FLEXIBLE SPECTRUM USE IN CHANNEL BONDING WIRELESS NETWORKS

By

Xi Yang

A THESIS

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

Computer Science - Master of Science

2014

ABSTRACT

FLEXIBLE SPECTRUM USE IN CHANNEL BONDING WIRELESS NETWORKS

By

Xi Yang

Channel bonding, which assembles multiple narrow channels into one logical channel, can speed up data transmission and achieve better bandwidth utilization in wireless networks. Since introduced in 802.11n, channel bonding has been extended continually to support wider channels, making low-lag high-speed wireless communication possible. However, different radio technologies have different requirements on channel width. Devices that use different channel widths coexist in a contention domain may cause inefficiency and unfairness issues. For example, narrowband devices are easier to obtain medium access opportunities because they do not need to wait for the entire wide band to be idle. Therefore, although wideband devices can provide higher transmission speed, they are at an unfavorable position in contentions with narrowband devices.

To this end, we propose a flexible spectrum use channel bonding (FSUB) protocol in which a node is allowed to start a transmission whenever there are some idle narrow channels and gradually increases the channel width during transmission. Because a node dynamically adjusts the channel width in a communication, we use a convolution method to achieve fast spectrum agreement between the transmitter and the receiver. To address contentions between devices in a wide band of spectrum, we introduce a compound preamble to make the collisions detectable in the frequency domain and use a parallel bitwise arbitration mechanism to quickly determine the winner. We implement and evaluate the proposed protocol through both the GNU Radio/USRP platform and ns-2 simulations. The results show that the proposed protocol well addresses the inefficiency and unfairness issues caused by heterogeneous radio coexistence. Channel bonding devices using FSUB have more medium access opportunities and can aggregate wider channels than using other channel bonding protocols in presence of narrowband interference. The FSUB enables a device to always benefit from channel bonding without concerns about narrowband interference level.

ACKNOWLEDGEMENTS

I would like to express the deepest appreciation to my advisor, Professor Li Xiao for her continual support. Without her guidance and persistent help this dissertation would not have been possible.

I would also like to thank my committee members, Professor Abdol-Hossein Esfahanian and Professor Sandeep Kulkarni, for their valuable comments and guidance.

In addition, a thanks to my collaborators and classmates, Pei Huang and Chin-Jung Liu for sharing their technical knowledge and selfless assistance.

Finally, I am grateful for my caring, loving and supportive parent and husband for supporting me all the time.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Motivation

The demand for high speed is pushing wireless communication toward wider channels. Channel bonding is an efficient method that increases data rate regardless of other technologies in use. To support high speed wireless communication, the 802.11 standards extend channel bonding to wider radio frequency(RF) bandwidths, from 40 MHz in 802.11n to 80, and even 160 MHz under certain conditions in 802.11ac. However, the requirements of channel width for various applications are different. Inefficiency and unfairness issues arise when heterogeneous radios coexist in a contention domain. When a device operates in a wideband channel that spans across multiple narrow channels, it has to defer its transmission to the time when all of the narrow channels are idle. This is inefficient because the device cannot utilize other narrow channels when only one narrow channel is occupied. Further, when various narrowband devices independently work in different narrow channels, the wideband devices that use channel bonding to get wider channels are harder to win medium access opportunities.

To address the issue, a natural solution is to use each narrow channel independently. WiFi-NC [4] introduces a compound radio design that splits a channel into multiple narrow channels and use them independently. The strategy is efficient when a device has small packets to multiple receivers. However, when a device has a bulk of data to one receiver, it is more efficient to bond multiple narrow channels as one logical wide channel by removing the guard bands between contiguous narrow channels. This increases the spectral efficiency. Although WiFi-NC tries to reduce the guardbands with sharp filters, signal spreading introduced by sharp filters will increase the inter symbol interference (ISI)[4]. Addressing the ISI requires a tradeoff between spectral efficiency and capability of tolerating frequency offset. Further, it demands higher processing power and more resources to support multiple parallel signal processing flows.

On the contrary, spectrum-agile designs [13] [18] [19] have shown that it is practical to aggregate noncontiguous narrow channels as one channel. Therefore, a wideband device does not need to wait for the periods when all narrow channels are idle. It can use any available narrow channels to achieve the channel bonding. However, current spectrum-agile designs are frame-based. When a new channel becomes available, it cannot be utilized immediately until the next frame. An unfairness issue arises as not all nodes get contention opportunities for the new channel. The new channel is only visible to nodes that are not in transmission. Some nodes may never be able to acquire the channel. Moreover, current spectrum-agile designs use carrier sense multiple access with collision avoidance (CSMA/CA) to address the multiple access issue. It is not efficient in wideband spectrum contention because there are many contending transmitters. When a narrow channel becomes available, a lot of nodes may attempt to acquire it at the same time. A large contention window is needed in CSMA/CA in order to reduce collisions. However, a large contention window will lead to high overhead and the contention window is hard to decrease because a transmission may fail even when two nodes have a collision only in a small fraction of the used spectrum. Low collision probabilities are required.

In addition, after a transmitter has won some narrow channels, it needs to inform the receiver of the used channels; otherwise, the receiver is unable to decode packets sent by the transmitter. A common way to address the issue is to send control packets in a channel on which all nodes have agreed before. The method can introduce high overhead because of medium access contention and physical layer convergence procedure. A better spectrum agreement method is highly demanded.

1.2 FSUB Challenges

In this thesis, we introduce a flexible spectrum use channel bonding (FSUB) protocol in which a node is allowed to start transmit data when it has some idle narrow channels and gradually adds new idle narrow channels into its transmission. This design imposes three major challenges.

First, when some narrow channels are available, the medium access contention is severe because a transmitter has to contend with nodes that are within the same band of spectrum and nodes whose spectrum is partially overlapped with its spectrum. Therefore, it is critical to address the contention issue in wideband spectrum sharing, which is the key to ensure that nodes will benefit from channel bonding. In this thesis, a compound preamble is designed to make collisions detectable in the frequency domain and a parallel bitwise arbitration is designed to quickly resolve the collisions in the time domain. Nodes are allowed to contend for various channels with different priorities at the same time. A node is unlikely to lose all channels in a contention. The time used to resolve collisions on all channels has an upper bound and most of the time it is shorter than the designed maximum length. We show that the bitwise arbitration provides very low collision probabilities with low overhead.

Second, FSUB gradually aggregates the narrow channels obtained by a transmitter as one wide channel. To enable communication over uncertain channels, the receiver needs to know the channels used by the transmitter; otherwise, the receiver cannot decode any packet sent by the transmitter. Since the idle narrow channels are added during transmission, we demand a fast spectrum agreement between the transmitter and the receiver. To solve this problem, we utilize a common Fourier transform pair and the convolution technique. The transmitter encodes the receiver's unique sequence in the frequency domain in all new channels that the transmitter just won. The receiver calculates the expected results when its sequence is present in each channel. Through convolving received signal with all expected results, the receiver is able to identify channels used by the transmitter. Although there are n expected results when the target wide band is divided into n narrow channels, the searching for the signature in all channels can be done at the same time.

Third, the transmitter was unable to monitor the medium state while it is transmitting. Therefore, current spectrum-agile designs [13] [17] [18] [19] [3] are frame-based. This raises an unfairness issue for contending channels. Because a new available narrow channel is invisible to nodes that are in transmission, the channel may be acquired by another node. It is possible that channels used by two nodes are never available to each other because their contentions are not synchronized. However, the condition has been improved with recent advances in self-interference cancelation

[6] [2], which have made full duplex wireless communication possible. This allows a transmitter to parallelize transmission and detection. It becomes feasible to detect new spectrum availability during transmission without using another antenna. Therefore, FSUB can break the frame-based framework, updating spectrum use during transmission.

This thesis addresses the three challenges and integrates all components as one complete system. We implement it on the GNU Radio / USRP platform to validate its effectiveness. Experimental results show that the proposed FSUB is practicable. In addition, extensive simulations in ns-2 demonstrate that the FSUB significantly improves the efficiency of wideband spectrum sharing.

1.3 Organization

The remainder of the thesis is organized as follows. Chapter 2 presents the related work. Chapter 3 gives an overview of the FSUB design. Chapter 4 and Chapter 5 introduce the details of two major components: spectrum contention and spectrum agreement. The performance evaluation by experiments and simulations is presented in Chapter 6. We finally conclude this thesis and present the future work in Chapter 7.

CHAPTER 2

RELATED WORK

With the rapid growth of wireless traffic, there arises a need to increase channel width to achieve high speed wireless communication. However, the radio spectrum is a finite resource and certain parts are preferable due to desirable propagation properties and antenna design constraints. To satisfy the growing demands on channel width, spectrum sharing is unavoidable and reasonable. Because different applications demand different bandwidth and energy efficiency requirements, a variety of channel widths have been defined in different standards. Since many heterogeneous devices have to coexist in a contention domain and their channel widths are different, it is difficult to occupy a wide band of spectrum for exclusive use. Therefore, it is critical to address the inefficiency and unfairness issues caused by heterogeneous radio coexistence.

2.1 Spectrum Use Strategies

A natural solution is to revert to narrow channel operations based on two observations. First, a wideband device can start transmission only when all used channels are idle. If a small part of the target spectrum is occupied by another node, it cannot transmit data [4] [19]. The availability of narrow channels is much higher than that of wide channels due to the presence of many uncoordinated narrowband devices [3]. It is thus attractive to use spectrum in the unit of narrow channels. Second, when a node has a small amount of data to transmit, the high PHY data rate achieved by wide channel increases the proportion of medium access overhead in data transmission [17]. Motivated by these observations, various designs have been proposed to improve communication efficiency by using narrow channels.

WiFi-NC [4] introduces a compound radio design that splits a channel into multiple narrow channels and use them independently. WiFi-NC adopts the orthogonal frequency-division multiplexing (OFDM) for transmission. It implements multiple receiving chains and transmitting chains in a radio, which demands a significant amount of FPGA resources. To reduce overhead generated by setting guardbands between contiguous narrow channels, sharp filters are designed. The sharp filter relies on spreading (smoothing) signal in time can reshape the spectrum. It will decrease the width of guardbands and thus increase the width of narrow channels. However, this aggravates the inter symbol interference (ISI). To address the ISI issue, a common way is to increase the cyclic prefix (CP). Due to the spreading, WiFi-NC needs a longer CP for each OFDM symbol. The spectral efficiency is reduced as the CP does not carry data. The CP only prefixes an OFDM symbol with a repetition of the end. To maintain the spectral efficiency after extending the CP, WiFi-NC increases the number of subcarriers, but it reduces the subcarrier spacing and width, making WiFi-NC sensitive to frequency offset and ill-suited for mobile scenarios where the Doppler effect is obvious. Further, it duplicates multiple signal processing flows that would be redundant if data in all narrow channels are from the same transmitter.

In one-to-one transmissions, it is more efficient to bond multiple narrow channels as one wide channel. This removes the guard bands between contiguous narrow channels and reduces the complexity of wireless communication. However, with a wide channel, the data transmission efficiency decreases because random backoff for collision avoidance is no longer efficient for small packets in high-speed transmission. FICA [17] thus allocates channel width according to the typical frame size and the physical layer (PHY) data rate achieved at a certain channel width. Although in FICA multiple nodes send simultaneously in orthogonal narrow channels, each narrow channel is a part of a wide band. The entire wide band has to be detected to be idle before a new transmission can be commenced. The method is thus subject to the adverse impact of heterogeneous radios.

2.2 Spectrum Agreement

In OFDM, we can deactivate some subcarriers by feeding 0 instead of modulation symbols (e.g., 1+0i, -1+0i for BPSK) to the corresponding subcarriers. The non-contiguous OFDM (NC-OFDM) technique allows a node to use some but not all subcarriers. As a result, instead of waiting for the entire wide band to be idle, a node can transmit on spectrum fragments that are currently available. However, communication over uncertain subcarriers imposes a spectrum agreement challenge. The transmitter interleaves bits across all used subcarriers and the receiver reconstructs the original packet by extracting bits from all used subcarriers. If the receiver does not know the subcarriers used by the transmitter, it may get insertions or deletions of bits from the data stream. The unknown insertions and deletions of bits make it impossible for the receiver to decode the packet. Therefore, it is critical to achieve spectrum agreement between the transmitter and the receiver.

To address the issue, earlier spectrum-agile designs [13] [18] use control packets. This means a dedicated control channel must be available, or the control packets can be send only when the entire wide band or a specific channel becomes available, which works against the spectrum-agile feature. The exchange of control packets also introduces high overhead because of the random backoff in collision avoidance and the preamble overhead in physical layer convergence procedure (PLCP).

RODIN [3] divides a wide band of spectrum into n nonoverlapping narrow channels and reshapes the spectrum of a frame to n_f ($< n$) of these narrow channels. RODIN supports direct connection to commercial off-the-shelf (COTS) devices but it has overhead of setting guard bands similar to WiFi-NC [4]. Because narrow channels are isolated in RODIN, the transmitter transmits a set of sequences simultaneously in all used channels. Each sequence is unique in a narrow channel. The receiver can identify the channels used by the transmitter through time domain signal correlation. RODIN needs multiple filters to isolate each channel, and it is not easy to determine how many narrow channels are used by the transmitter. RODIN thus fixes the number to n_f. The receiver must identify all of the n_f narrow channels in each transmission. The method does not adapt to a node's traffic where fewer or more channels are needed.

CHAPTER 3

OVERVIEW OF FLEXIBLE SPECTRUM USE CHANNEL BONDING

Flexible Spectrum Use Channel Bonding (FSUB) imposes three major challenges: *how to detect newly available channels during transmission, how to contend for available channels, and how to achieve spectrum agreement between the transmitter and the receiver*.

In this chapter, we take a quick view of our design that addresses these challenges. The FSUB system includes four phases: spectrum contention, spectrum agreement, synchronization preamble detection, and changing channel width. The first two phases are the core design of the protocol. We will present detailed designs of the two phases in Chapter 4 and Chapter 5, respectively.

3.1 Spectrum Contention

We assume that a wide band of spectrum is divided into *n* narrow channels of equal size for independent clear channel assessment (e.g., 6 MHz channels in the TV bands). In the remaining part of the thesis, a channel means a narrow channel unless otherwise stated.

Previously, when a node starts transmitting, it was unable to receive other signals simultaneously with the same RF front end and antenna because the self-interference is so strong that it buries other signals. However, recent advances in self-interference cancelation [10] [6] show that new radio designs allow simultaneous transmission and reception even with the same antenna. Therefore, a transmitter can monitor the medium state even when it is transmitting its own data. This allows all nodes to perceive a channel that changes from busy to idle. As a result, all nodes get equal chances to occupy the new channel.

Before commencing a transmission, the transmitter uses the power spectral density (PSD) of the wide band to check which narrow channels are available in the target spectrum [18] [19]. Once available channels are identified, the transmitter contends for data transmission in these channels. Because there may exist many transmitters in a wide band of spectrum, medium access contention based on carrier sense multiple access with collision avoidance (CSMA/CA) may need a large contention window (CW) to ensure low collision probabilities. However, a large contention window leads to a long average backoff time, significantly reducing the efficiency. We thus introduce a parallel bitwise arbitration mechanism to reduce the medium access overhead and provide low collision probabilities. The parallel bitwise arbitration also allows a node to concurrently contend for different channels with different priorities. A node is unlikely to lose all channels in a contention.

The highlight portions of Fig. 3.1 show the architecture and process of our spectrum contention design. On transmitter part, when a node has data to send and there are some idle narrow channels, the compound preamble construction module will generate a compound preamble according to the transmitter's unique sequence and a random number. In addition, when the data packet are transmitting in the channels won by the transmitter, the compound preamble can be sent in other idle channels for spectrum contention. The data packet and the compound preamble are mixed together. The mixed data are converted to OFDM symbols through the S/P Converter and the IFFT module. After adding the cyclic prefix to each symbol, they are sent out. On receiver part, when a node gets an OFDM symbol, the FFT module will transform the data from the time domain to the frequency domain. The Clear Channel Assessment (CCA) checks which channels are idle and sends the medium state result to the compound preamble construction module for generating the compound preamble. At the same time, FSUB system detects collisions in the frequency domain. If there are collisions, FSUB will use the Parallel Bitwise Arbitration (PBA) module to resolve the collisions in all idle channels simultaneously. In our spectrum contention design, compound preamble construction, collision detection, and parallel bitwise arbitration are the core modules. We will introduce them in section 4.2.2, section 4.2.4, and section 4.2.5, respectively.

3.2 Spectrum Agreement

After a transmitter wins some narrow channels, it needs to inform the receiver of the used channels; otherwise, the receiver is unable to decode packets sent by the transmitter. To achieve fast spectrum agreement between the transmitter and the receiver, we utilize convolution and a com-

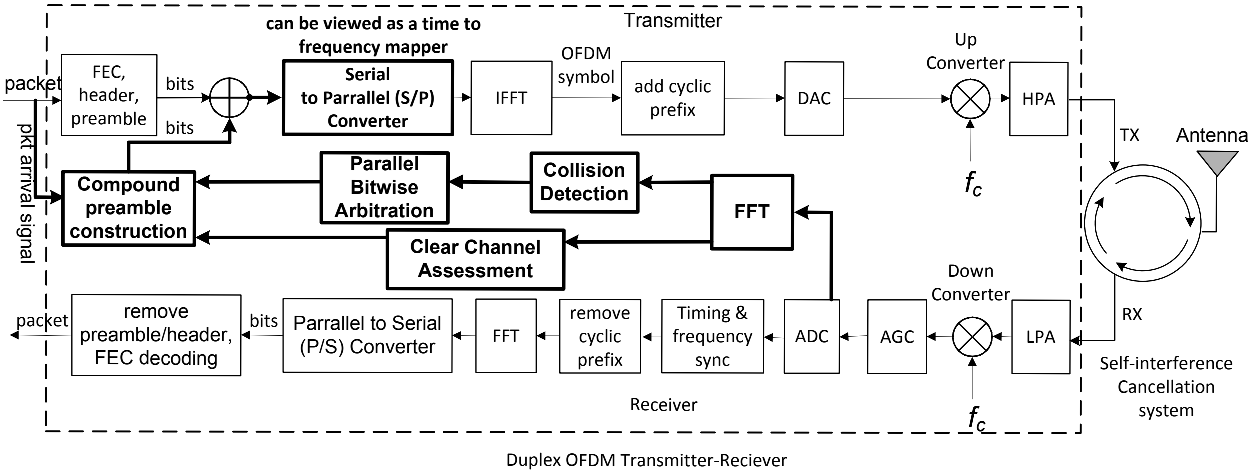

Transmitter
can be viewed as a time to frequency mapper
packet
FEC, header, preamble
bits
bits
Serial to Parrallel (S/P) Converter
IFFT
OFDM symbol
add cyclic prefix
DAC
Up Converter
HPA
TX
Antenna
pkt arrival signal
Compound preamble construction
Parallel Bitwise Arbitration
Collision Detection
FFT
Clear Channel Assessment
f_c
packet
remove preamble/header, FEC decoding
bits
Parrallel to Serial (P/S) Converter
FFT
remove cyclic prefix
Timing & frequency sync
ADC
AGC
Down Converter
LPA
RX
Self-interference Cancellation system
Receiver
f_c
Duplex OFDM Transmitter-Reciever

Figure 3.1: The architecture and process of FSUB spectrum contention design.

mon Fourier transform pair. The transmitter maps the receiver's unique sequence to subcarriers of each idle channel it just won. The receiver keeps convolving received signals with the signals that are expected when its unique sequence is present in each channel. If the receiver detects a convolution spike, it learns that the transmitter plans to use the channel for data transmission. With the efficient spectrum agreement method, the receiver is able to identify channels used by the transmitter and filter out unwanted signals in other channels, allowing the receiver to restore standard OFDM preamble detection on clean signals from the transmitter. The receiver uses the same method to inform the transmitter of confirmed channels. Once they agree on channel use, the transmitter bonds all selected channels together for data transmission.

3.3 Synchronization Preamble Detection

The spectrum agreement works without knowing the boundary of each OFDM symbol, but to decode a packet sent by the transmitter, the receiver must synchronize to the start of each OFDM symbol and estimate the frequency offset. OFDM is designed to work in a contiguous band. The preamble detection is based on a time domain delayed correlation property embedded in the preamble . Because some narrow channels in the wide band are occupied by other transmissions, the delayed correlation property may not hold on the mixed signals. Therefore, it is important to filter out unwanted signals in the channels that are not used by the transmitter so that the synchronization preamble can be successfully detected. The spectrum agreement has identified channels used by the transmitter and thus made the isolation possible.

Once the bitwise arbitration ends, the transmitter applies a random cycle shift on the receiver's unique sequence and sends it in all channels it has won. The receiver confirms that the channels are available. The receiver then starts to filter out signals that are not in channels used by the transmitter. The transmitter transmits the OFDM synchronization preamble as usual once it obtains the confirmation. The receiver is able to detect the OFDM synchronization preamble on the clean signals and it then follows the standard OFDM transmission.

3.4 Changing Channel Width

Because a node can monitor the medium state even during transmission as long as the ADC is not saturated, when new channels become available, the node performs the medium access procedure in new channels without interrupting the current transmission by using the NC-OFDM. Since the receiver filters out signals that are not in currently used channels. Even when the transmitter is sending some symbols in other channels for contention, the current transmission will not be affected. If the transmitter wins some new channels, it informs the receiver and gets confirmation through the spectrum agreement mechanism. The receiver then changes to receive data in a wider bonded channel. The OFDM synchronization preamble needs to be transmitted again to achieve the synchronization and the channel estimation in a wider bonded channel.

CHAPTER 4

SPECTRUM CONTENTION

This chapter introduces the details of the spectrum contention design in FSUB. The spectrum contention design has three main functions: (1) a node will detect idle narrow channels even during transmission; (2) once it detects idle narrow channels, it will probe collision on these idle narrow channels; (3)if the contention channel has collision, it triggers the parallel bitwise arbitration to quickly resolve the collision and determine the winner who can occupy the channel.

4.1 Motivation

To show the benefits of our spectrum contention design, we analysis the limitations of Collision Avoidance (CA) in 802.11 and the high collision probability in current Collision Detection (CD) designs. Although we use Wi-Fi as an example, the idea can be generalized to other wireless networks that use multicarrier modulation.

4.1.1 802.11 Medium Access Overhead

The fundamental medium access method of IEEE 802.11 MAC is a distributed coordination function (DCF) that relies on the CSMA/CA. Due to physical constraints (e.g., processing delay, TX/RX turnaround time), a gap is inserted between contiguous frames in CSMA/CA. A node thus must sense that the medium is idle for at least such a length of duration before attempting to transmit. As shown in Fig. 4.1, the CSMA/CA algorithm mandates that a node must verify that the medium is idle for a duration of DIFS (DCF interframe space), which comprises a SIFS (short interframe space) and 2 slots. After the DIFS, a node shall defer its transmission for a random number of slots. The random number is drawn from a uniform distribution over the interval [0,CW], unless the random number generated in the last contention has not been decreased to 0.

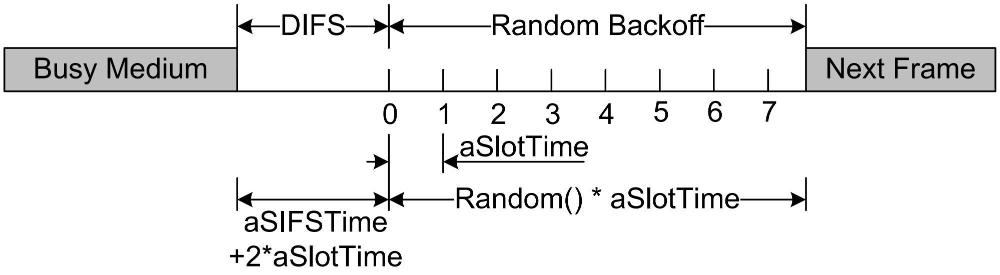


Figure 4.1: 802.11 DCF.

Table 4.1: OFDM PHY characteristics

aSlotTime	9 μs	aSIFSTime	16 μs
aRxTxTurnaroundTime	< 2 μs	aCCATime	< 4 μs
aAirPropagationTime	$\ll 1$ μs	aCWmin	15
aMACProcessingDelay	< 2 μs	aCWmax	1023

It is possible that two nodes defer their transmissions to the same slot, leading to a collision. The two nodes are unaware of the collision until they complete their transmissions. Without a collision detection mechanism, they can only infer a collision from their unacknowledged data packets. To reduce the collision probability, the CW takes an initial value of aCWmin and then is increased exponentially every time an unsuccessful transmission occurs, until the CW reaches the value of aCWmax. The CW cannot be reset to aCWmin unless the node successfully transmits a packet. The process makes the CW vary with the load, quickly switching to a large size when the channel contention is severe.

Table 4.1 summarizes the parameters defined in IEEE 802.11 standard [1]. To ensure accurate detection of busy state and leave enough time for radio receive-transmit switch, the slot size defined in IEEE 802.11 standard is close to the minimum feasible value. Even if there is no collision and the CW stays at the aCWmin, the average backoff time incurred by CSMA/CA is $aDIFS + aCWmin/2 * aSlotTime = aSIFSTime + (2 + aCWmin/2) * aSlotTime = 16s + (2 + 7.5) * 9s =$ 101.5 μs for the OFDM PHY, but the data rate with the OFDM PHY can reach 600 Mbps in 802.11n, leading to a transmission time of 20 μs for a 1500 byte packet. Therefore, the medium access overhead becomes a major factor that causes the inefficiency of Wi-Fi.

4.1.2 High Collision Probability in CD

With the emerging full duplex wireless communication and the NC-OFDM, collision detection is made possible by analyzing received signals in the frequency domain. OFDM divides a wide band of wireless spectrum into multiple narrowband channels known as subcarriers. Each subcarrier carries a data stream in parallel with others. Since multiple data streams are transmitted simultaneously, each data stream can be modulated at a low rate while maintaining a high aggregated rate. With NC-OFDM, a node can map data streams only to a subset of subcarriers. A node thus can easily detect a collision if it observes activities on subcarriers that are not used by itself. Based on the idea, recent CD proposals [17] [14] let a node transmit on a randomly selected subcarrier. Contending nodes learn each other's random number in the frequency domain and the node with the largest random number wins. In practice, however, only a few subcarriers are available for contention resolution. The probability that two nodes choose the same subcarrier is high. As a result, the collision cannot be detected and both nodes will start data transmission. Therefore, although medium access overhead is reduced, retransmission overhead may be increased.

With a few number of subcarriers available for contention resolution, we introduce a binary mapping from a random number to active/inactive subcarriers to decrease the collision probability. The winner in a contention is selected through a bitwise arbitration mechanism, which forces low priority nodes to abort their transmissions gradually. Most of the time only one node survives and proceeds to data transmission. The bitwise arbitration is very efficient because it allows the winer to transmit data immediately when the collision is resolved. The arbitration phase is on average shorter than the designed length.

4.2 The Design of Spectrum Contention

FSUB employs OFDM to opportunistically utilize spectrum fragments in a wide band of spectrum. Besides contending transmitters in the same band of spectrum, there are transmitters contending for partially overlapped spectrum as shown in Fig. 4.2 (difference applications may choose different

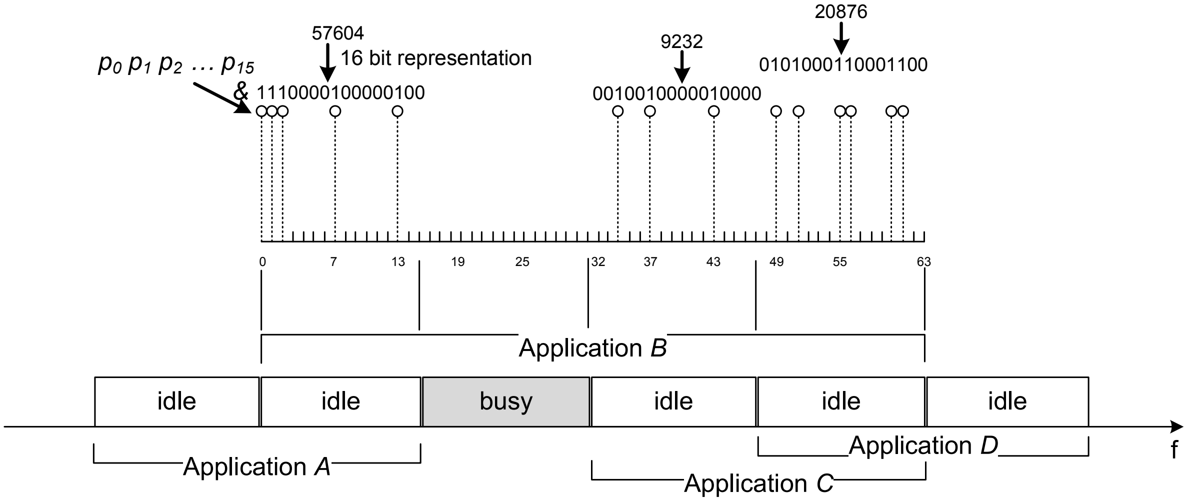


Figure 4.2: Binary code mapping with NC-OFDM for collision detection.

center frequencies and bandwidths). A collision in a small fraction of the used spectrum may cause the entire transmission to fail. The cost is high and thus a large contention window may be needed in CSMA/CA in order to reduce the collision probability. However, a larger contention window leads to a longer period of backoff on average. A parallel bitwise arbitration mechanism is introduced in this thesis to provide low collision probabilities with low overhead.

The bitwise arbitration relies on collision detection [8] [9]. When a node has data to send, it first checks which channels are available. It then transmits a compound preamble that occupies all of the channels but the preamble is designed to set some subcarriers to inactive as shown in Fig. 4.2. There are two conditions in which we set the subcarriers to inactive. First, some channels are busy, hence all subcarriers in those channels are set to inactive. Second, it intentionally deactivates some subcarriers in available channels for collision detection.

4.2.1 Deactivating Subcarriers

OFDM employs inverse fast Fourier transform (IFFT) to efficiently modulate multiple orthogonal subcarriers at the same time. A bit stream is first transformed into a stream of modulation symbols (e.g., 1+0i, -1+0i when BPSK is used). The stream of modulation symbols is divided into vectors of *N* modulation symbols. These vectors are inputted to a *N*-point IFFT module one by one. The

IFFT returns the inverse discrete Fourier transform (DFT) of each vector, which consists of N samples that are referred to as one OFDM symbol in the time domain. The receiver can recover the modulation symbols through performing FFT on the received OFDM symbol.

To deactivate some subcarriers, we feed 0 instead of modulation symbols to the IFFT module. In a vector of N modulation symbols, the ith symbol is used to modulate the ith subcarrier. If we replace the ith symbol with 0, this leads to zero power at the ith subcarrier. Therefore, when we want to use only M out of N subcarriers, we take M symbols from the modulation symbol stream and map them to active subcarriers. Other values in the vector are set to 0. For example, suppose $N = 4$ and the vector of N modulation symbols are $1+0i, -1+0i, -1+0i, 1+0i$. To deactivate the 3rd subcarrier, we feed 0 instead of the 3rd symbol $-1+0i$ to the IFFT module. As a result, the modulation symbols become $1+0i, -1+0i, 0, 1+0i$.

4.2.2 Compound Preamble Design

In the frequency domain, a compound preamble occupies the entire wide band with some subcarriers set to inactive [8] [7] [9] . In the time domain, a compound preamble is one OFDM symbol that consists of N samples when a N-point IFFT is used. Suppose the target wide band is composed of n narrow channels. Each channel contains $k = N/n$ subcarriers. A unique sequence of k symbols $\{p_0 p_1 p_2 ... p_{k-1}\}$ is assigned to each node. For collision detection, the actual sequence is of no meaning. We thus postpone the discussion about the sequence to the introduction of spectrum agreement. Here we focus on how to create high magnitudes at some but not all subcarriers.

If a node repeats its unique sequence by n times ($k \times n = N$ symbols) and performs N-point IFFT, it generates a compound preamble that causes high magnitudes at all N subcarriers. To deactivate some subcarriers for collision detection, a node draws a random number from a uniform distribution over the interval $[1,2^k - 1]$ for each available channel. For busy channels, the number 0 is used to make sure that all subcarriers in these channels are set to inactive. Because each random number can be represented by a k bit binary code, the binary code is bitwise AND with the node's unique sequence. As we feed some 0s to the IFFT, not all subcarriers in each available channel are used and all subcarriers in busy channels are set to inactive as shown in Fig. 4.2. The binary representation of each number is essentially a map of active and inactive subcarriers in the corresponding channel with ‘1’ indicating active and ‘0’ indicating inactive.

In fact, the random number for each contending narrow channel indicates the priority of a node in the channel. The node that has the largest random number will win the channel. Since a node can contend all available channels at the same time with different priorities, for each idle channel, the node will generate a different random number. In Fig. 4.2, channel 1,3,4 are available. The transmitter generates 57604 for contending channel 1, 9232 for channel 3, and 20876 for channel 4. Since random numbers are different for different channels, the node is unlikely to lose all channels at the same time. Fig. 4.3 shows an example of compound preamble generation process.Suppose the wide band is split into 4 narrow channels and a 16-point FFT is applied on the entire band. Each narrow channel has 4 subcarriers (k=4). A compound preamble can be generated based on the medium state, random numbers and the unique sequence of a node. After a compound preamble is generated, it is mixed with the data and sent to the receiver.

With the compound preamble design, a collision is detectable if a node observes high magnitudes at subcarriers that are deactivated by it. The collision detection fails only when two nodes choose the same random number, but the probability is low as it decreases exponentially with the increased subcarrier number k. For example, assuming a 64-point IFFT is used on a 20 MHz band that comprises four 5 MHz channels, each channel contains 16 subcarriers. With the binary mapping, in total $2^{16}-1$ random numbers can be represented (0 is excluded). The probability that two nodes choose the same number for a channel is very low. To get the same low collision probability in CSMA/CA, the contention window (CW) will be too large to be practical.

4.2.3 Collision Probability

The advances in self-interference cancelation have motivated some frequency domain backoff designs [17] [14]. Both FICA [17] and Back2F [14] use one subcarrier to represent the priority of a node. Nodes learn each other’s priority through the occupied subcarrier. The node with the highest

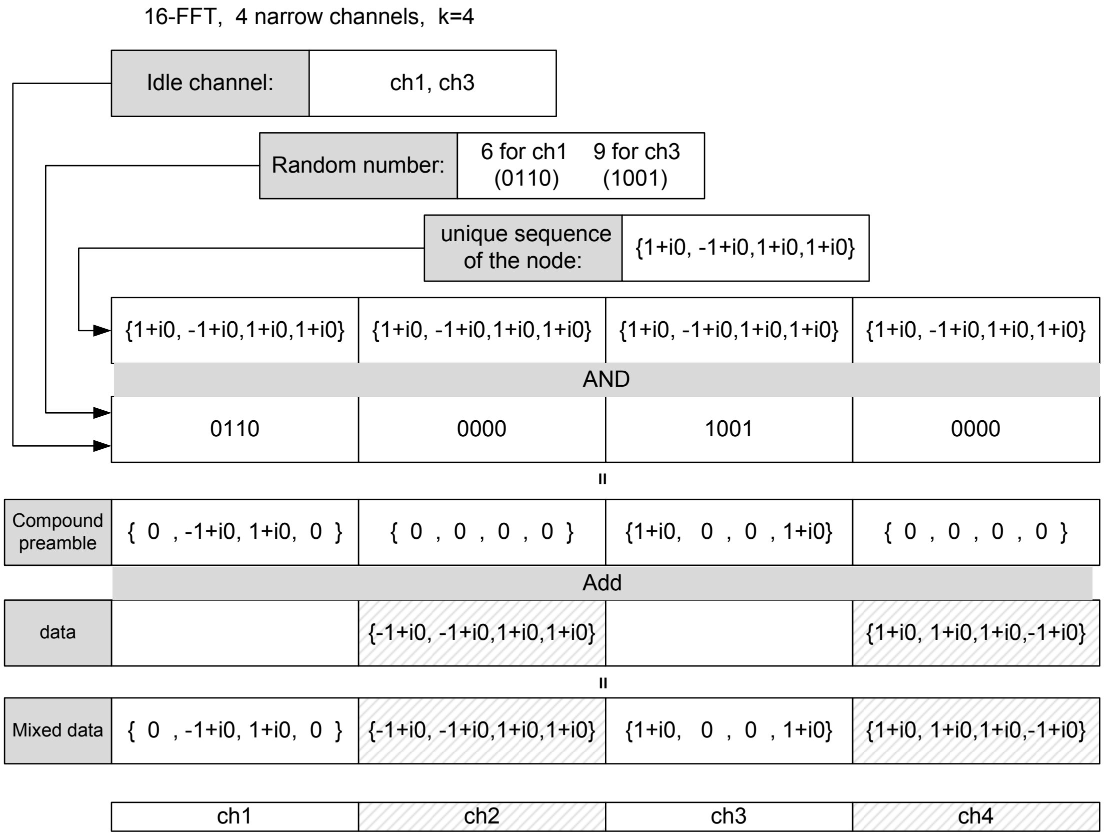


Figure 4.3: An example of compound preamble generation process.

priority can transmit. Because the number of subcarriers available for contention may be small, the probability that two nodes choose the same subcarrier is high and the collision will not be detected. Back2F [14] uses two rounds of contention to address the issue.

Suppose there are N subcarriers for contention and m contending transmitters. Any of the N numbers can be the highest priority. If the ith largest number becomes the highest priority, it means that no node chooses a number greater than the ith largest number. All nodes choose a number only in the remaining $N-i$ numbers, and the probability is $(\frac{N-i}{N})^m$ where $i = 0, 1, 2, ..., N-1$.

Under the condition, the probability that at least two nodes choose the ith largest number is

$$P_i = (\frac{N-i}{N})^m \sum_{j=2}^{m} \binom{m}{j} (\frac{1}{N-i})^j (\frac{N-i-1}{N-i})^{m-j}, \tag{4.1}$$

where

$$p = \binom{m}{j} (\frac{1}{N-i})^j (\frac{N-i-1}{N-i})^{m-j} \tag{4.2}$$

is the probability that exactly j nodes choose the ith largest number while others choose numbers in the remaining $N-i-1$ numbers.

The j collided nodes enter in the second round of contention and again they can choose any of the N numbers. The collision probability in the second round of contention is

$$P_j = \sum_{i=0}^{N-1} (\frac{N-i}{N})^j \cdot p_j, \tag{4.3}$$

where

$$p_j = 1 - (\frac{N-i-1}{N-i})^j - \binom{j}{1}(\frac{1}{N-i})(\frac{N-i-1}{N-i})^{j-1} \tag{4.4}$$

is the probability that at least two nodes choose the ith largest number.

Therefore, the collision probability after two round of contention is

$$P_{Back2F} = \sum_{i=0}^{N-1} P_i P_j. \tag{4.5}$$

In our design, we use only one round of contention, but the collision probability is reduced by increasing the contention window. A number is represented by multiple subcarriers. The collision probability is

$$P_{FSUB} = \sum_{i=0}^{N'-1} (\frac{N'-i}{N'})^m \cdot p_m, \tag{4.6}$$

where p_m is the probability that at least two nodes choose the ith largest number (i.e, Equation 4.4). Because N subcarriers can represent $N' = 2^N$ numbers, the collision probability of FSUB is much lower than that of Back2F as shown in Fig. 4.4.

Since we are targeting at wideband spectrum contention, the number of contending nodes may be high. It is thus important to maintain a low collision probability in the severe contention conditions. In addition, using multiple subcarriers reduces the probability of misdetection due to frequency-selective fading.

4.2.4 Collision Detection

Because collisions are detectable in the frequency domain, instead of deferring nodes' transmissions randomly for collision avoidance, we allow a node to transmit immediately when it has

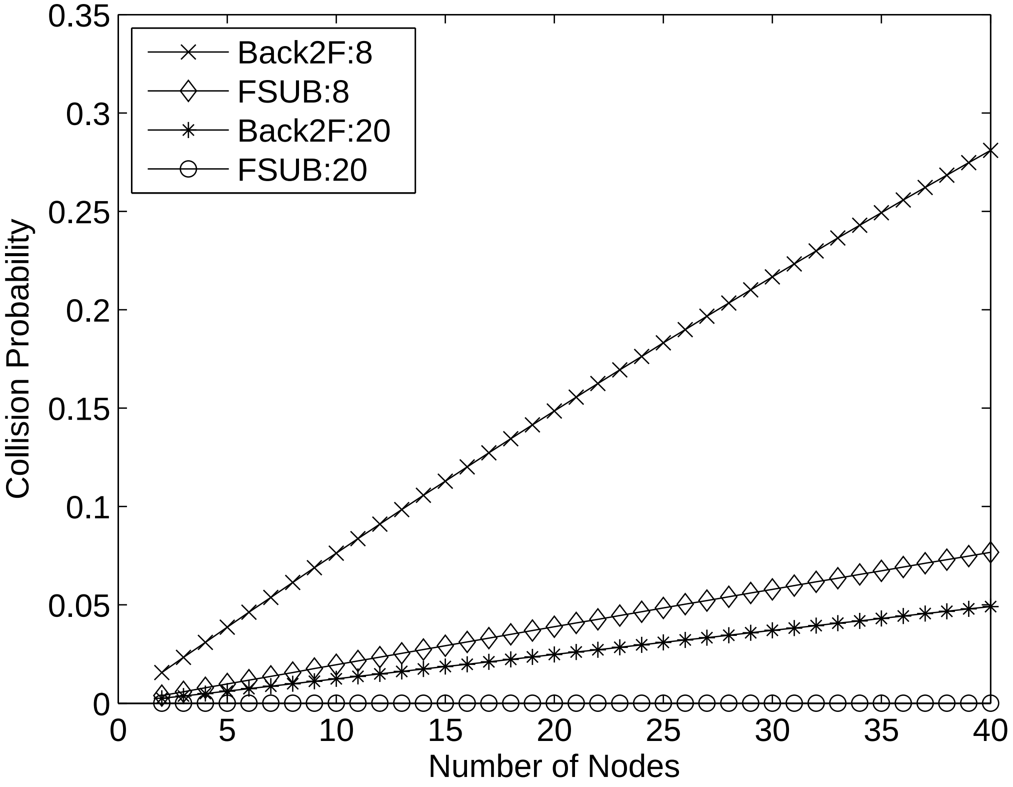


Figure 4.4: The comparison of collision probability between FSUB and Back2F.

won some idle channels. To check whether the bold attempt will cause a collision, a node performs spectrum analysis on received signal while it is transmitting its compound preamble. The simultaneous transmission and reception is made possible by recent advances in self-interference cancelation.

Commercially used 14-bit ADCs have 86 dB of dynamic range where the dynamic range is calculated as $DR(dB) = 6.02 \times n + 1.76dB$ for $n = 14$ [6]. The dynamic range defines the ratio between the highest and the lowest detectable signal power. Considering a typical noise floor -90 dBm (1 picowatt) for Wi-Fi systems , the self-interference cannot be over -4 dBm; otherwise, weak signals from other transmitters cannot be preserved in sampling. However, the highest transmission power on a Wi-Fi 2.4 GHz antenna can be around 30 dBm [6], which is strong enough to bury other weak signals. This is why duplex wireless communication cannot be implemented in the past.

Recent researches in self-interference cancelation have reached at least 45 dB cancelation in

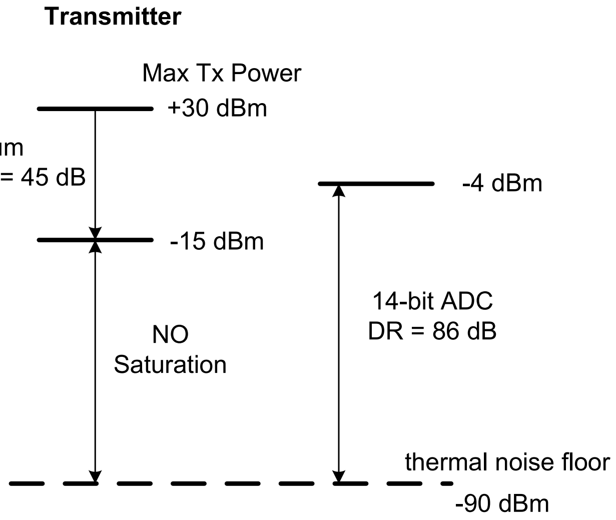


Figure 4.5: Self-interference cancelation is sufficient to prevent ADC saturation.

practice . As shown in Fig. 4.5 , the self-interference cancelation is sufficient to prevent ADC saturation (30 - 45 = -15 dBm < -4 dBm). Although we need more self-interference cancelation to enable simultaneous transmission and reception of packets in the same spectrum fragment, our collision detection does not require such a perfect cancelation. In our design, each node sets some subcarriers to inactive, leading to zero power at these subcarriers. As long as the self-interference does not cause ADC saturation, a node can detect signals from other transmitters at inactive subcarriers if there is a collision. Although a node is still unable to decode packets due to the strong self-interference, the node can learn that there is collision if it detects high magnitudes at subcarriers that are deactivated by it. The collision detection does not incur additional cost of an additional antenna.

4.2.5 Medium Access Procedure

The self-interference cancelation and spectrum analysis make it possible for a transmitter to detect medium state even during transmission. Relying on the two techniques, the proposed parallel bitwise arbitration mechanism can quickly determine the winner in a contention [8] [7] [9]. To see the benefits introduced by self-interference cancelation and spectrum analysis, we first examine how bitwise arbitration can work in the time domain.

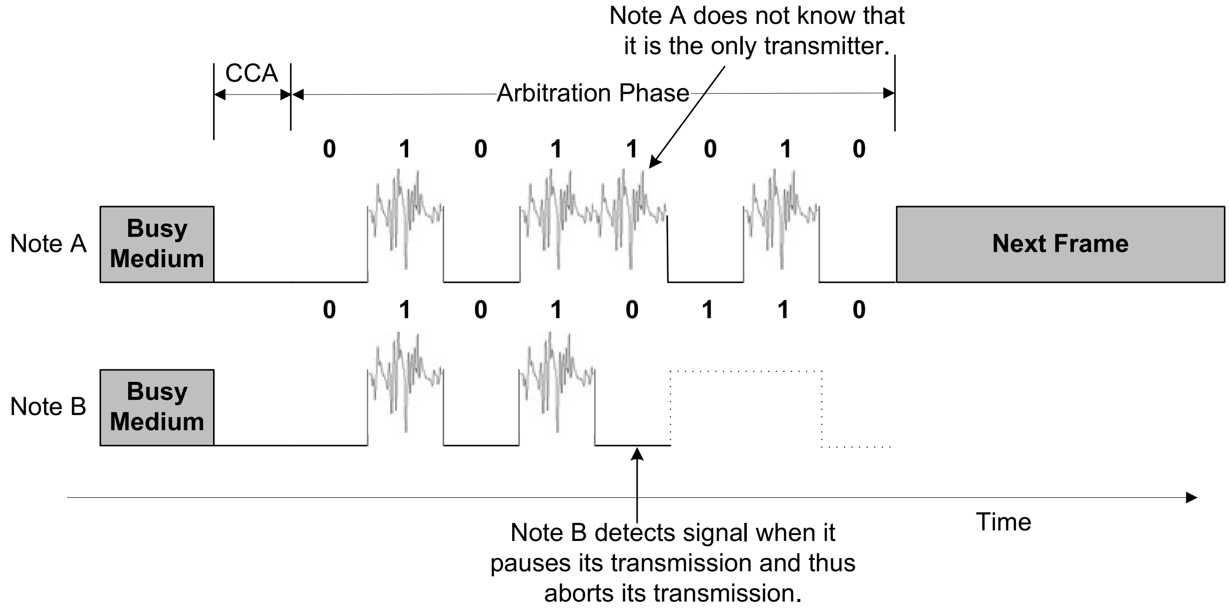


Figure 4.6: Bitwise arbitration works in time domain without self-interference cancelation.

Before we have self-interference cancelation, we have to rely solely on time domain information for bitwise arbitration. A node can be equipped with a fast-switching radio or two radios, one for transmitting and one for receiving. Suppose nodes' contentions are implicitly synchronized by the busy medium. When the medium is detected to be idle, nodes start to transmit their arbitration preambles.

As mentioned before, a node chooses a random number from a uniform distribution over [1, $2^k - 1$] to be its priority in a contention. The number is represented in the form of a k bit binary code. Each node goes through its binary code starting from the most significant bit. If the ith bit is '1', the node transmits its arbitration preamble in the ith OFDM symbol slot (note that the arbitration preamble is only one OFDM symbol); if the ith bit is '0', the node pauses its transmission for one OFDM symbol long. If a node detects signal when it pauses its transmission, it loses the contention and aborts its transmission as shown in Fig. 4.6. Because contending nodes gradually abort their transmissions, only one node will survive at the end of the arbitration phase as long as nodes choose different random numbers.

In this design, a node cannot learn that it has won the medium access in a slot because it is unable to determine whether or not there is any other transmission during its own transmission. As a result, a node has to go through all bits to determine the winner. To reduce the probability that two nodes choose the same random number, the *k* should be increased and thus the length of the arbitration phase is increased. This makes bitwise arbitration less appealing than the CSMA/CA. If we can obtain perfect self-interference cancelation, a node is able to end the arbitration phase early (e.g., the fifth slot in the example). However, this imposes a high requirement on the full duplex capability.

If self-interference cancelation is sufficient to prevent ADC saturation, frequency domain information can be introduced to help detect collisions. As collisions are detectable, it is important to determine which node is the winner of a narrow channel in a contention. Note that each node generates different random numbers for different channels. This gives a node higher chances to win some channels. Fig. 4.7 shows an overview of medium access procedure in FSUB. Although the example shows three nodes, the principle applies to an arbitrary number of contending nodes.

There are two phases: *collision probe* and *bitwise arbitration*. When a node has data to send, it first checks which channels are available. Once the CCA is done, the node transmits its first compound preamble to check whether or not there is a collision in any of the channels. We call this phase *collision probe*.

If there is no collision in a channel, the node wins the channel immediately. It transmits its unique sequence with all subcarriers in the channel to prevent other nodes from taking it. If there is no collision in all contending channels, the bitwise arbitration phase is skipped. However, if there is a collision in any of the channels, the bitwise arbitration is triggered to resolve the collisions.

In the bitwise arbitration phase, the compound preamble of a node is updated based on three conditions. First, for busy channels, all subcarriers are set to inactive. Second, for a channel that is won by the node, all subcarriers are occupied by the node's unique sequence. Third, for a channel in which the winner has not been determined, the *i*th bit of the corresponding binary code is checked for constructing the *i*th compound preamble in the bitwise arbitration phase.

The compound preamble is used to contend for all channels at the same time. If we focus on one arbitrary channel, a node traverses the corresponding binary code starting from the most significant bit. If the *i*th bit is 0, all subcarriers in the channel are set to inactive in the *i*th compound preamble. If the *i*th bit is 1, the node maps its unique sequence to active subcarriers according to the corresponding binary code. The binary code determines the priority of a node in acquiring the channel.

When a node is transmitting its compound preamble, it also performs spectrum analysis on received signals. If it sets all subcarriers in a channel to inactive but observes high magnitudes at subcarriers in the channel, it loses the channel. On the contrary, if the channel is idle, the winner has not been determined and it needs to check the next bit. If a node uses some subcarriers in a channel and the collision still exists (detecting high magnitudes at subcarriers that are deactivated by it), it also proceeds to check the next bit. However, if the node detects that the frequency spectrum of the channel matches its corresponding binary code, the collision has been resolved and the node wins the channel. An advantage of introducing frequency domain information in bitwise arbitration is that the arbitration may end earlier without traversing all bits.

Once a node wins a channel, all subcarriers in the channel are used for the node's unique sequence. The node shall occupy the channel to prevent other nodes from taking it. This also addresses the condition that a binary code may dominate others. For example, if a node selects the binary code 1101, it does not detect another node that selects the binary code 1001. The node with the higher priority wins the channel immediately after the collision probe while the other node enters in the arbitration phase. To let nodes learn that there is no need of arbitration, all nodes agree to not use a selective subset of subcarriers in the bitwise arbitration phase. When a node wins a channel immediately after the collision probe, it keeps transmitting on all subcarriers in the channel while waiting for the arbitration results on the other channels. Because other nodes detect that all subcarriers in the channel are occupied, they abort the bitwise arbitration for the channel. A node initiates its spectrum agreement when bitwise arbitration results (either win or lose) in all channels are obtained.

The time length of a compound preamble is equivalent to one OFDM symbol, which contains N samples when a N-point IFFT algorithm is used. Meanwhile, the spectrum analysis needs N

samples when a N-point FFT algorithm is used. The simultaneous transmission and reception of a compound preamble takes N/B seconds where B is the bandwidth. For example, assuming a 64 point IFFT/FFT algorithm is adopted, it takes 3.2 μs to complete the preamble transmission at 20 MHz. In the same 3.2 μs, the receiving chain completes the collection of 64 samples for the spectrum analysis.

4.3 Exemplify

An example illustrated in Fig. 4.7 shows how bitwise arbitration gradually determines the winner of each channel. Collided transmissions are implicitly synchronized by the collision. Nodes detect collisions in the 1st compound preamble. In the 2nd OFDM symbol, the compound preamble sent by a node is updated according to the first bit of each binary code. For node A, the first bit for channel 1 is 1. It occupies some subcarriers according to the binary code. On the other hand, the first bit of the binary code for channel 3 is 0. All subcarriers in channel 3 are set to inactive. Following the same principle, node B and node C use a binary map where all subcarriers are set to inactive because the first bit of either binary code is 0.

While nodes are transmitting the compound preamble, they also perform FFT on received signals. For channel 1, node A notices that there is no collision and thus it wins the channel. Node B and node C notice that channel 1 is busy when they have no active subcarrier in the channel. They lose the channel. The arbitration for channel 1 is done. Node B and node C will not contend for channel 1 until next round. For channel 3, all nodes detect that the channel is idle. The winner has not been determined. They must check the next bit to determine the winner.

In the 3rd OFDM symbol, node A has won channel 1 and it occupies all subcarriers in channel 1. In channel 3, both node A and node C have no active subcarriers because the 2nd bit of their binary codes is 0. Node B, on the contrary, occupies some subcarriers according to its binary code in channel 3. Since there is no collision, node B wins channel 3. Node A and node C learns that they lose the channel.

The example shows that as long as two nodes do not choose the same binary code for a channel,

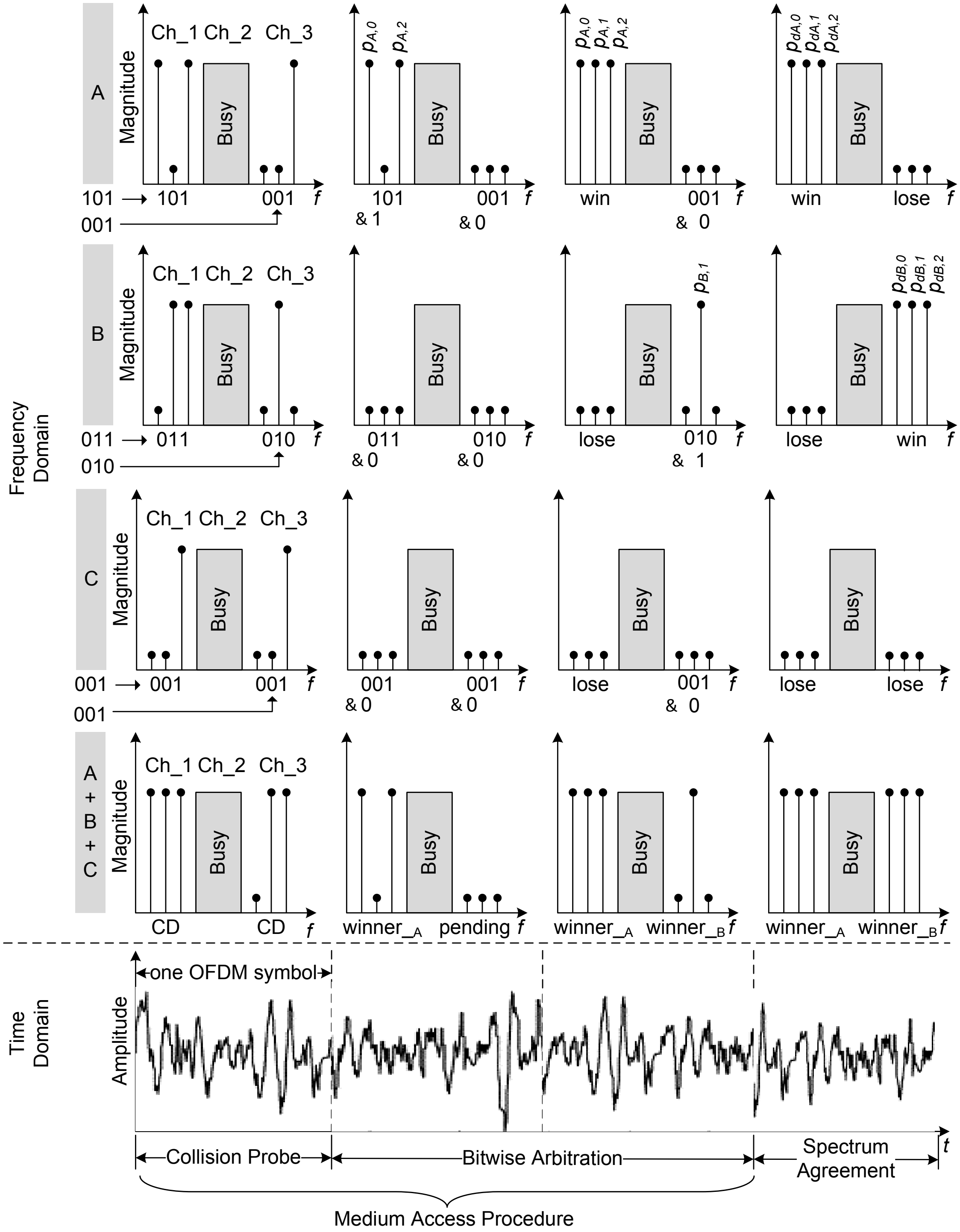


Figure 4.7: The initial three phases in flexible spectrum use channel bonding.

the collision will be resolved within k OFDM symbols where k is the number of subcarriers used for medium access contention in a narrow channel. Note that the bitwise arbitration may end earlier without traversing all bits. Therefore, the overhead is upper bounded by k but most of the time it is lower.

CHAPTER 5

SPECTRUM AGREEMENT

After the spectrum contention, a transmitter has won some narrow channels. However, the receiver is unable to decode any packet sent by the transmitter if the occupied spectrum is unknown. The receiver also needs to verify that the channels selected by the transmitter are indeed available at the receiver. To achieve fast spectrum agreement, we employ convolution and a common Fourier transform pair [8] [7].

5.1 Basic Idea

As introduced earlier, nodes employ N-point IFFT/FFT algorithms to efficiently modulate/demodulate N subcarriers in a wideband channel. Suppose the wideband channel is divided into n narrow channels for independent clear channel assessment (CCA). Each narrow channel contains $k = N/n$ subcarriers. Each node is assigned a unique sequence of length k, known to neighbors like the ID.

After the spectrum contention, the transmitter maps the receiver's unique sequence $\{p_0 p_1 p_2 ... p_{k-1}\}$ to subcarriers of each channel it has won. All subcarriers in other channels are fed with 0s as shown in Fig. 5.1. After performing the N-point IFFT, the output N complex numbers (referred to as one OFDM symbol) are transmitted.

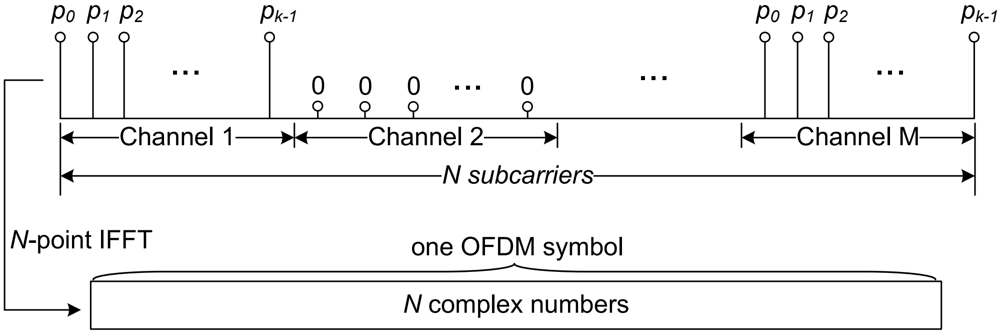


Figure 5.1: Spectrum agreement preamble construction.

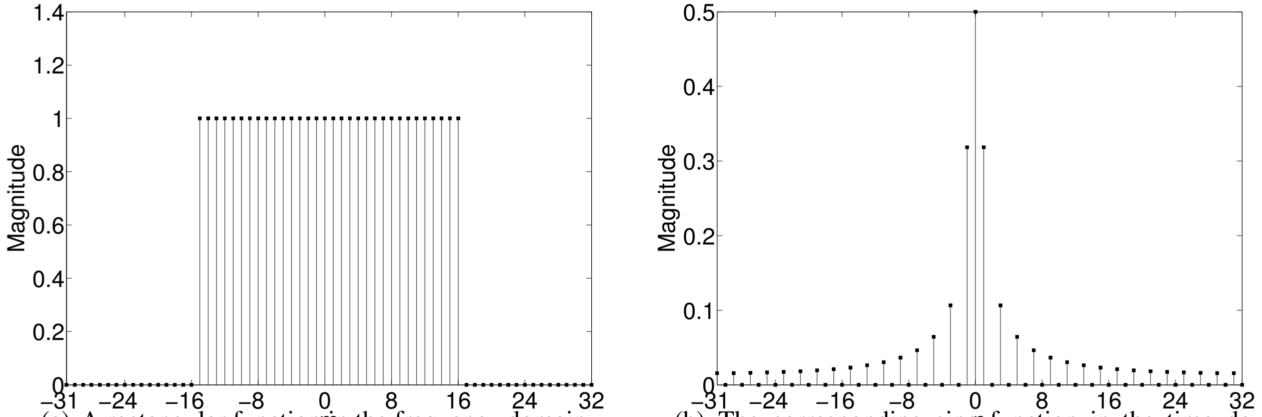
The receiver may extract the sequence in each channel and use the correlation technique to check whether or not a known sequence is presented in the received signal. However, to extract the sequence from each channel, the receiver needs to obtain a complete OFDM symbol and perform the FFT. Since the transmitter and the receiver have not been synchronized, the receiver does not know the start of an OFDM symbol. The receiver needs to perform the N-point FFT each time a new sample is obtained and conduct the correlation with the decoded sequence.

To reduce the overhead, we employ a common Fourier transform pair: the rectangular function and the sinc function. A rectangular function in the frequency domain matches a sinc function in the time domain. The Duality property of the Fourier transform provides that the reverse is also true. Given a general rectangular function $X[m]$ in which N samples contain K unity-valued samples from index $-n_o$ to $-n_o+(K-1)$ in the frequency domain, we can get the corresponding time domain expression through the inverse discrete Fourier transform (IDFT) :

$$\begin{aligned} x[n] &= \frac{1}{N}\cdot\sum_{m=-\frac{N}{2}+1}^{\frac{N}{2}} X[m]\cdot e^{j\frac{2\pi mn}{N}} \\ &= \frac{1}{N}\cdot\sum_{m=-n_o}^{-n_o+(K-1)} 1\cdot e^{j\frac{2\pi mn}{N}} \\ &= \frac{1}{N}\cdot\frac{sin\frac{\pi nK}{N}}{sin\frac{\pi n}{N}}\cdot e^{j\frac{2\pi n}{N}(\frac{K-1}{2}-n_o)}. \end{aligned} \tag{5.1}$$

The function has a main lobe that is centered at $n=0$ as shown in Fig. 5.2(b). The ratio of sine terms is the amplitude of x[n] and the exponential term is the phase angle of x[n].

To get a spike that is similar to the sinc function in the time domain, we need to create a rectangular function in the frequency domain. In the time domain, it is easy to implement convolution of two signals. Because the convolution in the time domain corresponds to the multiplication in the frequency domain [16] [11], we need sequences that hold $p_i \times p_i^* = c$ where c is a constant and p_i^* is the complex conjugate of p_i. To achieve this, we use constant amplitude zero auto-correlation waveform (CAZAC) sequences such as the Zadoff-Chu (ZC) sequences [5]. A ZC sequence $p_\gamma[i]$



(a) A rectangular function in the frequency domain.

(b) The corresponding sinc function in the time domain.

Figure 5.2: The waveform of rectangular function and sinc function.

is defined as [5]

$$p_\gamma[i] = e^{\left(-j\frac{\gamma\pi i(i+(k\,mod\,2))}{k}\right)} \tag{5.2}$$

for $i = 0, 1, ..., k-1$, where k is the length of the sequence, and γ is a positive integer (called the root index) relatively prime to k.

Instead of trying to obtain a complete OFDM symbol and decode the sequence, the receiver assumes that the complex conjugate of its unique sequence is present in each channel and calculate the expected IFFT results. The receiver then simply convolves the expected OFDM symbols with the received signal. Because the convolution corresponds to the multiplication in the frequency domain, we have

$$\begin{aligned} p_\gamma^*[i] \times p_\gamma[i] &= e^{\left(j\frac{\gamma\pi i(i+(k\,mod\,2))}{k}\right)} \times e^{\left(-j\frac{\gamma\pi i(i+(k\,mod\,2))}{k}\right)} \\ &= e^{j0}. \end{aligned} \tag{5.3}$$

We obtain a rectangular function in the frequency domain as shown in Fig. 5.3. Consequently, the receiver can observe a spike similar to Fig. 5.2(b) if the sequence is indeed transmitted in the channel. Through the convolution, the receiver is able to check whether or not its unique sequence is present in signals in each channel.

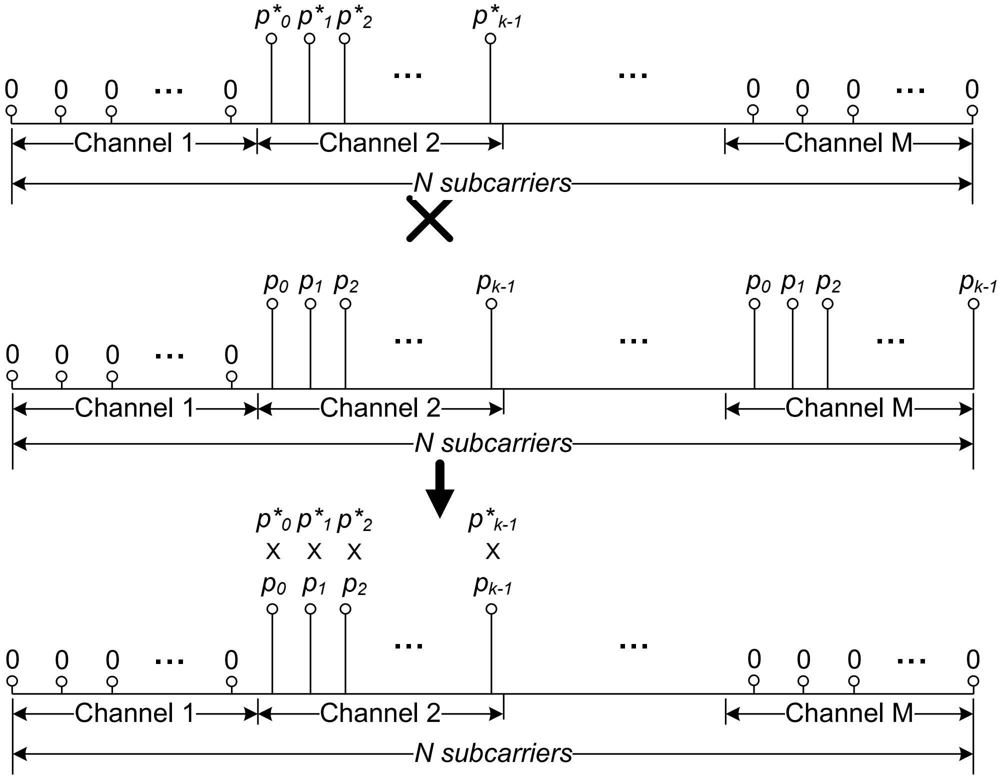


Figure 5.3: Identify whether or not a intention sequence is present in a narrow channel.

5.2 Hidden Terminal

In wireless networks, the hidden terminal problem occurs when a node is transmitting to a receiver, but the transmission is undetectable at other nodes that are trying to communicate with the same receiver. This leads to difficulties in medium access control. Fig. 5.4 gives an example of hidden terminal problem. In the example, node A and node C can communicate with node B, but they are hidden from each other. When nodes A and C start to send packets simultaneously to node B, they cannot detect the interference since node A and node C are out of each other's interference range.

To address the hidden terminal problem, the transmitter applies a random cyclic shift to the sequence of the receiver before mapping the sequence to subcarriers. The cycle shift will not remove the detectable spike. Suppose the sequence is cyclically shifted by δ. When k is an even

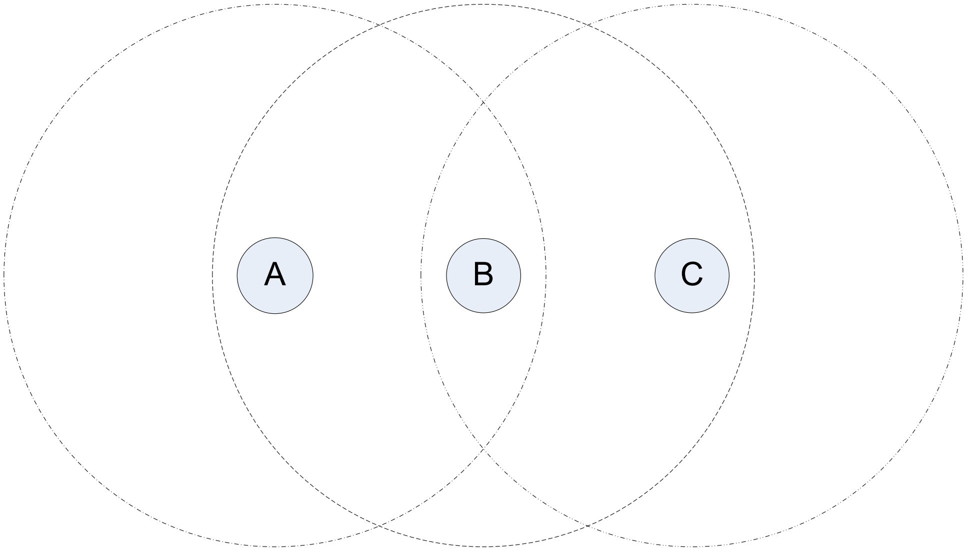


Figure 5.4: An example of hidden terminal problem.

number, we have

$$\begin{aligned} p_\gamma^*[i] \times p_\gamma[i-\delta] &= e^{(j\frac{\gamma\pi i^2}{k})} \cdot e^{(-j\frac{\gamma\pi(i-\delta)^2}{k})} \\ &= e^{j\frac{\gamma\pi(i^2-(i-\delta)^2)}{k}} \\ &= e^{j0} \cdot e^{j\frac{\gamma\pi(2i\delta-\delta^2)}{k}}. \end{aligned} \tag{5.4}$$

Equation 5.4 shows that the rectangular function is multiplied by a linear phase shift $e^{j\frac{\gamma\pi(2i\delta-\delta^2)}{k}}$ (note that i is the variable and δ is a constant). When k is odd, it can be proven that the phase shift is also linear. The shifting theorem of DFT states that a constant phase shift of $X[m]$ corresponds to a circular shift of the output sequence $x[n]$ [11]. Therefore, the receiver still can observe a spike during convolution. The spike of the sinc function is just shifted in the time domain.

Because two transmitters may apply different cycle shifts to the receiver's unique sequence, the receiver will observe more than one spike in the same channel. The receiver learns that there is a collision between hidden terminals. Even if two transmitters choose the same cycle shift, it is unlikely that they have won the same set of channels because the contention in each channel is independent. Therefore, the receiver will observe higher spikes in collided channels because two spikes are superposed. The receiver still learns that there is a collision.

The receiver picks a transmitter and applies the same cycle shift to its own unique sequence. It maps the shifted sequence to subcarriers of channels used by the picked transmitter. On the other hand, the sender convolves the receiver's unique sequence with the received signal once it finishes the transmission of the receiver's unique sequence. Due to the temporal relationship, the sender can learn the channels that are confirmed by the receiver through the convolution.

The transmitter validates that only channels occupied by itself are confirmed by the receiver; otherwise, this may be a confirmation to another transmitter. It also validates that the spike is shifted according to its selected random cycle shift on the sequence. If any validation fails, the transmitter assumes that there is a collision and waits for the next round of contention.

5.3 Sequence Assignment

The unique sequence of a node should not create a sinc function in the time domain when convolved with another node's unique sequence. We can adjust the root index of ZC sequences, γ, to meet the requirement. If two nodes use ZC sequences that have different root indexes, we have

$$\begin{aligned} p^*_\gamma[i] \cdot p_{\gamma'}[i] &= e^{(j\frac{\gamma\pi i^2}{k})} \cdot e^{(-j\frac{\gamma'\pi i^2}{k})} \\ &= e^{j\frac{\pi(\gamma-\gamma')i^2}{k}} \end{aligned} \tag{5.5}$$

when k is even. It can be proven that the phase shift is also not linear when k is odd. Because the phase shift is not linear, the multiplication does not correspond to a shift of the sinc function in the time domain. The receiver will not observe a spike during convolution. Therefore, each node can obtain a unique sequence based on its unique ID.

CHAPTER 6

PERFORMANCE EVALUATION

We evaluate flexible spectrum use channel bonding (FSUB) through both GNU Radio experiments and ns-2 simulations. The first step is to evaluate the collision detection and bitwise arbitration. We compare it with CSMA/CA to demonstrate that the bitwise arbitration is more efficient in medium access contention. We then evaluate the spectrum agreement design, verifying that the receiver can be informed of used channels and can confirm the channels. Finally we evaluate the performance of FSUB using simulations over various network configurations.

6.1 Medium Access Contention

There is a delay up to hundreds of microseconds between the GNU Radio and the USRP platform as measured in some studies [12]. We can hardly tell the difference between bitwise arbitration and random backoff if the system delay dominates the total delay. Therefore, we implement the collision detection and bitwise arbitration in Verilog / VHDL on the FPGA of USRP E110. Because packet encoding and decoding are more complex, we implement them in GNU Radio. A packet is buffered on FPGA before transmission. Due to the limited FPGA resources, each packet is set to 200 bytes and the modulation scheme is QPSK.

To obtain full duplex wireless communication, we use the WBX transceiver, which has independent local oscillators (LOs) for transmitting and receiving chains. To reduce self-interference, we separate the transmitting and receiving antennas by about 2 feet. The performance of collision detection will be improved with better self-interference cancelation techniques [6] [10]. These advanced self-interference cancelation techniques allow a device to implement full duplex even with only one antenna. Note that although a device evaluates and contends each narrow channel independently, the device transmits and receives in the same wide band of spectrum. It does not need to tune the center frequency to different frequencies for transmit and receive, hence one antenna is sufficient as long as we can prevent ADC saturation. The device obtains signals from one wideband channel but processes them as they are from multiple narrow channels.

6.1.1 Reliability of Collision Detection

When a node is transmitting the compound preamble, it should not falsely detect noises as collisions; otherwise it may defer its transmission unnecessarily. We set one USRP E110 to the full duplex mode. In Fig. 6.1, the left side shows the case when there is no transmission and the right side is the case when there exists a transmission. From the figure, we can see that when the node is transmitting, the noise floor is increased. If a node measures the noise floor during the interval when there is no transmission and uses the noise floor to determine whether there is a signal at a subcarrier, it may incorrectly regard that there is a collision. To reduce false alarm rate, the noise threshold is increased slightly.

A node first measures the noise floor when the channel is idle. Fig. 6.1 shows that there is a DC offset at the middle of the spectrum in USPR. If the DC offset has the highest magnitude, we regard that the channel is idle. When only one node is transmitting, it should not identify inactive subcarriers as active. We set the noise threshold slightly higher than the measured noise floor. A subcarrier is considered active if its magnitude is above the threshold. To get the detection accuracy, we compare the indices of detected active subcarriers with the indices of actually used subcarriers. As shown in Fig. 6.2 , if the noise threshold is increased by 7 dB from the measured noise floor, it is unlikely that a node will mistake an inactive subcarrier as an active subcarrier.

Once the noise threshold is determined, we study whether a node can detect the collision from a weak signal. We start another USRP E110 and gradually increase the transmission power. Fig. 6.3 shows that a node is unlikely to miss the detection of another node's compound preamble if the SNR is above 9 dB.

The design of the compound preamble makes it easy to detect. Normally, when two transmissions collide, the weak signal is buried under the strong signal. However, when two compound preamble transmissions collide, they do not affect each other because they use different subcar-

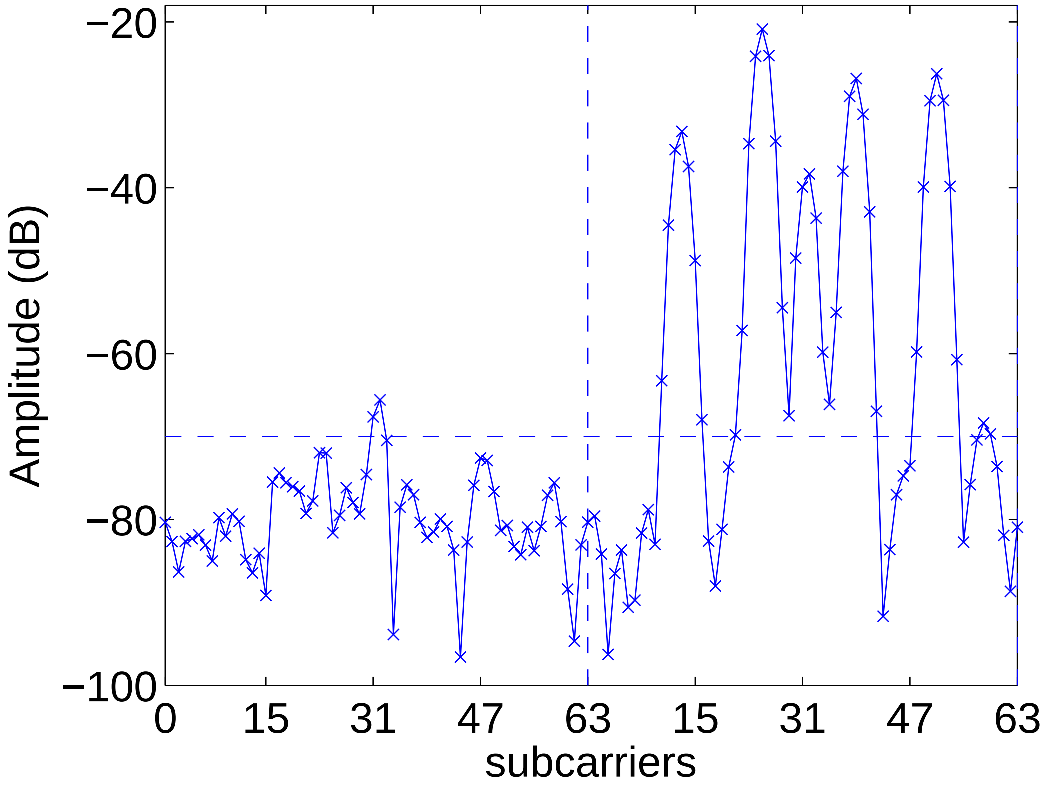


Figure 6.1: Results of performing 64 point FFT on received signal.

riers. As long as the SNR is above 9 dB, a collision is detectable. If two nodes use the same subcarriers, the aggregated power increases the collision detection probability at a third node. A collision is undetectable only if two nodes choose the same binary code, but the probability is reduced exponentially with the length of the binary code. If the self-interference cancelation is good enough, the collision is detectable even when two nodes choose the same binary code.

6.1.2 Throughput Gain

The bitwise arbitration reduces the medium access overhead, leading to higher throughput. We make three USRP E110 keep sending packets to a E110. Each E110 has two ADCs of a sampling rate at 64 Mega samples per second (Msps). Because the decimation rate can be set to multiples of 2, we evaluate the throughput at data rates of 0.5, 1, 2, 4, and 8 Msps. We change the data rate to adjust the transmission time because the packet size is fixed by the limited buffer size. To compare

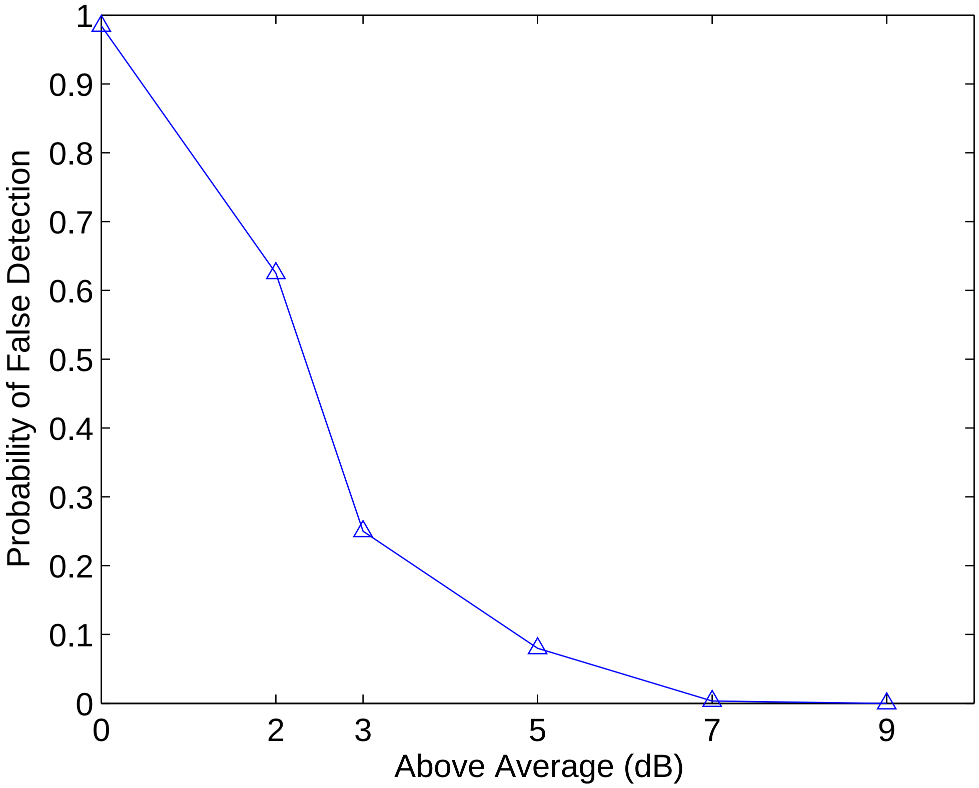


Figure 6.2: Setting noise threshold above the noise floor measured when the channel is idle.

with CSMA/CA, we use the parameters defined in 802.11n [1]. Fig. 6.4 shows the improvement of FSUB over 802.11n.

We implemented a 64 point IFFT/FFT algorithm on FPGA of USRP E110. To restrict the length of the bitwise arbitration phase, 8 subcarriers are selected for medium access contention. With 8 subcarriers, the collision probability between any pair of nodes in bitwise arbitration is $\binom{n}{2}(\frac{1}{2^8-1})^2 = \binom{n}{2}(\frac{1}{255})^2$. On the contrary, in CSMA/CA the collision probability may be high with the small initial CW [0, 15]. To obtain the same low collision probability, 802.11 needs a fixed CW size of 255. Fig. 6.4 shows that the throughput of 802.11 is improved a little bit with the larger contention window, but the throughput is still low because the average backoff time is increased with a larger contention window.

The throughput improvement of FSUB over 802.11 is more significant at high speeds as shown in Fig. 6.4. At high speeds, the medium access contention is a large portion of a transmission

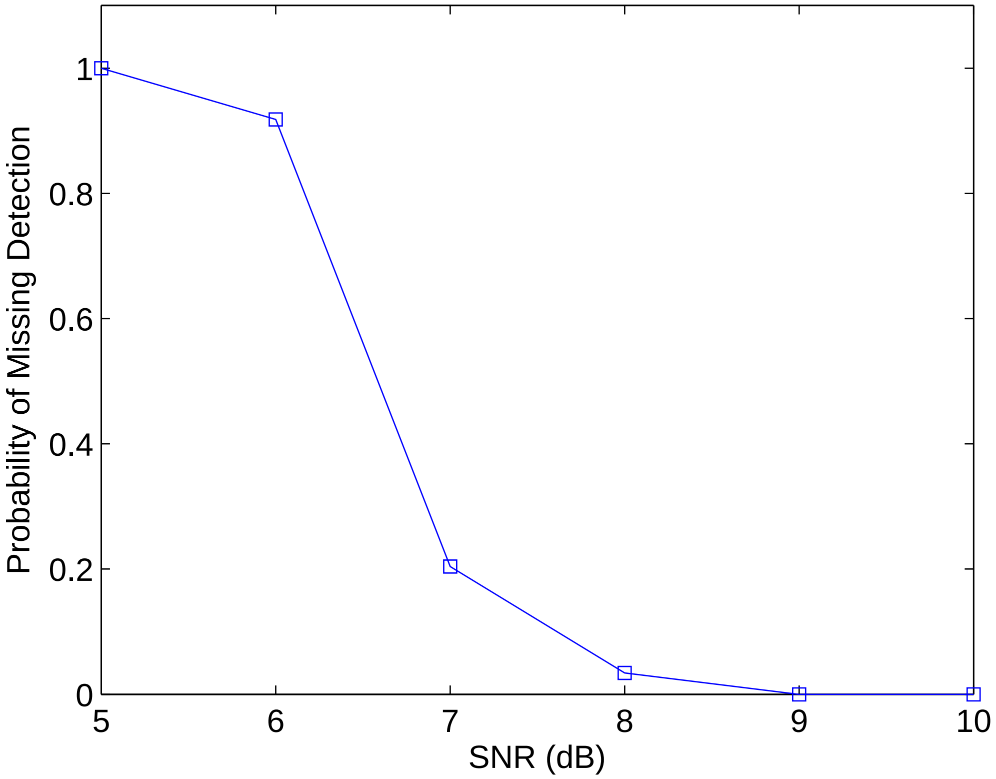


Figure 6.3: The misdetection rate is reduced with higher SINR.

where the actual data transmission is completed in a short time. The overhead of medium access contention in 802.11 is invariable to the physical layer (PHY) data rate. The throughput is thus constrained by the medium access overhead. On the contrary, in the bitwise arbitration, the time used to collect samples is reduced with higher data rates. The collision detection and bitwise arbitration is upper bounded by $(1+k)$ FFT frames when k bit binary codes are used for medium access contention. Suppose each FFT frame consumes a vector of 64 samples. When the data rate is 4 Msps, each FFT frame takes 16 μs. When the data rate is increased to 8 Msps, each FFT frame takes only 8 μs. The channel access overhead is thus reduced with higher data rates, allowing high PHY data rates to be translated to high throughput.

In summary, the FSUB system is efficient in medium access contention. It provides low collision probability due to the binary mapping design. In addition, the medium access overhead is reduced by the proposed collision detection and bitwise arbitration mechanism. Because of the

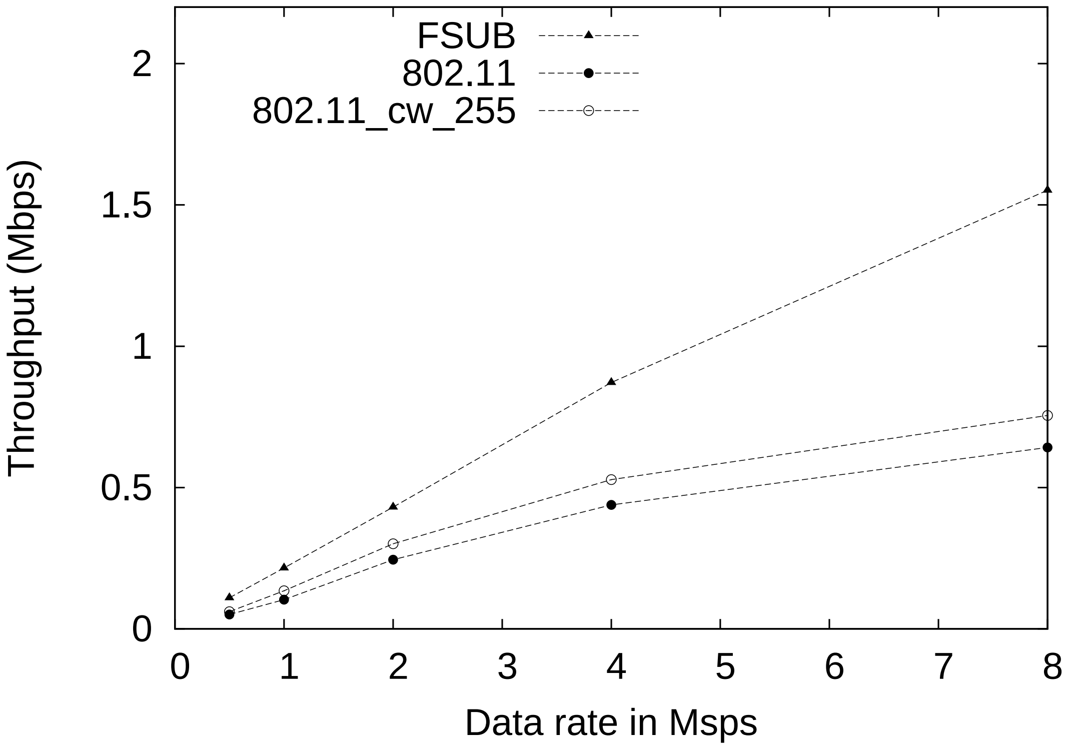


Figure 6.4: Throughput gain over 802.11.

low collision probability and the low medium access overhead, the throughput of FSUB is significantly improved over 802.11. With the increase of data rate, the transmission time of data is shorter and there are more rounds of contentions. The advantages of having low collision probability and low medium access overhead become more obvious. A small amount of reduction on medium access overhead may lead to a significant amount of improvement on efficiency because the medium access overhead dominates the total airtime in high speed transmission. Therefore, with the increasing data rate, the improvement of FSUB over 802.11 continues to grow.

6.2 Spectrum Agreement

Once a node wins some narrow channels, it needs to inform the receiver of occupied channels. To validate the effectiveness of spectrum agreement, we use 128 point IFFT/FFT on a band of 4 MHz. The band is divided into four channels of 1 MHz and each contains 32 subcarriers.

One USRP is used as the receiver, and one USRP keeps transmitting data in all channels to simulate noises. One transmitter applies a random shift on the receiver's unique sequence and transmits it in two of the four channels (channel 1 and channel 3). To create a collision, another transmitter applies a random shift on the same receiver's unique sequence and transmits it in one of the two occupied channels.

The receiver assumes that the complex conjugate of its unique sequence is mapped to subcarriers of each channel and calculates the expected IFFT results. If a spike is detected in the convolution of an expected OFDM symbol and the received signal, the receiver learns that a neighboring node has data to it in the channel.

Fig. 6.5 shows the convolution result for the channel in which the receiver's unique sequence is indeed transmitted. The x-axis is the temporal progression of collected samples and the y-axis is the convolution value normalized to the expected convolution value of the unique sequence. The figure shows that a node can easily detect its unique sequence in a channel through the convolution spikes. The convolution spikes are clear even at a signal-to-interference-plus-noise ratio (SINR) of 0 dB. Note that Wi-Fi requires a minimum SINR of $4 \sim 5$ dB to decode the lowest rate packet Therefore, the spectrum agreement method works well within the communication range.

Two nodes may transmit to the same receiver at the same time. If the receiver can detect the collision, it can choose a transmitter to avoid the collision in data transmission. One OFDM symbol consists of 128 complex numbers when a 128-point IFFT algorithm is used. All possible values are represented in a segment of 128 convolution values. When two nodes collide in a channel, Fig. 6.6 shows that two convolution spikes are observed in an arbitrary segment of 128 convolution values if they applied different shifts on the sequence. The sinc functions created by the convolution are shifted by different amount.

If the sequence is not transmitted in a channel, the receiver should not detect one. Fig. 6.7 shows that if the unique sequence is not transmitted in a channel, there is no spike during convolution. Although the absolute magnitudes of convolution values may increase with the interference and noise, there is no abrupt change of convolution values. To detect a real spike in convolution, we

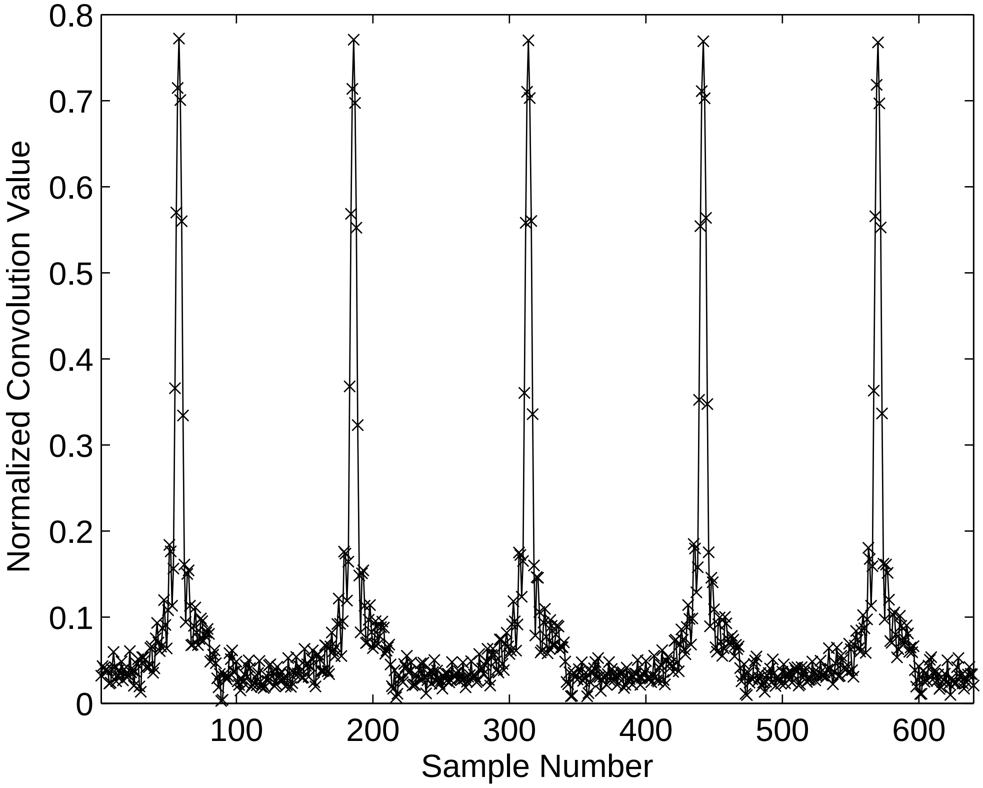


Figure 6.5: One transmitter is transmitting the sequence in the channel (SINR: 0 dB).

calculate the moving average while performing the convolution. A threshold that is *x* dB above the moving average is used to determine whether a real spike occurs. Fig. 6.8 shows that 5 dB can exclude most false detections of convolution spikes.

With the threshold discovered above, we check the required SINR for detecting the convolution peaks. We reduce the transmitter's transmission power to obtain the desired SINR. Each experiment checks the number of missed detections in 1000 transmissions. An accurate detection should indicate that both channel 1 and channel 3 are used. Fig. 6.9 shows that the detection of used channels is reliable even at low SINR. Note that the transmitter has won these channels in bitwise arbitration phase, the interference from other nodes is expected to be low.

In summary, in our FSUB system, the receiver does not need to obtain a complete OFDM symbol and then decode and extract the transmitted unique sequence to accomplish the spectrum agreement between the transmitter and the receiver. By using the convolution technique and a

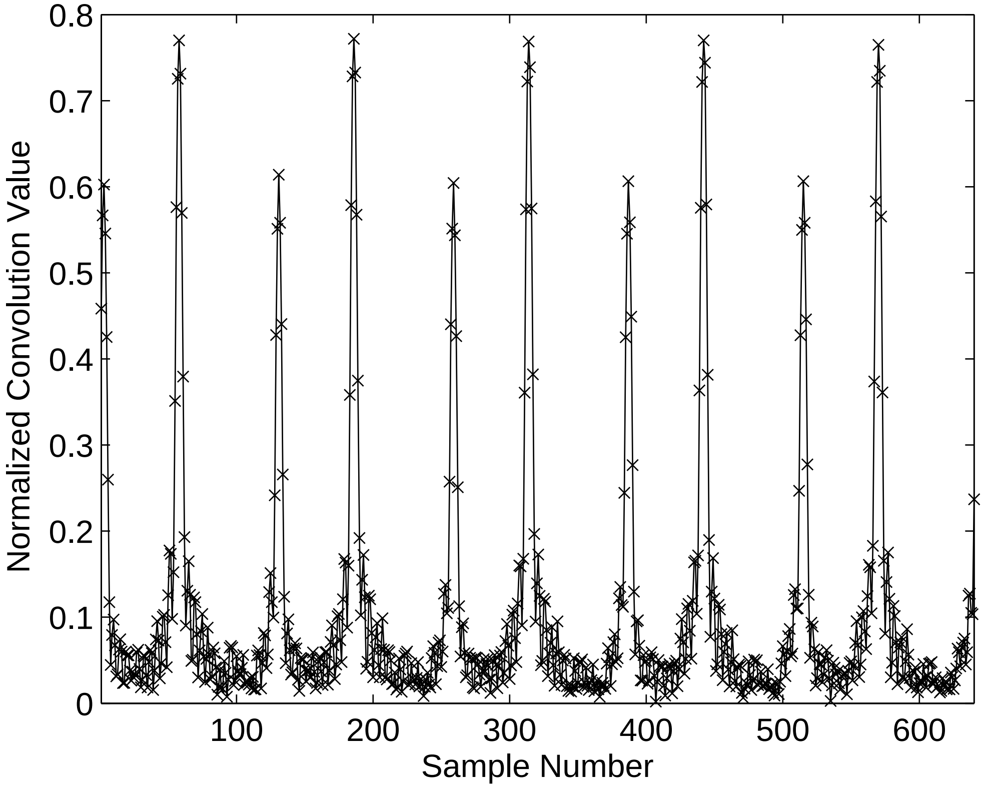


Figure 6.6: Two transmitters are transmitting different shifted sequences in the channel.

common Fourier transform pair, the receiver and transmitter can achieve fast and reliable spectrum agreement through simply observing spikes in target channel without decoding signals. This avoids heavy signal processing work and thus reduces the power consumption and latency.

6.3 Simulation Results

We evaluate the performance of FSUB using simulations in ns-2.35. We simulate a scenario where several pairs of nodes contend for a band of 40 MHz and the band is divided into 8 channels of 5 MHz. With one antenna, the maximum data rate is 135 Mbps using 64-QAM modulation and 5/6 coding rate as indicated in 802.11 standard [1]. To study the difference between bitwise arbitration and random backoff, we do not add narrow channel interferers in the beginning. We increase the number of wideband transmitters from 1 to 30. All transmitters generate fully backlogged CBR

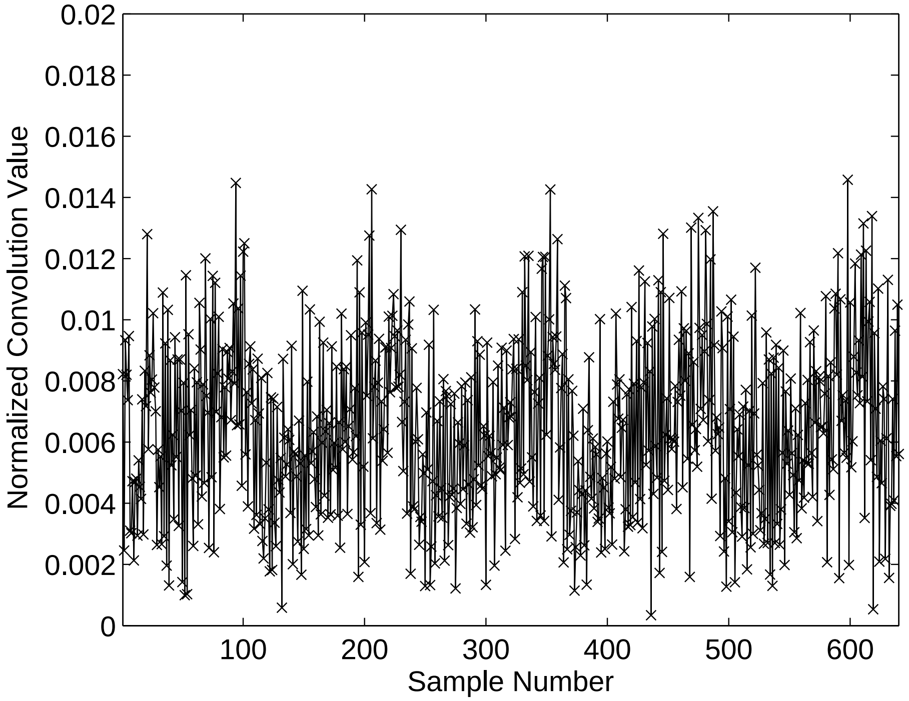


Figure 6.7: No transmitter is transmitting the sequence in the channel.

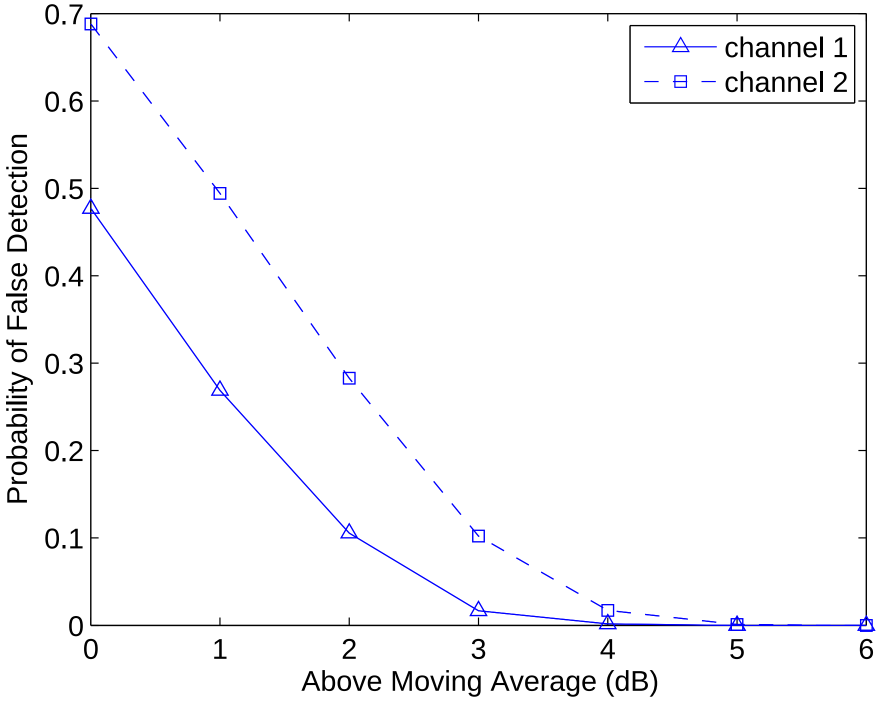


Figure 6.8: Setting threshold for convolution spike indication.

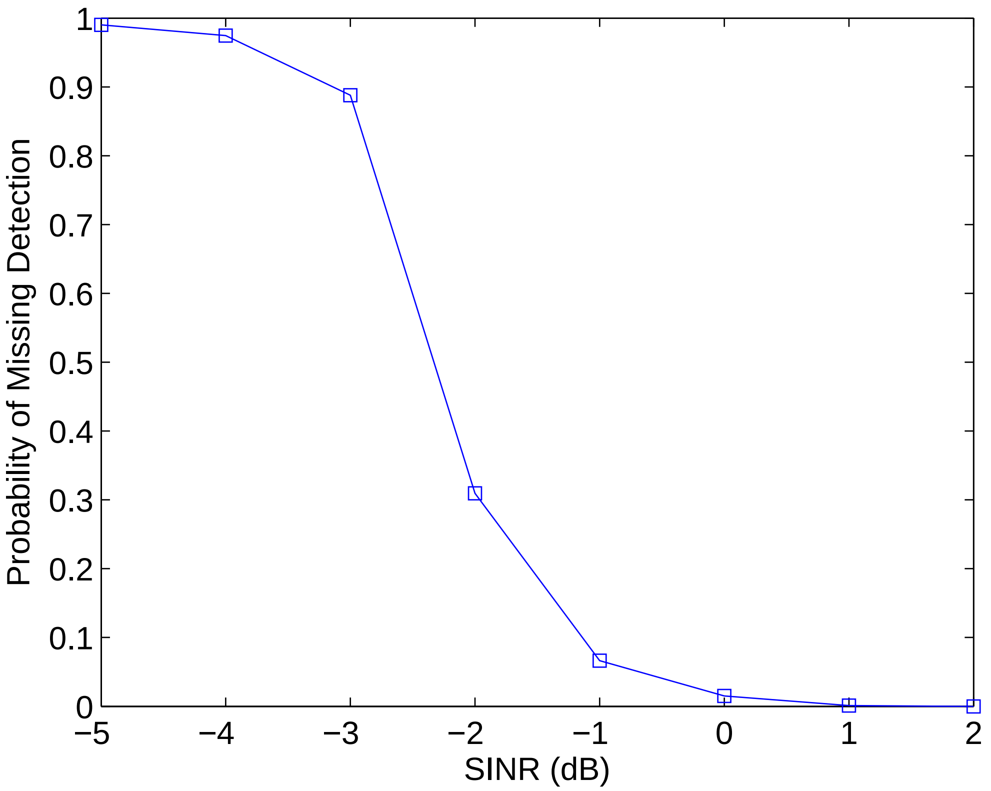


Figure 6.9: Probability of missing detections of occupied channels.

traffic with packet size of 1500 bytes. Fig. 6.10 shows the throughput gain of FSUB over 802.11.

If there is only one pair of nodes, the transmitter can transmit immediately after collision probe in FSUB. Performing 128 point FFT needs 128 samples per frame. It takes 3.2 μ s to collect 128 samples at 40 MHz. The channel access overhead is a duration of 3.2 μs. In 802.11, the average channel access backoff is $7.5 \times 9\mu s = 67.5\ \mu s$ with the contention window [0, 15]. The difference in channel access overhead leads to the difference in throughput.

When more nodes are contending for medium access opportunities, collisions and enlarged contention window contribute to the decrease of throughput in 802.11. A node can infer a collision only after it completes the transmission. To enhance 802.11, we implemented the collision notification (CN) mechanism introduced in [15]. We assume an ideal case where a collision can be detected and notified whenever it happens. Fig. 6.10 shows that CN improves the throughput of 802.11 quite a lot but has a marginal improvement over FSUB. The result shows that the number

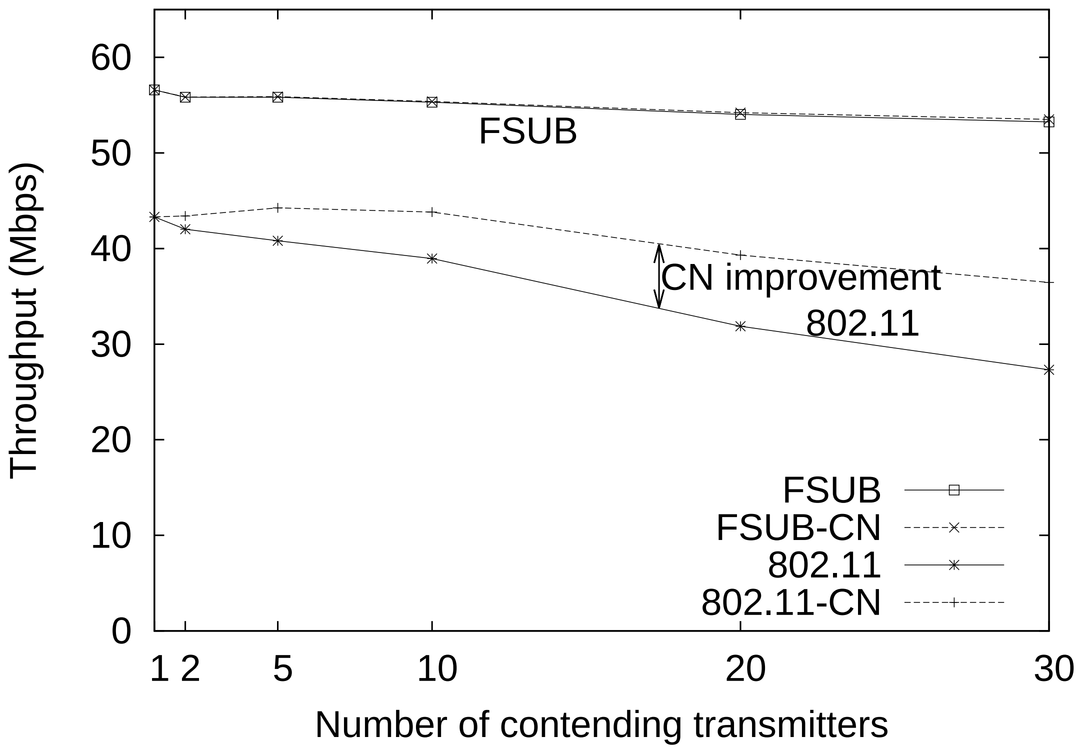


Figure 6.10: Throughput comparison with different number of contending transmitters.

of collisions in FSUB is few due to the binary mapping in frequency domain and efficient bitwise arbitration.

When all nodes have the same channel width, they have equal chance to win medium access opportunities. When some nodes change to use narrow channels, they get more medium access opportunities. To study the impact of narrow channel interference, we add narrow channel devices to the 8 channels one by one. Because narrow channel devices work independently in the 8 channels, they are likely to win medium access opportunities as they do not contend with devices in other channels. On the contrary, a channel bonding device in 802.11 has to contend with all narrow channel devices. If any channel is occupied, the channel bonding device cannot transmit. As shown in Fig. 6.11 , when all eight narrow channels are occupied by narrow channel devices, the channel bonding device nearly has no chance to transmit.

In frame-based channel bonding, a wideband device keeps counting down the random number as long as one channel is available. When the random backoff is completed, the wideband device

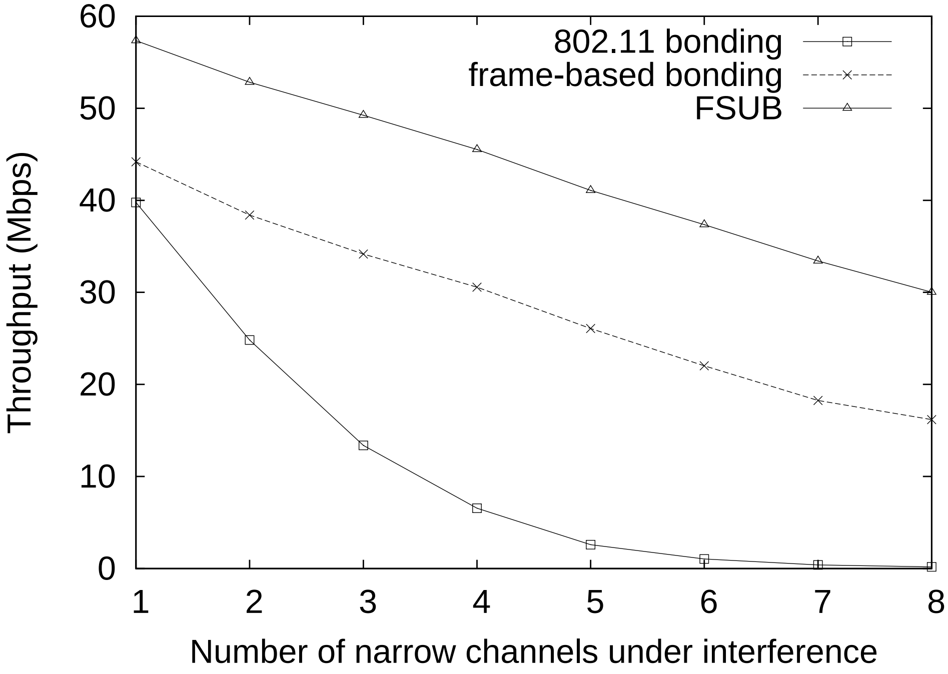


Figure 6.11: Throughput with different number of narrow channels under interference.

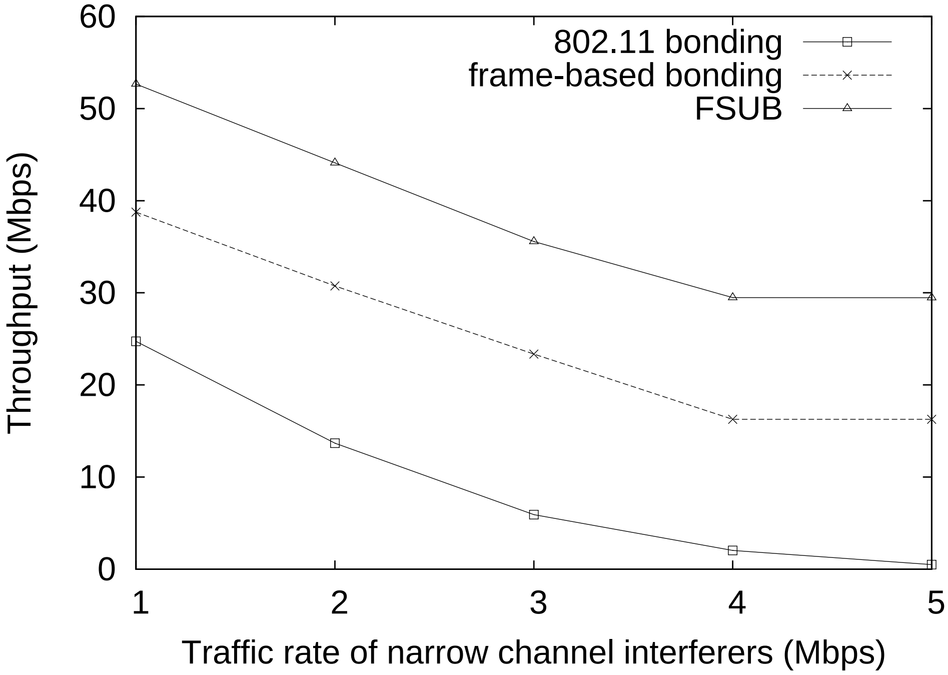


Figure 6.12: The throughput of wideband device is affected by narrowband devices' activities.

transmits in all channels that are still idle. Fig. 6.12 shows that the method allows a wideband device to survive in a network where many uncoordinated narrow channel devices exist. The spectrum resource, however, is not fully exploited. FSUB lets a wideband device monitor the state of the entire band even when the device is transmitting. Whenever a channel becomes available, the wideband device will participate in the contention and it may win the channel for further bonding. In joint with the efficient parallel bitwise arbitration, the throughput of channel bonding is the highest under the severe narrow channel interferences.

When there exist narrow channel devices in all eight narrow channels, the 802.11 channel bonding can still provide service if the narrow channel devices are not very active as shown in Fig. 6.12 . However, if narrow channel devices always have data to transmit, the wideband device can hardly find a gap when all narrow channels are idle. FSUB allows a wideband device to initiate a transmission as long as some channels are available. The chance to deliver a packet is higher. FSUB also bonds other channels whenever they become available. The throughput is thus higher than frame-based channel bonding where new channels can be used only in the next frame.

In summary, the FSUB system is more tolerant to narrowband interference. It allows wideband devices to coexist with narrowband devices and fully exploit the spectrum resources. Especially under intense narrowband interferences, the throughput of FSUB is improved to 1.6 times of current frame-based channel bonding and the 802.11 standard channel bonding devices can hardly find chances to transmit. The FSUB system also maintains fairness for all nodes because any node can contend for a channel. Wideband devices do not need to wait for other channels to be idle and nodes that are in transmission can also participate in contention. Therefore, even in severe contention, wideband devices that are in or not in transmission still have equal chances to acquire medium access opportunities in contention with narrowband devices.

CHAPTER 7

CONCLUSION AND FUTURE WORK

7.1 Conclusion

In this thesis we introduced a flexible spectrum use channel bonding (FSUB) protocol to address the inefficiency and unfairness issues caused by coexistence of heterogeneous radios.

The FSUB employs a compound preamble design to detect channel collisions in the frequency domain. In the compound preamble design, a binary mapping method greatly reduces the collision probability and allows a node to contend each channel with a different priority. Since the collisions are detectable, the FSUB introduces a parallel bitwise arbitration mechanism to address the severe contention in wideband spectrum sharing. It significantly reduces the medium access overhead by quickly determining the winner in a contention. To inform the receiver of channels used by the transmitter for data transmission, the FSUB applies a convolution solution to achieve fast spectrum agreement between the transmitter and the receiver.

Experiments on testbed validate the effectiveness of the spectrum contention and the spectrum agreement. The throughput of FSUB can be 2 times of that of 802.11 at high data rates due to the efficient spectrum contention achieved by binary mapping and bitwise arbitration with collision detection. Extensive simulations show that FSUB allows a wideband device to obtain medium access opportunities even under intense narrowband interferences and the throughput is improved by at least 20% over current frame-based channel bonding. It allows wideband devices to better coexist with narrowband devices and indeed benefit from using wide channels through channel bonding.

We conclude that our FSUB system is an effective method to enable efficient channel bonding in contention-based wireless communication. It provides low collision probability with low medium access overhead, leading to high throughput. By using the convolution technique and a common Fourier transform pair, the receiver and the transmitter can achieve fast and reliable spectrum agreement through simply observing spikes in target channel without decoding signals. The FSUB system allows a device to initiate a transmission immediately when some narrow channels are sensed to be idle. It also allows node to detect channel state and increase channel width during transmission by using self-interference cancelation techniques. Therefore, the spectrum resources can be fully exploited and the method ensures fairness for all nodes (narrowband and wideband, either in transmission or not) to contend new idle channels.

7.2 Future Work

In the GNU Radio experiments, we separate the transmitting and receiving antennas by about 2 feet to reduce self-interference. This is not the optimal solution. To gain better collision detection performance, our future work is to implement self-interference cancelation techniques and study the impact on channel bonding performance.

Due to hardware constraints, we are unable to use a large band of spectrum for experiments. The fading is similar for frequencies that are close. With more powerful devices, we can study the impact of frequency-selective fading on spectrum contention.

Setting noise threshold is another issue on which we need more considerations. In this thesis, a static noise threshold is investigated. However, in different communication conditions, the noise threshold may be different. Therefore, it is desirable to make the noise threshold adaptive to the communication environment.

In this thesis, the receiver picks one transmitter to respond. Since the receiver can isolate channels using filters, the receiver may be able to receive data from multiple transmitters at the same time. In our future work, we will implement multiple receiving chains and study the performance of parallel communication.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] IEEE Standard for Information technology–Telecommunications and information exchange between systems Local and metropolitan area networks–Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. *IEEE Std 802.11-2012 (Revision of IEEE Std 802.11-2007)*, pages 1 –2793, 2012.

[2] D. Bharadia, E. McMilin, and S. Katti. Full duplex radios. In *Proc. SIGCOMM*, pages 375–386, 2013.

[3] E. Chai, J. Lee, S.-J. Lee, R. Etkin, and K. G. Shin. Building efficient spectrum-agile devices for dummies. In *Proc. MobiCom*, pages 149–160, 2012.

[4] K. Chintalapudi, B. Radunovic, V. Balan, M. Buettener, S. Yerramalli, V. Navda, and R. Ramjee. WiFi-NC: WiFi over narrow channels. In *Proc. NSDI*, 2012.

[5] D. Chu. Polyphase codes with good periodic correlation properties (corresp.). *IEEE Transactions on Information Theory*, 18(4):531–532, 1972.

[6] S. S. Hong, J. Mehlman, and S. Katti. Picasso: flexible RF and spectrum slicing. In *Proc. SIGCOMM*, pages 37–48, 2012.

[7] P. Huang, X. Yang, and L. Xiao. Adaptive channel bonding in multicarrier wireless networks. In *MobiHoc*, pages 297–300, 2013.

[8] P. Huang, X. Yang, and L. Xiao. Dynamic channel bonding in multicarrier wireless networks. In *ICNP*, pages 1–10, 2013.

[9] P. Huang, X. Yang, and L. Xiao. Wifi-ba: Choosing arbitration over backoff in high speed multicarrier wireless networks. In *INFOCOM*, pages 1375–1383, 2013.

[10] M. Jain, J. I. Choi, T. Kim, D. Bharadia, S. Seth, K. Srinivasan, P. Levis, S. Katti, and P. Sinha. Practical, real-time, full duplex wireless. In *Proc. MobiCom*, pages 301–312, 2011.

[11] R. G. Lyons. *Understanding Digital Signal Processing, Third Edition*. Prentice Hall, 2010.

[12] G. Nychis, T. Hottelier, Z. Yang, S. Seshan, and P. Steenkiste. Enabling MAC protocol implementations on software-defined radios. In *Proc. NSDI*, pages 91–105, 2009.

[13] H. Rahul, N. Kushman, D. Katabi, C. Sodini, and F. Edalat. Learning to share: narrowband-friendly wideband networks. In *Proc. SIGCOMM*, pages 147–158, 2008.

[14] S. Sen, R. Roy Choudhury, and S. Nelakuditi. No time to countdown: migrating backoff to the frequency domain. In *Proc. MobiCom*, pages 241–252, 2011.

[15] S. Sen, R. Roy Choudhury, and S. Nelakuditi. CSMA/CN: Carrier sense multiple access with

collision notification. *IEEE/ACM Transactions on Networking*, 20(2):544–556, April 2012.

[16] S. W. Smith. *The scientist and engineer's guide to digital signal processing*. California Technical Publishing, San Diego, CA, USA, 1997.

[17] K. Tan, J. Fang, Y. Zhang, S. Chen, L. Shi, J. Zhang, and Y. Zhang. Fine-grained channel access in wireless LAN. In *Proc. SIGCOMM*, pages 147–158, 2010.

[18] L. Yang, W. Hou, L. Cao, B. Y. Zhao, and H. Zheng. Supporting demanding wireless applications with frequency-agile radios. In *Proc. NSDI*, pages 5–5, 2010.

[19] X. Zhang and K. Shin. Adaptive subcarrier nulling: Enabling partial spectrum sharing in wireless LANs. In *Proc. ICNP*, pages 311 –320, 2011.