rate can be used whenever the SNR of a node falls below the SNR requirement of 64QAM providing the best effort data rate, 16QAM in this case, for that node. **GR4(b)** provides throughput equal to that of QPSK, but requires more SNR than 16QAM to be decoded correctly. It might seem that this group rate does not provide the best effort rate to the base layer. Careful observations

reveals that **GR4(g)** provides a throughput equal to that of 16QAM, and has a lower SNR requirement than 64QAM. So, we can transmit a packet in **GR4(g)** to a node when its SNR falls below 64QAM, and **GR4(b)** then can be used to transmit any extra bits as a GRaTIS layer.

The SNR required to maintain a 2% PER for all the GRaTIS and 802.11a/g are listed in table 7.1. These experimental results are used to verify that the technique yields practical gains over individual packet transmissions and benefits a network with wide diversity of SNR.

7.5 GRaTIS: Practical Gains

Theoretical and experimental results in §7.2 and §7.4.2 respectively show the potential benefits of using GRaTIS in modern networks like 802.11a/g and WiMax. It is sometimes difficult to understand the benefits of a particular wireless optimization from PER plots and bench experiments. Also, testbed implementations, where the network is flooded with UDP packets of the same packet length do not represent a realistic scenario of wireless networks. We want to know -

- How often do stations have sufficient SNR diversity to exploit GRaTIS?
- Does GRaTIS gain in variable packet lengths of the users?
- How useful is the combined coding efficiency over 802.11a/g network?
- Does GRaTIS work in different scenarios, like a conference hall, university cafeteria or home network?

To determine the **system** benefits of GRaTIS, we analyzed captured packet traces from SIGCOMM 2008 dataset [1]. The SIGCOMM traces reported signal and noise power in the Prism header [71], which have been converted to report SNR in dB. We have also captured packet traces in the common areas of our university with more than 50 active users, and in a home network of two users. The captured packet traces are referred to as 'Lobby 1', 'Lobby 2', 'Cafe' and 'Home'. 'Lobby 1' and 'Lobby 2' traces have been captured at the exact same location in a common area to identify the temporal variation of traffic pattern. 'Cafe' denotes another trace, which is also a common area for dining services. 'Lobby 1', 'Lobby 2' and

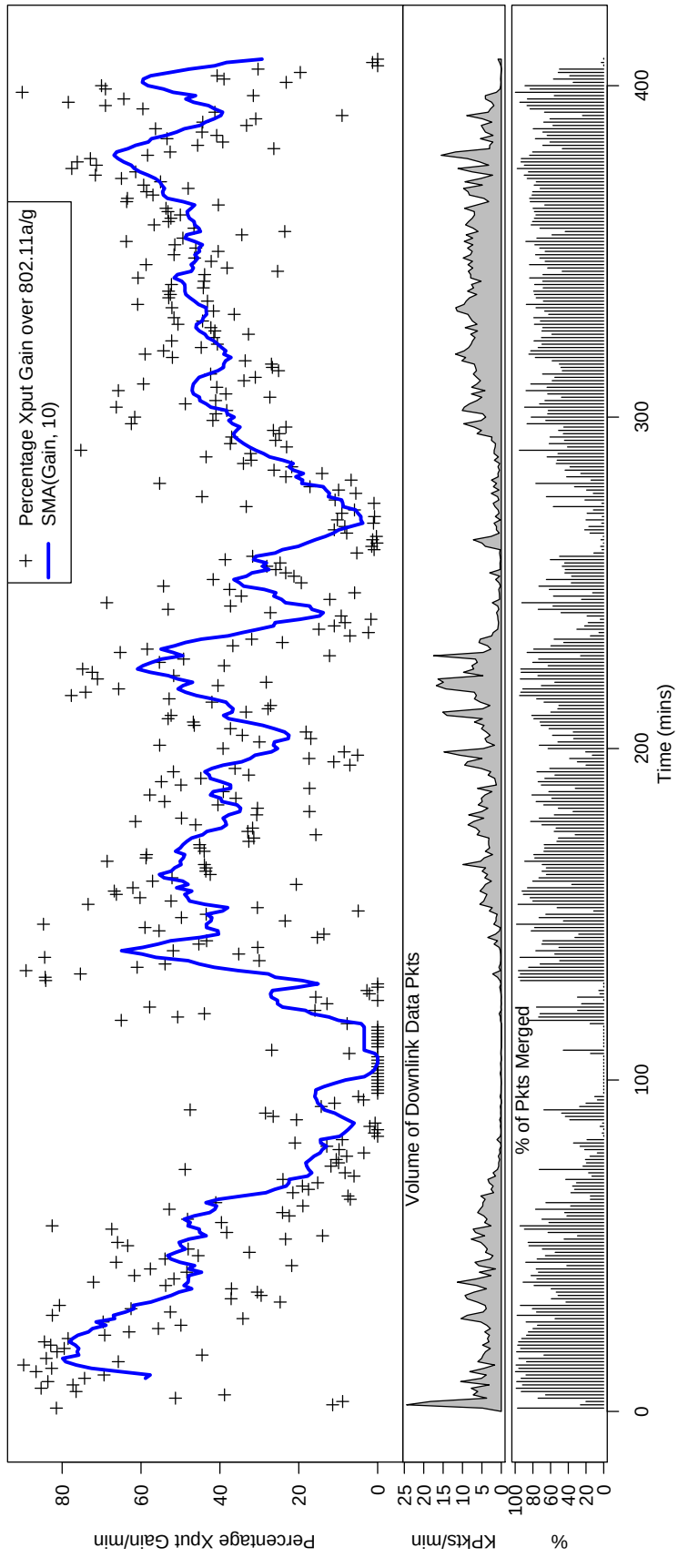


Figure 7.11: An example implementation of GRaTIS when applied on the downlink data packets in an infrastructure 802.11a/g network with > 50 users. The first plot shows the percentage gain in aggregate throughput per minute with GRaTIS. A 10 minute Simple Moving Average (SMA) of the gain in also shown. The second plot shows the volume of packets in Kpackets/min while the third plot shows the percentage of packets transmitted per minute using GRaTIS.

'Cafe' had more than 50 active users at any given time, while people walked through the area. Our captures use the Radiotap header [72], which also reports the signal and noise power, from which SNR has been computed.

We are interested in knowing the SNR variation in all the scenarios, to access whether GRaTIS can be used to improve the network performance. Benefits of this protocol are maximized when there are clients that are reachable at a wide variety of SNR facilitating packet combination at different GRaTIS rates. Figure 7.1(b) and 7.1(c) shows the SNR density variation of uplink data packets in the SIGCOMM and captured packet traces. Within the SIGCOMM traces, we have chosen 4 monitors that have the most data packets to generate the histogram of SNR. The histograms from all the monitors show similar distribution but the SNR is found to be evenly distributed within the set of clients. This presents good opportunity to merge packets that can be sent to clients reachable at different SNR levels. This similarity can be attributed to the confined nature of a conference room, where users were evenly distributed in the room. However, we notice 'Lobby 1' and 'Lobby 2' show significant variation in the histogram, with the maximum reaching around 20dB SNR. The trace for 'Cafe' shows two prominent spikes in the distribution, which are due to the spatial diversity of two very active users in the network. In the 'Home' trace, this diversity is more prominent as there were only two users in the network. All the scenarios show different diversity of the users, but undoubtedly enough variation in SNR to use GRaTIS.

We intend to use GRaTIS to merge downlink data packets, which constitutes most of the packet transmission over the air. The AP would be able to merge two packets intended for two clients, operating in different SNR regime, into one GRaTIS packet, if it has information of the SNR of the packets received from AP by each client. The traces did not provide the SNRs at which the clients are reachable from the AP (this requires packet monitoring at each client). However, the traces report a) the **SNRs** at which the monitor is reachable from each client, which gets updated whenever a client transmits a packet and is overheard by the monitor, and b) the actual downlink data packets transmitted by the AP, with the information of **packet size**. Now, if we place an AP at the monitor's location, it would use this information about the SNR to merge packets and transmit the same data packets as the AP is currently transmitting to its clients.

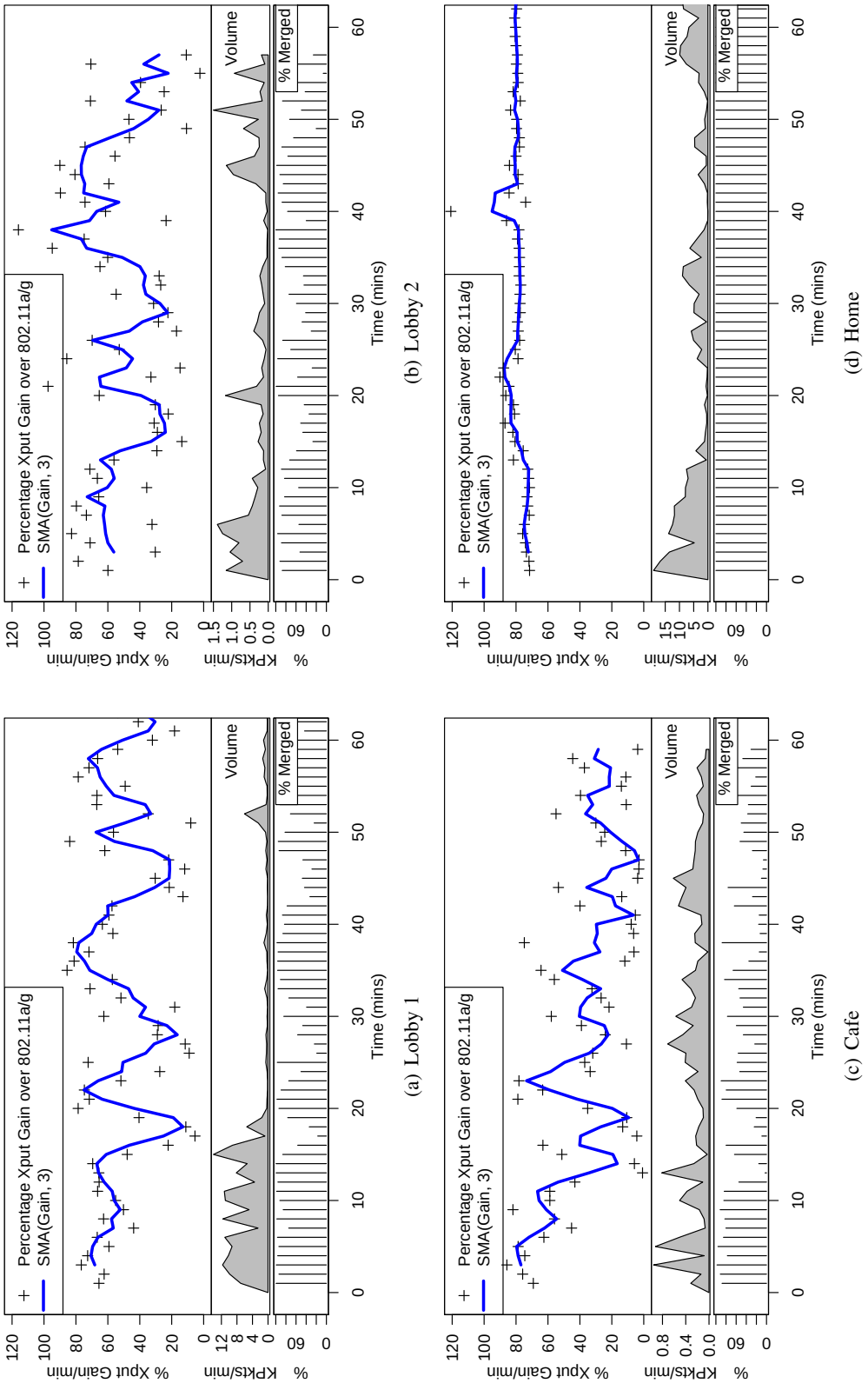


Figure 7.12: Gains of using GRaTIS in four different scenarios, based on captured packet traces.

We analyze the traces based on the monitor's view of the network, merging downlink data packets based on the SNRs received from the clients. Two downlink data packets are merged only if the time required to transmit a merged GRaTIS packet is less than the time required to transmit two individual packets using 802.11a/g. The queue length is finite, and only 10 packets have been considered to be available at any time for merging. Off-the-shelf wireless AP queue capacity varies from 39 to 337 packets [73]. In congested network scenarios with more packets available in the queue, there will be more options available, which will improve the performance of GRaTIS. To maintain low latency and fairness in the network, the merging algorithm always transmits the packet in the lowest position of the queue, and searches for any possible combination of GRaTIS among the next 9 packets that will reduce the overall time required for transmission. Often in our evaluation, merging algorithm could not find a suitable combination between the lowest packet in the queue and 9 packets above it, and at that time, the lowest packet in the queue is transmitted without any merging. The airtime requirement in 802.11a/g not only depends on the packet air-time, but also on the medium access time, which equals the DIFS time and a random back-off time. In our analysis, we used the SIGCOMM 2004 dataset [74] to estimate an average medium access time per packet, which equals $2730\mu s$. We consider this time for a single packet transmission for both 802.11a/g and GRaTIS. We used the experimental results, as reported in Table 7.1, to compute the airtime usage in both of the cases. Since we have not implemented $3/4$ coding rate for GRaTIS, we do not consider those in the trace analysis. With both the coding rates available, there will be more opportunities to merge packets, for example merging a packet of $3/4$ coding rate with another of $1/2$ coding rate. However, we do consider all the coding rates available for 802.11a/g. This is a conservative approach to show the improvement of using GRaTIS over 802.11a/g, and will give a lower bound on the possible gains.

We selected one random monitor on one of the days at SIGCOMM, Monitor 4 on Aug-19, and compute the throughput gain achieved per minute using our method. Figure 7.11 shows the temporal variation per minute of the percentage gain in throughput if GRaTIS is used. The volume of downlink data packets transmitted per minute, and the percentage of packets merged per minute is also shown. A simple moving average (SMA) of 10 data points shows that the average gain goes up to 80%, with instantaneous gain/min. goes up to 90% over 802.11a/g network. The gain is proportional to the percentage of packets being merged.

Most of the time, when there is network traffic, GRaTIS could merge $\approx 70\%$ of the packets. We notice that the volume of packets drops significantly at time 100 minutes, which is probably the lunch hour, and most of the packets observed by the monitor have very low SNR. GRaTIS did not find much opportunity to combine packets as there was limited variety in the SNR among the clients, and the volume of packets/minute. was extremely low. To ensure that we receive similar gains in other days using the trace from other monitors as well, we computed average gain in each day for each monitor. Results show consistent gains in other scenarios as well.

All networks are not the same, and the traffic pattern varies from one network to another. Hence, we performed similar analysis on our captured dataset to investigate whether GRaTIS can be beneficial in a variety of wireless network scenarios. Figure 7.12 shows the percentage gain in throughput per minute by using GRaTIS over 802.11a/g network. The trace for 'Lobby 1' shows very high volume for the first 20 mins, which reduces significantly later. However, average gain goes up to 80% when there are enough packets available in the queue to be merged. We captured the trace termed 'Lobby 2' just after completing the capture of 'Lobby 1'. There is significant difference in the volume of downlink data packets. However, GRaTIS still maintains 80% average gain in throughput, with instantaneous gain of up to 120%. We also find that $\approx 70\%$ of the packets have been merged. It is to be noted that GRaTIS will get some savings in DCF and contention times due to aggregating packets, but this gain for concatenating two frames is less than 20%, even in the best case scenario [75]. Nonetheless, aggregated frames can be GRaTIS frames as well, indicating that our technique has potential for even better throughput if we aggregate GRaTIS frames.

The trace termed 'Cafe' corresponds to a very dynamic scenario, where we notice multiple users logging onto the network, using it for few minutes and logging off. Similar to the detailed analysis of the SIGCOMM 2004 trace [29], our analysis shows that the AP spends most of the time in back-off, leading to an overall low throughput on the down-link. Even in this congested scenario, we notice an average gain of 80%. We believe that GRaTIS will perform better in this type of network to reduce the transmission time, and thus more time will be available for data transmission, which will increase individual throughput of any client in the network.

The home network scenario is another common use of 802.11a/g network, where often two or more users in a family share the same wireless network on a regular basis. They often have spatial diversity, leading to variation in SNR. We captured the trace in such a scenario with only two users. The SNR variation, as seen in figure 7.1(c) is not uniform, and there are only two very high densities in the histogram due to two users. GRaTIS successfully merged more than 99% of the packets on average. However, we notice that most of the time **Gr3** and **Gr4** were used due to the two users being in the operating range of these two GRaTIS rate. The average gain remained constant at 80% with highs of 120%.

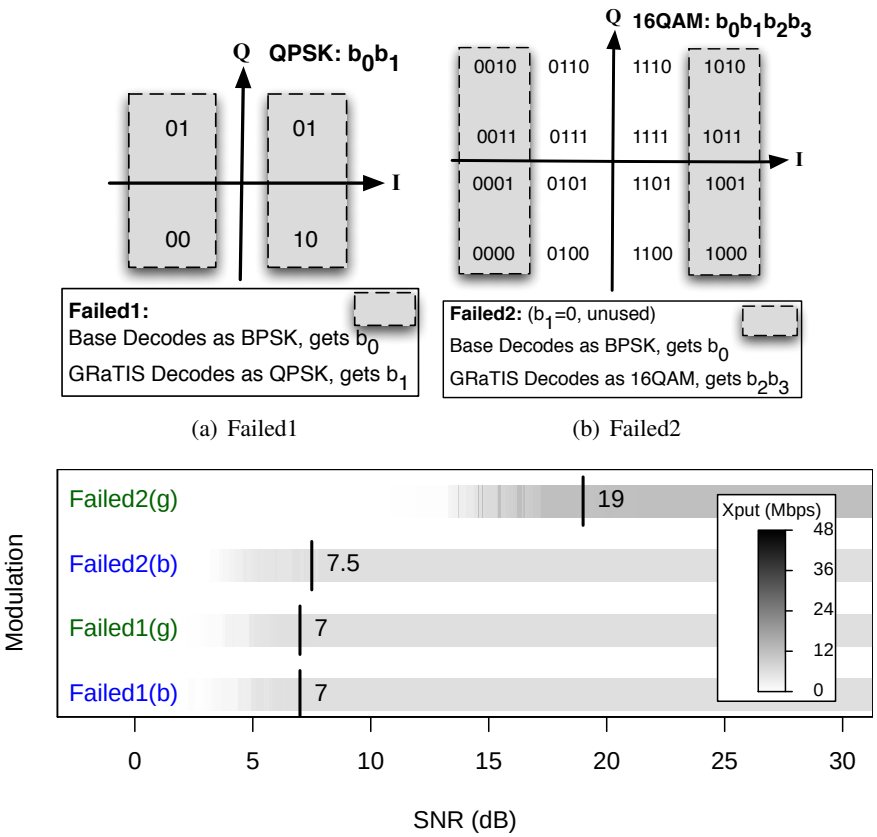
This analysis shows that when the clients SNR vary in a diverse range, GRaTIS can be used to combine packets and gain airtime, which can essentially be used to transmit more packets and increase the overall throughput of the network.

7.6 GRaTIS - A Non-trivial Solution

This work presents six different types of combinations to generate GRaTIS rates. However, an exhaustive search can be done to find more combinations for any constellation. It is to be noted that this search can be done one time and the results can be stored for reference. Also, a combination from a constellation can be extrapolated to it's corresponding higher constellation, making GRaTIS easy to incorporate for bigger constellations. For example, we can envision GR3 and GR4 for 64QAM as an extension of GR2 for 16QAM. In the same way, it can also be extrapolated for 256QAM as well. However, exhaustive evaluation is necessary to ensure the expected performance gains.

In this work, we show GRaTIS for two receivers, such that the base layer is transmitted to one receiver, while the GRaTIS layer is transmitted to another receiver. It is possible to generate multiple GRaTIS layers on top of a base layer to transmit to three or more receivers at the same time, as long as each constellation point denotes more than 2 bits. In other words, we cannot use QPSK to transmit to three receivers. Nonetheless, it requires careful examination and evaluation to determine whether the generated layers yield better performance.

This work presents GRaTIS but also opens up a new set of open research problems. While GRaTIS provides more flexibility by providing different intermediate rates and merging multiple packets to enhance



(a) Failed1 (b) Failed2

(c) Throughput of Failed1 and Failed2

Figure 7.13: Example of two "group rates" that don't result in improved performance, showing that group rates must be carefully designed.

the network throughput, the combinations cannot be chosen randomly to ensure better performance. Careful inspection has to be made before selecting any combination. We present two rates **Failed1** and **Failed2** shown in figure 7.13(a) and 7.13(b) respectively, chosen randomly from a QPSK and 16QAM constellations. The throughput and SNR requirement of these two GRaTIS rates are shown in figure 7.13(c). Clearly, the base layer of both failed rates provide a BPSK data rate but requires a SNR that can support QPSK data rates. Hence, these two group rates cannot be used to yield improved aggregate throughput as the base layer has a sub-optimal data rate.

In our implementation and analysis discussed in this work, we have used $1/2$ rate convolution coding. Considering a $3/4$ coding rate we can obtain a similar set of results but at presumably higher SNR, where the base layer and the GRaTIS layer will have a $3/4$ coding rate. By using all the basic rates and their

GRaTIS derivatives, we can have almost a linear relationship between SNR and achievable throughput, instead of the conventional step-wise discrete rate allocation. Therefore, from a MAC layer point of view, GRaTIS inspires us to delve into the modalities of a novel rate adaptation algorithm for wireless networks and compare it with related SNR-based rate adaptation algorithms [76, 77, 23, 65]. Also in wireless mesh networks we can forward packets to multiple users from a common point (router/relay node) using GRaTIS. This is particularly helpful when the involved node-pairs cannot over-hear each other's transmission and hence cannot use network coding. We also intend to explore the possibilities of using GRaTIS to improve protocols of higher layers such as opportunistic routing algorithms.

7.7 Comparing with Superposition Coding

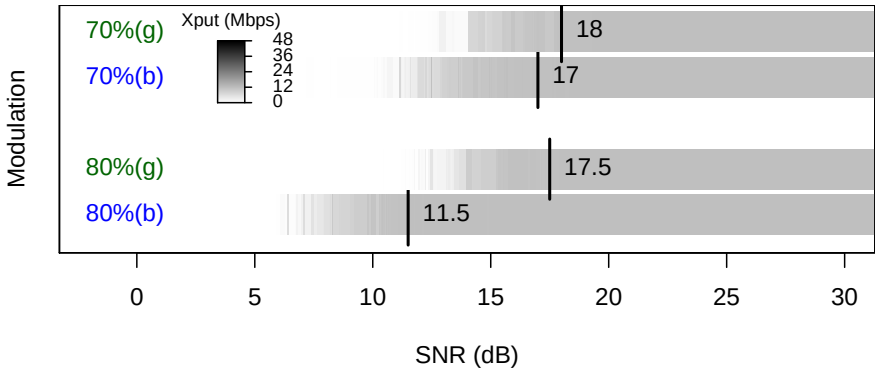


Figure 7.14: Over-the-air link throughput for superposition coding with 70% and 80% of total energy allocated to the far node. The numbers against each rate denote the SNR required to decode a 128 byte, $1/2$ rate convolution coded packet with 2% PER.

GRaTIS is distinctly different from Superposition Coding (SC) in many ways. SC merges two packets by superimposing one signal on another, which results in mapping the bits in such a way that an error in one layer introduces error in another. GRaTIS uses the bits of existing Gray Code to merge two packets in such a way that the bits of the GRaTIS layer are protected from errors due to symbols crossing either axes due to channel noise. In figure 7.2(a), it is evident that bits $b_1 b_3$, which encodes the GRaTIS layer of **GR2**, do not change while crossing axes. To compare with our results, we have also implemented SC using our hardware and measured the throughput and PER with the same radio prototype used to evaluate

GRaTIS. The SNR requirement for the two layers is shown in figure 7.14 for 70% and 80% of total energy allocated to the far node. These two modes of SC are the most commonly used energy allocation ratio used in SC [78, 79]. We find from table 7.1 that **GR2** of GRaTIS outperforms 'SC-70%' in terms of minimum SNR requirement by 6.5dB for the base layer and 3dB for the enhanced or the GRaTIS layer. However, we noticed that if the energy allocation ratio is $80:20$ to the two layers, the mapping thus generated resembles that of 16QAM, which is used to encode **GR2**. Even using the same mapping we notice that **GR2** performs better than 'SC-80%'. This noticeable improvement of GRaTIS over SC is attributed to the unique mapping technique of GRaTIS.

From an implementation stand-point, implementing SC at the transmitter requires changing the I/Q vectors, which needs a high degree of programmability in the radio. Also, the receiver has to perform Successive Interference Cancellation (SIC) to receive the second layer. On the other hand, GRaTIS involves mapping data bits to desired sub-constellations and hence can use the same I/Q vector configurations of the standard 802.11a/g modulation system. It is easier to implement (combine bits) in the existing transmitter pipeline than to super-impose signals. At the receiver end, no complex signal processing algorithms are required – only digital bit manipulation.

GRaTIS provides more combination options compared to SC, provides a fine-grained control over the SNR-throughput space and easily implementable, while outperforming Superposition Coding.

7.8 Related Work

In this research we discuss a form of multiuser communication that utilizes information embedded within a modulation constellation. We compare this scheme to other multiuser communications methods currently used in wireless networks. The transmitted signal contains bits for two users and each user can receive, equalize and decode exactly in the same way as done in conventional 802.11a/g packets. We also utilize the existing constellations in 802.11a/g and reinterpret them to encode multiple packets in one data stream. This ensures co-existence with commercial WiFi technologies.

Multiuser Detection: Our implementation and analysis of GRaTIS uses Orthogonal Frequency Division Multiplexing (OFDM), although it is not dependent on OFDM. Simultaneous transmissions can also be

detected by the multi-user detection scheme in CDMA. However, in CDMA networks, this technique is expensive to implement and is only implemented by the CDMA base-station and not the mobile devices. The technique requires capturing the waveform data and successively subtracting the signal of individual transmitters [80]. Transmitters use an elaborate power control mechanism to insure the signal received by the base station has a similar SNR for each transmitter. This adds to the latency of CDMA data networks. To reduce the complexity in correlating for different user codes, the authors in [52] use various heuristic methods to obtain a sub-optimal solution. In contrast to these multiuser techniques, GRaTIS requires no control over different users and does not suffer from error propagation due to channel estimation error. While SNR diversity is detrimental to other multiuser techniques, our scheme actually use the difference in SNR experienced by multiple receivers to encode additional data.

Multiuser communication using Orthogonal Frequency Division Multiple Access (OFDMA) is also prevalent in WiMAX [46] technology. OFDMA has also been introduced in cellular networks as a simultaneous communication mechanism, where separate contiguous sets of subcarriers are assigned for carrying multiuser data [49, 50]. Packets are decoded by multiple users by using a 2-dimensional map in time and frequency. In this way a significant amount of time is saved in contending for the shared wireless channel. However, GRaTIS can still be useful for multi-user systems such as WiMAX although our implementation is based on WiFi. In both WiMAX and WiFi, low rate modulation schemes are used to transmit data to stations with poor SNR; our scheme simply uses varying SNR experienced by multiple receivers to encode additional data.

Hierarchical Coding: Another domain where constellations are used to transmit to multiple users is digital video broadcast. The authors in [81] discuss a layered modulation technique, often termed as **hierarchical modulation** or **multiresolution modulation**. The higher order bits in a dense constellation are used to decode a poorer quality signal that can be decoded by a user with low SNR while users with high SNR will be able to decode all the bits in the symbol. In the DVB digital television standard, 16QAM and QPSK are used for hierarchical modulation. Most of the work involving hierarchical modulation, including [82] and [83], finds its application in a multicast or broadcast environment of digital video.

Hierarchical modulation is used for transmitting the same information (**e.g.**, video signals) to multiple

users. GRaTIS can transmit completely different information to multiple users with similar or greater reliability. While prior work in related fields calls for optimizing the signal structure and the signal constellations, our effort is focused on harnessing the strengths embedded in unexplored areas of existing specifications, which makes it novel and yet compatible with coexisting technologies. Also, the wide range of operation of our method (higher number of transmission modes over a range of SNR) puts our scheme in a unique position among its peers and predecessors.

Superposition Coding and Network Coding: Other packet mixing techniques in wireless networks either employ superposition coding [84, 85, 79, 78, 59] or network coding [66, 63, 86, 87] or a combination of the two [58]. Superposition coding relies on iterative decoding by decoding the base layer (lower order bits) first and then re-modulating it to extract the higher layer (higher order bits); this is more complicated to implement, which is why it is difficult to find experimental data for the performance of superposition coding. Also, superposition coding offers less flexibility by limiting the SNR ranges where it can be used with acceptable error performance and often requires very high SNR as the constellation gets denser, which may not be available for any wireless node. In comparison GRaTIS provides a simpler decoder structure and offers more flexibility by providing more data rates for the MAC to select among to merge multiple packets for receivers within realistic SNR ranges. Also under similar operating conditions, GRaTIS provides higher network throughput compared to superposition coding. Network coding relies on the overhearing of a packet which it uses to decode a second packet encoded using network coding techniques. However, network coding fails if the first packet is not overheard by the intended receiver – under such a case the relay node falls back to multiple packet transmissions, which can be avoided by using GRaTIS. Most 802.11a/g based communication is in the form of an AP with multiple clients. The AP acts as a router between the wireless clients and the wired back-end network. Network coding cannot be used in such environments since the routing medium is not common and packets cannot be overheard. In such cases GRaTIS can still be used to merge downlink packets.

On the other hand, GRaTIS can be combined with network coding. Since network coding simply transmits binary information, either the base or group message can be part of a network coded packet. In fact, GRaTIS increases the applicability of network coding because the network coding message can always

be used as a GRaTIS-layer message even if other messages are not queued.

In [88], authors propose mixing bits for relaying purposes, but the mapping of bits to the constellations is different from GRaTIS. The work does not consider the fact that packet mixing will increase the SNR requirements for decoding both the layers. On the contrary, our work is practical and the SNR requirement for each layer has been shown both theoretically and experimentally. The packet trace analysis with variable packet size shows GRaTIS is applicable to most common wireless network scenarios.

7.9 Conclusion

GRaTIS provides an efficient method of simultaneous packet transmission and reception. This increases the network throughput without compromising the throughput of one node while using widespread channel variability to simultaneously transmit an independent packet destined for another node. The GRaTIS packet is indeed extra free bits to the high SNR node, which it would have received after the completion of the first packet using a serialized medium access pattern as in 802.11a/g. We have implemented the protocol in hardware and have shown the ease of implementation if the signal processing is done using a hybrid platform of software and hardware components. The experimental results show several possibilities of use of GRaTIS giving unforeseen gains in throughput in wireless networks. Applying GRaTIS on real-time packet trace analysis reveals that even with a few simple combinations, we can gain significant airtime, which can be further utilized to transmit more packets. Also through our analysis we show that GRaTIS provides more flexibility with better error performance than other contemporary simultaneous packet transmission techniques, making it a suitable candidate for emerging wireless networks.

Chapter 8

Secret Agent Radio: Covert Communication through Dirty Constellations

There are many times when communication needs to be secure. Common and obvious examples include providing security for electronic commerce or privacy for personal matters. At other times, communication must also be **covert**, or undetectable which has a **low probability of intercept** (LPI) or a **low probability of detection** (LPD). LPD communication mechanisms are useful when the very act of communication can raise concerns, such as communication during war-time or during surveillance. Usually it is difficult to detect the receiver of communication mechanisms that exploit the characteristics of radio propagation.

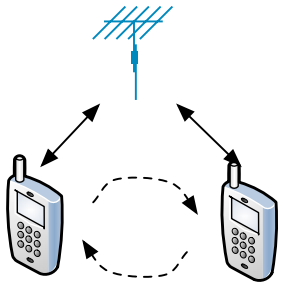


Figure 8.1: Undetected Side-Channel Communication

In this work, we explore methods that provide LPD and LPI for high-bandwidth networks. Our method provides a high-bandwidth covert side-channel between multiple radios using a common wireless network, as indicated in Figure 8.1. The method is covert because the devices (laptops or smartphones) function as normal devices. Again, the devices "hide in plain sight". Rather than raising suspicions by exchanging encrypted messages with each other or some centralized server, they appear to be conducting

normal network communication (browsing web pages, sending mail, streaming multimedia) when in reality, they are able to communicate undetected. The adversary will face great challenge in discovering the side channel because the covert channel is being transmitted by mobile nodes. Monitoring to locate such nodes would require significant investment or infrastructure, such as monitoring in every coffee shop, bus or public venue where people may be near each other.

The technique uses a common, physical-layer protocol to mask the communication that takes advantage of the hardware imperfections present in commodity hardware, intrinsically noisy channel of wireless communication and receiver diversity. We have implemented this mechanism using software-defined radios, operating in 2.4GHz ISM band, but can also be easily extended to TV whitespaces. Our prototype uses an OFDM waveform. Most consumer electronic devices use OFDM waveforms for high-bandwidth networks (including DVB, DAB, WiFi, WiMAX and LTE), and there are some benefits in "hiding" in such a ubiquitous waveform.

Imperfections in off-the-shelf Network Interface Cards (NICs) [89], coupled with an additive random wireless channel cause the signal to degrade over time and distance. To mask our communication, we "predistort" the signal to mimic the normal imperfection of the hardware and Gaussian distortion arising from the channel. This distortion appears as noise to the unobservant receiver, be it the Wi-Fi access point or an adversary. However, a receiver aware of the presence of the signal and its encoding technique can decode the "noise" to reveal the hidden message.

Our motivation for hiding the data in physical layer (analog waveform domain) of common wired and wireless protocols are the following:

- **Hide in plain sight** - Using the physical properties of the transmission medium will allow the covert channel to resemble a common waveform, only distorted by channel noise, or transmitted by a NIC with imperfections.

- **Access to covert channel** - Since the covert channel uses the signal waveform, an adversary is easily abstracted from the covert channel, as opposed to other packet level techniques using higher layers [90]. In our method, the bits of the cover packet are not altered and hence the presence of

the covert message is not detected at higher layers, or more specifically in digital domain.

- **Sample collection** - The ubiquitous nature of wireless devices and their localized transmission make it difficult to detect the presence of a covert channel. As opposed to digital contents on the Internet (music, picture, video), which can be accessed from one physical location, acquiring signal waveforms requires hauling expensive, bulky equipment (signal analyzers) to every possible hotspot.

- **Search complexity** - A $500byte$ packet, modulated with QPSK-$1/2$ rate coding, results in $\approx 19KB$ (calculation omitted due to space constraints) of I/Q information. This increases the search space by ≈ 38 times, compared to packet level analysis of a covert channel.

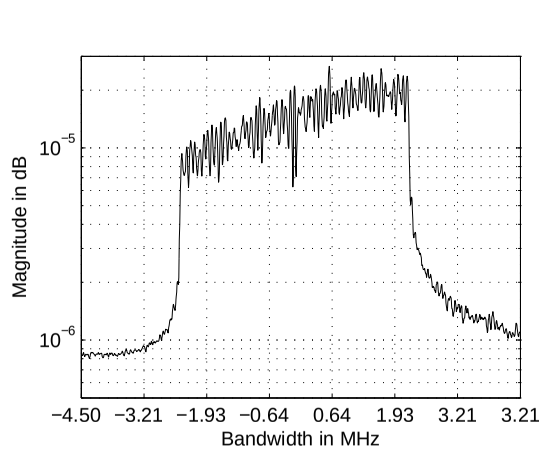
- **Statistically Undetectable** - In higher layer techniques, an adversary can search the header fields (known as unused fields) of a packet stream and find the covert channel [91], whereas in physical layer, the adversary needs to perform several statistical tests on the I/Q samples, which are already tainted by time varying channel noise.

- **Capacity** - Compared to conventional techniques using higher layers, where only a few unused bits of any header field of a packet is used, our technique can easily utilize 10% of the cover signal to transmit covert messages.

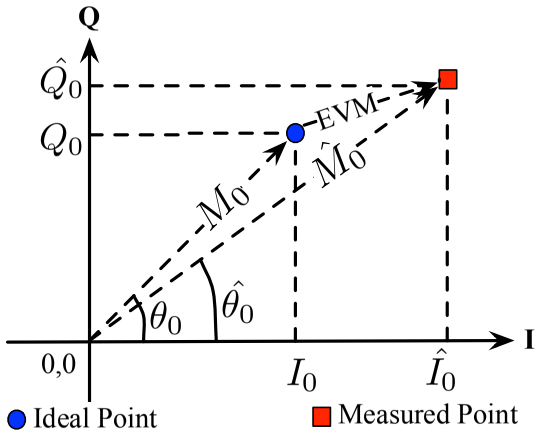
These advantages coupled with relative ease of implementation using now popularized software defined radio, makes this technique extremely useful in providing high capacity covert channels.

8.1 Characterizing OFDM Signals

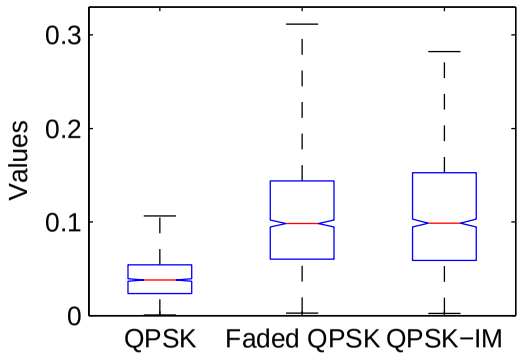
Signal quality in wireless channel depends primarily on two factors: channel impairments and hardware impairments. Channel impairments typically range from additive white noise to frequency selective fading and/or hidden terminal and Doppler shifts, which degrade signal properties in time and frequency domain. Figure 8.2(a) plots the spectrum for an OFDM waveform from a bench measurement that is skewed



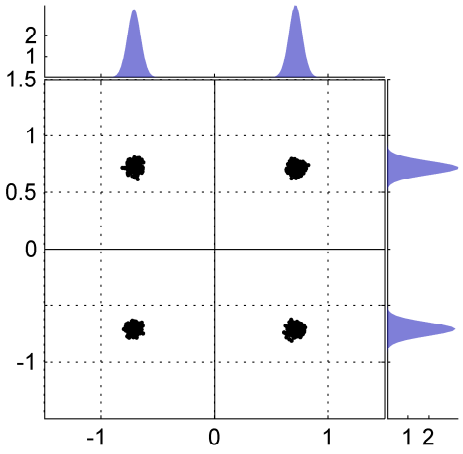
(a) Frequency Selective Fading



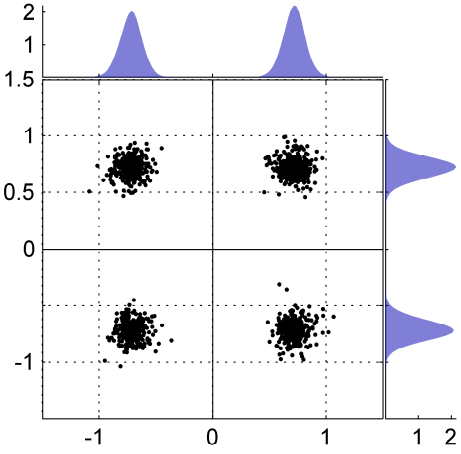
(b) Modulation Error (EVM)



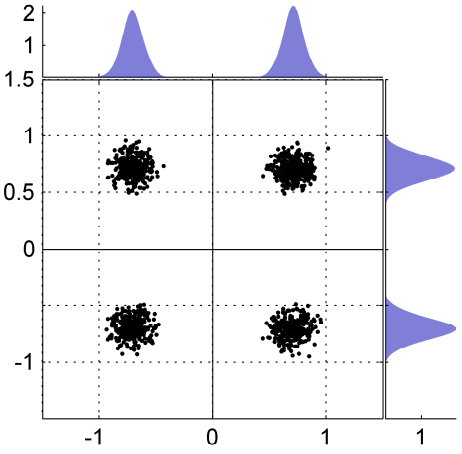
(c) EVM Comparison of QPSK



(d) Ideal QPSK



(e) Faded QPSK



(f) QPSK-IM

Figure 8.2: Characterizing channel and hardware impairments with three waveforms: ideal QPSK, faded QPSK and "impaired," QPSK-IM. The QPSK-IM signal is indistinguishable from QPSK-faded signal using statistical measures.

because of a frequency selective fading in the left-most subcarriers. Similarly, impairments due to various non-linearities in the transceiver pipeline are often reflected in the signal characteristics as well. Since these types of impairments are hardly deterministic, estimating the errors and compensating for them is a non-trivial task.

Signal-to-Noise Ratio (SNR) is a widely used metric, often measured in the time or frequency domain using averaged power measurements. A simple interpretation of the SNR is **"the higher the SNR, the higher the probability that the information can be extracted with acceptable error performance"**. However, high spatial-decorrelation of the wireless channel may render portions of the OFDM signal undecodable even though a high "average" SNR indicates otherwise. Figure 8.2(a) is an example of an OFDM spectrum of an ongoing communication that has an average SNR of 21dB but degraded in the frequency domain.

The Error Vector Magnitude (EVM), shown in Figure 8.2(b) is another metric that measures the deviation of the complex modulation vectors in the I/Q-plane from the ideal position. A bad channel leads to higher dispersion of these vectors and hence higher EVM, which affects the error performance as well. Modulation errors can also be introduced as imperfections in the transceiver hardware itself, which can cause the intended I/Q sample to be transmitted (or received) at a slight offset. In the IEEE 802.11a/g standard [45], this modulation error at the transmitter for a QPSK modulation is mandated to be no more than $10dB$ from an "ideal" I/Q mapping.

Figure 8.2(c) shows the distribution of EVM (in a boxplot) for three bench measurements of an OFDM waveform using QPSK modulation **where each of the transmissions have the same SNR**. The first measurement is based on an "ideal" transmission with low noise resulting in a low EVM with minimal variance, called ideal QPSK. The second measurement, faded QPSK, from a bench measurement with slightly different antenna orientation, has higher average EVM and wider variance. The difference between ideal QPSK and faded QPSK are due to multipath effects. The last measurement, termed the "impaired QPSK" or QPSK-IM signal, was recorded from a transmitter that pre-distorted the signal such that the average EVM is 10dB worse than the ideal. On the surface, the QPSK-IM signal appears to have similar properties to faded QPSK – both have higher average EVM and wider variance. Figures 8.2(d)-8.2(f) show the three constella-

tions corresponding to the measurements described above. It is indeterminable whether the deterioration in the EVM is due to intentionally introduced noise at the transmitter, or due to imperfections in the hardware that is operating within tolerable limits, or is the result of poor channel quality.

From these examples, it is evident that impairments, whether in the channel or in the hardware, will cause statistical variation in the perceived value of the metrics and that the bounds on these metrics are only loosely defined and can only be formalized by various descriptive statistics and statistical tests.

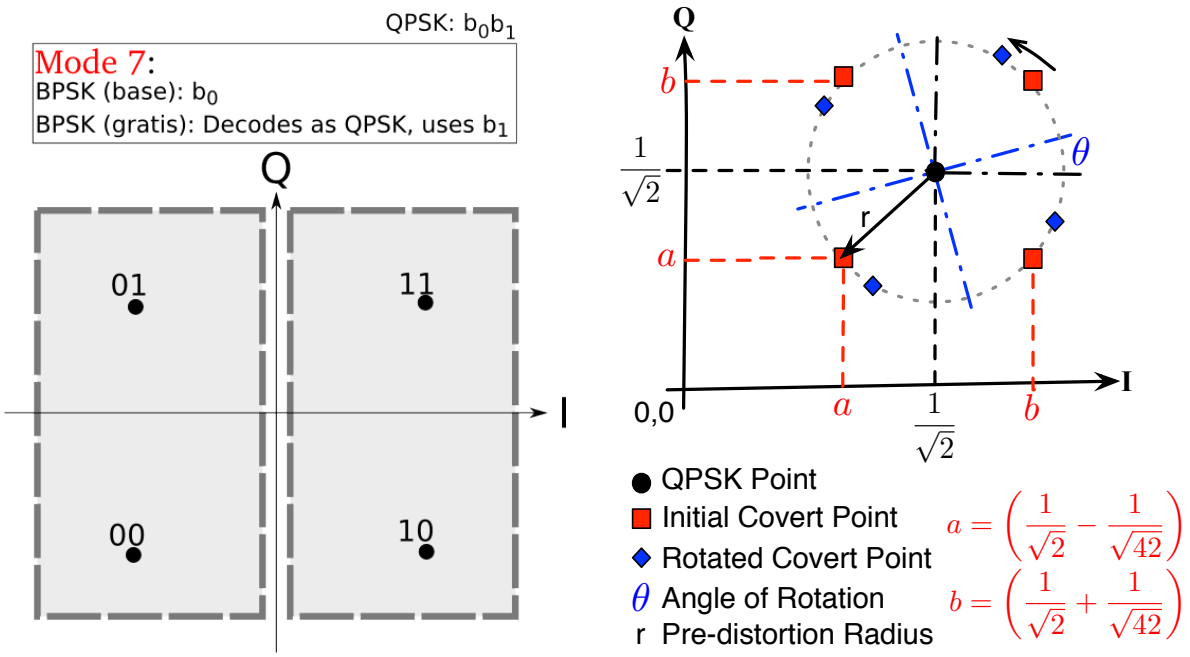
8.2 Dirty Constellation

Our method relies on being able to embed one message in another in the wireless channel, but goes well beyond that to then insure that the covert message is undetectable. There are several ways to embed messages by encoding the constellation symbols using bits of two distinct messages [83, 78] but we use a simpler technique that uses existing modulation methods of OFDM.

Using a combination of adaptive modulation and efficient packet sharing using joint constellations we encode the covert channel. If a receiver is aware of our irregular mapping of bits, and it has sufficient SNR for that subcarrier, it is able to decode the covert message while to an uninformed user, the covert constellation points will be treated as random dispersed sample of a low-rate modulation, that reveals an innocuous message.

The key to such covert communication using the physical layer of an OFDM based wireless protocol are four fold: **1)** packets containing covert data must be indistinguishable from non-covert packets to all uninformed observers; **2)** the presence of any irregularity in the covert packets has to be kept hidden under rigorous **statistical tests** on the signal; **3)** the covert channel should be non-trivial to replicate, making it secure from spoofing and impersonation; and finally, **4)** it should have high capacity. In this work we satisfy each of these requirements through a set of techniques.

Requirement 1: Identifying a Covert Channel: Our technique relies on encoding "cover packets" that are transmitted at a low rate (BPSK or QPSK) with supplemental information that can be decoded as an additional QPSK signal by an informed receiver. In the examples below, we use QPSK for both the cover and covert channel.



(a) Cover QPSK constellation

(b) Covert constellation

Figure 8.3: Encoding Dirty Constellation

In a QPSK encoding, the constellation points encode two bits of information as shown in Figure 8.3(a). To encode the covert channel, we deflect the placement of the QPSK points. This is similar to having a "covert QPSK" encoding with an origin around the ideal QPSK constellation points of the cover traffic. Figure 8.3(b) corresponds to the upper right quadrant of the cover QPSK constellation shown in Figure 8.3(a). To modulate a subcarrier carrying both the cover and covert message, first the cover constellation point (QPSK) is chosen (as per the cover message stream), specifying the quadrant, followed by re-mapping that point to one of the four "covert-QPSK" points around the "cover QPSK" point.

Clearly, the goal is to leave the cover message decodable by standard receivers. Only the covert receiver aware of the joint constellation will decode the subcarriers properly and extract the **two** covert bits to form the hidden packet. An adversary will decode at the base rate or the rate for cover message, as specified in the **signal symbol** of the packet; while the covert points will be treated as noisy points. The cover message could be intended for an access point (as part of a web browsing session) while the covert message can be overheard and decoded by a nearby radio. In this way we implement a covert channel while

making it appear as completely innocuous to other users receiving the same transmission.

Requirement 2: Low Probability of Detection: How would an adversary detect such communication? As long as the packet can be decoded, a legacy receiver has no way of knowing how signals are being encoded at the core of the physical layer, because conventional packet decoding is performed by identifying the data rates embedded at the beginning of the packet which will always contain the base rate (QPSK) information. However, adversaries using measurement equipment like vector-signal analyzers or software defined radios can extract the digital samples from the radio pipeline at different stages of the signal processing. Therefore, our ultimate goal is to provide very low probability of detection not only at the packet level but also at the signal level.

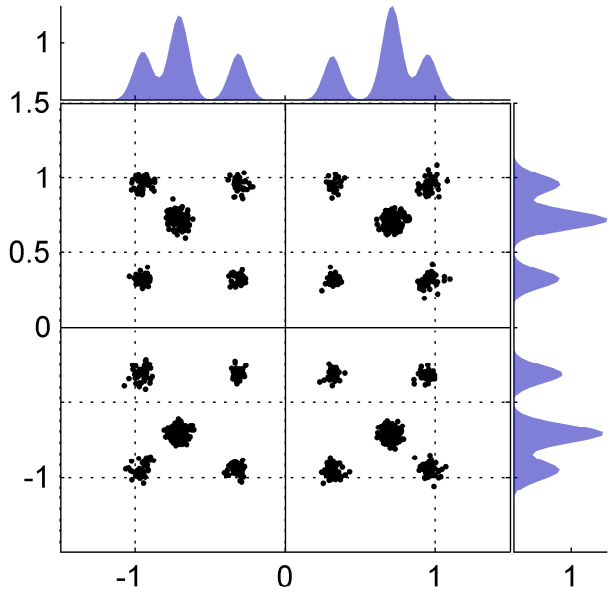


Figure 8.4: Constellation without random pre-distortion of the QPSK points and using existing 16QAM points to map the joint covert constellations.

One simple form of analysis is to look at the equalized I/Q vectors of the jointly encoded packet. The presence of the covert constellation at regular interval will appear as distinct point clouds that will set themselves apart from the cover QPSK point cloud and will reveal the presence of the covert channel, as shown in Figure 8.4.

We solve this problem by changing the I/Q vectors of the covert transmitter in three steps:

Step1: We bring the covert constellation points closer to the ideal QPSK point and re-map the covert constellation points symmetrically around the QPSK points, with a mutual separation of $\frac{2}{\sqrt{42}}$, a distance equal to that of a 64QAM constellation, so that a covert receiver can operate within the operating range of a WiFi receiver.

Step2: We randomize the I/Q vectors of the covert QPSK points with a Gaussian distribution but limit their dispersion to a radius of $\sqrt{\frac{2}{42}}$ as shown in figure 8.3(b). We call this as the **pre-distortion circle**; pre-distortion of the QPSK signal at the transmitter ensures that the covert constellations are hidden in the cloud of a dispersed (noisy) QPSK point cloud. We introduce imperfections to the transmitted signal in such a way that the average EVM error is equal to or less than 10dB compared to the ideal QPSK constellation points, which is within the limits of hardware anomaly allowed in the IEEE 802.11 standard [45]. Thus, it cannot be ascertained with certainty if the EVM error is due to hardware impairments, channel impairments or intentionally injected distortion.

Step3: To accommodate a higher rate covert channel, **e.g.**, when 50% of the OFDM subcarriers are covert, then at high SNR there is always a finite probability that the covert constellations are visible. To have the covert symbols blend with the pre-distorted QPSK point cloud, the covert symbols are rotated along the circumference of the pre-distortion circle for every subcarrier that is mapped to a covert constellation as shown in Figure 8.3(b). The rotation is performed using a monotonically increasing angle θ; the transmitter and receiver both start with $\theta = 0^\circ$ at the start of the packet and increment θ for each covert subcarrier. In our implementation we use a 15° counter-clockwise rotation for the covert points.

These 3 steps allow us to hide the covert channel, even when an adversary has access to the I/Q samples of the packet. The adversary will interpret the point cloud as a noisy version of a valid (albeit noisy) QPSK constellation and would not suspect the presence of a covert communication. This compound constellation involving a covert channel hidden within a cover constellation is termed a **"Dirty Constellation"**. However, in order to avoid raising suspicion by any RF fingerprinting algorithms [89], a QPSK-IM waveform should **always** be used for non-covert transmissions, to avoid sudden changes in the modulation characteristics.

Requirement 3&4: Security and Higher Efficiency: These requirements are considered as an enhance-

ment to the basic scheme of Dirty Constellation. We have implemented 10%, 30% and 50% encoding of subcarriers, as shown in Figure 8.7, yielding up to $9Mbps$ datarate with QPSK modulation and $3/4$ encoding rate. Using higher modulation constellation, **e.g.**, 256-QAM, we can further increase the capacity of the covert channel by encoding more bits per subcarrier. Due to space constraints we leave this as future work. Finally, we discuss the security aspect in §8.6.

8.3 Dirty Constellation on SDR

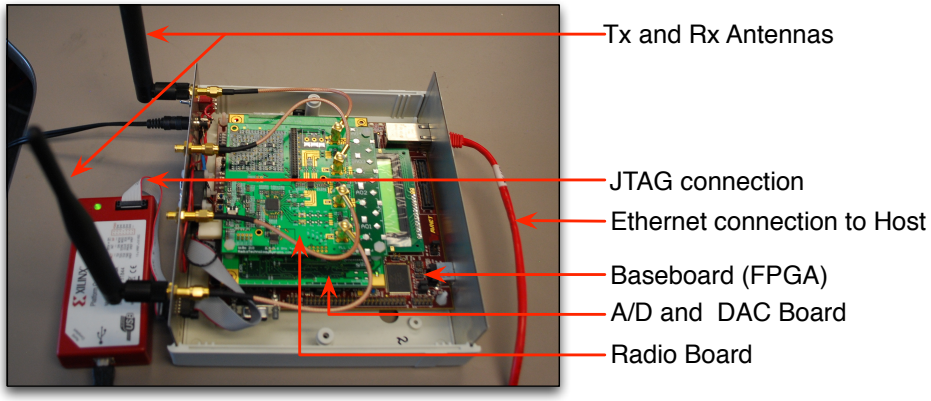
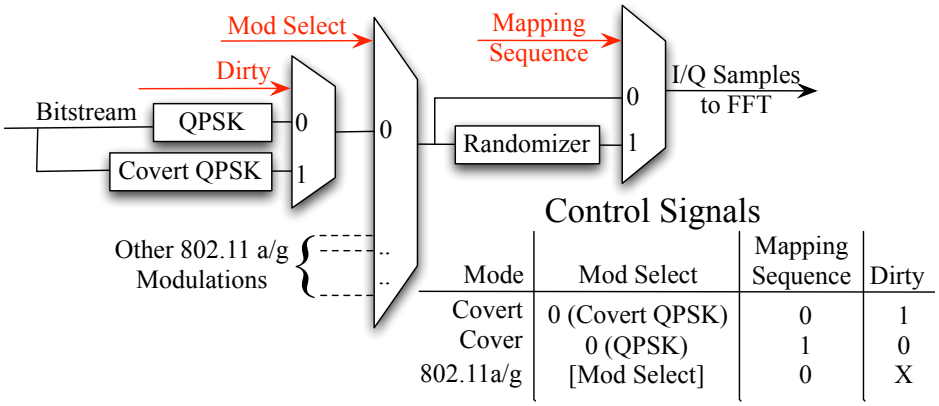


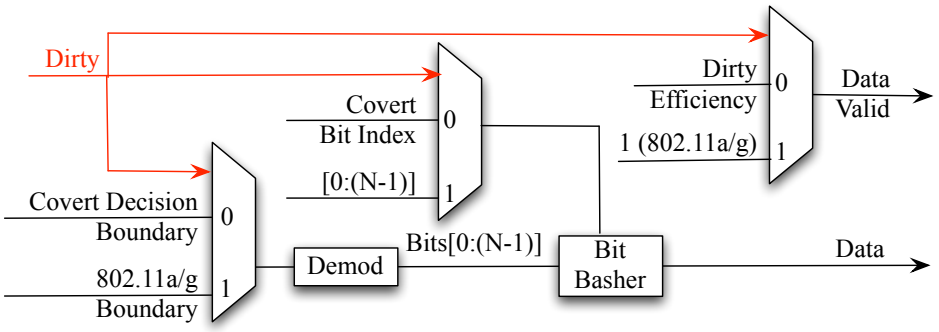
Figure 8.5: SDR prototype using Virtex-V FPGA

Hiding a message in a randomized modulation constellation requires a programmable modulator and demodulator. Conventional radio 802.11 PHYs modulate all the subcarriers with one type of pre-defined modulation, For this scheme to work, we used a FPGA-based software defined radio platform based on our previous work [11, 14], as shown in figure 8.5, and modified the modulator and demodulator to program each subcarrier with different modulations, adding either noise or covert constellations. Figure 8.6(a) shows the functional diagram of the programmable modulator. The notable parameters in the design are the **dirty** bit and the **mapping sequence** bit which are used to select the appropriate mapping for covert joint constellations and randomize (Gaussian) the cover symbols to engulf the higher order modulation points. The cover and the covert bits are independently packetized as per the 802.11a/g specification and the covert joint symbols are formed by merging the bits of the two packets prior to sending it to the modulator. The merging of packets is performed in software and then fed to the hardware along with the control information to

create the Dirty Constellation. The QPSK-IM constellation is generated by using the randomizer unit that emulates an overall modulation error of 10dB, by setting the **dirty** bit to '0' and **mapping sequence** to '1' for all subcarriers.



(a) Transmitter Pipeline



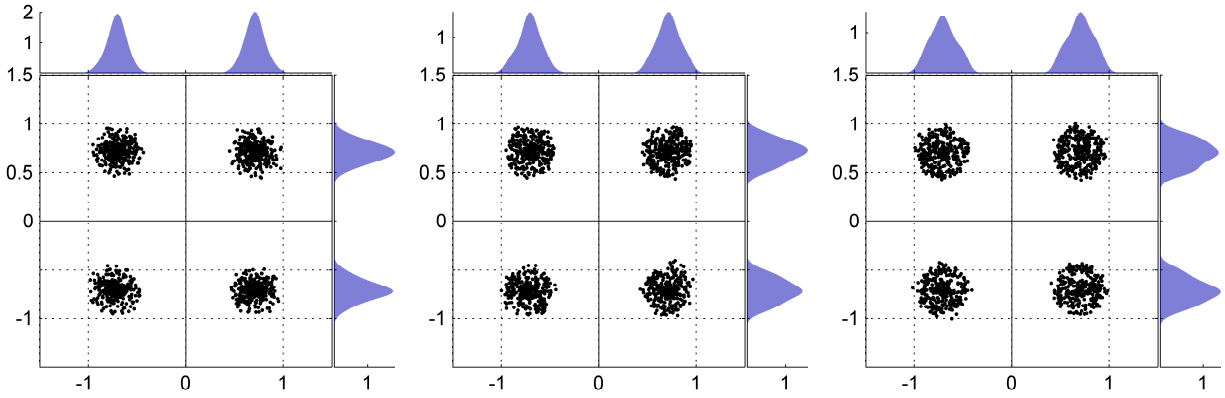
(b) Receiver Pipeline

Figure 8.6: Mod/Demodulator for Dirty Constellation

The decoder employs maximum likelihood decoding and uses pre-defined thresholds to decode the constellation. Figure 8.6(b) shows the functional diagram of the demodulator. First the covert receiver demodulates the signal using the covert decision boundaries, 64QAM in this case and then extracts the covert bits. Since all subcarriers do not contain the hidden message, the receiver then uses the pre-assigned mapping sequence and its rotation information to filter out the covert subcarriers' information to form the covert packet.

Figure 8.7 shows an example of Dirty Constellation with varying frequency of the covert channel that has been transmitted by the SDR prototype and captured using a VSA. The I and Q histograms along-

side the constellation shows the similarity of the distributions and that they are from the family of normal distributions.



(a) Frequency – 10% (b) Frequency – 30% (c) Frequency – 50%

Figure 8.7: Examples of over-the-air transmission of Dirty Constellations with varying embedding frequency using the SDR prototype

8.4 Experiments and Measurements

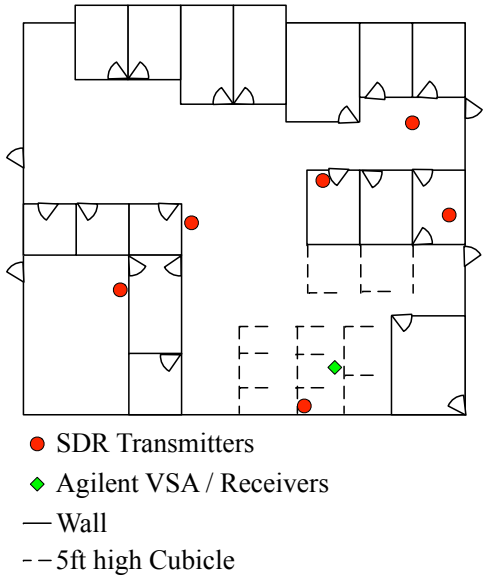


Figure 8.8: Node placement

Using the hardware described in §8.3 as the transmitter, signal samples are collected in a lab/office

environment. The transmitter nodes were placed as shown in the Figure 8.8. The signals were captured using a high-end Agilent vector signal analyzer (VSA) that provides the raw I/Q vectors of the packets transmitted by the SDR nodes. We record data from 6 locations, for ideal QPSK, QPSK-IM and Dirty Constellation with 10% covert channel efficiency. Each dataset contains measurement of 500 data packets of each type per transmit power level. The transmit power is varied in steps of 2.5dB such that the measured SNR at the VSA has a range of 7dB to 20dB. We have chosen this range because 7dB is the minimum SNR required to decode a QPSK packet with 98% packet reception rate. This has been empirically validated using bench measurements using our SDR transceivers. Likewise, 20dB was selected as the upper limit because the EVM doesn't decrease appreciably with higher SNR. After filtering out the required data range we find the average sample size is $10,000$ packets per type. We bin the packets by SNR in bins of size 1dB; each bin contains $500 - 800$ packets per SNR value. We perform all the statistical testing using this dataset which captures a wide range of SNR and channel conditions for all the type of modulations. In these measurements, the VSA is treated as both the covert receiver **and** a very aggressive adversary. As a covert receiver, the messages sent by the different transmitters can be received by the VSA receiver and the covert data can be extracted. As an adversary, the receiver has a high quality measurement device and also acts as the "most aggressive adversary" because it shares the same channel state as the receiver.

8.5 Analyzing Dirty Constellation

The core idea of testing a sample for adherence to a particular family of signals is performed by comparing test results with a known set of statistics for the same class. Therefore, the first step of the analysis process is to formalize the database of these statistics that characterizes an entire family of signals. In this work, we intend to compare a Dirty Constellation with a QPSK waveform. We formulate the problem as a hypothesis test, with the null hypothesis:

$\mathcal{H}_0$: Given a random sample from a Dirty Constellation packet, it is statistically same as any other QPSK packet.

Whereas the alternative hypothesis is:

$\mathcal{H}_1$: Given a random sample from a Dirty Constellation packet, it can be statistically identified that it is not a QPSK packet.

In this section, we analyze whether the packets containing covert data can be distinguished from normal packets at the packet level or at the waveform level in the time and frequency domain. The test statistics of standard QPSK signals is lower bounded by the statistics of an "ideal QPSK" packet and upper bounded by a "QPSK-IM" packet. We used "QPSK-IM" packets to mimic a radio with hardware imperfections, but operating within the limits of IEEE 802.11 standard requirements. Each of these bounds have been empirically derived from the measurements collected as described in §8.4. If the Dirty Constellation sample is within the bounds set for that test then the null hypothesis is "not rejected", meaning that the Dirty Constellation packet is statically indistinguishable from any other QPSK transmission within the expanse of 802.11a/g transmissions using that test.

8.5.1 Packet Based Analysis

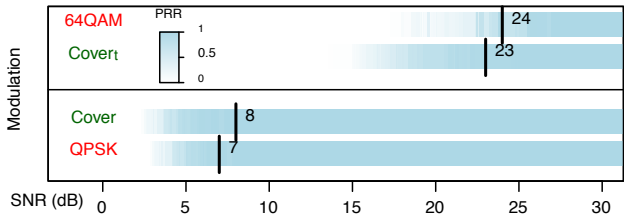


Figure 8.9: Packet Reception Rate

Packet based analysis involves looking at parameters that can be extracted at the packet level, or in the digital domain, where there is no trace of the covert packet. To measure if the pre-distortion of the constellation effects the packet reception rate (for both the covert and the cover packet) we performed measurements over a one hop link between two SDR nodes over a wide range of SNR. Figure 8.9 shows the packet reception rate for the standard modulations used in 802.11a/g and also the SNR required by the intended receiver of the covert packet and the cover packet. The minimum SNR levels required for 98% packet reception rate is marked. For the cover packets, our mechanism is within 1dB of that required by standard 802.11a/g modulation. Given the stochastic nature of the wireless channel and high spatial

de-correlation of the nodes, this difference is indistinguishable to an end user (the user would experience greater variance simply by moving their receiver a few inches). The covert receiver requires an SNR of 24dB, similar to the SNR needed to decode a 64QAM packet.

8.5.2 Signal Domain Analysis

A time varying signal is often characterized either by time-domain measurements (power envelope and peak to average power ratio) or by performing spectral measurements such as power spectral density, phase and magnitude distributions. Since OFDM encodes data in the frequency domain as coefficients of an inverse Fourier Transformation, a frequency domain analysis is of utmost importance and hence we conduct a set of frequency domain analysis, followed by tests in the time domain.

Frequency Domain Tests –

Test 1: EVM of Constellations: The real and imaginary vectors (I & Q) are available at the output of the Fourier transform unit. EVM is the absolute value of the dispersion of the I/Q-vector averaged over all OFDM symbols in a packet. Figure 8.10 shows EVM with varying SNR for the QPSK and QPSK-IM bounds and for the Dirty Constellation as well. The inter-quartile distances represents the spread of the I/Q vectors as they are degraded by channel noise. The EVM of the Dirty Constellation is distributed within the bounds set for QPSK making it statistically undetectable when compared with the empirical benchmarks. The plot also shows the average of EVM of a frequency faded random QPSK measurement, which emphasizes the non-deterministic effects of the channel that can push the envelope of the set bounds in either direction. That sample has the same parameters and configuration as the "ideal QPSK", but with the antenna moved by 2 inches. We expect the test statistic to be correlated with the variation in the bounds.

Test 2: Measure of I/Q Dispersion: The relative dispersion of the I/Q vectors result in a change in the position of the constellation point. Although all receivers employ channel equalization to compensate for the channel distortion, there are always residual errors that cause the points to violate their respective decision threshold leading to bit errors. Figures 8.11(a) and 8.11(b) show how the **deviation** from an ideal QPSK constellation is distributed within the dataset. Deviations in the the Dirty Constellation packets are within the bounds for most of the SNR values. To ascertain that the distributions are indeed similar and highly

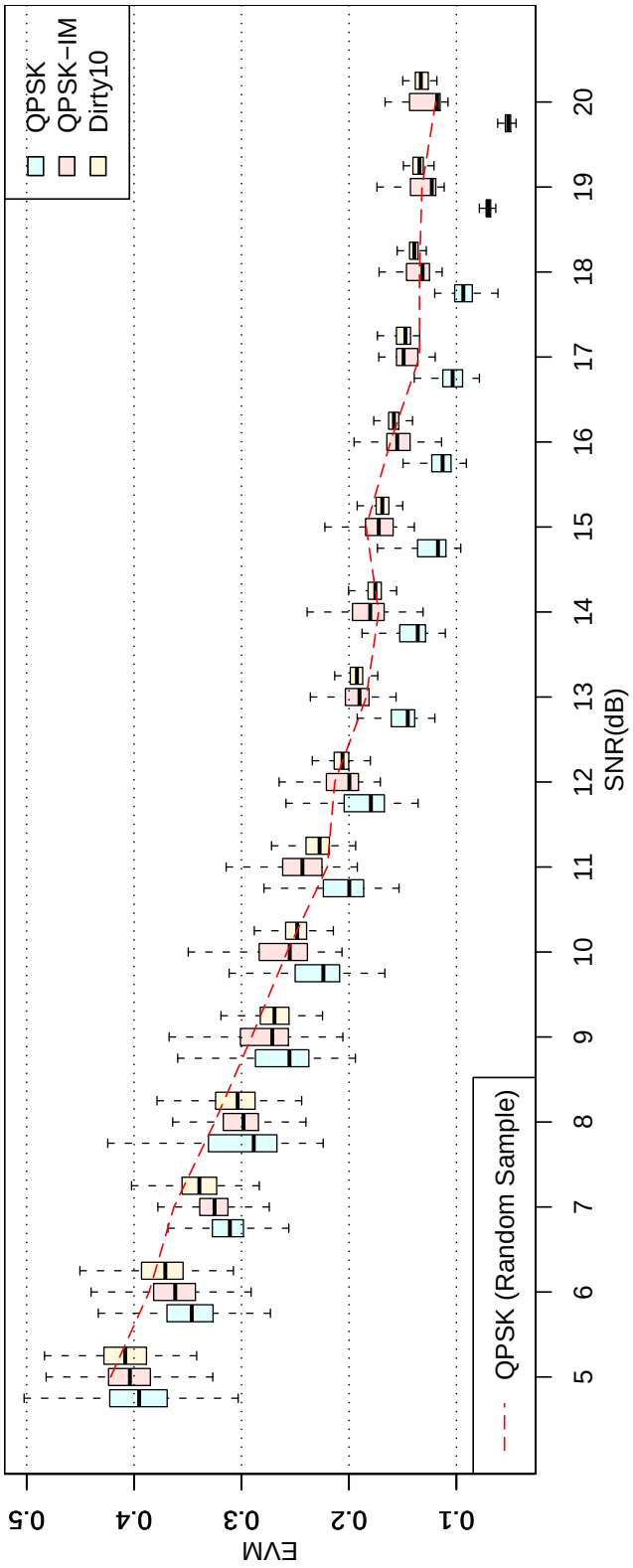
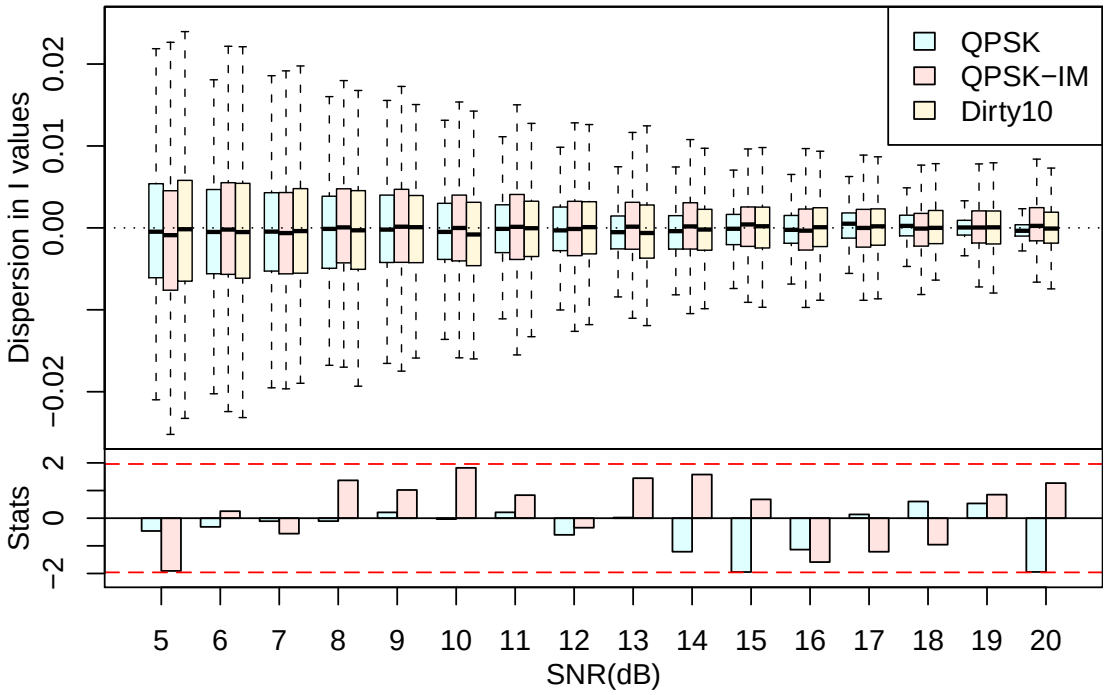
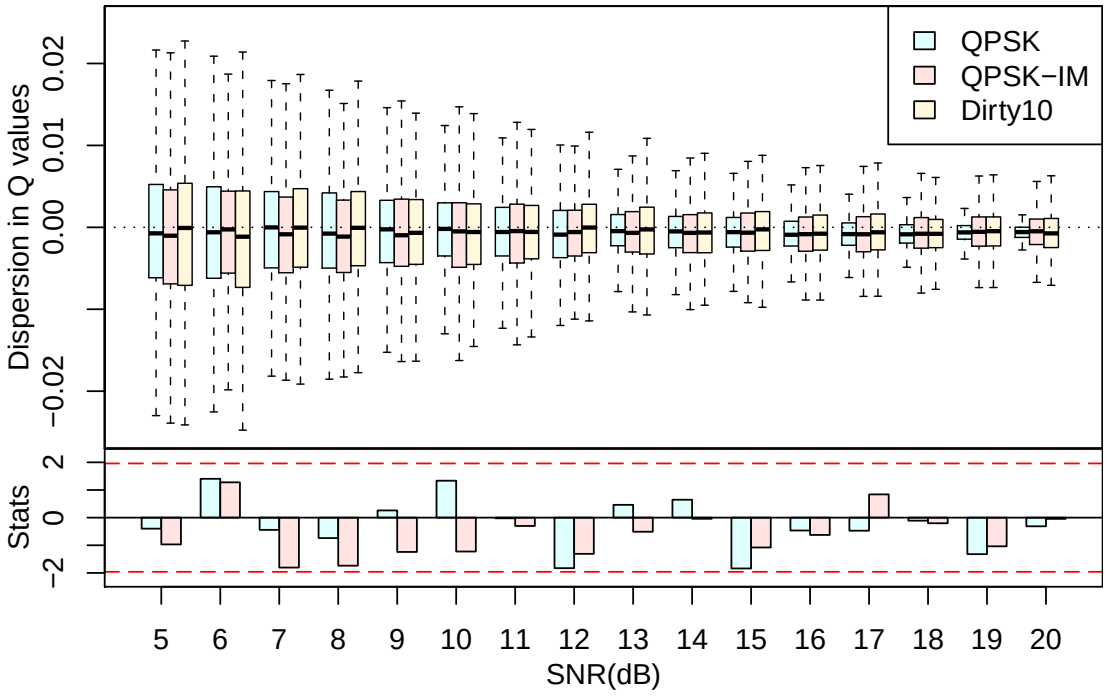


Figure 8.10: Distribution of EVM. A faded ideal QPSK sample is also shown.



(a) Dispersion in I vector



(b) Dispersion in Q vector

Figure 8.11: Dispersion of I and Q vectors from ideal QPSK mapping. The distribution of the I/Q dispersion is verified with that of ideal QPSK and QPSK-IM using a two sample t-test.

correlated, and that they are normally distributed about the ideal QPSK constellation, we perform a two sample t-test with the ideal QPSK packet and the QPSK-IM packet. The test statistics for all the SNR are found to be less than the critical value at the 0.05 significance level, as shown in the bottom part of figure 8.11. This also satisfies the test that the I/Q dispersion for all the three types are distributed in similar fashion and are from the family of normal distribution with statistically similar means.

Test 3: Phase and Magnitude Distribution: Often it is important to know how the phase and magnitude vary with the subcarrier index. Figure 8.13 shows a histogram of the subcarrier phases of all packets in the collected dataset at two SNR levels, low SNR (7dB) and high SNR (18dB). At low SNR the subcarriers undergo distortion over a wider range and so the phases have a wider distribution, while at high SNR the signal is closer to the ideal QPSK signal. However, in both the SNR levels, the phases from the Dirty Constellation packets are distributed similarly to the ideal QPSK and QPSK-IM. The four distinct peaks at multiples of 45° ascertain that Dirty Constellation preserves the phase properties of the QPSK constellation. Similarly, the magnitude distribution across the subcarriers show that the magnitude of the subcarriers in a packet encoded with Dirty Constellation are distributed within the bounds of QPSK waveforms, as shown in figure 8.12. It is also seen that there is a high degree of correlation among the subcarrier from the three types of packets: the same multipath affects all three transmissions. To show that the distributions are correlated we also show the quantile-quantile (QQ) plot for subcarrier magnitudes of the QPSK-IM and the Dirty Constellation packets, as shown in figure 8.14. The linearity of the QQ plot indicates the signals have similar distributions.

Time Domain Tests –

Test 1: Temporal Variation of Average Signal Power: To test if the Dirty Constellation affects the signal power, we compare the temporal variation with that of a QPSK packet. In an experiment, 20 packets were captured using the VSA for all three types of packet at intervals of $\approx 500ms$. The average power is shown in Figure 8.15(a). The power envelope for the packets are randomly distributed even though the packets all have similar signal to noise ratios. Therefore, from this test we conclude that our method does not change the average signal power that is different from that of other QPSK packets.

Test 2: Peak to Average Power Ratio (PAPR): OFDM can produce spurious increase in the peak power

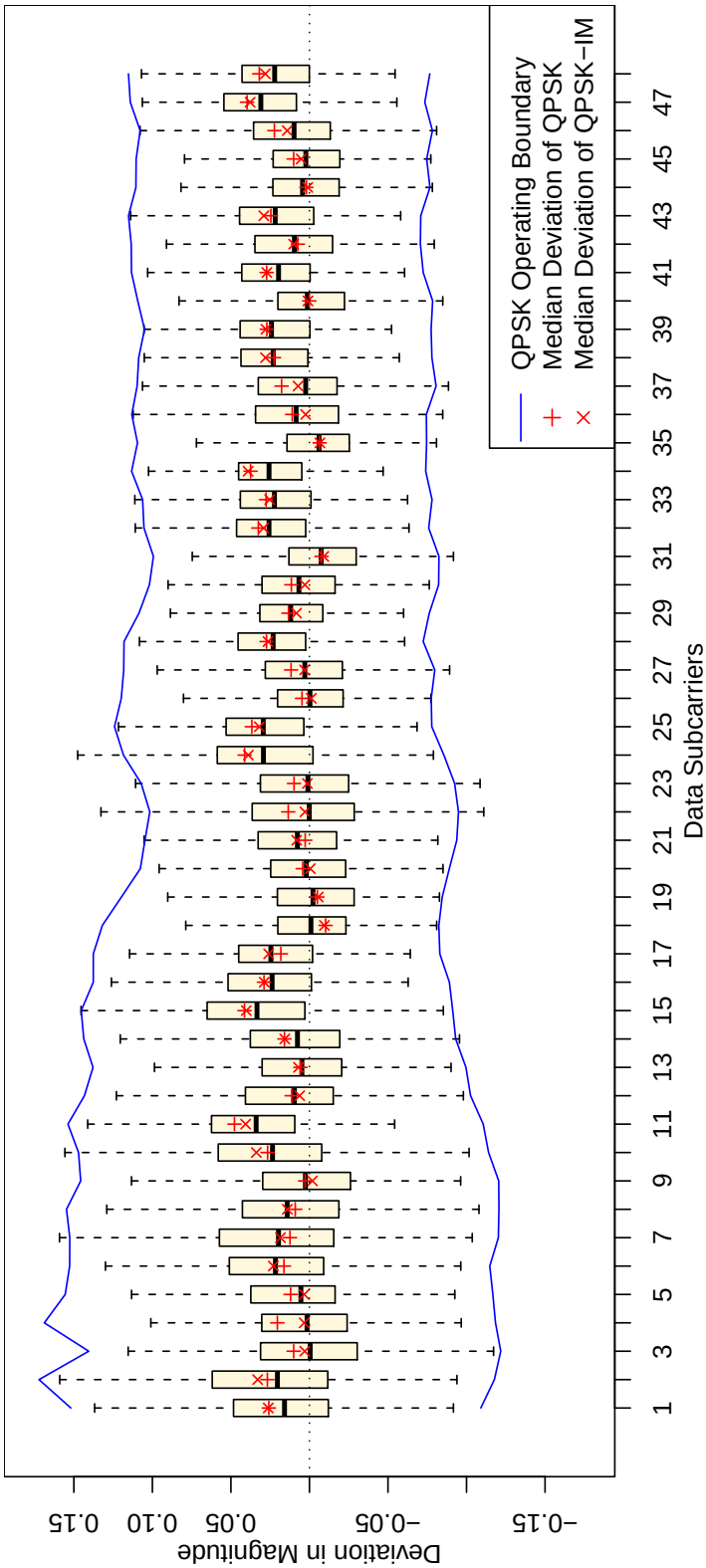


Figure 8.12: Magnitude Dispersion per Subcarrier

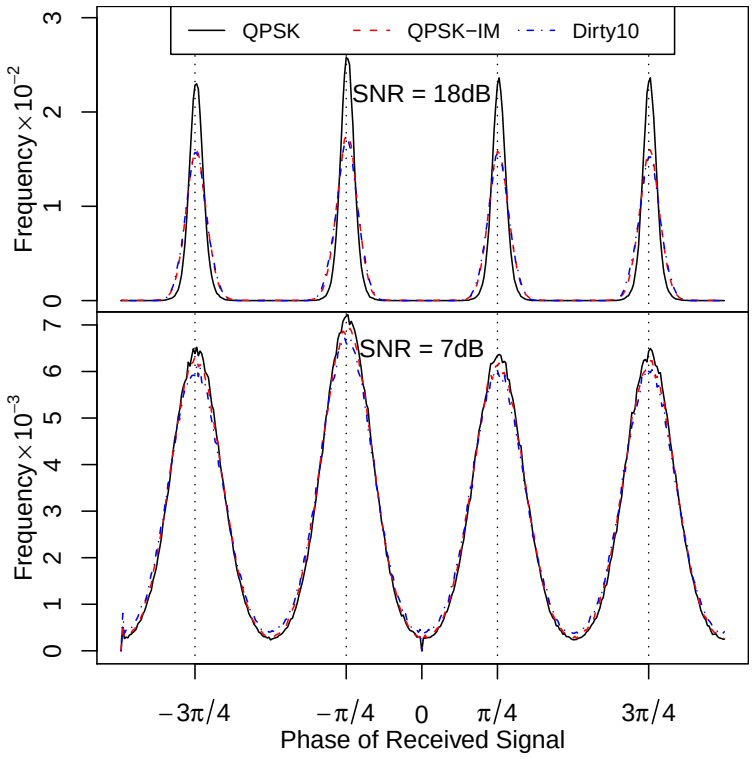
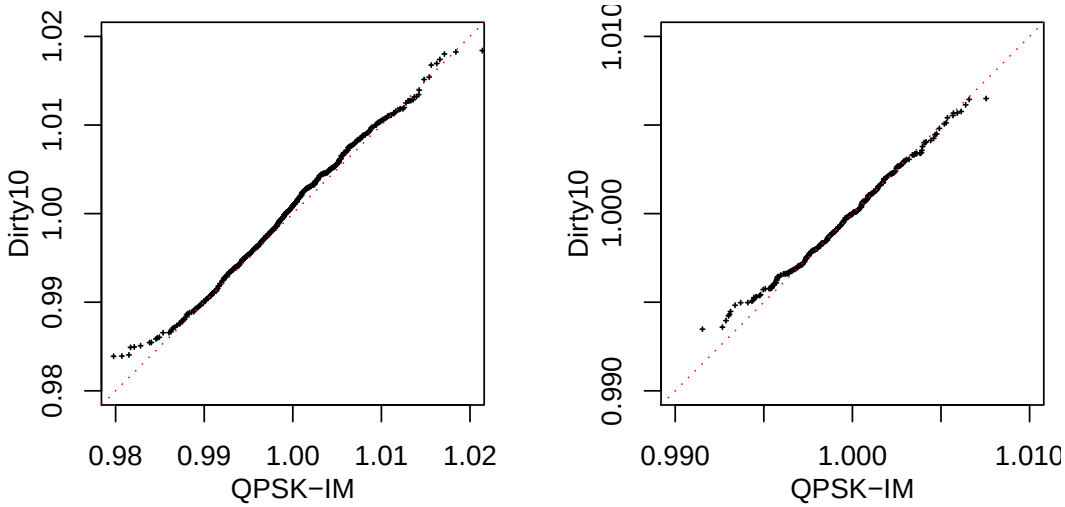


Figure 8.13: Phase Distribution

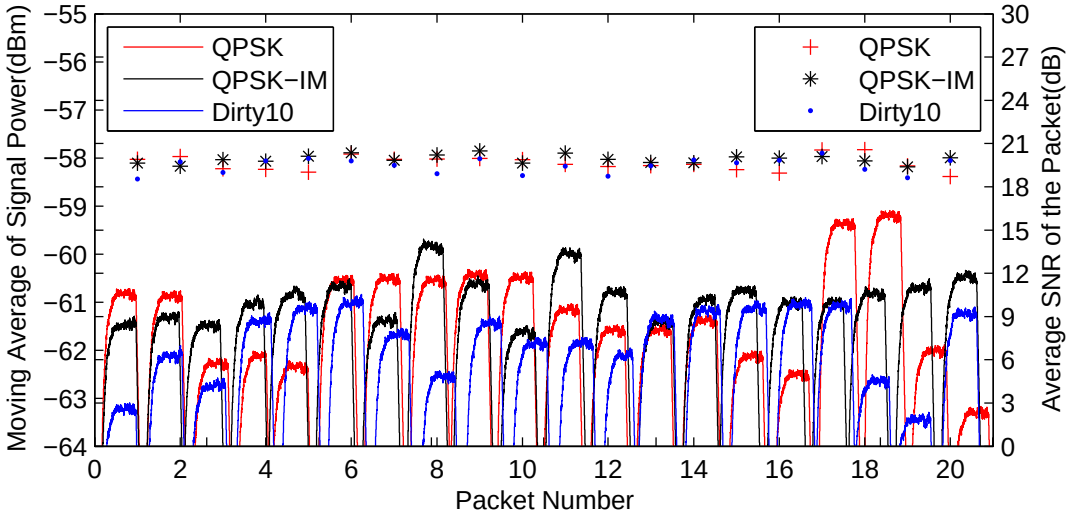


(a) At SNR = 7dB

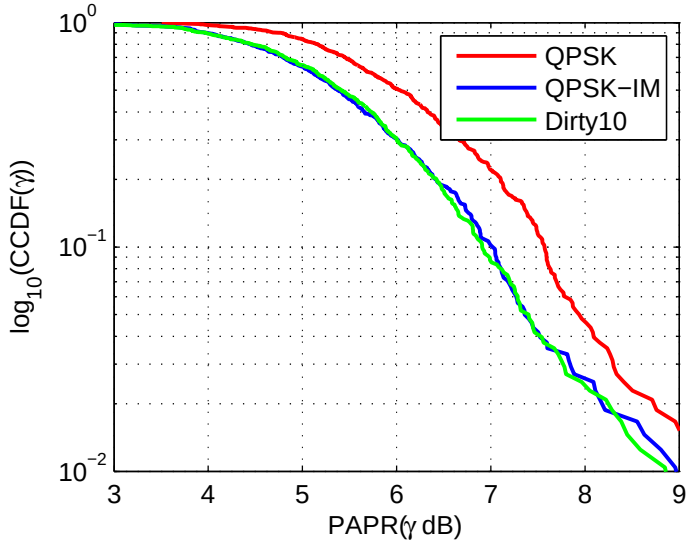
(b) At SNR = 18dB

Figure 8.14: QQ-Plot of Magnitude

when the packet contains different types of modulations. PAPR is the measure of the spurious increase in power in the time domain. Figure 8.15(b) shows the complementary CDF (CCDF) of the PAPR for the



(a) Moving average of signal power



(b) CCDF of PAPR

Figure 8.15: Time Domain Analysis

three packet types. Research [92, 93] shows that the PAPR in 802.11a/g can vary over a wide range with various PAPR optimization techniques. The PAPR for Dirty Constellation falls within that range and follows closely with that of QPSK-IM. Hence it cannot be distinguished as an anomaly compared to the ideal QPSK transmission.

In this section we conducted tests that fail to reject the null hypothesis leading us to conclude that

our method is statistically undetectable when compared to known waveforms that spans over a wide range of SNR. The analysis in frequency as well as time domain ensures the completeness of the testing. Thus, we conclude that our method can be successfully used as a covert channel that has very low probability of detection.

8.5.3 Exceptions

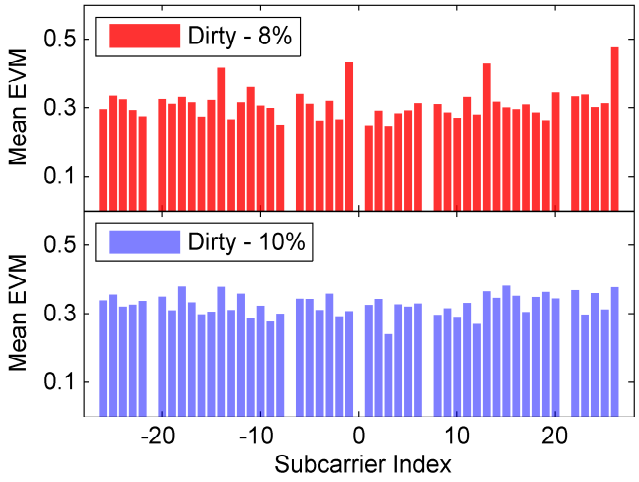


Figure 8.16: Average EVM per subcarrier

In this section we provide examples of Dirty Constellation that **are** easily detectable, indicating that the methods and bit-mapping of the covert channel is non-trivial and requires careful analysis before adopting. One would guess that a lower embedding rate is better even though that results in a lower covert data rate. To see that this is not the case, we changed the embedding frequency to $\approx 8\%$ (1 in 12 subcarriers). Figure 8.16 shows the mean EVM of each data subcarrier of Dirty-8% compared to that of Dirty-10%. Since the Dirty-8% affects 1 in 12 subcarriers, a regular pattern is emphasized in the EVM of certain subcarriers. The mean EVM for Dirty-8% clearly shows that four out of 48 subcarriers has significantly higher EVM. On the contrary, Dirty-10% has a more even distribution of mean EVM in all of its subcarriers because 48 is not evenly divisible by 10.

8.6 Security

In §8.5.3 we discussed that mapping of covert channel is a non-trivial problem. This mapping sequence could be generated using a pseudo random number (PRN) sequence generator. Dirty Constellation employs two forms of sequence or pattern: the covert carrier mapping sequence and the angle of rotation for the covert constellation along the pre-distortion circle. While one PR sequence controls the embedding frequency, another specifies the rotation parameters, such as the angle of rotation "θ" for the covert constellation and the direction of rotation. The receiver needs to know which packets contain covert communication as well as the PRN's used to mix the covert message into the cover message. The frequency of covert messages can also be randomly varied without the need for additional coordination. The PRN used to intermix the covert message is synchronized with the receiver at the beginning of a transmission and can vary over time using an agreed-upon PRN based on **e.g.** the time of day. Any existing encryption method (like AES, DES) can be used in each packet as an added measure to increase the security of the proposed method. However, due to space constraints, we do not analyze the details of the security aspects of this technique in this work.

8.7 Related Work

Hiding information has been prevalent since ancient times; however hiding data in digital format is more a recent developments with the popularization of Computer Science. Much of the early work [94] in data hiding with low probability of detection and interception has been done by altering a few bits of the digital representation of an image [95], a sound [96] or video [97] files.

A relatively recent field of study called **network stenography** exploits the redundant fields present in various network protocols headers, like HTTP and TCP. Zander et. al. [98] provides a comprehensive survey of covert channels in computer network protocols. All of the methods detailed in the work are confined to identifying anomalies or using the protocol properties at the application, transport or the data link layer. Also [90] proposes another scheme to hide data based on utilizing redundant fields in IPv4 header while [91] presents a practical analysis of covert channels in wireless LAN protocols at the transport layer.

Information hiding at the application layer of a mobile telephony network has been discussed in [99]. These protocols depend on altering the data itself, which is susceptible to higher probability of interception, when the altered data is tested. Our procedure is significantly different from previous work in the sense that we modify the way of data transmission without altering the bits of any digitally transmitted data. In other words, higher layer stenography operates in the **digital** domain while our method operates in the **analog** domain.

Examples of covert channel implementation utilizing the physical layer are few and far between. A PHY layer based security scheme has been proposed in [100]. However, this method works only when more than one user is available to transmit stenographic packets to a common node. Also it relies on very tight synchronization between multiple transmitter and single receiver entity, which is not a practical assumption in real networks and will lead to erroneous formation of the joint constellations leading to degraded performance. Therefore, comparing to prior work, our method presents a more practical solution to implement covert channels at the PHY layer, while making it secure, high capacity, easily implementable and backward compatible.

8.8 Conclusion

In this work, we discussed a technique to implement a covert channel at the physical layer of 802.11a/g wireless protocol. By hiding the covert channel within the perceived noise at the receiver, we can ensure high degree of undetectability. We have implemented the covert communication method using a SDR prototype and present results of a wide variety of statistical tests that confirms the low probability of detection of Dirty Constellation. Higher datarate, very low probability of detection coupled with easy implementation within existing protocol stacks make Dirty Constellation a very successful method to implement covert channels in wireless communication.

Chapter 9

Conclusion

In this thesis we design, implement and evaluate techniques that fundamentally change the way modern wireless networks operate. Instead of confining our solutions to the layered architecture of networking, we focus on optimizing the MAC and the PHY as one entity. By exchanging crosslayer information, we are able to harness the benefits of both the layers and architect practical solutions to open problems in next generation wireless networks.

Simultaneous multiuser communication is made possible by using orthogonal channels and its efficient implementation using the OFDM waveform. In this thesis, we analyze common OFDM PHY used in wireless systems and identify novel features and properties that enable a wider realm of research, with focus on how higher layer protocols are designed. This flow of new information from the PHY has been harnessed to design MAC layer protocols that provide unprecedented gains in various aspects of wireless networks: whether making group communications faster or making the communication covert. All these techniques employ simultaneous multiuser communication. An important component of such crosslayer optimization is a mutable PHY that is modified as required in order to implement these novel techniques. Hence, this thesis is made successful by deep understanding of the PHY, its implementation using highly programmable radio platforms and extensive experimentation using testbed setup.

The orthogonality of the OFDM subcarriers have been utilized to design simultaneous multiuser communication methods in the PAMAC [39] and the SMACK [64] crosslayer techniques. In both the cases, nodes use unique OFDM subcarriers to send distinct orthogonal *tones*, simultaneously, to send simple responses to common broadcast queries. Although this techniques is not limited to broadcast scenario only,

we show that this makes group communications faster by order of magnitude under practical wireless channels. We also extend this concept of OFDM subcarriers to design collision avoidance systems in vehicular networks called Active Radar [101].

Along with the OFDM subcarriers, we also utilize the modulation constellations used in OFDM to design multiuser communication methods. In GRaTIS [102], we modify the existing constellations of 802.11a/g to send independent packets to multiple users. This is made possible due to the widespread variation of SNR in wireless networks, where a near node and a far node decodes information at different SNR. Hence the transmitter can *merge* their packets in one transmission and rely on the variability of the channel to implement multiuser communication. Extensive theoretical and practical analysis of this technique has shown gains of up to 80% in throughput compared to contention based MAC layer protocols. Also, the modulation constellations are used to implement a covert channel at the PHY which has very low probability of detection. Using a combination of programmable radio, to inject intentional pre-distortion at the transmitter, and by taking advantage of common radio impairments, we are successful in designing a high throughput side-channel that is statistically indistinguishable from normal wireless transmissions.

Therefore, this thesis is an example of MAC- PHY crosslayer optimization that utilizes the unexplored facets of the PHY layer to build networked systems providing substantial gain in network metrics that would not have been possible otherwise. Also, we show that in order to make these techniques practical, the PHY should no longer be fixed in function but a highly mutable substrate with precise control over the various subsystems. It is our belief that this thesis is a promising step in the direction of practical MAC-PHY crosslayer protocols that usher a new era of wireless research.

Bibliography

[1] A. Schulman, D. Levin, and N. Spring, "{CRAWDAD} trace set umd/sigcomm2008/pcap (v. 2009-03-02)," Downloaded from http://crawdad.cs.dartmouth.edu/umd/sigcomm2008/pcap, Mar. 2009.

[2] CISCO, "Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 20112016 [Visual Networking Index (VNI)]." [Online]. Available: http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/white_paper_c11-520862.html

[3] M. L. Doelz, E. T. Heald, and D. L. Martin, "Binary Data Transmission Techniques for Linear Systems," Proceedings of the IRE, vol. 45, no. 5, pp. 656–661, May 1957.

[4] R. W. Chang, "Orthogonal Frequency Division Multiplexing," U.S. Patent, Jan. 1970.

[5] J. A. C. Bingham, "Multicarrier modulation for data transmission: an idea whose time has come," Communications Magazine, IEEE, vol. 28, no. 5, pp. 5–14.

[6] A. Fort, J.-W. Weijers, V. Derudder, W. Eberle, and A. Bourdoux, "A performance and complexity comparison of auto-correlation and cross-correlation for OFDM burst synchronization," Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03). 2003 IEEE International Conference on, vol. 2, pp. II–341–4 vol.2, Apr. 2003.

[7] J. M. III, "Cognitive Radio - An Integrated Agent Architecture for Software Defined Radio," Ph.D. dissertation, Royal Institute of Technology (KTH), 2000.

[8] F. K. Jondral, "Software-Defined RadioBasics and Evolution to Cognitive Radio," EURASIP Journal on Wireless Communications and Networking, vol. 2005, no. 3, pp. 275–283, 2005.

[9] R. T. Fielding, "Architectural Styles and the Design of Network-based Software Architectures," Ph.D. dissertation, Department of Information and Computer Science, University of California Irvine, 2000.

[10] V. Bose, M. Ismert, M. Welborn, and J. Guttag, "Virtual Radios," IEEE/JSAC Special Issue on Software Radios, Apr. 1999.

[11] J. Fifield, P. Kasemir, D. Grunwald, and D. Sicker, "Experiences with a platform for frequency agile techniques," in DYSPAN, 2007.

[12] J. Chapin and V. Bose, "The Vanu Software Radio System," in 2002 Software Defined Radio Technical Conference, Nov. 2002.

[13] K. Amiri, Y. Sun, P. Murphy, C. Hunter, J. R. Cavallaro, and A. Sabharwal, "WARP, A Unified Wireless Network Testbed for Education and Research," in Proceedings of IEEE MSE, 2007.

[14] A. Dutta, J. Fifield, G. Schelle, D. Grunwald, and D. Sicker, "An Intelligent Physical Layer For Cognitive Radio Networks," in WICON '08: Proceedings of the 4th international conference on Wireless internet. New York, NY, USA: ACM, 2008.

[15] G. Schelle, J. Fifield, and D. Griinwald, "A Software Defined Radio Application Utilizing Modern FPGAs and NoC Interconnects," Field Programmable Logic and Applications, 2007. FPL 2007. International Conference on, pp. 177–182, 2007.

[16] D. A. Ilitzky, J. D. Hoffman, A. Chun, and B. P. Esparza, "Architecture of the Scalable Communications Core's Network on Chip," IEEE Micro, vol. 27, no. 5, pp. 62–74, 2007.

[17] G. Chinya, J. Collins, M. Girkar, H. Jiang, G. Lueh, L. Pearce, X. Tian, H. Wang, P. Wang, and S. Yakoushkin, "Accelerator Exoskeleton," Intel Technology Journal, Aug. 2007.

[18] M. Woh, S. Seo, H. Lee, Y. Lin, S. Mahlke, C. Chakrabarti, and K. Flautner, "{T}he {N}ext {G}eneration {C}hallenge for {S}oftware {D}efined {R}adio," in SAMOS, 2007.

[19] Y. Lin, H. Lee, M. Woh, Y. Harel, S. Mahlke, T. Mudge, C. Chakrabarti, and K. Flautner, "SODA: A Low-power Architecture For Software Radio," in ISCA '06: Proceedings of the 33rd annual international symposium on Computer Architecture. Washington, DC, USA: IEEE Computer Society, 2006, pp. 89–101.

[20] A. Duller, D. Towner, G. Panesar, A. Gray, and W. Robbins, "picoArray Technology: The Tool's Story," in DATE '05: Proceedings of the conference on Design, Automation and Test in Europe. Washington, DC, USA: IEEE Computer Society, 2005, pp. 106–111.

[21] D. Halperin, T. Anderson, and D. Wetherall, "Taking the {S}ting out of {C}arrier {S}ense: {I}nterference {C}ancellation for {W}ireless {LAN}s," in MOBICOM '08: Proceedings of the ACM MOBICOM 2008 conference, ser. MobiCom '08. New York, NY, USA: ACM, 2008, pp. 339–350.

[22] S. Gollakota and D. Katabi, "Zigzag Decoding: Combating Hidden Terminals in Wireless Networks," in SIGCOMM '08: Proceedings of the ACM SIGCOMM 2008 conference on Data communication. New York, NY, USA: ACM, 2008, pp. 159–170.

[23] H. Rahul, N. Kushman, D. Katabi, C. Sodini, and F. Edalat, "Learning to Share: Narrowband-Friendly Wideband Networks," in SIGCOMM '08: Proceedings of the ACM SIGCOMM 2008 conference on Data communication. New York, NY, USA: ACM, 2008, pp. 147–158.

[24] A. Dutta, D. Saha, D. Grunwald, and D. Sicker, "A Concurrent Acknowledgement Scheme for Broadcast Messages in Wireless Networks," University of Colorado at Boulder, Tech. Rep., Sep. 2008.

[25] E. Blossom, "GNURadio as an Experimental Platform: Current Capabilities and Future Directions," in WINTECH, 2007, pp. 1–2.

[26] Wireless@VirginiaTech, "OSSIE: Open Source SCA Implementation," \url{http://ossie.wireless.vt.edu/}, 2008. [Online]. Available: http://ossie.wireless.vt.edu/

[27] J. Camp and E. Knightly, "{M}odulation {R}ate {A}daptation in {U}rban and {V}ehicular {E}nvironments: {C}ross-layer {I}mplementation and {E}xperimental {E}valuation," in MOBICOM '08: Proceedings of the ACM MOBICOM 2008 conference. New York, NY, USA: ACM, 2008.

[28] A. Dutta, D. Saha, D. Grunwald, and D. Sicker, "An architecture for software defined cognitive radio," in Proceedings of the 6th ACM/IEEE Symposium on Architectures for Networking and Communications Systems, ser. ANCS '10. New York, NY, USA: ACM, Sep. 2010, pp. 5:1—-5:12. [Online]. Available: http://doi.acm.org/10.1145/1872007.1872014

[29] M. Rodrig, C. Reis, R. Mahajan, D. Wetherall, and J. Zahorian, "Measurement-based Characterization of 802.11 in a Hotspot Setting," in Proceedings of the ACM SIGCOMM 2005 Workshop on experimental approaches to wireless network design and analysis (E-WIND-05), Philadelphia, PA, Aug. 2005.

[30] J. Choi, J. Yoo, S. Choi, and C. Kim, "EBA: an enhancement of the IEEE 802.11 DCF via distributed reservation," Mobile Computing, IEEE Transactions on, vol. 4, no. 4, pp. 378–390, 2005.

[31] C. Wang, B. Li, and L. Li, "A new collision resolution mechanism to enhance the performance of IEEE 802.11 DCF," Vehicular Technology, IEEE Transactions on, vol. 53, no. 4, pp. 1235–1246, Jul. 2004.

[32] Y. Wang and B. Bensaou, "Achieving fairness in IEEE 802.11 DFWMAC with variable packet lengths," Global Telecommunications Conference, 2001. GLOBECOM '01. IEEE, vol. 6, pp. 3588–3593 vol.6, 2001.

[33] "Reconfigurable OFDM Receiver for Next Generation Wireless Mesh," Master's thesis, University of Colorado, Boulder, Colorado, United States, 2008.

[34] "IEEE Std 802.11-2007, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications." [Online]. Available: http://standards.ieee.org/

[35] S. N. Technologies, "QualNet Network Simulator, version 4.0."

[36] M. Heusse, F. Rousseau, R. Guillier, and A. 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. New York, NY, USA: ACM Press, 2005, pp. 121–132.

[37] C. E. Perkins and E. M. Royer, "Adhoc On-Demand Distance Vector Routing," in Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications, 1999, pp. 90–100.

[38] M. A. Abu-Rgheff, Introduction to CDMA Wireless Communications, 1st ed. Academic Press, 2007.

[39] D. Saha, A. Dutta, D. Grunwald, and D. Sicker, "PHY Aided MAC: A New Paradigm," INFOCOM 2009. The 27th Conference on Computer Communications. IEEE, Apr. 2009.

[40] S.-Y. Ni, Y.-C. Tseng, Y.-S. Chen, and J.-P. Sheu, "The broadcast storm problem in a mobile ad hoc network," in MobiCom '99: Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking. New York, NY, USA: ACM, 1999, pp. 151–162.

[41] V. Bhandari and N. H. Vaidya, "Reliable Broadcast in Wireless Networks with Probabilistic Failures," INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE, pp. 715–723, May 2007.

[42] M. H. Ammar and G. N. Rouskas, "On the performance of protocols for collecting responses over a multiple-access channel," INFOCOM '91. Proceedings. Tenth Annual Joint Conference of the IEEE Computer and Communications Societies. Networking in the 90s., IEEE, pp. 1490–1499 vol.3, Apr. 1991.

[43] M. Demirbas, O. Soysal, and M. Hussain, "A Singlehop Collaborative Feedback Primitive for Wireless Sensor Networks," INFOCOM 2008. The 27th Conference on Computer Communications. IEEE, pp. 2047–2055, Apr. 2008.

[44] P. Dutta, R. Musaloiu-E., I. Stoica, and A. Terzis, "Wireless {ACK} Collisions Not Considered Harmful," in Proceedings of the Seventh Workshop on Hot Topics in Networks (HotNets-VII), Oct. 2008.

[45] Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, IEEE Computer Society : LAN/MAN Standards Committee. [Online]. Available: http://standards.ieee.org/getieee802/802.11.html

[46] "Part 16: Air Interface for Broadband Wireless Access Systems." [Online]. Available: http://standards.ieee.org/getieee802/download/802.16-2009.pdfhttp://standards.ieee.org/getieee802/download/802.11-2007.pdf

[47] "IEEE 802 LAN/MAN Standards Committee 802.22 Working Group on WRANs." [Online]. Available: http://www.ieee802.org/22/

[48] S. Coleri, M. Ergen, A. Puri, and A. Bahai, "Channel estimation techniques based on pilot arrangement in OFDM systems," in Broadcasting, IEEE Transactions on, vol. 48, pp. 223–229.

[49] T. Thanabalasingham, S. V. Hanly, L. L. H. Andrew, and J. Papandriopoulos, "Joint Allocation of Subcarriers and Transmit Powers in a Multiuser OFDM Cellular Network," Communications, 2006. ICC '06. IEEE International Conference on, vol. 1, pp. 269–274, Jun. 2006.

[50] F. Kojima, H. Harada, and M. Fujise, "Adaptive sub-carriers control scheme for OFDM cellular systems," Vehicular Technology Conference Proceedings, 2000. VTC 2000-Spring Tokyo. 2000 IEEE 51st, vol. 2, pp. 1065–1069 vol.2, 2000.

[51] J. Acharya, H. Viswanathan, and S. Venkatesan, "Timing Acquisition for Non Contiguous OFDM Based Dynamic Spectrum Access," New Frontiers in Dynamic Spectrum Access Networks, 2008. DySPAN 2008. 3rd IEEE Symposium on, pp. 1–10, 2008.

[52] P. H. Tan and L. K. Rasmussen, "Multiuser detection in CDMA - a comparison of relaxations, exact, and heuristic search methods," Wireless Communications, IEEE Transactions on, vol. 3, no. 5, pp. 1802–1809, 2004.

[53] B. Roman, F. Stajano, I. Wassell, and D. Cottingham, "Multi-Carrier Burst Contention (MCBC): Scalable Medium Access Control for Wireless Networks," Wireless Communications and Networking Conference, 2008. WCNC 2008. IEEE, pp. 1667–1672, 2008.

[54] H. Arslan and T. Ycek, "Spectrum Sensing for Cognitive Radio Applications." Springer Netherlands, 2007, ch. 9, pp. 263–289.

[55] R. Tandra and A. Sahai, "SNR Walls for Signal Detection," Selected Topics in Signal Processing, IEEE Journal of, vol. 2, no. 1, pp. 4–17, 2008.

[56] D. Halperin, T. Anderson, D. Wetherall, and T. A. Daniel Halperin, "Taking the sting out of carrier sense: interference cancellation for wireless LANs," in Proceedings of the 14th ACM international conference on Mobile computing and networking, ser. MobiCom '08. New York, NY, USA: ACM, 2008, pp. 339–350. [Online]. Available: http://doi.acm.org/10.1145/1409944.1409983

[57] S. Sen, N. Santhapuri, R. R. Choudhury, and S. Nelakuditi, "Successive interference cancellation: a back-of-the-envelope perspective," in Proceedings of the Ninth ACM SIGCOMM Workshop on Hot Topics in Networks, ser. Hotnets '10. New York, NY, USA: ACM, 2010, pp. 17:1—-17:6. [Online]. Available: http://doi.acm.org/10.1145/1868447.1868464

[58] C.-H. Liu and A. Arapostathis, "Joint Network Coding and Superposition Coding for Multi-User Information Exchange in Wireless Relaying Networks," in Global Telecommunications Conference, 2008. IEEE GLOBECOM 2008. IEEE, 2008, pp. 1–6.

[59] S. Bopping and J. M. Shea, "Superposition coding in the downlink of CDMA cellular systems," in Wireless Communications and Networking Conference, 2006. WCNC 2006. IEEE, vol. 4, 2006, pp. 1978–1983.

[60] M. Vutukuru, H. Balakrishnan, and K. Jamieson, "Cross-layer wireless bit rate adaptation," in SIGCOMM '09: Proceedings of the ACM SIGCOMM 2009 conference on Data communication. New York, NY, USA: ACM, 2009, pp. 3–14.

[61] S. Sen, N. Santhapuri, R. R. Choudhury, and S. Nelakuditi, "AccuRate: Constellation Based Rate Estimation in Wireless Networks," in NSDI. USENIX Association, Apr. 2010, pp. 175–190. [Online]. Available: http://dl.acm.org/citation.cfm?id=1855711.1855723

[62] D. D. Clark, R. Braden, A. Falk, and V. K. Pingali, "FARA: reorganizing the addressing architecture," Computer Communication Review, vol. 33, no. 4, pp. 313–321, 2003.

[63] S. Katti, S. Gollakota, and D. Katabi, "Embracing wireless interference: analog network coding," in SIGCOMM '07: Proceedings of the 2007 conference on Applications, technologies, architectures, and protocols for computer communications. New York, NY, USA: ACM, 2007, pp. 397–408.

[64] A. Dutta, D. Saha, D. Grunwald, and D. Sicker, "SMACK: a SMart ACKnowledgment scheme for broadcast messages in wireless networks," in SIGCOMM '09: Proceedings of the ACM SIGCOMM 2009 conference on Data communication, vol. 39, no. 4. New York, NY, USA: ACM, 2009, pp. 15–26.

[65] D. Halperin, W. Hu, A. Sheth, and D. Wetherall, "Predictable 802.11 packet delivery from wireless channel measurements," in SIGCOMM, S. Kalyanaraman, V. N. Padmanabhan, K. K. Ramakrishnan, R. Shorey, and G. M. Voelker, Eds. ACM, 2010, pp. 159–170.

[66] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Médard, and J. Crowcroft, "XORs in the air: practical wireless network coding," SIGCOMM Comput. Commun. Rev., vol. 36, no. 4, pp. 243–254, 2006.

[67] D. Qiao, S. Choi, and K. G. Shin, "Goodput analysis and link adaptation for IEEE 802.11a wireless LANs," in Mobile Computing, IEEE Transactions on, vol. 1, no. 4, 2002, pp. 278–292.

[68] P. Frenger, P. Orten, and T. Ottosson, "Convolutional codes with optimum distance spectrum," Communications Letters, IEEE, vol. 3, no. 11, pp. 317–319, Nov. 1999.

[69] Y. Yasuda, K. Kashiki, and Y. Hirata, "High-Rate Punctured Convolutional Codes for Soft Decision Viterbi Decoding," Communications, IEEE Transactions on, vol. 32, no. 3, pp. 315–319, Mar. 1984.

[70] D. Haccoun and G. Begin, "High-rate punctured convolutional codes for Viterbi and sequential decoding," Communications, IEEE Transactions on, vol. 37, no. 11, pp. 1113–1125, Nov. 1989.

[71] "Madwifi Driver, http://madwifi-project.org/." [Online]. Available: http://madwifi-project.org/

[72] "Radiotap Header, http://www.radiotap.org/." [Online]. Available: http://www.radiotap.org/

[73] F. Li, M. Li, R. Lu, H. Wu, C. Mark, and K. Robert, "Measuring queue capacities of IEEE 802.11 wireless access points," in Broadband Communications, Networks and Systems, 2007. BROADNETS 2007. Fourth International Conference on, 2007, pp. 846–853.

[74] M. Rodrig, C. Reis, R. Mahajan, D. Wetherall, J. Zahorjan, and E. Lazowska, "{CRAWDAD} data set uw/sigcomm2004 (v. 2006-10-17)," Downloaded from http://crawdad.cs.dartmouth.edu/uw/sigcomm2004, Oct. 2006.

[75] Y. Kim, S. Choi, K. Jang, and H. Hwang, "Throughput enhancement of IEEE 802.11 WLAN via frame aggregation," in Vehicular Technology Conference, 2004. VTC2004-Fall. 2004 IEEE 60th, vol. 4, 2004, pp. 3030 – 3034 Vol. 4.

[76] J. Zhang, K. Tan, J. Zhao, H. Wu, and Y. Zhang, "A Practical SNR-Guided Rate Adaptation," in INFOCOM 2008. The 27th Conference on Computer Communications. IEEE, Apr. 2008, pp. 2083–2091.

[77] J. Camp and E. Knightly, "Modulation rate adaptation in urban and vehicular environments: cross-layer implementation and experimental evaluation," in MobiCom '08: Proceedings of the 14th ACM international conference on Mobile computing and networking. New York, NY, USA: ACM, 2008, pp. 315–326.

[78] R. K. Ganti, Z. Gong, M. Haenggi, C.-H. Lee, S. Srinivasa, D. Tisza, S. Vanka, and P. Vizi, "Implementation and Experimental Results of Superposition Coding on Software Radio," in 2010 IEEE International Conference on Communications (ICC'10), Cape Town, South Africa, May 2010.

[79] R. Alimi, L. E. Li, R. Ramjee, H. Viswanathan, and Y. R. Yang, "{iPack}: in-Network Packet Mixing for High Throughput Wireless Mesh Networks," in Proceedings of {IEEE} {INFOCOM}, Phoenix, AZ, Apr. 2008.

[80] P. Patel and J. Holtzman, "Analysis of a simple successive interference cancellation scheme in a DS/CDMA system," Selected Areas in Communications, IEEE Journal on, vol. 12, no. 5, pp. 796–807, Jun. 1994.

[81] K. Ramchandran, A. Ortega, K. M. Uz, and M. Vetterli, "Multiresolution broadcast for digital HDTV using joint source-channel coding," pp. 556–560 vol.1, Jun. 1992.

[82] K. M. Uz, M. Vetterli, and D. J. LeGall, "Interpolative multiresolution coding of advance television with compatible subchannels," Circuits and Systems for Video Technology, IEEE Transactions on, vol. 1, no. 1, pp. 86–99, Mar. 1991.

[83] N. Shacham, "Multipoint communication by hierarchically encoded data," in INFOCOM '92. Eleventh Annual Joint Conference of the IEEE Computer and Communications Societies, IEEE, May 1992, pp. 2107–2114 vol.3.

[84] E. S. Lo and K. B. Letaief, "Network coding versus superposition coding for two-way wireless communication," in WCNC'09: Proceedings of the 2009 IEEE conference on Wireless Communications & Networking Conference. Piscataway, NJ, USA: IEEE Press, 2009, pp. 307–311.

[85] R. Ramjee, J. Shi, Y. Sun, H. Viswanathan, and Y. R. Yang, "Extended Abstract: Superposition Coding for Wireless Mesh Networks ABSTRACT."

[86] S. Katti, D. Katabi, H. Balakrishnan, and M. Medard, "Symbol-level network coding for wireless mesh networks," SIGCOMM Comput. Commun. Rev., vol. 38, no. 4, pp. 401–412, 2008.

[87] M. Kim, D. Lucani, X. Shi, F. Zhao, and M. Medard, "Network Coding for Multi-Resolution Multicast," ArXiv e-prints, Aug. 2009.

[88] Z. Yang, Y. Luo, and L. Cai, "Network modulation: A new dimension to enhance wireless network performance," in INFOCOM, 2011 Proceedings IEEE, Apr. 2011, pp. 2786–2794.

[89] V. Brik, S. Banerjee, M. Gruteser, and S. Oh, "Wireless device identification with radiometric signatures," in Proceedings of the 14th ACM international conference on Mobile computing and networking, ser. MobiCom '08. New York, NY, USA: ACM, 2008, pp. 116–127. [Online]. Available: http://doi.acm.org/10.1145/1409944.1409959

[90] C. Krätzer, J. Dittmann, A. Lang, and T. Kühne, "WLAN steganography: a first practical review," in MM&S;Sec '06: Proceedings of the 8th workshop on Multimedia and security. New York, NY, USA: ACM, 2006, pp. 17–22.

[91] K. Ahsan and D. Kundur, "Practical Data Hiding in {TCP/IP}," in Proc. Workshop on Multimedia Security at ACM Multimedia '02, French Riviera, Dec. 2002.

[92] A. Aggarwal and T. H. Meng, "Minimizing the Peak-to-Average Power Ratio of OFDM Signals Using Convex Optimization," in Signal Processing, IEEE Transactions on, vol. 54, no. 8, 2006, pp. 3099–3110.

[93] A. Jayalath and C. Tellambura, "Peak-to-Average Power Ratio of IEEE 802.11 a PHY Layer Signals," in Advanced Signal Processing for Communication Systems, ser. The International Series in Engineering and Computer Science, T. A. Wysocki, M. Darnell, and B. Honary, Eds., vol. 703. Springer US, 2002, pp. 83–96. [Online]. Available: http://dx.doi.org/10.1007/0-306-47791-2_7

[94] F. A. P. Petitcolas, R. J. Anderson, and M. G. Kuhn, "Information hiding-a survey," Proceedings of the IEEE, vol. 87, no. 7, pp. 1062–1078, Jul. 1999.

[95] M. Wu, E. Tang, and B. Lin, "Data hiding in digital binary image," in Multimedia and Expo, 2000. ICME 2000. 2000 IEEE International Conference on, vol. 1, 2000, pp. 393 –396 vol.1.

[96] D. Gruhl, W. Bender, and A. Lu, "Echo Hiding," in Information Hiding: 1st Int. Workshop (Lecture Notes in Computer Science), 1996.

[97] C. Xu, X. Ping, and T. Zhang, "Steganography in Compressed Video Stream," in Innovative Computing, Information and Control, 2006. ICICIC '06. First International Conference on, vol. 1, 2006, pp. 269–272.

[98] S. Zander, G. Armitage, and P. Branch, "A Survey of Covert Channels and Countermeasures in Computer Network Protocols," Communications Surveys Tutorials, IEEE, vol. 9, no. 3, pp. 44–57, 2007.

[99] S. S. Agaian, D. Akopian, and S. D'Souza, "Wireless steganography," in Multimedia on Mobile Devices II, R. Creutzburg, J. H. Takala, and C. W. Chen, Eds., no. 1. SPIE, 2006, p. 60740G. [Online]. Available: http://link.aip.org/link/?PSI/6074/60740G/1

[100] G. R. Tsouri and D. Wulich, "Securing OFDM over wireless time-varying channels using subcarrier overloading with joint signal constellations," EURASIP J. Wirel. Commun. Netw., vol. 2009, p. 2, 2009.

[101] D. Saha, A. Dutta, D. Grunwald, and D. Sicker, "Active radarA cooperative approach using multicarrier communication," in Local Computer Networks (LCN), 2010 IEEE 35th Conference on. IEEE, 2010, pp. 627–630.

[102] ——, "Gratis: Sensing and intelligence for performance in the presence of legacy networks," in IEEE CROWNCOM, 2012.