6-19-2014

Scalable System Design for Covert MIMO Communications

Jason R. Pennington

SCALABLE SYSTEM DESIGN FOR COVERT MIMO COMMUNICATIONS

DISSERTATION

Jason R. Pennington,

AFIT-ENG-DS-14-J-5

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

The views expressed in this dissertation are those of the author and do not reflect the official policy or position of the United States Air Force, the Department of Defense, or the United States Government.

AFIT-ENG-DS-14-J-5

SCALABLE SYSTEM DESIGN FOR COVERT MIMO COMMUNICATIONS

DISSERTATION

Presented to the Faculty

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

in Partial Fulfillment of the Requirements for the

Degree of Doctor of Philosophy

Jason R. Pennington, BS, MS

June 2014

AFIT-ENG-DS-14-J-5

SCALABLE SYSTEM DESIGN FOR COVERT MIMO COMMUNICATIONS

Jason R. Pennington, BS, MS

Approved:

//signed//	22 April 2014
Richard K. Martin, PhD (Chairman)	Date
//signed//	10 April 2014
Maj Mark D. Silvius, PhD (Member)	Date
//signed//	22 April 2014
Matthew C. Fickus, PhD (Member)	Date

Accepted:

//signed//	2 May 2014
ADEDEJI B. BADIRU, PhD Dean, Graduate School of Engineering and Management	Date

AFIT-ENG-DS-14-J-5

Abstract

In modern communication systems, bandwidth is a limited commodity. Bandwidth efficient systems are needed to meet the demands of the ever-increasing amount of data that users share. Of particular interest is the U.S. Military, where high-resolution pictures and video are used and shared. In these environments, covert communications are necessary while still providing high data rates. The promise of multi-antenna systems providing higher data rates has been shown on a small scale, but limitations in hardware prevent large systems from being implemented.

Discussed here are the effects of the topology of communication nodes on Inter-Block Interference in Orthogonal Frequency Division Multiplexing (OFDM) systems. This effect can be leveraged such that eavesdroppers experience a lower Signal to Interference plus Noise Ratio (SINR) resulting in a poor quality communication link. Simulations show that an eavesdropper has a 10 dB worse SINR. The reverse is also considered where the point of view is taken as the eavesdropper. A study into improving the eavesdropping communication link performed. A pivotal calculation for the eavesdropper is found to be the estimation of the time of arrival of the received waveforms. The relative delays between users' waveforms is used to reduce the interference at the eavesdropper. The *van de Beek* and *Acharya* methods are considered. Simulations and experiments show that the *Acharya* method provides a more accurate measurement. Also discussed are hardware limitations such as on board slice logic and Digital Signal Processing (DSP) resources blocks. The utilization of these logic blocks proves to be a limiting factor in large scale multi-antenna systems. Particularly the inversion and equalization processes are the most expensive in terms of computation time and hardware resources. The trade-off between data rate and resource usage is provided with comments on interfacing multiple FPGAs to provide more available resources.

To my future wife for all her support and to my friends and family for all they have done.

Acknowledgments

I'd first like to express my deepest thanks to my advisor. His guidance was invaluable in the dissertation process and now I'm officially out of Chuck E. Cheese tokens. I'd also like to thank the CCR and the CCR staff for their help particularly in the area of hardware design. Finally, my committee members were always there to bounce ideas off of and I thank you for your support.

Jason R. Pennington

Table of Contents

Page

Page

List of Figures

List of Tables

List of Acronyms

Acronym	Definition
ADCs	Analog to Digital Converters
AWGN	Additive White Gaussian Noise
BER	Bit Error Rate
CE	Channel Estimate
CFO	Carrier Frequency Offset
CP	Cyclic Prefix
CPU	Central Processing Unit
CSI	Channel State Information
DFT	Discrete Fourier Transform
DSP	Digital Signal Processing
Ex	Eavesdropping Receiver
FDCE	Frequency Domain Channel Estimation
FDE	Frequency Domain Equalization
FFT	Fast Fourier Transform
FIFO	First-In First-Out Memory Block
FIR	Finite Impulse Response
FPGA	Field Programmable Gate Array
GPU	Graphical Processing Unit
IBI	Inter-Block Interference
ICI	Inter-Carrier Interference
IFFT	Inverse Fast Fourier Transform
ISI	Inter-Symbol Interference
LO	Local Oscillator

Acronym	Definition
LPD	Low Probability of Detection
LPF	Low Pass Filter
LS	Least Squares
LTE	Long-Term Evolution
LUTs	Look-Up Tables
LVDS	Low Voltage Differential Signaling
MAC	Multiply Accumulate
MIMO	Multiple-Input Multiple-Output
NLOS	Non Line-of-Sight
OFDM	Orthogonal Frequency Division Multiplexing
OFDMA	Orthogonal Frequency Division Multiple Access
PAPR	Peak-to-Average Power Ratio
PDF	Probability Density Function
PMI	Precoding Matrix Index
PPC	Performance Optimization With Enhanced RISC Performance Computing
PSD	Power Spectral Density
QAM	Quadrature-Amplitude Modulation
QRD	QR Decomposition
RAM	Random Access Memory
RF	Radio Frequency
RMSE	Root-Mean Squared Error
ROA	Region of Activity
Rx	intended receiver
SC-FDMA	Single Carrier - Frequency Division Multiple Access
SDR	Software Defined Radio

Acronym	Definition
SIMD	Single Instruction Multiple Data
SINR	Signal to Interference plus Noise Ratio
SISO	Single-Input Single-Output
SNR	Signal to Noise Ratio
SRRC	Square-Root Raised Cosine
STBC	Space-Time Block Codes
SVD	Singular Value Decomposition
TDCE	Time Domain Channel Estimation
TDOA	Time Difference of Arrival
TOA	Time of Arrival
TOC	Tactical Operation Centers
TOR	Time Of Reference
UAVs	Unmaned Aerial Vehicles
US	United States
VHDL	VHSIC Hardware Description Language
VLSI	Very Large Scale Integration
WARP	Wireless open-Access Research Platform
WLAN	Wireless Local Area Network

SCALABLE SYSTEM DESIGN FOR COVERT MIMO COMMUNICATIONS

I. Motivation

Within the last 20 years major advances in Multiple-Input Multiple-Output (MIMO) communication technology has occurred, from Telatar [1] and Foschini's [2] ground breaking work to Alomouti codes and Space-Time Block Codes (STBC) which provide a more reliable data link [3]. In more recent years work has been aimed at the effects of the physical limitations of antenna spacing experienced in cellular communication technology [4].

The attractive attribute of MIMO communication techniques is the improved bandwidth efficiency. Today bandwidth is a limited commodity, being more efficient with the limited resources is a must and MIMO communications provide that feature. As the amount of digital data around the world increases, the need to share that data with the rest of the world rises as well. Once again, to do this bandwidth is needed to provide the desired data rates.

The United States (US) Military has a particular interest in MIMO communications, where communication between Tactical Operation Centers (TOC), ground troops, air support, and heavy artillery is needed. Unmaned Aerial Vehicles (UAVs) are taking high resolution pictures and video from all around the world. The environments in which this data is taken are often hostile. In these situations covert communications are needed to provide the bandwidth needed to get the pictures, video, voice data, and intelligence to the right people.

In the literature, [31] looks at utilizing MIMO spatial gains to reduce transmission power, thus providing a Low Probability of Detection (LPD) waveform and [32] utilizes

Precoding Matrix Index (PMI) secret keys to provide secure communications. It has been found that typical encryption algorithms are susceptible to bit errors which in turn reduces throughput of the system [33]. Physical layer methods, in some instances, utilize mechanisms that do not experience this phenomena thus preserving throughput [34]. This is advantageous especially when these physical layer techniques are used in tandem with traditional encryption making the system more robust to security threats.

Often in these systems multiple users are sharing resources. A good example of this is cellular communications. In this example, the shared resource is bandwidth where users are allocated bandwidth dynamically as needed per user. An adverse result of this bandwidth sharing is the adverse affect in timing recovery of the received signal, which reduces the fidelity of the communication link. As the number of users increases, the higher the data rate requirements become and because of this added requirement dynamic allocation is used to reuse resources. However, when this occurs non-contiguous bands of spectrum are allocated to users which also reduces timing recovery ability.

In recent years Orthogonal Frequency Division Multiplexing (OFDM), and its variations, have become increasingly popular. In 4G communications, satellite radio and Wireless Local Area Network (WLAN) OFDM has been utilized for its multi-path resistance. OFDM relies on the Cyclic Prefix (CP) to transform linear convolution to circular convolution, however this adds structure that can be exploited by an Eavesdropping Receiver (Ex). The Ex can simply use correlation to determine where symbol boundaries occur and demodulate the payload. Some work has been done to reduce this structure by including random data in random OFDM blocks [35]. A transmitted signal with varying OFDM block lengths makes it harder to gain an accurate timing estimate.

Due to OFDM's vulnerability to phase errors, timing estimates are required to be accurate. For this reason, jamming techniques for OFDM often focus on disrupting the

receiver's ability to gain accurate symbol timing [36]. In this particular instance the Ex is also actively disrupting the signal for the intended receiver (Rx).

Another potential source of interference in OFDM systems is caused by the channel impulse response itself, if the CP is shorter than the impulse response [18]. This common problem in OFDM can be exacerbated by the topology used in the system [37]. If there is a difference in arrival times between multiple sources, the delay effectively adds to the length of the impulse response. If this delay is known, the CP length can accommodate the longer impulse response or synchronization techniques can be used to avoid delay related interferences. If the delay can not be characterized, or the channel itself is longer than the cyclic prefix, Inter-Block Interference (IBI) is introduced resulting in a lower Signal to Interference plus Noise Ratio (SINR).

A concept of Inter-Symbol Interference (ISI) to degrade a non-cooperative receiver's performance is used in [38]. The transmitted signal is preconditioned with columns of the Singular Value Decomposition (SVD) of the channel convolution matrix in such a way that Rx can demodulate the signal but the Ex experiences a coded signal distorted by the wireless channel.

Capacity gains over a Single-Input Single-Output (SISO) system have been shown for two transmitter two receiver systems, denoted 2×2 and also 4×4 systems [5–7]. Theoretically, larger systems offer higher gains, however the hardware technology limits fully functional large MIMO systems. Research into larger systems provide operational techniques needed for higher capacity gains in realizable systems.

The computational complexity of the MIMO receiver pushes the boundaries of modern processing platforms [28]. This is apparent in the literature where the approach taken by researchers to reduce computational complexity is done by focusing on the complex subprocesses of the receiver algorithm. For example, the architecture of a Single Instruction Multiple Data (SIMD) co-processor is designed for symbol recovery in a Field

Programmable Gate Array (FPGA) [42]. The design of this co-processor hopes to balance resource usage and latency. By reducing the latency of the co-processor the data rate is potentially increased.

Some literature assumes a realistic data rate for the receiver to service. Under the assumption of burst mode communication, the receiver has a maximum amount of time to process a data block [43]. To process the data block quickly, custom hardware is interfaced with a processor, such as the MicroBlaze, to reduce latency [44].

Reducing latency and FPGA resources are not the only constraints. An optimal or near optimal Bit Error Rate (BER) is desired. Investigation into an FPGA implementation of the *K-Best* and *Trellis-search* algorithms show near optimal equalization and bit mapping with cited resources used in the algorithm, [45] and [46] respectively.

A *MIMO Square Root Decoder* reduces the complexity of the pseudo-inverse calculation while maintaining a constant BER [47], while [32] focuses on computing the inversion by using some approximations. The exact inversion and approximate inversion are weighed in [48] and they show that the number of antennas at the base station is the gauge by which to determine the best algorithm for a Very Large Scale Integration (VLSI) implementation.

The rest of this document is organized as follows. In Chapter II, background information is covered starting with a standard notation set for the rest of the document. Next, multi-carrier waveforms are discussed, highlighting OFDM and Orthogonal Frequency Division Multiple Access (OFDMA). Matrix decomposition algorithms are introduced such as the LU and QR decompositions, which are tools used to handle the inversion of the channel matrix in hardware. The second chapter concludes with an overview the FPGA.

Next, Chapter III discusses the first of three MIMO specific problems. The first of these considers the delays between the transmitted waveforms of randomly distributed

transmitters in a circular region of activity. The topologically induced delay induces IBI, Inter-Carrier Interference (ICI), and ISI in multi-carrier systems. The performance of a receiver suffering from the effects of IBI, ICI, and ISI are considered.

Then, Chapter IV further builds on the topology and system described in Chapter III. The performance of the eavesdropping receivers would increase if timing estimates were calculated and used in the equalization process. The view is taken from the eavesdropper to improve performance where by estimating timing delays the degrading effect of the delays can be reduced. An implementation of the algorithms is discussed and real-world results are provided.

Chapter V then moves to the perspective of the intended receiver. Since data rate is the primary concern and considering the lessons learned in Chapter III a MIMO receiver is developed in VHSIC Hardware Description Language (VHDL). Resource utilization is reported and the tradeoffs between data rate and resource usage is weighed. Furthermore, trends for resource usage as a function of the number of antennas used is extrapolated to get some intuition for the size of implementation needed for a desired data rate.

Chapter VI concludes the dissertation with a summary of the three contributions and provides final thoughts on future work.

II. Background MIMO Theory

This chapter outlines background information about MIMO communications. First in Section 2.1, an overview the notation used in this dissertation is provided. In Section 2.2, multi-carrier waveforms are discussed. Section 2.3 provides the system model and topology of transmitters and receivers used throughout this dissertation. Taking the point of view of an eavesdropper, Section 2.4 discusses the ability to estimate arrival times in a multiple user scenario. This provides the ability to equalize the non-cooperative users' waveforms with less interference but also provides the basis for Time Difference of Arrival (TDOA) positioning efforts. Finally, Section 2.5 provides an overview of FPGAs in general and also the Wireless open-Access Research Platform (WARP) boards specifically used in this dissertation.

2.1 Notation

To analyze MIMO systems, some notation is needed to manage the waveforms being transmitted and received.

2.1.1 Vectors and Matrices.

Vectors and matrices are used quite often in analyzing MIMO systems. Column vectors are denoted by bold face lower case letters such as $\mathbf{x} \in \mathbb{C}^{N_t}$. The use of the transpose, $()^T$, or conjugate transpose, $()^H$, denotes row vectors. In MIMO specific vectors, the transmit vector $\mathbf{x}$ can be defined as a vector of symbols across the transmit antennas as a function of time or $\mathbf{x}$ can be defined as a vector of symbols that are transmitted on a single antenna. These two definitions are used on a case by case basis and $\mathbf{x}$ is defined explicitly to avoid any ambiguity.

Matrices are denoted by bold face upper case letters such as $\mathbf{H} \in \mathbb{C}^{N_r \times N_t}$. In MIMO systems $\mathbf{H}$ denotes the channel matrix. Estimates of any variable are denoted by the same letter of the variable with a hat, for example a estimate of the channel matrix is denoted, $\hat{\mathbf{H}}$.

2.1.2 MIMO Specific Variables.

Table 2.1 contains a list of all the MIMO specific variables used in this dissertation. A frequency domain vector is distinguished from a time domain vector by the *tilde* such as $\tilde{\mathbf{h}}$ is the frequency response where the impulse response is $\mathbf{h}$. The remaining variables in Table 2.1 represent general system parameters such as the number of antennas at the transmitter and receiver.

Table 2.1: MIMO specific variables

Variable	Description	Variable	Description
$\mathbf{H}$	channel matrix	$\mathbf{h}$	impulse response
$\tilde{\mathbf{h}}$	frequency response	$\mathbf{\Phi}$	covariance of transmit waveform
N_t	number of transmitters	N_r	number of receivers
N_s	number of samples	N_p	length of preamble $N_s \leq N_p$
B	bandwidth	M	modulation order
P_T	total transmit power	L	impulse response length
N_{pil}	number of pilot tones	N	number of sub-carriers
N_{dt}	number of data tones	$\mathbf{k}$	vector of sub-carrier indices
$\mathbf{k}_{pil}$	vector of pilot tone indices	N_a	$N_a = N_r = N_t$ in Ch. 5

2.2 Multi-Carrier Waveforms

Channel equalization in single-carrier waveforms involves the computationally intensive process of deconvolution, which grows more expensive as the impulse response

gets larger. For this reason, multi-carrier waveforms are introduced in this section. Defining the information carrying symbols in the frequency domain, the convolution process becomes a multiplication process, for which the computational complexity of division is a better trade off than that of deconvolution.

2.2.1 OFDM Waveforms.

As MIMO communication systems gain popularity because of their higher bandwidth efficiency, they are replacing SISO communication systems. When replacing SISO systems, the MIMO system inherits a large bandwidth with which to operate [8]. As with any large bandwidth system, multi-path effects are more prevalent. As a result, multi-carrier waveforms, such as OFDM, are more attractive because of the multi-path resistance of the waveform. Two mechanisms are utilized by OFDM to provide resistance to multi-path: frequency domain pilot assisted channel estimation, and a CP also referred to as guard time. Details of these mechanisms are described in Section 2.2.1.1 and Section 2.2.1.2, respectively.

Unfortunately, a drawback to OFDM is sensitivity to synchronization error. A popular method for synchronizing OFDM signals is exploiting the CP structure of the OFDM waveform. Correlation is used to determine when the samples repeat, corresponding to the prepended CP samples and the end of the OFDM symbol. Errors occur in this estimation technique when the Signal to Noise Ratio (SNR) is low. Another contributing factor for synchronization error is the impulse response. If the impulse response has a dominant channel tap at a positive delay $\ell^* \neq 0$ resulting from a Non Line-of-Sight (NLOS) waveform, the energy of the signal is delayed by ℓ^*. Correcting for this effect utilizes the channel estimate. Using correlation to exploit the CP structure and determining the TOR is discussed in Section 2.2.1.2.

2.2.1.1 Optimal Pilot Schemes.

Pilot assisted channel estimation assigns pilot tones in the frequency domain with the purpose of estimating the frequency response at the receiver. The number of pilot tones, the amount of power each tone is allotted from the fixed power budget and where the pilots are placed are design parameters. How these design parameters affect the channel estimate accuracy, throughput, and capacity is considered in this section. If only channel estimation accuracy is considered, all the sub-carriers should be pilot tones. This results in an accurate channel estimate but no data is transmitted. However, a channel estimate is needed to accurately receive data so using only data sub-carriers would result in a low fidelity channel. The goal of this section is to determine a balance between these criteria and maximize capacity.

It is shown in [9, 10] that the optimal number of pilots with respect to capacity is L. The frequency response is the Fourier Transform of the impulse response. The impulse response is L samples in duration and the N resulting frequency response samples only have L degrees of freedom. These L pilots are also equally spaced and the power allocated to a single pilot tone is the total power allocated to pilot tones divided by L. This results in L equally spaced and equally powered pilot tones.

Each pilot tone is allocated the same amount of power, but a pilot tone and information tone are not necessarily allocated the same power. It is shown in [9, 11] that the amount of power allocated is dependent on L and N. Their result is found by maximizing capacity which considers a balance between channel estimation accuracy and symbol estimation. Under the condition where $L = N$ equal power is allocated to both the pilots and information tones. However, when $L < N$ more power is allocated to the information tones.

So far the sub-carriers at which the pilots are located have been discussed. The amount of power allocated to the pilots over information sub-carriers has also been optimized and

the pilot tones are equally powered. The values used for the pilots have not been discussed, which is the topic of [12]. For a single transmitter, the L pilots are equal powered and equal spaced. For multiple transmitters with a flat fading, $L = 1$, channel, the pilots are also orthogonal, but for frequency selective channels ($L \geq 2$) the pilots should be phase-shift-orthogonal. The level correlation of the pilot tones across the transmitters is related to how well the pilots are suited to estimate the channel [13]. If the pilots are phase shift orthogonal, the correlation is zero, but with Radio Frequency (RF) front-ends, correlation may be introduced.

Described above are methods used for allocating pilot tones to be transmitted to the receiver for channel estimation. Unfortunately, the tones have to be known at the receiver in which case information can not be sent during this time. As an alternative, channel tracking is also a valid method of channel equalization. Where pilots and preambles are used in the beginning of transmissions to estimate the channel, then the channel fluctuations are tracked via a reduced amount of pilots or done blindly; this scheme is presented in [14]. Analysis under Rayleigh [15] or Ricean [16] channel conditions is also explored using the Extended Kalman Filter.

2.2.1.2 Cyclic Prefix Length.

The CP in OFDM waveforms offer performance boosting capabilities. The length of the CP must be chosen carefully, the CP length should be chosen to be just long enough to capture the entire impulse response of the channel. Too long, and time is used to transmit redundant information and negatively impacts capacity. Reducing the CP length increases capacity until the CP is shorter than the impulse response, in this case interference is induced which negatively impacts the fidelity of the communication link.

OFDM waveforms experience no IBI, ISI, and ICI when the CP is longer than the channel's impulse response [17–21]. In practice however, the impulse response can be longer than the cyclic prefix. The energy outside the CP degrades the system performance.

The amount of interference induced on the N sub-carriers is then characterized to obtain an estimate of the SINR of the system. If IBI, ISI, and ICI occur, choosing the Time Of Reference (TOR) intelligently mitigates the power of the interference.

The analysis of IBI, ISI, and ICI is found in [17, 18, 21]. Ref. [17] analyzes how the OFDM symbol length, CP length, and timing mismatch with a certain variance affect the SINR and capacity. This chapter also considers the trade-off between CP length and capacity. If the CP is assigned to be longer, the capacity degrades, but if the interference induced by keeping the CP the same length does not reduce the performance as much as increasing the CP length, the interference in this case is the welcomed trade-off.

The interference induced by a particular channel tap increases linearly as the channel tap index increases outside the CP. Shown in [18], the energy at channel tap $N_{cp}+1$ does not contribute to the power of the interference as much as the energy at channel tap $N_{cp} + 10$; because of this, [18] chooses the optimal TOR to reduce IBI effects. For example, the energy at channel tap $N_{cp} + 1$ is weighted by a factor of 1 and the energy at $N_{cp} + 10$ is weighted by 10. So the energy at $N_{cp} + 10$ is probably smaller but because of the weighting factor, may contribute more to the power of the interference. This calculation of the IBI is shown explicitly here [18]:

$$P_{\text{IBI}} = 2\sigma_X^2 \sum_{n\geq 0} (n+1)\left[h_a^2(n) + h_b^2(n)\right], \tag{2.1}$$

where $h_a(n)$ and $h_b(n)$ represent the portions of the impulse response outside the CP, $h_a(n)$ denotes the portion before the CP starts and $h_b(n)$ represents the portion after the CP ends.

Another technique to minimize the power of the interference, or equivalently maximize the SINR, pilot tones are used in the time domain to synchronize the Discrete Fourier Transform (DFT) window [19]. Interference is also a problem in situations where the transmitters are not synchronized at the receiver. Delay between received signals can be represented as a longer impulse response. This is the case in cellular networks where geometry introduces delay into the system. In this situation, the channel length may be

less than the CP length, but the delay between received signals makes the resulting impulse response violate this constraint [20].

Characterization of the SINR in SISO communications starts with the definition of a received OFDM signal, $d(n)$, that models an ideal circular convolution. The transmitted signal, $\bar{x}_{nt}(n)$, for this idealistic signal does not consist of a CP instead the N point OFDM block is repeated.

The traditional OFDM signal that consists of a CP is then denoted by $\bar{x}_{cp,nt}(n)$. IBI is then induced by the channel if the channel is longer than the CP in the transmitted signal. At the receiver $y(n)$ is then the received signal with effects of IBI included.

The amount of IBI at each time domain sample is then the subtraction of $d(n)$ and $y(n)$, $q(n) = d(n) - y(n)$ and put into vector form $\mathbf{q} = [q(0)\ q(1) \dots q(N-1)]^T$. The difference between the true data symbols and the estimated symbols is then $\tilde{e}_k = \frac{\tilde{q}_k}{\tilde{h}_k}$. The Power Spectral Density (PSD) of this error is then [18]

$$S_e(k) = \frac{\mathscr{F}\left\{r_q(m)\right\}}{\left|\tilde{h}(k)\right|^2}. \tag{2.2}$$

The $\mathscr{F}\{\cdot\}$ is the Fourier transform operator and $r_q(m)$ is the autocorrelation function of q. This IBI analysis is extended to MIMO communications in Section 3.1.2.

2.2.2 OFDMA.

OFDMA is a waveform used in the Long-Term Evolution (LTE) standard for allowing multiple user access. This waveform is only used in the communication from the cell phone tower to the mobile user. Each user is assigned a time-frequency slot or *resource block* where the user can transmit its data on the sub-carriers in the resource block. A resource block consists of 12 sub-carriers and data symbols are allocated to the 12 sub-carriers in the same fashion as an OFDM system [22].

A drawback of OFDMA and OFDM is that the Peak-to-Average Power Ratio (PAPR) is high [22]. A higher PAPR results in a more expensive power amplifier with a larger linear range. For this reason, OFDMA is only used in the communication from the cell

phone tower to the mobile user. For the communication from the mobile user to the cell phone tower, Single Carrier - Frequency Division Multiple Access (SC-FDMA) is used because it has a reduced PAPR compared to OFDMA and OFDM.

2.2.3 SC-FDMA.

The difference between OFDMA and SC-FDMA is an added Fast Fourier Transform (FFT) operation for SC-FDMA is needed before the symbols are defined to the subcarriers, consequently an added FFT operation is needed at the receiver as well. In the SC-FDMA waveform, the Quadrature-Amplitude Modulation (QAM) symbols are defined in the time domain, an FFT operation converts the time domain data to the frequency domain. The frequency domain coefficients are then assigned to the 12 sub-carriers allotted to the user. The Inverse Fast Fourier Transform (IFFT) is used to convert the frequency domain information to the time domain to determine the transmit waveform. With this waveform the feature of providing access to multiple users is preserved, while reducing PAPR [22].

2.3 System Model

In this section, the topology for the MIMO communication system is described. The locations for the N_t transmitters and the collocated N_r receive antennas play a vital role in SINR characterization. The relation the Ex's position has to the transmitters plays a role in the expected performance that is obtained by the Ex. Following that, the model for traditional MIMO communications is outlined along with Channel Estimate (CE) algorithms for the simulations.

2.3.1 Proposed Topology.

Figure 2.1 shows the topology of the transmitters and receivers. The Region of Activity (ROA) is pictured, which represents the area in which transmitters are located. The transmitters are communicating with Rx in the direction of $\theta = 0$, where θ is measured with respect to the line from the ROA center to Rx. At $\theta \neq 0$, an Ex is potentially present.

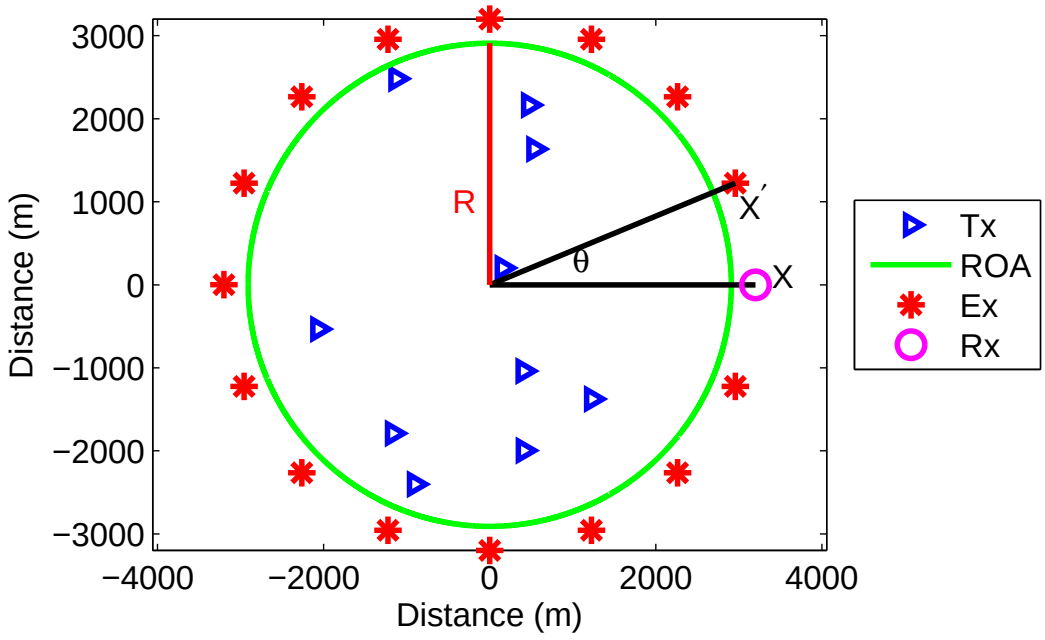


Figure 2.1: Region of Activity (ROA) of transmitters communicating with Rx. Also depicted are other eavesdropping receivers close to the ROA.

In the ROA N_t transmitters are considered to be equipped with one antenna each. The Rx and Ex are equipped with $N_r \geq N_t$ antennas each.

In the ROA, the transmitter positions are randomly distributed. First, the uniform distribution is considered for transmitter locations. The radius of the ROA then corresponds to the outer limit a transmitter can be positioned. Later, a Gaussian distribution is used to model transmitter positions where the radius of the ROA is related to the variance of the Gaussian distribution.

The transmitters are synchronized in the direction of Rx. This reduces channel effects such as IBI in the cooperative system. However, the Ex at some θ does not have this luxury of synchronizing with the transmitters. If the performance of the Ex is reduced while maintaining the fidelity for the Rx, then leveraging IBI is a valid physical layer security technique.

2.3.2 Traditional MIMO-OFDM Communications.

In traditional MIMO Communications a channel matrix, denoted $\mathbf{H} \in \mathbb{C}^{N_r \times N_t}$, represents the frequency flat complex channel between the *nt* and *nr* transmitter and receiver pair. For a frequency selective channel a received time-domain sample is modeled as:

$$\mathbf{y}_{nr} = \left(\sum_{nt=1}^{N_t} \mathbf{h}_{nr,\ nt} \star \mathbf{x}_{nt} \right) + \mathbf{n} \tag{2.3}$$

where $\mathbf{y}_{nr}$ is a series of samples indexed by n, $\mathbf{y}_{nr} = \left[y_{nr}(0)\ y_{nr}(1) \ldots y_{n_r}(N + L - 2) \right]^T$, where N is the number of subcarriers, $\mathbf{x}_{nt}$ is the transmitted signal including the N_{cp} CP samples, $N' = N + N_{cp}$, $\mathbf{x}_{nt} = \left[x_{nt}(0)\ x_{nt}(1) \ldots x_{n_t}(N' - 1) \right]^T$, $\mathbf{n}$ is Additive White Gaussian Noise (AWGN) with zero mean and power of σ_n^2, $\mathbf{n} \sim \mathcal{CN}(\mathbf{0}, \sigma_n^2 \mathbf{I})$. The Rayleigh fading frequency selective channel between the *nt* transmitter and *nr* receiver is $\mathbf{h}_{nr,\ nt}$, which is of length L with delay profile as in [23].

Since OFDM defines the data symbols in the frequency domain Equation (2.3) is analogous to

$$\tilde{\mathbf{y}}_{nr} = \left(\sum_{nt=1}^{N_a} \tilde{\mathbf{h}}_{nr,nt} \odot \tilde{\mathbf{x}}_{nt} \right) + \tilde{\mathbf{n}}, \tag{2.4}$$

where all vectors in Equation (2.4) have a length of N and $\odot$ denotes Hadamard (element wise) multiplication. Each of the elements in $\tilde{\mathbf{h}}_{nr,nt}$ represents the frequency response at a particular subcarrier, k. If the N_r symbols at a particular subcarrier are considered in vector $\tilde{\mathbf{y}}_k$ and the channel mixing, per subcarrier, is represented by the matrix $\tilde{\mathbf{H}}_k$, $\tilde{\mathbf{y}}_k$ is given by

$$\tilde{\mathbf{y}}_k = \tilde{\mathbf{H}}_k \tilde{\mathbf{x}}_k + \tilde{\mathbf{n}}_k, \tag{2.5}$$

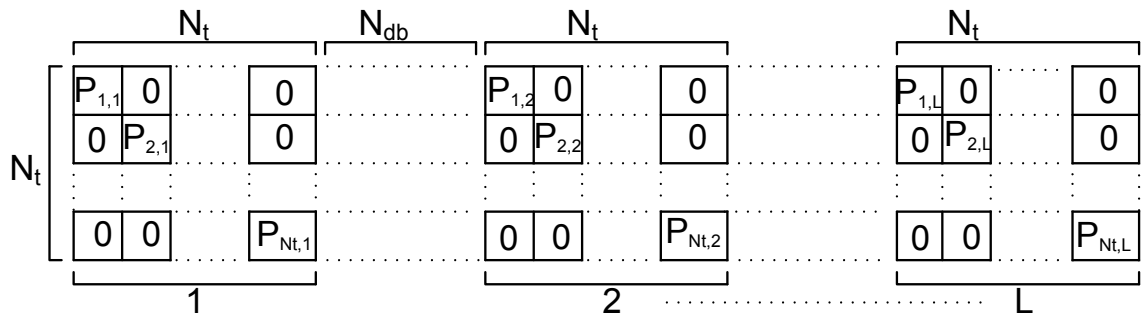


Figure 2.2: In the frequency domain pilot tones are allocated. Blocks of $N_t \times N_t$ subcarriers with pilots along the diagonal are interspersed with blocks of $N_t \times N_{db}$ used for data transmission.

where $\tilde{\mathbf{y}}_k = [\tilde{y}_{k,1} \dots \tilde{y}_{k,N_r}]^T$, $\tilde{\mathbf{H}}_k \in \mathbb{C}^{N_r \times N_t}$ represents the mixing matrix for the MIMO channel on the k^{th} subcarrier, $\tilde{\mathbf{x}}_k = [\tilde{x}_{k,1} \dots \tilde{x}_{k,N_t}]^T$, and $\tilde{\mathbf{n}}_k \in \mathbb{C}^{N_r}$ is noise.

To obtain estimates of the transmitted data symbols, $\hat{\tilde{\mathbf{x}}}_{k,nt}$, Frequency Domain Equalization (FDE) is used in OFDM systems where simple division equalizes the SISO channel. For MIMO-OFDM systems matrix inversion is used. This requires the estimation of the N channel matrices, $\tilde{\mathbf{H}}_k$, in Equation (2.5).

2.3.3 Channel Estimation Algorithms.

Two methods of CE in OFDM communications are pilot based (frequency domain) and preamble based (time domain) methods. This section discusses one type of pilot allocation then moves to preamble based CE.

2.3.3.1 Frequency Domain CE.

In Frequency Domain Channel Estimation (FDCE) pilot-tones are used at the receiver to estimate the Channel State Information (CSI). The number of pilot-tones used has been shown to be the number of degrees of freedom for the frequency response. For now, L is assumed to be known. Each pilot symbol is denoted by $P_{nt,\ell}$.

Consider the first transmitter. The first pilot is placed in the first subcarrier, $P_{1,1}$. The first pilot for the second transmitter is placed on the second subcarrier, $P_{2,1}$, and the first subcarrier is zeroed for the second transmitter. This ensures that at the first subcarrier there is no interference between the two transmitters. Likewise, the first transmitter has its second subcarrier zeroed for the same reason. This scheme is extended for all transmitters and for all L pilots. Once the pilots and zeroed subcarriers are assigned, the remaining $N_t - 1$ blocks of subcarriers can be assigned with the payload data. Each block is $N_{db} = \frac{N - N_t L}{N_t - 1}$ subcarriers wide for each transmitter.

At the nr^{th} receiver, the frequency response is estimated for each of the pilots allocated at the transmitter. Since the pilots are separated in frequency, frequency response estimates at the pilot subcarriers are calculated by division

$$\hat{\tilde{\mathbf{p}}}_{nr,nt}(\ell) = \frac{\tilde{\mathbf{y}}_{nr}(\mathbf{k}_{nt}(\ell))}{P_{nt,\ell}}, \tag{2.6}$$

where $\hat{\tilde{\mathbf{p}}}_{nr,nt}(\ell)$ are the L frequency response estimates at the pilot-tone subcarriers as a function of transmitter denoted $\mathbf{k}_{nt}(\ell)$, such that $\tilde{\mathbf{x}}_{nt}(\mathbf{k}_{nt}(\ell)) = P_{nt,\ell}$.

This method for estimating the frequency response coefficients is the same as in a SISO communication system, since frequency separation is being relied on to reduce interference for pilot-tones. However, $\hat{\tilde{\mathbf{p}}}_{nr,nt}(\ell)$ contains L estimates, however the full N frequency response coefficients are needed to spatially separate the payload data symbols.

Interpolation is used to determine estimates for the rest of the N subcarriers in the frequency response. For this, a sub-matrix of the N-point DFT matrix, $\mathcal{F}_N$, is used. $\mathcal{W}_{L,nt}$ is defined as

$$\mathcal{W}_{L,nt} = \mathcal{F}_N(1 : L, \mathbf{k}_{nt}), \tag{2.7}$$

where $\mathcal{W}_{L,nt}$ consists of the first L rows of $\mathcal{F}_N$ and the columns that correspond to the pilot tone subcarrier indices as a function of the transmitter number. The L frequency domain samples, $\hat{\tilde{\mathbf{p}}}_{nr,nt}$, are used to estimate the impulse response,

$$\hat{\mathbf{h}}_{nr,nt} = \mathcal{W}_{L,nt}^{-1} \hat{\tilde{\mathbf{p}}}_{nr,nt}, \tag{2.8}$$

which is used to calculate the frequency response by zero-padding the L samples with $N - L$ zeros and performing an FFT operation to achieve the N sample frequency response. Equation (2.8) is used for each transmitter and receiver pair to obtain the full $N_r N_t$ frequency responses. These estimates are then used in the Least Squares (LS) solution to channel equalization,

$$\hat{\tilde{\mathbf{x}}} = \left(\tilde{\mathbf{H}}_k^H \tilde{\mathbf{H}}_k\right)^{-1} \tilde{\mathbf{H}}_k^H \tilde{\mathbf{y}}_k. \tag{2.9}$$

2.3.3.2 Time Domain CE.

Pilot-tones are not used in Time Domain Channel Estimation (TDCE), instead the entire transmitted signal is known. In this case a preamble is known at the receiver for this type of CE. The benefits to this method is that there are many more known values with which to estimate the CSI, however, no data is transmitted during the preamble's duration. For this to be a valid form of CE, the channel coherence time must be longer than the preamble [24]. The longer the channel is coherent, the longer the current CSI estimate can be used to equalize received data.

This section outlines the matrix structure used at the receiver to estimate the $N_r N_t$ impulse responses. A matrix, $\mathbf{X}$, denotes this matrix that considers the N_t transmit signals, the delay experienced for each transmitter, $\mathbf{d}$, and the number of taps in the impulse response to be estimated, L. First, $\mathbf{x}_{nt,cp} = \left[x_{nt}(N - N_{cp}) \ldots x_{nt}(N)\ x_{nt}(0) \ldots x_{nt}(N-1)\right]$ with length $N' = N + N_{cp}$ is considered along with the delay experienced for each transmitter which is used to prepend zeros to $\mathbf{x}_{nt,cp}$:

$$\mathbf{x}_{nt,delay} = \begin{bmatrix} \mathbf{0}_{d(nt)} \\ \mathbf{x}_{cp,nt} \\ \mathbf{0}_{max(\mathbf{d})-d(nt)} \end{bmatrix}. \tag{2.10}$$

$\mathbf{X}_{cp}$ consists of N_t columns and $N' + \max(\mathbf{d})$ rows;

$$\mathbf{X}_{cp} = \left[\mathbf{x}_{1,delay} \ldots \mathbf{x}_{N_t,delay}\right]. \tag{2.11}$$

Next, zeros are prepended to $\mathbf{X}_{cp}$ for each value of $\ell = 0, 1, \ldots, (L-1)$,

$$\mathbf{X}_\ell = \begin{bmatrix} \mathbf{0}_{\ell,N_t} \\ \mathbf{X}_{cp} \\ \mathbf{0}_{L-\ell-1,N_t} \end{bmatrix}. \tag{2.12}$$

The L matrices that Equation (2.12) defines are then used to make $\mathbf{X}$;

$$\mathbf{X} = [\mathbf{X}_0 \ldots \mathbf{X}_{L-1}]. \tag{2.13}$$

The structure of the $\mathbf{X}$ matrix then determines the structure of $\mathbf{h}_{nr}$ where,

$$\mathbf{h}_{nr} = \begin{bmatrix} \mathbf{h}_{nr,1} \\ \vdots \\ \mathbf{h}_{nr,\ell} \\ \vdots \\ \mathbf{h}_{nr,L-1} \end{bmatrix} \text{ where } \mathbf{h}_{nr,\ell} = \begin{bmatrix} h_{nr,1,\ell} \\ h_{nr,2,\ell} \\ \vdots \\ h_{nr,N_t,\ell} \end{bmatrix}. \tag{2.14}$$

The received signal, $\mathbf{y}_{nr}$, is then modeled as

$$\mathbf{y}_{nr} = \mathbf{X}\mathbf{h}_{nr} + \mathbf{n} \tag{2.15}$$

with the dimensions of $\mathbf{X}$ being $(N' + (L-1) + \max(\mathbf{d})) \times (N_t L)$, $\mathbf{h}_{nr}$ is $(N_t L \times 1)$, and both $\mathbf{y}_{nr}$ and $\mathbf{n}$ are $(N' + (L-1)) \times 1$. To estimate $\mathbf{h}_{nr}$ a LS solution is used [25] where

$$\hat{\mathbf{h}}_{nr} = \left(\mathbf{X}^H\mathbf{X}\right)^{-1}\mathbf{X}^H\mathbf{y}_{nr}. \tag{2.16}$$

Equation (2.16) is used at each receiver, to obtain an impulse response estimate for each transmitter and receiver pair. The equalization process uses this channel estimate in Equation (2.9).

2.4 Multi-User TOA Estimation Algorithms

This system described in this section consists of N_t users transmitting to one receiver. The users are allocated a subset of available subcarriers, N. If each user is allocated $\frac{N}{N_t}$ subcarriers they may not be contiguous. This happens when the users are dynamically allocated subcarriers [26].

Considering a single user where bits are mapped to a constellation and are used to assign the subcarriers in which they were alloted. $\mathcal{S}_{nt}$ denotes the set of subcarriers allocated to user nt. The frequency domain signal for user nt is then

$$\tilde{\mathbf{x}} = [x_1 \; x_2 \ldots x_N]^T \tag{2.17}$$

where k indexes the subcarriers. A puncture matrix is defined where the diagonal elements $\mathbf{P}_{k,k} = 0$ if $k \notin \mathcal{S}_{nt}$. The N point DFT matrix is denoted $\mathcal{F}_N$ and the CP is added with the operator, $\mathbf{Q}$

$$\mathbf{Q} = \begin{bmatrix} \mathbf{0}_{N_{cp} \times (N - N_{cp})} \; \mathbf{I}_{N_{cp}} \\ \mathbf{I}_N \end{bmatrix}. \tag{2.18}$$

The baseband signal to be transmitted for each user is then given by

$$\mathbf{x}_{cp,nt} = \mathbf{Q}\mathcal{F}_N\mathbf{P}\tilde{\mathbf{x}}_{nt}. \tag{2.19}$$

To simplify the analysis a frequency flat fading is assumed, however in the implementation in Section 4.3 this assumption is lifted. The Time of Arrival (TOA) for each user at the receiver is a function of two parameters. The first is local clock differences between the transmitter and receiver. $\phi_{nt} = \tau_{Rx} - \tau_{nt}$ where τ_{Rx} denotes the local clock at the receiver and τ_{nt} is the local clock at the transmitter. θ_{nt} is then the propagation delay experienced for each user based on the position of the user. The impulse response experienced by each user is

$$\mathbf{h}_{nt} = \left[\mathbf{0}_{\phi_{nt}}^T \; \mathbf{0}_{\theta_{nt}}^T \; h_{nt}\right]^T \tag{2.20}$$

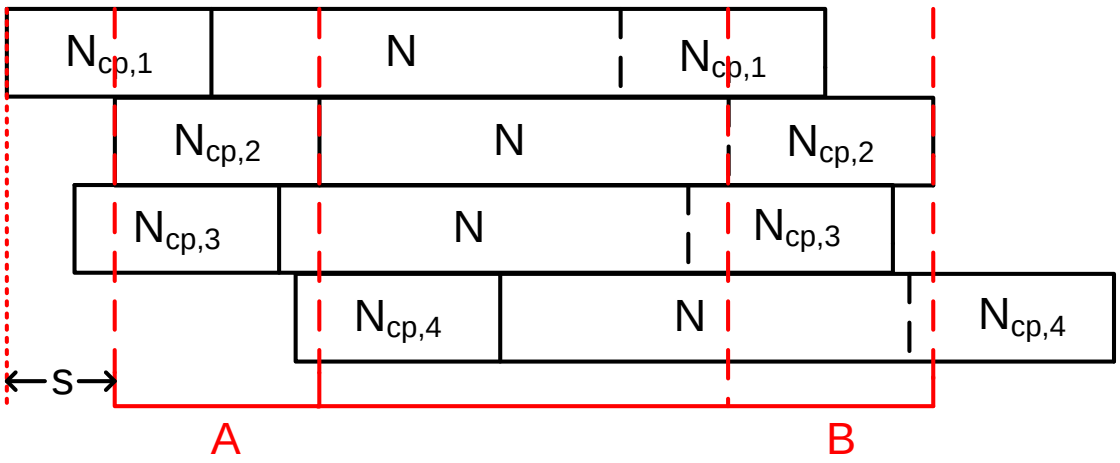


Figure 2.3: Illustration of the van de Beek method for determining the TOA for OFDM symbols where the CP is leveraged.

where h_{nt} is the complex scalar denoting the frequency flat fading channel between the nt user and the receiver. The total delay experienced at the receiver is $\bar{\theta}_{nt} = \phi_{nt} + \theta_{nt}$. The received signal $\mathbf{y}$ is then

$$\mathbf{y} = \left(\sum_{nt=1}^{N_t} \mathbf{x}_{cp,nt} \star \mathbf{h}_{nt} \right) + \mathbf{n} \tag{2.21}$$

where $\mathbf{n}$ is complex AWGN and $\star$ denotes convolution.

2.4.1 van de Beek Method.

The blind estimation of the TOA for OFDM leverages the structure of the CP in the received waveform. The correlation between blocks that are N_{cp} in length and are $N - N_{cp}$ samples apart peak when the symbol is aligned with the sliding window indexed by s. An illustration of this method is shown in Figure 2.3.

The maximum likelihood estimator for $\bar{\theta}$ is given by [27]

$$\hat{\bar{\theta}} = \underset{\bar{\theta}}{\operatorname{argmax}} \left\{ \left| \gamma(\bar{\theta}) \right| - \rho \Phi(\bar{\theta}) \right\} \tag{2.22}$$

where

$$\gamma(s) = \sum_{m=s}^{s+N_{cp}-1} y(m) y^*(m+N), \tag{2.23}$$

ρ is the magnitude of the correlation coefficient between $y(m)$ and $y(m+N)$ and

$$\Phi(m) = \frac{1}{2} \sum_{m=s}^{s+N_{cp}-1} \left[|y(m)|^2 + |y^*(m+N)|^2 \right]. \tag{2.24}$$

Equation (2.22) is derived in [27] with consideration given to Carrier Frequency Offset (CFO), which is assumed to be zero here and the second term, $\rho\Phi(\theta)$, is assumed to be negligible in a system where the SNR is constant.

When determining the TOA for N_t users Equation (2.22) is used for each of the users. Once a maximum is found at an s, $\gamma\left(s \pm N_{cp}\right)$ is set to zero and the next maximum is found. In this scheme, an estimate of delay can not be matched to a particular user definitively. This would not be a valid method for estimating the TOA for the N_t users for Ex since the delay for each user is needed. The next method described, the *Acharya* method, does provide specific user delay.

2.4.2 Acharya Method.

The log-likelihood function to be maximized is [26]

$$\mathcal{L} = \log\ p(\mathbf{y}|\bar{\theta}_{nt}). \tag{2.25}$$

The Probability Density Function (PDF), given the delay $\bar{\theta}_{nt}$, is

$$p(\mathbf{y}|\bar{\theta}_{nt}) = \frac{1}{\pi^{N'+N}\,|\mathbf{C}_{\bar{\theta}}|} \exp\left\{ -\frac{1}{2}\mathbf{y}^H \mathbf{C}_{\bar{\theta}}^{-1} \mathbf{y} \right\}, \tag{2.26}$$

where $\mathbf{C}_{\bar{\theta}}^{-1}$ is the covariance matrix as a function of the delay to be estimated. $\mathbf{C}_{\bar{\theta}}^{-1}$ is

$$\mathbf{C}_{\bar{\theta}} = \mathrm{diag}\left[\mathbf{I}_{\theta},\ \tilde{\mathbf{C}},\ \mathbf{I}_{N-\theta} \right] \tag{2.27}$$

where $\tilde{\mathbf{C}} = \mathrm{E}\left\{\mathbf{y}\mathbf{y}^H\right\}$ is the received covariance matrix. In Ref. [26] the covariance matrix is broken into subblocks for the samples corresponding to the CP, denoted by $\mathbf{u}$ and the $N - N_{cp}$ remained samples denoted by $\mathbf{v}$. They then define $\mathbf{X}$, $\mathbf{Y}$ and $\mathbf{Z}$ as:

$$\mathbf{X} = \mathrm{E}\left\{\mathbf{u}\mathbf{u}^H\right\},\ \mathbf{Y} = \mathrm{E}\left\{\mathbf{u}\mathbf{v}^H\right\},\ \mathbf{Z} = \mathrm{E}\left\{\mathbf{v}\mathbf{v}^H\right\}. \tag{2.28}$$

$\tilde{\mathbf{C}}$ is then provided in [26] as:

$$\tilde{\mathbf{C}} = \begin{bmatrix} \sigma_n^2\mathbf{I}_L + \mathbf{X} & \mathbf{Y} & \mathbf{X} \\ \mathbf{Y}^H & \sigma_n^2\mathbf{I}_{N-L} + \mathbf{Z} & \mathbf{Y}^H \\ \mathbf{X} & \mathbf{Y} & \sigma_n^2\mathbf{I}_L \end{bmatrix} \tag{2.29}$$

The estimate for $\bar{\theta}$ is found by maximizing Equation (2.25):

$$\hat{\bar{\theta}} = \underset{\theta}{\operatorname{argmin}} \log\left(|\mathbf{C}_{\bar{\theta}}|\right) + \frac{1}{2}\mathbf{y}^H\mathbf{C}_{\bar{\theta}}^{-1}\mathbf{y}. \tag{2.30}$$

2.5 FPGA Overview

An FPGA, as the name implies, is a programmable gate array. A gate array is a series of logic blocks that are configured via software. On the FPGA itself is a blank slate of logic blocks and VHDL or Verilog is used to program these logic blocks to do a specific task. This is opposed to a Central Processing Unit (CPU) or Graphical Processing Unit (GPU) in the sense that instructions are provided to the processing unit and the instructions are executed in order, ignoring parallel processing for now, the instructions are given, executed, and the result is provided. In an FPGA the hardware itself is programed. If a multiplier is needed, a multiplier is instantiated in the logic blocks. The power of the FPGA comes from the ability to perform calculations in parallel, for example Multiply Accumulate (MAC) operations, if 256 MACs are needed in a FFT operation, 256 MACs are instantiated and can be completed concurrently.

Parallel processing occurs in a CPU or a GPU as well. In these processors, there are multiple units of a particular function such as the MACs in the above example, but the number of the MAC units are not variable as in an FPGA. Making use of an FPGA to develop hardware specific to the application provides faster computation times, making FPGA development a very powerful tool.

2.5.1 MIMO Receiver Latency and Throughput.

A receiver is *real-time* compliant if each calculation block in the MIMO receiver can maintain a constant data rate. If the data rate going into the block is higher than the data rate

going out of the block, the block is then not real-time compliant. A possible architecture change or simply reducing the data rate are possible solutions. The latter is not ideal since the reason for investigating MIMO communications is to maximize the data rate.

A receiver may also be *burst mode* capable where cycles of communication and waiting are provided to the receiver. During the communication section, data is transmitted and the receiver synchronizes to the data and begins the demodulation process. The demodulation process continues into the waiting section to allow for further computations. The duty cycle ratio of communication and waiting times is known at the transmitter and does not overwhelm the receiver. A *real-time* compliant receiver may also operate in *burst mode* but the transmitter may send data as often as needed.

If the computational complexity of the MIMO receiver is too high for a specific field grade platform, logging data and using a super computer with more resources is possible [28]. This situation is not investigated in this paper, but is included for completeness. This paper first focuses on designing a MIMO receiver that is real-time compliant. Then this design criteria is lifted to provide a reduced data rate *Burst Mode* communication system in which the amount of resources used is reduced allowing for a more flexible implementation.

2.5.2 FPGA Resources.

Figure 2.4 shows a diagram of how the FPGA fabric is laid out. Logic blocks are shown in a grid pattern with dedicated multipliers and Random Access Memory (RAM) blocks. The dedicated multipliers save logic blocks since the multiplication is an expensive operation and used frequently. The operands and results are saved in RAM Blocks on the FPGA next to the logic blocks.

In complex designs, careful consideration is taken into resource utilization. For example, the logic blocks that make up the FFT operation are instantiated on the FPGA close to the RAM that stores the time domain data. This avoids the issue of logic switches being used for transporting time domain values to the FFT block. However, in

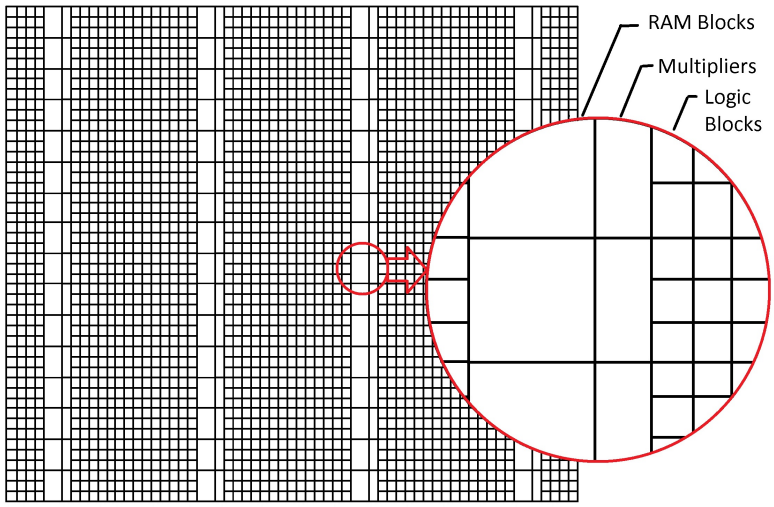


Figure 2.4: FPGA fabric with resources highlighted [29].

practice transceiver designs are complex and it is unavoidable to use some logic switches to transport results from a process to another process.

2.5.3 WARP Board.

The WARP Board is a Software Defined Radio (SDR) developed by Rice University for physical layer development. The boards have been used in some network protocol research [30], but most experiments using the WARP Boards are in the area of wireless communications, for example, WARPLab [30]. A major benefit to the boards for MIMO communications is the higher end clock support, which is then shared to the four radio cards.

The WARP Boards used in this project are the second version Rice has developed which has an FPGA and an embedded processor on the board. The FPGA is the Xilinx Virtex-4 and has IBM's Performance Optimization With Enhanced RISC Performance Computing (PPC)-405 processor as well. The FPGA and PPC work in tandem to control a multitude of peripheral devices. These devices include [30]

1. DDR2 SO-DIMM Slot 2GB SO-DIMM Installed

2. Daughter-card Slots

3. Push Button, LEDs, DIP Switches, and Hex Displays

4. Ethernet Port

5. USB and Serial UART

6. 16-Bit Digital I/O Header Pins

7. Multi-Gigabit Transceivers, HSSDC2, SATA, and SFP

Item 2 provides an interface to a radio card, where four radio cards can be installed on the main board. The radio cards are the RF front-ends that allow the FPGA to transmit and receive signals in the ISM bands, 2.4 and 5 GHz ranges. The radio card specifications are described in Table 2.2.

Table 2.2: Radio card operating parameters

Parameter Name	Parameter Value
Digital to Analog Converters	160 MS/s 16-bit
Analog to Digital Converters	65 MS/s 14-bit
Dual Band Operation	2.4 - 2.48 GHz and 4.9 - 5.875 GHz
Bandwidth	40 MHz
RSSI Range	60 dB
Tx Power Control Range	30 dB
Rx Gain Control Range	93 dB
Output Power at Full Gain	18 dBm

2.5.4 WARPLab.

WARPLab is a project developed by Rice University and maintained in an open-source environment [30]. This allows MATLAB® to control the WARP Board via an Ethernet connection. More than one board can be controlled by a single computer by the use of an Ethernet switch with a simple local network.

The WARPLab reference design provides a vehicle for baseband signals designed in MATLAB to be loaded onto the board. Also the reference design allows transmit and receiver parameters like center frequency, gain etc to be set, and then allows for transmit and receiver operation to commence. With this system the transmitter and receiver synchronization is rough due to Ethernet jitter, but it provides a mechanism for real world transmit and received signals to be analyzed while remaining in a MATLAB environment.

III. Covert MIMO Communication

In Chapter I a scenario is described where information is shared between many entities. The scenario is further described as a hostile environment. The topology for this hostile environment is outlined in Section 2.3.1. For this system a mechanism is needed to provide level of security.

In this chapter, IBI is used to degrade the eavesdropping receiver's performance by designing the system with a CP that is just long enough for synchronized communication among many uplink users. Synchronized communication is done in cell phone networks to provide resource blocks to users with division in time and frequency [22], and a similar process is used to ensure zero delay between user waveforms.

Characterization of the effect of IBI is developed as a function of delay based on the distribution of the users' locations. In the system analyzed, the delay between transmit waveforms is the only source of IBI. The distribution of user locations is calculated for normal and uniformly distributed transmitter locations.

Section 3.1 analyzes the distributions of delays in the topology defined in Section 2.3.1 then the SINR is derived for the MIMO communication system. The SINR and BER performance as a function of angle is simulated and compared to the analytical SINR and BER in Section 3.2.

3.1 SINR and Bit Error Rate Derivation

In this section, the distribution of delays is derived as a function of transmitter location. Since delay induces IBI on the system, a derivation of SINR for the MIMO system follows the derivation of delay distribution. The expected SINR for given channel is calculated. Using the SINR the theoretical BER is calculated with the assumption that the noise including IBI follows a Gaussian distribution.

3.1.1 Distributions of Delays.

Asynchronous delay between transmissions effectively makes the impulse response longer. A longer impulse response may potentially violate $N_{cp} \geq L - 1$ which would have a degrading effect on the fidelity on the communication system. The aim of this section is to characterize the delay as a function of θ, once this is accomplished the delay is used in calculating the interference induced by the delay.

In Figure 2.1 the ROA is defined as the area in which a transmitter can be located. N_t transmitters are considered and their locations in the ROA are randomly distributed. In the following sections the formulation of relative distances that affect delay are introduced. Based on these distances the distributions of arrival times, with respect to the center of the ROA, are considered with the transmitters being uniformly distributed and normally distributed. The distribution of arrival times as a function of θ is derived and compared to simulations.

3.1.1.1 General Delay.

The absolute delay between a transmitter to the receiver is not of concern but the relative delay between two transmissions needs to be characterized. For this, the difference in path length between two transmitters is the important parameter. To simplify analysis the small angle approximation is made to simplify the following derivation, as shown in Figure 3.1. The impact of this assumption is further explored in Sections 3.1.1.4 and 3.2.1.3. This assumption simplifies the delay dependence to only the x dimension, the y dimension is assumed to be negligible.

Recall the transmitters add delays so that the received signals are synchronized in the $\theta = 0$ direction. For synchronization purposes, D_1 denotes the delays in the $\theta = 0$ direction, which simply relates the x coordinate to propagation time:

$$D_1 = \frac{x}{c}, \tag{3.1}$$

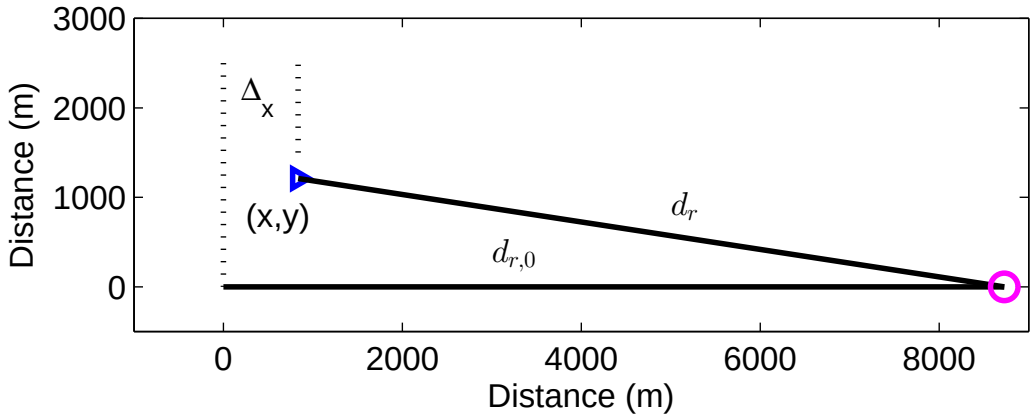


Figure 3.1: Depiction of small angle assumption. $D_1 = \frac{1}{c}(d_r - d_{r,0}) \approx \frac{\Delta_x}{c}$

where c is the speed of light. Since y is considered to be small in relation to the distance the signal is propagated Equation (3.1) does not depend on y. In Equation (3.1), x represents the x coordinate of a randomly located transmitter. Values for D_1 are negative when the transmitter is closer to the receiver and are positive when the transmitter is further away from the receiver with respect to the origin (ROA center).

For $\theta \neq 0$, D_2 denotes the delays in the direction of θ, for this, a coordinate rotation is needed for the analysis where:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}. \tag{3.2}$$

For the rotated coordinates and the D_2 calculation, D_2 is similar to D_1 in the rotated coordinates, however the dependence on x and y are needed explicitly:

$$D_2 = \frac{x'}{c} \tag{3.3}$$

$$= \frac{1}{c}(x\cos\theta + y\sin\theta). \tag{3.4}$$

The Equations for D_1 and D_2 are used in determining the delay distribution for all θ. D_2 represents the delays experienced as a function of θ without synchronization. By subtracting the delay in the $\theta = 0$ direction, namely D_1, the system is then synchronized in the $\theta = 0$ direction. This difference is then Δ defined as

$$\begin{aligned} \Delta &= D_2 - D_1 && (3.5) \\ &= \left(\frac{\cos\theta - 1}{c}\right)x + \left(\frac{\sin\theta}{c}\right)y. && (3.6) \end{aligned}$$

Here, Δ is a function of θ and represents the arrival time in seconds in relation to a hypothetical transmitter placed at the origin. For one realization of Δ one transmitter is considered with coordinates (x_{tx}, y_{tx}), and consider the case where $\theta = 0$, this yields $\Delta = \frac{1}{c}(x_{tx}\cos(0) + y_{tx}\sin(0)) - \frac{x_{tx}}{c} = 0$ which holds $\forall x_{tx},\ y_{tx}$.

For $\theta \neq 0$ and with random transmitter locations, Δ is also random. To determine the distribution of Δ the distribution of x is needed. The PDF of Δ, $f(\Delta)$ is related to $f(x)$ by [39]

$$f(\Delta) = f(x)|_{x=g(\Delta)}\,\frac{dg}{d\Delta}. \tag{3.7}$$

To determine $g(\Delta)$, Equation (3.6) is considered, where Equation (3.6) defines a line for a given Δ, shown in Figure 3.2. In general, the distance between the line defined by Δ and the origin is [40]

$$\begin{aligned} x' &= \frac{\Delta c}{\sqrt{(\cos\theta - 1)^2 + (\sin\theta)^2}} && (3.8) \\ &= c\alpha\Delta && (3.9) \\ \text{where} \quad \alpha &= \frac{1}{\sqrt{(\cos\theta - 1)^2 + (\sin\theta)^2}} && (3.10) \end{aligned}$$

3.1.1.2 Uniform Distributed Transmitters.

The general analysis above does not consider a specific distribution for x and y. For a ROA of radius R, the PDF of the x and y coordinates are

$$f(x, y) = \frac{1}{\pi R^2} \quad \text{where } x^2 + y^2 \leq R^2. \tag{3.11}$$

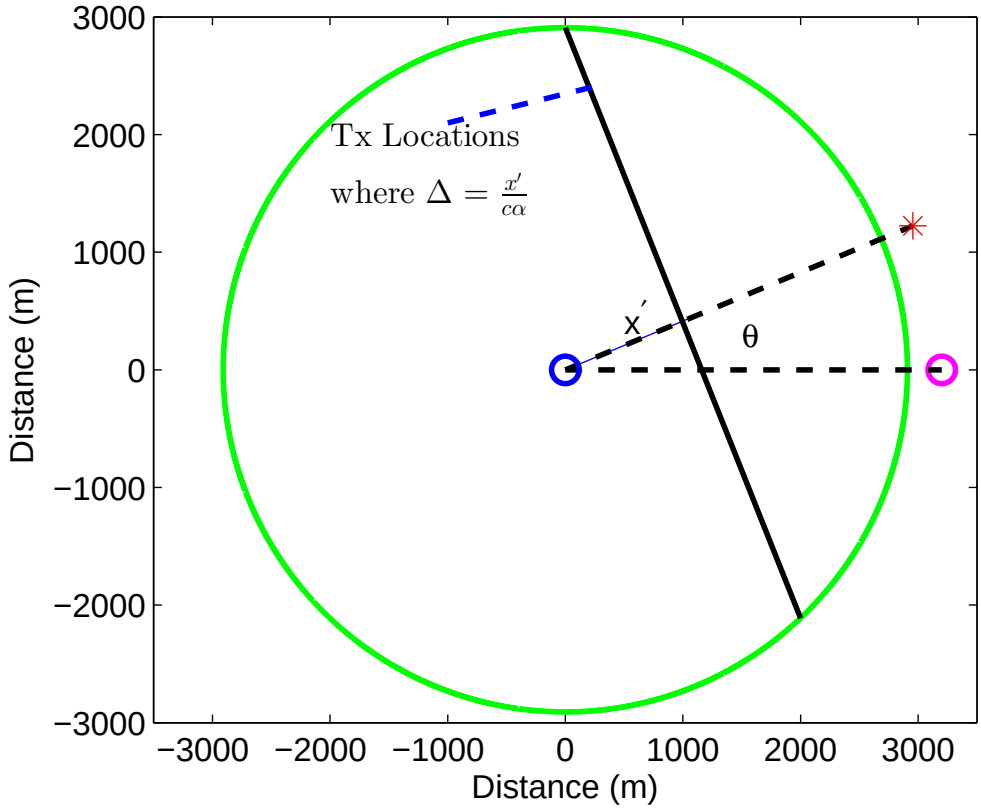


Figure 3.2: Depiction of the transmitters positions with delay $\Delta(\theta)$

Since Δ is a function of x only, and y is considered negligible, the PDF of x is

$$f(x) = \int_{-\infty}^{\infty} f(x,y)\,dy \tag{3.12}$$

$$= \int_{-\sqrt{R^2-x^2}}^{\sqrt{R^2-x^2}} \frac{1}{\pi R^2}\,dy \tag{3.13}$$

$$= \frac{2\sqrt{R^2-x^2}}{\pi R^2} \tag{3.14}$$

Substituting into Equation (3.7):

$$f(\Delta) = \frac{2c\alpha\sqrt{R^2-(c\alpha\Delta)^2}}{\pi R^2}, \tag{3.15}$$

which is the distribution of the delays at Ex w.r.t. Rx for a given θ. Further analysis of the delay is discussed in Section 3.2.1.

3.1.1.3 Gaussian Distributed Transmitters.

The definition of the ROA is slightly different for the Gaussian distributed transmitters, just in the sense that the Gaussian distribution has infinite support. For this, σ_R is introduced as the standard deviation used for the Gaussian PDF. The distribution of the coordinates of the transmitters is:

$$f(x, y) = \frac{1}{2\pi\sigma_R^2} \exp\left[-\frac{1}{2\sigma_R^2}(x^2 + y^2)\right]. \tag{3.16}$$

Since the coordinates are independent, and the x coordinate is the only coordinate of interest the scalar Gaussian distribution of x is used,

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma_R} \exp\left[-\frac{1}{2\sigma_R^2}x^2\right]. \tag{3.17}$$

Here, Equation (3.7) and $x' = c\alpha\Delta$ still holds. Substitution provides the distribution of Δ when the transmitters are distributed following a Gaussian distribution:

$$\begin{aligned} f(\Delta) &= \frac{c\alpha}{\sqrt{2\pi}\sigma_R} \exp\left[-\frac{c^2\alpha^2}{2\sigma_R^2}\Delta^2\right] && (3.18)\\ &= \frac{1}{\sqrt{2\pi}\frac{\sigma_R}{c\alpha}} \exp\left[-\frac{1}{2\left(\frac{\sigma_R}{c\alpha}\right)^2}\Delta^2\right] && (3.19)\\ &= \frac{1}{\sqrt{2\pi}\sigma_\Delta} \exp\left[-\frac{1}{2\sigma_\Delta^2}\Delta^2\right]. && (3.20) \end{aligned}$$

where $\sigma_\Delta = \frac{\sigma_R}{c\alpha}$.

3.1.1.4 Close vs. Far Receiver and Eavesdropper.

Equations (3.1) and (3.4) represent the delay in seconds when the y coordinate is considered to be negligible. In the following, this assumption is removed and we show why this approach is intractable. Figure 3.3 shows the distances that are of interest. The

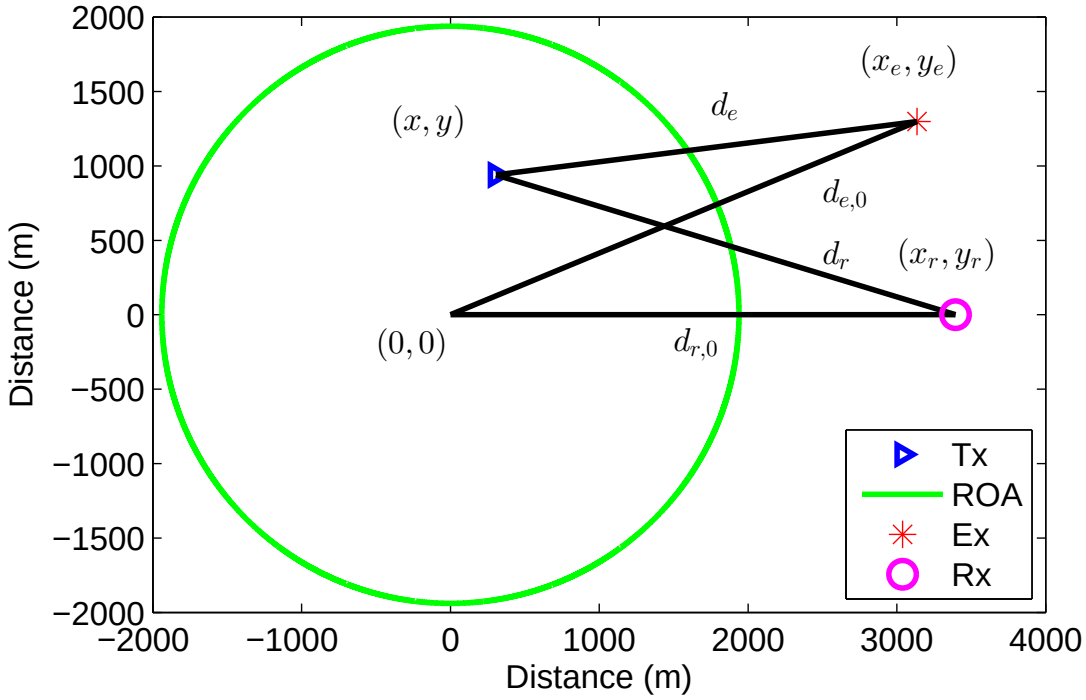


Figure 3.3: Illustration of distances of interest: $d_{r,0}$, d_r, $d_{e,0}$, and d_e

general distance formulas for the four distances in Figure 3.3 are

$$\begin{aligned} d_{r,0} &= \sqrt{x_r^2 + y_r^2}, \\ d_r &= \sqrt{(x - x_r)^2 + (y - y_r)^2}, \\ d_{e,0} &= \sqrt{x_e^2 + y_e^2}, \\ d_e &= \sqrt{(x - x_e)^2 + (y - y_e)^2}. \end{aligned}$$

The delay in seconds, analogous to Equations (3.1) and (3.4), are defined here:

$$\begin{aligned} D_1 &= \frac{1}{c}\left(d_r - d_{r,0}\right), \\ D_2 &= \frac{1}{c}\left(d_e - d_{e,0}\right). \end{aligned}$$

Δ is defined the same way:

$$\Delta = D_2 - D_1.$$

Equations for x and y are needed for substitution into the general form of Equation (3.7). To do this, the square roots are removed by solving for values that do not depend on the transmitter location, $d_{r,0}$ and $d_{e,0}$, then squaring the resulting equations. First, $c\Delta$ is found explicitly,

$$c\Delta = d_e - d_{e,0} - d_r + d_{r,0}. \tag{3.21}$$

Let $\beta = (c\Delta + d_{e,0} - d_{r,0})$. Removing all square roots from Equation (3.21),

$$d_r^2 = \left(\frac{d_e^2 - \beta^2 - d_r^2}{2\beta}\right)^2$$

Unfortunately, a fourth order polynomial is not solvable for x and y. This approach is analytically intractable. However, simulations of this system are designed, and are presented in Section 3.2.1.3.

3.1.2 SINR Derivation.

In MIMO communication systems consideration for each of the N_rN_t transceiver pairs is needed because each pair contributes to the overall IBI. For this reason $i(n)$, $y(n)$ and $q(n)$ are defined for each transmitter and receiver pair.

$$\begin{aligned} i_{nr,nt}(n) &= \sum_{\ell=0}^{L-1} h_{nr,nt}(\ell)\bar{x}_{nt}(n-\ell), \\ y_{nr,nt}(n) &= \sum_{\ell=0}^{L-1} h_{nr,nt}(\ell)\bar{x}_{cp,nt}(n-\ell), \\ q_{nr,nt}(n) &= i_{nr,nt}(n) - y_{nr,nt}(n). \end{aligned}$$

The $q_{nr,nt}(n)$ samples are put into a vector $\mathbf{q}_{nr,nt} = [q_{nr,nt}(0)\ q_{nr,nt}(1)\ldots q_{nr,nt}(N-1)]^T$, and $\tilde{\mathbf{q}}_{nr,nt}$ describes $\mathbf{q}_{nr,nt}$ in the frequency domain. However, the equalization is more complicated in the MIMO system. Just as in Section 2.3.3 where a LS solution is used to estimate the channel, an LS approach is used to equalize the N_r received symbols on

subcarrier k, producing the N_t estimated transmitted symbols, $\hat{\tilde{\mathbf{x}}}_k$ at each subcarrier by inverting the channel matrix:

$$\hat{\tilde{\mathbf{x}}}_k = \left(\tilde{\mathbf{H}}_k^H \tilde{\mathbf{H}}_k\right)^{-1} \tilde{\mathbf{H}}_k^H \tilde{\mathbf{y}}_k.$$

To characterize IBI specifically, the channel is assumed to be known for each subcarrier. Section 2.3.3 considers a CE process in an IBI inducing environment. The error in the frequency domain is the difference between the estimated symbols and the true symbols:

$$\begin{aligned}
\tilde{\mathbf{e}}_k &= \hat{\tilde{\mathbf{x}}}_k - \tilde{\mathbf{x}}_k \\
&= \tilde{\mathbf{H}}_k^{-1} \tilde{\mathbf{y}}_k - \tilde{\mathbf{x}}_k \\
&= \tilde{\mathbf{H}}_k^{-1} (\tilde{\mathbf{H}}_k \tilde{\mathbf{x}}_k + \tilde{\mathbf{q}}_k) - \tilde{\mathbf{x}}_k \\
&= \tilde{\mathbf{x}}_k + \tilde{\mathbf{H}}_k^{-1} \tilde{\mathbf{q}}_k - \tilde{\mathbf{x}}_k \\
&= \tilde{\mathbf{H}}_k^{-1} \tilde{\mathbf{q}}_k. \qquad (3.22)
\end{aligned}$$

The error vector calculated in Equation (3.22) is an $(N_t \times 1)$ vector representing the error induced on the system by IBI for each estimated transmit data symbol. For further analysis consider an element of $\tilde{\mathbf{e}}_k$

$$\tilde{e}_{k,nt} = \sum_{nr=1}^{N_r} \tilde{W}_{nt,nr,k}\, \tilde{q}_{nr,k},$$

where $\tilde{\mathbf{W}}_k = \tilde{\mathbf{H}}_k^{-1}$. To find the PSD of the error, $S_{\tilde{e}_{k,nt}}(f)$ the auto correlation function $r_{\tilde{e}_{k,nt}}(m)$ is needed. This is found to be

$$\begin{aligned}
r_{\tilde{e}_{k,nt}}(m) &= \mathrm{E}\left\{\tilde{e}_{nt}(k)\tilde{e}_{nt}^*(k+m)\right\} \\
&= \mathrm{E}\left\{\left(\sum_{nr_1=1}^{N_r} \tilde{W}_{nt,nr_1,k}\, \tilde{q}_{nr_1,k}\right)\left(\sum_{nr_2=1}^{N_r} \tilde{W}_{nt,nr_2,(k+m)}^*\, \tilde{q}_{nr_2,(k+m)}^*\right)\right\} \\
&= \sum_{nr_1=1}^{N_r}\sum_{nr_2=1}^{N_r}\left(\tilde{W}_{nt,nr_1,k}\, \tilde{W}_{nt,nr_2,(k+m)}^*\, \mathrm{E}\left\{\tilde{q}_{nr_1,k}\, \tilde{q}_{nr_2,(k+m)}^*\right\}\right) \qquad (3.23) \\
&\approx \sum_{nr=1}^{N_r} \left|\tilde{W}_{nt,nr,k}\right|^2 r_{\tilde{q}_{nr}}(m). \qquad (3.24)
\end{aligned}$$

The cross correlation term in Equation (3.23) considers correlation between received signals. In the case where $nr_1 = nr_2$ the positive values constructively add. When $nr_1 \neq nr_2$ the complex values destructively add producing a good approximation in Equation (3.24). Finally, the PSD of the error as a function of subcarrier and transmitter is

$$\begin{aligned} S_{\tilde{e}_{k,nt}}(f) &= \sum_{m=-\infty}^{\infty} r_{\tilde{e}_{k,nt}}(m)\, e^{-j2\pi fm} \\ &\approx \sum_{m=-\infty}^{\infty} \sum_{nr=1}^{N_r} \left|\tilde{W}_{nt,nr,k}\right|^2 r_{\tilde{q}_{nr}}(m)\, e^{-j2\pi fm} \\ &= \sum_{nr=1}^{N_r} \left|\tilde{W}_{nt,nr,k}\right|^2 \sum_{m=-\infty}^{\infty} r_{\tilde{q}_{nr}}(m)\, e^{-j2\pi fm} \\ &= \sum_{nr=1}^{N_r} S_{\tilde{q}_{nr}} \left|\tilde{W}_{nt,nr,k}\right|^2 . \end{aligned} \tag{3.25}$$

$S_{\tilde{e}_{k,nt}}(f)$ characterizes the noise as a function of subcarrier at the receiver after equalization. The SINR per subcarrier is simply the ratio of signal power and $S_{\tilde{e}_{k,nt}}(f)$. The performance in terms of BER is related to SINR for 4-QAM by [24]:

$$P_b = Q\left(\sqrt{\frac{2E_b}{N_0}}\right) \tag{3.26}$$

where the SINR is the ratio of the energy per bit, E_b, and the noise power, N_0 [24]. Using the noise and interference derivation in Equation (3.25) the BER can be found per subcarrier. However, Equation (3.26) assumes that the noise in the system is Gaussian.

3.2 Simulations

In this section, the theoretical distribution of delays and MIMO SINR and BER are confirmed with MATLAB simulations.

3.2.1 Distribution of Delays.

MATLAB is used to simulate the arrival times of N_t transmitters in the topology described in Section 2.3.1. Table 3.1 shows the parameters used for the simulations. In the following sections these parameters are used to simulate the distribution of arrival times for the Uniform and Gaussian distributed transmitters. Section 3.1.1.4 demonstrated

that dismissing the assumption of the small angle approximation is not mathematically tractable. For this, two propagation distances are considered. The first, d_F, is used when the small angle approximation applies. The distribution of arrival times is compared to the system that has the receivers d_C meters from the center of the ROA. This issue is addressed via simulation and conclusions about the differences in distributions are discussed.

Table 3.1: Simulation parameters for Figs. 3.4-3.7

Variable	Description	Value
R	Radius of ROA	484.8m
N_t	Number of transmitters	$10,000$
σ_R^2	Var. of x (Normal Distro)	$\left(\frac{R}{5}\right)^2$
θ_{step}	Increment of θ investigated	$\frac{\pi}{8}$
d_F	Far Rx distance from (0,0)	$R \times 10^4$
d_C	Close Rx distance from (0,0)	$R + 0.1R$

3.2.1.1 Uniform Distributed Transmitters.

Figure 3.4 shows the theoretical and simulated distribution of relative arrival times. The graph shows the distributions for $\theta = \{22.5,\ 78.75,\ 180\}$. The simulation of relative arrival times confirms the analysis in Section 3.1.1.2.

3.2.1.2 Gaussian Distributed Transmitters.

Figure 3.5 shows the distribution of Δ as a function of θ similar to the Uniform case. Here, the transmitters are distributed normally, with zero mean and variance σ_R^2.

3.2.1.3 Close vs. Far Receiver and Eavesdropper.

In Section 3.1.1.4 the distribution of the relative arrival times when the receivers are close enough to violate the small angle approximation assumption was shown to be

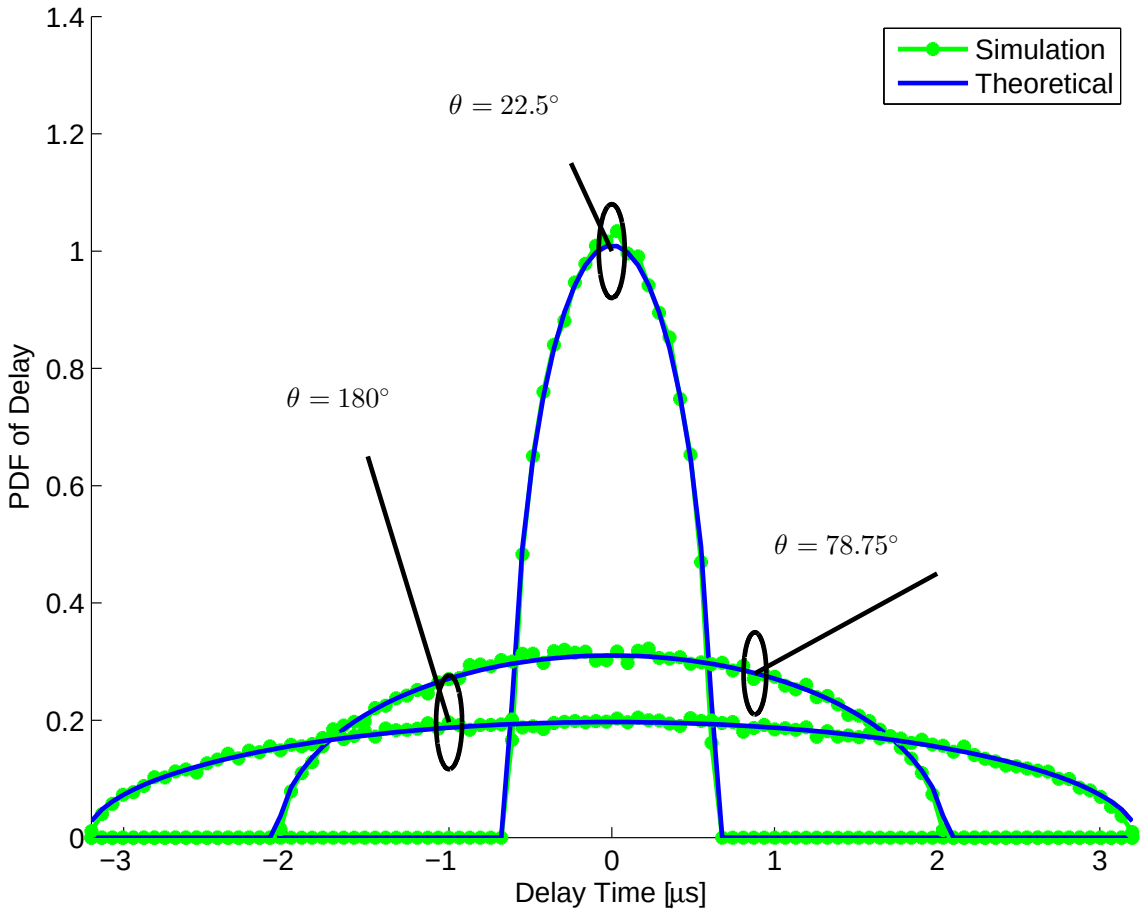


Figure 3.4: Delay for Uniform transmitter locations with the Ex d_F meters from the center of the ROA.

intractable. This section provides results through simulation, for both the Uniform and Gaussian distributed transmitters.

Figure 3.6 shows this effect for the Uniformly distributed transmitters. The *Simulation* line is compared to the *Theoretical* line from Figure 3.4 which depicts the delay distribution under the *far* assumption. A slight disagreement is more pronounced as θ approaches 180°. However, when the transmitters follow a Gaussian distribution, the small angle assumption does not play as large of a role. This is because the transmitters further away from the origin

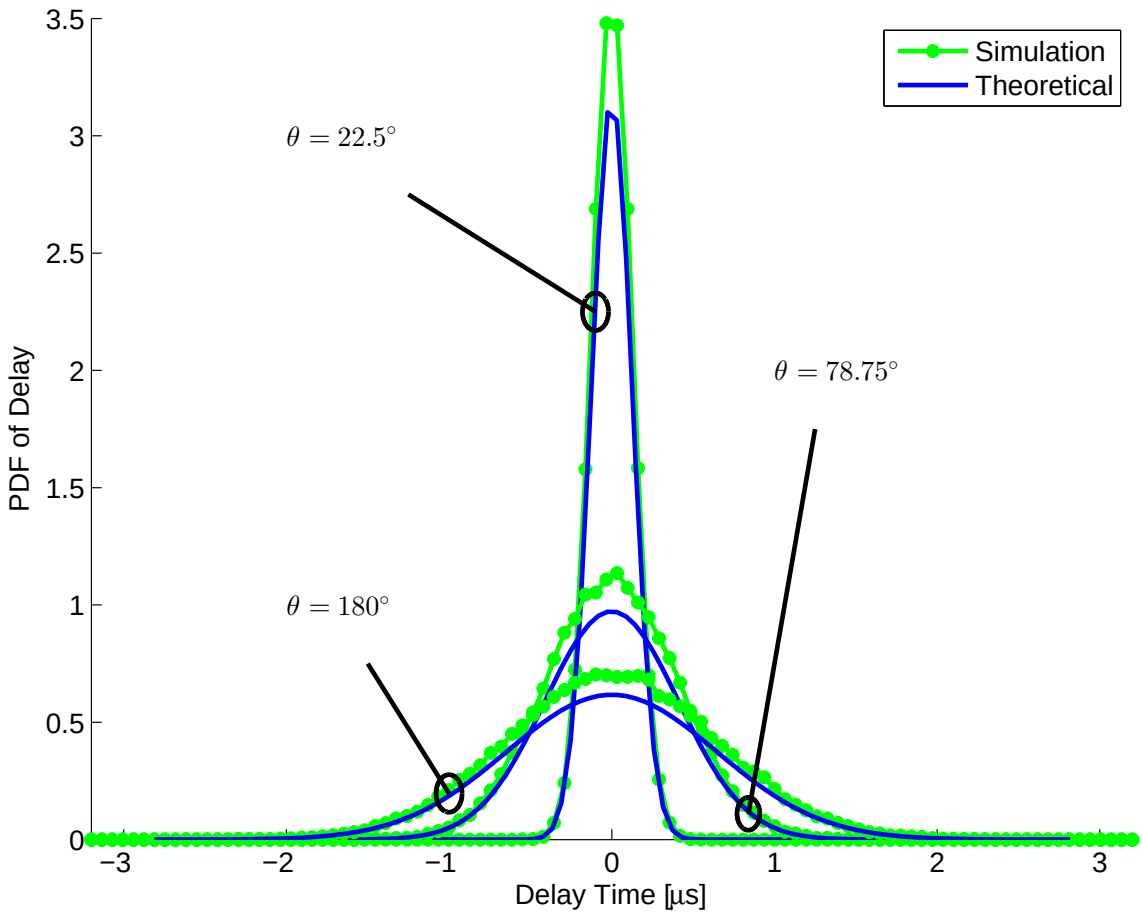


Figure 3.5: Delay for Gaussian transmitter locations with the Ex d_F meters from the center of the ROA.

of the ROA play the largest role in maximum delay. For the Gaussian distributed locations, there is a lower chance to have a transmitter further away from the origin compared to the Uniform case. This is a function of the standard deviation which is set to $\sigma_R = \frac{R}{5}$.

3.2.2 SINR and BER Simulation Setup.

In an effort to characterize the performance of a MIMO receiver under effects of IBI the key parameters are investigated in this section. The first of these is transmitter location in the ROA. Transmitters near the edge of the circle induce greater delays than those

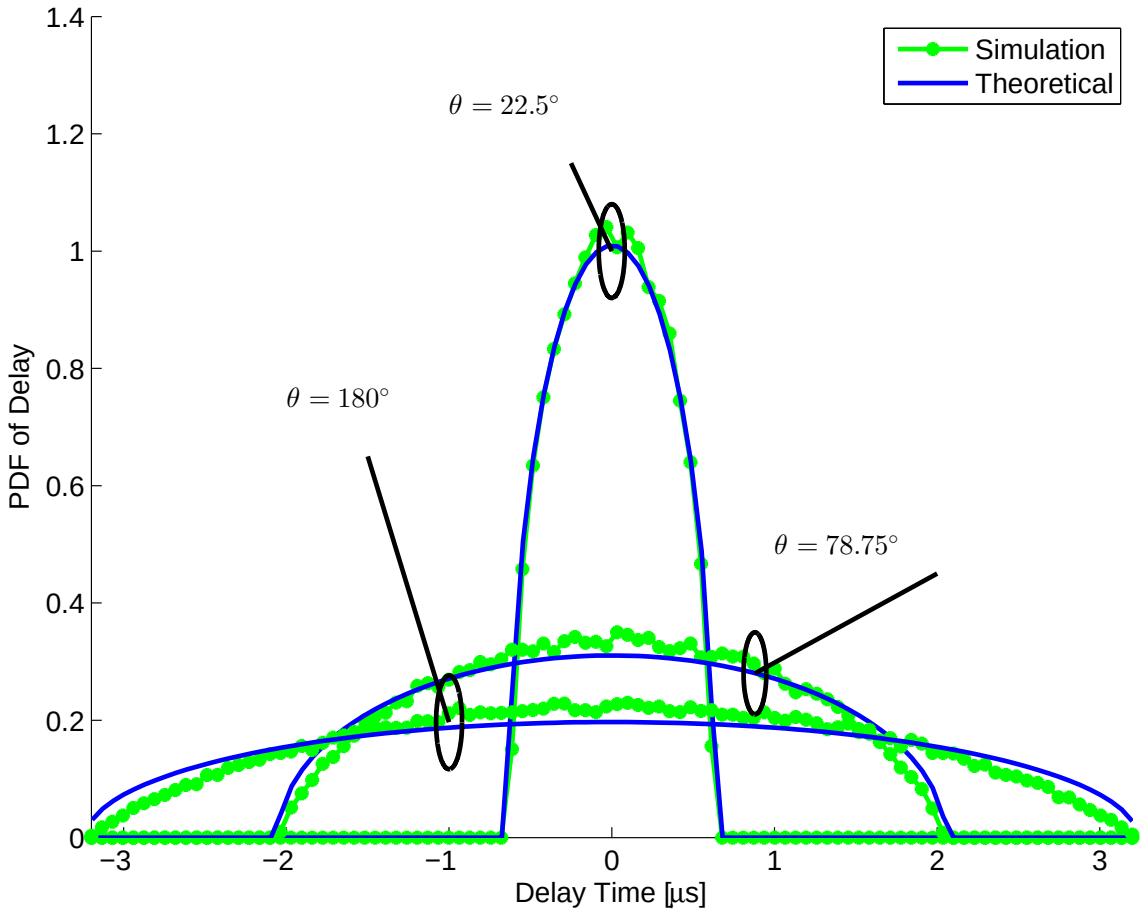


Figure 3.6: Delay for Uniform transmitter locations with the Ex d_C meters from the center of the ROA.

transmitters located near the middle of the circle. This leads to the second parameter, the size of the ROA. The larger the ROA the larger the delays in the system. The third parameter is the length of the impulse response at $\theta = 0°$. Since the cooperative system operates in this direction the channel is not lengthened by delay. The cooperative system would be designed to experience no IBI on average, where $N_{cp} \geq L - 1$ would hold, but would want to have IBI for a system operating at $\theta \neq 0°$. If $N_{cp} = L - 1$ then the delay experienced at $\theta \neq 0$ would induce IBI. The last two parameters are the number of transmitters and angle at which the receiver is located. As the number of transmitters

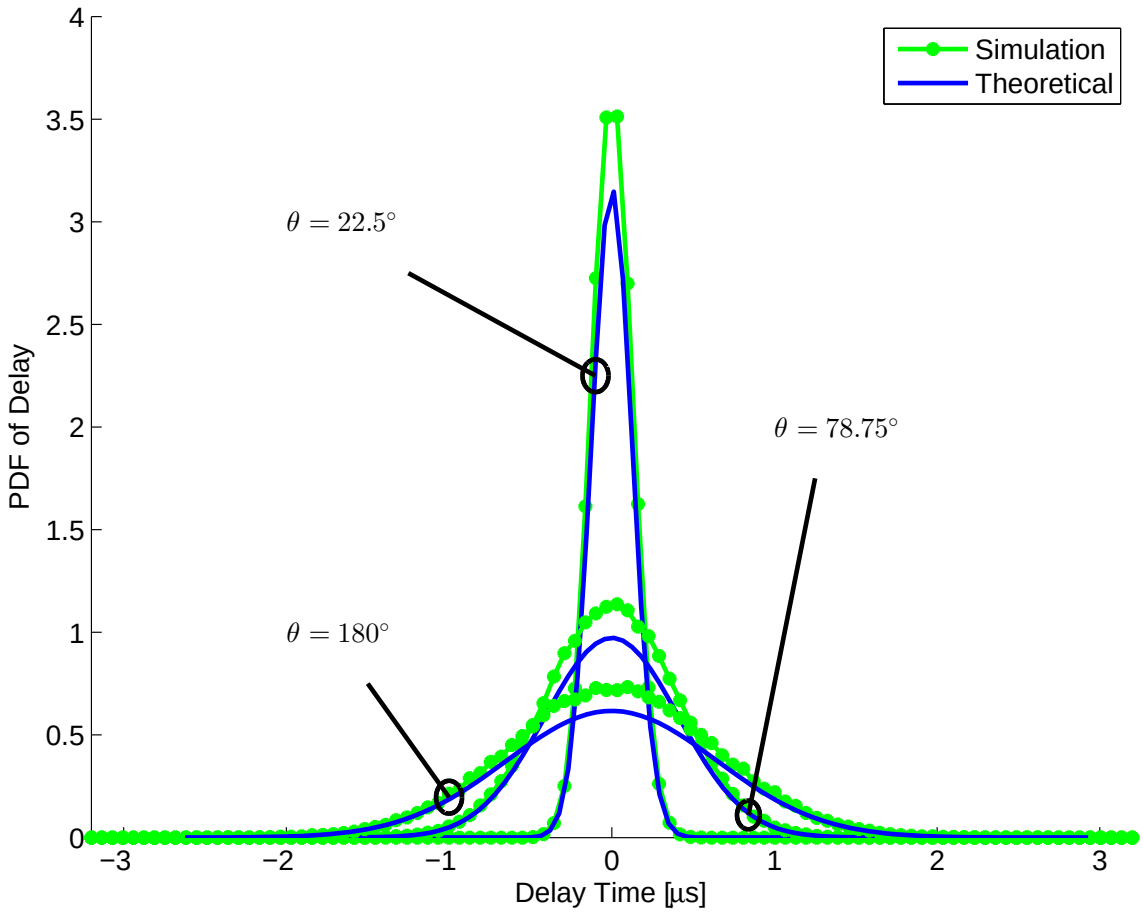


Figure 3.7: Delay for Gaussian transmitter locations with the Ex d_C meters from the center of the ROA.

increases the number of channels increases which provides more opportunities for IBI. Finally, as θ increases to 180° the system experiences larger differences in delay.

Table 3.2 shows the key parameters that influence the amount of IBI a receiver experiences. The channel length and CP length are set such that one sample of delay induces IBI. The number of transmitters is chosen to be seven with one transmitter placed at the origin of the circle and the other six transmitters are set equally spaced from each other and at the radius of the ROA. This provides a worst case picture of the trends of IBI

Table 3.2: Simulation parameters for Figs. 3.8-3.13

Variable	Description	Value
R_1	Radius of ROA	2420 m
R_2	Radius of ROA	12.1 km
L	Channel Length	17
N_{cp}	Cyclic Prefix Length	16
N_t	Number of Transmitters	7
N_r	Number of Receivers	7
θ	Angle for a Receiver	0°-180°
N_{pil}	Num. pilots used for CE	19

without having to average over many transmitter locations. The range for θ considered is $0° - 180°$ since $180° - 360°$ is redundant.

3.2.3 SINR per Subcarrier.

The signal power of a 4-QAM symbol with constellation of $\{\pm 1 \pm i\}$ is $P_x = 2$. Equation (3.25) represents the error in the system. The theoretical SINR is the ratio of $\frac{P_x}{S_{\tilde{e}_{k,nt}}(f)}$ which is considered as a function of frequency in Figure 3.8. The theoretical SINR is compared to MATLAB simulations where the equalizing CSI is obtained by the known CSI, FDCE and TDCE.

The error power after equalization is calculated by finding the variance of the subtraction of the known transmitted symbols from the received signal.

3.2.4 SINR Performance.

To determine how IBI reduces expected performance of a receiver at some angle θ, the average SINR across subcarriers is used as an intermediate metric to reach BER performance. Average SINR as a function of θ is provided in Figs. 3.9 and 3.10. Also

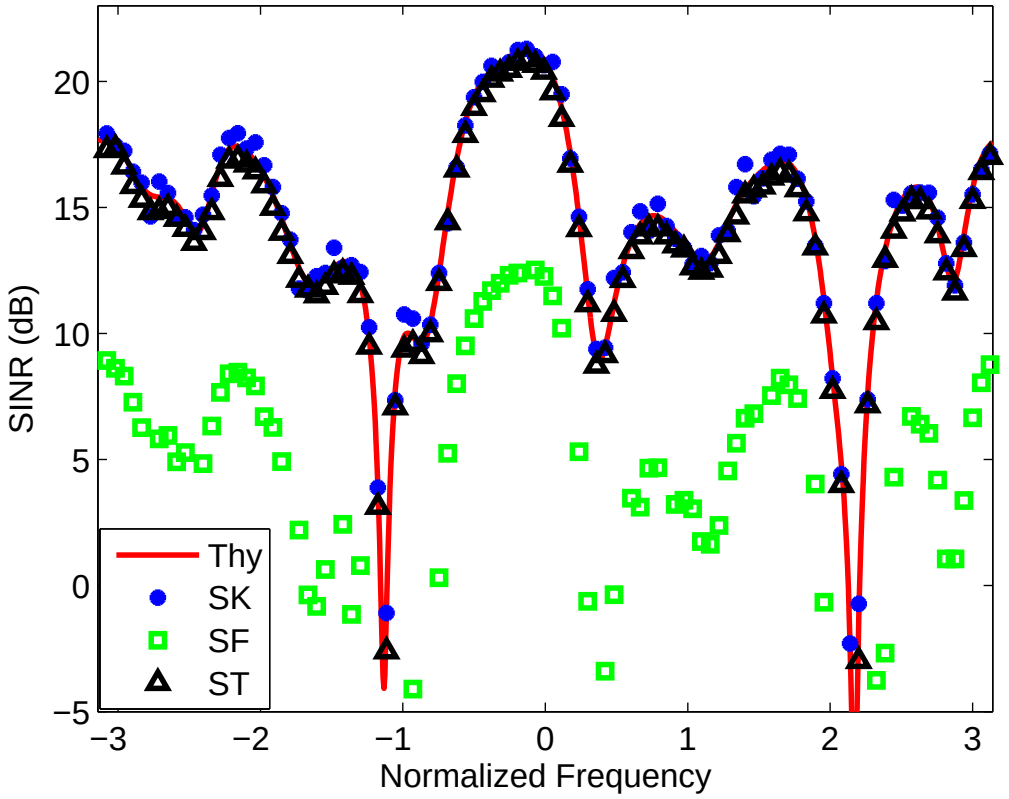


Figure 3.8: At an SNR of 20 dB and $\theta = 0$, the theoretical (Thy) SINR, as function of frequency, is compared to simulated SINR values where the channel is equalized with the known CSI (SK), FDCE (SF) and TDCE (ST).

shown, is the trend as R varies, as R increases, larger delays are experienced which results in lower SINR.

Figure 3.9 shows the performance of FDCE as a function of θ and R. Without the perfect knowledge of the CSI the FDCE performs significantly worse than the theoretical performance. This degradation in performance is a result of phase rotations in the frequency domain that cannot be equalized because accurate CE cannot be obtained due to the violation of $N_{cp} \geq L - 1$.

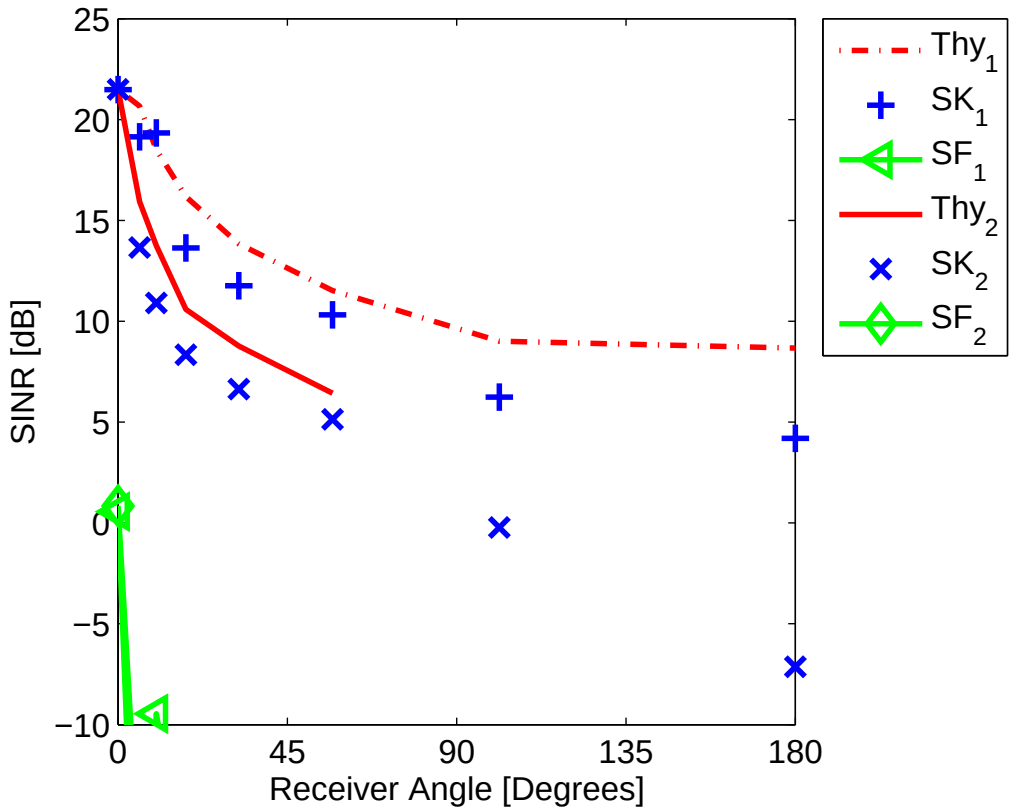


Figure 3.9: The SINR as a function of θ and R at an SNR of 20 dB. The FDCE (SF_i) performance is compared to the simulated system with known CSI (SK_i) as well as the theoretical (Thy_i) performance. Subscripts correspond to R_i in Table 3.2.

However, TDCE does have the ability to correct for delays in the time domain at the cost of computational complexity. Figure 3.10 shows the performance of the TDCE is about a 5 dB improvement at $\theta \neq 0$.

In Figs. 3.9 and 3.10 at $\theta = 90°$ there is a performance gap between the theoretical curve and simulated system with known CSI. This effect is noticed when $\theta > 0$ but is more pronounced as θ increases. To show the cause of this effect Figure 3.11 shows the SINR performance as a function of subcarrier similar to Figure 3.8 where $\theta = 0°$, here $\theta = 90°$. It is clear from Figure 3.11 that all three simulated schemes have compromised

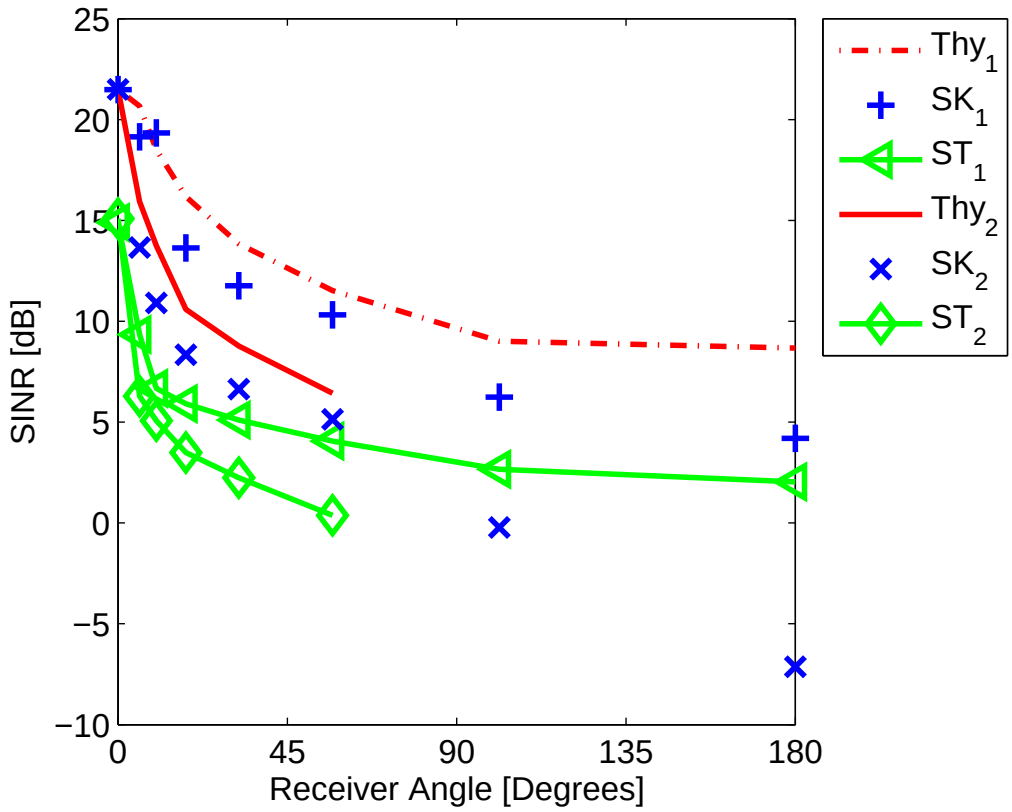


Figure 3.10: The SINR as a function of θ and R at an SNR of 20 dB. The TDCE (ST_i) performance is compared to the simulated system with known CSI (SK_i) as well as the theoretical (Thy_i) performance. Subscripts correspond to R_i in Table 3.2.

accuracy compared to Figure 3.8 with the worst being the FDCE. At $\theta = 90°$ there is a significant amount of IBI in the received signal. Even when the CSI is known at the receiver the performance is degraded due to this IBI. In this case, the signal is fundamentally compromised and cannot be equalized with the known channel. Then with TDCE the performance is degraded even further by estimation error.

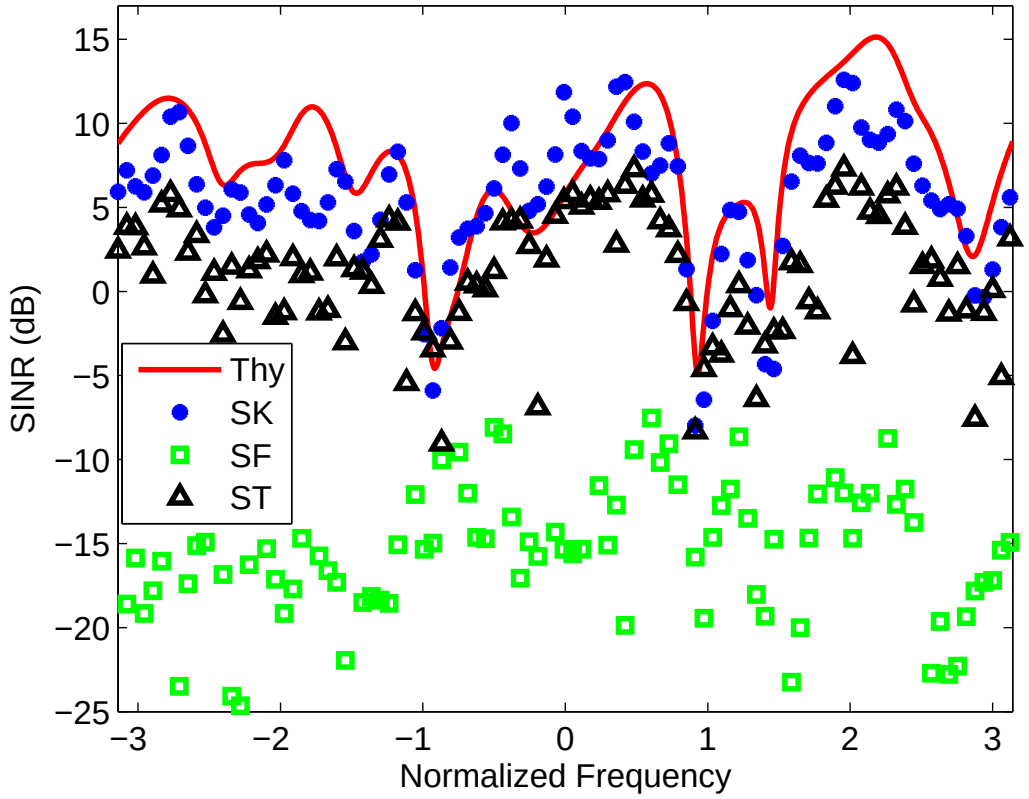


Figure 3.11: At an SNR of 20 dB and $\theta = 90°$, the theoretical (Thy) SINR, as function of frequency, is compared to simulated SINR values where the channel is equalized with the known CSI (SK), FDCE (SF) and TDCE (ST).

3.2.5 BER Performance.

The theoretical BER calculated from the SINR in Figs. 3.9 and 3.10 is shown in Figs. 3.12 and 3.13. The theoretical curves are calculated by converting the SINR at each subcarrier to BER, by Equation (3.26), which assumes the noise is Gaussian. This assumption is what accounts for the slight disagreement between the theory and known CSI curves. The BER is then averaged over subcarrier and transmitter for a particular value of θ. The result is a BER performance for the receiver that considers all the transmitted data. At a $\theta > 50°$ for R_2 the delay between transmitted waveforms is so large that the

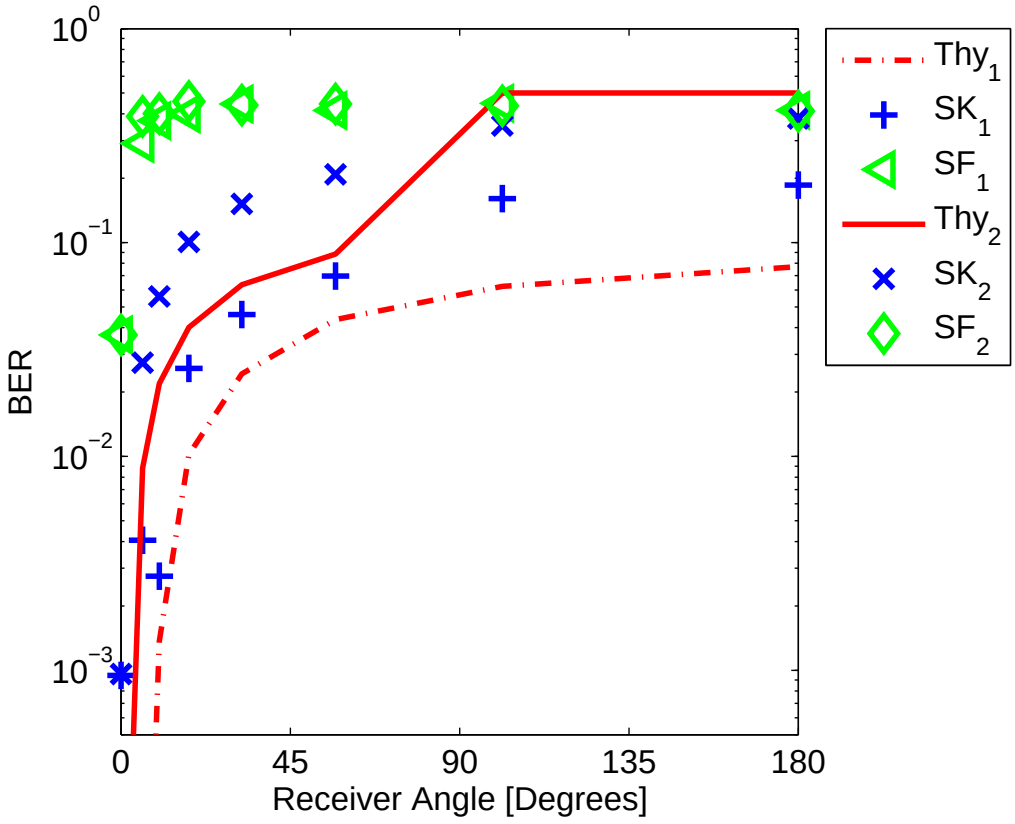


Figure 3.12: BER as a function of θ and R at an SNR of 20 dB. The FDCE (SF_i) performance is compared to the simulated system with known CSI (SK_i) as well as the theoretical (Thy_i) performance. Subscripts correspond to R_i in Table 3.2.

communication link is unusable. This is represented in Figs. 3.12 and 3.13 by the ramp up to $BER = 0.5$.

3.3 Conclusions

In multi-carrier communication systems, IBI occurs if the CP is shorter than the impulse response. In MIMO-OFDM systems each impulse response could contribute to the total IBI power. Generally, all channel lengths are the same in MIMO communication

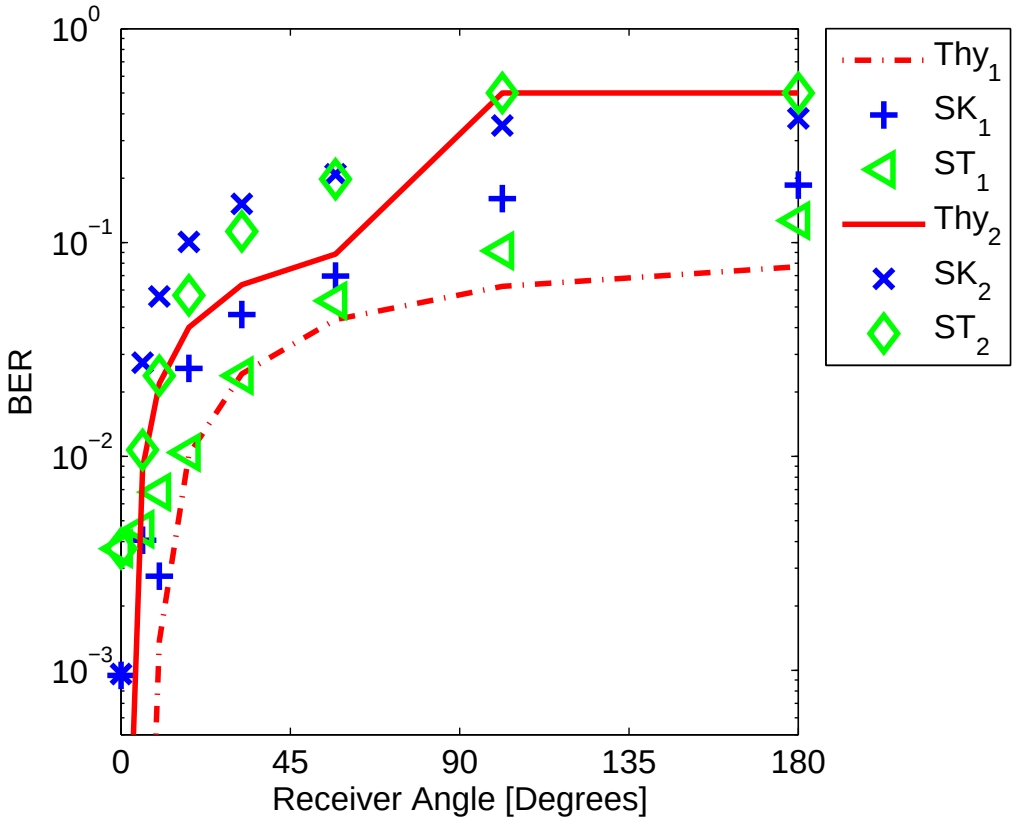


Figure 3.13: BER as a function of θ and R at an SNR of 20 dB. The TDCE (ST_i) performance is compared to the simulated system with known CSI (SK_i) as well as the theoretical (Thy_i) performance. Subscripts correspond to R_i in Table 3.2.

systems. However, topological delays induce longer impulse responses for distant transmitters with respect to close transmitters. Derived in this chapter is the theoretical SINR for such a system with a fixed set of delays. The expected performance in terms of BER is shown to agree with simulations of the system.

In a cooperative system design the $\theta = 0$ direction is considered. The FDCE and TDCE are favorable CE techniques for Rx. With these designs the systems perform favorably with poor performance for the Ex, making it less likely to demodulate data not intended for it, making these processes a viable physical layer security technique. For

example, even with a known channel, Ex at 35° experiences a 10 – 15 dB drop in SINR and a 10x increase in BER for our simulation scenario.

IV. Implementation of Multi-User NC-OFDM TOA Estimation Algorithms

The performance of an Ex is improved if the delays between the N_t users are known making the TDCE usable. This chapter discusses estimating these delays between users using an OFDMA waveform. Two algorithms are considered. The first algorithm assumes each user occupies all N subcarriers. Since this is not the case in OFDMA the performance of this estimate is degraded compared to the second algorithm. In the second algorithm the correlation induced by a user using a subset of the available subcarriers is leveraged to determine the delay between the N_t users and assign the delays to each user. Simulation results for the two algorithms are provided. A small scale hardware implementation is then outlined and results are provided. This chapter ends with a discussion of a large scale implementation that is set up for future work with the goal of fixing synchronization concerns between all receivers.

4.1 Estimators

If the number of OFDM symbols, $N_b > 1$, then $\mathbf{y}_{nb}$ represents the nb^{th} received OFDM symbol, in which case all N_b OFDM symbols are used to estimate the delay. The Appendix derives the estimator under this condition and the result is provided here [26]:

$$\hat{d} = \underset{d}{\operatorname{argmin}} \left(N_b \log |\mathbf{C_d}| + \sum_{nb=1}^{N_b} \frac{1}{2} \mathbf{y}_{nb}^H \mathbf{C}_d^{-1} \mathbf{y}_{nb} \right) \tag{4.1}$$

where $\mathbf{C}_d$ is defined in Equation (2.27). Equation (2.22) is the cost function of the *van de Beek* method which is repeated here [27]

$$\hat{\bar{d}} = \underset{\bar{d}}{\operatorname{argmax}} \left\{ \left| \gamma(\bar{d}) \right| - \rho \Phi(\bar{d}) \right\} \tag{4.2}$$

where

$$\gamma(s) = \sum_{m=s}^{s+N_{cp}-1} y(m) y^*(m+N), \tag{4.3}$$

the term $\rho\Phi\left(\bar{d}\right)$ is assumed to be zero in these simulations and experiments. In the next three sections, the *van de Beek* and *Acharya* methods are compared in MATLAB simulations and in an implementation using the WARP boards.

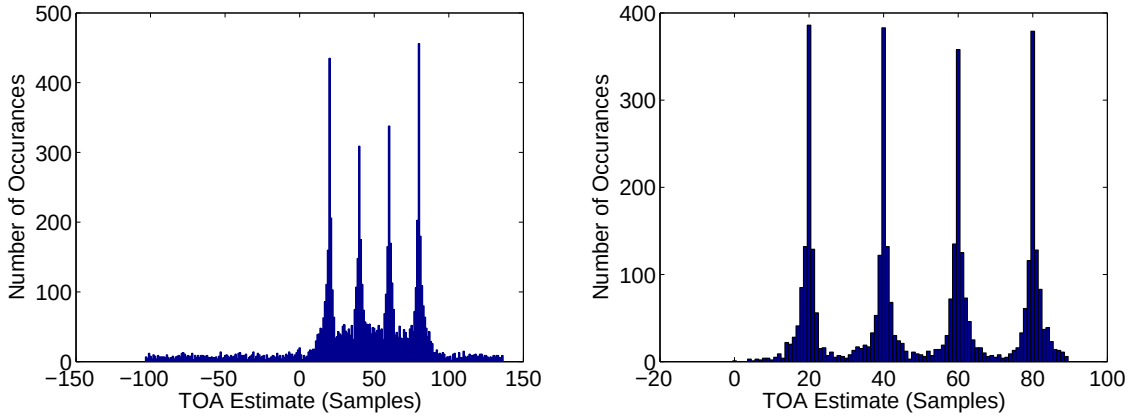
4.2 Simulation

Table 4.1: Simulation parameters for Figs. 4.1a-4.1b

Variable	Description	Value
N	Number of Subcarriers	256
N_b	Number of OFDM Blocks	1
N_{cp}	CP Length	16
N_t	Number of users	4
N_{mc}	Number of Monte-Carlo Trials	1,000
N'	OFDMA Block Length	272
$\mathbf{d}$	Propagation Delay	[20 40 60 80]
SNR	Signal to Noise Ratio	5 dB

This section demonstrates the accuracy of the *van de Beek* method [27] and *Acharya* method [26] via MATLAB simulation. The parameters for the simulations are provided in Table 4.1. The peaks in Figure 4.1a correspond to $\hat{\mathbf{d}}$. The estimates are in the range of -136 to 136 which correspond to $\pm 0.5N'$.

Each of the $N_t = 4$ users are allocated subcarriers in which to transmit data on. The subcarrier assignment scheme used is the same as in Figure 3 in [26] where each user is allotted a fourth of the subcarriers in four equally spaced dis-contiguous sub-bands. As [26] states, under dynamic allocation of subcarriers this scheme is a possible scenario.



(a) *van de Beek* method

(b) *Acharya* method

Figure 4.1: MATLAB simulation of the TOA estimation algorithms with system parameters provided in Table 4.1.

Figure 4.1b shows the performance of Equation (4.1) which assumes knowledge of which subcarriers are used by each user. In this work it is assumed that the subcarrier assignments are known at Ex. With the knowledge of the subcarriers used by each user the TOA is determined for each user. The accuracy is improved with the *Acharya* method because the correlation in the time domain is modeled in the estimator.

4.3 Small-Scale Hardware Implementation

In this section the implementation of two TOA algorithms is compared in terms of accuracy and execution speed. First, the WARP board test bed used in this experiment is outlined. Then the *van de Beek* method is implemented and studied with multiple OFDMA symbols used at the receiver to estimate the TOA. Finally, the *Acharya* method is also implemented on the same platform and its accuracy and computational complexity are compared.

4.3.1 WARP Board Test-bed.

Recall that the delay experienced at the receiver is made up of two components $\phi_{nt} = \tau_{Rx} - \tau_{nt}$ where τ_{Rx} denotes the local clock at the receiver and τ_{nt} is the local clock at the transmitter. $\mathbf{d}_{nt}$ is then the propagation delay induced by the path length. The goal in this TOA study is to estimate $\mathbf{d}_{nt}$, however with ϕ_{nt} playing a role in the true timing estimate a repeatable experiment is not easily attainable.

For the test-bed to have a repeatable experiment a single WARP board is used with four transmit antennas. Each antenna is considered one user. In this scenario the four users have the same local clock making ϕ_{nt} constant across n_t henceforth denoted ϕ. The receiver estimates the $N_t = 4$ user TOA values. The relative delay between each user is representative of $\mathbf{d}_{nt}$ with an overall delay corresponding to ϕ.

The transmitter and receiver WARP boards are controlled by MATLAB via an Ethernet switch. The Ethernet cables between the controller computer and the two WARP boards are used for rough synchronization. At the sampling frequency of the WARP boards, $f_s = 40$ MHz, the Ethernet synchronization is reliable to ± 100 samples in the received signal.

The Ethernet cable length constricts the physical size of the test-bed; Ethernet cables are 100 feet in length. For this reason, the delay between transmitted waveforms is artificially added by prepending zeros to the transmitted waveform in MATLAB before transmission. The number of zeros prepended corresponds to $\mathbf{d}_{nt}$ and is what is to be estimated by the *van de Beek* and *Acharya* methods. The experiment parameters used are provided in Table 4.2.

4.3.2 van de Beek Method.

Figure 4.2 shows a histogram of $\hat{\tilde{\mathbf{d}}}$ where the peaks at $[26,\ 156,\ 70,\ 114]$ correspond to $\mathbf{d} + \phi$ where $\phi = 20$. Figure 4.2 also shows the estimator accuracy as a function of the number of OFDM symbols used for the estimate. The cost function, $\gamma(s)$, associated with

Table 4.2: Experiment parameters for Figs. 4.2-4.3

Variable	Description	Value
N	Number of Subcarriers	64
N_b	Number of OFDM Blocks	6
N_{cp}	CP Length	16
N_{mc}	Number of Monte-Carlo Trials	2,000
$\mathbf{d}$	Propagation Delay	[6.25, 135.625, 50, 93.75]

each OFDM symbol is averaged then the maxima are found. As the number of OFDM symbols increases the effect of noise is reduced due to the averaging which results in more reliable estimates. $N_b = 6$ OFDM symbols are used in Figure 4.2

An issue with this method is that there is not a way to discern users. The values of $\gamma(s)$ do not relay this information. This issue is problematic in estimating the timing delays for the Ex to equalize the users in the ROA. Next, the *Acharya Method* does allow each user's TOA to be calculated.

4.3.3 Acharya Method.

The *Acharya* method assumes the subcarriers used by each user are known. This information is captured in the construction of $\mathbf{P}_{nt}$ as a function of nt. Each user is allocated subcarriers for which to transmit their data. The subcarriers that are not used by a specific user are zeroed, which introduces correlation between time domain samples that degrades the *van de Beek* method's performance.

The *Acharya* method is more accurate compared to the *van de Beek* method. The Root-Mean Squared Error (RMSE) for the *Acharya* method is 20.08 and for the *van de Beek* method the RMSE is 38.71. This is shown in Figure 4.3 where each user is separated out and an estimate histogram is presented for $N_{mc} = 2000$ monte-carlo trials.

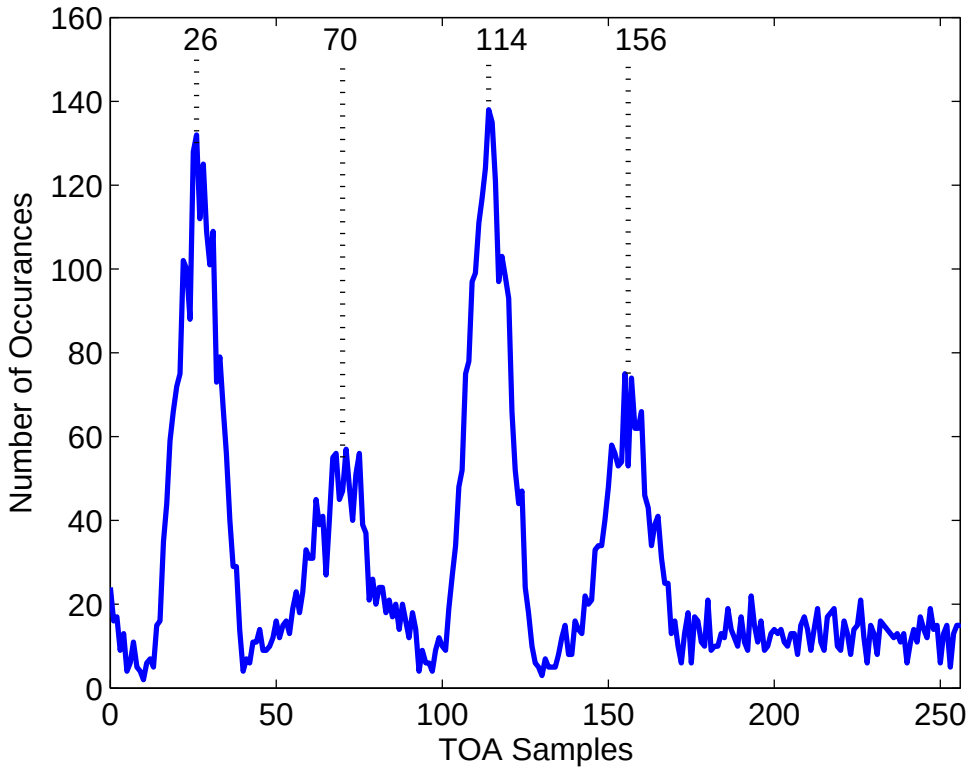
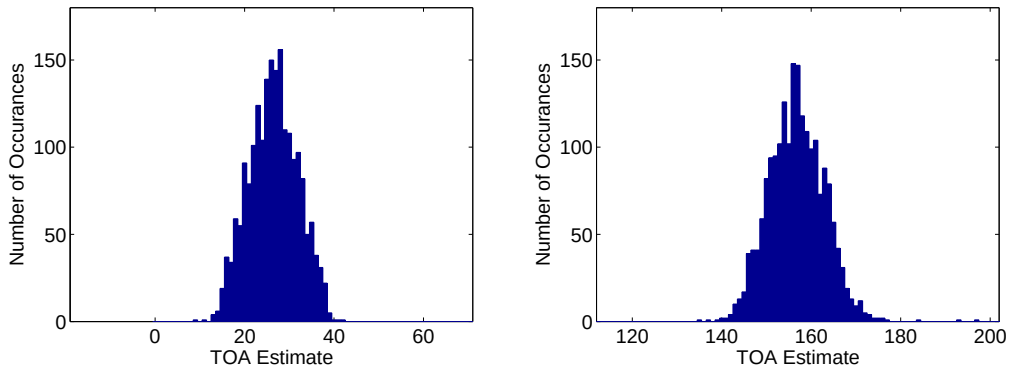


Figure 4.2: Histogram of TOA estimates for the *van de Beek* method. Peaks correspond to the four users in the system. The number of OFDM symbols used in the averaging of the cost function are also plotted. As the number of OFDM blocks increase a more accurate estimate is obtained. The RMSE for each user is 38.71.

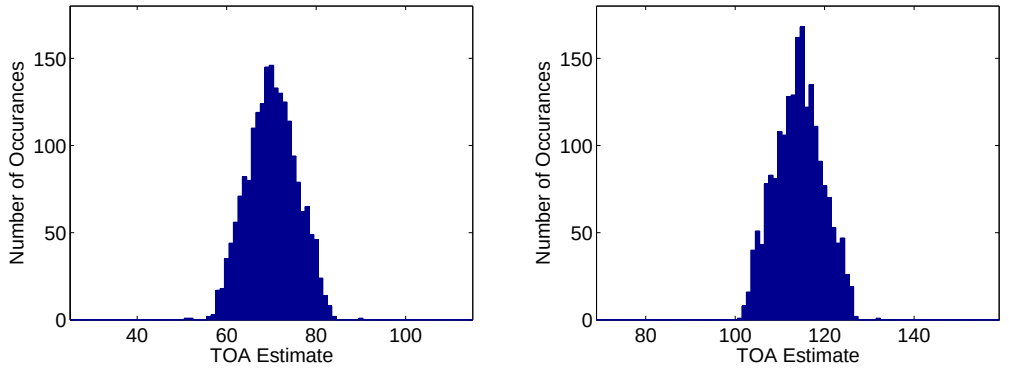
4.4 Large-Scale Hardware Implementation

Thus far, the application for the TOA estimates were to determine the delays between the received waveforms for Ex to improve its fidelity. A small extension of this concept is TDOA position estimation. However, a large issue with TDOA is synchronization between the N_r receive antennas. This is needed for a time of reference from which the time difference is taken.



(a) User 1 TOA estimate histogram

(b) User 2 TOA estimate histogram



(c) User 3 TOA estimate histogram

(d) User 4 TOA estimate histogram

Figure 4.3: TOA estimates according to the *Acharya* method for each user. The RMSE for each user is 20.08.

The small-scale implementation solved this issue by having all the users synchronized using the one Local Oscillator (LO) on the WARP board. The WARP board itself is designed for work in MIMO communications where the antennas are synchronized. This platform lends itself to TDOA estimation. However, in a large scale implementation the receivers need to be synchronized over hundreds of feet.

To synchronize the $N_r = 4$ receivers in our testbed a *reference transmitter* (denoted Tx_{ref}) is used. The position of this transmitter is known along with the positions of each of

the receivers as depicted in Figure 4.4. The delays between Tx_{ref} to each receiver is known denoted $\Delta_r = [\Delta_{r,1}\ \Delta_{r,2} \ldots \Delta_{r,N_r}]^T$. Each transmitter has a local time associated with its LO the local time is denoted t_{ref} for Tx_{ref} and t_{tx} for the transmitter to be located Tx_{ukn}. The local time for each receiver is denoted by τ_{nr} for $nr \in \{1,\ 2, \ldots,\ N_r\}$.

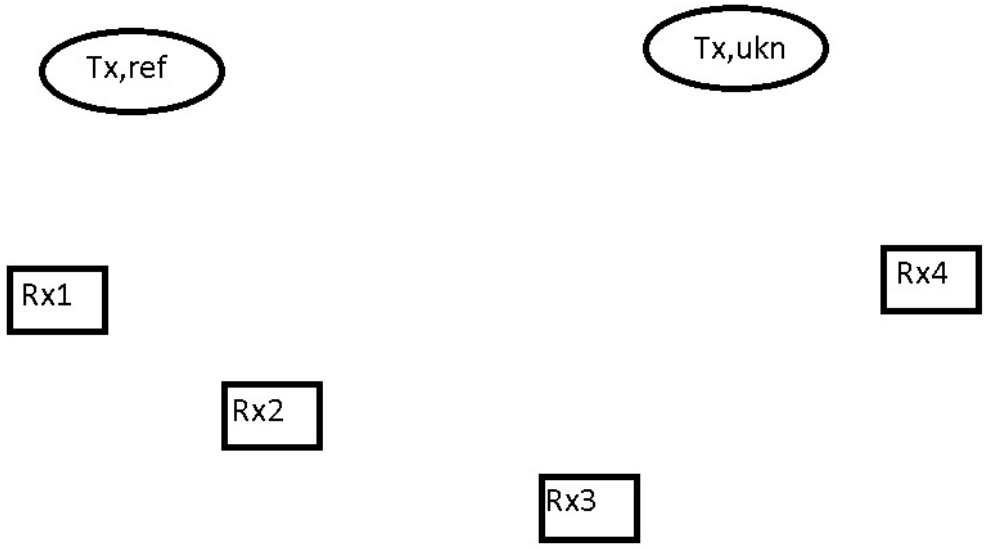


Figure 4.4: Illustration of the four receivers used to determine TDOA measurements for the unknown transmitter, Tx_{ukn}. Also depicted is the reference transmitter, Tx_{ukn}, used to synchronize the four receivers.

At the nr^{th} receiver the TOA is found for both the reference transmitter, $t_{nr,ref}$, and the unknown transmitter, t_{nr}, which are assumed to be operating in two different frequency bands. The TOAs for both transmitters are modeled as

$$t_{nr} = t_{tx} + \Delta_{u,nr} + \tau_{nr},$$

$$t_{nr,ref} = t_{ref} + \Delta_{r,nr} + \tau_{nr},$$

where $\Delta_{u,nr}$ denotes the unknown propagation delay from the unknown transmitter to the nr^{th} receiver.

To determine the TDOA between the nr_1^{th} and nr_2^{th} receivers consider the following equations:

$$\begin{aligned} t_{nr_1} &= t_{tx} + \Delta_{u,nr_1} + \tau_{nr_1}, \\ t_{nr_2} &= t_{tx} + \Delta_{u,nr_2} + \tau_{nr_2}, \\ t_{nr_1,ref} &= t_{ref} + \Delta_{r,nr_1} + \tau_{nr_1}, \\ t_{nr_2,ref} &= t_{ref} + \Delta_{r,nr_2} + \tau_{nr_2}. \end{aligned} \tag{4.4}$$

The values $\hat{t}_{nr_1}$, $\hat{t}_{nr_2}$, $\hat{t}_{nr_1,ref}$ and $\hat{t}_{nr_2,ref}$ are estimated at each receiver and are provided to a central processing center for TDOA estimation. At the processing center, $\mathbf{m} = [m_1, m_2, \ldots, m_{N_r}]^T$ is defined where

$$\begin{aligned} m_{nr} &= \hat{t}_{nr} - \hat{t}_{nr,ref} + \Delta_{r,nr} \\ &= t_{tx} - t_{ref} + \Delta_{u,nr} + e, \\ \text{TDOA}_{nr_1,nr_2} &= m_{nr_1} - m_{nr_2} \\ &= \Delta_{u,nr_1} - \Delta_{u,nr_2} + e. \end{aligned} \tag{4.5}$$

To test the accuracy of this method the WARP boards where set up in a hallway. The hallway itself is 150 feet long. The four receivers are set equally spaced in the hallway. Figure 4.5 illustrates how the receivers are spaced in the hallway. The unknown transmitter and reference transmitter are co-located at one end of the hall.

This test is then ran to ensure that the desired TDOA values are obtained. First the expected values are calculated. The speed of light $c = 9.8 \times 10^8$ ft/s and the sampling frequency $f_s = 40$ MHz are needed. If the path length of a transmitted signal to two WARP boards differs by $c/f_s = 24.5$ ft the resulting TDOA between those two WARP boards is 1

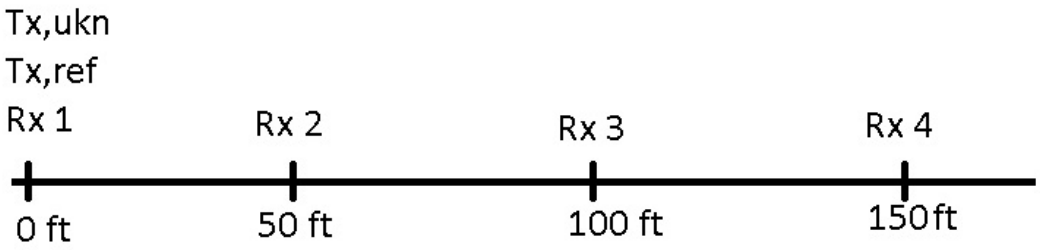


Figure 4.5: Reference transmitter, unknown transmitter and the four receiver locations in a *hallway* to verify the TDOA measurement accuracy.

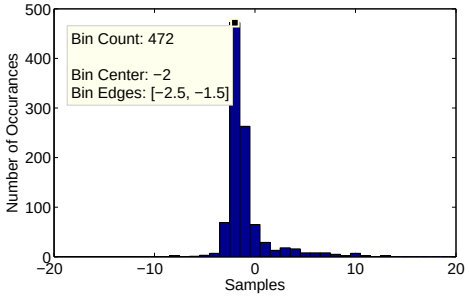
sample. For the test depicted in Figure 4.5 each WARP board has a two sample difference from an adjacent receiver.

Figure 4.6 shows the TDOA estimates for 1000 trials. The *van de Beek* method is used where one user transmits on all subcarriers. In each case the expected number of samples ± 1 is estimated the majority of the time.

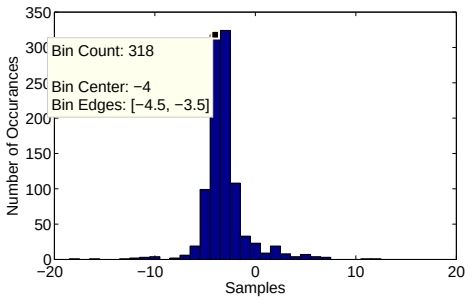
Each TDOA estimate is negative. The reason for this is that the *first* WARP board receiver is at the same location as the transmitters. The *first* receiver TOA estimate is used as the reference; from which the other TOA values are subtracted from in Equation (4.5). As a result, the TDOA values are negative because the signal is received after the *first* receiver.

The hallway test is designed to verify the use of Equation (4.4) to Equation (4.5). However, the test is not interesting in terms of locating the *unknown transmitter*. For more interesting topologies, as shown in Figure 4.4, the physical limitation of Ethernet length is addressed to provide a larger test-bed.

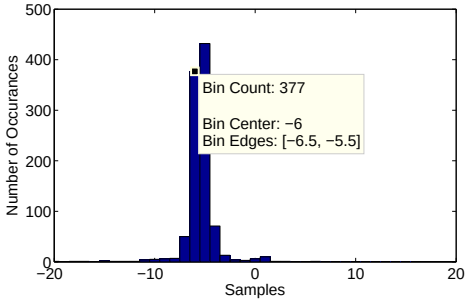
To obtain a larger test-bed, the wired connection between the Ethernet switch and each receive WARP board is converted to a wireless connection via an Ethernet bridge [41]. The wired connection speed is a Gigabit per second. The wireless communication



(a) TDOA $m_1 - m_2$ where the true delay is -2.



(b) TDOA $m_1 - m_3$ where the true delay is -4.



(c) TDOA $m_1 - m_4$ where the true delay is -6.

Figure 4.6: Hallway Test

connection does not have this capability. An Ethernet switch is used to convert between Gigabit Ethernet to 100 Mb/s connection speeds.

4.5 Conclusions

This chapter discussed the feasibility of obtaining TOA estimates with an application for an Ex to reduce interference and obtain a reliable estimate of the transmitted signal. The application of using the TOA estimates in TDOA position estimation is also considered.

The two methods for estimating the TOA, *van de Beek* and *Acharya*, were considered. Simulations of the methods showed positive results for both algorithms however the *Acharya* method is more accurate compared to the *van de Beek* method. This is a result

induced by the time domain correlation introduced when users occupy a subset of the available subcarriers.

Two implementations followed the simulation results. A small scale implementation showed the performance of the *van de Beek* and *Acharya* methods where the $N_t = 4$ users shared an LO. This small scale implementation did not attempt to solve the synchronization issue between receivers since only one receiver is used and the physical limitations of the WARP board test-bed were not an issue since the delays between users were artificially added to the transmitted signal.

Preliminary work on the large scale implementation showed progress in the synchronization between receivers with the use of a reference transmitter. The physical limitation of the WARP board implementation is also circumvented by the use of an Ethernet bridge which removed the wired connection between the receiving WARP board and the controlling computer.

V. FPGA Resource Utilization for MIMO-OFDM Receivers

Rx is designed in this chapter based on the results found in Section 3.2.4. The FDCE method is employed due to its added physical layer security attributes. However, in this chapter maximizing the data rate is used as the design criteria. By increases the size of the MIMO system data rate is increased but to implement the MIMO system the amount of resources increases. This chapter shows the trends of the resource usage as a function of the size of the MIMO system.

The goal of this chapter is not to optimally determine the correct balance between resource usage, latency and BER for the entire MIMO receiver design. The goal is to show how resource usage and data rate scale as a function of the size of the MIMO system deployed.

This chapter first breaks down the MIMO receiver into smaller calculation blocks. These blocks are VHDL modules that are analyzed in the design. For each block, the number of slice registers, slice Look-Up Tables (LUTs), and Digital Signal Processing (DSP) resource blocks are reported. With this information, some intuition is gained regarding the resource hungry blocks and some ways to reduce the resource usage are discussed.

The MIMO OFDM receiver is then designed for N_a transmit and N_a receive antennas with $N_a \in \{2,\ 3,\ 4\}$. A comparison between resource utilization shows how the blocks scale as a function of the number of antennas in the system. From here, extrapolation is used to draw a relation between desired throughput of a communication link and the number of resources required to implement the MIMO receiver with a target FPGA of the VX980T. Also considered are the VX1140T and VX2000T FPGAs. If the desired throughput is more ambitious than the largest FPGA can fit we discuss some architecture possibilities to alleviate the resource constraint. This requires designing a board with multiple FPGAs

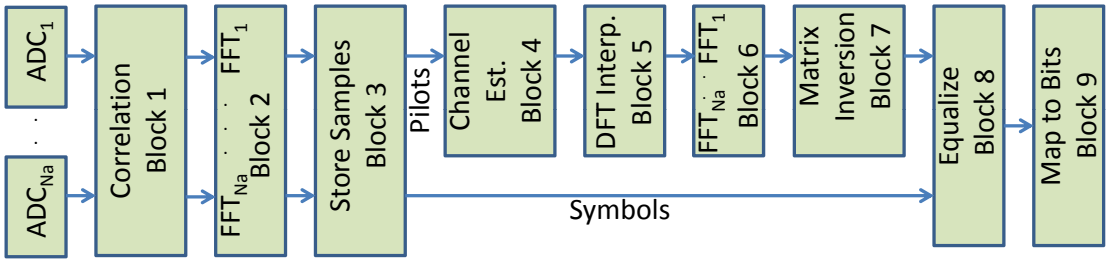


Figure 5.1: Block diagram of a MIMO receiver that utilizes frequency separated pilot tones.

and considering data rates between FPGAs to effectively double or triple available resource counts.

5.1 MIMO Receiver Architecture

The MIMO receiver is decomposed into eight calculation blocks. These blocks consider functions such as the FFT, matrix vector multiplication, matrix inversion and also a block that routes data and pilot samples to the correct locations. The nine blocks pictured in Figure 5.1 represent the full MIMO receiver where the FFT block is needed in two locations. The functional description of these blocks starts with the matched filter that synchronizes the receiver with the transmitted preamble.

5.1.1 Correlation.

The correlation process, *Block 1*, provides three features. The first is the matched filter that is used to synchronize the receiver. Detection of the peak of the output of the matched filter signifies the preamble has been received and data demodulation can begin. The second and third features are a Low Pass Filter (LPF) and the downsampling process. The LPF avoids aliasing and for our studies a 32-tap Square-Root Raised Cosine (SRRC) filter is used at an oversampling factor of 8.

A matched filter is used at each of the N_a receive antennas. A multichannel transposed Finite Impulse Response (FIR) filter architecture is chosen for its resource reuse properties

[49]. Each channel's data rate is two times the sampling frequency, f_s. The multichannel design allows this block to scale very well as N_a increases. As N_a increases, the calculation speed of the filter, $N_a f_s$, is increased by a factor of N_a. This is a valid approach until the calculation clock frequency exceeds the resource switching speed of the FPGA.

5.1.2 Fast Fourier Transform.

The FFT algorithm is used in two locations, the first of which (*Block 2*) is after synchronization is accomplished. The N_a received sample chains are divided up into blocks of N subcarriers excluding the CP and are transformed to the frequency domain. The second location the FFT algorithm is used is after the impulse responses are estimated, *Block 6*. The total number of FFT calculations needed are N_a in *Block 2* and N_a^2 in *Block 6*.

The Xilinx Core Generator provides a core that calculates the FFT algorithm using Block RAM and DSP48E1 blocks. The option is also provided to instantiate the FFT in logic, which is beneficial when many DSP operations are being considered with limited DSP48E1 blocks available on the FPGA [50]. The tradeoff between using DSP48E1 blocks and slice logic is discussed in Section 5.2.

5.1.3 Store Samples.

Pilot and data tones are defined in the frequency domain; as the FFT blocks calculate and push out frequency domain samples, pilots and data samples are separated in *Block 3*. The pilot samples are stored and routed to the CE processes. The data samples are stored and await equalization.

Consideration is given to storing the samples into Block RAM. This ensures slice registers are not occupied with storage responsibilities when Block RAM is available. The architecture of the Virtex-7 Series allows dual-port Block RAM access [51]. For Block RAM to be utilized, two or less accesses to a single Block RAM is allowed in one clock cycle. To use Block RAM, resources are used wisely, but if resources are not constrained

or latency is a tighter constraint LUTs can provide a way to access more than two samples in one clock cycle.

5.1.4 Frequency Response Estimation.

Recall the pilot tones are frequency separated and the pilot samples that were transmitted are known at the receiver. The frequency response estimation block, *Block 4*, divides the received pilot sample, $y_{nr}(k)$, by the known transmitted pilot, $x_{nt}(k)$. The Xilinx divider core is instantiated to perform this calculation. This core has the ability to utilize slice logic or DSP48E1s depending on resource utilization requirements. As the number of receivers increases the number of divider cores also increases to reduce latency.

5.1.5 DFT Matrix Interpolation.

The result of the frequency response estimation block is N_a^2 vectors each having L samples representing frequency responses at particular (non-contiguous) subcarriers. The DFT matrix interpolation block, *Block 5*, uses the L samples to interpolate the full N samples of the frequency response.

To do this calculation, a pseudo-inverse of a submatrix of the DFT is used. Since the pilot tone subcarriers are known *a priori*, the submatrix and its inverse are calculated off line. $\mathcal{F}_N$ denotes the full DFT matrix. The submatrix $\mathcal{W}_{nr,nt}$ is then calculated by

$$\mathcal{W}_{nr,nt} = \mathcal{F}_N\left(1 : L, \mathbf{k}_{nr,nt}\right) \tag{5.1}$$

where $\mathbf{k}_{nr,nt}$ denote the subcarriers used as a function of the transmitter and receiver. The task of this calculation block is to simply multiply the calculated inverse times each $\hat{\tilde{\mathbf{h}}}_{nr,nt}$ to estimate each impulse response,

$$\hat{\mathbf{h}}_{nr,nt} = \mathcal{W}^{\dagger}_{nr,nt}\hat{\tilde{\mathbf{h}}}_{nr,nt} \tag{5.2}$$

where $\mathcal{W}^{\dagger}_{nr,nt} = \left(\mathcal{W}^{H}_{nr,nt}\mathcal{W}_{nr,nt}\right)^{-1}\mathcal{W}^{H}_{nr,nt}$. The total number of matrix-vector multiplications calculated by this block is N_a^2.

The result of one matrix-vector multiplication is an estimate of the impulse response between the nt^{th} and the nr^{th} pair denoted by $\hat{\mathbf{h}}_{nr,nt}$. This vector is then pushed into an FFT block, *Block 6*, outlined in Section 5.1.2 which results in the full estimated frequency response.

5.1.6 Channel Matrix Inversion.

The architecture and timing of the FFT block provides a sample every clock cycle once the FFT is calculated. This is leveraged by instantiating N_a^2 FFT blocks. Each of them is started at the same time and once the calculation is complete samples are provided at each clock cycle. During the first clock cycle that data is valid out of the FFT blocks the N_a^2 samples are reshaped into matrix $\mathbf{H}_{k=1} \in \mathbb{C}^{N_a \times N_a}$.

To invert the complex $\mathbf{H}_k$, the QR Decomposition (QRD) takes in a real matrix of the form:

$$\bar{\mathbf{H}}_k = \begin{bmatrix} \mathrm{Re}\{\mathbf{H}_k\} & -\mathrm{Im}\{\mathbf{H}_k\} \\ \mathrm{Im}\{\mathbf{H}_k\} & \mathrm{Re}\{\mathbf{H}_k\} \end{bmatrix} \tag{5.3}$$

where $\bar{\mathbf{H}}_k \in \mathbb{R}^{2N_a \times 2N_a}$. The QRD calculates $2N_a$ iterations [52]. The result is then a unitary matrix $\mathbf{Q}_k$ and an upper triangular matrix $\mathbf{R}_k$.

This process is computationally intensive and needs to be calculated $N - N_aL$ times. It is a waste of resources to instantiate $N - N_aL$ QRD blocks. This is especially true since the FFT blocks are providing a new matrix to decompose every clock cycle, excluding pilot tones. At each clock cycle the N_a^2 FFT blocks each provide a sample to populate the matrix. The data are not available all at once. Said another way, the matrix associated with $k = 10$ does not have to be known before the matrix associated with $k = 1$ is decomposed. The ability to use the time between matrix arrivals for calculation along with a higher clock frequency for QRD calculations provide great resource savings [53].

5.1.7 Equalization.

The result of this block are the equalized data symbols $\hat{\bar{\mathbf{x}}}_k$ where $\mathbf{y}_k = \mathbf{H}_k\hat{\bar{\mathbf{x}}}_k$. The QRD block decomposed $\bar{\mathbf{H}}_k$ such that $\bar{\mathbf{H}}_k = \mathbf{Q}_k\mathbf{R}_k$. Substitution then provides $\mathbf{y}_k = (\mathbf{Q}_k\mathbf{R}_k)\hat{\bar{\mathbf{x}}}_k$.

The equalization process consists of two stages. The first is matrix-vector multiplication and the second is backwards substitution, both of which are contained in *Block 8*.

The first stage calculates the matrix-vector multiplication corresponding to $\mathbf{b}_k = \mathbf{Q}^T\mathbf{y}_k$. Since $\mathbf{Q}_k$ is unitary its inverse is its transpose (since $\mathbf{Q}_k$ is real), which is considered with the format defined in Equation (5.3).

The second stage leverages the upper-triangular structure of $\mathbf{R}_k$ and after the calculations of stage one we have $\mathbf{b}_k = \mathbf{R}_k\hat{\mathbf{x}}_k$. The last row then represents one equation and one unknown which can be solved for. The second to last row now has one unknown since the last value of $\hat{\mathbf{x}}$ is known. This process is repeated $2N_a$ times. Once this is complete the equalized symbols have been calculated.

The implementation of this algorithm is straightforward, however there are many divide operations needed which on an FPGA is resource intensive. The use of DSP48E1s for the divide operation is available but depending on the number of divides needed the number of available DSP48E1s may run low.

5.1.8 Map to Bits.

After equalization the estimated symbols are mapped to bits. Since M-ary-QAM is used with $M = 4$ at the transmitter, the map to bits algorithm is just the sign bits from the real and imaginary components of the complex symbol. The amount of resources this logic, in *Block 9*, uses is negligible compared to the QRD and backwards substitution.

5.2 Resource Use Measurements

To gain intuition on the number of resources needed as a function of N_a, a VLSI implementation is carried out for $N_a \in [2\ 3\ 4]$. Extrapolation is used to determine trends as a function of the number of QRD blocks instantiated N_{QR} and N_a. This section discusses how the receiver design leverages a pipelined architecture to improve resource usage and increase the data rate. Results are reported for resource usage per component as a function of N_a.

5.2.1 Pipelined Latency.

The MIMO receiver is developed as a series of calculation blocks. Each calculation block performs its particular task and provides its outputs to the next block. In designing each block, consideration is taken to ensure that the block is able to handle the assumption that the Analog to Digital Converters (ADCs) are constantly taking data.

In Figure 5.2 the *ADC Sample* column represents the sampling of data. Each numbered block of data in Figure 5.2 represents one OFDM symbol with the CP removed. Each OFDM symbol is then N samples long. Once the receiver is synchronized the samples are fed directly into the FFT. The relation between the *ADC Sample* column and the *FFT(in)* column is that as data are being sampled there is no delay to provide the samples as input to the FFT.

The next column, *FFT Calc*, represents the pipelined architecture of the Xilinx core. Each block in this column lists the OFDM symbols that are currently being calculated by the core. Up to three OFDM symbols can be calculating in the core. Once the FFT core is completed with converting the time domain signal to the frequency domain the samples are output serially as represented in the *FFT(out)* column.

The *FFT(out)* samples are provided to the store samples block, which is used to delineate between data symbols and pilot symbols. The input to this block, column *Store Samples(in)*, stores the samples, assigns the samples in column *Assign Pilot/Data Syms* then outputs the pilots, *Pilots Out* and outputs the data symbols *Data Syms Out*.

The *Pilot Out* column provides the inputs to the impulse response estimator. The pilot tones are used to estimate each impulse response. The output of the process is shown in column *h est(out)*. Since the estimated impulse response needs to be zero padded for the FFT algorithm, *FFT (in)* starts as *h est(out)* starts but also provides the $N - L$ zeros needed to complete the N input samples for the FFT block.

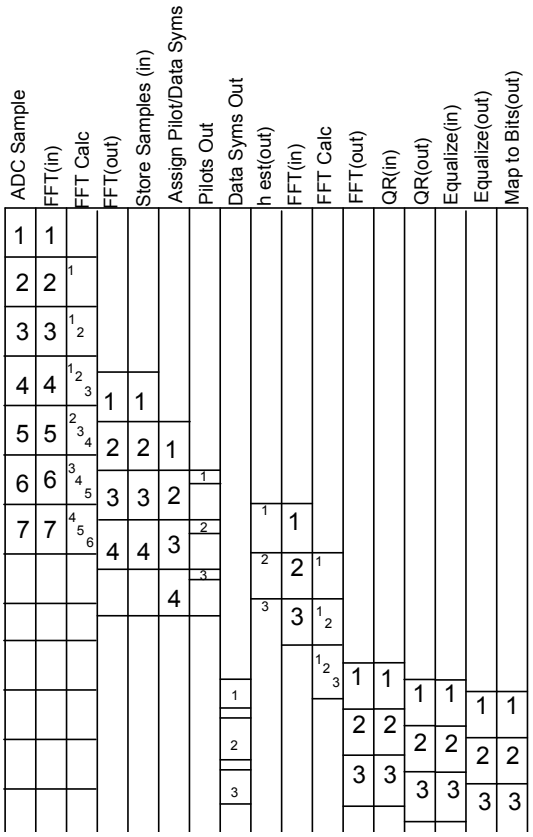


Figure 5.2: OFDM symbol cycle diagram showing the flow of OFDM symbols as numbered blocks. The N samples in each numbered block pass through the stages of calculation in the MIMO receiver. Some calculations have a higher latency (e.g. FFT) or low latency where samples are passed into a calculation while sampling.

Once again *FFT Calc* and *FFT (out)* columns provide timing delays similar to the previous FFT cores. The result of the FFT is then provided to the QRD block, *QR(in)*. After $N_{QR} = 11$ clock cycles at the data rate clock speed the first QRD is complete. A more detailed timing diagram of how the QRD cores are scheduled is provided in Figure 5.3.

The outputs of the QRD block *QR(out)* are the inputs to the equalize block. Column *Equalize(in)* not only considers the inputs from the QRD but also from the *Data Syms Out* column because these are the received samples that are going to be equalized. It is shown

that the store samples block provides the data symbols needed by the equalization block when the CE is calculated. A First-In First-Out Memory Block (FIFO) used to store the data symbols until they are needed.

Once the QRD data is provided to the equalize block in *Equalize(in)* the equalize block then calculates and outputs the equalized data symbols. The equalized data symbols are then mapped to bits, which can be done on the fly since 4-QAM is used and the sign bit is used as the demodulated bit.

The QRD algorithm is used to invert the channel matrices. This calculation represents the majority of the computational load of the MIMO receiver. For this reason, the QRD time multiplexing arbitrator is discussed in further detail. The backwards substitution process is also included in this discussion because the two processes are used to equalize the received data symbols.

The parameters used for the calculation of N_{QR} are provided in Table 5.1. After downsampling the frequency response samples are output from the FFT blocks at a rate of T_{nyq}. The QRD blocks operate at a faster clock speed with period T_{calc}. The number of clock cycles needed for the QRD as a function of N_a are reported along with the total time needed for one QRD block to finish calculation, $T_{QR} = C_{QR}T_{calc}$ and finally,

$$N_{QR} = \left\lceil \frac{T_{QR}}{T_{nyq}} \right\rceil. \tag{5.4}$$

Figure 5.3 starts with the second *FFT(out)* from Figure 5.2. The first set of samples out of the FFT represents $\mathbf{H}_1$ and the second clock cycle provides $\mathbf{H}_2$ for an $N_a = 2$ system both of these subcarriers are used for pilot tones. The QRD is not needed for these matrices. The next clock cycle provides $\mathbf{H}_3$ which is the first matrix that is provided to the QRD blocks. The first of the $N_{QR} = 11$ instantiated QRD blocks is assigned the task of decomposing $\mathbf{H}_3$. The next clock cycle $\mathbf{H}_4$ is provided. The second QRD block is used to decompose this matrix. This process continues until $\mathbf{H}_{14}$ arrives. As the $\mathbf{H}_{13}$ is assigned to the 11^{th} QRD

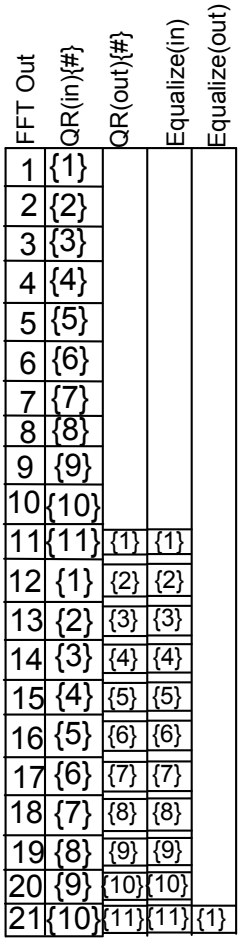


Figure 5.3: Sample based resolution of the QRD and equalization processes in the MIMO receiver, for $N_{QR} = 11$.

block, the first QRD block is completed and $\mathbf{H}_{14}$ can be assigned to the first QRD block. At this point, the QRD blocks finish, at worst, the clock cycle before the next channel matrix is available. The skipping of pilot tone subcarriers frees up QRD blocks but is inconsequential in the total execution time of the QRD calculations, particularly if the last subcarrier is a data subcarrier.

Table 5.1: N_{QR} calculation parameters

Para. Name	$N_a = 2$	$N_a = 3$	$N_a = 4$
T_{nyq}	200 ns	200 ns	200 ns
T_{calc}	10 ns	10 ns	10 ns
C_{QR}	204 cycles	300 cycles	400 cycles
T_{QR}	2040 ns	3000 ns	4000 ns
N_{QR}	11	15	20

This same process is used in the backwards substitution calculations, however the backwards substitution block needs to await the output of the QRD block before calculations can begin. But once again there are $N_{QR} = 11$ Equalize blocks instantiated to keep up with the data rate needs. As an equalize block completes the next QRD finishes to provide inputs for the next equalize process to begin.

Figure 5.2 is representative of the latency of a MIMO-OFDM receiver for any N_a. Generally, as N_a increases each calculation requires a longer amount of time for the inputs to be provided which is scaled vertically in Figure 5.2 and Figure 5.3

5.2.2 Resource Usage.

In this section, possible target FPGAs are discussed. The amount of resources each FPGA provides are considered for the MIMO receiver. These available resources are then compared to the usage of the MIMO receiver as a function of N_{QR} and N_a. The number of QRD blocks, and subsequently the equalization blocks, greatly vary the amount of resources used. If N_{QR} is decreased resources are saved at the cost of data rate. The tradeoffs of N_a, N_{QR} and data rate are considered for the MIMO receiver.

The FPGAs considered are the three largest currently provided by Xilinx. The amount of resources available on the VX980T, VX1140T, and VX2000T are provided in Table 5.2. The VX980T is the target for the implementations carried out in this paper.

Table 5.2: Resources available on Xilinx Virtex-7 FPGAs [51].

FPGA	Slice Regs	Slice LUTs	DSP48E1s
VX980T	1,224K	612K	3,600
VX1140T	1,424K	712K	3,360
VX2000T	2,443K	1,221.6K	2,160

Table 5.3 provides implementation results for the nine blocks for $N_a = 2$. The block numbers correspond to block numbers provided in Section 5.1. For each of the blocks the number of resources were calculated. The Block RAM usage is not reported since the DSP48E1s and LUTs were found to be the limiting factor in the resource constraints on the FPGA.

As N_{QR} is increased from one the scalability is determined as a function of the number of QRD blocks instantiated. *Blocks 7* and *8* scaled linearly with N_{QR}. This is completed to provide intuition for the $N_a = 3$ and $N_a = 4$ case, where the full real-time implementation does not fit on the VX980T FPGA. For $N_a = 3$ up to $N_{QR} = 5$ QRD and Equalize blocks were instantiated. The true resource counts for the nine blocks for each value of $N_{QR} \in \{1, 2, 3, 4, 5\}$ are then linearly extrapolated to determine how many resources are needed for each block for $N_a = 3$. The same technique is used for $N_a = 4$ where $N_{QR} \in \{1, 2, 3, 4\}$.

In Table 5.4, the estimated amount of resources used for $N_a = 3$ are provided. Comparing the values in Table 5.3 and Table 5.4 in most cases there is an increase in resources need. These blocks show a dependance on N_a. However, *Block 1* or the correlation block does not increase, this is because of the multi-channel architecture. The same amount of resources are used but the logic is run at N_a times the ADC's clock frequency. However, this architecture has a maximum operating frequency. As N_a

increases, the limit is reached. Once this occurs, another instance of the multi-channel FIR filter would need to be instantiated.

There is one block that decreases in resources used. *Block 3* or the block that stores the pilot samples and the data symbol samples. This is because the Xilinx tools made use of Block RAM saving Slices for use in arithmetic calculations. This could have been done in the $N_a = 2$ case but it was not necessary since the amount of slices was not constrained.

The use of Block RAM also impacts latency of the calculation blocks where the dual-port architecture, described in Section 5.1.3, limits the number of memory accesses per clock cycle. If LUTs are used to store the values they are all available in a clock cycle. However, if Block RAM is used the latency is increased because more clock cycles are needed to retrieve the data.

Finally comparing Table 5.4 to Table 5.5 the amount of resources for each block increase with again the exception of the multi-channel FIR filter. However, *Blocks 5* and *8* reduce in registers and LUTs and utilize DSP48E1 blocks. There is a trade off between DSP48E1s and slice logic, namely, execution speed but if one resource is constrained the other may alleviate the constraint. Attributes in VHDL control whether DSP48E1 blocks are inferred from the design or if they are avoided. Consider the three FPGAs outlined in Table 5.2. Even though the VX2000T is a larger FPGA in terms of registers and LUTs it is not in terms of DSP48E1 blocks. The amount of resources available dictates what design attributes are needed.

To gain intuition on how the MIMO receiver design grows as a function of N_a extrapolation is used once again with the resource counts for $N_a \in \{2\ 3\ 4\}$. The extrapolation is done for each block separately; in this case a quadratic fit is used for most blocks since, for example, the backwards substitution algorithm is $O\left(n^2\right)$. Some problematic blocks

Table 5.3: Block utilization in MIMO receiver for $N_a = 2$

Block	Slice Regs	Slice LUTs	DSP48E1s
1	1,114	2,976	0
2	3,527	2,466	36
3	9,160	7,618	0
4	6,601	5,844	72
5	2,578	3,451	0
6	6,939	4,930	72
7	158,986	327,015	1,584
8	19,148	35,965	99
9	1	0	0
Total	208,054	390,265	1,863

such as equalization, where DSP48E1s are used instead of slices, makes the extrapolation an estimate. In this case an increasing linear fit is used.

Figure 5.4 shows the number of LUTs vs DSP48E1s. For values of $N_a \in \{4,\ 6,\ 8,\ 10\}$ asterisks are used to represent the estimated utilization of these real-time implementation. The number of FPGAs needed to satisfy the amount of resources needed by these implementations are represented by the lines. It is necessary to note that in this type of implementation, where multiple FPGAs are used, an added step of optimizing the network of FPGAs is needed. It is not a trivial task in assigning calculation blocks to specific FPGAs.

Table 5.4: Block utilization in MIMO receiver for $N_a = 3$

Block	Slice Regs	Slice LUTs	DSP48E1s
1	1,114	2,976	0
2	5,233	3,698	54
3	6,580	7,058	0
4	14,851	13,149	162
5	2,773	3,807	0
6	15,469	11,093	162
7	501,649	1,423,901	4,860
8	33,709	91,209	135
9	1	0	0
Total	581,378	1,556,889	5,373

With the estimates of resource utilization calculated, the number of resources per calculation block for each value in N_{QR} and N_a are available. Looking at Table 5.3, Table 5.4 and Table 5.5 the block with the largest resource usage is the QRD. The second largest is the equalization block. Both of these blocks are linked in the way that if N_{QR} is reduced the input to the equalization block is reduced, requiring less matrix-vector multipliers and backwards substitution blocks. The number of these blocks is directly related to how much time is available for calculation before the next OFDM symbol is ready for calculation (illustrated in Figure 5.3). If there were a delay between OFDM symbols; operating

Table 5.5: Block utilization in MIMO receiver for $N_a = 4$

Block	Slice Regs	Slice LUTs	DSP48E1s
1	1,114	2,976	0
2	6,939	4,932	72
3	8,848	9,401	0
4	26,401	23,360	288
5	2,434	2,813	40
6	25,705	18,480	270
7	2,243,379	81,706,853	79,113
8	51,132	44,757	1,921
9	1	0	0
Total	2,366,000	81,813,527	82,704

the burst communication mode, the number of QRD, matrix-vector multiplication and backwards substitution blocks could be reduced.

Figure 5.5 shows the total number of resources used as a function of N_a. The total number of resources is compared to the usage of just the QRD block, *Block 7*. The QRD makes up about $76 - 98$ percent of the total design depending on N_a and the resource. As N_a increases the QRD increases faster than any other block.

The delay between OFDM symbols is denoted T_L. If T_L is set to zero, there is no gap between OFDM symbols. In this case the MIMO receiver is real-time compatible.

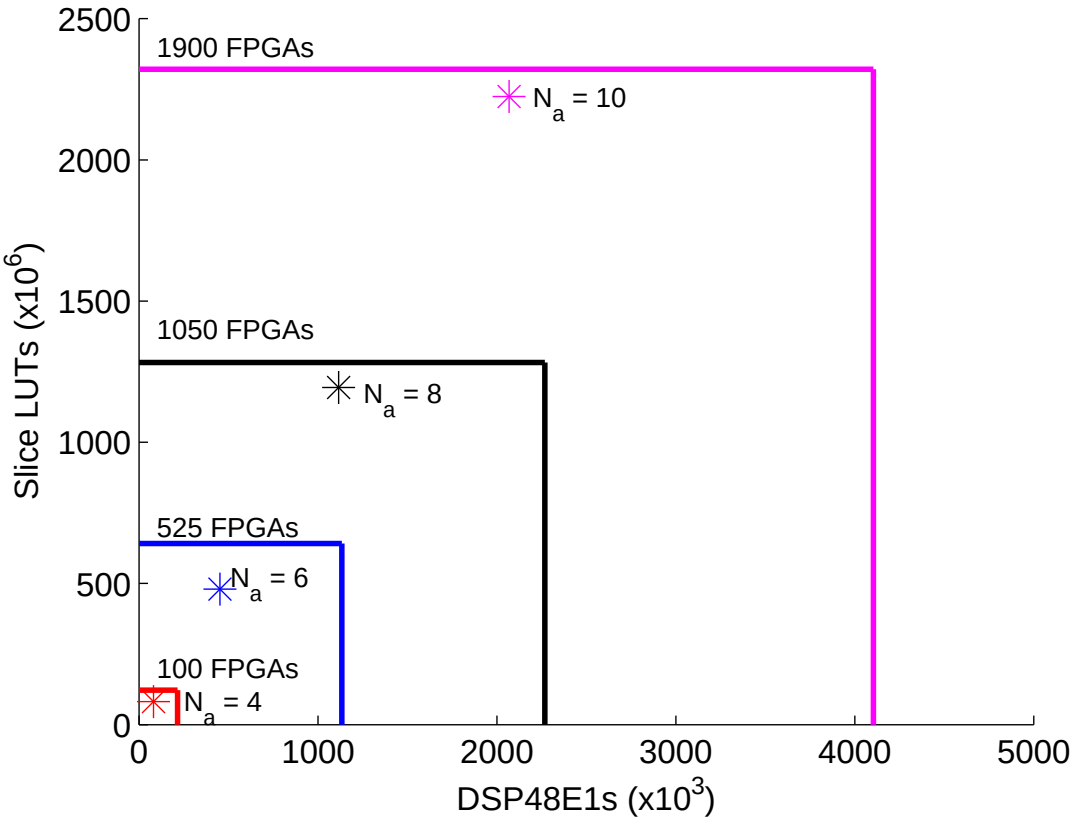
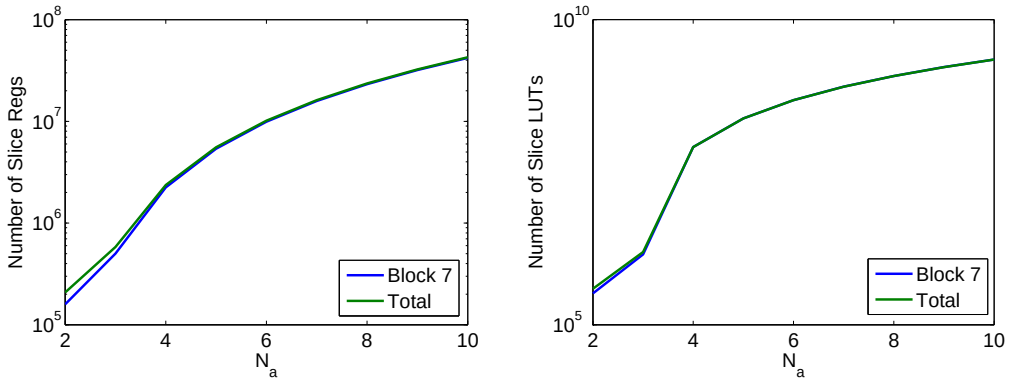


Figure 5.4: Extrapolated slice LUTs and DSP48E1 usage for $N_a = [4, 6, 8, 10]$. Also pictured are the available resources for 100, 525, 1050 and 1900 FPGAs.

If the real-time capability constraint were lifted, effectively making $T_L > 0$, the design size could be reduced at the cost of the data rate capable by the receiver. The data rate the receiver is capable of is given by:

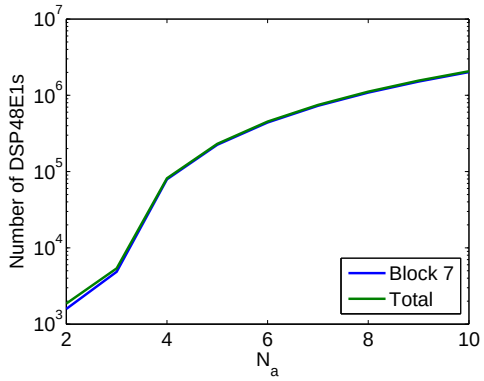
$$R = \frac{\log_2(M)N_a(N - LN_a)}{(T_s N') + T_L} \tag{5.5}$$

where M is the constellation order, $N - N_a L$ is the number of data subcarriers, T_s is the sampling frequency at base band, $N' = N + N_{cp}$, N_{cp} is the CP length and T_L is the amount of time between OFDM symbols which corresponds to the amount of additional time now available to the QRD and equalization blocks for calculation.



(a) Slice Register Usage

(b) Slice LUT Usage



(c) DSP48E1 Usage

Figure 5.5: Total resource usage as a function of N_a compared to the QRD block for real-time compliant MIMO receiver.

Figure 5.6 shows the relation between N_{QR} and data rate. The values for T_L were determined empirically by considering the arrival rate from the FFT block, certain subcarriers do not need to be demixed since they are pilot tones, and the calculation time for a QRD block as a function of N_a. These T_L values represent the minimum amount of time required between the OFDM symbols for the QRD and equalization chain to complete in time for the next OFDM block. Since the minimum T_L is found the maximum data rate is shown in Figure 5.6. When $N_{QR} = 10$ the data rate is increasing rapidly then at $N_{QR} = 11$

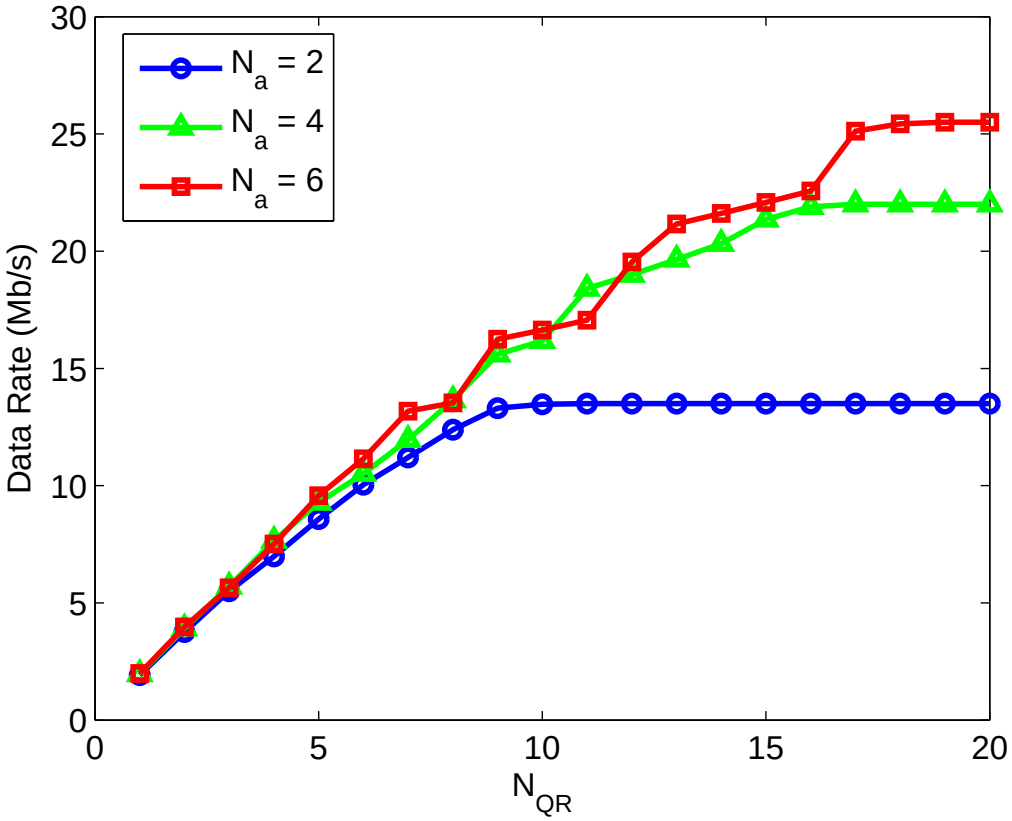


Figure 5.6: As the number of QRD blocks instantiated increases, the data rate possible at the receiver increases since subcarriers can be demixed at a faster rate. As the number of antennas in the MIMO system increase the data rate also increases. For these calculations $M = 4$, $N = 64$, $T_s = 200$ ns.

the data rate is starting to level out. At this point the maximum data rate has been acheived for the system parameters the QRD is no longer to limiting factor in the system. To increase the data rate at this point, the constellation order or bandwidth need to be increased. Also, N_a may be increased but this does not necessarily improve the data rate. For instance, $N = 64$, $N_a = 6$ and $L = 5$, this results in $N_{dt} = 34$ data tones. If N_a is increased to $N_a = 7$, of the $N = 64$ newly added subcarriers available for data transmission only 24 of them are

assign data the rest are used for pilots. Increasing the number of subcarriers provides a larger data rate increase in this scenario.

Figure 5.7 shows the number of FPGAs required to implement all the of logic for the MIMO receiver design. There is an assumption that the FPGAs can be interfaced with a bus that is capable of maintaining a data rate needed for the FPGAs to communicate effectively. The Low Voltage Differential Signaling (LVDS) interface methods are simple to implement and provide a high data rate [54].

The log of the number of FPGAs are reported to show trends at lower N_{QR} and for a given N_a there is some N_{QR} where the FPGA that should be targeted changes which is clearer with the log scale. The reason for the change is that the QRD algorithm uses more LUTs as opposed to DSP48E1s. When N_{QR} is small, the DSP48E1s (from the FFTs, etc.) dictate which FPGA to target. As N_{QR} increases, the LUTs introduced by the newly instantiated QRD blocks dictate which FPGA to target.

5.3 Conclusions

This chapter discusses a full MIMO-OFDM communications receiver where the number of antennas used at each transmitter and receiver is $N_a = 2$, 3 and 4. Each component in the receiver is described and resource utilization reports are provided. Trends in resource usage were extrapolated for up to $N_a = 10$. The number of QRD and equalization blocks were varied to determine the tradeoff between data rate and resource usage.

It is found that for $N_a = 2$ the full real-time system is implementable on the Xilinx VX980T FPGA. For $N_a = 3$ and $N_a = 4$ the full real-time system could not be implemented on a single FPGA since the design uses too many resources. The number of QRD and equalize blocks were reduced (reducing the data rate) for the $N_a = 3$ and $N_a = 4$ design to successfully fit on the VX980T.

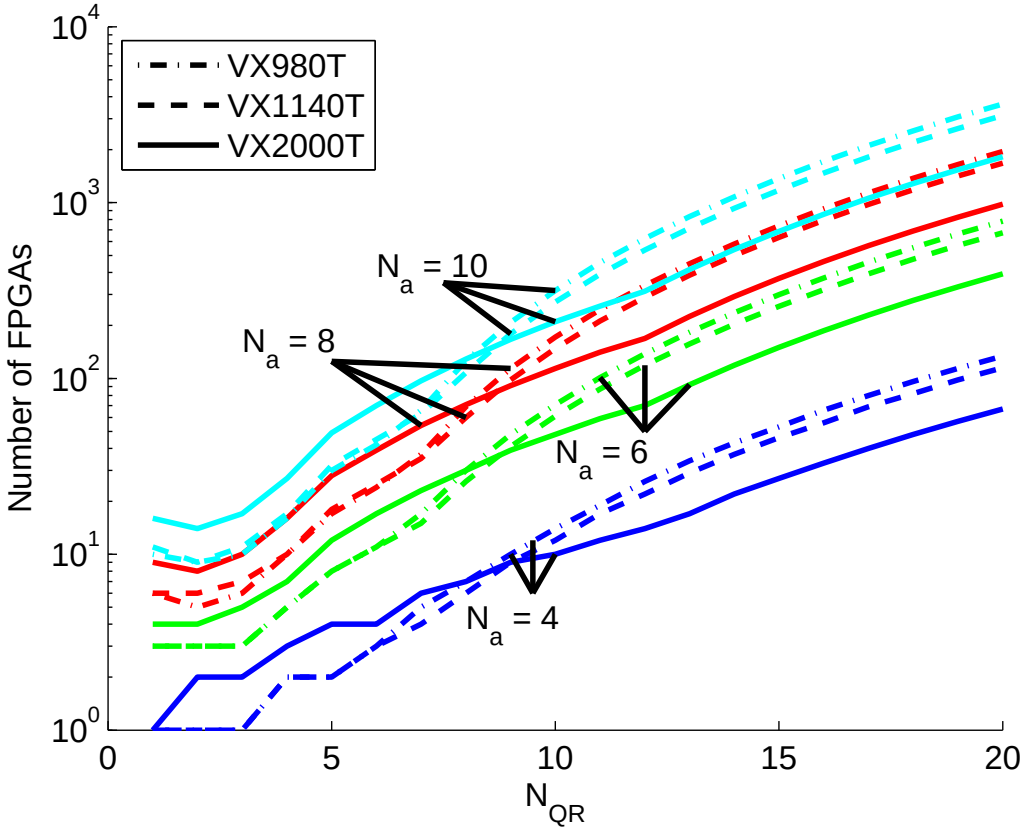


Figure 5.7: As the number of QRD block instantiated increases, the number of FPGAs needed to implement the receiver increases. As the number of antennas in the MIMO system increase the number of FPGAs needed to implement the receiver also increases.

With this information, the relationships between data rate and N_{QR} and N_a were discovered. The number of FPGAs and N_{QR} and N_a are also related by the graphs provided. Finally, the decision for the target FPGA is considered when a variable number of Slices and DSP48E1s are needed.

The QRD has been presented as the most expensive, in terms of resources and computational time, in the MIMO receiver algorithm. In going forward with this work QRD algorithms described in [55] offer a reduced area algorithm that may be suitable for

the receiver described in this paper. Other methods for matrix decomposition may also have an advantage for this particular application such as [56].

VI. Conclusions

This chapter concludes the dissertation by reiterating the results from the three thrusts of the dissertation. Also mentioned are the journal articles published as a result of the findings in this dissertation.

6.1 Covert MIMO Communications

Chapter III describes and analyzes a transmitter and receiver topology in which IBI is leveraged to provide a physical layer security measure. The intended receiver's performance is orders of magnitude better in terms of BER compared to an Ex at $\theta = 45°$. This result shows that IBI considered a hindering effect in traditional multicarrier communications can be used to degrade unintended users' performance.

Also in Chapter III the derivation of IBI in MIMO communications is presented. This derivation considers the influence of IBI for a given set of $N_r N_t$ channels. A particular topology is not assumed in the derivation; only the channel lengths and their relation to the CP length drives the effect of IBI. Whether the length of the channel is elongated by topological delays or the environment itself the IBI and ultimately the SINR are characterized for a general MIMO communication system for specific channels.

The task of estimating the channels in a MIMO system is also considered. The CE accuracy affects the performance of the receiver. Two CE algorithms are considered, the TDCE and FDCE and the performance of both are compared. The FDCE is the method of choice for the cooperative system. For Ex the TDCE should be leveraged provided that the delays between users are known or can be estimated.

6.2 TOA Estimation with TDOA extension

As the Ex, the relative delays between users are needed to improve detection performance. Chapter IV provides simulation and experimental results to support the

ability to estimate the TOA. First, simulations are conducted of the *van de Beek* and *Acharya* methods. The success of both algorithms then lead into a small scale hardware implementation.

The small scale hardware implementation consists of two WARP boards. The first of these uses four radio cards that represent four users. For each of the users the transmit signal is delayed. The second WARP board utilizes a single receive radio card that estimates the four delay values. The *van de Beek* and *Acharya* methods are compared while the *van de Beek* method is able to provide relative delays the delay is not able to be tied to a specific user. This is problematic for Ex because in the equalization process the delay for each user is needed.

The *Acharya* method does however provide the delay per user. This is accomplished by leveraging the subcarriers occupied by each user. In the OFDMA scheme each user is allotted specific subcarriers. If this is known at Ex the delays can be estimated.

A natural extension of this work is to TDOA positioning. Determining TDOA measurements are also looked at via a large scale hardware implementation. For the large scale hardware implementation to be successful synchronization between receivers needs to be established. This is successfully accomplished by a reference transmitter. A hallway test is completed to provide a proof of concept for this method.

To provide a more interesting testbed another limitation must be overcome. The Ethernet cables that link the controlling computer to the WARP board receivers must be removed to enlarge the testbed. This is accomplished by the Ethernet bridge for future students to implement TDOA based positioning algorithms on.

6.3 FPGA Implementation Scalability for MIMO Receivers

Chapter V implements a MIMO receiver with 2, 3, and 4 transmit and receive antennas. The FPGA resources are reported for each of these implementations in terms of Slice LUTs, Slice Registers and DSP48E1s for Xilinx FPGAs. The resources used by

each of the nine components of the receiver are extrapolated for larger MIMO systems. The trends gathered from the extrapolation provide intuition on the number of FPGAs needed to implement a MIMO receiver for a specific data rate.

Data rate is also considered as a function of resource usage. The tradeoff between resources used and the data rate the receiver is capable of is elaborated on in the form of the number of QRD blocks instantiated. As the number of QRD blocks instantiated decreases the data rate is reduced, and the receiver requires the OFDM symbols to arrive less frequently.

6.4 Publications

The work done in this dissertation lead to two journal article submissions and potentially a conference paper. The work described in Chapter III has been submitted to *IEEE Transactions on Information Forensics and Security* on 10 January 2014, under the title "Throughput Preserving Physical Layer Security Leveraging Inter-Block Interference". The article is still under review at this time.

The second journal article takes the form of Chapter V. The work described is submitted to *IEEE Transactions on Circuits and Systems I* on 5 March 2014 under the title "FPGA Resource Scalability Study for Large-Scale MIMO-OFDM Receivers". The article is still under review.

The implementations of the TDOA work in Chapter IV is being prepared for submission to Asilomar Conference on Signal, Systems, and Computers with a due date of 1 May 2014.

6.5 Future Work

The topology analyzed in Chapter III provides insight for that particular military application. Some other interesting topologies could include cellular networks where the role of the eavesdropper is a mobile user in an adjacent cell. Natural extensions of this

contribution would be to develop a testbed for this work. An FPGA implementation of the TDCE would be very complex for such a platform. The added ability to estimate the TOA values for each user via the *Acharya* method on multiple FPGAs would be very interesting. However, the *Acharya* method is too complex for a real-time implementation. Determining a less computationally complex estimator is beneficial for practical applications.

Further work into the calculations involved in channel equalization is needed. For MIMO receivers to grow in size the ability to equalize faster is needed. The point of increasing the number of antennas is to increase data rate but the computation complexity of the matrix inversion and equalization are the biggest issues in the design.

Appendix: Multiple OFDM Symbol TOA Estimator

Section 5.1 discusses the TOA estimator for multiple OFDM symbols. The PDF of the nbth received OFDM symbol is:

$$p(\mathbf{y}_{nb}|\bar{\theta}_{nt}) = \frac{1}{\pi^{N'+N}\,|\mathbf{C}_\theta|}\exp\left\{-\frac{1}{2}\mathbf{y}_{nb}^H\mathbf{C}_\theta^{-1}\mathbf{y}_{nb}\right\}. \tag{A.1}$$

where

$$\begin{aligned}
\tilde{\mathbf{C}} =& \mathrm{E}\left\{\mathbf{x}_{cp}\mathbf{x}_{cp}^H\right\} && \text{(A.2)}\\
=& \mathrm{E}\left\{(\mathbf{Q}\mathcal{F}_N^H\mathbf{P}\tilde{\mathbf{x}})(\mathbf{Q}\mathcal{F}_N^H\mathbf{P}\tilde{\mathbf{x}})^H\right\} && \text{(A.3)}\\
=& \mathrm{E}\left\{(\mathbf{Q}\mathcal{F}_N^H\mathbf{P}\tilde{\mathbf{x}})(\tilde{\mathbf{x}}^H\mathbf{P}^H\mathcal{F}_N\mathbf{Q}^H)\right\} && \text{(A.4)}\\
=& \mathbf{Q}\mathcal{F}_N^H\mathbf{P}\mathrm{E}\left\{\tilde{\mathbf{x}}\tilde{\mathbf{x}}^H\right\}\mathbf{P}^H\mathcal{F}_N\mathbf{Q}^H. && \text{(A.5)}
\end{aligned}$$

and

$$\mathbf{C}_\theta = \mathrm{diag}\left[\mathbf{I}_\theta,\ \tilde{\mathbf{C}},\ \mathbf{I}_{N-\theta}\right] \tag{A.6}$$

Assuming each received OFDM symbol is independent, the log-likelihood function to be maximized is:

$$\begin{aligned}
\mathcal{L} =& \log\left\{\prod_{nb=1}^{N_b} p(\mathbf{y}_{nb}|\bar{\theta}_{nt})\right\} && \text{(A.7)}\\
=& \sum_{nb=1}^{N_b}\log p(\mathbf{y}_{nb}|\bar{\theta}_{nt}) && \text{(A.8)}\\
=& \sum_{nb=1}^{N_b} -\left[\log|\mathbf{C}_\theta| + \frac{1}{2}\mathbf{y}_{nb}^H\mathbf{C}_\theta^{-1}\mathbf{y}_{nb}\right] && \text{(A.9)}\\
=& -\left(\sum_{nb=1}^{N_b}\log|\mathbf{C}_\theta| + \sum_{nb=1}^{N_b}\frac{1}{2}\mathbf{y}_{nb}^H\mathbf{C}_\theta^{-1}\mathbf{y}_{nb}\right) && \text{(A.10)}\\
=& -\left(N_b\log|\mathbf{C}_\theta| + \sum_{nb=1}^{N_b}\frac{1}{2}\mathbf{y}_{nb}^H\mathbf{C}_\theta^{-1}\mathbf{y}_{nb}\right). && \text{(A.11)}
\end{aligned}$$

The minimization of the negation of Equation (A.11) provides the estimator used in Section 5.1:

$$\hat{\theta} = \underset{\theta}{\operatorname{argmin}} \left(N_b \log |\mathbf{C}_\theta| + \sum_{nb=1}^{N_b} \frac{1}{2} \mathbf{y}_{nb}^H \mathbf{C}_\theta^{-1} \mathbf{y}_{nb} \right) \tag{A.12}$$

Bibliography

[1] E. Telatar, "Capacity of Multi-antenna Gaussian Channels," *European Transactions on Telecommunications*, vol. 10, pp. 585–595, September 1999.

[2] G. J. Foschini and M. J. Gans, "On Limits of Wireless Communications in a Fading Environment when using Multiple Antennas," *Wireless Personal Communications*, vol. 6, pp. 311–335, March 1998.

[3] S. Alamouti, "A Simple Transmit Diversity Technique for Wireless Communications," *IEEE Journal on Selected Areas in Communications*, vol. 16, pp. 1451–1458, August 1998.

[4] A. Tulino, A. Lozano, and S. Verdu, "Impact of Antenna Correlation on the Capacity of Multiantenna Channels," *IEEE Transactions on Information Theory*, vol. 51, pp. 2491–2509, July 2005.

[5] "IEEE 802.11n-2009 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. Amendment 4: Enhancements for Higher Throughput.," *IEEE Std P802.11n*, October 2000.

[6] S. Boumard, M. Weissenfelt, H. Chi, and J. Nurmi, "A Wireless MIMO-STC-OFDM System Implementation," in *International Symposium on Personal, Indoor and Mobile Radio Communications*, pp. 1–5, September 2006.

[7] S. Yoshizawa and Y. Miyanaga, "VLSI Implementation of a 4x4 MIMO-OFDM transceiver with an 80-MHz channel bandwidth," in *IEEE International Symposium on Circuits and Systems*, pp. 1743–1746, May 2009.

[8] R. Couillet and M. Debbah, *Random Matrix Methods for Wireless Communications*.

[9] S. Ohno and G. Giannakis, "Capacity Maximizing MMSE-Optimal Pilots for Wireless OFDM over Frequency-Selective Block Rayleigh-Fading Channels," *IEEE Transactions on Information Theory*, vol. 50, pp. 2138–2145, September 2004.

[10] R. Negi and J. Cioffi, "Pilot Tone Selection for Channel Estimation in a Mobile OFDM System," *IEEE Transactions on Consumer Electronics*, vol. 44, pp. 1122–1128, August 1998.

[11] T. Kim and J. Andrews, "Optimal Pilot-to-Data Power Ratio for MIMO-OFDM," in *Global Telecommunications Conference*, vol. 3, pp. 1481–1485, November 2005.

[12] I. Barhumi, G. Leus, and M. Moonen, "Optimal Training Design for MIMO-OFDM Systems in Mobile Wireless Channels," *IEEE Transactions on Signal Processing*, vol. 51, pp. 1615–1624, June 2003.

[13] X. Geng, H. Hu, W. Cui, and Y. Dun, "Optimal Pilot Design for MIMO-OFDM Channel Estimation," in *International Conference on Signal Processing Systems*, vol. 2, pp. 404–408, July 2010.

[14] E. Eitel and J. Speidel, "Efficient Training of Adaptive MIMO Channel Tracking Algorithms," in *European Wireless Conference*, pp. 1–7, April 2012.

[15] G. Ignatius, U. Murali Krishna Varma, N. S. Krishna, P. Sachin, and P. Sudheesh, "Extended Kalman Filter Based Estimation for Fast Fading MIMO Channels," in *International Conference on Devices, Circuits and Systems*, pp. 466–469, March 2012.

[16] A. Kazmi and N. Khan, "Channel Tracking and Equalization using Kalman Estimation for MIMO Systems in Non-Isotropic Ricean fading environment," in *International Conference on Information and Emerging Technologies*, pp. 1–5, June 2010.

[17] M. Batariere, K. Baum, and T. Krauss, "Cyclic Prefix Length Analysis for 4G OFDM Systems," in *Vehicular Technology Conference*, vol. 1, pp. 543–547, September 2004.

[18] S. Celebi, "Interblock Interference (IBI) and Time of Reference (TOR) Computation in OFDM Systems," *IEEE Transactions on Communications*, vol. 49, pp. 1895–1900, November 2001.

[19] A. Filippi and S. Serbetli, "OFDM Symbol Synchronization using Frequency Domain Pilots in Time Domain," *IEEE Transactions on Wireless Communications*, vol. 8, pp. 3240–3248, June 2009.

[20] V. Kotzsch, W. Rave, and G. Fettweis, "ISI Analysis in Network MIMO-OFDM Systems with Insufficient Cyclic Prefix Length," in *International Symposium on Wireless Communication Systems*, pp. 189–193, September 2010.

[21] J. Seoane, S. Wilson, and S. Gelfand, "Analysis of Intertone and Interblock Interference in OFDM when the Length of the Cyclic Prefix is Shorter than the Length of the Impulse Response of the Channel," in *Global Telecommunications Conference*, vol. 1, pp. 32–36 vol.1, November 1997.

[22] M. B. Stefania Sesia, Issam Toufik, *LTE The UMTS Long Term Evolution: From theory to Practice*. West Sussex, United Kingdom: Wiley, 2011.

[23] R. Martin, J. Walsh, and C. Johnson, "Low-Complexity MIMO Blind, Adaptive Channel Shortening," *IEEE Transactions on Signal Processing*, vol. 53, pp. 1324–1334, April 2005.

[24] B. Sklar, *Digital Communications: Fundamentals and Applications*. Prentice Hall Communications Engineering and Emerging Technologies Series, Prentice-Hall PTR, 2001.

[25] S. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. No. v. 1 in Prentice Hall signal processing series, Prentice-Hall PTR, 1998.

[26] J. Acharya, H. Viswanathan, and S. Venkatesan, "Timing acquisition for non contiguous ofdm based dynamic spectrum access," in *IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks*, pp. 1–10, Oct 2008.

[27] J.-J. van de Beek, M. Sandell, and P. Borjesson, "ML Estimation of Time and Frequency Offset in OFDM Systems," *IEEE Transactions on Signal Processing*, vol. 45, pp. 1800–1805, July 1997.

[28] O. Font-Bach, N. Bartzoudis, A. Pascual-Iserte, and D. L. Bueno, "A real-time MIMO-OFDM mobile WiMAX receiver: Architecture, design and FPGA implementation," *Computer Networks*, vol. 55, pp. 3634–3647, November 2011.

[29] C. Maxfield, *The Design Warrior's Guide to FPGAs: Devices, Tools and Flows*. Elsevier Science, 2004.

[30] "WARP Project." http://warpproject.org.

[31] W. Zhu, B. Daneshrad, J. Bhatia, H.-S. Kim, D. Liu, K. Mohammed, R. Prabhu, S. Sasi, and A. Shah, "MIMO Systems for Military Communications," in *Military Communications Conference*, pp. 1–7, October 2006.

[32] C.-Y. Wu, P.-C. Lan, P.-C. Yeh, C.-H. Lee, and C.-M. Cheng, "Practical Physical Layer Security Schemes for MIMO-OFDM Systems Using Precoding Matrix Indices," *IEEE Journal on Selected Areas in Communications*, vol. 31, pp. 1687–1700, September 2013.

[33] M. Haleem, C. Mathur, R. Chandramouli, and K. P. Subbalakshmi, "Opportunistic Encryption: A Trade-Off between Security and Throughput in Wireless Networks," *IEEE Transactions on Dependable and Secure Computing*, vol. 4, pp. 313–324, October 2007.

[34] A. Chorti and I. Kanaras, "Masked M-QAM OFDM: A simple approach for enhancing the security of OFDM systems," in *IEEE 20th International Symposium on Personal, Indoor and Mobile Radio Communications*, pp. 1682–1686, September 2009.

[35] T. Yucek and H. Arslan, "Feature Suppression for Physical-layer Security in OFDM Systems," in *Military Communications Conference*, pp. 1–5, October 2007.

[36] M. Pan, T. Clancy, and R. McGwier, "Jamming Attacks Against OFDM Timing Synchronization and Signal Acquisition," in *Military Communications Conference*, pp. 1–7, October 2012.

[37] Z. Li and X.-G. Xia, "A Distributed Differentially Encoded OFDM Scheme for Asynchronous Cooperative Systems with Low Probability of Interception," *IEEE Transactions on Wireless Communications*, vol. 8, pp. 3372–3379, July 2009.

[38] C. Sperandio and P. Flikkema, "Wireless Physical-Layer Security via Transmit Precoding Over Dispersive Channels: Optimum Linear Eavesdropping," in *Military Communications Conference*, vol. 2, pp. 1113–1117 vol.2, October 2002.

[39] J. Milton and J. Arnold, *Introduction to Probability and Statistics: Principles and Applications for Engineering and the Computing Sciences*. MCGRAW-HILL Higher Education, 2002.

[40] M. Boas, *Mathematical Methods in the Physical Sciences*. John Wiley & Sons Australia, Limited, 2006.

[41] "Netgear Universal WiFi Adapter." http://www.netgear.com/home/products/connected-entertainment/gaming-home-theater/WNCE2001.aspx.

[42] A. Alimohammad and B. Cockburn, "An Efficient Parallel Architecture for Implementing LST Decoding in MIMO Systems," *IEEE Transactions on Signal Processing*, vol. 54, pp. 3899–3907, October 2006.

[43] A. Burg, S. Haene, D. Perels, P. Luethi, N. Felber, and W. Fichtner, "Algorithm and VLSI architecture for linear MMSE detection in MIMO-OFDM systems," in *IEEE International Symposium on Circuits and Systems*, pp. 4102–4105, May 2006.

[44] X. Huang, C. Liang, and J. Ma, "System Architecture and Implementation of MIMO Sphere Decoders on FPGA," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 16, pp. 188–197, February 2008.

[45] R. Shariat-Yazdi and T. Kwasniewski, "Reconfigurable K-best MIMO detector architecture and FPGA implementation," in *International Symposium on Intelligent Signal Processing and Communication Systems*, pp. 349–352, November 2007.

[46] Y. Sun and J. Cavallaro, "Trellis-Search Based Soft-Input Soft-Output MIMO Detector: Algorithm and VLSI Architecture," *IEEE Transactions on Signal Processing*, vol. 60, pp. 2617–2627, May 2012.

[47] H. Wang, P. Leray, and J. Palicot, "A reconfigurable architecture for MIMO square root decoder," in *Reconfigurable Computing: Architectures and Applications*, pp. 317–322, Springer, March 2006.

[48] B. Yin, M. Wu, C. Studer, J. R. Cavallaro, and C. Dick, "Implementation Trade-offs for Linear Detection in Large-Scale MIMO Systems," *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 2679–2683, May 2013.

[49] *DSP: Designing for Optimal Results : High-Performance DSP Using Virtex-4 FPGAs*, December 2013. http://www.xilinx.com/publications/archives/books/dsp.pdf.

[50] A. Jiménez-Pacheco, Á. Fernández-Herrero, and J. Casajús-Quirós, "Design and implementation of a hardware module for MIMO decoding in a 4G wireless receiver," *VLSI Design*, vol. 2008, pp. 1–8, October 2008.

[51] *Xilinx 7 Series FPGAs Overview*, 1.14 ed., July 2013. http://www.xilinx.com/support/documentation/data_sheets/ds180_7Series_Overview.pdf.

[52] C. Singh, S. H. Prasad, and P. Balsara, "VLSI Architecture for Matrix Inversion using Modified Gram-Schmidt based QR Decomposition," in *International Conference on VLSI Design*, pp. 836–841, January 2007.

[53] C. Studer, P. Blosch, P. Friedli, and A. Burg, "Matrix Decomposition Architecture for MIMO Systems: Design and Implementation Trade-offs," in *Asilomar Conference on Signals, Systems and Computers*, pp. 1986–1990, November 2007.

[54] *Virtex-7 T and XT FPGAs Data Sheet : DC and AC Switching Characteristics*, 1.18 ed., November 2013. http://www.xilinx.com/support/documentation/data_sheets/ds183_Virtex_7_Data_Sheet.pdf.

[55] D. Patel, M. Shabany, and P. Gulak, "A low-complexity high-speed QR decomposition implementation for MIMO receivers," in *IEEE International Symposium on Circuits and Systems*, pp. 33–36, May 2009.

[56] S. Yoshizawa, Y. Yamauchi, and Y. Miyanaga, "A complete pipelined MMSE detection architecture in a 4x4 MIMO-OFDM receiver," in *IEEE International Symposium on Circuits and Systems*, pp. 2486–2489, May 2008.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704–0188

1. REPORT DATE *(DD–MM–YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From — To)*
19–06–2014	Doctoral Dissertation	June 2011–June 2014

4. TITLE AND SUBTITLE

Scalable System Design for Covert MIMO Communications

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Pennington, Jason R.,

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 Hobson Way
WPAFB, OH 45433-7765

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT-ENG-DS-14-J-5

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Intentionally Left Blank

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

DISTRIBUTION STATEMENT A:
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

13. SUPPLEMENTARY NOTES

14. ABSTRACT

In modern communication systems, bandwidth is a limited commodity. Bandwidth efficient systems are needed to meet the demands of the ever-increasing amount of data that users share. Of particular interest is the U.S. Military, where high-resolution pictures and video are used and shared. In these environments, covert communications are necessary while still providing high data rates. The promise of multi-antenna systems providing higher data rates has been shown on a small scale, but limitations in hardware prevent large systems from being implemented.

Discussed here are the effects of the topology of communication nodes on Inter-Block Interference in OFDM systems. This effect can be leveraged such that eavesdroppers experience a lower SINR resulting in a poor quality communication link. Simulations show that an eavesdropper has a 10 dB worse SINR. The reverse is also considered where the point of view is taken as the eavesdropper. A study into improving the eavesdropping communication link performed. A pivotal calculation for the eavesdropper is found to be the estimation of the time of arrival of the received waveforms. The relative delays between users' waveforms is used to reduce the interference at the eavesdropper. The *van de Beek* and *Acharya* methods are considered. Simulations and experiments show that the *Acharya* method provides a more accurate measurement. Also discussed are hardware limitations such as on board slice logic and DSP resources blocks. The utilization of these logic blocks proves to be a limiting factor in large scale multi-antenna systems. Particularly the inversion and equalization processes are the most expensive in terms of computation time and hardware resources. The trade-off between data rate and resource usage is provided with comments on interfacing multiple FPGAs to provide more available resources.

15. SUBJECT TERMS

MIMO Communications, OFDM, TOA Estimation, FPGA

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Dr. Richard K. Martin (ENG)
U	U	U	UU	113	19b. TELEPHONE NUMBER *(include area code)* (937) 255-3636 x4625 richard.martin@afit.edu