

## ABSTRACT

Title of Thesis: UPLINK CHANNEL ESTIMATION IN WIMAX  
Kenneth T. Ho, Master of Science, 2008

Directed By: Associate Professor, Min Wu  
Department of Electrical and Computer Engineering

Over the last 20 years, we have seen a tremendous increase in demand for broadband internet access and cellular use. Many technological advances were made throughout this time period to increase internet access speeds and improve the voice service quality. But despite these advances, one of the current limitations in the current technology is that it cannot simultaneously provide both high-speed internet access equivalent to cable/DSL and coverage equivalent to the cellular networks. To overcome this limitation, WiMAX was developed to be the solution to this limitation. WiMAX is an emerging and powerful technology in broadband wireless communications in that it can provide high-speed broadband voice and data services over distances much greater than Wi-Fi.

However, like in any wireless system, signal distortion due to channel fading, noise, and Doppler can limit the overall transmission data rate and coverage. To minimize the degradation in system performance caused by the channel, channel estimation must be performed to remove the effects of the channel. To aid the channel estimation process, known pilot subcarriers are embedded into each OFDM symbol and used by the receiver to measure the channel. Because the pilot arrangement

depends on the mode or feature employed by WiMAX, different channel estimation algorithms must be developed to optimize performance.

We first develop and present several channel estimation algorithms for the various modes and features supported by WiMAX. More specifically, we develop channel estimation algorithms that can be used in the PUSC and AMC subcarrier permutation and channel sounding feature. We then use analytical modeling and simulation to illustrate and analyze the performance of each algorithm under various channel conditions and compare them to an estimator with perfect channel knowledge. Our results show that channel delay spread and signal-to-noise ratio influence the performance of each algorithm and that some estimators perform better in certain channels and worse in others. With this knowledge, we can use a combination of our proposed algorithms and tune them according to the channel conditions to enhance the system's performance. By the end of this study, we will have a better understanding of the types of channels each algorithm performs best in and explain why each algorithm performs the way it does.

# UPLINK CHANNEL ESTIMATION IN WIMAX

by

Kenneth T. Ho

Thesis submitted to the Faculty of the Graduate School of the  
University of Maryland, College Park in partial fulfillment  
of the requirements for the degree of  
Master of Science  
2008

Advisory Committee:

Associate Professor Min Wu, Advisor

Associate Professor Steven Tretter

Associate Professor Sennur Ulukus

© Copyright by  
Kenneth T. Ho  
2008

## ACKNOWLEDGEMENTS

First, I would like to thank Texas Instruments in providing me the invaluable opportunity to be a Texas Instruments Scholar and work on challenging projects and learn from experienced professionals. In particular, I would to give a special thanks to Brian Johnson and his team for their patience and guidance throughout my research. It was a privilege to work with and learn from such talented people and without them, this thesis would not have been possible.

I also have to thank my advisor, Dr. Min Wu, for mentoring me as both a graduate and undergraduate student. She has always believed in me and challenged me to get the most out of myself and I would not be where I am today without her help. I would also like to thank Dr. Steven Tretter for his continual help and support throughout my research. The time he spent with me to discuss ideas and answer my questions were invaluable. I also would like to express my sincere appreciation to Dr. Sennur Ulukus for serving on my thesis committee.

Finally, I also owe a deep amount of gratitude to my mom and dad who have always been there for me and kept me headed in the right direction. I feel blessed to have them as parents because their unconditional support has helped get through some of the toughest times. In addition, I would also like to acknowledge all my friends for their help and support throughout my life and to let them know that everything they have done for me is greatly appreciated.

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Overview of Uplink Channel Estimation . . . . .	2
1.2	Background: OFDM Channel Estimation . . . . .	5
1.2.1	OFDM Blind Channel Estimation . . . . .	5
1.2.2	OFDM Pilot-Assisted Channel Estimation . . . . .	6
1.3	WiMAX Channel Estimation . . . . .	9
1.4	Contributions . . . . .	10
1.5	Outline . . . . .	12
<b>2</b>	<b>System Overview</b>	<b>13</b>
2.1	Introduction to WiMAX . . . . .	13
2.1.1	Benefits to WiMAX . . . . .	14
2.1.2	Basic OFDM/OFDMA Principles . . . . .	17
2.1.3	Subcarrier Structure . . . . .	21
2.1.4	Frame Structure . . . . .	21
2.1.5	Subcarrier Permutations . . . . .	23
2.1.6	Uplink Channel Sounding . . . . .	26
2.1.7	System Profiles . . . . .	26
2.2	Channel Model . . . . .	28
2.2.1	Multipath . . . . .	28
2.2.2	Doppler Spread . . . . .	30
2.2.3	Channel Impulse Response Model . . . . .	31
2.2.4	SUI and ITUR Channel Models . . . . .	33
2.2.5	Noise . . . . .	34
2.3	System Model . . . . .	34
<b>3</b>	<b>Channel Estimation in PUSC (Partial Usage of Subcarrier)</b>	<b>37</b>
3.1	Introduction . . . . .	37
3.2	Channel Estimation Algorithms . . . . .	38
3.2.1	Linear Interpolation . . . . .	38
3.2.2	4-pilot Averaging . . . . .	43
3.2.3	Genie-Aided Channel Estimator (Perfect Channel Knowledge)	46
3.3	Simulation . . . . .	47
3.3.1	Objective . . . . .	47
3.3.2	Simulation Parameters . . . . .	47

3.3.3	Linear Interpolation vs. Genie-Aided Estimator . . . . .	50
3.3.4	Linear Interpolation vs. 4-pilot Averaging . . . . .	53
3.4	Conclusions . . . . .	59
<b>4</b>	<b>Channel Estimation in AMC (Adaptive Modulation and Coding)</b>	<b>60</b>
4.1	Introduction . . . . .	60
4.2	Channel Estimation Algorithms . . . . .	61
4.2.1	Linear Interpolation . . . . .	62
4.2.2	Frequency Smoothing . . . . .	65
4.3	Simulation . . . . .	69
4.3.1	Objective . . . . .	69
4.3.2	Simulation Parameters . . . . .	70
4.3.3	Results: Mean-Squared Error . . . . .	72
4.3.4	Observations: Mean-Squared Error . . . . .	75
4.3.5	Results: Error Vector Magnitude . . . . .	76
4.4	Conclusion . . . . .	84
<b>5</b>	<b>Channel Estimation in Channel Sounding</b>	<b>85</b>
5.1	Introduction . . . . .	85
5.2	Channel Estimation in a Subset of Sounding Zone . . . . .	87
5.2.1	Frequency Smoothing . . . . .	87
5.2.2	Linear Minimum Mean-Squared Error (LMMSE) Estimator . . . . .	87
5.2.3	Simulation: Subset of Sounding Zone . . . . .	91
5.3	Channel Estimation with Decimation Separability . . . . .	105
5.3.1	Linear Interpolation . . . . .	106
5.3.2	Frequency Smoothing . . . . .	113
5.3.3	Simulation: Decimation Separability . . . . .	113
5.4	Channel Estimation with Cyclic-Shift Separability . . . . .	121
5.4.1	Staircasing Channel Estimation . . . . .	122
5.4.2	Frequency Smoothing . . . . .	127
5.4.3	Simulation: Cyclic-Shift Separability . . . . .	127
5.5	Decimation Separability vs. Cyclic-Shift Separability . . . . .	134
5.6	Conclusion . . . . .	135
<b>6</b>	<b>Conclusion</b>	<b>138</b>
6.1	Future Work . . . . .	140
<b>A</b>	<b>SUI and ITUR Channel Models</b>	<b>141</b>
<b>B</b>	<b>Impact of Frequency Smoothing Window on Channel Estimates</b>	<b>145</b>
<b>C</b>	<b>Effect of Using IDFT/DFT Larger than Number of Occupied Sub-carriers</b>	<b>150</b>

<b>D</b>	<b>Mean-Squared Error Derivations</b>	<b>153</b>
D.1	PUSC: Linear Interpolation . . . . .	153
D.2	PUSC: 4-Pilot Averaging . . . . .	156
D.3	AMC and Channel Sounding: Linear Interpolation . . . . .	158

# List of Tables

2.1	WiMAX OFDMA Symbol Parameters [25]	27
2.2	WiMAX supported Modulation and Code Rates [13]	28
3.1	Simulation Parameters for PUSC Channel Estimation Evaluation	49
4.1	Simulation Parameters for AMC Channel Estimation Evaluation	71
5.1	Channel Sounding Simulation Parameters	93
5.2	Channel Estimation Parameters	94
5.3	Channel Sounding Simulation Parameters: Decimation Separability	115
5.4	Channel Sounding Simulation Parameters: Cyclic Shift Separability	129
A.1	ITU-R: Pedestrian Channel Model Parameters	142
A.2	ITU-R: Vehicular Channel Model Parameters	143
A.3	SUI Channel Model Parameters	144
B.1	Window Length Calculations	146

# List of Figures

2.1	WiMAX System Level Block Diagram . . . . .	14
2.2	WiMAX OFDM Baseband System Model . . . . .	17
2.3	OFDM Spectrum with 7 Orthogonal Subcarriers . . . . .	19
2.4	Function of the Cyclic Prefix . . . . .	20
2.5	WiMAX Frame Structure [13] . . . . .	22
2.6	Illustration of Distributed and Adjacent Subcarrier Permutation . . . . .	23
2.7	PUSC Tile and Permutation Process . . . . .	25
2.8	AMC Subchannel . . . . .	25
3.1	PUSC Tile . . . . .	38
3.2	Average MSE Over PUSC tile using Linear Interpolation . . . . .	42
3.3	Average MSE Over PUSC tile using 4-Pilot Averaging . . . . .	45
3.4	Genie-aided Channel Estimator . . . . .	46
3.5	BER curves of Linear Interpolator vs. Genie-Aided Channel Estimator in Ped-A, 3km/hr model . . . . .	50
3.6	BER curves of Linear Interpolator vs. Genie-Aided Channel Estimator in Ped-A, 120km/hr model . . . . .	51
3.7	BER curves of Linear Interpolator vs. Genie-Aided Channel Estimator in Veh-B, 3km/hr model . . . . .	51
3.8	BER curves of Linear Interpolator vs. Genie-Aided Channel Estimator in Veh-B, 120km/hr model . . . . .	52

3.9	BER curves of Linear Interpolator vs. 4-pilot Averaging in Ped-A, 3km/hr model . . . . .	54
3.10	BER curves of Linear Interpolator vs. 4-pilot Averaging in Ped-A, 120km/hr model . . . . .	55
3.11	BER curves of Linear Interpolator vs. 4-pilot Averaging in Veh-B, 3km/hr model . . . . .	55
3.12	BER curves of Linear Interpolator vs. 4-pilot Averaging in Veh-B, 120km/hr model . . . . .	56
3.13	Average MSE Over PUSC tile: Linear Interpolation and 4-pilot Averaging . . . . .	58
4.1	AMC Subchannel . . . . .	61
4.2	Arrangement of Pilots in one OFDM symbol within one AMC subchannel under assumption that channel remains constant over 3 OFDM symbols . . . . .	62
4.3	Average MSE over 1 AMC Subchannel ( $M = 18$ ) using Linear Interpolation . . . . .	65
4.4	Block Diagram of WiMAX with Frequency Smoothing Channel Estimation Algorithm . . . . .	67
4.5	WiMAX Matlab Simulator for AMC and Channel Sounding Simulations	70
4.6	MSE, 1 AMC Subchannel ( $M=18,36$ ), SUI-1 . . . . .	72
4.7	MSE, 1 AMC Subchannel ( $M=18,36$ ), SUI-4 . . . . .	73
4.8	MSE, 1 AMC Subchannel ( $M=18,36$ ), SUI-5 . . . . .	73
4.9	MSE, 1 AMC Subchannel ( $M=18,36$ ), Ped-A . . . . .	74
4.10	MSE, 1 AMC Subchannel ( $M=18,36$ ), Veh-B . . . . .	74
4.11	Error Vector Magnitude, SNR = -10dB, 1 AMC Subchannel ( $M = 18,36$ Subcarriers), SUI-1 . . . . .	77
4.12	Error Vector Magnitude, SNR = 20dB, 1 AMC Subchannel ( $M = 18,36$ Subcarriers), SUI-1 . . . . .	78
4.13	Error Vector Magnitude, SNR = -10dB, 1 AMC Subchannel ( $M = 18,36$ Subcarriers), SUI-4 . . . . .	78

4.14	Error Vector Magnitude, SNR = 20dB, 1 AMC Subchannel (M = 18,36 Subcarriers), SUI-4 . . . . .	79
4.15	Error Vector Magnitude, SNR = -10dB, 1 AMC Subchannel (M = 18,36 Subcarriers), SUI-5 . . . . .	79
4.16	Error Vector Magnitude, SNR = 20dB, 1 AMC Subchannel (M = 18,36 Subcarriers), SUI-5 . . . . .	80
4.17	Error Vector Magnitude, SNR = -10dB, 1 AMC Subchannel (M = 18,36 Subcarriers), Ped-A . . . . .	80
4.18	Error Vector Magnitude, SNR = 20dB, 1 AMC Subchannel (M = 18,36 Subcarriers), Ped-A . . . . .	81
4.19	Error Vector Magnitude, SNR = -10dB, 1 AMC Subchannel (M = 18,36 Subcarriers), Veh-B . . . . .	81
4.20	Error Vector Magnitude, SNR = 20dB, 1 AMC Subchannel (M = 18,36 Subcarriers), Veh-B . . . . .	82
5.1	Frame Sequence of Channel Sounding [16] . . . . .	85
5.2	Block Diagram of WiMAX with LMMSE channel estimator . . . . .	88
5.3	Average MSE in Channel Sounding in Subset of Sounding Zone (M = 72), Frequency Smoothing vs. LMMSE, SUI-1 . . . . .	95
5.4	Average MSE in Channel Sounding in Subset of Sounding Zone (M = 72), Frequency Smoothing vs. LMMSE, SUI-4 . . . . .	96
5.5	Average MSE in Channel Sounding in Subset of Sounding Zone (M = 72), Frequency Smoothing vs. LMMSE, SUI-5 . . . . .	96
5.6	Average MSE in Channel Sounding in Subset of Sounding Zone, Frequency Smoothing vs. LMMSE, (M = 72), Ped-A . . . . .	97
5.7	Average MSE in Channel Sounding in Subset of Sounding Zone (M = 72), Frequency Smoothing vs. LMMSE, Veh-B . . . . .	97
5.8	Error Vector Magnitude of Frequency Smoothing vs. LMMSE, SNR = 20dB, M = 72, SUI-1 . . . . .	101
5.9	Error Vector Magnitude of Frequency Smoothing vs. LMMSE, SNR = 20dB, M = 72, SUI-4 . . . . .	101

5.10	Error Vector Magnitude of Frequency Smoothing vs. LMMSE, SNR = 20dB, M = 72, SUI-5 . . . . .	102
5.11	Error Vector Magnitude of Frequency Smoothing vs. LMMSE, SNR = 20dB, M = 72, Ped-A . . . . .	102
5.12	Error Vector Magnitude of Frequency Smoothing vs. LMMSE, SNR = 20dB, M = 72, Veh-B . . . . .	103
5.13	Decimation Separability Example (D = 4) . . . . .	105
5.14	Linear Interpolation Average MSE over Sounding Zone With Decimation Separability, SUI-1 . . . . .	109
5.15	Linear Interpolation Average MSE over Sounding Zone With Decimation Separability, SUI-4 . . . . .	109
5.16	Linear Interpolation Average MSE over Sounding Zone With Decimation Separability, SUI-5 . . . . .	110
5.17	Linear Interpolation Average MSE over Sounding Zone With Decimation Separability, Ped-A . . . . .	110
5.18	Linear Interpolation Average MSE over Sounding Zone With Decimation Separability, Veh-B . . . . .	111
5.19	Frequency-Domain Channel Autocorrelation Function . . . . .	112
5.20	MSE, Channel Sounding with Decimation Separability, SUI-1 . . . . .	116
5.21	MSE, Channel Sounding with Decimation Separability, SUI-4 . . . . .	117
5.22	MSE, Channel Sounding with Decimation Separability, SUI-5 . . . . .	117
5.23	MSE, Channel Sounding with Decimation Separability, Ped-A . . . . .	118
5.24	MSE, Channel Sounding with Decimation Separability, Veh-B . . . . .	118
5.25	Cyclic Shift Separability . . . . .	121
5.26	Average MSE over Sounding Zone With Cyclic-Shift Separability, SUI-1 . . . . .	124
5.27	Average MSE over Sounding Zone With Cyclic-Shift Separability, SUI-4 . . . . .	124
5.28	Average MSE over Sounding Zone With Cyclic-Shift Separability, SUI-5 . . . . .	125

5.29	Average MSE over Sounding Zone With Cyclic-Shift Separability, Ped-A . . . . .	125
5.30	Average MSE over Sounding Zone With Cyclic-Shift Separability, Veh-B . . . . .	126
5.31	MSE, Channel Sounding with Cyclic Shift Separability, SUI-1 . . . .	130
5.32	MSE, Channel Sounding with Cyclic Shift Separability, SUI-4 . . . .	131
5.33	MSE, Channel Sounding with Cyclic Shift Separability, SUI-5 . . . .	131
5.34	MSE, Channel Sounding with Cyclic Shift Separability, Ped-A . . . .	132
5.35	MSE, Channel Sounding with Cyclic Shift Separability, Veh-B . . . .	132
B.1	Time-Domain Response after M-point IDFT in Frequency Smoothing Algorithm, No Noise, SUI-1 . . . . .	147
B.2	Time-Domain Response after M-point IDFT in Frequency Smoothing Algorithm, No Noise, SUI-4 . . . . .	147
B.3	Time-Domain Response after M-point IDFT in Frequency Smoothing Algorithm, No Noise, SUI-5 . . . . .	148
B.4	Time-Domain Response after M-point IDFT in Frequency Smoothing Algorithm, No Noise, Ped-A . . . . .	148
B.5	Time-Domain Response after M-point IDFT in Frequency Smoothing Algorithm, No Noise, Veh-B . . . . .	149

# Chapter 1

## Introduction

Over the last 20 years, there has been a tremendous growth in demand for broadband internet access and cellular use. Broadband internet access and cellular use have grown to the point that it has become much a part of daily life that most people cannot live without now. At the same time, we have also seen the requirements for the technology providing these services increase in order to provide users with a better overall experience. The technology for internet access has developed from low dial-up speeds on the order of kilobits per second to high broadband cable/DSL speeds on the order of megabits per second. In addition, the development of Wi-Fi has allowed broadband internet access to be provided to users within a small area. In parallel, cellular technology has also developed rapidly from 1G analog networks to 2G digital networks to 3G networks which provide both voice and low-speed data services.

However, one of the limitations to the current technology is that it does not allow users to have high speed broadband internet access and be completely mobile

simultaneously. Cable/DSL forces a user to be fixed to obtain broadband internet access and Wi-Fi only provides broadband internet coverage to those within a radius of a couple hundred meters which severely restricts the amount a user can move. The 3G cellular network allows a user to be mobile but was designed more for voice services and can only provide data services at speeds equivalent to dial-up which could not support most multimedia applications. To overcome this limitation, WiMAX was developed to allow users to have high-speed broadband internet access at speeds equivalent to cable/DSL and the same mobility provided by the cellular network simultaneously. WiMAX is a broadband wireless network that combines the fixed broadband and mobile cellular network into one flexible and easily deployable network.

## 1.1 Overview of Uplink Channel Estimation

A problem with any communication system is that its data rate can be limited by distortion caused by the channel. To mitigate the effect of the channel, the receiver must perform channel estimation to remove this distortion. The task of channel estimation is made more difficult in a wireless environment because the channel is fast-changing and unpredictable. Because of this, good channel estimators are required to accurately estimate the channel while minimizing the amount of time, data, and computations necessary to do so.

Recently, we have seen an increased popularity in OFDM-based (Orthogonal Frequency Division Multiplexing) broadband wireless communication systems

because of its high spectral efficiency and robustness to multipath channel fading. OFDM is a multicarrier modulation scheme in which data is modulated onto orthogonal subcarriers spaced throughout the channel bandwidth. In an OFDM system, all the subcarriers are simultaneously transmitted so we can model the channel bandwidth as a sum of several narrowband subchannels in which the channel response within each subchannel can be modeled as a flat fading channel. This allows a simple frequency-domain equalizer to be implemented and hence, significantly reduce the computational complexity in channel estimation.

WiMAX is also an OFDM-based system, but unique in that it employs OFDMA (Orthogonal Frequency Division Multiple Access) which is the multi-user version of OFDM. The difference between WiMAX and other OFDM systems is that it allows multiple users to simultaneously access the channel by allocating a different set of subchannels to each user. This extra degree of freedom allows WiMAX to exploit the frequency-selective channel by allocating subchannels to users with favorable conditions in those subchannels and/or avoid allocating subchannels to users in which their channel conditions are poor. However, this unique feature also makes channel estimation in WiMAX more difficult because fewer subcarriers can be used in estimating the channel.

To allow multiple access to the channel, WiMAX uses two types of subcarrier allocation modes or permutations: Distributed and Adjacent. A distributed subcarrier permutation pseudorandomly allocates subcarriers to subchannels to exploit frequency diversity. Partial Usage of Subcarriers (PUSC) is one example of a distributed subcarrier permutation. An adjacent subcarrier permutation forms

subchannels of adjacent subcarriers and leaves the responsibility of determining the optimum allocation to the scheduler. Adaptive Modulation and Coding (AMC) is one example of an adjacent subcarrier permutation. In AMC, channel sounding is an optional feature sometimes used to aid the base station in determining each user's unique channel condition. This technique is implemented by reserving OFDM symbols typically at the end of the uplink frame for users to transmit known sounding sequences for the base station to use to estimate the channel of each user in the system.

A specific pilot arrangement is defined by the IEEE 802.16 standard for each permutation mode supported by WiMAX so our primary focus in this thesis will be on channel estimation and interpolation algorithms for WiMAX. Mainly, our emphasis will be on channel estimation and interpolation algorithms that can be used for the pilot allocation defined in PUSC and AMC. We will also extend our analysis to channel estimation algorithms in channel sounding where users may transmit known signals in any number of subcarriers within an OFDM symbol. In addition, we will focus our work on channel estimation on the uplink because this is generally the more difficult problem to solve. On the uplink, the received signal is distorted by multiple frequency selective channels so the number of subcarriers that can be used for channel estimation is smaller than on the downlink.

## 1.2 Background: OFDM Channel Estimation

Channel estimation has received a lot of attention in recent years because of its importance in wireless communications. Without a good channel estimator, the throughput and coverage of the a wireless system is severely limited. OFDM has become very popular in wireless communication systems including WiMAX because of its robustness to multipath channel fading and simple channel equalizer so a lot of research work has been placed in designing a good channel estimator for OFDM systems.

### 1.2.1 OFDM Blind Channel Estimation

Channel estimators for OFDM systems can usually be classified as blind or pilot-assisted. In blind channel estimation, the receiver attempts to estimate the channel through the use of higher order statistics without the aid of any training signals. These estimators use an iterative algorithm based on higher order statistics to converge to the optimal solution. Muquet explored a blind channel estimation method based on a subspace approach by utilizing the cyclic prefix [6]. He also presents a semi-blind approach to his estimator to improve performance. While his approach shows that good performance can be obtained, it requires between 50 to 100 OFDM symbols before converging to the optimal solution. In [1], he presents a two-stage blind channel equalization approach in which it first uses second-order cyclostationary statistics to get a rough estimate of the channel and then use the Constant Modulus Algorithm (CMA) to adapt to the channel.

## 1.2.2 OFDM Pilot-Assisted Channel Estimation

While the blind channel estimation methods present above show that a high-performing solution can be obtained, their convergence rate is often too slow for most wireless communication systems. In most scenarios, the wireless channel will have changed before many blind channel estimators will have converged to the optimal solution so many wireless communication systems employ pilot-assisted channel estimators. In pilot-assisted channel estimation, some of the channel bandwidth is reserved for transmitting pilot (training) signals to the receiver and using them to estimate the channel. By sacrificing some bandwidth efficiency to transmit pilot signals, the pilot-assisted channel estimators can typically make fast and accurate measurements of the channel. With pilot-assisted channel estimators, there are generally two often researched topics: Pilot Allocation and Estimation/Interpolation algorithms.

### Pilot Arrangement

In [10], some commonly used pilot patterns in OFDM are presented. Among those that are presented are a Block-type arrangement in which one symbol is periodically reserved for just pilot transmission, a Comb-type arrangement in which a few subcarriers are allocated for pilot transmission in every OFDM symbol, and a rectangular, parallelogram-shaped and hexagonal grid arrangement in which pilots are distributed in time and frequency according to their respective geometry. In [2], Garcia shows that a hexagonal pilot arrangement performs better than the

rectangular arrangement in terms of bit error rate. In [3], Negi shows that a pilot arrangement distributed among subcarriers and symbols is better than grouping them all together in one symbol in the mean-squared error sense. And in [4], Zhang derives a theoretical BER and uses it to determine the optimal pilot arrangement. In his approach, he proposes to cluster two pilots together and increase the spacing between each cluster rather than an evenly distribute the pilots. He shows that this method has better noise reduction features and would be better suited in low SNR environments. But generally speaking, the optimal number of pilots will depend on the bandwidth efficiency that can be tolerated and the optimal pilot spacing will depend on the coherence bandwidth and coherence time of the channel.

### **Estimation/Interpolation**

In OFDM channel estimation, the first step is to perform estimation at the pilot subcarriers. Two common estimators for performing estimation at the pilot subcarriers are the Minimum Mean-Squared Error (MMSE) Estimator and the Least-Squares (LS) Estimator [10]. The MMSE estimator is one that depends on second-order channel statistics and can produce a very accurate estimate when the channel statistics are known. However, this estimator can be computationally complex because it depends on a matrix inversion and requires knowledge of the channel to be effective. The LS estimator is simple, low-complexity alternative to the MMSE, but cannot achieve the level of performance the MMSE estimator can. The LS estimator performs well in high SNR environments but quickly degrades as the noise level increases.

The next step in OFDM channel estimation is interpolation of the channel estimates to the data subcarriers. The classification of channel interpolators can be divided into adaptive and non-adaptive interpolators. Adaptive estimators are iterative algorithms that use training data to generate an estimate of the channel. Hou proposed an estimator based on the Least Mean Square (LMS) algorithm which uses the time and frequency domain correlation of the channel to adaptively track the channel [7]. Saeed compares a time-domain and frequency-domain Recursive Least Squares (RLS) estimator to iteratively estimate the channel [8]. Chen presents an iterative algorithm based on Kalman filtering by using Jake's channel model to model the channel's correlation in time [9].

Two common non-adaptive interpolators used in OFDM are a Polynomial-Based Interpolator [10] and a Frequency Smoothing/DFT-based Estimator [5]. The Polynomial-Based Interpolator are popular mainly for the fact that they are easy to implement and have low computational complexity. But these estimators tend to be sensitive to noise and depend much on the accuracy of the channel estimate at the pilots. The Frequency Smoothing approach is another interpolation approach in which the channel estimates at the pilots are transformed into the time domain where a window is applied and then transformed back into the frequency domain. This estimator is equivalent to applying a low-pass filter in the time-domain. The frequency smoothing algorithm interpolates the data subcarriers in the frequency domain by eliminating the images in the time-domain. The benefit to this approach is that this method has very good noise rejection capabilities and can perform well even without knowledge of the channel. However, this approach is more computa-

tionally complex because an IDFT and DFT must be used and requires more pilots that are evenly spaced to be effective.

### 1.3 WiMAX Channel Estimation

While adaptive algorithms can converge to a solution sufficiently fast even for a wireless channel, these algorithms cannot be applied for WiMAX because users are not guaranteed to occupy the same subcarriers. Because WiMAX utilizes different permutations, users may be constantly assigned different portions of the channel bandwidth so an adaptive algorithm cannot be considered for WiMAX.

WiMAX supports a wide variety of applications ranging from voice services that are latency-sensitive to real-time multimedia services where a consistent high data throughput must be maintained. In addition, users may be using any one of these services while walking or riding in a car where the wireless channel appears to be fast-changing to the receiver. Because of these requirements, it is particularly important that a WiMAX channel estimator be designed to be both fast and accurate. And while WiMAX is an OFDM-based system, channel estimation is more difficult in WiMAX than traditional OFDM systems because the estimator has fewer subcarriers to use. In addition, we generally cannot consider adaptive algorithms for channel estimation in WiMAX because users constantly assigned different portions of the channel bandwidth. In this thesis, we will design and analyze channel estimators that satisfy these constraints. More specifically, we will look at non-adaptive estimators and interpolators that are accurate and computationally fast for given

pilot allocations as defined in WiMAX permutation modes.

## 1.4 Contributions

While studies of uplink channel estimation algorithms for OFDM have been done before, none have been done specifically for WiMAX and its subcarrier permutation modes and channel sounding feature. In this thesis, we will develop, demonstrate, and analyze the performance of many uplink channel estimation algorithms that can be used for WiMAX. We focus our study on algorithms that can be used in PUSC (distributed subcarrier permutation) and AMC (adjacent subcarrier permutation) because the IEEE 802.16 standard requires support of these permutations. In addition, we also include a study of channel estimation algorithms for channel sounding, an optional feature in WiMAX typically used with AMC. More specifically, the main contributions of this thesis are as follows:

1. We illustrate and evaluate the performance of the linear interpolator in PUSC, AMC, and channel sounding and compare its performance to the genie-aided channel estimator (perfect channel knowledge). In addition, we include an analytical model of the linear interpolator that can be used to predict the performance of the estimator in different channel conditions and compare its results to our simulations results.
2. We develop and evaluate the performance of the 4-pilot averaging estimator in PUSC and compare its performance to the linear interpolator. We also include

an analytical model of the 4-pilot averaging estimator and compare the results predicted by the model to our simulation results.

3. We develop and evaluate the performance of a frequency smoothing algorithm in AMC and channel sounding and compare its performance to other channel estimators under different channel conditions. We also include an analytical model of the frequency smoothing algorithm and compare its results to our simulation results.
4. We develop and evaluate the performance of the linear minimum mean-squared error (LMMSE) estimator in channel sounding for different parameters and compare its results other algorithms. In addition, we include an analytical model of the LMMSE estimator and compare its results to the simulation generated results.
5. We illustrate the performance of the linear interpolator and frequency smoothing algorithm when users perform channel sounding over the entire channel allocation and are multiplexed within the sounding zone through decimation separability.
6. We illustrate the performance of the staircasing method and frequency smoothing algorithm when users perform channel sounding over the entire channel allocation and are multiplexed within the sounding zone through cyclic-shift separability.

## 1.5 Outline

The organization of this Master's thesis is as follows:

- In Chapter 2, we provide a more detailed description of WiMAX and some of the theory behind OFDM. In addition, we describe the system model used throughout this study and explain how the model was developed.
- In Chapter 3, we illustrate and analyze each channel estimator's performance in PUSC. We demonstrate and analyze the system level performance gap between the current linear interpolator with the genie-aided estimator. In addition, we introduce a 4-pilot averaging channel estimation technique and compare its performance to the linear interpolator under different channel conditions in PUSC.
- In Chapter 4, we illustrate and analyze each channel estimator's performance in AMC. We compare the performance of the linear interpolator to the frequency smoothing algorithm and analyze each under different channel conditions.
- In Chapter 5, we demonstrate the performance of each channel estimation algorithm in channel sounding. We introduce and show the performance of channel estimation algorithms when users transmit sounding signals over a part of or over the entire sounding zone.
- In Chapter 6, we summarize the conclusions made from this study and briefly discuss related areas where future work can be performed.

# Chapter 2

## System Overview

### 2.1 Introduction to WiMAX

WiMAX (Worldwide Interoperability for Microwave ACCess) is an emerging technology in wireless communications and will play a major role in broadband wireless metropolitan networks. WiMAX stands for Worldwide Interoperability for Microwave Access and is based on the IEEE 802.16 (Wireless MAN) standard. When the IEEE 802.16 standard was initially released in 2001, the only specifications defined were for the 10-66GHz range and targeted wireless networks where line of sight was present. However, in 2004, the specification was amended to revision D (IEEE 802.16d) to include the 2-11GHz frequency band where fixed and low mobility environments could be supported. In 2005, an amendment was added and the IEEE 802.16e standard was created to support full mobility and included features such as MIMO and Scalable OFDMA. In general, WiMAX refers to networks that meet specifications in the IEEE 802.16d and/or IEEE 802.16e revision of the standard.

WiMAX defines “profiles” which are composed of a subset specifications from the 802.16d and 802.16e standard that vendors can use to certify their products.

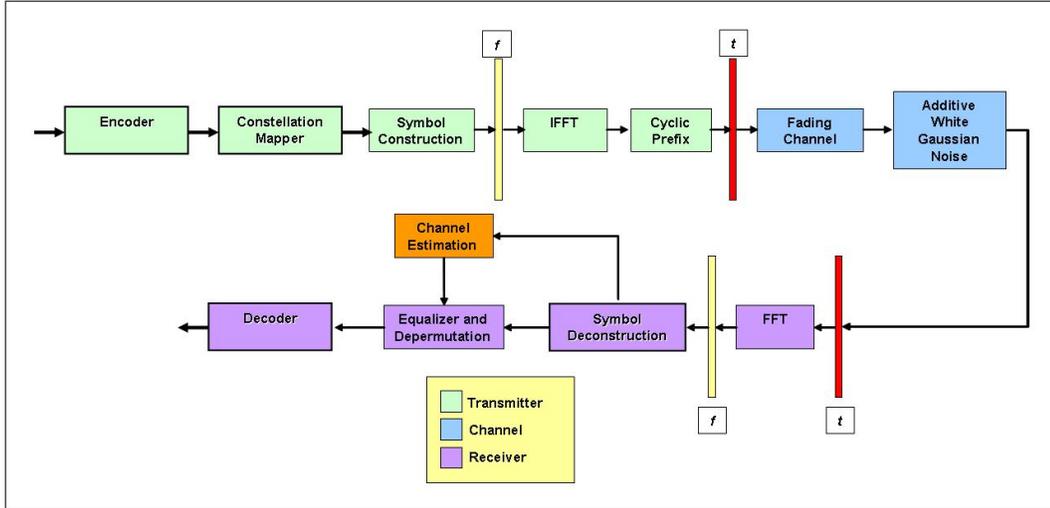


Figure 2.1: WiMAX System Level Block Diagram

### 2.1.1 Benefits to WiMAX

WiMAX has many features that make it one of the more attractive technologies to those seeking to deploy high-speed and low cost broadband wireless networks. From a marketing perspective, the benefits to WiMAX are [26]:

- *High-Speed and Large Coverage Simultaneously* - WiMAX provides high-speed voice, data, and multimedia service at speeds equivalent to cable and DSL while still offering the user the mobility equivalent allowed by the cellular network. Wi-Fi can only provide broadband internet access within a few hundred meters and the cellular network can only offer data rates equivalent to dial-up.

- *Solution to the “Last Mile Problem”* - One of biggest barriers to providing broadband internet access is deployment cost. Many residents and businesses residing in rural areas or environments with a limited wired infrastructure do not have broadband internet access because it is not cost effective to service providers. In these areas, the gain to deploying a broadband network does not outweigh the cost of digging trenches and deploying wires for a small number of users. However, with WiMAX, a single base station could provide broadband internet access to all users residing within 30 kilometers of the base station and greatly reduce the cost of providing broadband internet access.
- *Backhaul to Wi-Fi hotspots and cellular networks* - WiMAX can support large capacities and provide wide coverage which make it capable of being used for backhauling purposes. In addition, because WiMAX can be rapidly and cost-effectively deployed, service providers can use WiMAX to quickly connect new Wi-Fi hotspots to the Internet and new cell towers to their existing cellular network.
- *Connect businesses that are scattered within a metropolitan area* - Businesses with many locations scattered throughout a metropolitan area can be interconnected through WiMAX. Using WiMAX for this purpose eliminates the need for each of these locations to be geographically close to each other and reduces the cost and time required to deploy a wired infrastructure to connect each site.

The advanced technological infrastructure of WiMAX allows it to be more efficient and cost-effective solution to providing high-speed voice, data, and multimedia services. From a technological perspective, the benefits to WiMAX are:

- *Adopts OFDMA* - At the physical layer, WiMAX employs OFDMA which is the multiuser version of OFDM. An OFDM system is more spectrally efficient because the subcarrier spacing is smaller than traditional FDM systems. An OFDM system is also more robust to multipath channel fading because the symbol duration is longer and simplifies channel equalization because the entire channel allocation is divided into many narrowband subchannels.
- *Scalable Channel Bandwidths* - WiMAX can easily be adjusted to support a wide range of bandwidths allowing it to be deployed for many different spectrum allocation and usage requirements. For example, in rural areas where broadband access may not exist and wireless spectrum is more readily available, WiMAX can be deployed with the widest bandwidth to support as many users as possible. However, in environments where other broadband networks exist and spectrum is more limited, a WiMAX network can be deployed with a smaller bandwidth to enhance capacity in the area.
- *Dynamic Resource Allocation* - At the MAC layer, resources can be allocated to users on a frame-by-frame basis based on their unique channel conditions and application requirements. This feature gives the system an extra degree of freedom to optimize system performance by allowing it to take advantage of the frequency selective fading channel.

- *Quality of Service (QoS) Support* - Because WiMAX can dynamically allocate resources, QoS requirements can be better enforced and guaranteed. This helps the system operate more efficiently and predictably because each application is guaranteed a certain level of performance and not allocated more or less than it needs.

### 2.1.2 Basic OFDM/OFDMA Principles

At the core of WiMAX is OFDMA which is an OFDM-based system that allows multiple users access to the allocated channel. In an OFDM system, a high-rate data stream is converted into several lower-rate data streams, mapped to a constellation, and then modulated onto orthogonal subcarriers. OFDM-based systems have been around for many years and until recently, each parallel data stream had to be modulated/demodulated separately. However, with advances in the digital signal processors, an efficient implementation of the IFFT/FFT operation can be used to perform modulation/demodulation respectively. A basic block diagram illustrating the WiMAX OFDM baseband system model is shown in Figure 2.2.

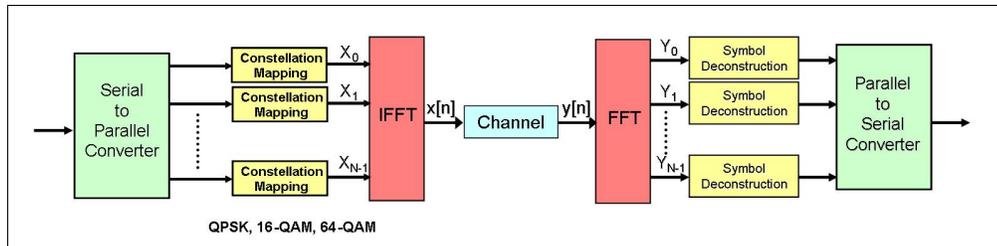


Figure 2.2: WiMAX OFDM Baseband System Model

## Orthogonality

In an OFDM system, the subcarrier frequencies are chosen so that the subcarriers are orthogonal to each other within each OFDM symbol. We define the  $k^{th}$  modulated subcarrier with frequency  $f_k$  as  $s_k(t) = g(t)\cos(2\pi f_k t)$  where  $k = 0, 1, 2, \dots, N-1$  and  $g(t)$  is the shaping pulse. For orthogonality,  $s_k(t)$  must be chosen such that

$$\frac{1}{T_s} \int_0^{T_s} s_i(t) \cdot s_j(t) dt = 0 \text{ for } i \neq j$$

where  $T_s$  is the symbol duration. If we choose the subcarrier spacing to be a multiple of  $1/T_s$ , then the subcarriers will be orthogonal to each other [28].

$$\begin{aligned} \frac{1}{T_s} \int_0^{T_s} s_i(t) \cdot s_j(t) dt &= \frac{1}{T_s} \int_0^{T_s} g(t) \cos(2\pi f_i t) \cdot g(t) \cos(2\pi f_j t) dt \\ &= \frac{1}{T_s} \int_0^{T_s} g(t) \cos\left(2\pi\left(f_0 + \frac{i}{T_s}\right)t\right) \cdot g(t) \cos\left(2\pi\left(f_0 + \frac{j}{T_s}\right)t\right) dt \\ &= \frac{1}{2T_s} \int_0^{T_s} g^2(t) \cos\left(2\pi\left(\frac{i-j}{T_s}\right)t\right) dt \\ &\quad + \frac{1}{2T_s} \int_0^{T_s} g^2(t) \cos\left(2\pi\left(2f_0 + \frac{i+j}{T_s}\right)t\right) dt \end{aligned}$$

If we assume that  $g(t)$  is a slowly varying pulse or approximately constant over the symbol duration, then the last line of the equation above is approximately 0. With this approximation, we obtain

$$\frac{1}{T_s} \int_0^{T_s} s_i(t) \cdot s_j(t) dt = \begin{cases} 0 & \text{for } i \neq j \\ \frac{1}{2} E_g & \text{for } i = j \end{cases}$$

where  $E_g$  is the energy of the shaping pulse.

Typically,  $g(t)$  is a raised cosine pulse. If we use this pulse shape with the orthogonal subcarriers as described above, then the subchannels can overlap without

interference. Figure 2.3 shows an example. This allows the subcarriers to be more closely spaced than traditional FDM systems and allows OFDM systems to achieve a high spectral efficiency. However, one of the weaknesses to an OFDM system is that these systems to be sensitive to any frequency deviation between the transmitter and receiver. Doppler and poor frequency synchronization between the transmitter and receiver could cause frequency deviation and introduce intercarrier interference (ICI) because the subcarriers lose orthogonality.

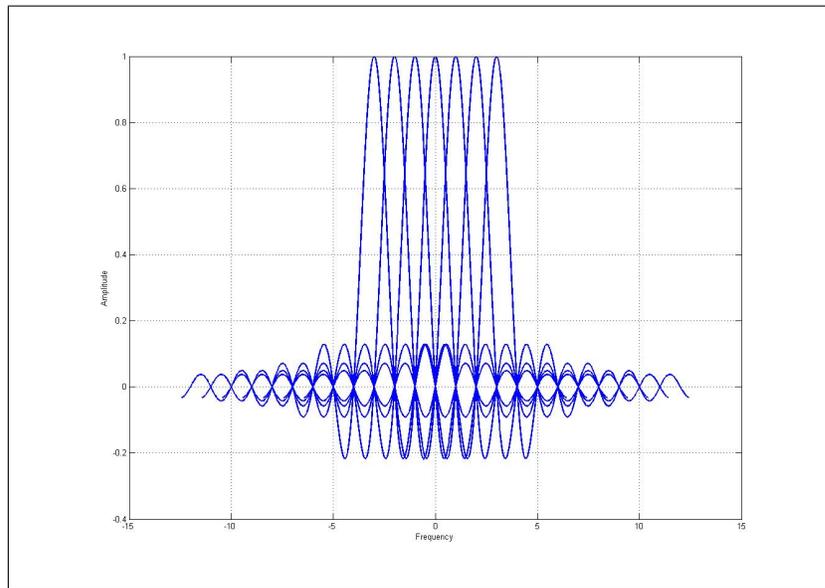


Figure 2.3: OFDM Spectrum with 7 Orthogonal Subcarriers

### Cyclic Prefix

One of the biggest advantages to an OFDM system is its ability to eliminate intersymbol interference (ISI). An OFDM system increases the symbol duration by transmitting several parallel data streams instead of a single high-rate data stream. Because the duration of each OFDM symbol is longer, a cyclic prefix can be used. A

cyclic prefix is the last portion of the original symbol duplicated and appended to the beginning of the OFDM symbol and removed once the receiver receives the symbol. If the cyclic prefix length is chosen such that it is longer than the maximum delay spread of the channel, then the system will be free of any intersymbol interference.

The cyclic prefix also makes the transmitted OFDM symbol appear periodic so the effect of the channel can be modeled with a circular convolution rather than a linear convolution. The benefit is that when the received signal is transformed into the frequency domain via the FFT operation, the effect of the channel becomes multiplicative. By combining this property with the assumption that the entire channel allocation can be viewed as a sum of many narrowband subchannels, we can use a simple frequency-domain equalizer to cancel out the effects of the channel at each subcarrier. Figure 2.4 illustrates the functionality of the cyclic prefix.

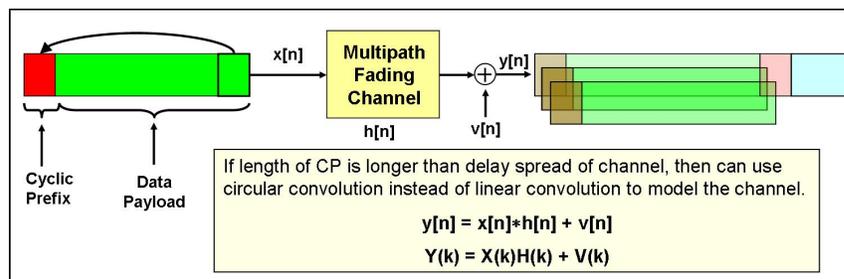


Figure 2.4: Function of the Cyclic Prefix

## Multuser Access

A unique feature of WiMAX is its use of the multiuser version of the OFDM. This feature allows multiple users to transmit simultaneously without interference since each user is allocated separate resources. Another benefit to multiple access

is that system parameters can be tuned to meet each user's Quality of Service (QoS) requirements. Each user will likely be using WiMAX for different applications and be transmitting in a different environment and this feature allows the system to have more control of the data rate for each user. However, this does make WiMAX more complex because algorithms adaptively assigning subcarriers need to be developed and coordination between other base stations is required to fully leverage this capability.

### 2.1.3 Subcarrier Structure

In WiMAX, each OFDM symbol is composed of data subcarriers, pilot subcarriers, and null subcarriers. Data subcarriers carry the actual data payload, pilot subcarriers are subcarriers with known values that are used in channel estimation, and null subcarriers are used as guard bands.

### 2.1.4 Frame Structure

Currently, WiMAX separates the uplink and downlink using time division duplexing (TDD). While other duplexing methods (Full and Half Duplex FDD) will be supported in the future, TDD provides the benefit of being able to adjust the data speed for the uplink and downlink depending on the amount of data that needs to be transferred.

In the downlink frame, some of the key components are [13]:

- *Preamble* - First symbol transmitted and used for synchronization purposes

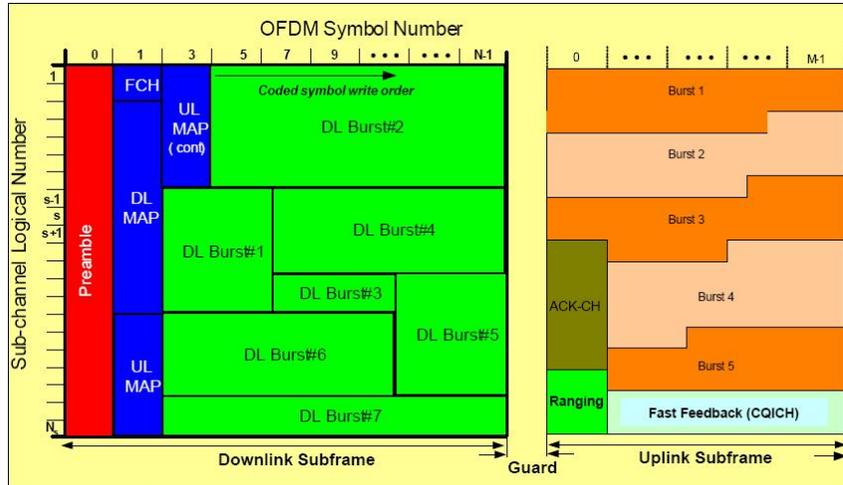


Figure 2.5: WiMAX Frame Structure [13]

- *Frame Control Header (FCH)* - Provides message lengths, coding schemes, and usable subchannels
- *DL/UL MAP* - Provides information about subchannel allocation

In the uplink frame, some of the key components are [13]:

- *Ranging* - Subchannels for mobiles to perform time, frequency, and power adjustments
- *CQICH (Channel Quality Indication)* - Subchannel used for mobiles to feedback channel state information to the base station. This information is used by the base station to determine the appropriate data rate for each user based on their channel conditions
- *ACK-CH* - Subchannels used by mobile to feedback DL hybrid ARQ (HARQ) information. HARQ is a feature in WiMAX that allows the system to respond quickly to packet errors.

## 2.1.5 Subcarrier Permutations

Subcarrier permutation describes the mapping of subcarriers to subchannels. These subchannels are then allocated to users in the system. WiMAX supports two types of subcarrier permutations: Distributed and adjacent. A distributed subcarrier permutation generally describes a pseudorandom allocation of subcarriers to subchannels and exploits a frequency selective channel through frequency diversity. The idea in a distributed subcarrier permutation is to use frequency hopping to help any single user avoid large bursts of errors. An adjacent subcarrier permutation describes a process of mapping adjacent subcarriers to form each subchannel and leaving the scheduler the responsibility of determining the optimal subchannel allocation based on each user's unique channel conditions. Ideally, if the scheduler knew each user's channel quality perfectly, it could allocate the high quality subchannels and/or avoid allocating the poor quality subchannels to each user. Figure 2.6 is an example of a simple 3 user system where the channel conditions of each user are shown and how each type of permutation would allocate subchannels to each user.

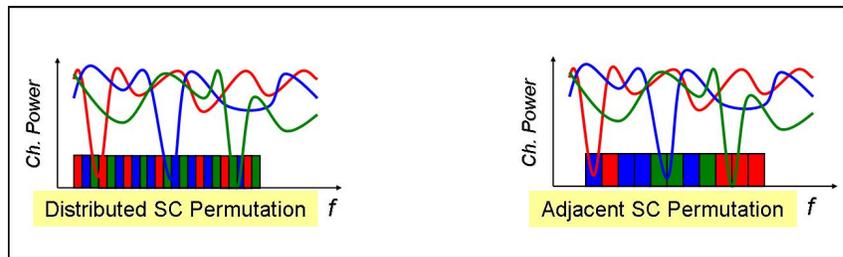


Figure 2.6: Illustration of Distributed and Adjacent Subcarrier Permutation

Currently, Partial Usage of Subcarriers (PUSC) is the distributed subcarrier

permutation and Adaptive Modulation and Coding (AMC) is the adjacent sub-carrier permutation WiMAX is required to support on the uplink. While other permutations will be supported later, we focus this study on WiMAX systems using PUSC and AMC.

### **PUSC - Distributed Subcarrier Permutation**

For PUSC, the mapping of subcarriers to subchannels is different between the uplink and downlink. Because this study focuses on the uplink, we will only describe uplink PUSC in detail.

In PUSC, the subcarriers across the entire frequency spectrum (excluding the guard bands) over 3 OFDMA symbols are divided and grouped into tiles. One tile is composed of 4 adjacent subcarriers over 3 OFDM symbols. Within each tile, there are 4 pilot subcarriers and 8 data subcarriers and are assigned as depicted in Figure 2.7. The tiles are then placed into 6 groups based on their location in the allocated channel spectrum and a subchannel formed by randomly selecting one tile from each of the 6 groups so that one subchannel is composed of tiles distributed across the entire spectrum. Therefore, each subchannel contains 48 data subcarriers and 24 pilot subcarriers.

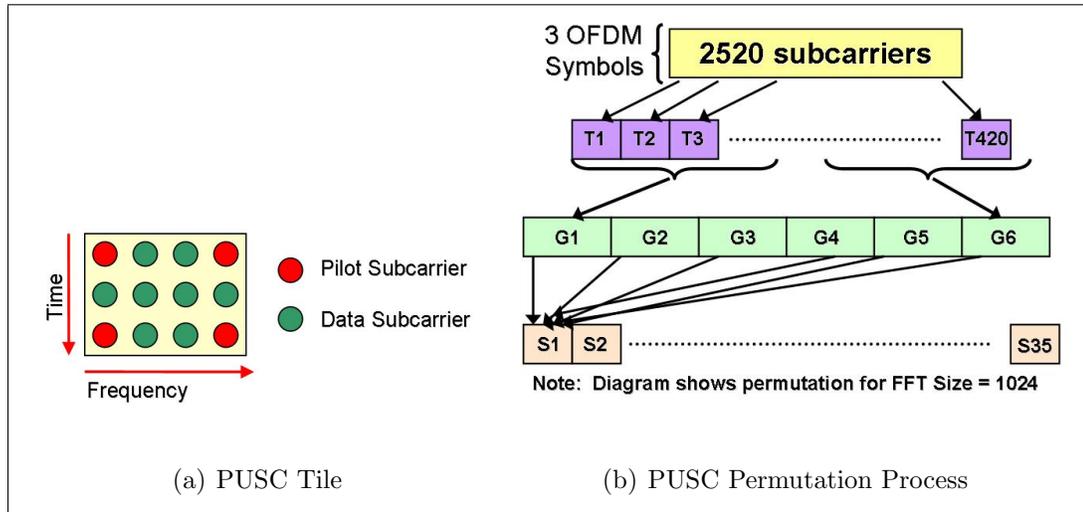


Figure 2.7: PUSC Tile and Permutation Process

### AMC - Adjacent Subcarrier Permutation

In AMC, the permutation is the same for both the uplink and downlink. In this permutation, one subchannel is composed of 18 adjacent subcarriers over 3 OFDM symbols. In each AMC subchannel, there are 48 data subcarriers and 6 pilot subcarriers assigned as depicted in Figure 2.8.

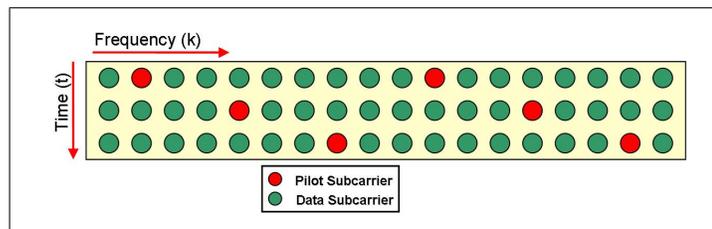


Figure 2.8: AMC Subchannel

Currently, the WiMAX system profile only requires support for this subchannel size. Future revisions of the profile may include support for AMC subchannels of different sizes. Since the scheduler decides how to allocate subchannels to users, it is possible for a user to be assigned adjacent subchannels.

### 2.1.6 Uplink Channel Sounding

A feature included in the IEEE 802.16e standard and the mobile WiMAX system profile is uplink channel sounding. Channel sounding is a technique used to provide the base station with channel response information between the base station and the mobile. In a TDD system, one of the benefits to using channel sounding is that the estimates calculated can be used for the downlink because of the reciprocity between the uplink and downlink channel. Therefore, to provide the base station with as current information as possible, the sounding zone is usually allocated at the end of the uplink frame. Channel sounding is a simple yet effective feature that can significantly increase the coverage and reliability of the system by allowing many closed-loop transmit antenna array techniques such as adaptive beamforming.

### 2.1.7 System Profiles

One of the benefits to WiMAX is that it can easily be scaled for many channel bandwidth sizes by simply adjusting the FFT size while keeping the subcarrier spacing fixed. This minimizes the impact to the higher layers and makes WiMAX easily deployable for many different requirements [13]. Currently, two system profiles that have been released are for channel bandwidth sizes of 5MHz and 10MHz. Without loss of generality, we performed our simulations for this study for the 10MHz channel bandwidth size. A table of the parameters are shown in Table 2.1. It should be noted that PUSC and AMC have different parameters and that their downlink and uplink parameters are different from each other.

Table 2.1: WiMAX OFDMA Symbol Parameters [25]

Parameter	DL PUSC	UL PUSC	DL AMC	UL AMC
System Channel Bandwidth	10 MHz			
FFT Size (N)	1024			
Null Subcarriers	184	184	159	159
Pilot Subcarriers	120	180	96	96
Data Subcarriers	720	560	768	768
Used Subcarriers ( $N_{\text{used}}$ )	940	740	865	865
Number of Subchannels	30	35	48	48
Subcarrier Spacing ( $T_f$ )	10.94kHz			
Sampling Frequency ( $1/T_n$ )	11.2MHz			
Cyclic Prefix Length	$1/4$ FFT Size = $22.8\mu\text{s}$ $1/8$ FFT Size = $11.4\mu\text{s}$ $1/16$ FFT Size = $5.7\mu\text{s}$			

The WiMAX profile also specifies many modulation and code rates to allow the system more control of the data rates. On the downlink, QPSK, 16-QAM, and 64-QAM are the required modulations. On the uplink, QPSK and 16-QAM are the required modulations while 64-QAM is optional. WiMAX also supports Convolutional Codes (CC) and Convolutional Turbo Codes with different code rates

and repetition factors [13]. Table 2.2 shows the modulation and code rates supported by WiMAX.

Table 2.2: WiMAX supported Modulation and Code Rates [13]

Downlink			Uplink		
Modulation	Code Rate	Repetition	Modulation	Code Rate	Repetition
QPSK	$\frac{1}{2}, \frac{2}{3}, \frac{3}{4}, \frac{5}{6}$	1, 2, 4, 6	QPSK	$\frac{1}{2}, \frac{2}{3}, \frac{5}{6}$	1, 2, 4, 6
16-QAM	$\frac{1}{2}, \frac{2}{3}, \frac{3}{4}, \frac{5}{6}$	1, 2, 4, 6	16-QAM	$\frac{1}{2}, \frac{2}{3}, \frac{5}{6}$	1, 2, 4, 6
64-QAM	$\frac{1}{2}, \frac{2}{3}, \frac{3}{4}, \frac{5}{6}$	1, 2, 4, 6	64-QAM	$\frac{1}{2}, \frac{2}{3}, \frac{5}{6}$	1, 2, 4, 6

## 2.2 Channel Model

In this section, we discuss briefly the development of the channel model used in our simulations. In a wireless environment, a signal propagating through a wireless channel is often distorted by a multipath fading channel. Fading describes an amplitude and/or phase change in the transmitted signal and the amount of fading a signal experiences is generally influenced by two factors: Multipath propagation and Doppler spread.

### 2.2.1 Multipath

Multipath distortion occurs when multiple copies of the transmitted signal arrives at the receiver from many different angles at different times. This type of

distortion is common in outdoor environments where RF waves reflect or refract off terrestrial objects like buildings and cars. When multiple copies of the signal arrive at roughly the same time, each copy is vectorially added together at the receiver. Because each copy undergoes a different amplitude and phase distortion, the signal strength at the receiver fluctuates. Generally, a multipath channel will have multiple multipath components because the transmitted signal may arrive at the receiver at different times. Depending on how spread out in time these multipath components arrive at the receiver will affect how rapidly the signal changes in frequency. One measure of this is coherence bandwidth. This measure is used to describe the frequency separation in which two frequency-domain samples become uncorrelated. The coherence bandwidth is inversely proportional to the delay spread of the channel which describes the difference in time between the first arriving signal path and the last arriving signal path. This dispersion in time can cause distortion in frequency and can cause intersymbol interference. For a broadband system, we generally describe the channel as frequency selective because the coherence bandwidth is smaller than the bandwidth of the transmitted signal. However, in an OFDM system, we divide the bandwidth into many narrowband subchannels and therefore, use the assumption that the fade is flat over each narrowband subchannel.

Because the received signal in a multipath channel is composed of copies of the transmitted signal that have been amplitude and/or phase distorted and delayed in time, a good model to use for the multipath channel is the tapped-delay line impulse response where each tap represents one multipath component at a particular time [18]. To model this distortion, a common statistical model for each multipath fading

component is Rayleigh fading. In this model, we assume that the envelope of the signal will fade with a Rayleigh distribution. This model holds in environments where there is a lot of scattering before the signal arrives at the receiver so that by Central Limit Theorem, the sum of all copies of the signal that arrive at the receiver can be modeled as a complex Gaussian process. In an environment where there is no direct line of sight between the transmitter and receiver, the complex Gaussian process will have zero mean and uniform phase distribution between 0 and  $2\pi$  radians. The amplitude response (or envelope) of the channel will then follow a Rayleigh distribution under the assumption that the real and imaginary components of the channel response are independent and identically distributed zero-mean Gaussian processes. If there is a direct line of sight, then it follows a Rician distribution.

### **2.2.2 Doppler Spread**

Fading can be described as slow or fast depending on the coherence time and symbol duration. Coherence time is a measure used to describe the time-varying nature of the channel and defined as the length of time between two time-domain samples in which they become uncorrelated. This parameter is inversely proportional to the Doppler spread which is defined as the difference between the highest and lowest Doppler shifts between signals arriving at the receiver at the same time. When the coherence time of a channel is smaller than the symbol duration, then we generally describe the channel as a fast fading channel. When the coherence

time of a channel is larger than the symbol duration, then we can assume that the channel stays relatively constant over one or more symbol durations. Doppler can create distortion because it causes the perceived frequency of the transmitted signal to be different at the receiver. When the transmitted signal is dispersed in frequency, distortion may occur in the time-domain [19]. The presence of Doppler can cause the subcarriers to lose orthogonality which introduces intercarrier interference (ICI) that can further degrade the system. Intercarrier interference describes the contribution of other subcarriers to the received signal at the subcarrier of interest. A common model to describe the correlation of a channel in the time is Jake's classical model which will be described in the next section.

### 2.2.3 Channel Impulse Response Model

To model this multipath fading channel, we represent the channel as a time-varying impulse response [19]

$$h(t, \tau) = \sum_{i=1}^P \alpha_i(t) \cdot \delta(\tau - \tau_i) \quad (2.1)$$

where  $\alpha_i(t)$  is the complex tap gain and assumed to be a complex Gaussian random variable,  $\tau_i$  is the delay of the  $i^{\text{th}}$  path, and  $P$  is the number of paths in the channel profile. Then, by definition, the channel frequency response is defined as

$$H(t, f) = \int_{-\infty}^{\infty} h(t, \tau) e^{-j2\pi f\tau} d\tau = \sum_{i=1}^P \alpha_i(t) e^{-j2\pi f\tau_i} \quad (2.2)$$

It is also useful to have the channel autocorrelation function because it can provide insight into the performance of each channel estimator. To obtain the channel

autocorrelation function, we can use the derivation provided in [11]. If we assume that the tap gains are wide-sense stationary (WSS) and independent of each path, then using Equation 2.2, the channel autocorrelation function is

$$\begin{aligned} R_H(\Delta t, \Delta f) &= E\{H(t + \Delta t, f + \Delta f)H^*(t, f)\} \\ &= \sum_{i=1}^P R_{\alpha_i}(\Delta t)e^{-j2\pi\Delta f\tau_i} \end{aligned} \quad (2.3)$$

If each path has the same time-domain correlation function, we can let  $R_{\alpha_i}(\Delta t) = \sigma_i^2 R_t(\Delta t)$  where  $R_t(\Delta t)$  is the normalized time-domain correlation function. Substituting this into Equation 2.3, we obtain

$$\begin{aligned} R_H(\Delta t, \Delta f) &= R_{\alpha_i}(\Delta t) \sum_i^P \sigma_i^2 e^{-j2\pi\Delta f\tau_i} \\ &= \sigma_H^2 R_t(\Delta t) R_f(\Delta f) \end{aligned} \quad (2.4)$$

where  $\sigma_H^2$  is the total average power of the channel impulse response and  $R_f(\Delta f)$  is the normalized frequency-domain correlation function respectively defined as

$$\begin{aligned} \sigma_H^2 &= \sum_i \sigma_i^2 \\ R_f(\Delta f) &= \sum_i \frac{\sigma_i^2}{\sigma_H^2} e^{-j2\pi\Delta f\tau_i} \end{aligned} \quad (2.5)$$

For an OFDM system, we redefine the time-domain correlation function and frequency-domain correlation function as such

$$R_t[\Delta t] = R_t(T_s \Delta t) \quad (2.6)$$

$$R_f[\Delta k] = R_f(T_f \Delta k) \quad (2.7)$$

$$R_H[\Delta t, \Delta k] = \sigma_H^2 R_t[\Delta t] R_f[\Delta k] \quad (2.8)$$

where  $k$  is the subcarrier index,  $t$  is the OFDM symbol index,  $T_s$  is the symbol duration and  $T_f$  is the subcarrier spacing in frequency.

A common model to use for  $R_t[\Delta t]$  is Jake's classical model where the assumption is that there is uniform scattering of RF waves in the environment [27]. Using Jake's classical model, the time-domain correlation function is defined as

$$R_t[\Delta t] = J_0(2\pi f_d T_s \Delta t) \quad (2.9)$$

where  $J_0(x)$  is the zeroth-order Bessel function of the first kind and  $f_d$  is the maximum Doppler spread in hertz.

#### 2.2.4 SUI and ITUR Channel Models

Two types of channel models that have been generally accepted for WiMAX simulations are the Stanford University Interim (SUI) and the International Telecommunication Union Radio Communication (ITU-R) channel models. The parameters defined in the SUI models are generally accepted as good models for fixed wireless applications while the parameters defined in the ITUR models are generally accepted as good models for mobile wireless applications. The table of parameters for the SUI and ITUR channel models are shown in Table A.3, Table A.1, and Table A.2 respectively in Appendix A. For purposes of this study, we only use the SUI-1, SUI-4, SUI-5, Pedestrian-A, and Vehicular-B models in our simulations. These models were chosen to represent channels of different delay spreads and Doppler spreads in order to evaluate the performance of our channel estimation algorithms under different channel conditions.

### 2.2.5 Noise

In our wireless communication system, we consider the presence of noise that can further distort the signal. A common model of noise is the additive white Gaussian noise model (AWGN) in which a noise signal with constant power spectral density and amplitude that follows the Gaussian distribution is added to the system. We include noise as part of the system to model the background distortion in a channel that is not already modeled by the multipath fading channel. Modeling this noise as AWGN comes from the fact that one source of noise is thermal noise (or Johnson noise) which occurs because of the fundamental property of matter [19]. Thermal noise is a result of electrons moving freely in all directions with random velocities. Over a large period of time, their average velocity is zero. But over a short period of time, there may be fluctuations in the velocity and these fluctuations are described as thermal noise. This noise was found to have a power spectral density that was approximately constant for all frequencies and an amplitude distribution that is Gaussian and at any frequency and uncorrelated in time [19].

## 2.3 System Model

Figure 2.2 illustrates the OFDM baseband system model. From the diagram, if we let  $X(t, k)$  be the data modulated using QPSK, 16-QAM, or 64-QAM at subcarrier  $k$  and assume a system with  $N$  subcarriers and sampling frequency  $\frac{1}{T_n}$

then for OFDM symbol  $t$ , the signal at time  $n$  after the IFFT function is

$$x[n] = \sum_{k=0}^{N-1} X(t, k) e^{j2\pi f_k n} \quad \text{for } 0 \leq n \leq N - 1 \quad (2.10)$$

where  $f_k$  is the subcarrier frequency at the  $k^{\text{th}}$  subcarrier defined as

$$f_k = f_0 + k \cdot T_f \quad \text{for } 0 \leq k \leq N - 1$$

After the FFT function, we add the cyclic prefix to protect the signal from ISI by copying the last fraction of the data payload and appending it to the beginning of the OFDM symbol. If we assume the cyclic prefix length is  $L_g$ , then the transmitted OFDM symbol is defined as [17]

$$x_g[n] = \begin{cases} x[N + n] & \text{for } n = -L_g, -L_g + 1, \dots, -1 \\ x[n] & \text{for } n = 0, 1, \dots, N - 1 \end{cases} \quad (2.11)$$

In our system, if we let  $h[n]$  be the impulse response of the multipath fading channel and  $v[n]$  be zero-mean complex additive white Gaussian noise, then we can model the received OFDM symbol as

$$y_r[n] = x_g[n] * h[n] + v[n] \quad (2.12)$$

where  $*$  is the convolution operator.

After removing the cyclic prefix from  $y_r[n]$ , the signal  $y[n]$  is demodulated using the FFT operator. If we assume that cyclic prefix eliminates all intersymbol interference, then we can replace the convolution operator in equation 2.12 with the circular convolution operator. After the received symbol is demodulated using the FFT operator, the received signal for the  $t^{\text{th}}$  OFDM symbol and the  $k^{\text{th}}$  subcarrier

is given by

$$Y(t, k) = X(t, k)H(t, k) + V(t, k) \quad \text{for } 0 \leq k \leq N - 1 \quad (2.13)$$

where  $X(t, k)$ ,  $H(t, k)$ , and  $V(t, k)$  are the transmitted signal, the channel response, and zero-mean additive white Gaussian noise with variance  $\sigma_v^2$  respectively. We can also rewrite Equation 2.13 in matrix notation as given by

$$\mathbf{Y} = \mathbf{X}\mathbf{H} + \mathbf{V} \quad (2.14)$$

where

$$\mathbf{Y} = [Y(t, 0), Y(t, 1), \dots, Y(t, N - 1)]^T$$

$$\mathbf{X} = \text{diag}([X(t, 0), X(t, 1), \dots, X(t, N - 1)])$$

$$\mathbf{H} = [H(t, 0), H(t, 1), \dots, H(t, N - 1)]^T$$

$$\mathbf{V} = [V(t, 0), V(t, 1), \dots, V(t, N - 1)]^T$$

# Chapter 3

## Channel Estimation in PUSC (Partial Usage of Subcarrier)

### 3.1 Introduction

PUSC is a distributed permutation mode in WiMAX in which subcarriers are grouped into tiles and pseudorandomly allocated to form subchannels. The tile is illustrated in 3.1. Each tile is composed of 12 received samples,  $Y(t, k)$  for  $t = 0, \dots, 2$  and  $k = 0, \dots, 3$  in which 4 are reserved for pilots. The set of pilot positions is  $\mathcal{P} = \{(0,0), (0,3), (2,0), (2,3)\}$ . Because subchannels are composed of randomly selected tiles, the channel estimation algorithm must process each tile independently. By this statement, we mean that no information from pilots in adjacent tiles can be used. In this chapter, we will analyze and evaluate the performance of channel estimation algorithms that can be used to estimate the channel at each subcarrier within a tile. Specifically, we will derive the MSE using our analytical model for each estimator

and illustrate each estimator’s performance at the link level using BER plots.

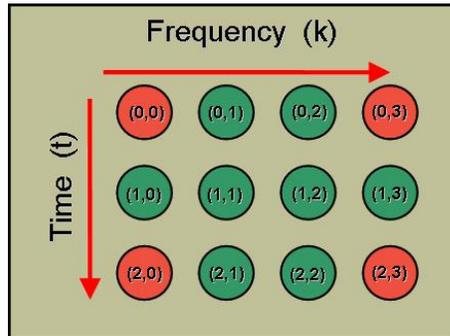


Figure 3.1: PUSC Tile

## 3.2 Channel Estimation Algorithms

In each of the algorithms, the first step is to generate a baseline channel estimate at each received pilot. Because both the transmitted and received signal are known at the receiver, we make estimates at the pilot subcarriers based on a least squares approach given by

$$\hat{H}(t, k) = \frac{Y(t, k)}{X(t, k)} \text{ for } (t, k) \in \mathcal{P} \quad (3.1)$$

The least squares estimate is the best estimate if there is no noise.

### 3.2.1 Linear Interpolation

The linear interpolation algorithm is currently the common approach to channel estimation in PUSC. This algorithm first produces least squares estimates at the pilots and then uses them to linearly interpolate the channel in time and then in frequency. Given the PUSC tile shown in Figure 3.1, the steps of the algorithm

described mathematically are

1. Perform Least Squares Estimation at Pilot Positions using Equation 3.1.
2. Interpolate Channel Estimates in Time at Subcarrier between Pilots

$$\begin{aligned}\hat{H}(1, 0) &= \frac{1}{2}\hat{H}(0, 0) + \frac{1}{2}\hat{H}(2, 0) \\ \hat{H}(1, 3) &= \frac{1}{2}\hat{H}(0, 3) + \frac{1}{2}\hat{H}(2, 3)\end{aligned}\tag{3.2}$$

3. Interpolate Channel Estimates in Frequency at each OFDM symbol

$$\begin{aligned}\hat{H}(t, 1) &= \frac{2}{3}\hat{H}(t, 0) + \frac{1}{3}\hat{H}(t, 3) \text{ for } t = 0, 2 \\ \hat{H}(t, 2) &= \frac{1}{3}\hat{H}(t, 0) + \frac{2}{3}\hat{H}(t, 3) \text{ for } t = 0, 2\end{aligned}\tag{3.3}$$

### Performance Analysis

We now consider the mean-squared error (MSE) performance of the linear interpolator in the PUSC tile. To evaluate its performance, we calculate the MSE at each data subcarrier and average them over the 8 data subcarriers in the PUSC tile. In the PUSC tile, there are 3 types of subcarriers when using the linear interpolator. The first are the subcarriers that are interpolated in time from step 2 of the estimation process. We define these subcarriers as  $\mathcal{A} = \{(1, 0), (1, 3)\}$ . The second are the subcarriers interpolated in frequency in OFDM symbols containing pilots. Because there are two subcarriers in the same OFDM symbol in which the channel estimate is produced by interpolating the pilots in frequency, we calculate the arithmetic average MSE over the pair of subcarriers. We define these subcarrier positions to be  $\mathcal{B} = \{\mathcal{B}_0, \mathcal{B}_2\}$ , where  $\mathcal{B}_0 = \{(0, 1), (0, 2)\}$  and  $\mathcal{B}_2 = \{(2, 1), (2, 2)\}$ . This simplifies calculations since the average MSE of the subcarriers in  $\mathcal{B}_0$  and  $\mathcal{B}_2$

are the same. The third are the subcarriers in the middle which are interpolated in both time and frequency. We define these subcarrier positions as  $\mathcal{C} = \{(1, 1), (1, 2)\}$

In our calculation of average MSE, we assumed that each channel estimate was made according to Equation 3.1, Equation 3.2, and Equation 3.3. In addition, we assume that the channel is wide-sense stationary (WSS) and that the noise is mutually uncorrelated and uncorrelated with the channel to simplify calculations. If we also assume that for the pilot constellation points all have the same magnitude,  $|X(t, k)| = A$  for  $(t, k) \in \mathcal{P}$ , then the mean-squared error derivation for each type of subcarrier is

1. Mean-squared error at  $(t, k) \in \mathcal{A}$

$$\begin{aligned} MSE_{\mathcal{A}} &= E[|\hat{H}(t, k) - H(t, k)|^2] \\ &= \sigma_H^2 \left\{ \frac{3}{2} R_t[0] + \frac{1}{4} (R_t[2] + R_t[-2]) - (R_t[1] + R_t[-1]) \right\} \\ &\quad + \frac{1}{2} \frac{\sigma_v^2}{A^2} \end{aligned} \quad (3.4)$$

where  $R_t[\Delta t]$  is the time-domain correlation function defined in Equation 2.6.

2. Average mean-squared error for  $\{(t, k), (t, k+1)\} \in \mathcal{B}_t$  for  $t = \{0, 2\}$ .

$$\begin{aligned} MSE_{\mathcal{B}_t} &= \frac{1}{2} [MSE(t, k) + MSE(t, k+1)] \\ &= \frac{1}{2} \sum_{l=1}^2 E[|\hat{H}(t, l) - H(t, l)|^2] \\ &= \sigma_H^2 \left\{ \frac{14}{9} R_f[0] + \frac{2}{9} (R_f[3] + R_f[-3]) \right. \\ &\quad \left. - \frac{4}{3} (R_f[1] + R_f[-1]) - \frac{2}{3} (R_f[2] + R_f[-2]) \right\} \\ &\quad + \frac{5}{9} \frac{\sigma_v^2}{A^2} \end{aligned} \quad (3.5)$$

where  $R_f[\Delta k]$  is the frequency-domain correlation function defined in Equation 2.7.

3. Mean-squared error at  $(t,k) \in \mathcal{C}$

$$\begin{aligned}
MSE_{\mathcal{C}} &= E[|\hat{H}(t,k) - H(t,k)|^2] \\
&= \frac{23}{18}R_H[0,0] + \frac{5}{36}(R_H[2,0] + R_H[-2,0]) + \frac{1}{9}(R_H[0,3] + R_H[0,-3]) \\
&\quad + \frac{1}{18}(R_H[2,3] + R_H[-2,3] + R_H[2,-3] + R_H[-2,-3]) \\
&\quad - \frac{1}{3}(R_H[1,1] + R_H[1,-1] + R_H[-1,1] + R_H[-1,-1]) \\
&\quad - \frac{1}{6}(R_H[1,2] + R_H[1,-2] + R_H[-1,2] + R_H[-1,-2]) \\
&\quad + \frac{5}{18} \frac{\sigma_v^2}{A^2}
\end{aligned} \tag{3.6}$$

where  $R_H[\Delta t, \Delta k]$  is the channel autocorrelation function for the frequency response at different times and frequencies as defined in Equation 2.8.

Given the MSE at each type of subcarrier within the PUSC tile, the average MSE in a PUSC tile is defined as

$$MSE_{LI-PUSC} = \frac{2}{8}MSE_{\mathcal{A}} + \frac{4}{8}MSE_{\mathcal{B}} + \frac{2}{8}MSE_{\mathcal{C}} \tag{3.7}$$

We see from the above theoretical calculations that the average MSE depends on two factors: interpolation error and noise. The interpolation error depends on the correlation of the channel. Interpolation error occurs because the estimator uses a first-order approximation of the channel when the channel may be nonlinear at the data subcarriers between the pilots. The contribution of noise depends on the signal to noise ratio.

Using the above theoretical calculations, we plotted the average MSE of the linear interpolator in PUSC. We assumed that the channel was Pedestrian-A 3km/hr, Pedestrian-A 120km/hr, Vehicular-B 3km/hr, and Vehicular-B 120km/hr with power delay profiles defined in Table A.1 and Table A.2 and calculated their respective correlation by using Equation 2.6 and Equation 2.7.

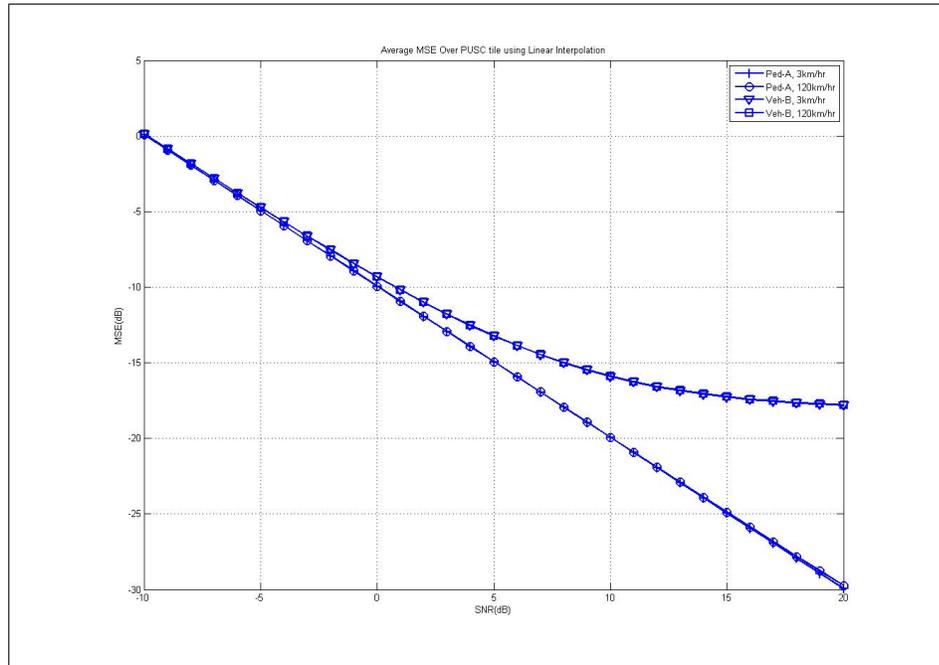


Figure 3.2: Average MSE Over PUSC tile using Linear Interpolation

The plot indicates that the linear interpolator channel estimation performance is better in Pedestrian-A than it is in Vehicular-B. We expect this because in a Pedestrian-A channel, the delay spread is short so the frequency-domain channel response is relatively flat and a first order approximation is fairly accurate in this type of channel. From Equation 3.4, 3.5, and 3.6, we see that the interpolation error equals zero if the channel is perfectly correlated within the PUSC tile. However, in Vehicular-B, the delay spread is much longer and hence, the channel between

the pilot subcarriers is less correlated. This means the channel can change rapidly between adjacent subcarriers and introduce more interpolation error. We can also see that this curve begins to floor at  $\text{MSE} = -17\text{dB}$  and  $\text{SNR} = 15\text{dB}$ . This indicates that for SNR levels greater than 15dB, there is no improvement in the linear interpolation channel estimation performance and a system operating in this channel must choose a modulation code rate that can tolerate at least a MSE of -17dB. The plot also suggests that Doppler spread does not affect channel estimation performance much.

### 3.2.2 4-pilot Averaging

Another approach to performing channel estimation within a PUSC tile is averaging the 4 received pilots and using that result to estimate the channel. The motivation to this method is that in channels where noise is the dominant contributor to distortion, the channel estimates at the pilot positions are too corrupted to use for interpolation. So the idea behind this technique is that a better estimate of the channel could be made by using all the pilots to average out some of the noise and using this estimate at all data subcarriers in the tile. The steps of this algorithm mathematically are given by

1. Perform Channel Estimation at pilot positions using Equation 3.1.
2. Average the 4 channel estimates at the pilot subcarriers and use at each data subcarrier. The set of data subcarriers within a tile is  $\mathcal{D} = \{(0,1), (0,2), (1,0),$

(1,1), (1,2), (1,3), (2,1), (2,2)}

$$\hat{H}(t, k) = \frac{1}{4}\hat{H}(0, 0) + \frac{1}{4}\hat{H}(2, 0) + \frac{1}{4}\hat{H}(0, 3) + \frac{1}{4}\hat{H}(2, 3) \quad (3.8)$$

### Analytical Model and Performance Analysis

Similar to the analytical model for the linear interpolator, we also created an analytical model to evaluate the channel estimation performance with the 4-pilot averaging technique. If we use the channel estimate calculated using Equation 3.8, then the MSE at each subcarrier using the 4-pilot averaging technique is given by

$$\begin{aligned} MSE_D(t, k) &= E[|\hat{H}(t, k) - H(t, k)|^2] \\ &= \frac{5}{4}R_H[0, 0] + \frac{1}{8}(R_H[2, 0] + R_H[-2, 0]) + \frac{1}{8}(R_H[0, 3] + R_H[0, -3]) \\ &\quad + \frac{1}{16}(R_H[2, 3] + R_H[-2, 3] + R_H[2, -3] + R_H[-2, -3]) \\ &\quad - \frac{1}{4}(R_H[t, k] + R_H[-t, -k]) - \frac{1}{4}(R_H[t - 2, k] + R_H[2 - t, -k]) \\ &\quad - \frac{1}{4}(R_H[t, k - 3] + R_H[-t, 3 - k]) \\ &\quad - \frac{1}{4}(R_H[t - 2, k - 3] + R_H[2 - t, 3 - k]) + \frac{1}{4}\frac{\sigma_v^2}{A^2} \end{aligned} \quad (3.9)$$

Therefore, the average MSE within the PUSC tile using the 4-pilot averaging method is given by

$$MSE_{4avg} = \frac{1}{8} \left\{ \sum_{k=1}^2 MSE_D(0, k) + \sum_{k=0}^3 MSE_D(1, k) + \sum_{k=1}^2 MSE_D(2, k) \right\} + \frac{1}{4}\frac{\sigma_v^2}{A^2} \quad (3.10)$$

Again, we see that the 4-pilot averaging channel estimation performance depends on the interpolation error and noise. Assuming that the channel was Pedestrian-A 3km/hr, Pedestrian-A 120km/hr, Vehicular-B 3km/hr, and Vehicular-B 120km/hr

with power delay profiles defined in Table A.1 and Table A.2 and calculating the correlation using Equation 2.6 and Equation 2.7, we generated the following plot.

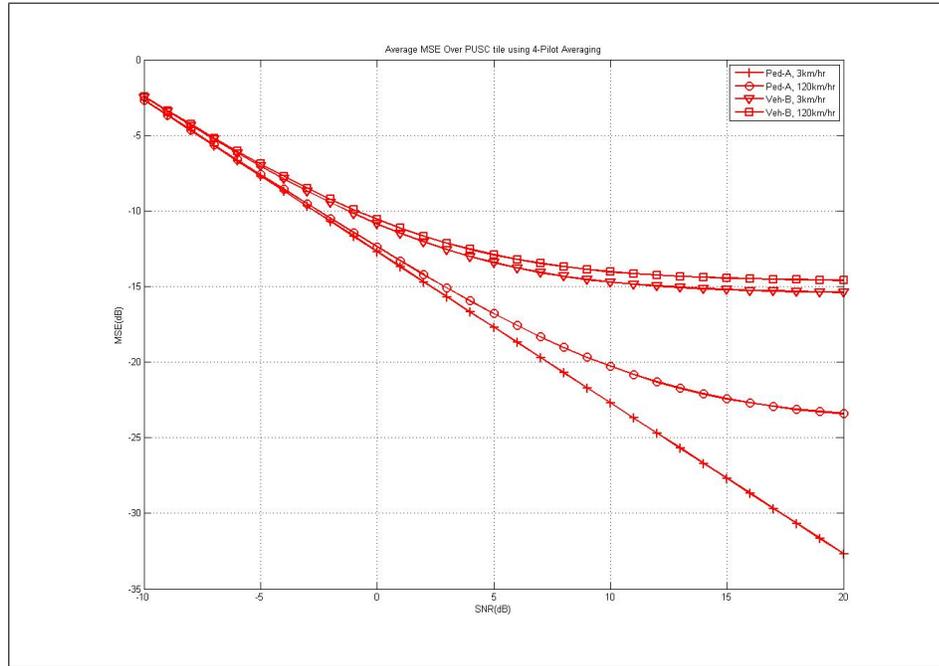


Figure 3.3: Average MSE Over PUSC tile using 4-Pilot Averaging

In essence, the 4-pilot averaging algorithm estimates the channel with a constant. Therefore, inherent in the 4-pilot averaging technique is the assumption that the channel is highly correlated. However, this estimator degrades rapidly as the channel becomes less correlated within the tile due to increased delay spread and/or Doppler spread. The plot in Figure 3.3 supports this statement because we do see that the performance is worse in a Vehicular-B channel than it is in a Pedestrian-A channel and worse at 120km/hr than at 3km/hr.

### 3.2.3 Genie-Aided Channel Estimator (Perfect Channel Knowledge)

In our study of PUSC channel estimation algorithms, we also use a genie-aided channel estimator to compare the performance of our channel estimators with one that has perfect channel knowledge. The idea to using the genie-aided estimator is to allow us to understand what the optimal channel estimation performance is in a WiMAX PUSC system. It can also be used to evaluate what performance a perfectly adaptive estimator could achieve. To implement the genie-aided estimator, we took a copy of the transmitted symbol and copy of the received signal without any noise and correlated them together. These estimates were then passed to the equalizer. A block diagram illustrating the implementation of the genie-aided channel estimator is shown in Figure 3.4.

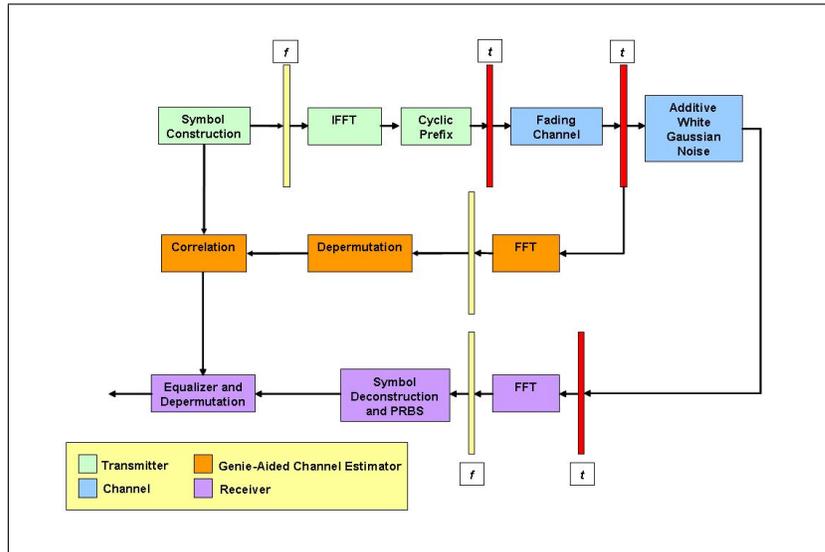


Figure 3.4: Genie-aided Channel Estimator

## 3.3 Simulation

For our study of channel estimation algorithms in PUSC, we used a WiMAX link-level simulator with all the components shown in the WiMAX system block diagram in Figure 2.1 and therefore, allows us to obtain system level bit-error rate (BER) curves to evaluate each algorithm's performance.

### 3.3.1 Objective

In our simulations, our objective was to first evaluate the performance the linear interpolation channel estimation algorithm in different types of channel conditions and compare it to the genie-aided channel estimator. We used the results to identify conditions where improving the channel estimator was warranted. Next, we then compared our results of the linear interpolator and the 4-pilot averaging channel estimator.

### 3.3.2 Simulation Parameters

To conduct our simulation, we tested each algorithm for 8 modulation code rates with 4 different channel models. The modulation code rates used were QPSK with code rate  $1/2$  and repetition 1,2 and 4, QPSK with code rate  $3/4$ , 16QAM with code rates  $1/2$  and  $3/4$ , 64QAM with code rates  $2/3$ , and  $3/4$ . These modulation code rates were selected to illustrate the performance of our channel estimators for varying levels of throughput. The channel models we selected were the ITUR Pedestrian-A at 3km/hr, Pedestrian-A at 120km/hr, Vehicular-B at 3km/hr, and

Vehicular-B at 120km/hr. This set of channel models were chosen to show how each channel estimator performs in low/high delay spread and low/high Doppler spread.

The other notable simulation parameters are shown in Table 3.1. One note to make from this table is that the cyclic prefix was selected to be 1/4 of the data payload, or about  $22.85\mu\text{s}$  in length. This cyclic prefix size was specifically selected in order to eliminate ISI from our results.

Table 3.1: Simulation Parameters for PUSC Channel Estimation Evaluation

Parameter	Value
Modulation Code Rate	QPSK, R1/2, RPT4 QPSK, R1/2, RPT2 QPSK, R1/2 QPSK, R3/4 16QAM, R1/2 16QAM, R3/4 64QAM, R2/3 64QAM, R3/4
Coding Option	CTC (Convolutional Turbo Coder)
Bandwidth Scale Factor	8 (1024 FFT)
Fundamental Bandwidth	1.25MHz
Cyclic Prefix	1/4 (22.85 $\mu$ s)
Number of Tx Antennas	1
Number of Rx Antennas	2
Sampling Rate ( $T_n$ )	11.2MHz
Symbol Duration ( $T_s$ )	114.3 $\mu$ s

### 3.3.3 Linear Interpolation vs. Genie-Aided Estimator

#### Results

The BER curves comparing the linear interpolator and the genie-aided estimator are shown in this section.

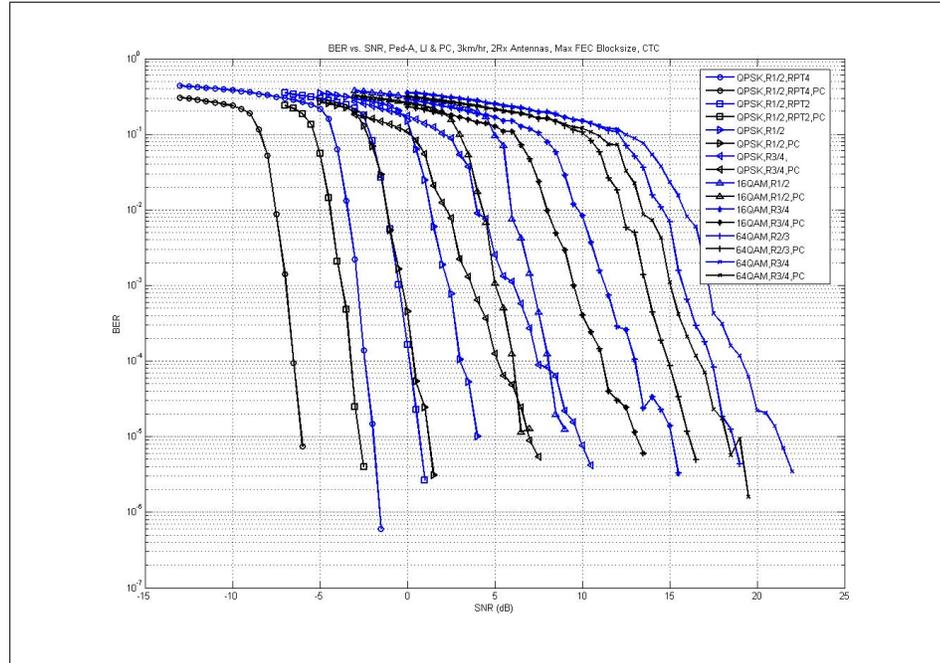


Figure 3.5: BER curves of Linear Interpolator vs. Genie-Aided Channel Estimator in Ped-A, 3km/hr model



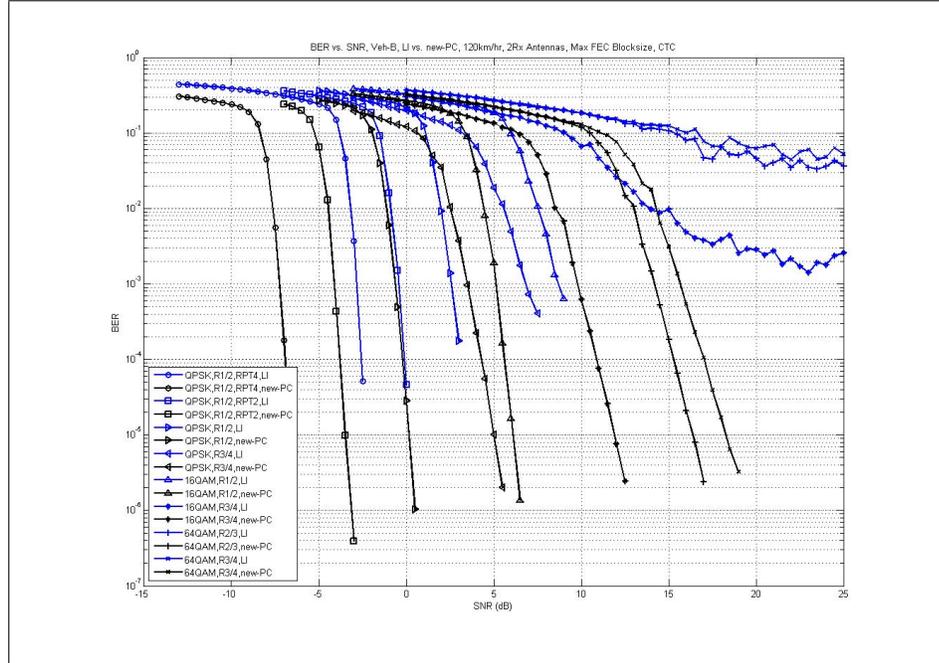


Figure 3.8: BER curves of Linear Interpolator vs. Genie-Aided Channel Estimator in Veh-B, 120km/hr model

### Observations

One observation made from these curves is that for Pedestrian-A, there is a wider gap in performance between the linear interpolator and the genie-aided channel estimator at low SNR values than at high SNR values. There is about a 4dB difference in SNR for QPSK and about a 2dB difference in SNR for 16QAM and 64QAM between the linear interpolator and genie-aided channel estimator. We can explain this result by looking at the linear interpolation MSE curve in Figure 3.2. We can see that the linear interpolator channel estimation error decreases with increasing SNR. And because the Pedestrian-A channel has a relatively flat frequency domain channel response, the linear interpolation channel estimation error

is mostly attributed to noise. So in this channel we would expect the performance gap between the genie-aided channel estimator and linear interpolator to close as the SNR increases.

However, we cannot make the same conclusion when we use the Vehicular-B channel model. While the gap between the linear interpolator and genie-aided estimator is about 4dB for QPSK, we can see that the gap begins to widen for 16-QAM and even floors for 16-QAM R3/4 and 64-QAM. We can use the MSE curve in Figure 3.2 to confirm that because the BER curves (Figure 3.7 and 3.8 and the MSE curve, we can see that the performance begins to floor at a SNR = 15dB. In this channel, the delay spread is long which introduces a large interpolation error that 16-QAM R3/4 and 64QAM cannot support because the constellation points are too close together to compensate for it.

### 3.3.4 Linear Interpolation vs. 4-pilot Averaging

#### Results: Bit-Error Rate

From the BER curves shown in Figures 3.5 - 3.8, we observed that the gap between the linear interpolator and genie-aided channel estimator is about 4dB for QPSK. QPSK is typically used in low SNR channels so a technique used to close that gap is the 4-pilot averaging technique. The idea is to use 4-pilots and average out some of the effect of noise to produce a better estimate. If we combine Equations 3.4, 3.5, 3.6 with Equation 3.7, we obtain the noise term in the average MSE of channel estimation in a PUSC tile using linear interpolation to be  $\frac{17}{36} \frac{\sigma_n^2}{A^2}$ . And if

we combine 3.9 with 3.7, we obtain the noise term in the average MSE of channel estimation in a PUSC tile using 4-pilot averaging to be  $\frac{1}{4} \frac{\sigma_v^2}{A^2}$ . From this perspective, we can see that 4-pilot averaging removes about twice as much noise as the linear interpolator. To illustrate their performances at the system level, we also generated BER curves comparing their performances. The BER curves are shown below.

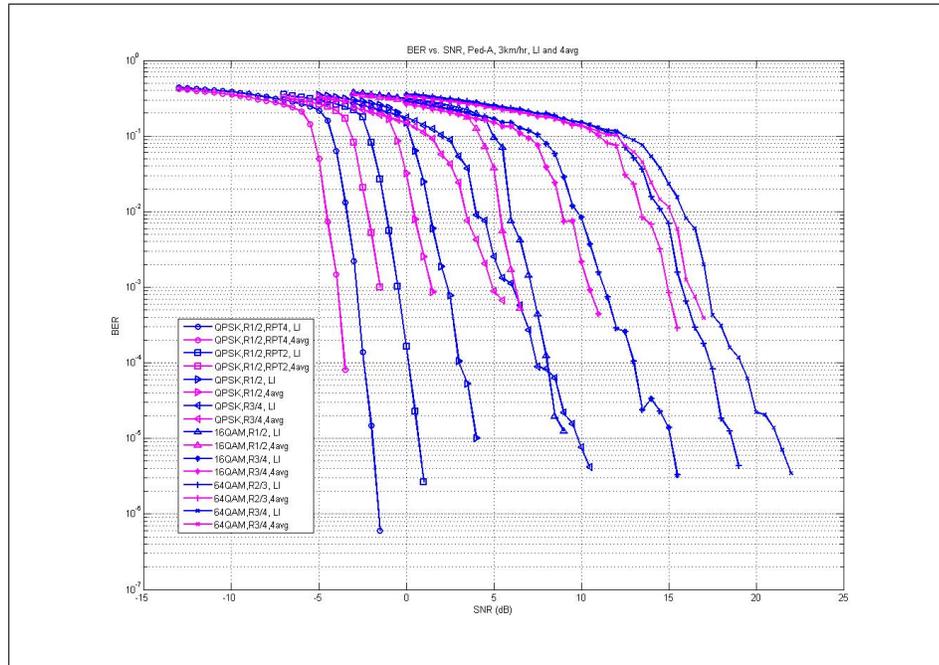


Figure 3.9: BER curves of Linear Interpolator vs. 4-pilot Averaging in Ped-A, 3km/hr model

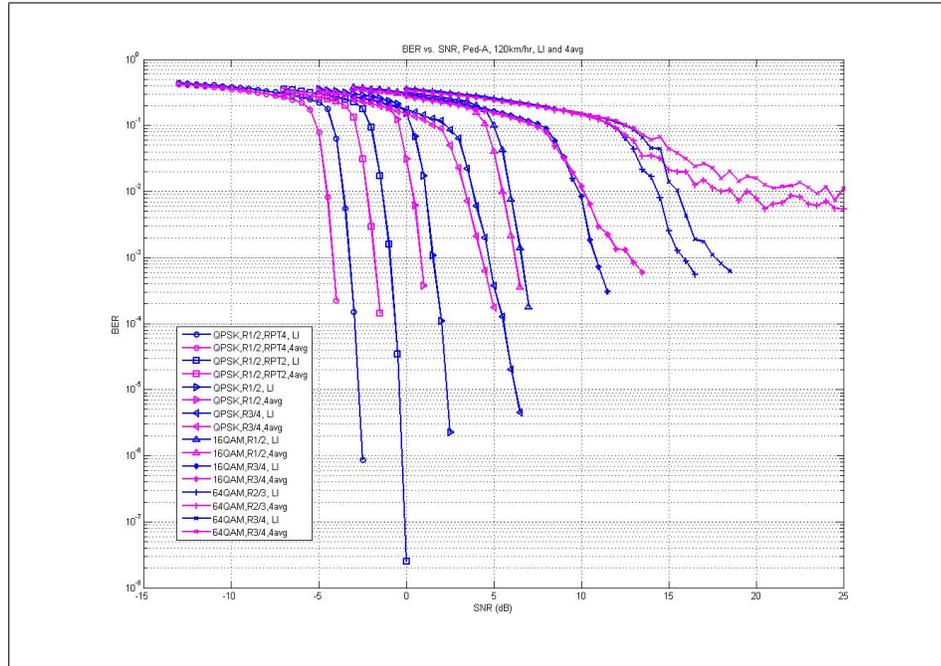


Figure 3.10: BER curves of Linear Interpolator vs. 4-pilot Averaging in Ped-A, 120km/hr model

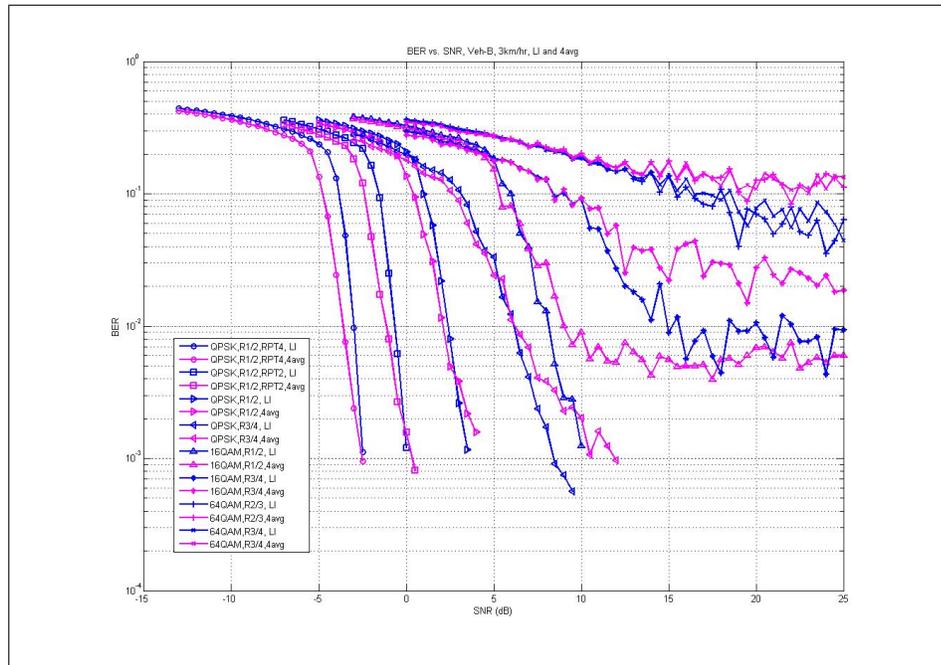


Figure 3.11: BER curves of Linear Interpolator vs. 4-pilot Averaging in Veh-B, 3km/hr model

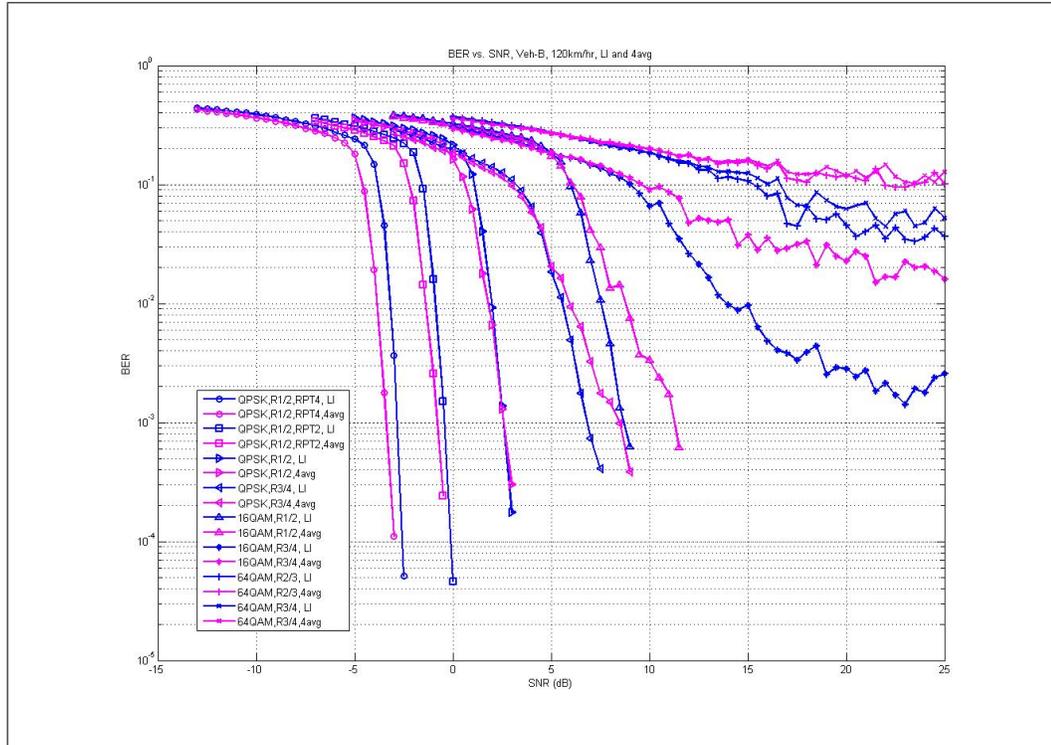


Figure 3.12: BER curves of Linear Interpolator vs. 4-pilot Averaging in Veh-B, 120km/hr model

### Observations: Bit-Error Rate

From the results, the 4-pilot averaging estimator appears to outperform the linear interpolation estimator for all modulation code rates by about 2 dB in the Pedestrian-A, 3km/hr model and by about 1.5dB for all modulation code rates, except 16QAM R3/4, 64QAM R2/3, and 64QAM R3/4 in the Pedestrian-A 120km/hr model in which the linear interpolation algorithm outperforms the 4-pilot averaging estimator. In the Vehicular-B models, the 4-pilot averaging estimator appears to outperform the linear interpolation channel estimator for QPSK, but performs worse than the linear interpolation channel estimator at the other modulation code rates.

From these plots, we can conclude that the 4-pilot averaging method does perform better than the linear interpolator at low SNR. We can explain this result by the fact that at low SNR, noise is the dominant contributor to distortion and the 4-pilot averaging method reduces about 3dB more noise than the linear interpolator. However, as we increase the SNR, we can see that the relative performance of the 4-pilot averaging estimator to the linear interpolator decreases in all channels except in the Pedestrian-A 3km/hr channel. This suggests that the interpolation error in the 4-pilot averaging method is higher than in the linear interpolator. In the 4-pilot averaging method, there is an implicit assumption that the channel is perfectly correlated within the PUSC tile. This is analogous to trying to approximate a curve with a constant whereas with the linear interpolator, we approximate a curve with a first order function. And at higher SNRs, the channel is the dominant contributor to distortion so removing more noise provides marginal improvements in performance. We can confirm this by overlaying Figure 3.2 and 3.3 onto the same plot as shown in Figure 3.13.

From this plot, we can identify the points where the linear interpolator begins to outperform the 4-pilot averaging method. First, we can identify that in Pedestrian-A 3km/hr, the 4-pilot averaging method is about 3dB better because in this scenario, the interpolation error is small for both estimators while the 4-pilot averaging method removes twice as much noise. In the Pedestrian-A 120km/hr channel, there's a crossover point at  $\text{SNR} = 11\text{dB}$ . In the Vehicular-B channel, at 3km/hr, the crossover point is at  $\text{SNR} = 6\text{dB}$  and at 120km/hr, the crossover point is at  $\text{SNR} = 4\text{dB}$ . If we compare these points, to the crossover points in the BER

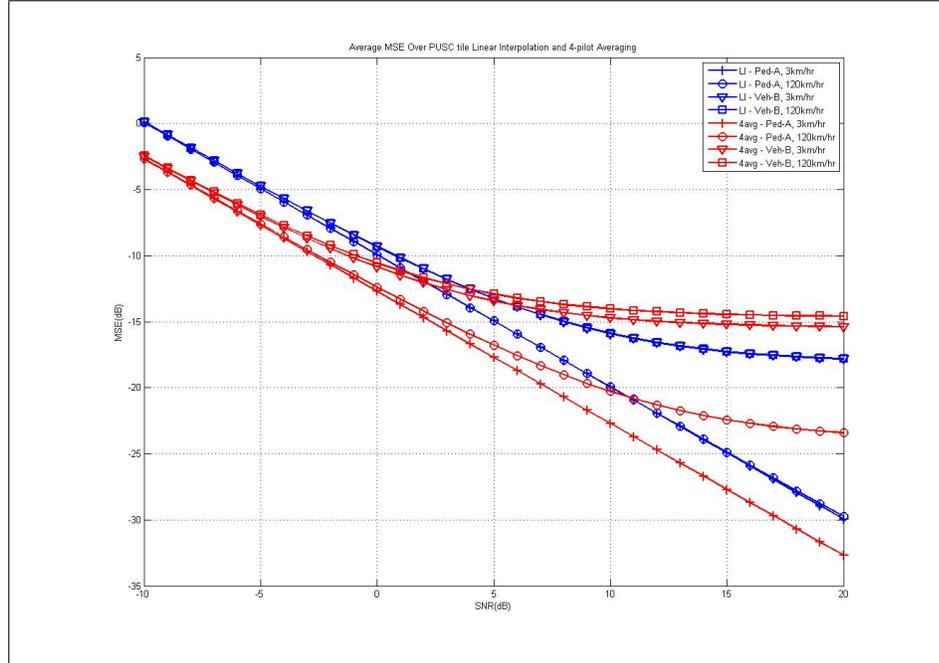


Figure 3.13: Average MSE Over PUSC tile: Linear Interpolation and 4-pilot Averaging

curves in Figure 3.5 - 3.8, we can see that they are very close. These results show that the interpolation error in the 4-pilot averaging increases more rapidly when the channel becomes less correlated due to delay spread and/or Doppler spread.

From an implementation perspective, one way to implement both estimators is to use the 4-pilot averaging estimator for SNR less than 4dB and the linear interpolator for SNR greater than 4dB. Because we observed that 4dB is the crossover point in Vehicular-B 120km/hr, we can assume that this channel represents the worst-case scenario and that using a 4-pilot averaging method for SNR less than 4dB would provide an improvement in the overall system.

### 3.4 Conclusions

In this section, we compared the performance of the linear interpolator to the genie-aided channel estimator and the performance of the linear interpolator to the 4-pilot averaging method. In both the linear interpolator and 4-pilot averaging method, we observed that the 2 factors that contributed to channel estimation error were interpolation error and noise. We observed that a higher delay spread and Doppler spread resulted in a channel that was less correlated within the PUSC tile and hence, led to a higher interpolation error. In comparing the linear interpolator to the genie-aided channel estimator, we observed that in the Pedestrian-A model, there was a 4dB difference in performance at low SNR and about a 2dB difference. However, for Vehicular-B, we observed that there were modulation code rates the system could not support using the linear interpolator and that the gap between the linear interpolator and genie-aided channel estimator diverged as the SNR increased. In comparing the 4-pilot averaging method to the linear interpolator, we observed that the 4-pilot averaging performed better at low SNR no matter what the channel conditions were because it averaged out about 3dB more noise than the linear interpolator. However, for channels with larger delay and Doppler spreads, we noticed that the linear interpolator outperformed the 4-pilot averaging method after a certain SNR point.

# Chapter 4

## Channel Estimation in AMC (Adaptive Modulation and Coding)

### 4.1 Introduction

Adaptive Modulation and Coding (AMC) is another permutation supported by WiMAX. AMC is classified as an adjacent subcarrier permutation in which subchannels are composed of adjacent subcarriers and allocated to users by the MAC scheduler based on their unique channel conditions. Each subchannel is composed of 18 adjacent subcarriers over 3 OFDM symbols with 6 pilot subcarriers per subchannel. In this chapter, we will explore channel estimation algorithms that can be used to estimate the channel within an AMC subchannel. In addition, we will also consider the scenario when two adjacent subchannels (36 adjacent subcarriers over

3 OFDM symbols) are allocated to a user in which pilots from both subchannels can be used to aid in channel estimation. Similar to the analysis performed for PUSC, we will also derive the MSE using our analytical model for each estimator and compare the analysis to results obtained in simulation.

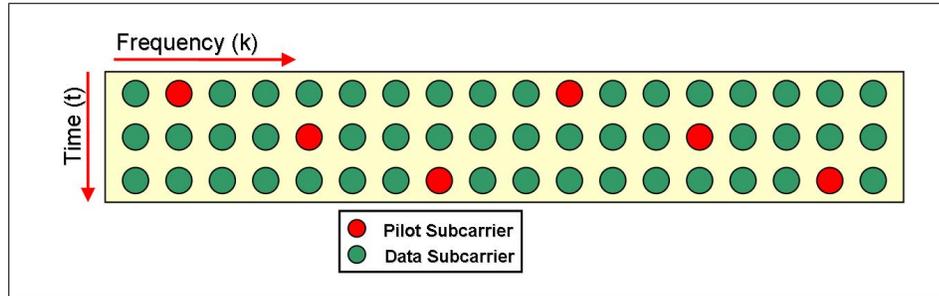


Figure 4.1: AMC Subchannel

## 4.2 Channel Estimation Algorithms

To simplify the channel estimation problem, we have assumed that the channel remains constant over 3 OFDM symbols. This assumption allows us to assume that the channel estimate at a particular subcarrier can be used to estimate the same subcarrier at a different time within the subchannel. By simplifying the problem in this manner, we can employ channel estimation algorithms that process one OFDM symbol with  $M = 18$ , 36 adjacent subcarriers and a pilot every 3 subcarriers and 1 data subcarrier at the edge of each subchannel. Figure 4.2 shows the arrangement of pilots within one OFDM symbol under this assumption.

The first step is to generate baseline channel estimates at the pilot subcarriers using the least squares estimate. Using the arrangement shown in Figure 4.2, if we

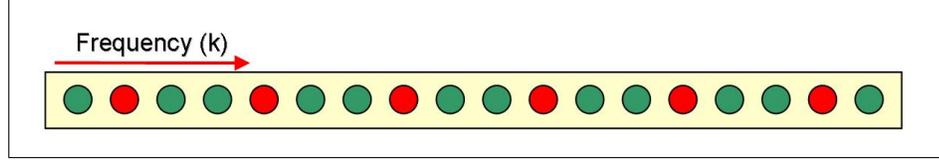


Figure 4.2: Arrangement of Pilots in one OFDM symbol within one AMC subchannel under assumption that channel remains constant over 3 OFDM symbols

let  $P$  be the total number of pilots for  $M$  total number of subcarriers, then there is a pilot every  $D = M/P$  subcarriers. For AMC,  $D=3$ . Therefore, the estimate of the channel at the pilot subcarriers is given by

$$\hat{H}(t, 3p + 1) = \frac{Y(t, 3p + 1)}{X(t, 3p + 1)} \text{ for } p = 0, 1, \dots, P-1 \quad (4.1)$$

#### 4.2.1 Linear Interpolation

Linear interpolation is the channel estimation algorithm commonly employed right now. This algorithm estimates the channel at each pilot and then linearly interpolates the channels at the data subcarriers between two subcarriers and extrapolates the channel at the data subcarriers at the edges of the subchannel. Given the arrangement shown in Figure 4.2, the steps of the algorithm described mathematically are

1. Generate Baseline Channel Estimates at Pilot Positions using Equation 4.1.
2. Interpolate Channel Estimates in Frequency at Subcarrier between Pilots

$$\begin{aligned} \hat{H}(t, 3p + 2) &= \frac{2}{3}\hat{H}(t, 3p + 1) + \frac{1}{3}\hat{H}(t, 3p + 4), \text{ for } p = 0, 1, \dots, P-1 \\ \hat{H}(t, 3p + 3) &= \frac{1}{3}\hat{H}(t, 3p + 1) + \frac{2}{3}\hat{H}(t, 3p + 4), \text{ for } p = 0, 1, \dots, P-1 \end{aligned} \quad (4.2)$$

3. Extrapolate Channel Estimates in Frequency at Subcarriers at Edges of Subchannel

$$\begin{aligned}\hat{H}(t, 0) &= \frac{4}{3}\hat{H}(t, 1) - \frac{1}{3}\hat{H}(t, 4) \\ \hat{H}(t, M - 1) &= \frac{4}{3}\hat{H}(t, M - 2) - \frac{1}{3}\hat{H}(t, M - 5)\end{aligned}\tag{4.3}$$

### Performance Analysis

To evaluate the linear interpolator, we derived the average MSE of an AMC subchannel using the arrangement shown in Figure 4.2. In this arrangement, there are 3 types of subcarriers: pilot subcarriers, interpolated subcarriers, and extrapolated subcarriers. To compute the average MSE in an AMC subchannel, we derived the MSE for each type of subcarriers. The derivations below are provided by [22]. In the derivation, we assume the channel is WSS and that the noise is mutually uncorrelated and uncorrelated with the channel. If we also assume that the pilot constellation points all have the same magnitude,  $|X(t, 3p + 1)| = A$  for  $p = 0, 1, \dots, P-1$ , then the MSE at each type of subcarrier is given by

1. MSE at pilot subcarriers from Step 1

$$\begin{aligned}MSE_P &= E\{|\hat{H}(t, 3p + 1) - H(t, 3p + 1)|^2\} \\ &= \frac{\sigma_v^2}{A^2}\end{aligned}\tag{4.4}$$

2. Arithmetic average MSE over interpolated subcarriers from Step 2

$$\begin{aligned}
MSE_I &= \frac{1}{2} \sum_{l=1}^2 E\{|\hat{H}(t, 3p+1+l) - H(t, 3p+1+l)|^2\} \\
&= \sigma_H^2 \left\{ \frac{14}{9} R_f[0] + \frac{2}{9} (R_f[3] + R_f[-3]) \right. \\
&\quad \left. - \frac{4}{3} (R_f[1] + R_f[-1]) - \frac{2}{3} (R_f[2] + R_f[-2]) \right\} \\
&\quad + \frac{5}{9} \frac{\sigma_v^2}{A^2}
\end{aligned} \tag{4.5}$$

3. MSE at extrapolated subcarriers from Step 3

$$\begin{aligned}
MSE_E &= E\{|\hat{H}(t, 0) - H(t, 0)|^2\} = E\{|\hat{H}(t, M-1) - H(t, M-1)|^2\} \\
&= \sigma_H^2 \left\{ \frac{26}{9} R_f[0] + \frac{1}{3} (R_f[4] + R_f[-4]) \right. \\
&\quad \left. - \frac{4}{3} (R_f[1] + R_f[-1]) - \frac{4}{9} (R_f[3] + R_f[-3]) \right\} \\
&\quad + \frac{17}{9} \frac{\sigma_v^2}{A^2}
\end{aligned} \tag{4.6}$$

Given the MSE at the pilot, interpolated, and extrapolated subcarriers, we can calculate the average MSE over an AMC subchannel with M adjacent subcarriers and P pilot subcarriers.

$$MSE_{LI-AMC} = \frac{1}{M} [P \cdot MSE_P + (M - P - 2) \cdot MSE_I + 2 \cdot MSE_E] \tag{4.7}$$

From the derivation above, we show that the performance of the linear interpolation channel estimator depends on the interpolation error and noise. The interpolation error term depends on the channel correlation and the noise term depends on the SNR of the channel. Using the derivation, we plotted the average MSE of the linear interpolator over one AMC subchannel (M = 18) for channel models SUI-1, SUI-4, SUI-5, Pedestrian-A, and Vehicular-B with power delay profiles defined in Table A.3

and Table A.1 and Table A.2 and correlation defined by Equation 2.7. Again we

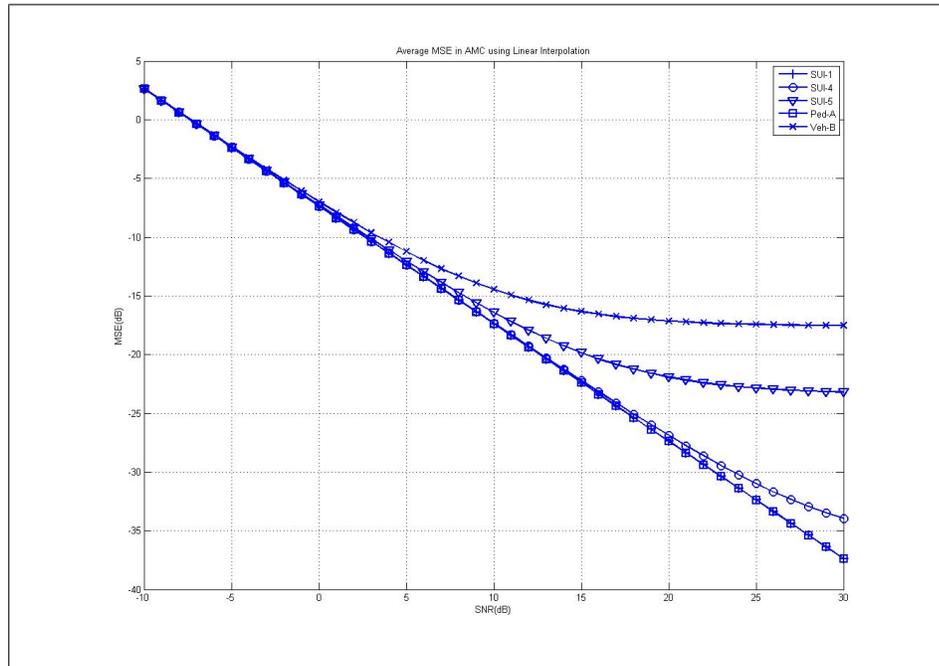


Figure 4.3: Average MSE over 1 AMC Subchannel ( $M = 18$ ) using Linear Interpolation

see that performance of the linear interpolator depends on the delay spread of the channel. However, we also observe that the performance of the linear interpolator floors in SUI-5 and Vehicular-B. In these channels, the frequency response changes too rapidly in frequency for the linear interpolator to accurately estimate the channel with the given pilot arrangement even as the SNR increases.

## 4.2.2 Frequency Smoothing

In this algorithm, the idea is to smooth the frequency-domain channel estimates at the pilots by applying a rectangular window in the time-domain. Because of the duality between the time and frequency domain, this method is analogous to

applying a low-pass filter in the time-domain (or convolution with a sinc function in the frequency-domain). However, because in AMC there are only pilots every  $D = 3$  subcarriers, 3 images of the channel impulse response will appear in the time-domain. Because of these images, the window size must be selected such that the images are eliminated. A block diagram illustrating the system implementing the frequency smoothing algorithm is shown in Figure 4.4.

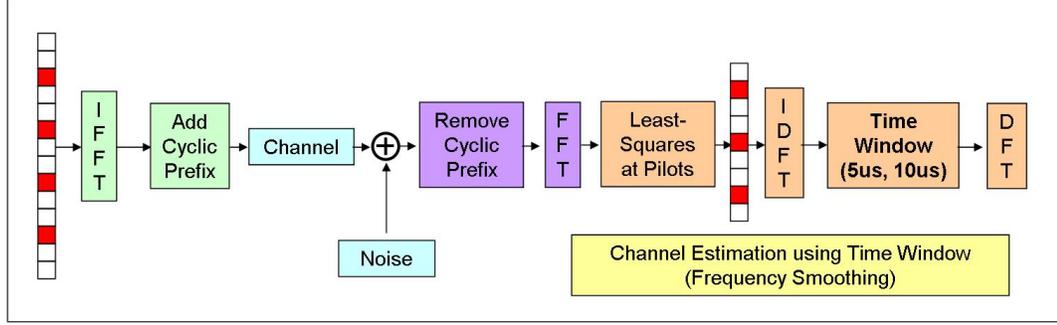


Figure 4.4: Block Diagram of WiMAX with Frequency Smoothing Channel Estimation Algorithm

If we assume that each pilot has the same magnitude,  $|X(t, 3p + 1)| = A$  for  $p = 0, 1, \dots, P-1$ , then the steps of the algorithm described mathematically can be described as

1. Generate Baseline Channel Estimates at Pilot Positions using Equation 4.1.
2. Construct vector of length M with baseline channel estimates at pilot subcarriers and zero at the data subcarrier positions. Use M-point IDFT to transform baseline channel estimate to time domain.

$$\hat{h}[m] = IDFT\{\hat{\mathbf{H}}\} = h[m] + \frac{v[m]}{A} \quad (4.8)$$

3. Zero out samples beyond a set time threshold or multiply time domain response by a rectangular window  $w[m]$  of length L equal to time threshold. To calculate L, we select a desired window size in time and use the formula below to calculate the number of samples that it corresponds to

$$L = \text{round} \left( M \cdot \frac{\text{Window Size in Time}}{\text{Symbol Duration}} \right) \quad (4.9)$$

$$w[m] = \begin{cases} 1 & \text{for } m = 0, 1, \dots, L-1 \\ 0 & \text{for } m = L, L+1, \dots, M-1 \end{cases} \quad (4.10)$$

$$\hat{h}_{fs}[m] = w[m]\hat{h}[m] \quad (4.11)$$

4. Transform windowed response back to the frequency domain using M-point DFT to obtain channel estimates

$$\hat{H}_{fs}(k) = \sum_{m=0}^{L-1} \hat{h}_{fs}[m]e^{-j2\pi k \frac{m}{M}} + \sum_{m=0}^{L-1} \frac{v[m]}{A}e^{-j2\pi k \frac{m}{M}} \quad (4.12)$$

In general, we would like to choose a window size that corresponds exactly to the maximum delay spread of the channel. Assuming that the delay spread of the channel is less than 1/3 the symbol duration, this would allow us to reject the images and the most noise possible without losing any channel energy and inducing a bias in our channel estimates. However, the delay spread is typically unknown so a fixed window size is pre-determined. But if we choose an arbitrary window length, L, such that it is longer than the channel delay spread and eliminates the images, then using Equation 4.12, the average MSE at each subcarrier within the AMC subchannel is given by

$$\begin{aligned} MSE_{FS} &= E\{||\hat{H}_{fs}(t, k) - H(t, k)||^2\} \text{ for } k = 0, 1, \dots, M-1 \\ &= \frac{L}{M} \frac{\sigma_v^2}{A^2} \end{aligned} \quad (4.13)$$

From Equation 4.13, we can see that the frequency smoothing channel estimation performance depends only on noise if we choose L to be longer than the length of the channel impulse response. The frequency smoothing channel estimator reduces the effect of noise by a factor of L/M. And when we compare the frequency

smoothing performance to the linear interpolation performance, we can see that even in a channel that was perfectly correlated, the performance of the frequency smoothing algorithm will be better as long as  $L$  is chosen to be less than  $\frac{4P+5M+24}{9}$ . In AMC ( $P = 6$ ,  $M = 18$ ), we see that if we choose  $L < 15$ , then the frequency smoothing estimator will outperform the linear interpolator. In WiMAX, for  $N = 1024$  and cyclic prefix fraction =  $1/8$ , the symbol duration is  $102.9\mu\text{s}$  so  $L = 15$  corresponds to window length of  $85.75\mu\text{s}$ . Typically, the channel delay spread will be less than  $10\mu\text{s}$  and out of all the channels we test with, Vehicular-B has the longest delay spread with a  $20\mu\text{s}$  delay spread. Therefore, for most channels, we can select a window length such that it will outperform the linear interpolator.

## 4.3 Simulation

### 4.3.1 Objective

In our study of AMC channel estimation algorithms, our objective was to compare the performance of the linear interpolator and frequency smoothing algorithm under different channel conditions. To evaluate the performance of each estimator, we used the mean-squared error and error vector magnitude metric. The metrics

are calculated as follows.

$$EVM(k) = |\hat{H}(t, k) - H(t, k)| \text{ for } k = 0, 1, \dots, M-1 \quad (4.14)$$

$$MSE_{dB} = 10 \log_{10} \left\{ \frac{1}{M} \sum_{k=0}^{M-1} |\hat{H}(t, k) - H(t, k)|^2 \right\} \quad (4.15)$$

$$(4.16)$$

Currently, we do not have a complete WiMAX link-level simulator supporting AMC so for purposes of this study, we created our own in MATLAB with a subset of the components. A block diagram of the simulator used to perform this study is shown in Figure 4.5. We use the same channel models as we did for our study in PUSC so for evaluation of channel estimation algorithms, this model is sufficient.

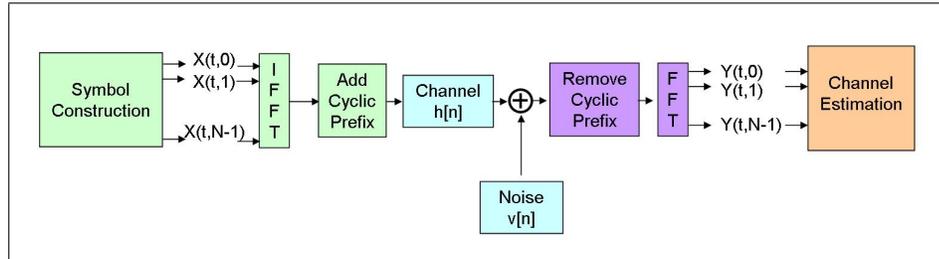


Figure 4.5: WiMAX Matlab Simulator for AMC and Channel Sounding Simulations

### 4.3.2 Simulation Parameters

For this experiment, we chose to test the performance of the linear interpolator and frequency smoothing algorithm for AMC subchannel sizes of  $M = 18$  and  $M = 36$  and assuming a pilot arrangement as shown in Figure 4.2. For the frequency smoothing algorithm, we chose to test time window lengths of  $5\mu s$  and  $10\mu s$ . The  $5\mu s$  window was chosen because in general, many of the channels encountered in

practice have a maximum delay spread less than  $5\mu s$ . We chose the  $10\mu s$  window because this is close to the nominal cyclic prefix size for WiMAX and a window of this length can accommodate those channels with longer than  $5\mu s$  delay spread. The system parameters used in this simulation are shown in Table 4.1.

Table 4.1: Simulation Parameters for AMC Channel Estimation Evaluation

<b>System Parameters</b>	
Parameter	Value
Bandwidth Scale Factor	8 (1024 FFT)
Fundamental Bandwidth	1.25MHz
Cyclic Prefix	1/8 ( $11.42\mu s$ )
Number of Tx Antennas	1
Number of Rx Antennas	2
Sampling Rate	11.2MHz
Symbol Duration	$102.9\mu s$
<b>Frequency Smoothing Parameters</b>	
Parameter	Value
Window Size	$5\mu s, 10\mu s$

### 4.3.3 Results: Mean-Squared Error

Using our AMC simulator, we first generated MSE curves using the linear interpolator, frequency smoothing with  $5\mu\text{s}$  window, and frequency smoothing with  $10\mu\text{s}$  window. In this simulation, we generated 100 realizations of the channel at each SNR and computed the average MSE over the 100 realizations to generate each point on the curve. The plots are shown below.

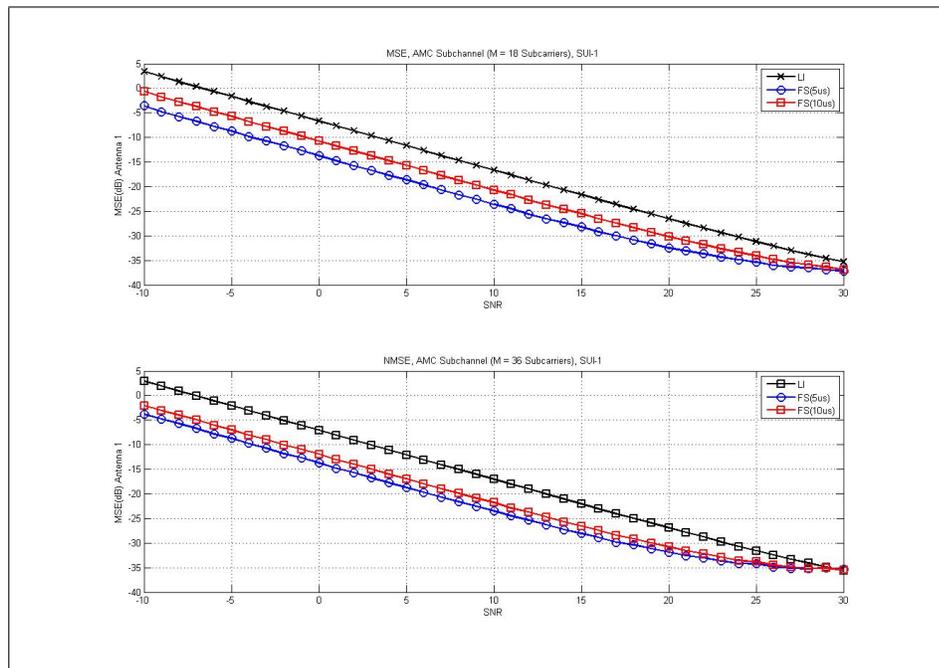


Figure 4.6: MSE, 1 AMC Subchannel (M=18,36), SUI-1

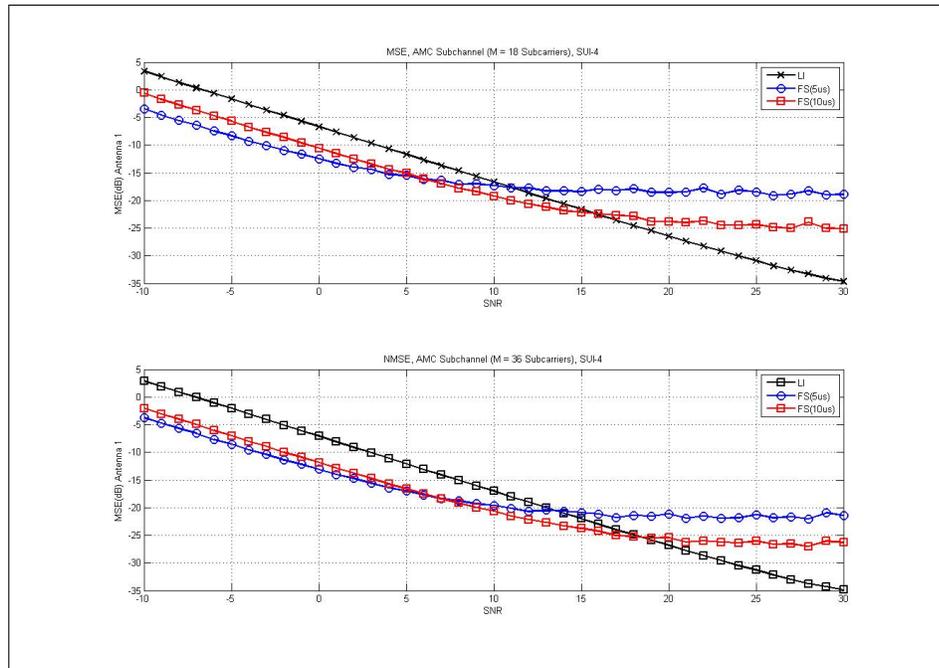


Figure 4.7: MSE, 1 AMC Subchannel (M=18,36), SUI-4

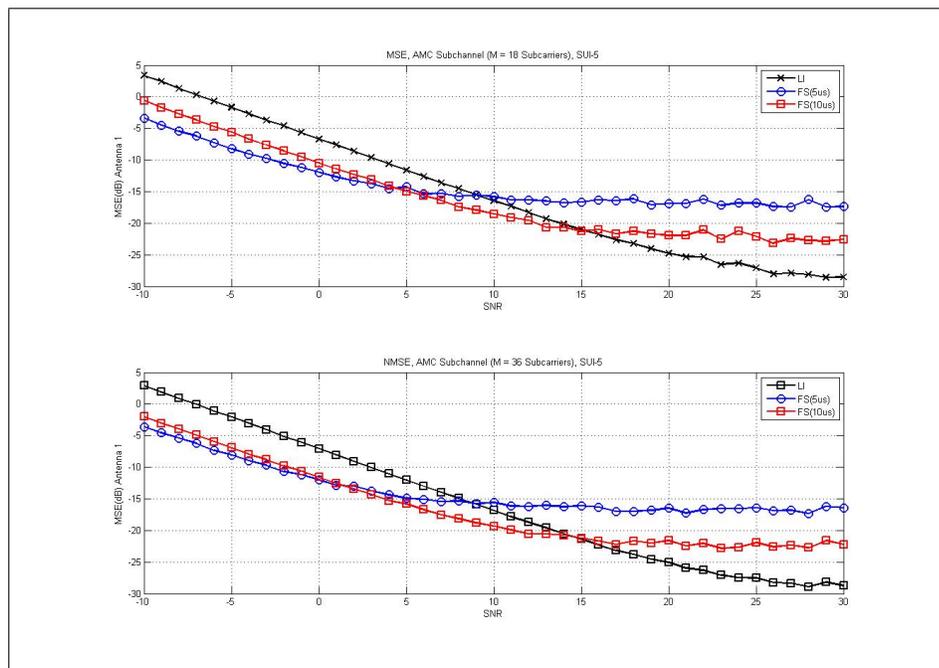


Figure 4.8: MSE, 1 AMC Subchannel (M=18,36), SUI-5

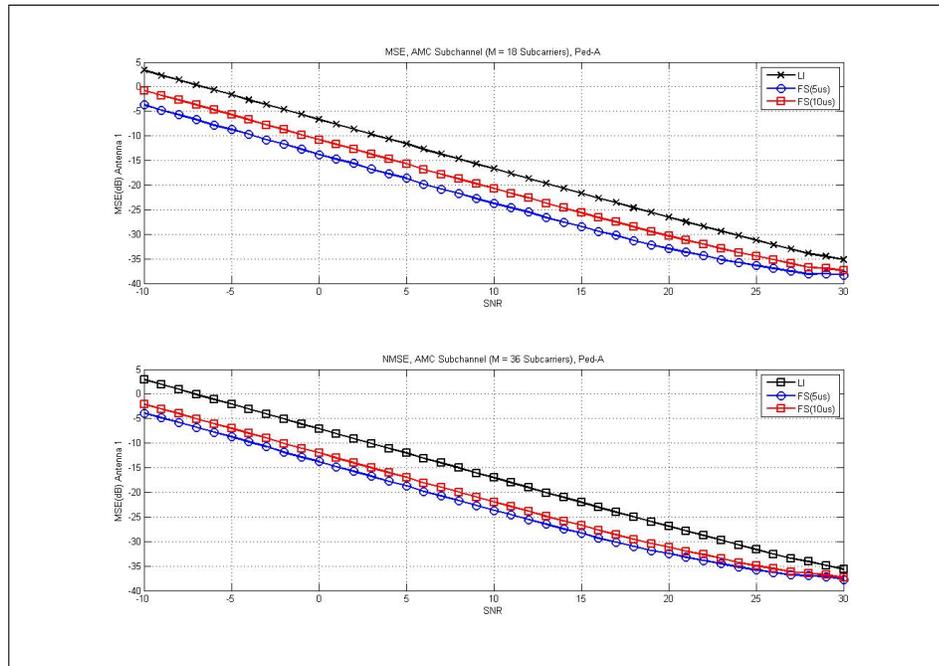


Figure 4.9: MSE, 1 AMC Subchannel (M=18,36), Ped-A

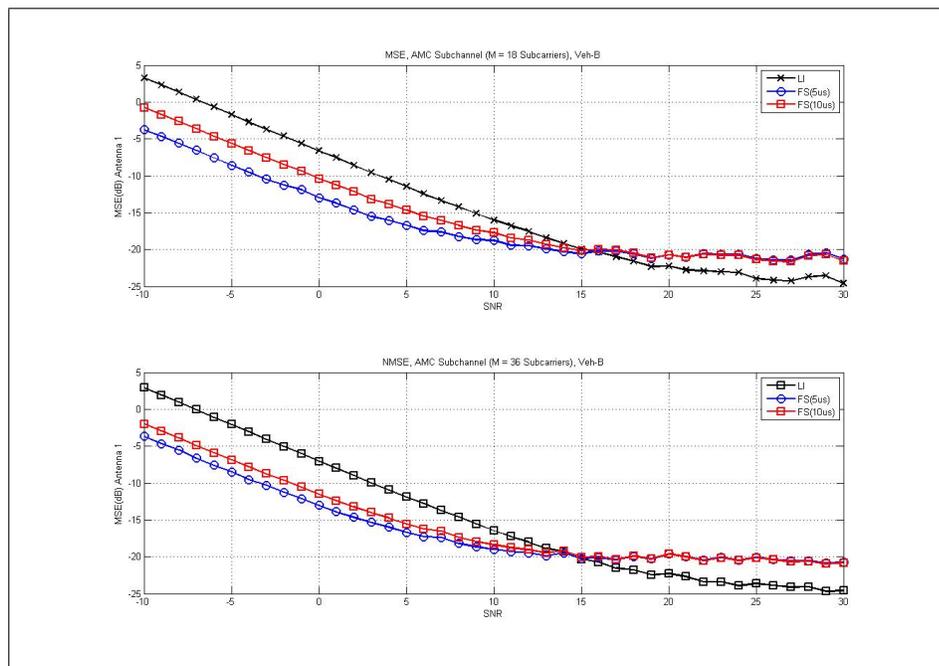


Figure 4.10: MSE, 1 AMC Subchannel (M=18,36), Veh-B

### 4.3.4 Observations: Mean-Squared Error

From the plots, we observe that the frequency smoothing algorithm is generally better than the linear interpolator. In the low delay spread channels like SUI-1 and Pedestrian-A, we expect the frequency smoothing algorithm to always be better because it rejects more noise than the linear interpolator. However, for the higher delay spread channels like SUI-4, SUI-5, and Vehicular-B, we see that the performance of the frequency smoothing algorithm will floor at some point and that the linear interpolator will outperform the frequency smoothing estimator beyond a certain SNR point. We can attribute this flooring to the frequency smoothing algorithm not using a window size large enough to capture all the channel's energy. Appendix B displays the channel impulse response of each channel model used in this experiment. From these plots, we observe that for SUI-4, SUI-5, and Veh-B, the  $5\mu\text{s}$  and  $10\mu\text{s}$  windows do not capture the entire channel impulse response and hence, introduces a bias in the channel estimates which causes the frequency smoothing algorithm to floor as the SNR increases.

We also see a tradeoff between bias and noise rejection in the frequency smoothing algorithm. By choosing a smaller window size, we can eliminate more noise but also run the risk of not capturing all of the channel's energy and biasing the estimates. In channels where the SNR is very low, the frequency smoothing with  $5\mu\text{s}$  window performs better than with the  $10\mu\text{s}$  window. More specifically, the frequency smoothing with  $5\mu\text{s}$  window is better than the frequency smoothing with  $10\mu\text{s}$  window for almost all SNR levels for SUI-1 and Ped-A but only better up

to about 7dB in SUI-4 and 4dB in SUI-5. For Veh-B, the performance of the  $5\mu\text{s}$  window is better at low SNR levels and begins to converge as the SNR increases. If we look at the impulse response of the channel (without noise) after an M-point IFFT in Appendix B, it can be seen that SUI-1 and Ped-A both have their energy concentrated in one tap because of their short delay spreads. From Table B.1 in Appendix B, both the  $5\mu\text{s}$  and  $10\mu\text{s}$  window capture this tap and induce no bias in the estimate. However, since the  $5\mu\text{s}$  window rejects more noise, its performance is always better than the  $10\mu\text{s}$  window in these channels. In Veh-B, the delay spread of the channel is large but has most of its channel energy concentrated in its first two taps. So when we observe its impulse response after an M-point IFFT, we see that the impulse response within both the  $5\mu\text{s}$  and  $10\mu\text{s}$  window are very similar. However, in SUI-4 and SUI-5, we see that the impulse response within the  $5\mu\text{s}$  window is very different from the response within a  $10\mu\text{s}$  window. We can see that the  $10\mu\text{s}$  window captures significantly more of the channel's energy than the  $5\mu\text{s}$  window and therefore, tends to perform better in high SNR environments. Hence, in these channels, there is a crossover point in which using a larger frequency smoothing window is preferred over rejecting more noise.

### 4.3.5 Results: Error Vector Magnitude

To obtain more insight in the performance of each channel estimation algorithm, we also produced error vector magnitude plots of each channel estimation algorithm at SNR = -10dB and 20dB. The results are displayed below. These plots

show the average error at each subcarrier in an AMC subchannel over 100 realizations of the channel.

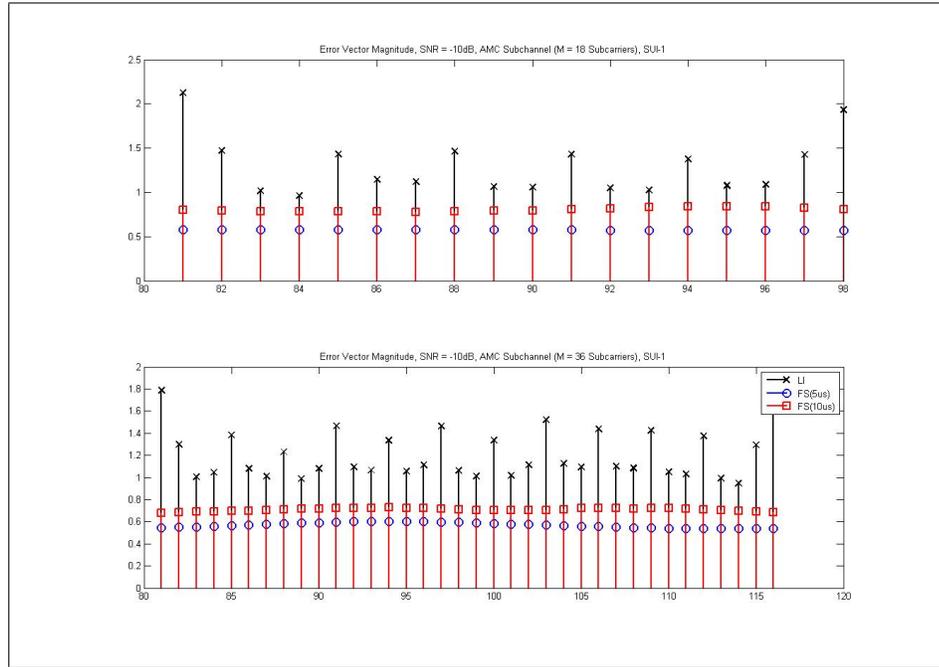


Figure 4.11: Error Vector Magnitude, SNR = -10dB, 1 AMC Subchannel (M = 18,36 Subcarriers), SUI-1

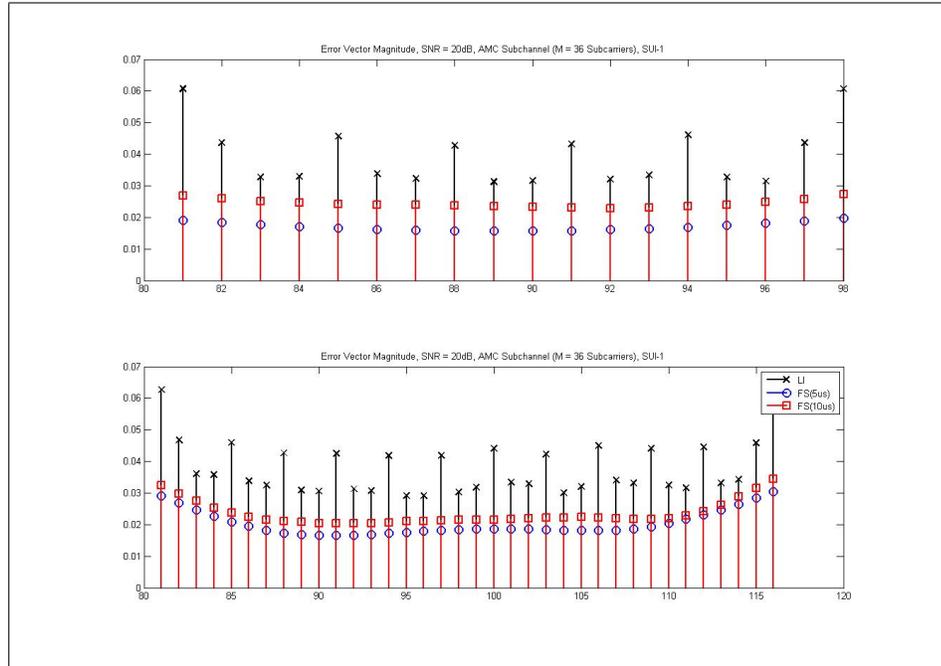


Figure 4.12: Error Vector Magnitude, SNR = 20dB, 1 AMC Subchannel (M = 18,36 Subcarriers), SUI-1

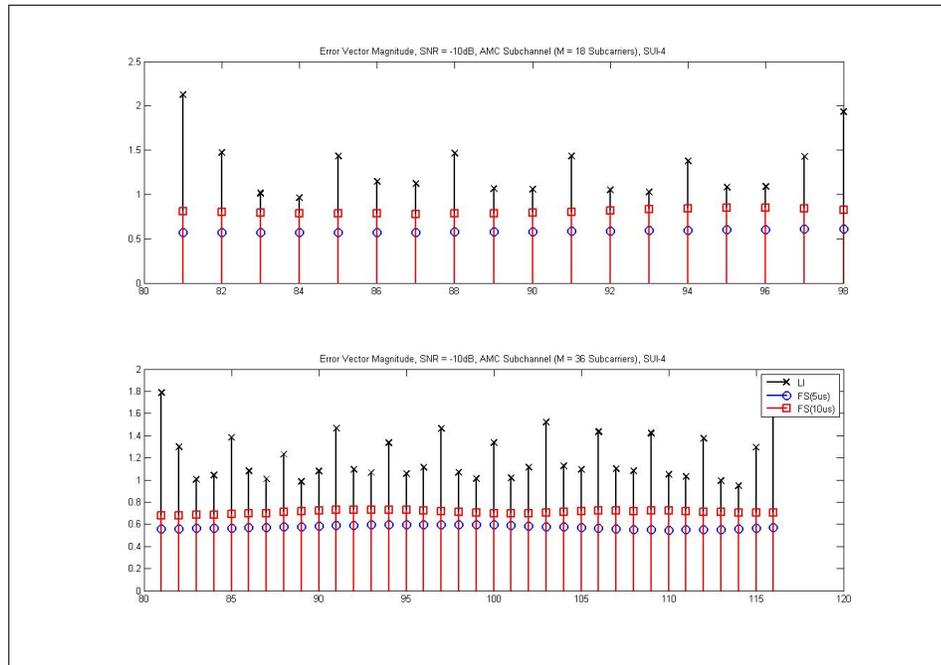


Figure 4.13: Error Vector Magnitude, SNR = -10dB, 1 AMC Subchannel (M = 18,36 Subcarriers), SUI-4

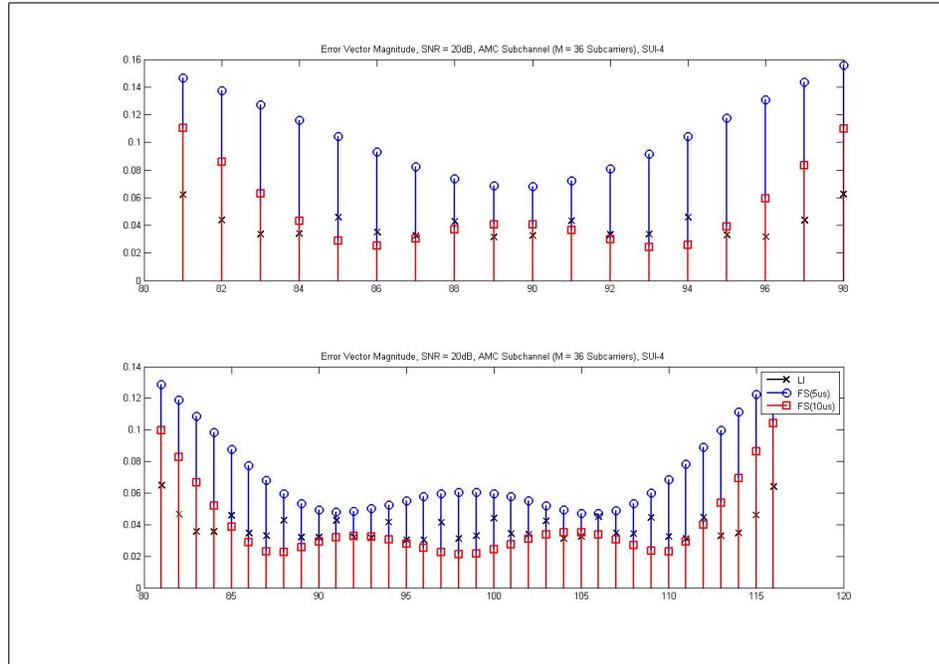


Figure 4.14: Error Vector Magnitude, SNR = 20dB, 1 AMC Subchannel (M = 18,36 Subcarriers), SUI-4

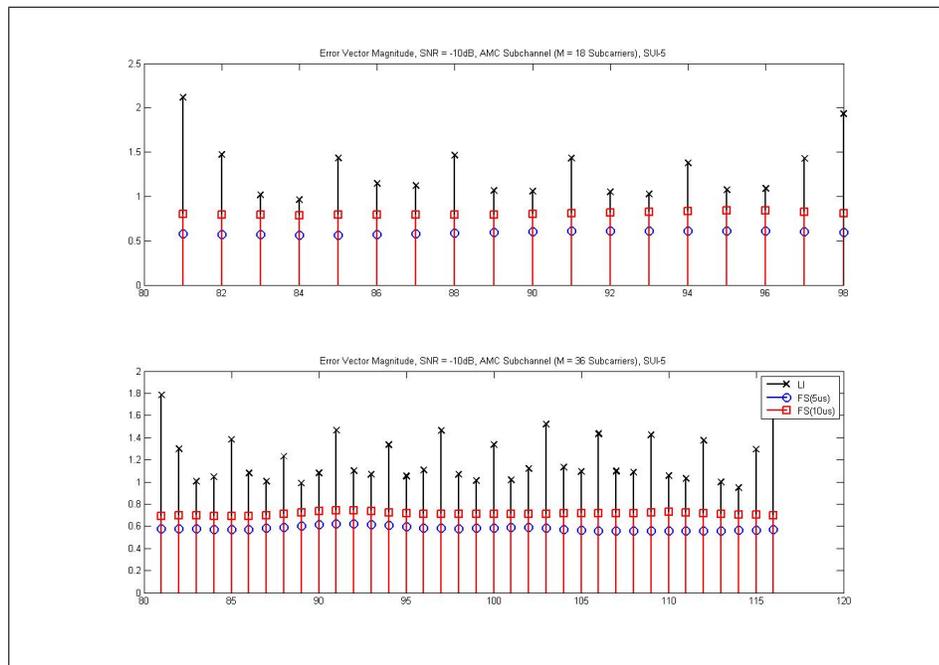


Figure 4.15: Error Vector Magnitude, SNR = -10dB, 1 AMC Subchannel (M = 18,36 Subcarriers), SUI-5

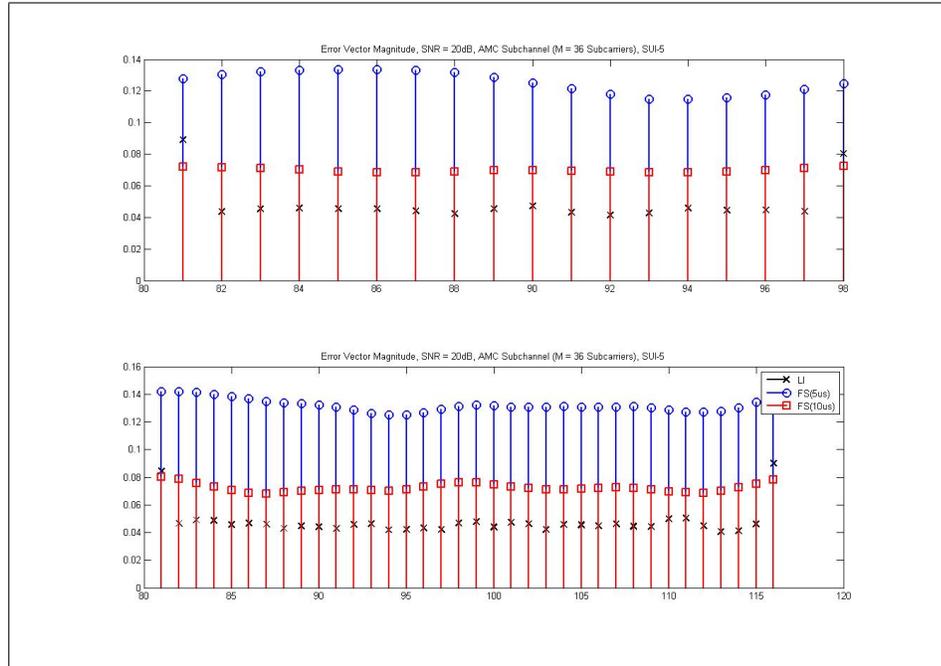


Figure 4.16: Error Vector Magnitude, SNR = 20dB, 1 AMC Subchannel (M = 18,36 Subcarriers), SUI-5

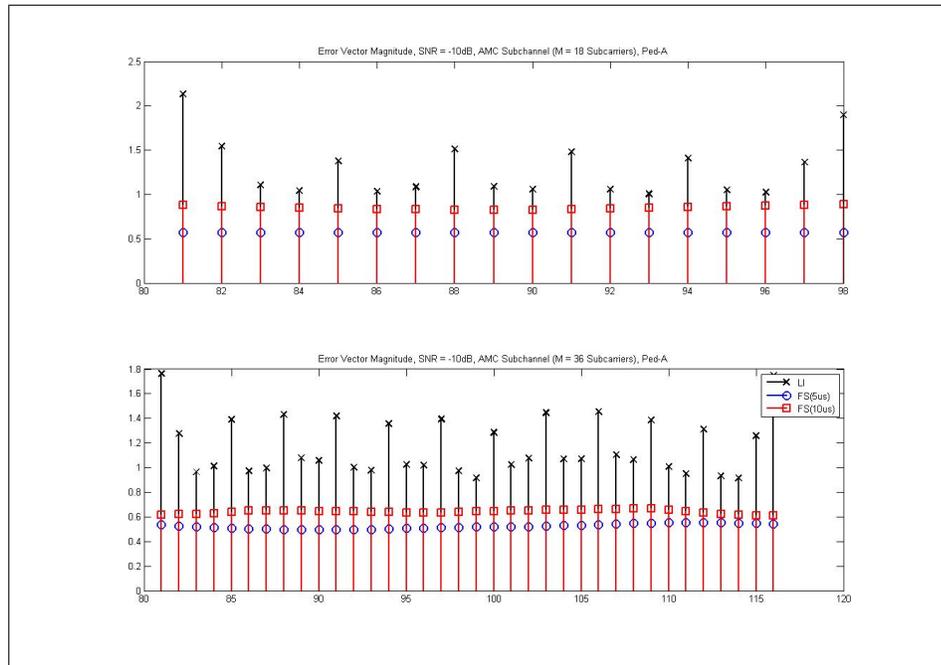


Figure 4.17: Error Vector Magnitude, SNR = -10dB, 1 AMC Subchannel (M = 18,36 Subcarriers), Ped-A

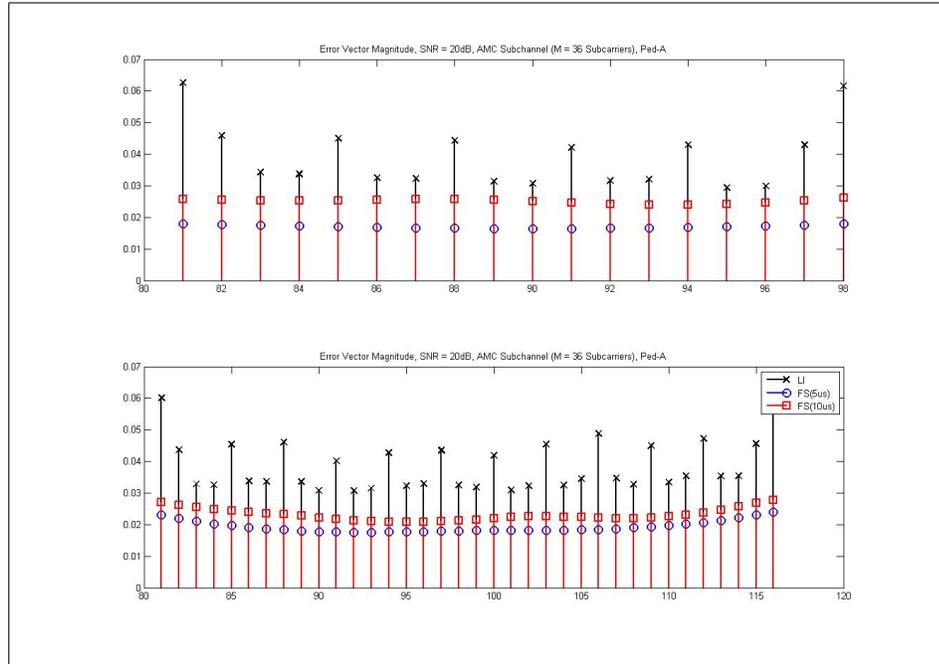


Figure 4.18: Error Vector Magnitude, SNR = 20dB, 1 AMC Subchannel (M = 18,36 Subcarriers), Ped-A

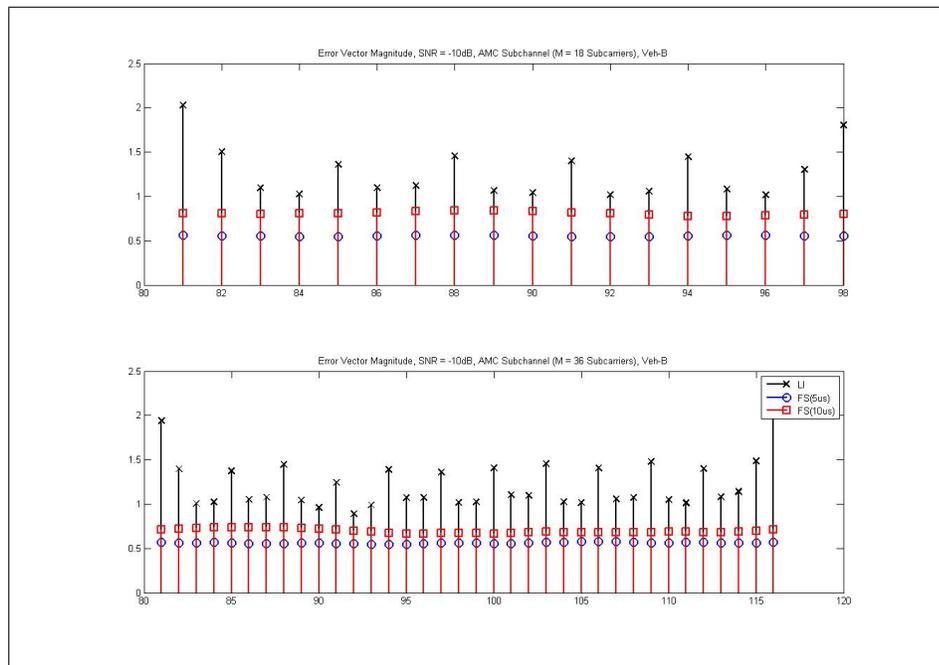


Figure 4.19: Error Vector Magnitude, SNR = -10dB, 1 AMC Subchannel (M = 18,36 Subcarriers), Veh-B

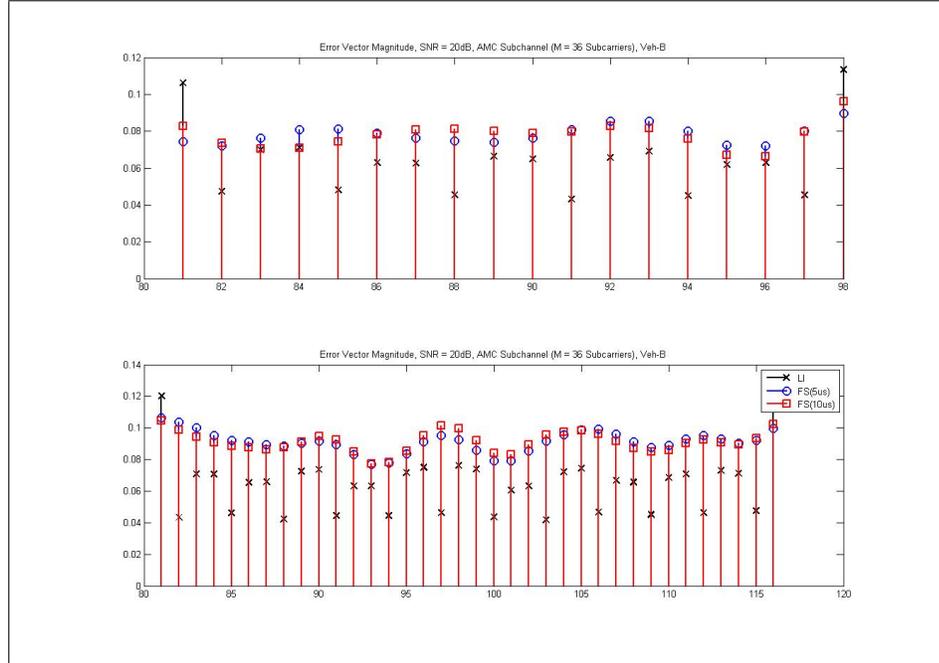


Figure 4.20: Error Vector Magnitude, SNR = 20dB, 1 AMC Subchannel ( $M = 18,36$  Subcarriers), Veh-B

### Observations: Error Vector Magnitude

At low SNR, the channel estimates at the pilot subcarriers are worse than the channel estimates that are linearly interpolated. This result occurs because at low SNR, noise is the dominant contributor to distortion. When two noise-corrupted pilots are used to interpolate the channel at the subcarriers in between, some of the noise can be averaged out interpolating these two pilots. And while linearly interpolating the pilots will introduce interpolation error, the error due to noise will dominate in low SNR channels. We can support this observation by comparing the noise terms in Equation 4.4 and 4.5. In addition, the channel estimates at the pilot subcarriers are worse than the channel estimates that are linearly interpolated in SUI-1 and Ped-A even at high SNR. In these channels, the frequency-domain channel

response is relatively flat so the interpolation error is very small. Because of this, the error at each subcarrier is dependent on the noise term and the interpolated subcarriers have a smaller noise term.

In these plots, we also see that in the frequency smoothing algorithm, the estimates at the edge of each subchannel are worse than the estimates in the middle in SUI-4, SUI-5, and Vehicular-B at high SNR. The result of this is caused by the edge effect. This occurs because applying a rectangular window to the estimates in the time domain is equivalent to applying a circular convolution with a sinc function to the estimates in the frequency domain. This means that  $\hat{H}(-k) = \hat{H}(M - k)$ . Since the baseline channel estimates are circularly convolved with a sinc function, the estimates at the edge subcarriers are generated by using channel estimates at both ends of the subchannel. Equation 4.17 shows this mathematically.

$$\begin{aligned} \hat{H}_{fs}(k) &= \text{FFT}(\hat{h}_{fs}) = (\mathbf{W} * \hat{\mathbf{H}})(k) \\ &= \sum_{l=0}^{M-1} W(l) \hat{X}(k - l) \text{ for } k = 0, 1, \dots, M-1 \end{aligned} \tag{4.17}$$

where  $\mathbf{W}$  is a sinc function,  $\hat{\mathbf{H}}$  is a vector containing the baseline channel estimates, and  $\hat{H}_{fs}(k)$  is the frequency smoothed channel estimates at subcarrier  $k$ .

Therefore, the accuracy of the channel estimates at the edges depends on the correlation between the subcarriers at the both edges of the subchannel. For higher delay spread channels like SUI-4, SUI-5, and Vehicular-B, the coherence bandwidth is smaller so the channel at the edges are uncorrelated with each other and hence, produce poor estimates at the edges. However, in SUI-1 and Ped-A, we can observe from Figure 4.12 and 4.18 that the performance at the edges is similar

to the performance in the middle of the subchannel because these channels have a larger coherence bandwidth and hence, the channel across the entire subchannel is highly correlated.

## 4.4 Conclusion

Based on the results of this experiment, it is observed that performing AMC channel estimation subchannel sizes of  $M = 18$  and  $M = 36$  adjacent subcarriers using the frequency smoothing channel estimator is better than the linear interpolator. We see that using a time window to reject noise and eliminate images of the channel impulse response is better in all channel conditions except those with high delay spread and high SNR. We also observed that there is a noise rejection and bias tradeoff from using the frequency smoothing algorithm. While a longer window creates a less biased estimate, it also captures more noise. Because of this, the selection of the window length will depend on the delay spread of the channel and SNR.

# Chapter 5

## Channel Estimation in Channel Sounding

### 5.1 Introduction

Channel sounding is a simple technique that enables the base station to obtain a snapshot of each user's unique channel conditions. This feature is often used with AMC to aid the scheduler in allocating resources to optimize performance. Figure 5.1 illustrates how channel sounding is enabled and utilized.

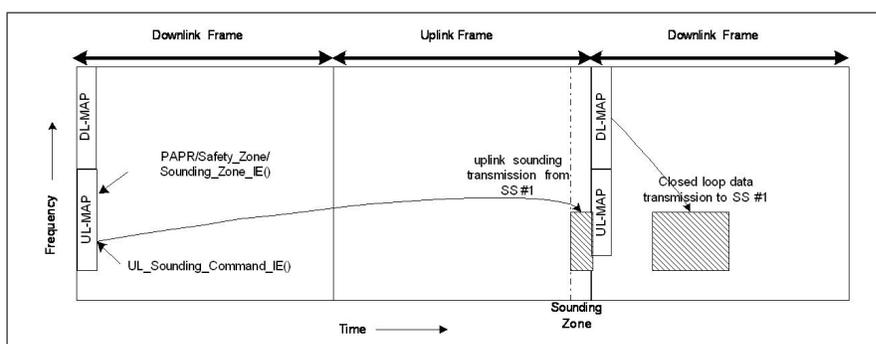


Figure 5.1: Frame Sequence of Channel Sounding [16]

To enable uplink channel sounding, the base station allocates an UL sounding zone within a frame (typically at the end of the UL frame) and sends a command on the DL to one or more user to transmit known sounding signals to the base station within this sounding zone. In the standard, the sounding zone can be composed of 1 to 8 OFDM symbols per uplink frame and partitioned into non-overlapping sounding bands that contain 18 consecutive subcarriers. The 18 subcarriers per sounding band coincides with the size of an AMC subchannel. A user may be requested to transmit sounding signals in a set of adjacent non-overlapping sounding bands or possibly over the entire sounding zone. If users are commanded to transmit sounding sequences over the entire sounding zone, it is possible to multiplex multiple sounding signals in the same time-frequency resource by using decimation separability or cyclic-shift separability to maintain signal orthogonality. In decimation separability, users transmit every  $D$ th subcarrier where  $D$  is the spacing between two occupied subcarriers. In cyclic-shift separability, users transmit in every subcarrier but adjust their transmitted sequence by a phase shift to maintain orthogonality. In this chapter, we will evaluate the performance of channel estimation algorithms for different size sounding allocations including over the entire sounding zone using decimation and cyclic-shift separability.

## 5.2 Channel Estimation in a Subset of Sounding Zone

In this section, we will develop and analyze the performance of channel estimation algorithms assuming that a subset of the sounding zone was allocated to each user and assume that the same time-frequency resource is not shared. For simplicity, we have assumed that the sounding zone only consists of 1 OFDM symbol per uplink frame and that each user is allocated  $M$  adjacent subcarriers within the sounding zone.

### 5.2.1 Frequency Smoothing

Similar to the study performed for AMC, we will again demonstrate the performance of the frequency smoothing channel estimator. The idea is the same as it was in AMC except in channel sounding, using the frequency smoothing estimator does not produce images of the channel impulse response because all the subcarriers within the sounding zone are known. Therefore, no aliasing of the impulse response can occur. Refer to Section 4.2.2 for more details.

### 5.2.2 Linear Minimum Mean-Squared Error (LMMSE) Estimator

In this algorithm, the objective is to find a linear estimator that best estimates the channel in the mean-squared error sense. This estimator will generate filter

coefficients to filter the baseline channel estimates such that the mean-squared error is minimized. A block diagram implementing the LMMSE estimator is shown in Figure 5.2.

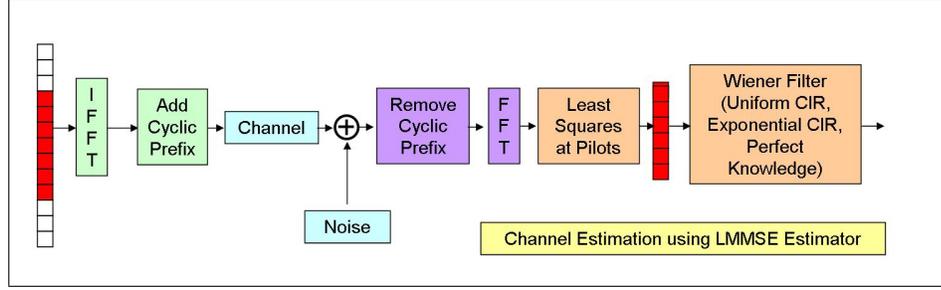


Figure 5.2: Block Diagram of WiMAX with LMMSE channel estimator

Assuming our OFDM system can be described as shown in Equation 2.14 for  $M$  subcarriers, we can derive the optimal linear estimator,  $\mathbf{A}_{opt}$  and its linear minimum mean-squared error estimates  $\hat{\mathbf{H}}_{LMMSE}$  [12].

$$\mathbf{A}_{opt} = \mathbf{R}_{\mathbf{H}}(\mathbf{R}_{\mathbf{H}} + \mathbf{X}^{-1}\mathbf{R}_{\mathbf{V}}\mathbf{X}^{-\mathbf{H}})^{-1} \quad (5.1)$$

$$\hat{\mathbf{H}}_{LMMSE} = \mathbf{A}_{opt}\hat{\mathbf{H}} \quad (5.2)$$

$$= \mathbf{R}_{\mathbf{H}}(\mathbf{R}_{\mathbf{H}} + \mathbf{X}^{-1}\mathbf{R}_{\mathbf{V}}\mathbf{X}^{-\mathbf{H}})^{-1}\hat{\mathbf{H}}$$

where  $\mathbf{R}_{\mathbf{H}}$  is the channel autocorrelation matrix,  $\mathbf{R}_{\mathbf{V}}$  is the noise autocorrelation matrix,  $\hat{\mathbf{H}} = \mathbf{X}^{-1}\mathbf{Y}$  is the least-squares channel estimate at each subcarrier, and the superscript  $(\cdot)^H$  is the Hermitian transpose. We can simplify the result if we assume that the noise components are mutually uncorrelated and have the same variance  $\sigma_v^2$  and if all the transmitted sounding signals have the same magnitude  $|X(t, k)| = A$ . Under these assumptions, we can then say that  $\mathbf{R}_{\mathbf{V}} = \sigma_v^2\mathbf{I}$  and  $\mathbf{X} = A\mathbf{I}$  where  $\mathbf{I}$

is an  $M \times M$  identity matrix. Then we can rewrite Equation 5.2 as

$$\hat{\mathbf{H}}_{\text{LMMSE}} = \mathbf{R}_{\mathbf{H}}(\mathbf{R}_{\mathbf{H}} + \frac{\sigma_v^2}{A^2}\mathbf{I})^{-1}\hat{\mathbf{H}} \quad (5.3)$$

### Low-rank Modeling using Singular Value Decomposition

One method to reduce the complexity of the LMMSE estimator is to perform a low-rank approximation on the optimal linear estimator,  $\mathbf{A}_{opt}$ , using singular value decomposition (SVD) [12]. By using SVD, we can diagonalize the channel auto-correlation matrix,  $\mathbf{R}_{\mathbf{H}}$ , by the unitary matrix  $\mathbf{Q}$  such that  $\mathbf{R}_{\mathbf{H}} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^{\mathbf{H}}$  where  $\mathbf{\Lambda} = \text{diag}([\lambda_1, \lambda_2, \dots, \lambda_M])$  contain the singular values of  $\mathbf{R}_{\mathbf{H}}$ . If we substitute this quantity into Equation 5.1, then we obtain

$$\begin{aligned} \mathbf{A}_{opt} &= \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^{\mathbf{H}}\{\mathbf{Q}(\mathbf{\Lambda} + \frac{\sigma_v^2}{A^2}\mathbf{I})\mathbf{Q}^{\mathbf{H}}\}^{-1} \\ &= \mathbf{Q}\mathbf{\Lambda}(\mathbf{\Lambda} + \frac{\sigma_v^2}{A^2}\mathbf{I})^{-1}\mathbf{Q}^{\mathbf{H}} \\ &= \mathbf{Q}\mathbf{\Delta}\mathbf{Q}^{\mathbf{H}} \end{aligned} \quad (5.4)$$

where  $\mathbf{\Delta} = \text{diag}([\delta_1, \delta_2, \dots, \delta_M])$ . If we arrange  $\mathbf{\Delta}$  such that  $\delta_1 \geq \delta_2 \geq \dots \geq \delta_M$ , then we can choose the  $p$  (less than  $M$ ) largest values and zero out the remaining singular values such that  $\mathbf{\Delta}$  has diagonal values equal to

$$\delta_m = \begin{cases} \frac{\lambda_m}{\lambda_m + \frac{\sigma_v^2}{A^2}} & \text{for } m = 1, 2, \dots, p \\ 0 & \text{for } m = p+1, p+2, \dots, M \end{cases} \quad (5.5)$$

Because we are using a low-rank approximation of  $\mathbf{R}_{\mathbf{H}}$ , we must use the pseudoinverse to calculate the optimal linear estimator. Therefore, the optimal low-rank linear estimator is defined as

$$\mathbf{A}_{opt} = \mathbf{Q}\mathbf{\Lambda}(\mathbf{\Lambda} + \frac{\sigma_v^2}{A^2}\mathbf{I})^\dagger\mathbf{Q}^{\mathbf{H}} \quad (5.6)$$

where  $(\cdot)^\dagger$  is the pseudoinverse operator.

## Performance Analysis

From Equation 5.1, we can see that in order to find the optimal LMMSE estimator,  $\mathbf{A}_{opt}$ , we need to perform a matrix inversion each time we need to generate an estimate. This operation is often too computationally complex for use in a real-time system so algorithms requiring this operation are often ignored. To workaroud this issue, we can assume a certain channel power delay profile and noise power and use them to calculate  $\mathbf{A}_{opt}$ . However, we will not know the true channel and noise power in general so this workaroud introduces model mismatch errors. But it is shown in [12] that the channel estimation error due to model mismatch is small even if we assume a uniform power delay profile (worst-case channel) and use low-rank modeling. This result comes from the fact that because each OFDM symbol duration is long relative to most channel power delay profiles encountered, most of the channel's power will be contained in the first  $p$  singular values. As for the noise mismatch, it is described in [12] that a design for high SNR channels is preferred because the distortion due to noise is small and therefore more advantageous to minimize the channel estimation error.

An expression for the MSE of a rank- $p$  estimate is also provided in [12] that is used to evaluate the performance of the LMMSE estimator. The expression is given as

$$MSE_{LMMSE}(p) = \frac{1}{M} \sum_{m=1}^p \left[ \mu_m (1 - \delta_m)^2 + \frac{\sigma_v^2}{A^2} \delta_m^2 \right] + \frac{1}{M} \sum_{m=p+1}^M \mu_m \quad (5.7)$$

where  $\delta_m$  is defined in Equation 5.5 and  $\mu_m$  are the singular values of the pre-

calculated channel autocorrelation matrix. If we have no mismatch, we can replace  $\mu_m$  with  $\lambda_m$

From Equation 5.7, we can reduce the impact of noise by approximately a factor of  $p/M$  if we use low-rank modeling. However, choosing a smaller value of  $p$  introduces a larger bias error. We can view  $\frac{1}{M} \sum_{m=p+1}^M \mu_m$  as the bias error introduced from low-rank modeling. As such, there is a tradeoff in bias and noise from using low-rank modeling. However, if we choose  $p$  such that most of the channel's power is contained in the first  $p$  singular values, then the bias error will be small.

### 5.2.3 Simulation: Subset of Sounding Zone

#### Objective

In this experiment, our objective was to compare the performance of the frequency smoothing algorithm to the LMMSE estimator under different channel conditions. We do not have a link-level simulator with all the components of a WiMAX system so we use the MATLAB simulator in Figure 4.5. Again, we use average MSE and the error vector magnitude to evaluate the performance of each algorithm.

$$EVM(k) = |\hat{H}(t, k) - H(t, k)| \text{ for } k = 0, 1, \dots, M-1 \quad (5.8)$$

$$MSE_{dB} = 10 \cdot \log_{10} \left\{ \frac{1}{M} \sum_{k=0}^{M-1} |\hat{H}(t, k) - H(t, k)|^2 \right\} \quad (5.9)$$

## Simulation Setup

In this experiment, we assumed that a user could be allocated between 1 to 4 sounding bands ( $M = 18, 36, 54, 72$ ) and that the sounding zone is only composed of 1 OFDMA symbol. We have also assumed sounding sequence transmitted at each subcarrier has the same amplitude and is known to the receiver. We also needed to define parameters for each channel estimation algorithm. For the frequency smoothing algorithm, we chose to test with a  $5\mu\text{s}$  and  $10\mu\text{s}$  window because  $5\mu\text{s}$  represents the delay spread of a typical urban environment and  $10\mu\text{s}$  is close to the nominal cyclic prefix size for WiMAX. For the LMMSE estimator, we chose to test with precomputed channel autocorrelation matrices for a uniform power delay profile with  $5\mu\text{s}$  delay spread, a uniform power delay profile with  $10\mu\text{s}$  delay spread, an exponential power delay profile with  $1\mu\text{s}$  RMS delay spread, and an exponential power delay profile with  $2.5\mu\text{s}$  RMS delay spread. We chose to test the LMMSE estimator assuming the channel had a uniform power delay profile and exponential power delay profile because the uniform power delay profile represents the worst-case channel and the exponential power delay profile is more representative of a typical urban channel. We also chose to set the parameter  $p$  such that the first  $p$  singular values capture at least 99 percent of the total channel's energy. A table of simulation parameters is shown in Table 5.1 and 5.2

Table 5.1: Channel Sounding Simulation Parameters

<b>System Parameters</b>	
Parameter	Value
Bandwidth Scale Factor	8 (1024 FFT)
Fundamental Bandwidth	1.25MHz
Cyclic Prefix	1/8 (11.42 $\mu$ s)
Number of Tx Antennas	1
Number of Rx Antennas	2
Sampling Rate	11.2MHz
Symbol Duration	102.9 $\mu$ s
<b>Sounding Parameters</b>	
Parameter	Value
Number of Sounding Bands	1,2,3,4 (M=18,36,54,72)

Table 5.2: Channel Estimation Parameters

<b>Frequency Smoothing Parameters</b>	
Parameter	Value
Window Size	$5\mu s, 10\mu s$
<b>LMMSE Estimator Parameters</b>	
Parameter	Value
Number of Singular Values ( $p$ )	$0.99 \cdot \text{Sum of Singular Values}$
Uniform ( $5\mu s$ Delay Spread)	$P_h(\tau) = \begin{cases} 1 & \text{for } 0 \leq \tau \leq 5\mu s \\ 0 & \text{otherwise} \end{cases}$
Uniform ( $10\mu s$ Delay Spread)	$P_h(\tau) = \begin{cases} 1 & \text{for } 0 \leq \tau \leq 10\mu s \\ 0 & \text{otherwise} \end{cases}$
Exponential ( $1\mu s$ RMS Delay Spread)	$P_h(\tau) = \begin{cases} \exp(\frac{-\tau}{1\mu s}) & \text{for } 0 \leq \tau \leq 5\mu s \\ 0 & \text{otherwise} \end{cases}$
Exponential ( $2.5\mu s$ RMS Delay Spread)	$P_h(\tau) = \begin{cases} \exp(\frac{-\tau}{2.5\mu s}) & \text{for } 0 \leq \tau \leq 10\mu s \\ 0 & \text{otherwise} \end{cases}$

## Results: Mean-Squared Error

Below are MSE curves comparing the Frequency Smoothing and LMMSE estimator. We only show the MSE plots assuming  $M = 72$  since results do not vary much for the other sounding zone sizes. For reference, we also included the performance of the LMMSE estimator assuming that we had perfect knowledge of the channel autocorrelation matrix and SNR. It sets an upper limit to the achievable performance that can be obtained in WiMAX. This is used to illustrate the performance gap between our proposed algorithms and one that would be able to perfectly adapt to the changes in the channel.

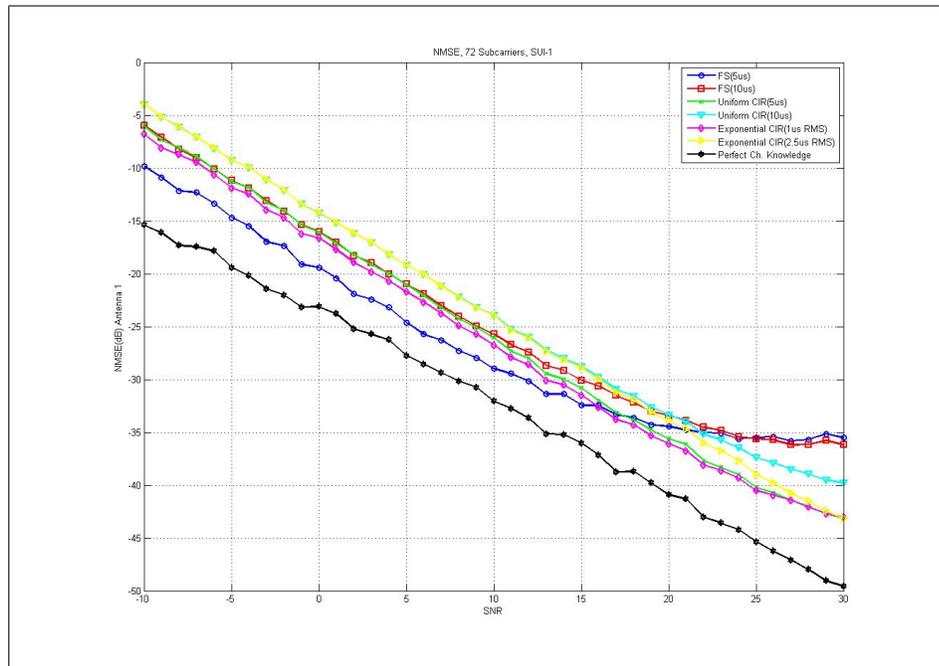


Figure 5.3: Average MSE in Channel Sounding in Subset of Sounding Zone ( $M = 72$ ), Frequency Smoothing vs. LMMSE, SUI-1

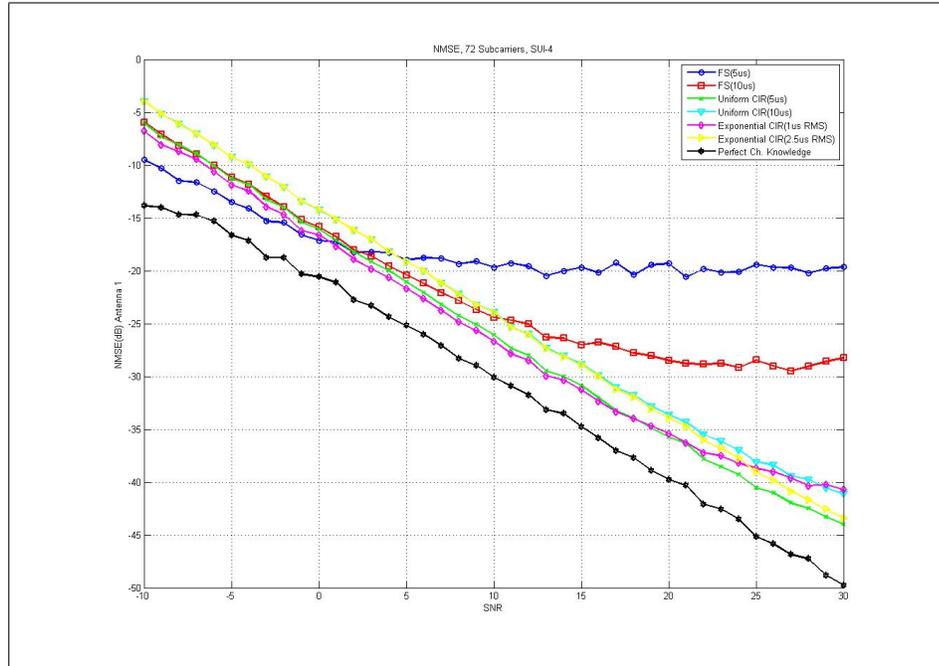


Figure 5.4: Average MSE in Channel Sounding in Subset of Sounding Zone ( $M = 72$ ), Frequency Smoothing vs. LMMSE, SUI-4

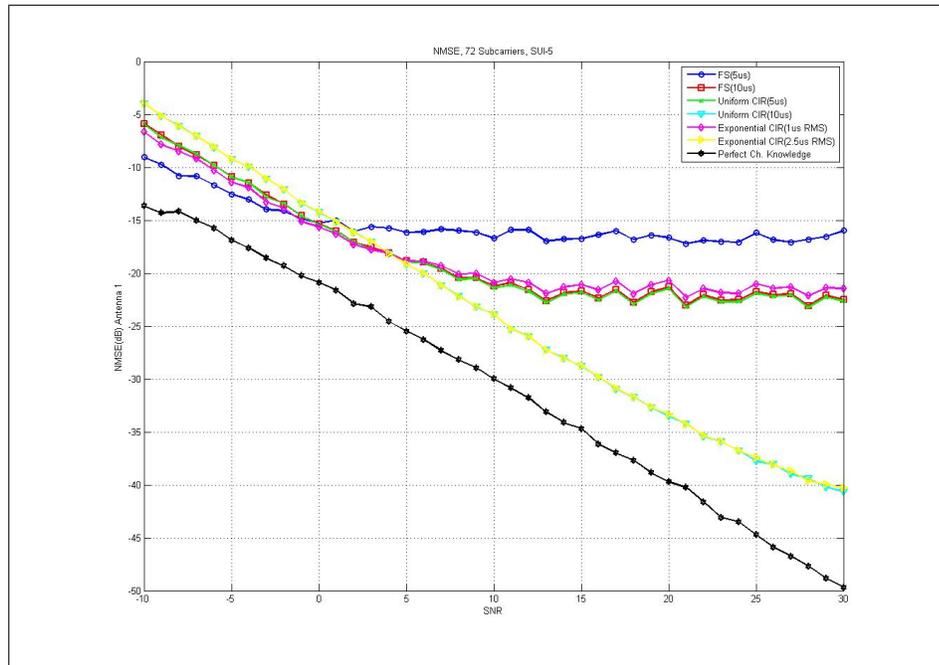


Figure 5.5: Average MSE in Channel Sounding in Subset of Sounding Zone ( $M = 72$ ), Frequency Smoothing vs. LMMSE, SUI-5

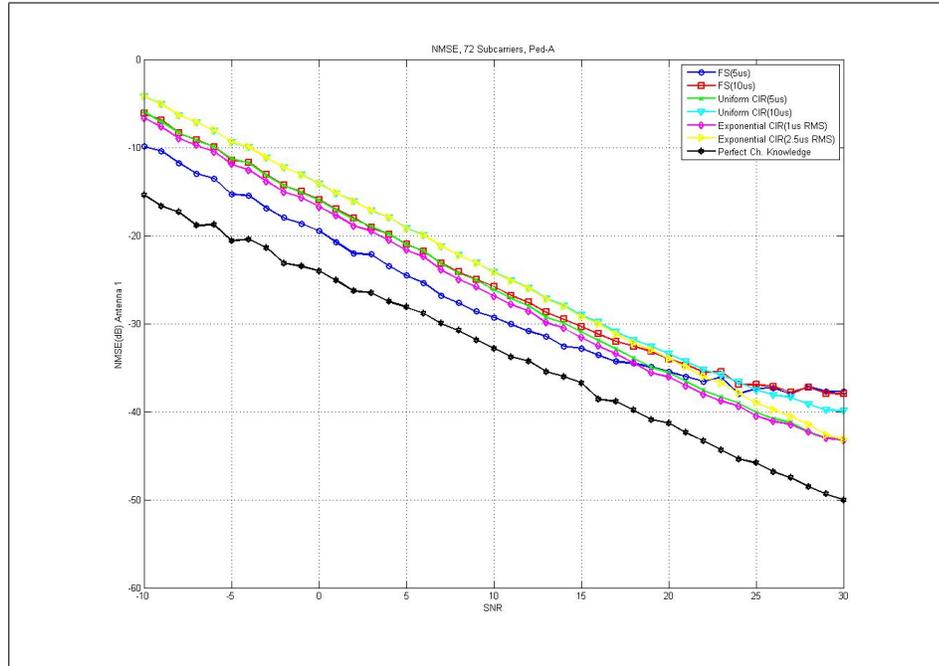


Figure 5.6: Average MSE in Channel Sounding in Subset of Sounding Zone, Frequency Smoothing vs. LMMSE, ( $M = 72$ ), Ped-A

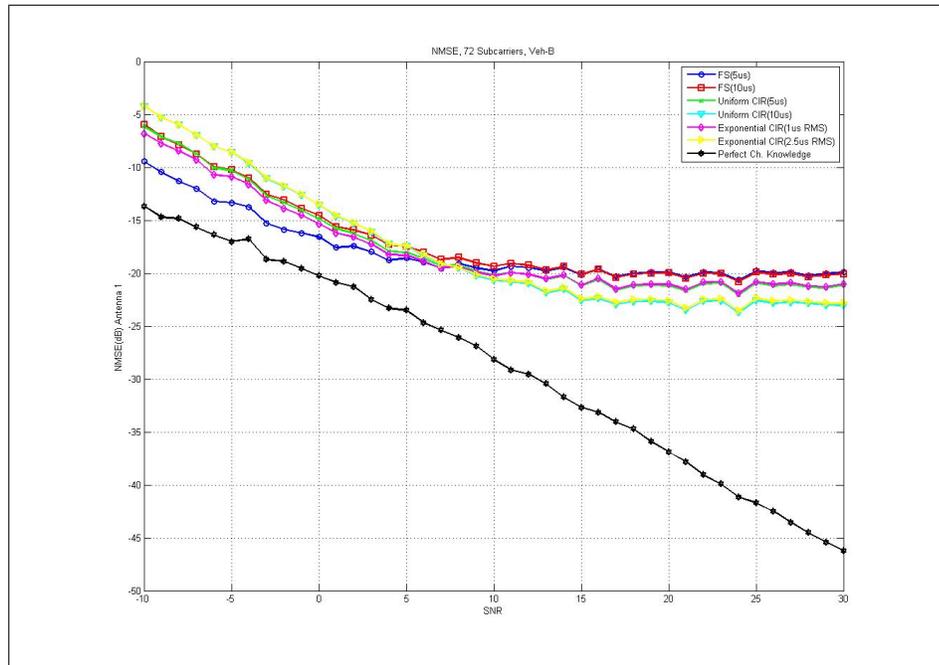


Figure 5.7: Average MSE in Channel Sounding in Subset of Sounding Zone ( $M = 72$ ), Frequency Smoothing vs. LMMSE, Veh-B

### Observations: Mean-Squared Error

When comparing the frequency smoothing and LMMSE estimator performance to the LMMSE estimator with perfect channel knowledge, we see that there is generally at least a 5dB improvement in MSE from having perfect channel knowledge. In the higher delay spread channels, we see the curves diverge as the SNR increases. This is attributed to the bias and noise error in each estimator.

We also observe that the performance of the frequency smoothing channel estimator is very similar to what we observed in AMC. First, we see that in SUI-4, SUI-5, and Vehicular-B, that there is a crossover point between the frequency smoothing with  $5\mu s$  window and frequency smoothing with  $10\mu s$  window. This illustrates the tradeoff between bias error and noise rejection. Using the smaller window size rejects more noise but creates more bias in the channel estimates. For low SNR channels, noise is the dominant contributor to distortion and we would prefer to reject more noise in this scenario. But as the SNR increases, the channel becomes the larger contributor to distortion and so we see the larger window size begin to perform better relative to the smaller window because there is less bias. In both cases, for SUI-4, SUI-5, and Veh-B, we see that the performance of the frequency smoothing channel estimator floors even as the SNR increases because of the bias introduced by the window. See Appendix B for more details.

We also see that the frequency smoothing algorithm outperforms the LMMSE estimator at low SNR. We can attribute this to the fact that more noise is rejected using this algorithm. We can attribute this mostly to the fact that the frequency

smoothing window size,  $L$ , is smaller than our value of  $p$  of the LMMSE estimator. Also, we chose to design the LMMSE estimator for high SNR so we would expect the performance to be poor at low SNR. At high SNR, we see that the LMMSE estimator outperforms the frequency smoothing algorithm except for in SUI-1 and Pedestrian-A. This is attributed to the fact that there is very little bias in the frequency smoothing algorithm for these channels since the window captures almost all of the channel's energy. Combined with its ability to reject noise better, this algorithm generally outperforms the LMMSE estimator when the channel delay spread is low. However, for the higher delay spread channels like SUI-4, SUI-5, and Vehicular-B, there is a bias introduced in the frequency smoothing and is often outperformed by the LMMSE estimator. And the performance gap is largely dependent on the channel autocorrelation matrix we choose. In SUI-4, we observe that the LMMSE estimator using the exponential power delay profile with  $1\mu\text{s}$  RMS delay spread and uniform power delay profile with  $5\mu\text{s}$  delay spread performs the best while in SUI-5 and Vehicular-B, using the exponential power delay profile with  $2.5\mu\text{s}$  RMS delay spread and uniform power delay profile with  $10\mu\text{s}$  delay spread performs the best. First, the LMMSE estimator performs better than the frequency smoothing algorithm at high SNR because we choose a rank  $p$  estimator such that most of the channel's power lies in the first  $p$  singular values. Because of this, the LMMSE estimator is less biased than the frequency smoothing algorithm. Second, we also see that a good approximation of the channel delay spread when calculating the channel autocorrelation matrix can make a significant difference in performance. The channel delay spread determines how the channel's power is distributed among

its singular values. Hence, an accurate measurement of the delay spread can significantly reduce the model mismatch error. We see that in the SUI-5 channel, the performance gap is large between using channels with  $10\mu\text{s}$  delay spread versus  $5\mu\text{s}$ . We also see that in Vehicular-B, all LMMSE estimators tested performed about the same because neither estimator estimates the channel delay spread accurately. The estimators using the exponential PDP with  $2.5\mu\text{s}$  RMS delay spread and uniform PDP with  $10\mu\text{s}$  delay spread performs slightly better because their approximation of the channel delay spread is better. In addition, we also see that approximating the shape of the power delay profile more accurately does not make a significant difference. The channels that we tested all have exponential-like power delay profiles so we expected that using an exponential PDP to calculate the channel autocorrelation matrix to perform better. However, results show that there is almost no difference between using an exponential PDP and the uniform PDP in all channels. This observation was made by [12] in which if we use low-rank modeling, the error due to model mismatch will be small as long as we choose a cutoff  $p$  large enough to capture most of the channel's energy.

### **Results: Error Vector Magnitude**

To observe the error at each subcarrier, we also plotted the error vector magnitude. Results are shown below comparing the error at each subcarrier for SNR = 20dB is shown below. We only display the EVM plots for  $M = 72$ .

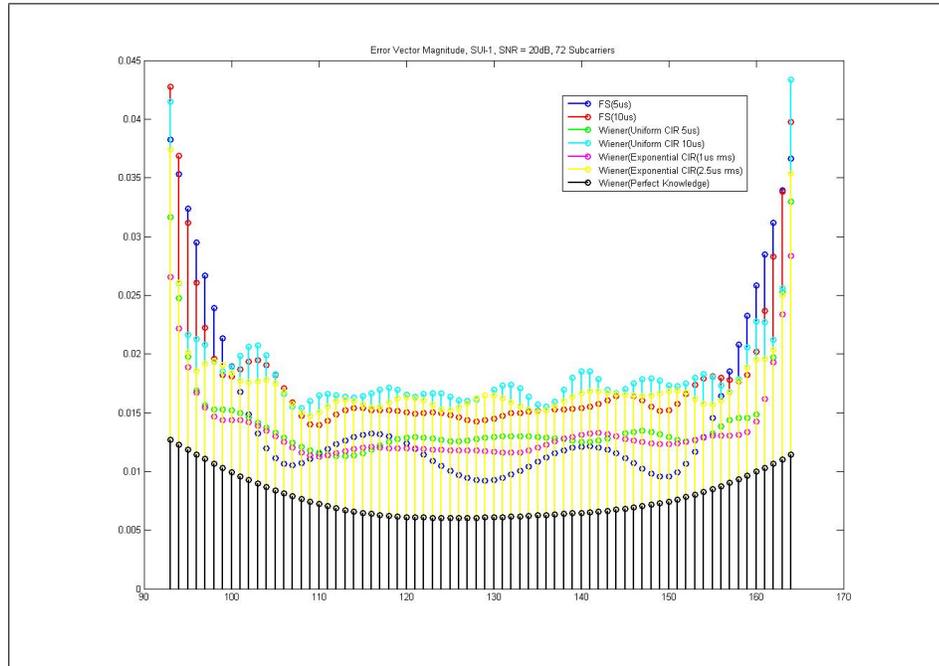


Figure 5.8: Error Vector Magnitude of Frequency Smoothing vs. LMMSE, SNR = 20dB, M = 72, SUI-1

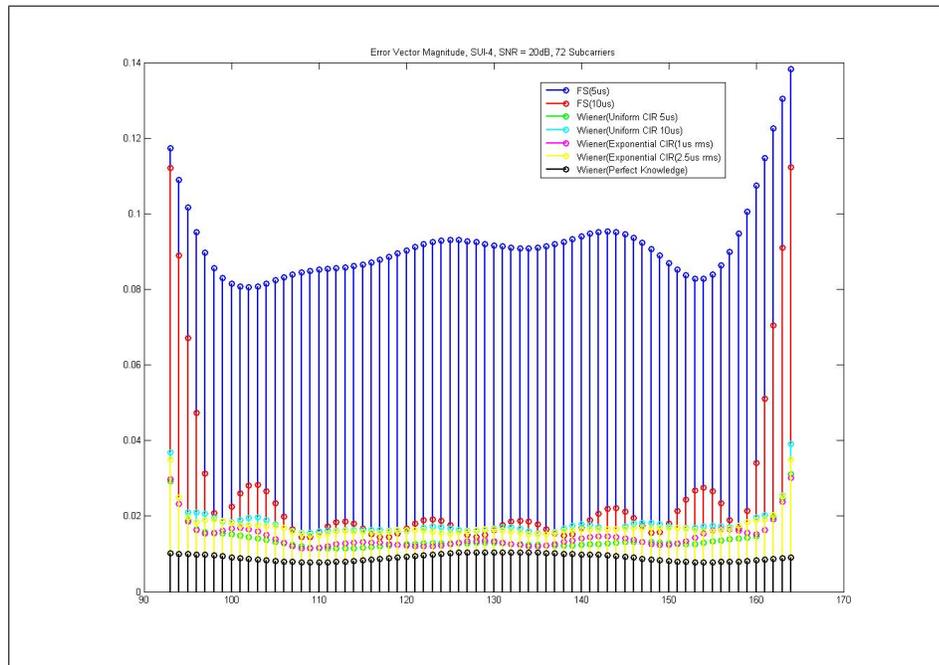


Figure 5.9: Error Vector Magnitude of Frequency Smoothing vs. LMMSE, SNR = 20dB, M = 72, SUI-4

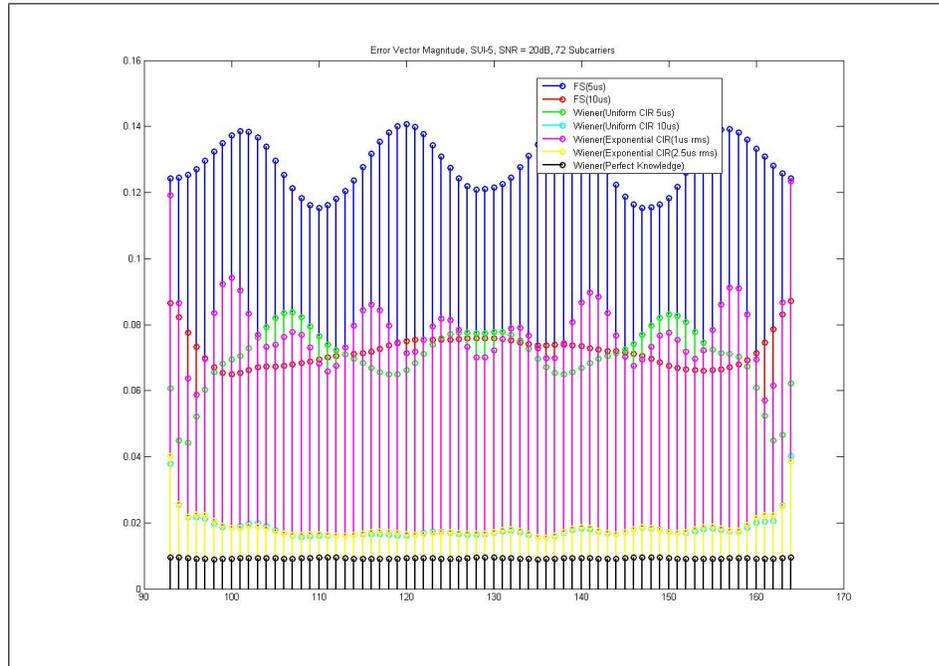


Figure 5.10: Error Vector Magnitude of Frequency Smoothing vs. LMMSE, SNR = 20dB, M = 72, SUI-5

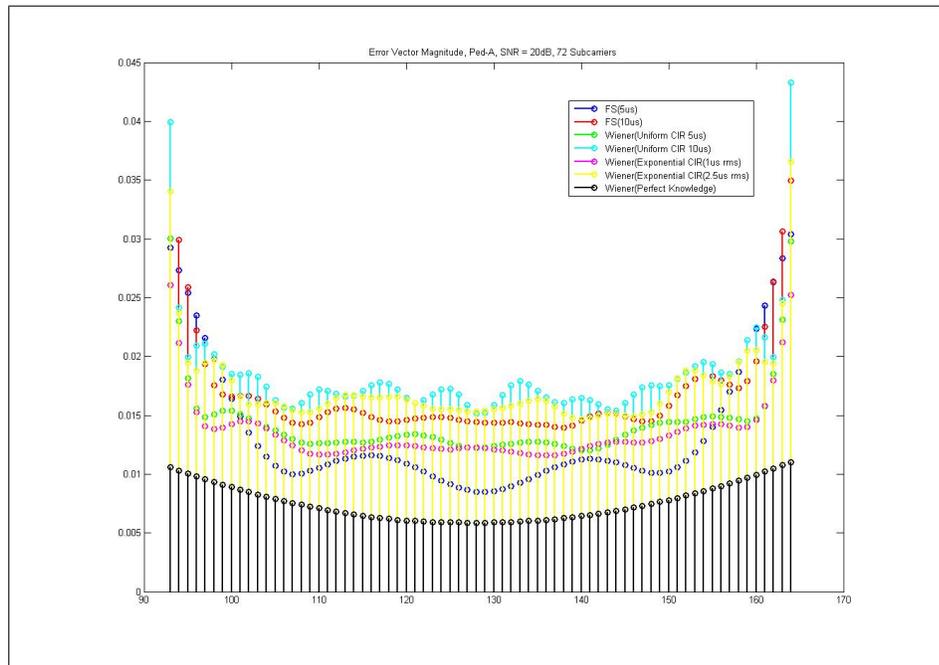


Figure 5.11: Error Vector Magnitude of Frequency Smoothing vs. LMMSE, SNR = 20dB, M = 72, Ped-A

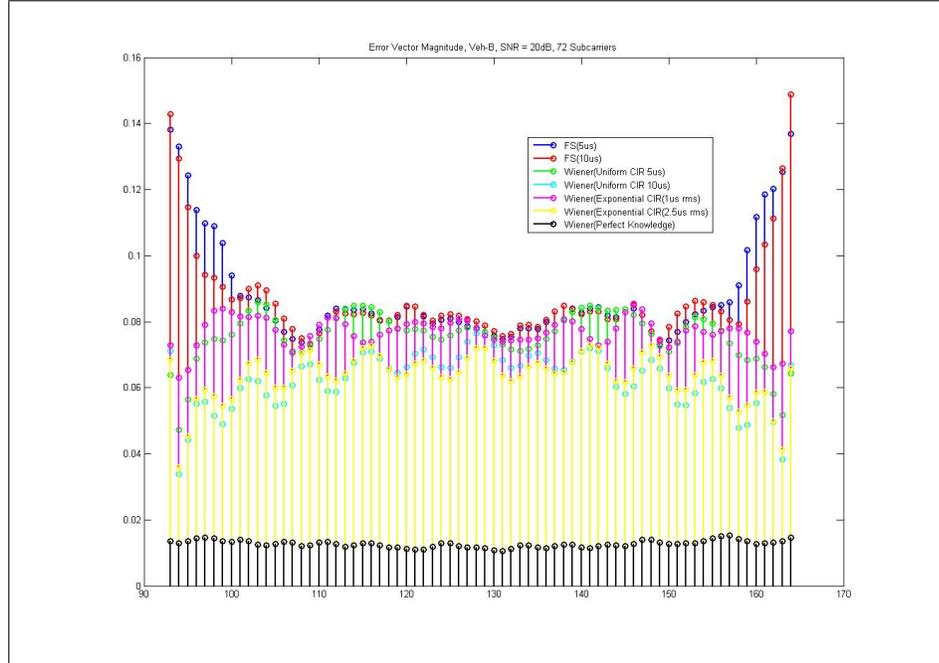


Figure 5.12: Error Vector Magnitude of Frequency Smoothing vs. LMMSE, SNR = 20dB, M = 72, Veh-B

### Observations: Error Vector Magnitude

One observation made from these plots is that the LMMSE estimates at the subcarriers at each edge of the sounding zone were generally better than their respective frequency smoothing channel estimates at SNR = 20dB. In other words, at the edges, the LMMSE estimator assuming uniform CIR (max delay spread  $5\mu\text{s}$ ) and exponential CIR ( $1\mu\text{s}$  RMS delay spread) were better than the frequency smoothing with  $5\mu\text{s}$  window and the LMMSE estimator assuming uniform CIR (max delay spread  $10\mu\text{s}$ ) and exponential CIR ( $2.5\mu\text{s}$  rms delay spread) were better than the frequency smoothing with  $10\mu\text{s}$  window. For the subcarriers in the middle of the sounding zone, the results were generally mixed. This is the result of the edge effect that occurs when using the frequency smoothing algorithm. Because we apply a

rectangular window to the estimates in the time domain, we are equivalently performing a filtering operation in the frequency-domain through a circular convolution of a sinc function with the channel. So when we estimate the channel at subcarriers at the edge of the sounding zone, we perform the estimate by interpolating the channel from subcarriers at both ends of the sounding zone. This effect is more severe in channels with higher delay spreads because the channel at the edges of the sounding zone are more uncorrelated. We can see from the error vector magnitude plots that the difference in error between the subcarriers at the edges and the subcarriers in the middle of the sounding zone is small in SUI-1 and Pedestrian-A because for low delay spread channels, the coherence bandwidth is large so the channel correlation is still high between subcarriers at both ends of the sounding zone.

### **Conclusion: Channel Estimation for a Subset of Sounding Zone**

The results of this experiment seem to suggest that we should use frequency smoothing at low SNR channels and the LMMSE estimator at high SNR. More specifically, we showed that we should use the LMMSE estimator at high SNR to estimate the channel at the edge subcarriers to avoid the edge effect. We also observed that one challenge in optimizing both estimators is in determining the channel delay spread. In the frequency smoothing algorithm, choosing the appropriate window length can achieve a better performance at high SNR because it can minimize the bias while rejecting as much noise as possible. We also see in the LMMSE estimator that the approximation of the channel delay spread in calculating the channel

autocorrelation plays a large role in improving performance because the model mismatch is reduced. Without a good estimate of the channel delay spread, we saw that both estimators significantly degraded in performance.

### 5.3 Channel Estimation with Decimation Separability

In channel sounding, users can also be commanded to transmit over the entire sounding zone. But to maintain orthogonality in the signal each user will be given a decimation factor  $D$  and decimation offset  $d$ . The decimation factor represents the spacing between subcarriers allocated to the same user and the decimation offset represents the starting position of the 1st occupied subcarrier. According to [23], decimation factors of 4, 8, 16, 32, and 64 are to be supported and the decimation offset is between 0 and  $D-1$ . In this section, we will analyze the performance of the linear interpolator and frequency smoothing algorithm when users sound every  $D^{th}$  subcarrier over the entire sounding zone.

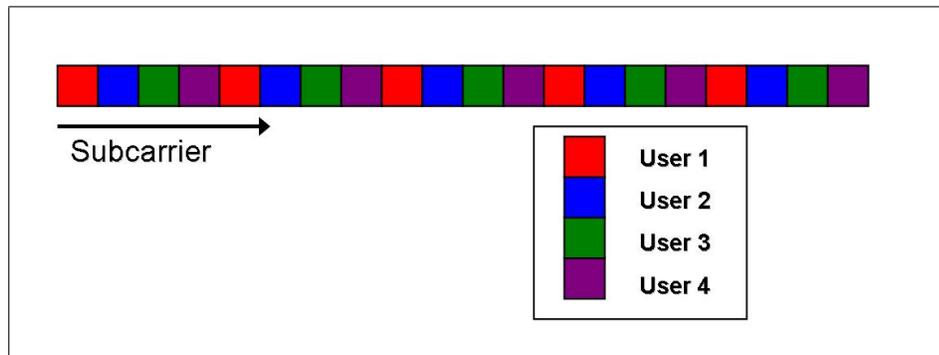


Figure 5.13: Decimation Separability Example ( $D = 4$ )

### 5.3.1 Linear Interpolation

When using linear interpolation, we must first perform the least-squares channel estimate at the subcarriers in which a sounding signal was transmitted and then interpolate the channel estimate linearly. At the ends of the sounding zone, some extrapolation will need to be done depending on the decimation offset. If we let  $P$  be the total number of occupied subcarriers, then the linear interpolator channel estimator in channel sounding with decimation separability can be summarized as such.

1. Generate Baseline Channel Estimates using Least Squares estimation at Occupied Subcarriers

$$\hat{H}(t, d + Dp) = \frac{Y(t, d + Dp)}{X(t, d + Dp)}, p = 0, 1, \dots, P-1 \quad (5.10)$$

2. Interpolate Channel Estimates in Frequency

$$\hat{H}(t, d + Dp + l) = \frac{D-l}{D}\hat{H}(t, d + Dp) + \frac{l}{D}\hat{H}(t, d + (D+1)p), l = 1, \dots, D-1 \quad (5.11)$$

3. Extrapolate Channel Estimates at Subcarriers at Edges of Sounding Zone

$$\begin{aligned} \hat{H}(t, d - l) &= \frac{-l}{D}\hat{H}(t, d + 2D) + \frac{D+l}{D}\hat{H}(t, d + D) \\ &\text{for } l = 1, \dots, d \\ \hat{H}(t, d + (P-1)D + l) &= \frac{-l}{D}\hat{H}(t, d + (P-2)D) + \frac{D+l}{D}\hat{H}(t, d + (P-1)D) \\ &\text{for } l = 1, \dots, D-d \end{aligned} \quad (5.12)$$

Using the derivation provided by [22], we can calculate the mean-squared error at the occupied subcarriers, the interpolated subcarriers, and extrapolated subcarriers. For simplicity in calculation, we will assume that the decimation offset is 0 and that the magnitude of the transmitted sounding signal at each occupied subcarrier is the same ( $|X(t, d + Dp)| = A$  for  $p = 0, 1, \dots, P-1$ )

1. MSE at pilot subcarriers

$$\begin{aligned} MSE_P &= E[|\hat{H}(t, Dp + 1) - H(t, Dp + 1)|^2] \\ &= \frac{\sigma_v^2}{A^2} \end{aligned} \quad (5.13)$$

2. MSE at interpolated subcarriers from Step 2

$$\begin{aligned} MSE_I &= \frac{1}{D-1} \sum_{l=1}^{D-1} E[|\hat{H}(t, Dp + l) - H(t, Dp + l)|^2] \\ &= \sigma_H^2 \left\{ \frac{5D-1}{3D} R_f[0] + \frac{D+1}{6D} (R_f[D] + R_f[-D]) - \alpha \right\} \\ &\quad + \frac{2D-1}{3D} \frac{\sigma_v^2}{A^2} \end{aligned} \quad (5.14)$$

where  $R_f[\Delta k]$  is the frequency-domain correlation function defined in Equation 2.7 and

$$\alpha = \frac{1}{D-1} \sum_{l=1}^{D-1} \left\{ \frac{D-l}{D} (R_f[l] + R_f[-l]) + \frac{l}{D} (R_f[D-l] + R_f[l-D]) \right\}$$

3. MSE at extrapolated subcarriers from Step 3

$$\begin{aligned} MSE_E &= \frac{1}{D-1} \sum_{l=1}^{D-1} E[|\hat{H}(t, (P-1)D + l) - H(t, (P-1)D + l)|^2] \\ &= \sigma_H^2 \left\{ \frac{11D-1}{3D} R_f[0] - \frac{5D-1}{6D} (R_f[D] + R_f[-D]) - \beta \right\} \\ &\quad + \frac{8D-1}{3D} \frac{\sigma_v^2}{A^2} \end{aligned} \quad (5.15)$$

where

$$\beta = \frac{1}{D-1} \sum_{l=1}^{D-1} \left\{ \frac{D+l}{D} (R_f[l] + R_f[-l]) - \frac{l}{D} (R_f[D+l] + R_f[-D-l]) \right\}$$

Using Equations 5.13, 5.14, and 5.15, we can derive the average MSE for each user using linear interpolation within the sounding zone as such.

$$MSE_{LI-Dec} = \frac{1}{D} MSE_P + \left( \frac{D-1}{D} - \frac{D-1}{M} \right) MSE_I + \frac{D-1}{M} MSE_E \quad (5.16)$$

where  $M$  is the number of subcarriers in the sounding zone

When we look at average MSE for linear interpolation, we see that the contributors to channel estimation error are interpolation error and noise. From Equations 5.13, 5.14, 5.15, and 5.16, we see that the interpolation error is dependent on the decimation factor  $D$  and the frequency domain correlation function  $R_f[\Delta k]$ . When we evaluated the linear interpolator in AMC in Section 4.2.1, we compared its performance in different channels with a decimation factor of 3. In this part, we compare the performance of the linear interpolator for different decimation factors given a specific channel. The plots are shown below.

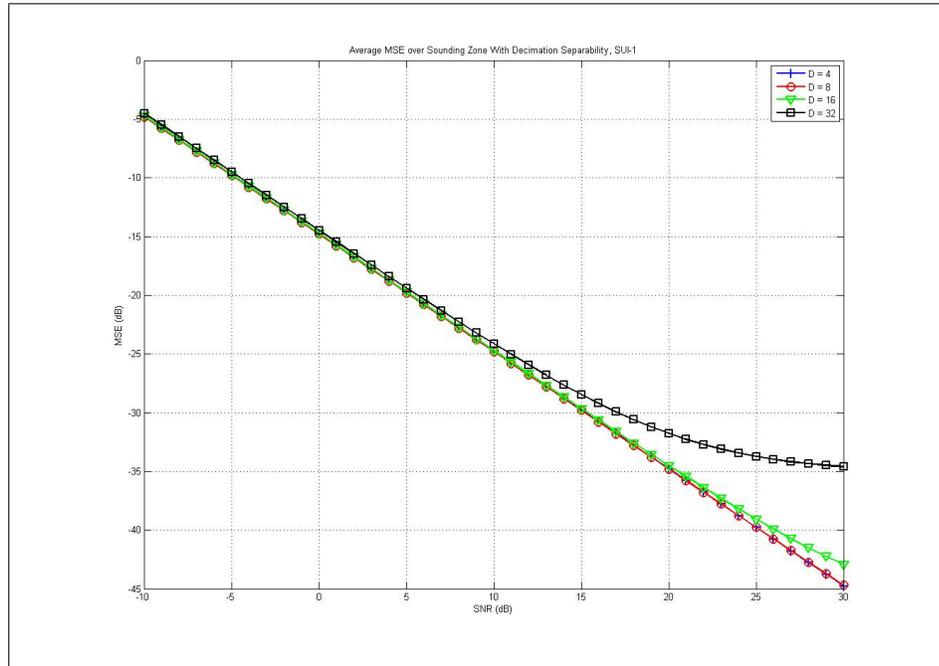


Figure 5.14: Linear Interpolation Average MSE over Sounding Zone With Decimation Separability, SUI-1

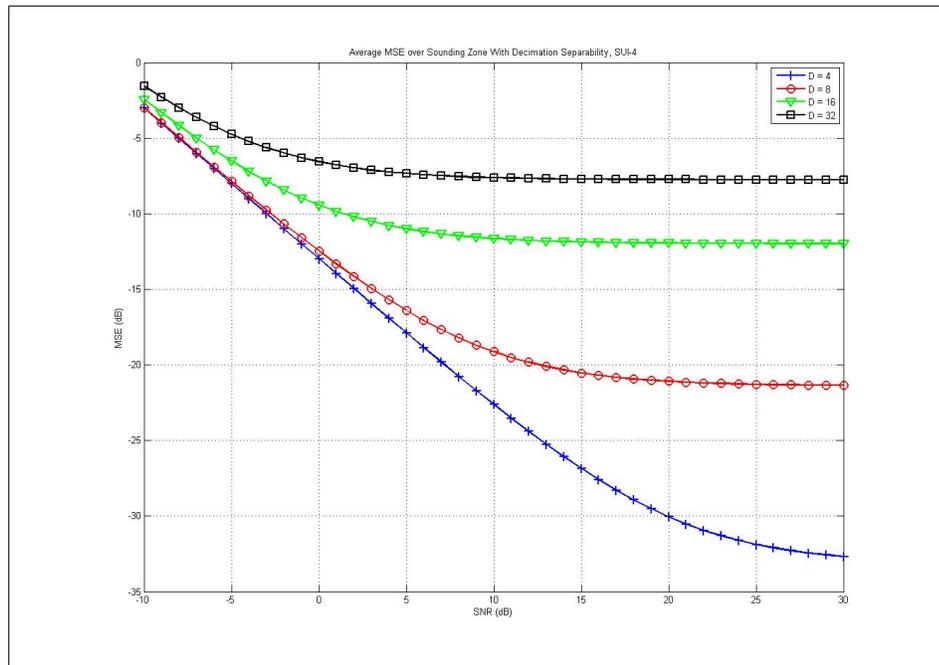


Figure 5.15: Linear Interpolation Average MSE over Sounding Zone With Decimation Separability, SUI-4

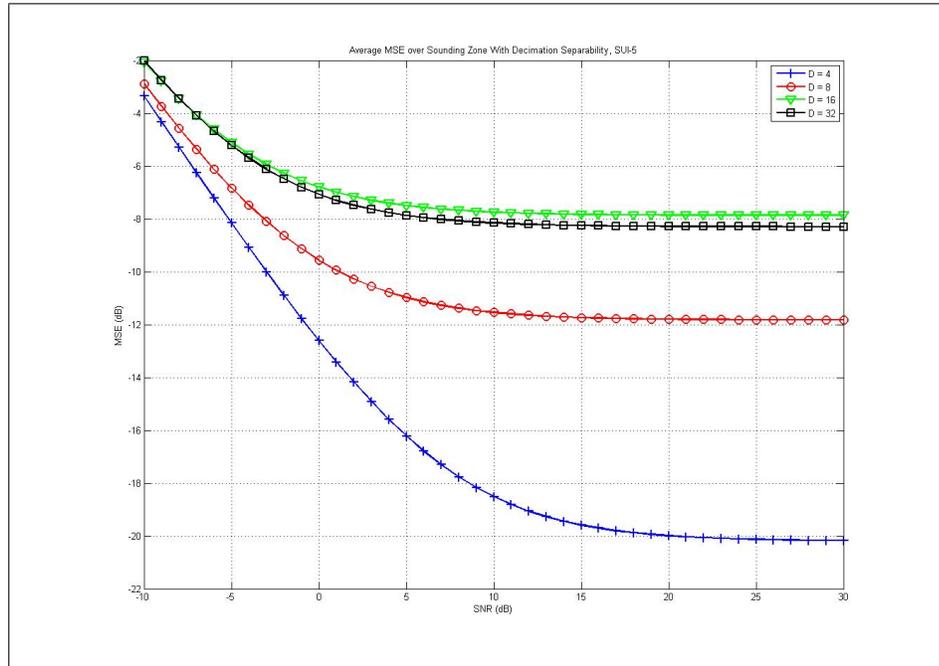


Figure 5.16: Linear Interpolation Average MSE over Sounding Zone With Decimation Separability, SUI-5

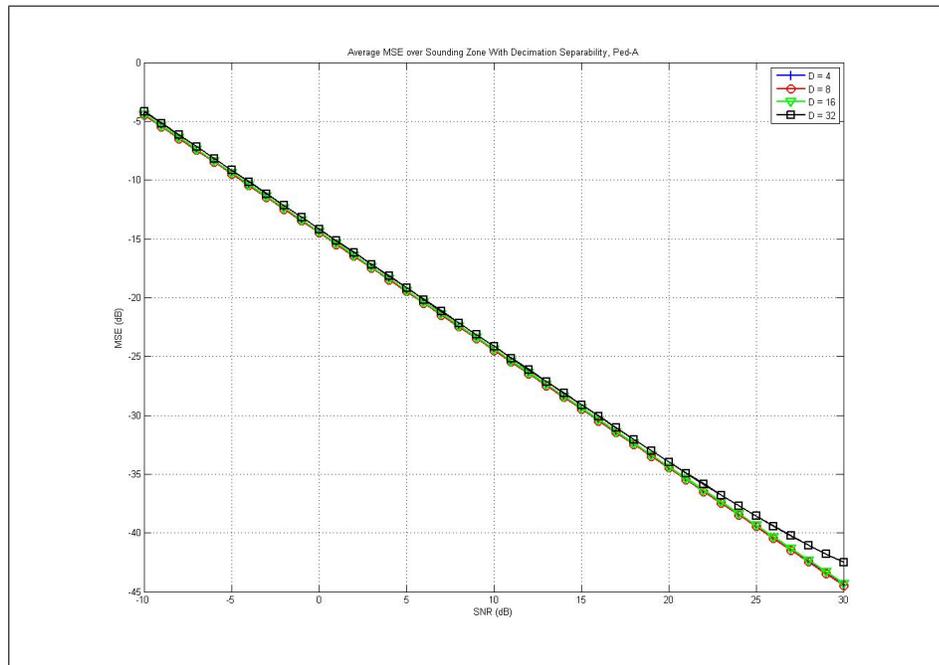


Figure 5.17: Linear Interpolation Average MSE over Sounding Zone With Decimation Separability, Ped-A

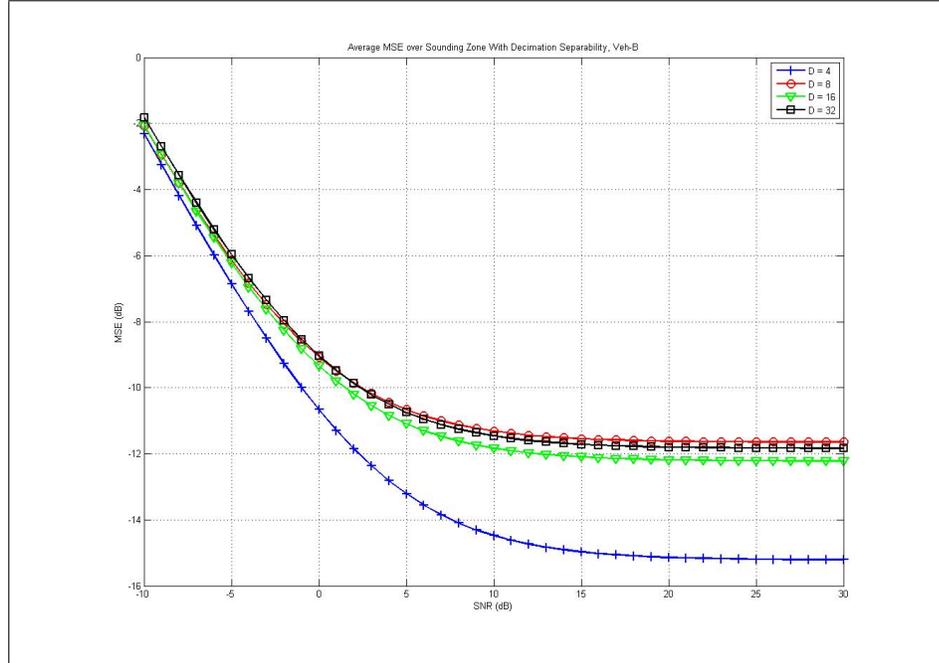


Figure 5.18: Linear Interpolation Average MSE over Sounding Zone With Decimation Separability, Veh-B

We expected to see the performance deteriorate as the decimation factor increased and the curves in the SUI-1, SUI-4, and Pedestrian-A channels illustrated that. We do see a slight improvement in performance for the larger decimation factor in the SUI-5 and Vehicular-B models. This can be attributed to the calculated frequency-domain channel correlation having interpolated or extrapolated subcarriers being more correlated with the occupied subcarriers. From Equation 5.14 and 5.15, we see that  $\alpha$  and  $\beta$  can reduce the MSE more if the correlation of the interpolated or extrapolated subcarriers with respect to the occupied subcarriers is higher. We plotted the channel correlation for each channel we tested in Figure 5.19.

For example, we see in the average MSE plots for SUI-5 that using decimation factor  $D = 32$  is slightly better than using decimation factor  $D = 16$ . When we

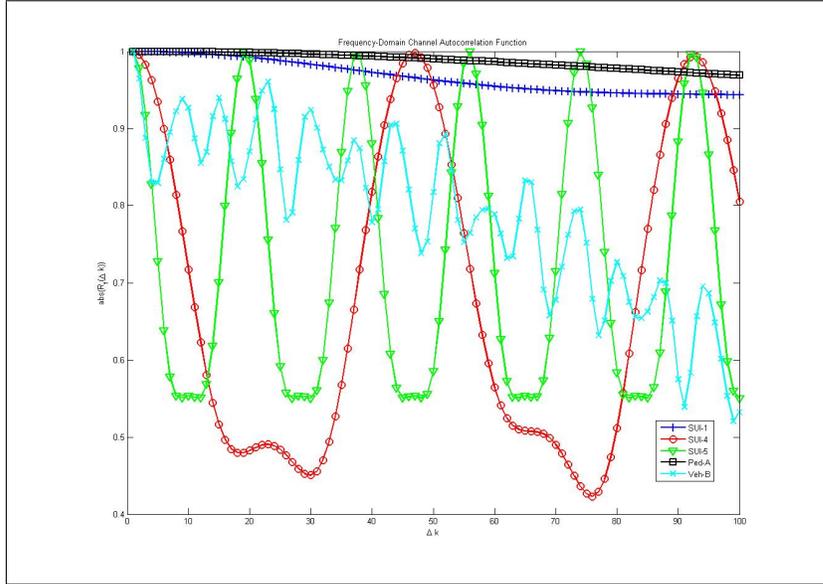


Figure 5.19: Frequency-Domain Channel Autocorrelation Function

look at the SUI-5 channel correlation in Figure 5.19, we see that after  $\Delta k = 16$ , the correlation continues to increase and peak at  $\Delta k = 20$ . Because the correlation increases in this range, when we use decimation factor  $D = 32$ , the average MSE of the interpolated and extrapolated subcarriers is slightly better for decimation factor  $D = 32$  than for  $D = 16$ . We will not read into this too much because the performance is poor in both cases. The conclusion that can be made from the average MSE plots in each channel is that linear interpolation works well for most decimation factors in channels like SUI-1 and Pedestrian-A. In SUI-4, we see a large degradation in performance for increasing decimation factors so linear interpolation should only be used in this channel if the decimation factor can be kept small. And for SUI-5 and Vehicular-B, we see that the performance is very poor except for decimation factor 4.

### 5.3.2 Frequency Smoothing

The frequency smoothing algorithm used in this study is the same algorithm that we have considered throughout this study. For specific details, refer to Section 4.2.2. However, in this particular section, we do have to take in consideration the potential overlap of the channel impulse response caused by subsampling in the frequency domain. In addition, we also have to be careful about the window size we choose because decimation factors supported by WiMAX can cause the images to move very close together. And in order to obtain an accurate estimate, the window size must be chosen such that only one image is captured. For example, for  $N = 1024$ , cyclic prefix fraction of  $1/8$ , and symbol duration of  $102.9\mu s$ , if the decimation factor is 32, then for no aliasing, the channel delay spread cannot be longer than  $3.22\mu s$ . At the same time, we have to avoid selecting a window size longer than  $3.22\mu s$  because we would capture more than one image.

### 5.3.3 Simulation: Decimation Separability

We compare the linear interpolation and frequency smoothing channel estimation performance in different channel conditions and different decimation factors. We will use the Matlab model illustrated in Figure 4.5 for simulation and mean-squared error for performance evaluation.

## Simulation Parameters

For simulation, we will assume that the sounding zone is only one OFDM symbol and that the sounding sequence is known to the transmitter and receiver. We will test the performance of the linear interpolator and frequency smoothing channel estimator for decimation factors  $D = 4, 8, 16,$  and  $32$ . For the frequency smoothing algorithm, we will choose window lengths corresponding to each decimation factor to ensure that only one image of the channel impulse response is captured. In addition, we will also test the performance of the frequency smoothing algorithm with fixed window sizes of  $5\mu\text{s}$  and  $10\mu\text{s}$ . A table summarizing the parameters are shown below.

Table 5.3: Channel Sounding Simulation Parameters: Decimation Separability

<b>System Parameters</b>	
Parameter	Value
Bandwidth Scale Factor	8 (1024 FFT)
Fundamental Bandwidth	1.25MHz
Cyclic Prefix	1/8 (11.42 $\mu$ s)
Number of Tx Antennas	1
Number of Rx Antennas	2
Sampling Rate	11.2MHz
Symbol Duration	102.9 $\mu$ s
<b>Sounding Parameters</b>	
Parameter	Value
Decimation Factors	$D = \{4, 8, 16, 32\}$
Decimation Offset	$d = 0, 1, \dots, D-1$
<b>Frequency Smoothing Parameters</b>	
Parameter	Value
Window Size	5 $\mu$ s, 10 $\mu$ s 25.73 $\mu$ s for $D = 4$ , 12.86 $\mu$ s for $D = 8$ 6.43 $\mu$ s for $D = 16$ , 3.22 $\mu$ s for $D = 32$

## Results: Mean-Squared Error

Below are MSE curves comparing the linear interpolator and frequency smoothing channel estimator.

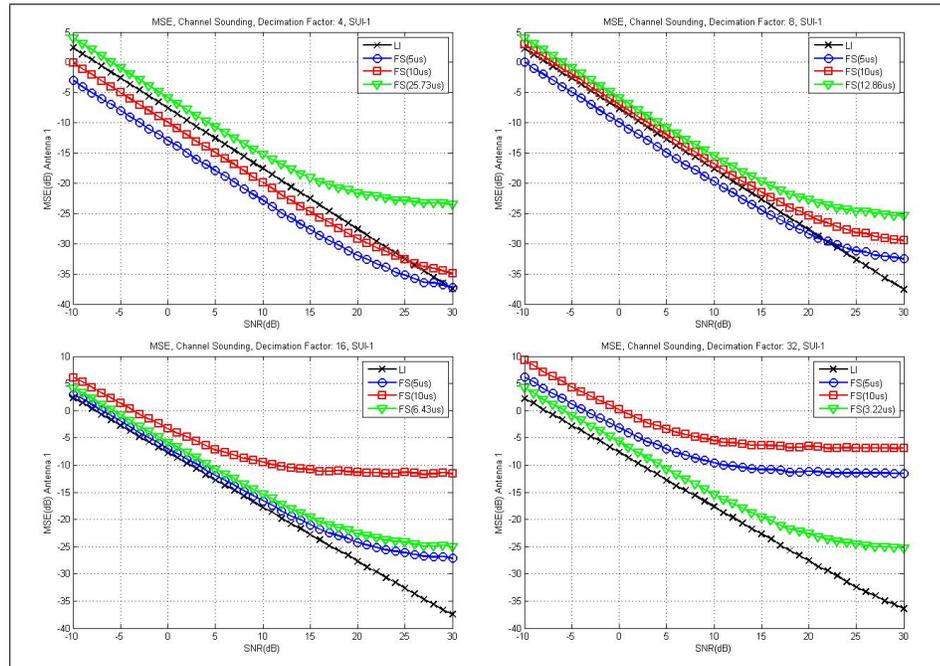


Figure 5.20: MSE, Channel Sounding with Decimation Separability, SUI-1

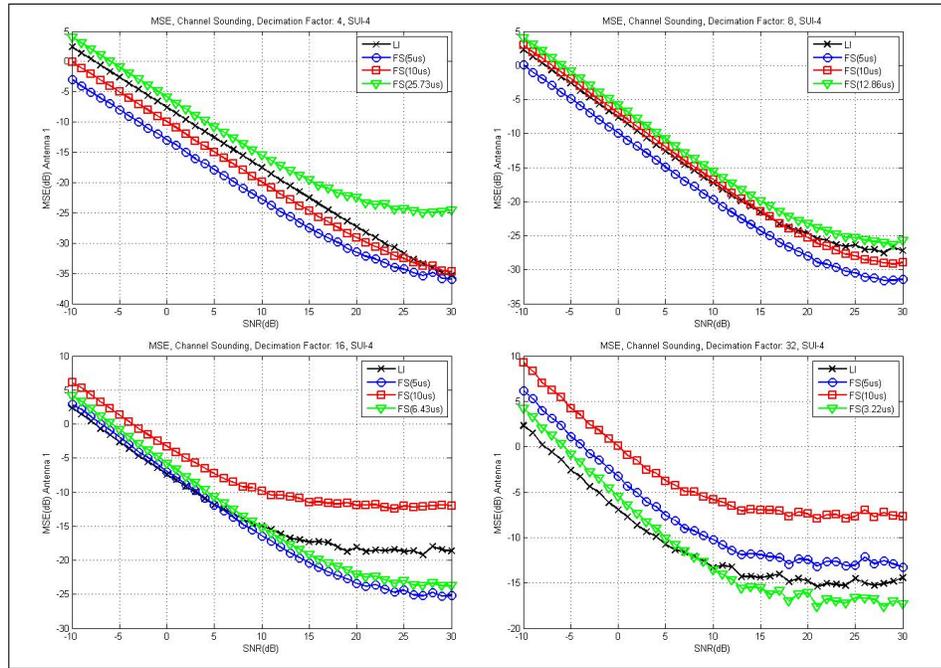


Figure 5.21: MSE, Channel Sounding with Decimation Separability, SUI-4

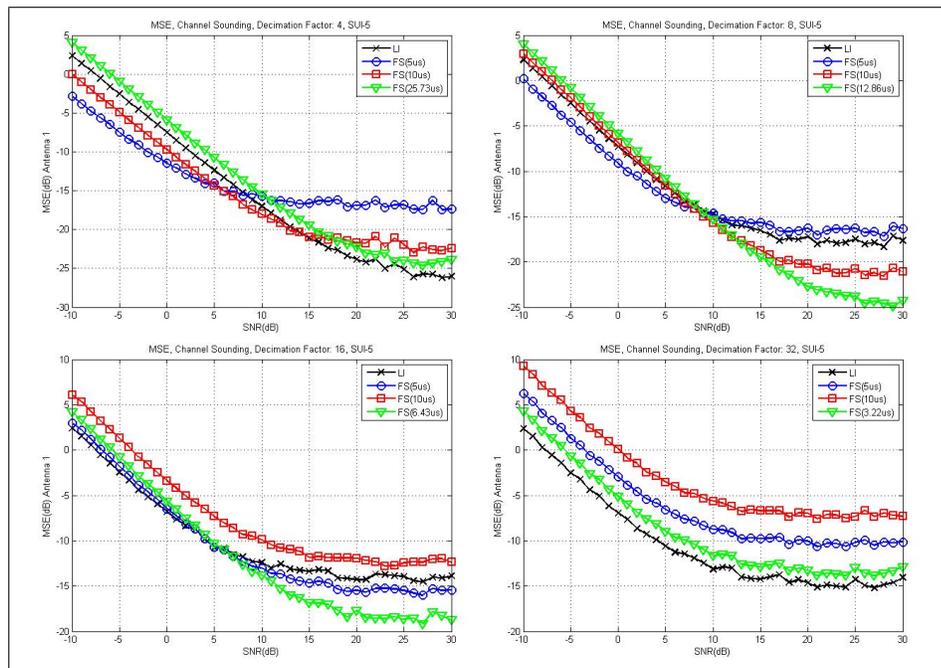


Figure 5.22: MSE, Channel Sounding with Decimation Separability, SUI-5

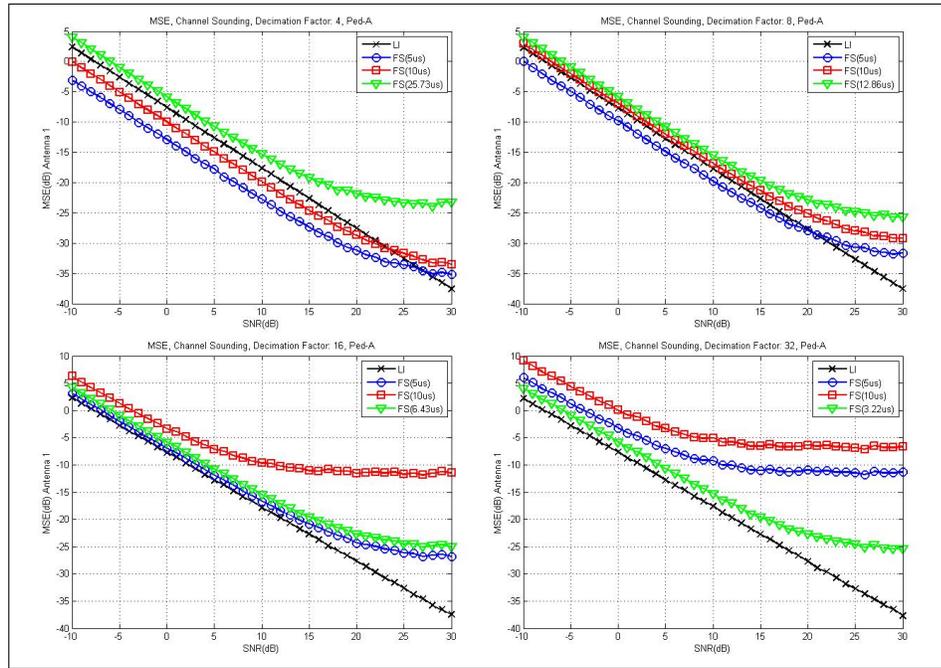


Figure 5.23: MSE, Channel Sounding with Decimation Separability, Ped-A

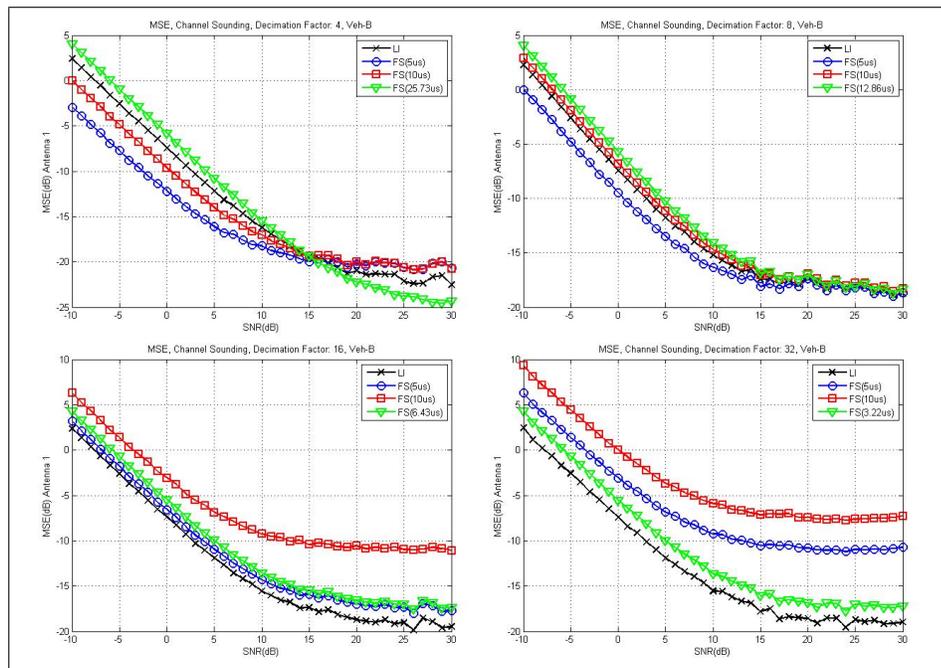


Figure 5.24: MSE, Channel Sounding with Decimation Separability, Veh-B

## Observations

First, we see that overlap of the channel impulse response can cause the frequency smoothing estimates to be biased. The relative performance of the frequency smoothing algorithm to the linear interpolator is dictated by how much bias is in the channel estimates or equivalently, how much of the channel's energy is not captured by the frequency smoothing window. For example, we see that in SUI-4 and decimation factor 32 that there is some overlap of the images because the spacing is only  $3.22\mu\text{s}$  but the delay spread is  $4\mu\text{s}$ . However, only one tap is lost and this tap has a small average power so even though this tap is not captured by the window, the bias error is still less than the interpolation error which explains why the frequency smoothing still performs better in this scenario. However, in SUI-5 and Vehicular-B, we see for decimation factor 32 that aliasing occurs and causes the frequency smoothing estimates to be very inaccurate. In addition, only 1 out of 3 taps of the SUI-5 channel profile and 2 out of the 6 taps of the Vehicular-B channel profile is captured by the frequency smoothing window with window length of  $3.22\mu\text{s}$ . This induces a large bias in the channel estimates and causes the error to be higher than the interpolation error of the linear interpolator which is already high.

We also see an interesting phenomenon in the shorter delay spread channels like SUI-1 and Pedestrian-A where no aliasing of the images occur. From these plots, we first observe that in SUI-1 and Pedestrian-A, the frequency smoothing with  $5\mu\text{s}$  and  $10\mu\text{s}$  window generally perform as well or outperform the linear interpolator for decimation factor 4 and 8. However, when we test the performance for decimation

factor 16 and 32, we see that the linear interpolator performs better than each of the frequency smoothing windows we tested with. For the  $5\mu s$  and  $10\mu s$  window, we can explain this degradation in performance because these windows capture more than one image of the channel impulse response at these decimation factors. However, it can be seen that even for the window adjusted to capture just one image is outperformed by the linear interpolator in these channels. This can be explained by the fact that there is a “sinc-like” spreading of each tap in the time-domain caused by the guard bands on each end of the sounding zone. In the frequency-domain, we only transmit sounding signals every  $D$  subcarriers for  $M$  subcarriers with guard subcarriers on each edge. When we transform this into the time-domain, we get  $D$  images of the channel impulse response convolved with a sinc function because we used an  $N$ -point IFFT and results in a “sinc-like” spreading in the time-domain. Now if we use the frequency smoothing algorithm and apply a rectangular window in the time-domain, we may capture the energy from another image depending on the length of the window and the decimation factor  $D$ . This could induce an error in our channel estimate and the magnitude of the error would depend on how close the images were to each other. Because of this result, we should choose a window length that is smaller than the designed spacing between images to avoid the effect of the “sinc-like” spreading caused by the guard bands of the sounding zone. More detail about this effect is provided in Appendix C.

## 5.4 Channel Estimation with Cyclic-Shift Separability

Another method of multiplexing users when commanded to transmit over the entire sounding zone is to use cyclic-shift separability. If we define all the subcarriers in the sounding zone to be in  $\mathcal{S}$  and  $M$  to be the size of the sounding zone, then in channel sounding with cyclic-shift separability, each user will transmit the same sounding sequence at all subcarriers in  $\mathcal{S}$ , but with a different phase shift  $e^{-j\frac{2\pi kI}{P}}$  in the frequency-domain (or cyclic-shift in the time-domain) where  $k$  is the subcarrier index,  $P$  is the maximum cyclic shift index and  $I$  is the cyclic time shift index between 0 and  $P-1$  assigned to each user in order to maintain signal orthogonality. According to [23], maximum cyclic shift indices of 4, 8, 16, and 32 are to be supported. In this section, we will compare a staircasing channel estimation technique to the frequency smoothing channel estimation algorithm and evaluate their performance for different maximum cyclic shift indices and channels.



Figure 5.25: Cyclic Shift Separability

### 5.4.1 Staircasing Channel Estimation

The first step in the staircasing method is to generate baseline channel estimates using the least squares estimate at each subcarrier. However, in this scenario, we will also need to separate each user's transmitted signal from the received signal because users are allowed to transmit in the same time and frequency resource. To accomplish this, we must group the channel estimates into P subcarriers and correlate over the P channel estimates in each group by undoing a user's unique cyclic-shift at each subcarrier and summing over all the subcarriers within the group to remove all other users. We summarize the steps of this algorithm mathematically below.

1. Generate Baseline Channel Estimates using Least Squares Estimate at each subcarrier

$$\hat{H}(t, k) = \frac{Y(t, k)}{X(t, k)} \text{ for } (t, k) \in \mathcal{S} \quad (5.17)$$

2. Correlate baseline channel estimate over P subcarriers to isolate user of interest. If we divide the sounding zone into groups of P subcarriers and let  $i$  be the group index where  $0 \leq i \leq \frac{N_{\text{used}}}{P}$  and  $k_i$  be the leftmost subcarrier in within each group, then the channel estimate for the  $i^{\text{th}}$  group and  $m^{\text{th}}$  subcarrier within each group is

$$\hat{H}_{sc}(t, k_i + m) = \frac{1}{P} \sum_{l=0}^{P-1} \hat{H}(t, k_i + l) e^{j \frac{2\pi(k_i+l)I}{P}} \text{ for } m = 0, 1, \dots, P-1 \quad (5.18)$$

The result of this technique is a channel estimate that resembles a staircase because we generate the same channel estimate for each group of P subcarriers. To analyze

the performance of the staircasing method, we derive the mean-squared error at each subcarrier and computed the arithmetic average over each group. To simplify the derivation, we will assume that the channel is WSS, noise is mutually uncorrelated and uncorrelated with the channel, and the magnitude of the transmitted sounding signal at each occupied subcarrier is the same  $|X(t, k)| = A$  for  $(t, k) \in \mathcal{S}$

$$\begin{aligned}
MSE_{sc} &= \frac{1}{P} \sum_{m=0}^{P-1} E[|\hat{H}_{sc}(t, k_i + m) - H(t, k_i + m)|^2] \\
&= \frac{1}{P} \sum_{m=0}^{P-1} E[|\frac{1}{P} \sum_{l=0}^{P-1} \hat{H}(t, k_i + l) e^{j\frac{2\pi(k_i+l)l}{P}} - H(t, k_i + m)|^2] \\
&= \frac{1}{P} \sum_{m=0}^{P-1} \left\{ \sigma_H^2 \left( \frac{1}{P^2} \sum_{r=0}^{P-1} \sum_{l=0}^{P-1} R_f[l-r] e^{j\frac{2\pi(l-r)l}{P}} + R_f[0] \right. \right. \\
&\quad \left. \left. - \frac{1}{P} \sum_{l=0}^{P-1} R_f[l-m] e^{j\frac{2\pi(k_i+l)l}{P}} - \frac{1}{P} \sum_{l=0}^{P-1} R_f[m-l] e^{-j\frac{2\pi(k_i+l)l}{P}} \right) \right. \\
&\quad \left. + \frac{1}{P} \frac{\sigma_v^2}{A^2} \right\} \tag{5.19}
\end{aligned}$$

From the derivation, we see similar to the linear interpolator that the components of channel estimation error is due to interpolation error and noise. The interpolation error comes from the fact that we essentially calculate an average of the least-squares channel estimates over P subcarriers and use this to estimate the channel at each subcarrier within the group. The magnitude of the interpolation error is dependent on the frequency-domain channel correlation and the maximum cyclic shift index P. We also see on average that the contribution of noise is reduced by approximately a factor of 1/P. Using the derivation in Equation 5.19, we also plotted the average MSE for each maximum cyclic-shift index and compared them in the SUI-1, SUI-4, SUI-5, Pedestrian-A, and Vehicular-B channels.

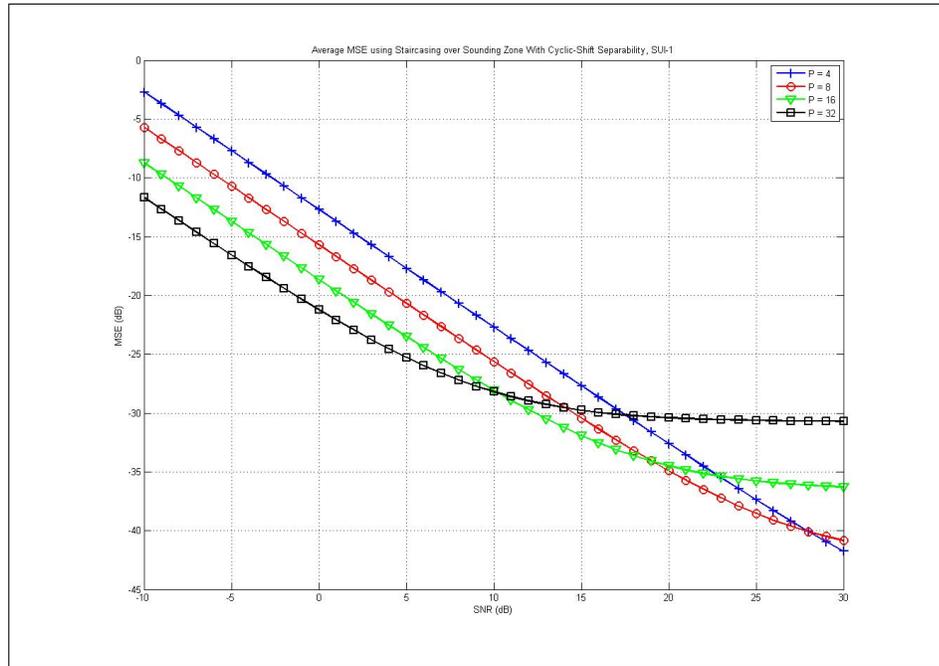


Figure 5.26: Average MSE over Sounding Zone With Cyclic-Shift Separability, SUI-1

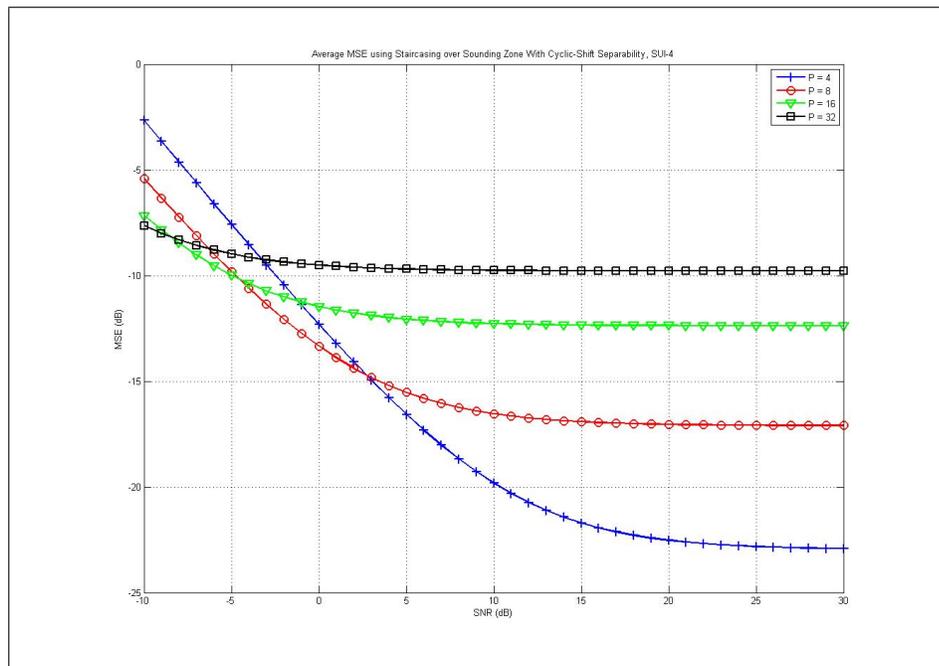


Figure 5.27: Average MSE over Sounding Zone With Cyclic-Shift Separability, SUI-4

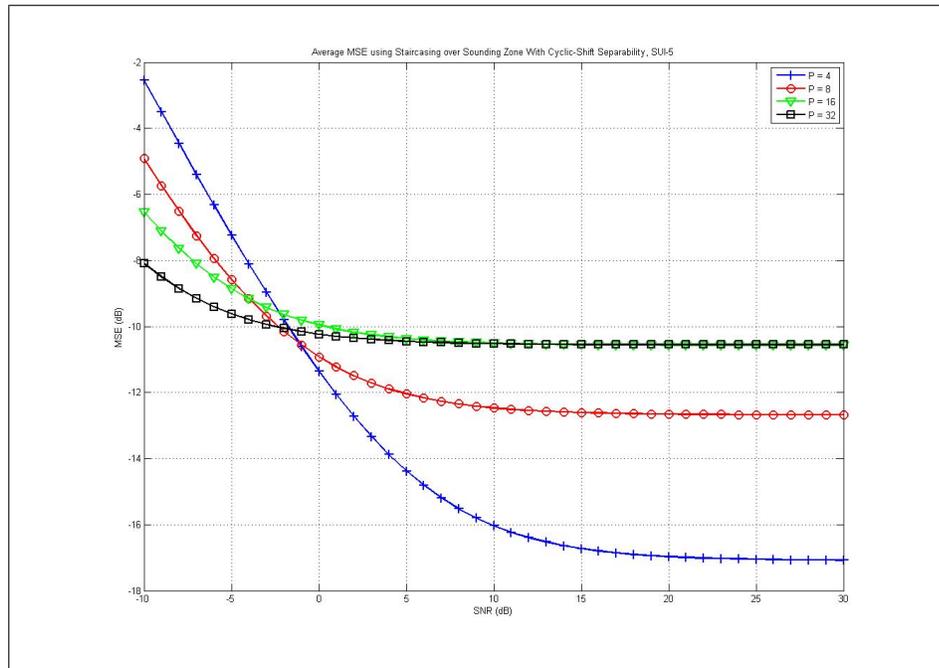


Figure 5.28: Average MSE over Sounding Zone With Cyclic-Shift Separability, SUI-5

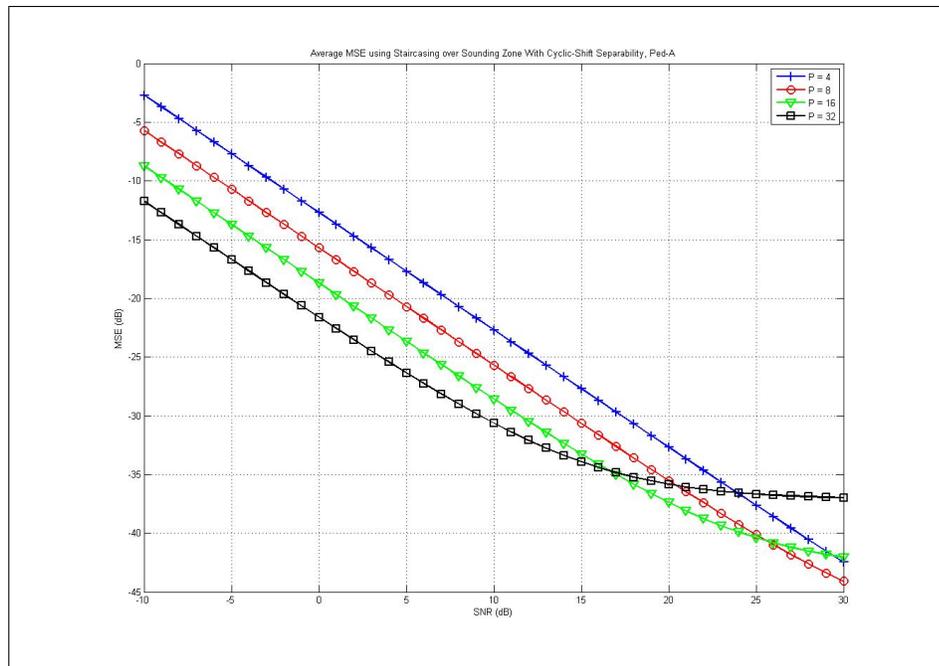


Figure 5.29: Average MSE over Sounding Zone With Cyclic-Shift Separability, Ped-

A

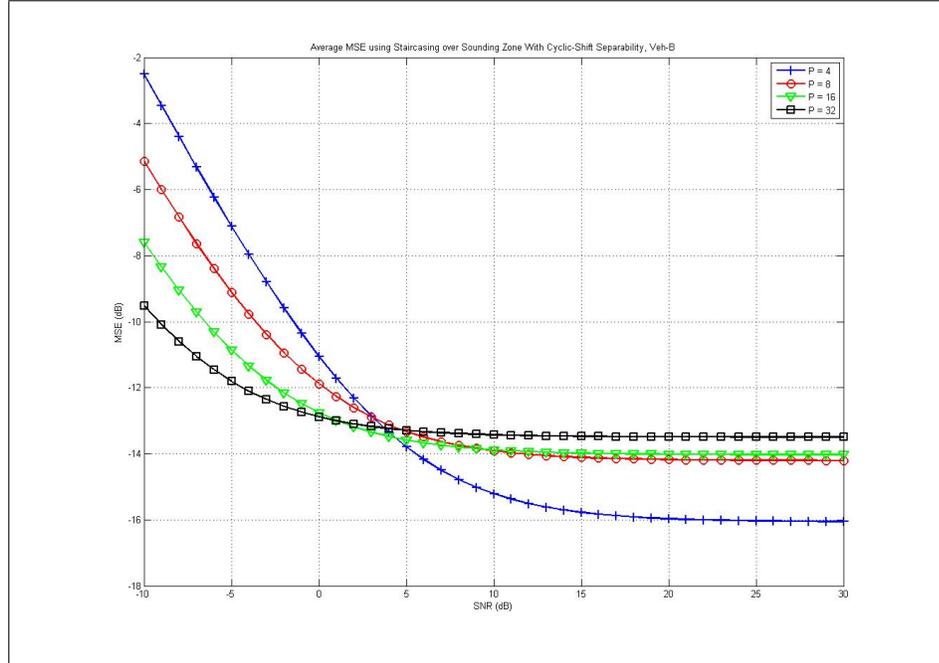


Figure 5.30: Average MSE over Sounding Zone With Cyclic-Shift Separability, Veh-B

From the plots, we observe that the performance of the staircasing estimator generally floors at a higher MSE for larger values of  $P$  because the resolution of the estimator is lower. We also observe that for SUI-1 and Pedestrian-A that performance is better for larger values of  $P$  because these channels are highly correlated so the interpolation error is very small no matter what value of  $P$  we use. In these channels the error depends only on noise and from Equation 5.19, we see that the noise error is lower for larger values of  $P$ . This also explains why at low SNR, using larger values of  $P$  for the staircasing method performs better.

## 5.4.2 Frequency Smoothing

The frequency smoothing algorithm can also be applied in channel sounding with cyclic-shift separability. Refer to Section 4.2.2 for more detail. With cyclic-shift separability, the frequency smoothing window must also be shifted cyclically in time by the same cyclic shift applied to the sounding sequence in order to eliminate the other users. As with decimation separability, the possibility of overlapping channel impulse responses is possible if users are not cyclically shifted far enough from each other. Assuming again that the number of subcarriers  $N = 1024$ , the cyclic prefix fraction is  $1/8$ , and the symbol duration is  $102.9\mu\text{s}$ , then if the system uses a maximum cyclic shift index of 32 and user 1 and user 2 have a cyclic time-shift index  $I$  of 0 and 1 respectively, they will only be separated by  $3.22\mu\text{s}$  in the time-domain. In this situation, if we use a frequency smoothing window longer than  $3.22\mu\text{s}$  to user 1, the window will capture part of user 2's channel response and result in a less accurate measurement. Similar to our study with decimation separability, we will evaluate the performance of frequency smoothing windows of length  $5\mu\text{s}$ ,  $10\mu\text{s}$ , and one that is adjusted according to the maximum cyclic shift index to ensure only one user's channel captured.

## 5.4.3 Simulation: Cyclic-Shift Separability

### Simulation Setup

In this study, we will test the performance of the staircasing and frequency smoothing algorithm for different maximum cyclic shift factors in different chan-

nel conditions. We will again use the Matlab model illustrated in Figure 4.5 for simulation and mean-squared error for performance evaluation. However, for this particular study, we assume only two users in the system and each are assigned a cyclic-shift time index of 0 and 1. The system and algorithm parameters are shown below.

Table 5.4: Channel Sounding Simulation Parameters: Cyclic Shift Separability

<b>System Parameters</b>	
Parameter	Value
Bandwidth Scale Factor	8 (1024 FFT)
Fundamental Bandwidth	1.25MHz
Cyclic Prefix	$1/8$ ( $11.42\mu s$ )
Number of Tx Antennas	1
Number of Rx Antennas	2
Sampling Rate	11.2MHz
Symbol Duration	$102.9\mu s$
<b>Sounding Parameters</b>	
Parameter	Value
Maximum Cyclic Shift Index	$D = \{4, 8, 16, 32\}$
Cyclic Shift Time Index	$I = 0, 1, \dots, P-1$
<b>Frequency Smoothing Parameters</b>	
Parameter	Value
Window Size	$5\mu s, 10\mu s$ $25.73\mu s$ for $P = 4$ , $12.86\mu s$ for $P = 8$ $6.43\mu s$ for $P = 16$ , $3.22\mu s$ for $P = 32$

## Results: Mean-Squared Error

Below are mean-squared error plots comparing the staircasing estimator to the frequency smoothing estimator.

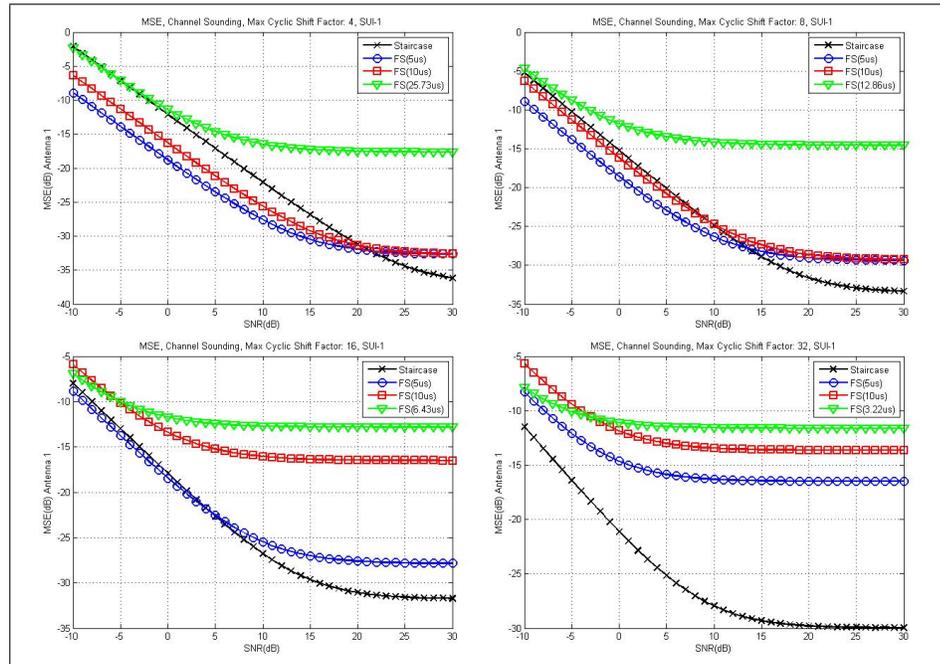


Figure 5.31: MSE, Channel Sounding with Cyclic Shift Separability, SUI-1

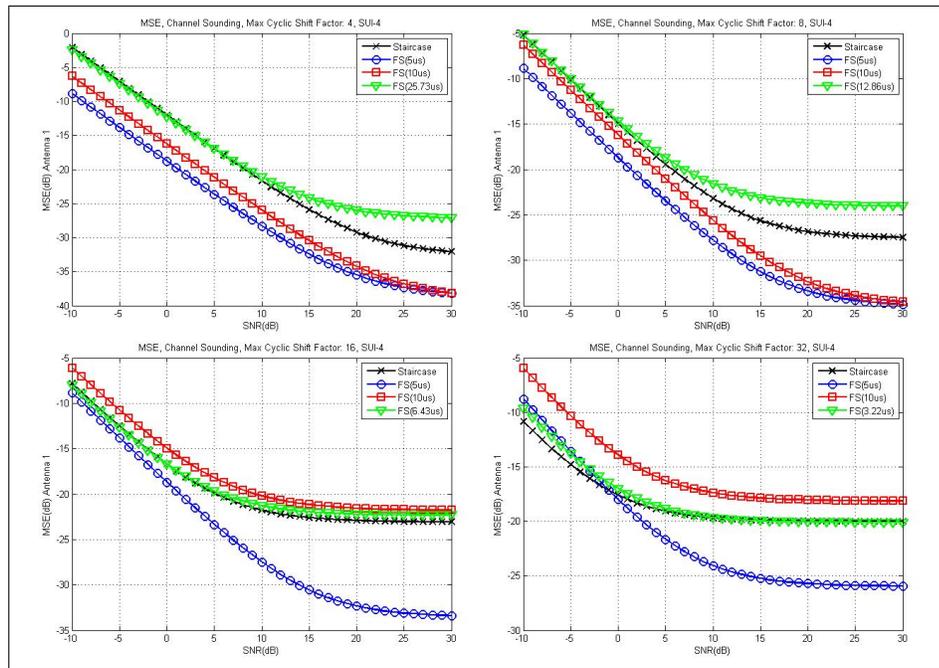


Figure 5.32: MSE, Channel Sounding with Cyclic Shift Separability, SUI-4

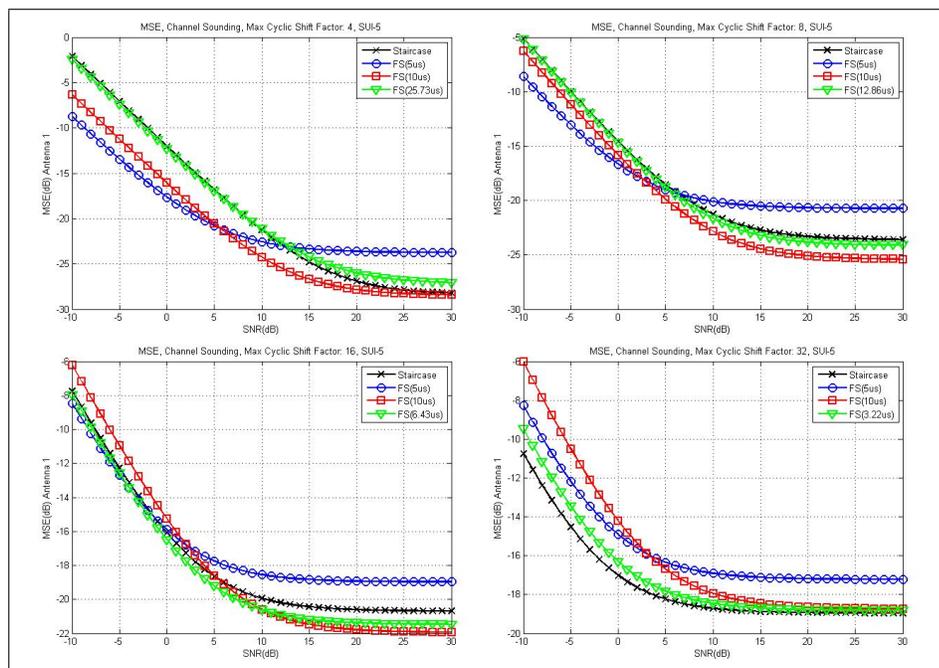


Figure 5.33: MSE, Channel Sounding with Cyclic Shift Separability, SUI-5

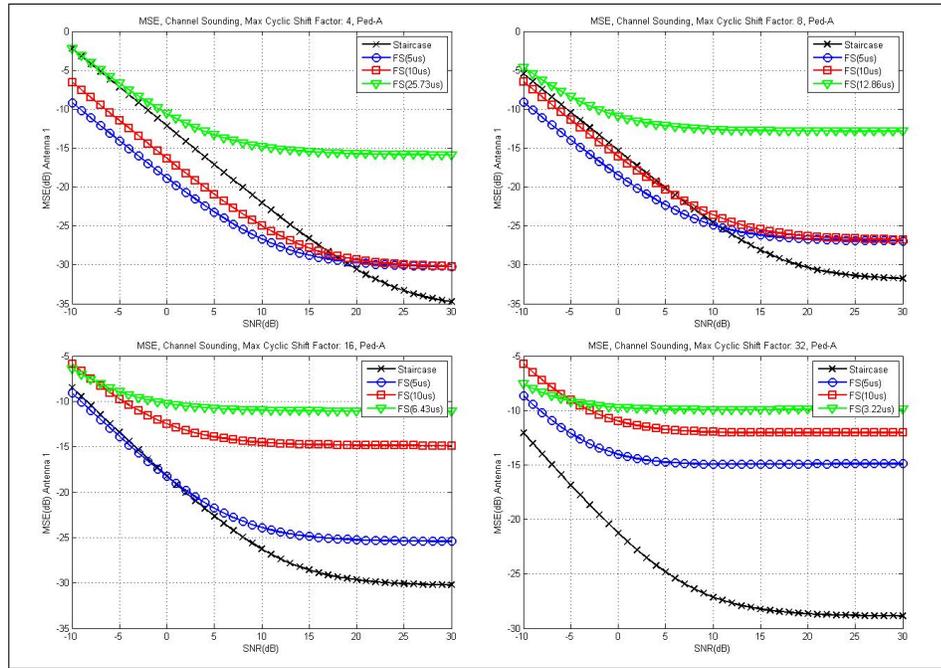


Figure 5.34: MSE, Channel Sounding with Cyclic Shift Separability, Ped-A

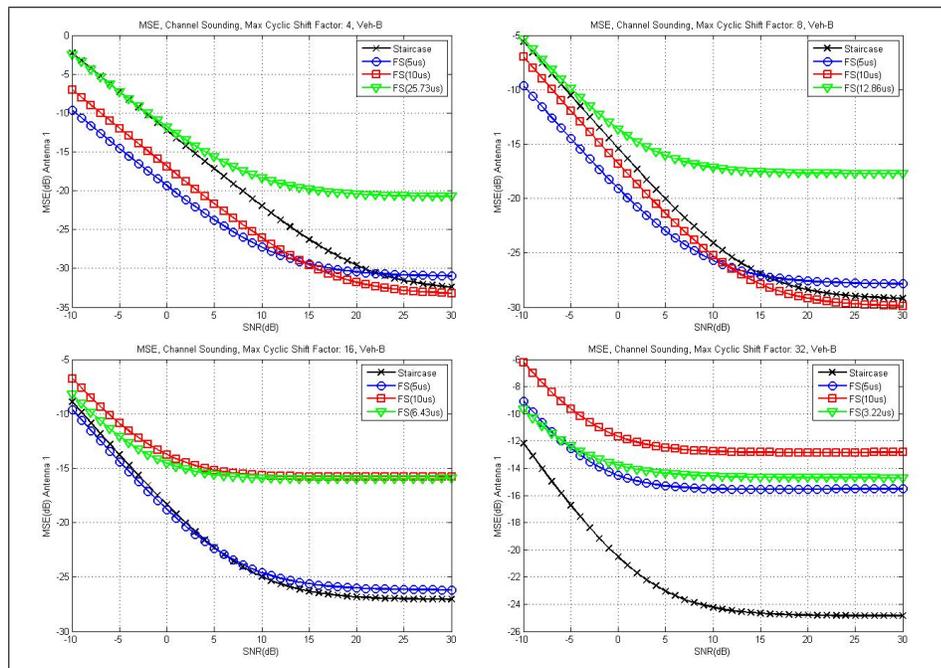


Figure 5.35: MSE, Channel Sounding with Cyclic Shift Separability, Veh-B

## Observations: Mean-Squared Error

At low SNR, we see that the frequency smoothing algorithm generally performs better than the staircasing estimator except for maximum cyclic shift factor 32. We can attribute this to the fact that the frequency smoothing algorithm rejects more noise than the staircasing estimator except when the staircasing estimator averages the estimates over a large number of subcarriers and averages out a significant amount of noise. It can be seen from Equation 5.19 that for  $P = 32$  that the noise is reduced a factor of  $1/32$  and this factor is often smaller than  $L/M$  which is the factor that the frequency smoothing algorithm reduces noise by.

For the frequency smoothing algorithm, we see that the performance relative to the staircasing estimator is worse in long delay spread channels and large maximum cyclic shift indices. This degradation in performance is due to the fact that when the delay spread of the channel exceeds the spacing between cyclically-shifted users, the received signal of each user overlap in the time-domain so the frequency smoothing window cannot capture all of the channel's energy without capturing another user's channel. Similarly, if we choose a frequency smoothing window length such that it exceeds the spacing between cyclically-shifted users, then the window will capture part of another user's channel and result in an inaccurate channel estimate.

We also observe that in low delay spread channels like SUI-1 and Pedestrian-A that frequency smoothing with a  $5\mu\text{s}$  and  $10\mu\text{s}$  window generally performs better than the staircasing estimator for maximum cyclic shift indices of 4 and 8. However, as we increase the maximum cyclic shift index to 16 and 32, we see the staircasing

estimator generally outperform the frequency smoothing window even though the channel response does not overlap with each other in the time-domain. We can attribute this to the “sinc-like” spreading of each tap in the time-domain caused by the guard bands on each end of the sounding zone. In the frequency-domain, we only transmit sounding signals in  $M$  out of  $N$  subcarriers with guard subcarriers on each edge. When we transform this into the time-domain, we get a sinc function because we use an  $N$ -point IFFT. This sinc function is then convolved with the channel impulse response and the result is a “sinc-like” spreading in the time-domain. The effect is that the spacing between cyclically shifted users is reduced. Now if we use the frequency smoothing algorithm and apply a rectangular window in the time-domain, we may capture the energy from another user’s channel depending on the length of the window and the choice of the maximum cyclic shift index  $P$ . This phenomenon also explains why the frequency smoothing algorithm with window length adjusted to capture exactly one user’s channel performs poorly for almost all channels. More detail about this effect is described in Appendix C.

## 5.5 Decimation Separability vs. Cyclic-Shift Separability

One question often asked is if channel sounding with decimation separability or cyclic-shift separability is preferred. According to [24], using cyclic-shift separability results in better overall link-level and system-level performance. One of the

advantages to using cyclic-shift separability is the higher processing gain. Because users transmit over every subcarrier within the sounding zone, it spreads the signal's energy over a wider range of frequencies and makes it less susceptible to interference. Another advantage to using cyclic-shift separability is its flexibility in multiplexing users in low and high delay spread channels. With cyclic-shift separability, the system has the flexibility to assign users a cyclic shift index according to their channel's delay spread such that there is adequate spacing between users. For example, if user 1 had a channel with  $10\mu\text{s}$  delay spread and user 2 had a channel with  $5\mu\text{s}$  delay spread, we could cyclically shift user 2 by an amount that is at least  $10\mu\text{s}$  to maintain orthogonality. However, in decimation separability, we do not have this flexibility because in most situations, we cannot reduce the decimation factor for high delay spread users or increase the decimation factor for low delay spread users because the occupied subcarriers may overlap. Therefore, in general, we would prefer to use cyclic-shift separability when performing channel sounding over the entire sounding zone.

## 5.6 Conclusion

In a system with channel sounding enabled, a user may be commanded by the base station to transmit sounding signals in a fixed number of sounding bands each containing 18 adjacent subcarriers or over the entire zone. If a user transmits over a fixed number of sounding bands, we can apply the frequency smoothing algorithm or use the LMMSE estimator to perform channel estimation. From the results, we

see that at low SNR, the frequency smoothing estimator generally performs better because it rejects more noise. However, as we increase the SNR, we see that the LMMSE estimator can outperform the frequency smoothing algorithm if our LMMSE estimator utilizes low-rank modeling and approximates the channel delay spread accurately. In addition, we tend to see significant improvement at the edge subcarriers of each sounding band from using the LMMSE estimator because using the frequency smoothing algorithm suffers from the edge effect. If a base station commands the users to sound over the entire sounding zone, it may do so through decimation separability or cyclic-shift separability. In decimation separability, a user occupies every  $D^{th}$  subcarrier so we can apply a linear interpolation algorithm or frequency smoothing algorithm to perform channel estimation. Our results showed that for small decimation factors, the frequency smoothing algorithm generally performed better than the linear interpolator. But for larger delay spread channels and higher decimation factors, we observed that the relative performance depended on whether the frequency smoothing window captured other images or did not capture all of the channel's energy. In cyclic-shift separability, each user occupies the entire sounding zone but with a different cyclic time-shift. For this multiplexing method, we evaluated the staircasing estimator and the frequency smoothing algorithm. We observed that again the frequency smoothing algorithm generally performed better in low SNR channels. But its performance relative to the staircasing estimator was mixed for higher maximum cyclic shift factors and higher delay spread channels. We observed that the staircasing method generally performed better when the spacing in time between each user's transmitted signal was too close for the frequency

smoothing window to capture just one user's signal energy or caused the channel impulse response of each user to overlap in time.

# Chapter 6

## Conclusion

In our study, we explored many channel estimation algorithms that can be used in WiMAX. Channel estimation is an important feature in any wireless communication system because it aids the receiver in undoing any distortion in the transmitted signal caused by the wireless channel. If the channel estimator has good tracking capabilities, it can significantly improve the coverage, throughput, and reliability of the overall system.

WiMAX utilizes pilots to aid in channel estimation and arranges the pilots differently depending on the permutation it employs. In PUSC, we analyzed the performance of the linear interpolator and 4-pilot averaging and observed that the 4-pilot averaging performed better in low SNR channels. But as the SNR increased and the channel delay spread and Doppler spread increased, we observed that the linear interpolator performed better. In AMC, we introduced a frequency smoothing algorithm and compared its performance to the linear interpolator. We observed that the frequency smoothing algorithm generally outperformed the linear interpo-

lator except when the window length was not chosen long enough to capture all of the channel's energy.

WiMAX also utilizes a feature called uplink channel sounding in which the base station allocates a sounding zone for users to transmit known sequences to allow it to obtain each user's unique channel conditions. We analyzed the performance of the frequency smoothing and linear minimum mean-squared error (LMMSE) channel estimator for when the base station commanded each user to transmit in a fixed number of sounding bands. We found that the frequency smoothing algorithm performed well in low SNR channels but was generally outperformed by the LMMSE estimator when we made a good estimate of the channel delay spread. We also analyzed channel estimation performance in channel sounding with decimation separability and cyclic-shift separability. In decimation separability, we compared the frequency smoothing algorithm to the linear interpolator and observed that frequency smoothing performed well when the images of the channel were not too close together and/or when the channel delay spread did not exceed the spacing between adjacent images. In cyclic-shift separability, we evaluated a staircasing estimator and compared it to the frequency smoothing algorithm. We found again that the frequency smoothing estimator performed better when each user were cyclically shifted in time far enough from each other such that the channel responses do not overlap with each other in time.

## 6.1 Future Work

While this thesis covers much in the area of channel estimation for WiMAX, there are still some areas in which this study can be expanded upon.

1. *Study of the time evolution of the channel* - In this study, our focus was primarily on channel estimation and interpolation in frequency. This study could be expanded to include an analysis about the evolution of the channel in time and algorithms that can be applied across the time.
2. *Development of a link-level simulator supporting AMC and channel sounding* - In our study for AMC and channel sounding, we used MSE as our primary metric to evaluate each algorithm. However, a better metric to evaluate each algorithm is bit-error rate, frame-error rate, and throughput and to accomplish this, we need to integrate our algorithm with a link-level simulator with all the components in a WiMAX system.
3. *Different windowing methods for the frequency smoothing algorithm* - For frequency smoothing, we tested the algorithm using a rectangular window and observed that one consequence of this is the edge effect. One alternative to combat the edge effect is to use a different window (e.g. Raised-cosine Window).
4. *Channel estimation in the MIMO case* - This study only considers the SIMO case and could be extended to study channel estimation algorithms that can be used in the MIMO case.

# Appendix A

## SUI and ITUR Channel Models

Table A.1: ITUR: Pedestrian Channel Model Parameters

Pedestrian-A						
	Tap 1	Tap 2	Tap 3	Tap 4	Tap 5	Tap 6
Delay( $\mu s$ )	0	0.11	0.19	0.41		
Power(dB)	0	-9.7	-19.2	-22.8		
K Factor	0	0	0	0		
Doppler(Hz)	3km/hr(10Hz), 30km/hr(100Hz), 120km/hr(400Hz)					
Pedestrian-B						
	Tap 1	Tap 2	Tap 3	Tap 4	Tap 5	Tap 6
Delay( $\mu s$ )	0	0.20	0.80	1.2	2.3	3.7
Power(dB)	0	-0.9	-4.9	-8.0	-7.8	-23.9
K Factor	0	0	0	0	0	0
Doppler(Hz)	3km/hr(10Hz)					

Table A.2: ITUR: Vehicular Channel Model Parameters

Vehicular-A						
	Tap 1	Tap 2	Tap 3	Tap 4	Tap 5	Tap 6
Delay( $\mu$ s)	0	0.31	0.71	1.09	1.73	2.51
Power(dB)	0	-1.0	-9.0	-10.0	-15.0	-20.0
K Factor	0	0	0	0	0	0
Doppler(Hz)	30km/hr(100Hz), 120km/hr(400Hz), 250km/hr(833Hz)					
Vehicular-B						
	Tap 1	Tap 2	Tap 3	Tap 4	Tap 5	Tap 6
Delay( $\mu$ s)	0	0.30	8.9	12.9	17.1	20.0
Power(dB)	-2.5	0.0	-12.8	-10.0	-25.2	-16.0
K Factor	0	0	0	0	0	0
Doppler(Hz)	30km/hr(100Hz), 120km/hr(400Hz), 250km/hr(833Hz)					

Table A.3: SUI Channel Model Parameters

SUI-1				SUI-4			
	Tap 1	Tap 2	Tap 3		Tap 1	Tap 2	Tap 3
Delay( $\mu$ s)	0	0.4	0.8	Delay( $\mu$ s)	0	2.0	4.0
Power(dB)	0	-15	-20	Power(dB)	0	-4	-8
K Factor	18	0	0	K Factor	0	0	0
Doppler(Hz)	0.4	0.4	0.4	Doppler(Hz)	1	1	1
SUI-2				SUI-5			
	Tap 1	Tap 2	Tap 3		Tap 1	Tap 2	Tap 3
Delay( $\mu$ s)	0	0.5	1.0	Delay( $\mu$ s)	0	5.0	10.0
Power(dB)	0	-12	-15	Power(dB)	0	-5	-10
K Factor	10	0	0	K Factor	0	0	0
Doppler(Hz)	0.4	0.4	0.4	Doppler(Hz)	2	2	2
SUI-3				SUI-6			
	Tap 1	Tap 2	Tap 3		Tap 1	Tap 2	Tap 3
Delay( $\mu$ s)	0	0.5	1.0	Delay( $\mu$ s)	0	14.0	20.0
Power(dB)	0	-5	-10	Power(dB)	0	-10	-14
K Factor	1	0	0	K Factor	0	0	0
Doppler(Hz)	1	1	1	Doppler(Hz)	0.4	0.4	0.4

# Appendix B

## Impact of Frequency Smoothing Window on Channel Estimates

In this appendix, we calculate the window lengths,  $L$ , used in this simulation. The purpose of this section is to illustrate how much of the channel impulse response after an  $M$ -point IFFT a  $5\mu s$  and  $10\mu s$  window captures.

$$L = \text{round} \left( M \cdot \frac{\text{Window Size in Time}}{\text{Symbol Duration}} \right) \quad (\text{B.1})$$

For a  $1/T_n = 11.2\text{MHz}$ ,  $N = 1024$ , and cyclic prefix fraction of  $1/8$ , the symbol duration is  $102.9\mu s$ . Using Equation B.1, we can calculate the window length for different size subchannel/sounding zone sizes. Table B.1 shows different the truncation points used in the frequency smoothing channel estimation algorithm. For example, for sounding zone size  $M = 54$ , a  $5\mu s$  window captures the first 3 taps of the impulse response and a  $10\mu s$  window captures the first 5 taps of the impulse response.

When analyzing the performance of the frequency smoothing algorithm, we look at how many taps of the channel impulse response the window captures. If the difference between the  $5\mu\text{s}$  and  $10\mu\text{s}$  window is large, then we expect the performance to be significantly better using the  $10\mu\text{s}$  window at high SNR. From these plots, we see that the most noticeable difference between the  $5\mu\text{s}$  and  $10\mu\text{s}$  window occur in SUI-4 and SUI-5. SUI-1 and Pedestrian-A both have short delay spreads so both capture most of the channel's energy. Vehicular-B has a very long delay spread so both windows fail to capture a lot of the channel's energy.

Table B.1: Window Length Calculations

Adjacent Subcarriers (M)	Length for $5\mu\text{s}$ Window	Length for $10\mu\text{s}$ Window
18	1	2
36	2	3
54	3	5
72	3	7

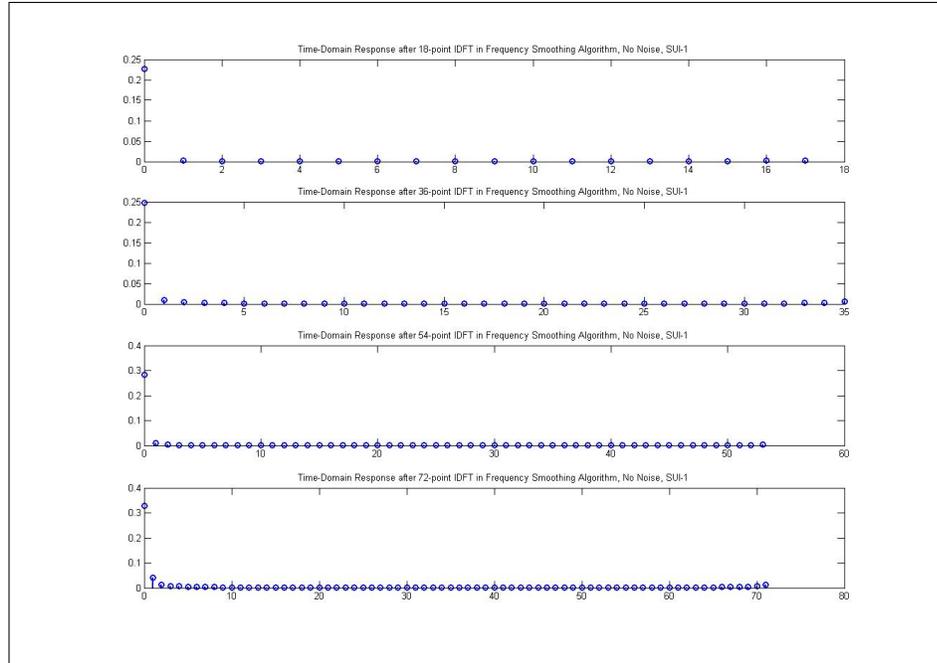


Figure B.1: Time-Domain Response after M-point IDFT in Frequency Smoothing Algorithm, No Noise, SUI-1

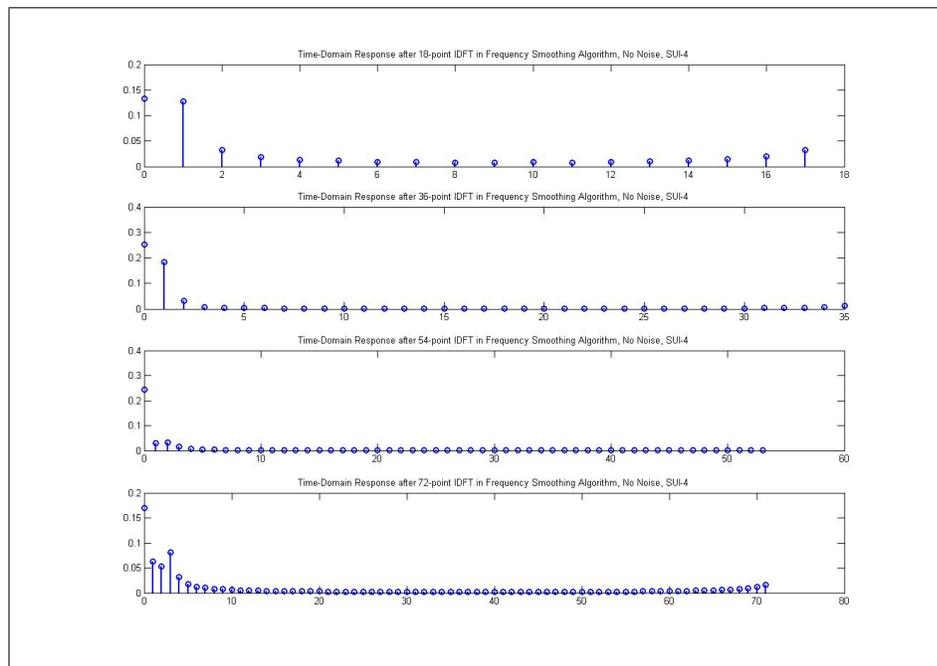


Figure B.2: Time-Domain Response after M-point IDFT in Frequency Smoothing Algorithm, No Noise, SUI-4

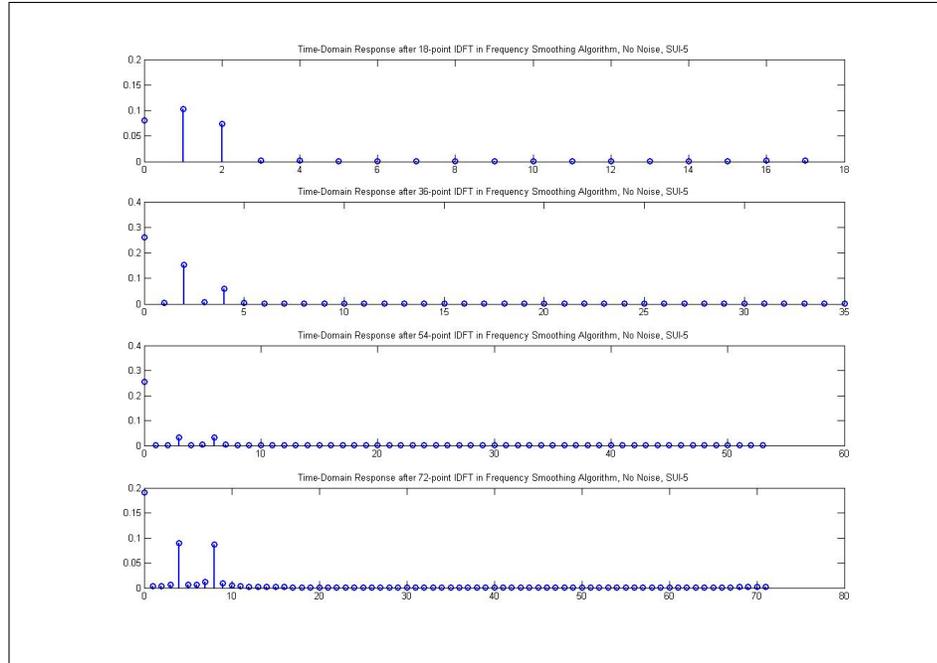


Figure B.3: Time-Domain Response after M-point IDFT in Frequency Smoothing Algorithm, No Noise, SUI-5

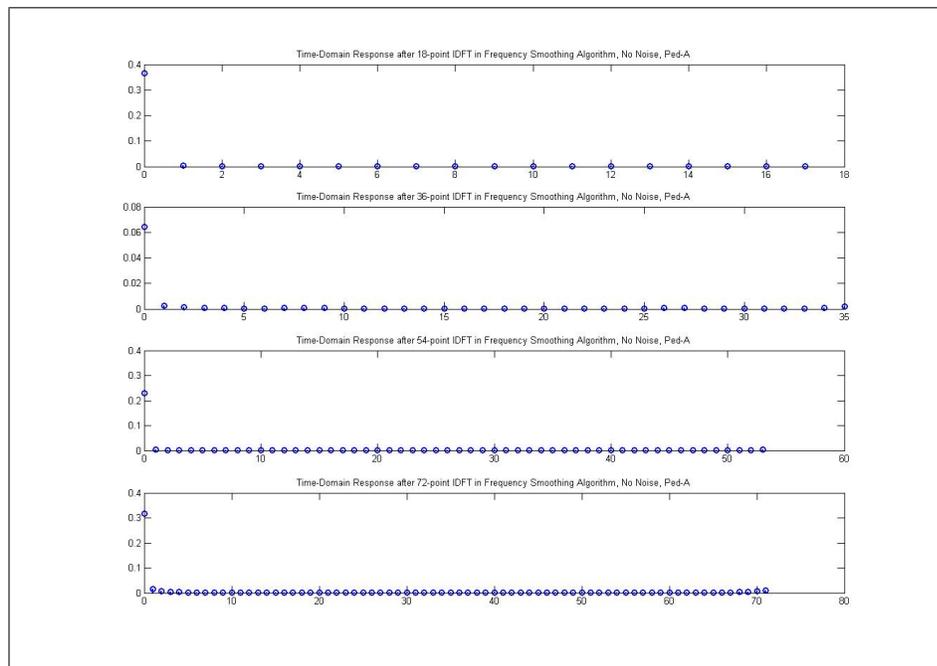


Figure B.4: Time-Domain Response after M-point IDFT in Frequency Smoothing Algorithm, No Noise, Ped-A

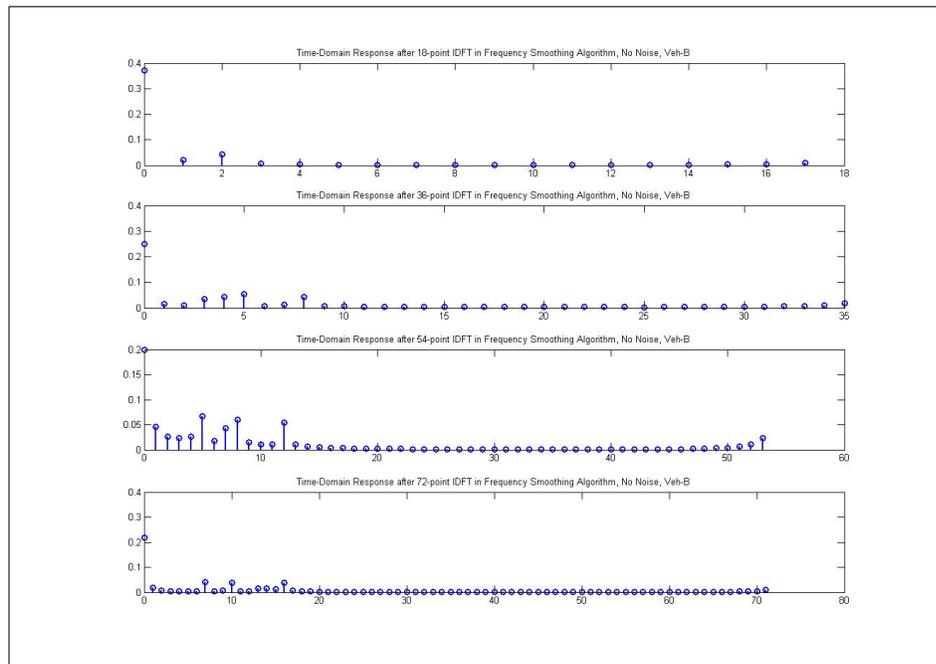


Figure B.5: Time-Domain Response after M-point IDFT in Frequency Smoothing Algorithm, No Noise, Veh-B

# Appendix C

## Effect of Using IDFT/DFT Larger than Number of Occupied

### Subcarriers

We observed in channel sounding that because we have guard subcarriers at the edges of each subchannel, it can cause a sinc-like spreading of the signal in the time-domain. This is also the reason for choosing IDFT/DFT that is equal to the subchannel size in AMC or sounding zone size in channel sounding in order to avoid this effect.

By definition, the N-point IDFT and DFT is defined as

$$\hat{h}[n] = \frac{1}{N} \sum_{k=0}^{N-1} \hat{H}(k) e^{j2\pi n \frac{k}{N}} \quad (\text{C.1})$$

$$\hat{H}(k) = \sum_{n=0}^{N-1} \hat{h}[n] e^{-j2\pi n \frac{k}{N}} \quad (\text{C.2})$$

If we let  $M$  ( $\leq N$ ) be the size of the subchannel or sounding zone, then we can

rewrite the N-point IDFT as

$$\hat{h}[n] = \frac{1}{N} \sum_{k=k_0}^{k_0+(M-1)} \hat{H}(k) e^{j2\pi n \frac{k}{N}} \quad (\text{C.3})$$

where  $k_0$  is the leftmost subcarrier of the subchannel or sounding zone.

Without loss of generality, we assume that the channel impulse response has one tap and therefore, the frequency response is a constant. With this assumption, we can simplify the above equation to

$$\begin{aligned} \hat{h}[n] &= \frac{1}{N} \sum_{k=k_0}^{k_0+(M-1)} \hat{H} e^{j2\pi n \frac{k}{N}} \\ &= \frac{\hat{H}}{N} e^{j2\pi n \frac{k_0}{N}} \sum_{k=0}^{M-1} e^{j2\pi n \frac{k}{N}} \\ &= \frac{\hat{H}}{N} e^{j2\pi n \frac{k_0}{N}} \left[ \frac{1 - e^{j2\pi n \frac{M}{N}}}{1 - e^{\frac{2\pi n}{N}}} \right] \end{aligned} \quad (\text{C.4})$$

This simplification shows that the time-domain impulse response is a sinc function spread over  $n$ . To use frequency smoothing with a rectangular window, we choose a window of length  $L$  and keep the first  $L$  samples of the impulse response. Therefore, the frequency response of the windowed impulse response is,

$$\begin{aligned} \hat{H}(k) &= \sum_{n=0}^{L-1} \hat{h}[n] e^{-j2\pi k \frac{n}{N}} \\ &= \sum_{n=0}^{L-1} \left\{ \frac{\hat{H}}{N} e^{j2\pi n \frac{k_0}{N}} \left[ \frac{1 - e^{j2\pi n \frac{M}{N}}}{1 - e^{\frac{2\pi n}{N}}} \right] \right\} e^{-j2\pi k \frac{n}{N}} \\ &= \frac{\hat{H}}{N} e^{j2\pi n \frac{k_0 - k}{N}} \sum_{n=0}^{L-1} \frac{1 - e^{j2\pi n \frac{M}{N}}}{1 - e^{\frac{2\pi n}{N}}} \end{aligned} \quad (\text{C.5})$$

We can see from the equation in the last line that instead of obtaining a constant frequency-domain channel response, we get a sinc function that is only summed from 0 to  $L-1$  instead of 0 to  $N-1$  and hence introducing a bias into the

estimates because we do not capture all of the channel's energy. For this reason, we choose the  $M$ -point IDFT/DFT in the frequency smoothing algorithm to be of the same length as the subchannel or sounding zone of interest to avoid spreading the signal more in the time-domain.

# Appendix D

## Mean-Squared Error Derivations

This section supplements the MSE derivations shown in Section 3.2.1, Section 3.2.2, Section 4.2.1, and Section 5.3.1. In these derivations, we assume  $|X(t, k)| = A$  for all  $(t, k)$  at pilot subcarriers.

### D.1 PUSC: Linear Interpolation

Given the PUSC tile shown in Figure 3.1, we define  $\mathcal{A} = \{(1, 0), (1, 3)\}$ ,  $\mathcal{B} = \{\mathcal{B}_0, \mathcal{B}_2\}$ , where  $\mathcal{B}_0 = \{(0, 1), (0, 2)\}$  and  $\mathcal{B}_2 = \{(2, 1), (2, 2)\}$ ,  $\mathcal{C} = \{(1, 1), (1, 2)\}$ , and  $\mathcal{P} = \{(0, 0), (0, 3), (2, 0), (2, 3)\}$ . If we perform linear interpolation at the data subcarriers using Equation 3.2 and Equation 3.3, then we can derive the MSE for subcarriers in  $\mathcal{A}, \mathcal{B}, \mathcal{C}$  to be given by

1. Mean-Squared Error for  $(t,k) \in \mathcal{A}$

$$\begin{aligned}
MSE_{\mathcal{A}} &= E[|\hat{H}(1,0) - H(1,0)|^2] = E[|\hat{H}(1,3) - H(1,3)|^2] \\
&= E\left[\left|\frac{1}{2}\hat{H}(0,0) + \frac{1}{2}\hat{H}(2,0) - H(1,0)\right|^2\right] \\
&= E\left[\left|\frac{1}{2}\left(H(0,0) + \frac{V(0,0)}{A}\right) + \frac{1}{2}\left(H(2,0) + \frac{V(2,0)}{A}\right) - H(1,0)\right|^2\right] \\
&= E\left[\frac{1}{4}H(0,0)H^*(0,0) + \frac{1}{4}H(2,0)H^*(0,0) - \frac{1}{2}H(1,0)H^*(0,0) \right. \\
&\quad + \frac{1}{4}H(0,0)H^*(2,0) + \frac{1}{4}H(2,0)H^*(2,0) - \frac{1}{2}H(1,0)H^*(2,0) \\
&\quad - \frac{1}{2}H(0,0)H^*(1,0) - \frac{1}{2}H(2,0)H^*(1,0) + H(1,0)H^*(1,0) \\
&\quad \left. + \frac{V(0,0)V^*(0,0)}{4A^2} + \frac{V(2,0)V^*(2,0)}{4A^2}\right]
\end{aligned}$$

Using Equation 2.3, we can simplify the equation to

$$\begin{aligned}
MSE_{\mathcal{A}} &= \frac{1}{4}R_H[0,0] + \frac{1}{4}R_H[2,0] - \frac{1}{2}R_H[1,0] \\
&\quad + \frac{1}{4}R_H[-2,0] + \frac{1}{4}R_H[0,0] - \frac{1}{2}R_H[-1,0] \\
&\quad - \frac{1}{2}R_H[-1,0] - \frac{1}{2}R_H[1,0] + R_H[0,0] + \frac{1}{2}\frac{\sigma_v^2}{A^2}
\end{aligned}$$

Because  $\Delta f = 0$ ,  $R_f[0] = 1$  and we can simplify the result to

$$MSE_{\mathcal{A}} = \sigma_H^2 \left\{ \frac{3}{2}R_t[0] + \frac{1}{4}(R_t[2] + R_t[-2]) - (R_t[1] + R_t[-1]) \right\} + \frac{1}{2}\frac{\sigma_v^2}{A^2}$$

2. Average mean-squared error for  $\{(t,k), (t,k+1)\} \in \mathcal{B}_t$  for  $t = \{0,2\}$ .

$$\begin{aligned}
MSE_{\mathcal{B}_t} &= \frac{1}{2}[MSE(t,k) + MSE(t,k+1)] \\
&= \frac{1}{2}\sum_{l=1}^2 E[|\hat{H}(0,l) - H(0,l)|^2] = \frac{1}{2}\sum_{l=1}^2 E[|\hat{H}(2,l) - H(2,l)|^2] \\
&= \frac{1}{2}E[|\hat{H}(0,1) - H(0,1)|^2] + \frac{1}{2}E[|\hat{H}(0,2) - H(0,2)|^2] \\
&= \frac{1}{2}E\left[\left|\frac{2}{3}\left(H(0,0) + \frac{V(0,0)}{A}\right) + \frac{1}{3}\left(H(0,3) + \frac{V(0,3)}{A}\right) - H(0,1)\right|^2\right] \\
&\quad + \frac{1}{2}E\left[\left|\frac{1}{3}\left(H(0,0) + \frac{V(0,0)}{A}\right) + \frac{2}{3}\left(H(0,3) + \frac{V(0,3)}{A}\right) - H(0,2)\right|^2\right]
\end{aligned}$$

Using Equation 2.3, we can simplify the derivation to

$$\begin{aligned}
MSE_{\mathcal{B}_t} &= \frac{1}{2} \left\{ \frac{4}{9} R_H[0, 0] + \frac{2}{9} R_H[0, 3] - \frac{2}{3} R_H[0, 1] + \frac{2}{9} R_H[0, -3] \right. \\
&\quad + \frac{1}{9} R_H[0, 0] - \frac{1}{3} R_H[0, -2] - \frac{2}{3} R_H[0, -1] - \frac{1}{3} R_H[0, 2] \\
&\quad + R_H[0, 0] + \frac{4}{9} \frac{\sigma_v^2}{A^2} + \frac{1}{9} \frac{\sigma_v^2}{A^2} \left. \right\} \\
&\quad + \frac{1}{2} \left\{ \frac{1}{9} R_H[0, 0] + \frac{4}{9} R_H[0, 3] - \frac{1}{3} R_H[0, 1] + \frac{2}{9} R_H[0, -3] \right. \\
&\quad + \frac{4}{9} R_H[0, 0] - \frac{2}{3} R_H[0, -2] - \frac{1}{3} R_H[0, -1] - \frac{2}{3} R_H[0, 2] \\
&\quad + R_H[0, 0] + \frac{1}{9} \frac{\sigma_v^2}{A^2} + \frac{4}{9} \frac{\sigma_v^2}{A^2} \left. \right\}
\end{aligned}$$

Since  $\Delta t = 0$ ,  $R_t[0] = 1$  and the result can be simplified to

$$\begin{aligned}
MSE_{\mathcal{B}_t} &= \sigma_H^2 \left\{ \frac{14}{9} R_f[0] + \frac{2}{9} (R_f[3] + R_f[-3]) \right. \\
&\quad \left. - \frac{4}{3} (R_f[1] + R_f[-1]) - \frac{2}{3} (R_f[2] + R_f[-2]) \right\} + \frac{5}{9} \frac{\sigma_v^2}{A^2}
\end{aligned}$$

### 3. Mean-squared error at $(t, k) \in \mathcal{C}$

$$\begin{aligned}
MSE_{\mathcal{C}} &= E[|\hat{H}(1, 1) - H(1, 1)|^2] = E[|\hat{H}(1, 2) - H(1, 2)|^2] \\
&= E \left[ \left| \frac{1}{3} \left( H(0, 0) + \frac{V(0, 0)}{A} \right) + \frac{1}{6} \left( H(0, 3) + \frac{V(0, 3)}{A} \right) \right. \right. \\
&\quad \left. \left. + \frac{1}{3} \left( H(2, 0) + \frac{V(2, 0)}{A} \right) + \frac{1}{6} \left( H(2, 3) + \frac{V(2, 3)}{A} \right) - H(1, 1) \right|^2 \right] \\
&= E \left[ \left\{ \left( \frac{1}{9} H(0, 0) + \frac{1}{9} H(2, 0) + \frac{1}{18} H(0, 3) + \frac{1}{18} H(2, 3) - \frac{1}{3} H(1, 1) \right) H^*(0, 0) \right. \right. \\
&\quad + \left( \frac{1}{9} H(0, 0) + \frac{1}{9} H(2, 0) + \frac{1}{18} H(0, 3) + \frac{1}{18} H(2, 3) - \frac{1}{3} H(1, 1) \right) H^*(2, 0) \\
&\quad + \left( \frac{1}{18} H(0, 0) + \frac{1}{18} H(2, 0) + \frac{1}{36} H(0, 3) + \frac{1}{36} H(2, 3) - \frac{1}{6} H(1, 1) \right) H^*(0, 3) \\
&\quad + \left( \frac{1}{18} H(0, 0) + \frac{1}{18} H(2, 0) + \frac{1}{36} H(0, 3) + \frac{1}{36} H(2, 3) - \frac{1}{6} H(1, 1) \right) H^*(2, 3) \\
&\quad - \left( \frac{1}{3} H(0, 0) + \frac{1}{3} H(2, 0) + \frac{1}{6} H(0, 3) + \frac{1}{6} H(2, 3) - H(1, 1) \right) H^*(1, 1) \\
&\quad + \frac{V(0, 0)V^*(0, 0)}{9A^2} + \frac{V(2, 0)V^*(2, 0)}{9A^2} + \frac{V(0, 3)V^*(0, 3)}{36A^2} \\
&\quad \left. \left. + \frac{V(2, 3)V^*(2, 3)}{36A^2} \right\} \right]
\end{aligned}$$

Using Equation 2.3, we can simplify the equation to

$$\begin{aligned}
MSE_C &= \frac{1}{9}R_H[0,0] + \frac{1}{9}R_H[2,0] + \frac{1}{18}R_H[0,3] + \frac{1}{18}R_H[2,3] - \frac{1}{3}R_H[1,1] \\
&+ \frac{1}{9}R_H[-2,0] + \frac{1}{9}R_H[0,0] + \frac{1}{18}R_H[-2,3] + \frac{1}{18}R_H[0,3] - \frac{1}{3}R_H[-1,1] \\
&+ \frac{1}{18}R_H[0,-3] + \frac{1}{18}R_H[2,-3] + \frac{1}{36}R_H[0,0] + \frac{1}{36}R_H[2,0] \\
&- \frac{1}{6}R_H[1,-2] + \frac{1}{18}R_H[-2,-3] + \frac{1}{18}R_H[0,-3] + \frac{1}{36}R_H[-2,0] \\
&+ \frac{1}{36}R_H[0,0] - \frac{1}{6}R_H[-1,-2] - \frac{1}{3}R_H[-1,-1] - \frac{1}{3}R_H[1,-1] \\
&- \frac{1}{6}R_H[-1,2] - \frac{1}{6}R_H[1,2] + R_H[0,0] + \frac{5}{18}\frac{\sigma_v^2}{A^2} \\
&= \frac{23}{18}R_H[0,0] + \frac{5}{36}(R_H[2,0] + R_H[-2,0]) + \frac{1}{9}(R_H[0,3] + R_H[0,-3]) \\
&+ \frac{1}{18}(R_H[2,3] + R_H[-2,3] + R_H[2,-3] + R_H[-2,-3]) \\
&- \frac{1}{3}(R_H[1,1] + R_H[1,-1] + R_H[-1,1] + R_H[-1,-1]) \\
&- \frac{1}{6}(R_H[1,2] + R_H[1,-2] + R_H[-1,2] + R_H[-1,-2]) + \frac{5}{18}\frac{\sigma_v^2}{A^2}
\end{aligned}$$

## D.2 PUSC: 4-Pilot Averaging

Given the PUSC tile shown in Figure 3.1, we define  $\mathcal{D} = \{(0,1), (0,2), (1,0), (1,1), (1,2), (1,3), (2,1), (2,2)\}$ . Assuming we use the 4-pilot averaging estimator in Equation 3.8 to estimate the channel at each data subcarrier, the MSE at each subcarrier is given by

$$\begin{aligned}
MSE_D(t, k) &= E[|\hat{H}(t, k) - H(t, k)|^2] \\
&= E\left[\left|\frac{1}{4}\hat{H}(0, 0) + \frac{1}{4}\hat{H}(2, 0) + \frac{1}{4}\hat{H}(0, 3) + \frac{1}{4}\hat{H}(2, 3) - H(t, k)\right|^2\right] \\
&= E\left[\left|\frac{1}{4}\left(H(0, 0) + \frac{V(0, 0)}{A}\right) + \frac{1}{4}\left(H(2, 0) + \frac{V(2, 0)}{A}\right) \right. \right. \\
&\quad \left. \left. + \frac{1}{4}\left(H(0, 3) + \frac{V(0, 3)}{A}\right) + \frac{1}{4}\left(H(2, 3) + \frac{V(2, 3)}{A}\right) - H(t, k)\right|^2\right] \\
&= E\left\{\left(\frac{1}{16}H(0, 0) + \frac{1}{16}H(2, 0) + \frac{1}{16}H(0, 3) + \frac{1}{16}H(2, 3) - \frac{1}{4}H(t, k)\right)H^*(0, 0) \right. \\
&\quad + \left(\frac{1}{16}H(0, 0) + \frac{1}{16}H(2, 0) + \frac{1}{16}H(0, 3) + \frac{1}{16}H(2, 3) - \frac{1}{4}H(t, k)\right)H^*(2, 0) \\
&\quad + \left(\frac{1}{16}H(0, 0) + \frac{1}{16}H(2, 0) + \frac{1}{16}H(0, 3) + \frac{1}{16}H(2, 3) - \frac{1}{4}H(t, k)\right)H^*(0, 3) \\
&\quad + \left(\frac{1}{16}H(0, 0) + \frac{1}{16}H(2, 0) + \frac{1}{16}H(0, 3) + \frac{1}{16}H(2, 3) - \frac{1}{4}H(t, k)\right)H^*(2, 3) \\
&\quad - \left(\frac{1}{4}H(0, 0) + \frac{1}{4}H(2, 0) + \frac{1}{4}H(0, 3) + \frac{1}{4}H(2, 3) - H(t, k)\right)H^*(t, k) \\
&\quad \left. + \frac{V(0, 0)V^*(0, 0)}{16A^2} + \frac{V(2, 0)V^*(2, 0)}{16A^2} + \frac{V(0, 3)V^*(0, 3)}{16A^2} + \frac{V(2, 3)V^*(2, 3)}{16A^2}\right\}
\end{aligned}$$

Using Equation 2.3, we can simplify the result to

$$\begin{aligned}
MSE_D(t, k) &= \frac{1}{16}R_H[0, 0] + \frac{1}{16}R_H[2, 0] + \frac{1}{16}R_H[0, 3] + \frac{1}{16}R_H[2, 3] - \frac{1}{4}R_H[t, k] \\
&+ \frac{1}{16}R_H[-2, 0] + \frac{1}{16}R_H[0, 0] + \frac{1}{16}R_H[-2, 3] + \frac{1}{16}R_H[0, 3] - \frac{1}{4}R_H[t - 2, k] \\
&+ \frac{1}{16}R_H[0, -3] + \frac{1}{16}R_H[2, -3] + \frac{1}{16}R_H[0, 0] + \frac{1}{16}R_H[2, 0] - \frac{1}{4}R_H[t, k - 3] \\
&+ \frac{1}{16}R_H[-2, -3] + \frac{1}{16}R_H[0, -3] + \frac{1}{16}R_H[-2, 0] + \frac{1}{16}R_H[0, 0] \\
&- \frac{1}{4}R_H[t - 2, k - 3] - \frac{1}{4}R_H[-t, -k] - \frac{1}{4}R_H[2 - t, -k] - \frac{1}{4}R_H[-t, 3 - k] \\
&+ \frac{1}{4}R_H[2 - t, 3 - k] + R_H[0, 0] + \frac{1}{4}\frac{\sigma_v^2}{A^2} \\
&= \frac{5}{4}R_H[0, 0] + \frac{1}{8}(R_H[2, 0] + R_H[-2, 0]) + \frac{1}{8}(R_H[0, 3] + R_H[0, -3]) \\
&+ \frac{1}{16}(R_H[2, 3] + R_H[-2, 3] + R_H[2, -3] + R_H[-2, -3]) \\
&- \frac{1}{4}(R_H[t, k] + R_H[-t, -k]) - \frac{1}{4}(R_H[t - 2, k] + R_H[2 - t, -k]) \\
&- \frac{1}{4}(R_H[t, k - 3] + R_H[-t, 3 - k]) - \frac{1}{4}(R_H[t - 2, k - 3] + R_H[2 - t, 3 - k]) \\
&+ \frac{1}{4}\frac{\sigma_v^2}{A^2}
\end{aligned}$$

### D.3 AMC and Channel Sounding: Linear Interpolation

When using linear interpolation in AMC and channel sounding, we calculate the MSE at the pilot subcarriers, interpolated subcarriers, and extrapolated subcarriers. In this derivation, we let  $P$  be the total number of pilots,  $M$  total number of subcarriers and  $D = M/P$  be the number of subcarriers for every pilot. For AMC,  $M = 18$  and  $P = 6$ . For simplicity, we let the leftmost subcarrier be a pilot subcarrier.

1. MSE at Pilot Subcarriers

$$\begin{aligned}
MSE_P &= E[|\hat{H}(t, Dp) - H(t, Dp)|^2] \text{ for } p = 0, 1, \dots, P-1 \\
&= E[|H(t, Dp) + \frac{V(t, Dp)}{A} - H(t, Dp)|^2] \\
&= E[|\frac{V(t, Dp)}{A}|^2] = \frac{\sigma_v^2}{A^2}
\end{aligned} \tag{D.1}$$

2. MSE at Interpolated Subcarriers

$$\begin{aligned}
MSE_I &= \frac{1}{D-1} \sum_{l=1}^{D-1} E[|\hat{H}(t, Dp+l) - H(t, Dp+l)|^2] \\
&= \frac{1}{D-1} \sum_{l=1}^{D-1} E[|\frac{D-l}{D} \hat{H}(t, Dp) + \frac{l}{D} \hat{H}(t, D(p+1)) - H(t, Dp+l)|^2] \\
&= \frac{1}{D-1} \sum_{l=1}^{D-1} E[|\frac{D-l}{D} (H(t, Dp) + \frac{V(t, Dp)}{A}) \\
&\quad + \frac{l}{D} (H(t, D(p+1)) + \frac{V(t, D(p+1))}{A}) - H(t, Dp+l)|^2] \\
&= \frac{1}{D-1} \sum_{l=1}^{D-1} \{ (\frac{D-l}{D})^2 R_H[0, 0] + \frac{(D-l)l}{D^2} R_H[0, -D] - \frac{D-l}{D} R_H[0, -l] \\
&\quad + \frac{l(D-l)}{D^2} R_H[0, D] + (\frac{l}{D})^2 R_H[0, 0] - \frac{l}{D} R_H[0, D-l] \\
&\quad - \frac{D-l}{D} R_H[0, l] - \frac{l}{D} R_H[0, l-D] + R_H[0, 0] \} \\
&\quad + \frac{1}{D-1} \sum_{l=1}^{D-1} \{ (\frac{D-l}{D})^2 \frac{\sigma_v^2}{A^2} + (\frac{l}{D})^2 \frac{\sigma_v^2}{A^2} \}
\end{aligned}$$

Since  $\Delta t = 0$  and  $\sum_{l=1}^{D-1} l = \frac{(D-1)D}{2}$  and  $\sum_{l=1}^{D-1} l^2 = \frac{(D-1)(D)(2D-1)}{6}$ , we can

simplify the above equation to

$$MSE_I = \sigma_H^2 \left\{ \frac{5D-1}{3D} R_f[0] + \frac{D+1}{6D} (R_f[D] + R_f[-D]) - \alpha \right\} + \frac{2D-1}{3D} \frac{\sigma_v^2}{A^2}$$

where

$$\alpha = \frac{1}{D-1} \sum_{l=1}^{D-1} \left\{ \frac{D-l}{D} (R_f[l] + R_f[-l]) + \frac{l}{D} (R_f[D-l] + R_f[l-D]) \right\}$$

3. MSE at Extrapolated Subcarriers assuming that all extrapolated subcarriers are on right-hand side of subchannel (decimation offset = 0)

$$\begin{aligned}
MSE_E &= \frac{1}{D-1} \sum_{l=1}^{D-1} E[|\hat{H}(t, (P-1)D+l) - H(t, (P-1)D+l)|^2] \\
&= \frac{1}{D-1} \sum_{l=1}^{D-1} E[|\frac{-l}{D}\hat{H}(t, (P-2)D) + \frac{D+l}{D}\hat{H}(t, (P-1)D) \\
&\quad - H(t, (P-1)D+l)|^2] \\
&= \frac{1}{D-1} \sum_{l=1}^{D-1} E[|\frac{-l}{D}(H(t, (P-2)D) + \frac{V(t, (P-2)D)}{A}) \\
&\quad + \frac{D+l}{D}(H(t, (P-1)D) + \frac{V(t, (P-1)D)}{A}) - H(t, (P-1)D+l)|^2] \\
&= \frac{1}{D-1} \sum_{l=1}^{D-1} \{(\frac{l}{D})^2 R_H[0,0] - \frac{(D+l)l}{D^2} R_H[0,D] + \frac{l}{D} R_H[0,D+l] \\
&\quad - \frac{(D+l)l}{D^2} R_H[0,-D] + (\frac{D+l}{D})^2 R_H[0,0] - \frac{D+l}{D} R_H[0,l] \\
&\quad + \frac{l}{D} R_H[0,-D-l] - \frac{D+l}{D} R_H[0,-l] + R_H[0,0]\} \\
&\quad + \frac{1}{D-1} \sum_{l=1}^{D-1} \{(\frac{l}{D})^2 \frac{\sigma_v^2}{A^2} + (\frac{D+l}{D})^2 \frac{\sigma_v^2}{A^2}\}
\end{aligned}$$

Since  $\Delta t = 0$  and  $\sum_{l=1}^{D-1} l = \frac{(D-1)D}{2}$  and  $\sum_{l=1}^{D-1} l^2 = \frac{(D-1)(D)(2D-1)}{6}$ , we can simplify the above equation to

$$MSE_E = \sigma_H^2 \left\{ \frac{11D-1}{3D} R_f[0] - \frac{5D-1}{6D} (R_f[D] + R_f[-D]) - \beta \right\} + \frac{8D-1}{3D} \frac{\sigma_v^2}{A^2}$$

where

$$\beta = \frac{1}{D-1} \sum_{l=1}^{D-1} \left\{ \frac{D+l}{D} (R_f[l] + R_f[-l]) - \frac{l}{D} (R_f[D+l] + R_f[-D-l]) \right\}$$

# Bibliography

- [1] Jiang Du, Qicong Peng, Hongying Zhang, *Adaptive Blind Channel Identification and Equalization for OFDM-MIMO Wireless Communication Systems*, IEEE 2003 International Symposium on Personal, Indoor and Mobile Radio Communication Proceedings, 2003.
- [2] M.J. Fernandez-Getino Garcia, Jose M. Paez-Borrillo, Santiago Zazo, *Efficient Pilot Patterns for Channel Estimation in OFDM Systems over HF Channels*, IEEE Electronic Letters, Vol. 36, No. 12, June 2000.
- [3] Rohit Negi, John Cioffi, *Pilot Tone Selection for Channel Estimation in a Mobile OFDM System*, IEEE Transactions on Consumer Electronics, Vol. 44, No. 3, August 1998.
- [4] Wei Zhang, Xiang-Gen Xia, P.C. Ching, *Optimal Training and Pilot Pattern Design for OFDM Systems in Rayleigh Fading*, IEEE Transactions on Broadcasting, Vol. 52, No. 4, December 2006.
- [5] M.J. Fernandez-Getino Garcia, Jose M. Paez-Borrillo, Santiago Zazo, *DFT-based Channel Estimation in 2D-Pilot-Symbol-Aided OFDM Wireless Systems*, IEEE Vehicular Technology Conference, Vol. 2, 2001.
- [6] Bertrand Muquet, Marc de Courville, and Pierre Duhamel, *Subspace-Based Blind and Semi-Blind Channel Estimation for OFDM Systems*, IEEE Transactions on Signal Processing, Vol. 50, No. 7, July 2002.
- [7] Xiaolin Hou, Shubo Li, Danpu Liu, Changchuan Yin, Guangxin Yue, *On Two-Dimensional Adaptive Channel Estimation in OFDM Systems*, Proc. of IEEE Vehicular Technology Conference, Vol. 1, May 2000.
- [8] M.A. Saeed, B.M. Ali, S. Khatun, N.K. Noordin, M. Ismail, *MIMO OFDM Channel Estimation Based on RLS Algorithm: The Time-versus Frequency-domain Implementations*, Asia-Pacific Conference on Communications, October 2007.

- [9] Wei Chen, Ruifeng Zhang, *Kalman-Filter Channel Estimator for OFDM Systems in Time and Frequency-Selective Fading Environment*, Proc of 2004 IEEE ICASSP, Vol. 4, May 2004.
- [10] Tzi-Dar Chiueh, Pei-Yun Tsai, *OFDM Baseband Receiver Design for Wireless Communications*, Singapore: John Wiley and Sons, 2007.
- [11] Ye (Geoffrey) Li, Leonard J. Cimini and Nelson R. Sollenberger, *Robust Channel Estimation for OFDM Systems with Rapid Dispersive Fading Channel*, IEEE Transaction On Communications, Vol. 46, No. 7, July 1998.
- [12] Ove Edfors, Magnus Sandell, Jan-Jaap van de Beek, Sarah Kate Wilson and Per Ola Brjesson, *OFDM Channel Estimation by Singular Value Decomposition*, IEEE Transactions On Communications, Vol. 46, No. 7, July 1998.
- [13] *Mobile WiMAX Part I: A Technical Overview and Performance Evaluation*, WiMAX Forum, 2006.
- [14] Juha Heiskala and John Terry. *OFDM Wireless LANs: A Theoretical and Practical Guide* Indianapolis: Sams Publishing, 2002.
- [15] Carl Eklund, et al. *WirelessMAN: Inside the IEEE 802.16 Standard for Wireless Metropolitan Networks*. New York: Standards Information Network/IEEE Press, 2006.
- [16] F. Vook, et.al., *Uplink Channel Sounding for Enabling Closed-Loop Transmit Antenna Array Techniques*, IEEE 802.22 Wireless RANs doc number IEEE 802.22-06/0013r0.
- [17] Meng-Han Hsieh and Che-Ho Wei, *Channel Estimation for OFDM Systems Based on Comb-type Pilot Arrangement in Frequency Selective Fading Channels*, IEEE Transactions on Consumer Electronics, Vol. 44, No. 1, February 1998
- [18] Theodore S. Rappaport, *Wireless Communications: Principles and Practice*, Upper Saddle River: Prentice Hall PTR, 2002.
- [19] Simon Haykin, Michael Moher, *Modern Wireless Communications*, Upper Saddle River: Pearson Prentice Hall, 2005.
- [20] Yun Hee Kim, *Performance Analysis of a Coded OFDM System in Time-Varying Multipath Rayleigh Fading Channels*, IEEE Transactions on Vehicular Technology, Vol. 48, No. 5, September 1999.

- [21] Mark Russell and Gordon L. Stuber, *Interchannel Interference Analysis of OFDM in a Mobile Environment*, IEEE Vehicular Technology Conference, Vol. 2, July 1995.
- [22] Jihyung Kim, Jeongho Park, *Performance Analysis of Channel Estimation in OFDM Systems*, IEEE Signal Processing Letters, Vol. 12, No. 1, January 2005.
- [23] F. Vook, et.al., *GCL Sequences for Uplink Channel Sounding in TDD OFDMA*, IEEE C802.16e-04/266r1 (08/28/04).
- [24] R1-062014, Texas Instruments, *Uplink Reference Signal Design in EUTRA*,
- [25] Hassan Yaghoobi, *Scalable OFDMA Physical Layer in IEEE 802.16 WirelessMAN*, Intel Technology Journal, Vol. 8, Issue 3, 2004.
- [26] Intel Corp., *Understanding Wi-Fi and WiMAX as Metro-Access Solutions*, 2004.
- [27] IEEE 802.20-PD-08, *802.20 Channel Models Document for IEEE 802.20 MBWA System Simulations - 802.20-PD-08*, September 2005.
- [28] Andrea Goldsmith, *Wireless Communications*. New York: Cambridge University Press, 2005.