國 立 交 通 大 學

電子工程學系 電子研究所碩士班

碩 士 論 文

IEEE 802.16e 與 802.16m OFDMA 下行通道之線性最小均方差通道估測

LMMSE Channel Estimation for IEEE 802.16e and 802.16m OFDMA Downlink

研 究 生：王志偉

指導教授：林大衛 博士

中 華 民 國 九 十 八 年 十 月

IEEE 802.16e 與 802.16m OFDMA 下行通道之線性最小均方差通道估測

LMMSE Channel Estimation for IEEE 802.16e and 802.16m OFDMA Downlink

研究生：王志偉　　Student: Chih-Wei Wang

指導教授：林大衛 博士　　Advisor: Dr. David W. Lin

國 立 交 通 大 學

電子工程學系　電子研究所碩士班

碩士論文

A Thesis

Submitted to Department of Electronics Engineering & Institute of Electronics

College of Electrical and Computer Engineering

National Chiao Tung University

in Partial Fulfillment of the Requirements

for the Degree of Master

in

Electronics Engineering

October 2009

Hsinchu, Taiwan, Republic of China

中華民國九十八年十月

IEEE 802.16e 與 IEEE 802.16m OFDMA 下行通道之線性最小均方差通道估測

研究生：王志偉　　指導教授：林大衛 博士

國立交通大學

電子工程學系　電子研究所碩士班

摘要

正交分頻多重進接(OFDMA)技術近來在行動環境中廣受注目且已經應用在許多數位通訊應用中。採用 OFDMA 一個最主要的原因是其抗頻率選擇性衰變及窄頻干擾的能力。我們聚焦在 IEEE 802.16e 與 802.16m OFDMA 下行傳輸的通道估測部分。

本篇論文採用兩種通道估測方法。第一種為進階 4 點群線性內插法，首先我們使用最小平方差的估測器來估計在導訊上的通道頻率響應，這是為了硬體的計算方便。最後我們在頻率域上使用線性內插法來得到在資料載波上的通道響應。第二種為線性最小均方差通到估測法，首先我們使用線性內插法來估計在導訊上的通道頻率響應，其次估計延遲參數，再利用指數函數的功率延遲曲線求出自相關函數，最後根據自相關函數做線性最小均方差以估測出資料載坡上的通道響應。我們先在 AWGN 通道上驗證我們的模擬模型，然後分別在 IEEE802.16e 與 802.16m 兩種標準下於多重路徑通道上模擬。

在本篇論文中，我們首先簡介 IEEE 802.16e 與 802.16m OFDMA 下行的標

準機制。接著，我們依造兩種標準機制分別各傳輸情形下介紹所用的通道估測方法並探討其估測效能。

LMMSE Channel Estimation For IEEE 802.16e and 802.16m OFDMA Downlink

Student：Chih-Wei Wang Advisor：Dr. David W. Lin

Department of Electronics Engineering

Institute of Electronics

National Chiao Tung University

Abstract

OFDMA (orthogonal frequency division multiple access) technique has drawn much interest recently in the mobile transmission environment and been successfully applied to a wide variety of digital communications applications over the past several years. One of the main reason to use OFDMA is its robustness against frequency selective fading and narrowband interference. We focus on the OFDMA downlink channel estimation based on IEEE 802.16e and 802.16m.

We present two channel estimation methods. One is advanced 4-point linear interpolation. First, we use LS estimator on pilot subcarriers because of its low computational complexity. Second, we estimate the channel response in the middle of the pilot location in the time domain. Finally we estimate the channel response on data subcarriers using linear interpolation in the frequency domain. The other method is LMMSE channel estimation, First, we use LS estimator on pilot subcarriers, and do

linear interpolation at the pilot location in time. Second, we estimate the delay parameters. Third, we find the autocorrelation function associated with the approximate PDP. Finally base on this autocorrelation function, do LMMSE filtering to estimate the data subcarrier response. We verify our simulation model on AWGN channel and then do the simulation on multipath channels for IEEE 802.16e and 802.16m.

In this thesis, we first introduce the standard of the IEEE 802.16e and 802.16m OFDMA downlink. Then we describe the channel estimation methods we use and discuss the performance in each transmission condition for IEEE 802.16e and 802.16m.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In response to ITU-R's plan for the fourth-generation mobile communication standard IMT-Advanced, the IEEE 802.16 standards group has set up the 802.16m (i.e., Advanced WiMAX) task group. The new frame structure developed by IEEE 802.16m can be compatible with IEEE 802.16e, reduce communication latency, support relay, and coexist with other radio access techniques (in particular, LTE). We study the channel estimation technique for IEEE 802.16e and 802.16m. In particular, we consider the linear minimum mean-square error (LMMSE) approach wherein we employ the technique proposed in [14]. Below, we follow [14] to introduce the pilot-aided LMMSE channel estimation problem.

Pilot-aided channel estimation is widely employed in today's coherent wireless orthogonal frequency-division multiplexing (OFDM) systems. The subcarriers that carry pilot signals are usually dispersed in frequency and in time. The LMMSE technique is also known as Wiener filtering. Given some initial channel estimates at pilot subcarriers, the LMMSE channel estimate at any subcarrier d, is given by [1]–[5].

$$\widehat{H}_d = \underline{w}_d^H \underline{\widehat{H}}_p \tag{1.1}$$

with

$$\underline{w}_d^H = (\mathbf{R}_p + \sigma_n^2 \mathbf{I})^{-1} \underline{r}_{dp}, \tag{1.2}$$

where $\widehat{H}_d$ is the desired channel estimate; $\underline{w}_d$ is the Wiener filter; superscript H denotes Hermitian transpose; $\underline{\widehat{H}}_p$ is the vector of given initial channel estimates; $\mathbf{R}_p = E(\underline{H}_p \underline{H}_p^H)$, with E denoting expectation and $\underline{H}_p$ being the vector of true channel responses at pilot subcarriers; $\underline{r}_{dp} = E(\underline{H}_p H_d^*)$ with H_d being the true channel response at subcarrier d; σ_n^2 is the variance of additive noise in $\underline{\widehat{H}}_p$, assumed white Gaussian (i.e., AWGN); and $\mathbf{I}$ denotes an identity matrix, although it does not matter for now, we note that a convenient and frequently used method to estimate $\underline{\widehat{H}}_p$ is the least-squares (LS) method, which merely divides the received signal at each pilot subcarrier by the known pilot value there to obtain the estimated response there [3]. If the pilots have unity magnitude, then σ_n^2 is equal to the variance of the additive channel noise.

To carry out the LMMSE estimation, one needs to know $\mathbf{R}_p$, $\underline{r}_{dp}$, and σ_n^2. The estimation of σ_n^2 can be achieved by measuring the received power at the null subcarriers. The estimation of $\mathbf{R}_p$ and $\underline{r}_{dp}$, however, presents a problem. One aspect of the problem has to do with the fact that an accurate estimate requires averaging over sufficiently many samples. But when the channel is time-varying, one may not have this luxury within the coherence time, especially if the system only transmits a small number of pilots. Another aspect of the problem concerns $\underline{r}_{dp}$. Each element of this vector is the crosscorrelation of the channel response at a pilot subcarrier and that at subcarrier d. The estimation of $\underline{r}_{dp}$ usually requires interpolation (which involves approximation) in addition to averaging. The problem in estimating $\mathbf{R}_p$ and $\underline{r}_{dp}$ makes strict-sense LMMSE channel estimation impractical in some cases.

To sidestep the above problem, one approach is to employ a simple model for the channel power-delay profile (PDP). A choice is the exponential model [3], [6], [7], for which the entire second-order channel statistics are defined by the mean delay τ_μ and the root-mean-square (RMS) delay spread τ_{rms}. Given τ_μ and τ_{rms} as well as σ_n^2, one can calculate $\mathbf{R}_p$ and $\underline{r}_{dp}$ and then calculate the Wiener filter $\underline{w}_d$. The price paid is that the PDP model employed may be

an oversimplification of the actual situation, which results in modeling error. But [14] shows that exponential PDP hased LMMSE channel estimation can yield good performance and is amenable to typical pilot-transmittily OFDM signal structures.

In this thesis, we consider using the method derived in [14] for downlink (DL) channel estimation in IEEE 802.16m. And we tie up some loose parts in [14] concerning the DL channel estimation in IEEE 802.16e OFDMA, where these loose parts concern the values of σ_n^2 in R_p. As a key step in the overall channel estimation process, we employ the method proposed in [14] to estimate the mean delay and the RMS delay spread from noisy channel response estimates at the pilot subcarriers.

In what follows, we introduce the IEEE 802.16e and 802.16m OFDMA downlink specifications in chapters 2 and 3 , respectively. In chapter 4, we introduce the channel estimation methods. In chapters 5 and 6, we discuss the performance of channel estimation methods for IEEE 802.16e and 802.16m downlink transmissions, respectively. At last, we give the conclusion and discuss some potential future work in chapter 7.

E S A 1896

Chapter 2

Introduction to IEEE 802.16e OFDMA

We first introduce some basic concepts of the orthogonal frequency-division multiple access (OFDMA) technique for multicarrier modulation. The downlink specifications of IEEE 802.16e are introduced afterwards.

2.1 Overview of OFDMA [11], [12]

OFDMA is considered one most appropriate scheme for future wireless systems, including fourth-generation (4G) broadband wireless networks. In an OFDMA system, users simultaneously transmit their data by modulating mutually exclusive sets of orthogonal subcarriers, so that each user's signal can be separated in the frequency domain. One typical structure is subband OFDMA, where all available subcarriers are divided into a number of subbands and each user is allowed to use one or more available subbands for the data transmission. Usually, pilot symbols are employed for the estimation of channel state information (CSI) within the subband. IEEE 802.16e and 802.16m are examples of such systems. Figure 2.1 shows an OFDMA system in which active users simultaneously communicate with the base station (BS).

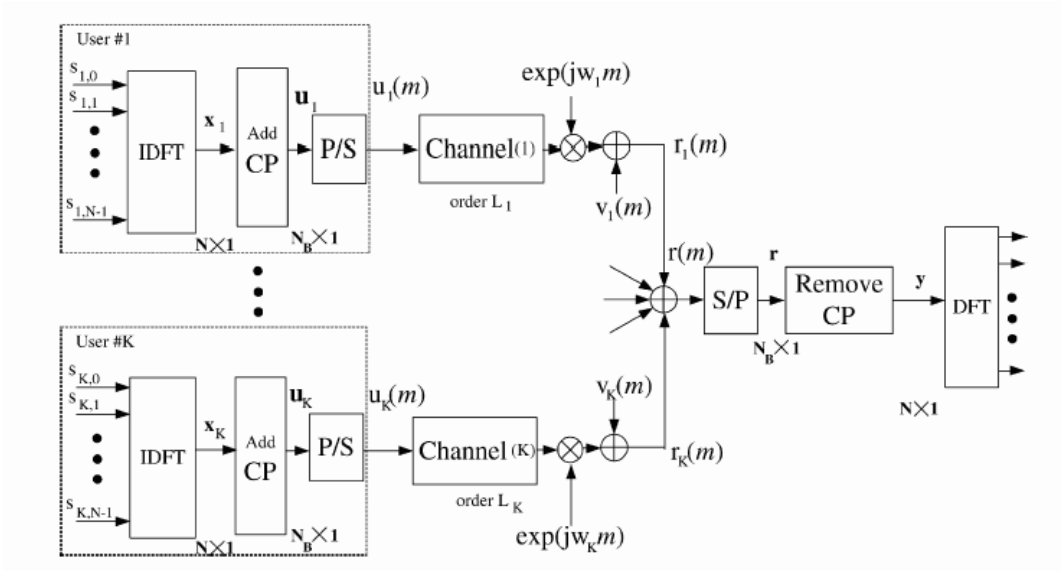


Figure 2.1: Discrete-time model of the baseband OFDMA system (from [11]).

2.1.1 Cyclic Prefix

Cyclic prefix (CP) or guard time is used in OFDM and OFDMA systems to overcome the intersymbol and interchannel intercarrier problems. The multiuser channel is usually assumed to be substantially invariant within one-block (or one-symbol) duration. The channel delay spread plus symbol timing mismatch is usually assumed to be smaller than the CP duration. In this condition, users do not interfere with each other in the frequency domain, when there is proper time and frequency synchronization.

A CP is a copy of the last part of the OFDMA symbol (see Fig. 2.2). A copy of the last T_g of the useful symbol period is used to collect multipaths while maintaining the orthogonality of subcarriers. However, the transmitter energy increases with the length of the guard time while the receiver energy remains the same (the CP is discarded in the receiver). So there is a $10\log(1-T_g/(T_b+T_g))/\log(10)$ dB loss in E_b/N_0.

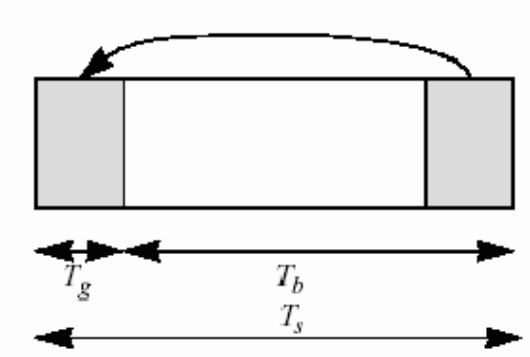


Figure 2.2: OFDMA symbol time structure (Fig. 428 in [10]).

2.1.2 Discrete-Time Baseband Equivalent System Model

The material in this subsection is mainly taken from [12]. Consider an OFDMA system with M active users sharing a bandwidth of $B = \frac{1}{T}$ Hz (T is the sampling period) as shown in Fig. 2.3. The system consists of K subcarriers of which K_u are useful subcarriers (excluding guard bands and DC subcarrier). The users are allocated non-overlapping subcarriers according to their needs.

The discrete-time baseband channel consists of, say, L multipath components and has the form

$$h(l) = \sum_{m=0}^{L-1} h_m \delta(l - l_m) \tag{2.1}$$

where h_m is a zero-mean complex Gaussian random variable with $E[h_i h_j^*] = 0$ for $i \neq j$. In the frequency domain,

$$\mathbf{H} = \mathbf{F}\mathbf{h} \tag{2.2}$$

where $\mathbf{H} = [H_0, H_1, ..., H_{K-1}]^T$, $\mathbf{h} = [h_0, ..., h_{L-1}, 0, ..., 0]^T$ and $\mathbf{F}$ is K-point discrete fourier transform (DFT) matrix. The impulse response length l_{L-1} is upper bounded by the length of CP (L_{cp}).

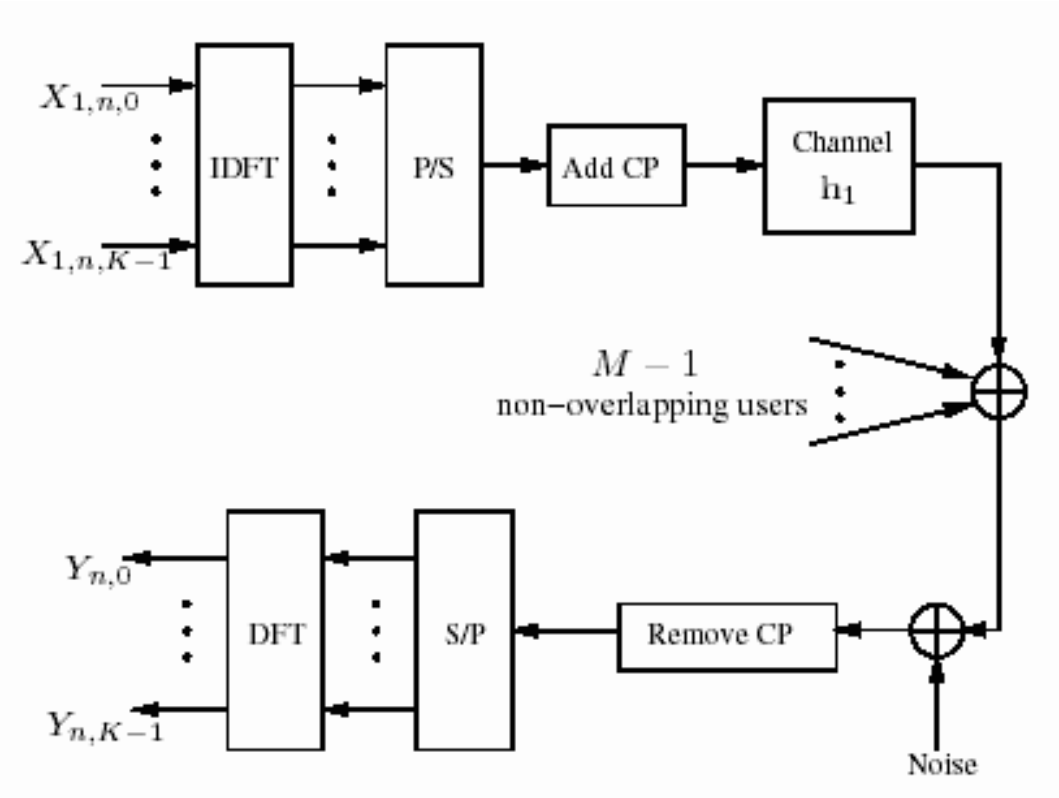


Figure 2.3: Discrete-time baseband equivalent of an OFDMA system with M users (from [12]).

The received signal in the frequency domain is given by

$$\mathbf{Y}_n = \sum_{i=1}^{M} \mathbf{X}_{i,n}\mathbf{H}_{i,n} + \mathbf{V}_n \tag{2.3}$$

where $\mathbf{X}_{i,n} = \text{diag}(X_{i,n,0}, ..., X_{i,n,K-1})$ is $K \times K$ diagonal data matrix and $\mathbf{H}_{i,n}$ is the $K \times 1$ channel vector (2.2) corresponding to the ith user in the nth symbol. The noise vector $\mathbf{V}_n$ is distributed as $\mathcal{CN}(0, \sigma^2 I_K)$.

2.2 Basic OFDMA Symbol Structure in IEEE 802.16e

The WirelessMAN-OFDMA PHY, is designed for nonline-of-sight (NLOS) operation in frequency bands below 11 GHz. For licensed bands, channel bandwidths allowed shall be limited

to the regulatory provisioned bandwidth divided by any power of 2 no less than 1.0 MHz. The material in this section is mainly taken from [8] and [9].

2.2.1 OFDMA Basic Terms

We introduce some basic terms in OFDMA PHY. These definitions would help us understand the concepts of subcarrier allocation and transmission of IEEE 802.16e OFDMA.

- Slot: A slot in the OFDMA PHY is a two-dimensional entity spanning both a time and a subchannel dimension. It is the minimum possible data allocation unit. For downlink (DL) PUSC (Partial Usage of SubChannels), one slot is one subchannel by two OFDMA symbols. For uplink (UL), one slot is one subchannel by three OFDMA symbols.

- Data region: In OFDMA, a data region is a two-dimensional allocation of a group of contiguous subchannels in a group of contiguous OFDMA symbols. All the allocations refer to logical subchannels. A two dimensional allocation may be visualized as a rectangle, such as the 4×3 rectangle shown in Fig. 2.4.

- Segment: A segment is a subdivision of the set of available OFDMA subchannels (that may include all available subchannels). One segment is used for deploying a single instance of the MAC.

2.2.2 Frequency Domain Description

An OFDMA symbol (see Fig. 2.5) is made up of subcarriers, the number of which determines the FFT size used. There are several subcarrier types:

- Data subcarriers: for data transmission.

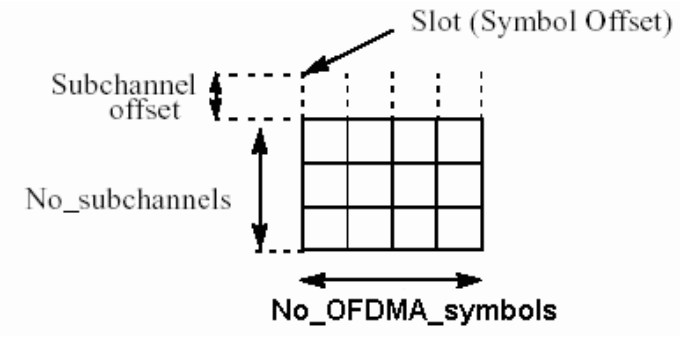


Figure 2.4: Example of the data region which defines the OFDMA allocation (from [8]).

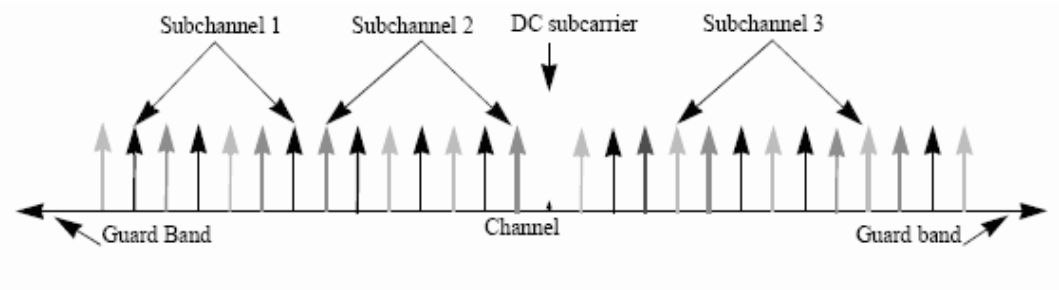


Figure 2.5: OFDMA frequency description (from [8]).

- Pilot subcarriers: for various estimation purposes.
- Null subcarriers: no transmission at all, for guard bands and DC subcarrier.

2.2.3 Primitive Parameters

Four primitive parameters characterize the OFDMA symbols:

- BW: the nominal channel bandwidth.
- N_{used}: number of used subcarriers (which includes the DC subcarrier).
- n: sampling factor. This parameter, in conjunction with BW and N_{used}, determines the subcarrier spacing and the useful symbol time. Its value is set as follows: For channel

bandwidths that are a multiple of 1.75 MHz $n = 8/7$, else for channel bandwidths that are a multiple of any of 1.25, 1.5, 2 or 2.75 MHz $n = 28/25$, else for channel bandwidths not otherwise specified $n = 8/7$.

- G: This is the ratio of CP time to "useful" time, i.e., T_{cp}/T_s. The following values shall be supported: 1/32, 1/16, 1/8, and 1/4.

2.2.4 Derived Parameters

The following parameters are defined in terms of the primitive parameters.

- N_{FFT}: smallest power of two greater than N_{used}.
- Sampling frequency: $F_s = \lfloor n \cdot \mathrm{BW}/8000 \rfloor \times 8000$.
- Subcarrier spacing: $\triangle f = F_s/N_{FFT}$.
- Useful symbol time: $T_b = 1/\triangle f$.
- CP time: $T_g = G \times T_b$.
- OFDMA symbol time: $T_s = T_b + T_g$.
- Sampling time: T_b/N_{FFT}.

2.2.5 Frame Structure

When implementing a time-division duplex (TDD) system, the frame structure is built from base station (BS) and subscriber station (SS) transmissions. Each frame in the DL transmission begins with a preamble followed by a DL transmission period and an UL transmission period. In each frame, the TTG and RTG shall be inserted between the downlink and uplink periods and at the end of each frame, respectively, to allow the BS to turn around. Fig. 2.6 shows an example of an OFDMA frame with only mandatory zone in TDD mode.

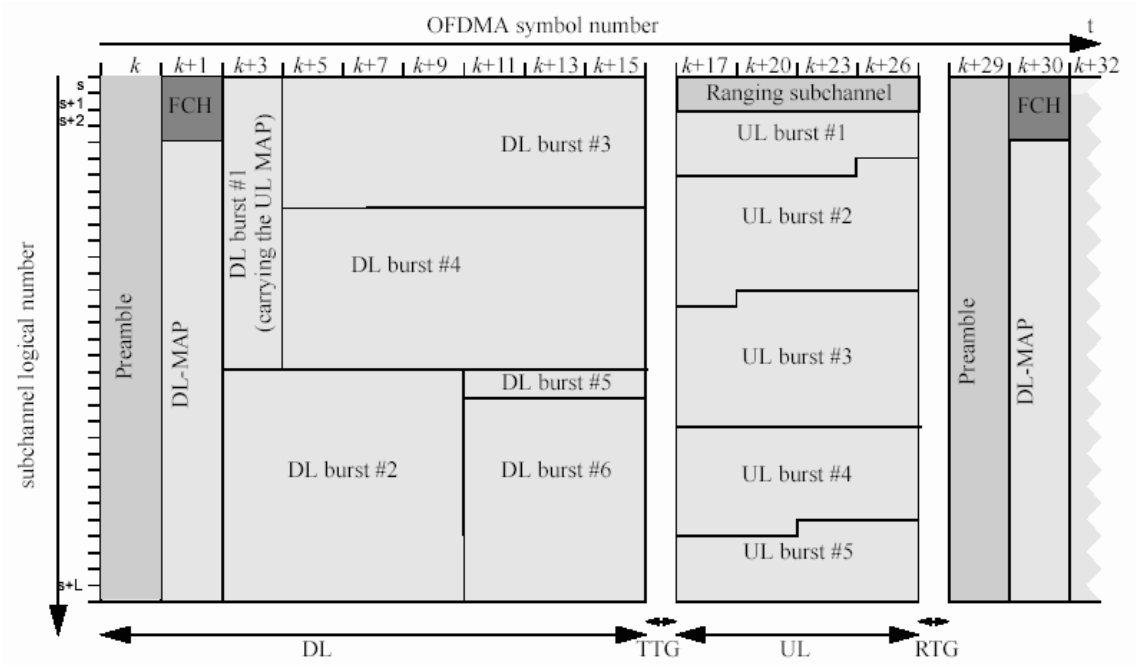


Figure 2.6: Example of an OFDMA frame (with only mandatory zone) in TDD mode (from [9]).

2.3 Downlink Transmission in IEEE 802.16e OFDMA

This section briefly introduces the specifications of IEEE 802.16e OFDMA PUSC downlink transmission. The material is mainly taken from [8] and [9].

2.3.1 Data Mapping Rules

The downlink data mapping rules are as follows:

1. Segment the data after the modulation block into blocks sized to fit into one OFDMA slot.

2. Each slot shall span one subchannel in the subchannel axis and one or more OFDMA symbols in the time axis, as per the slot definition mentioned before. Map the slots

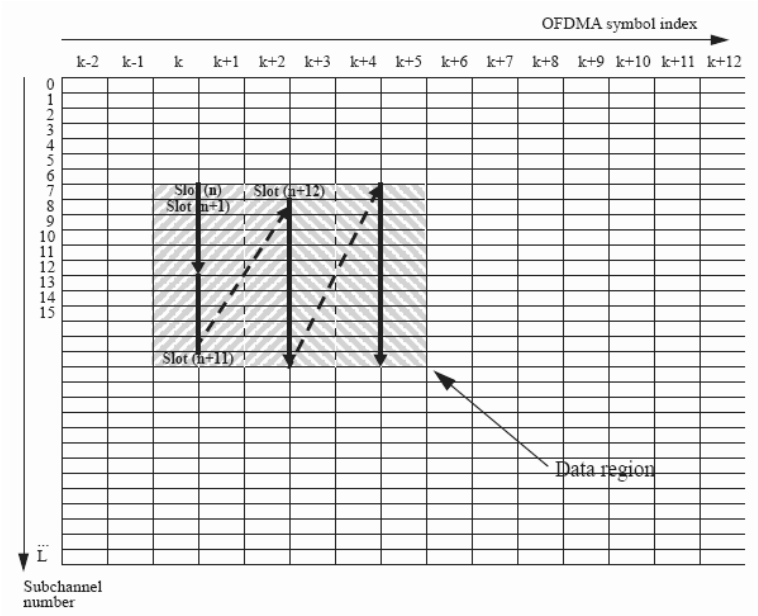


Figure 2.7: Example of mapping OFDMA slots to subchannels and symbols in the downlink in PUSC mode (from [9]).

such that the lowest numbered slot occupies the lowest numbered subchannel in the lowest numbered OFDMA symbol.

3. Continue the mapping such that the OFDMA subchannel index is increased. When the edge of the data region is reached, continue the mapping from the lowest numbered OFDMA subchannel in the next available symbol.

Figure 2.7 illustrates the order of OFDMA slots mapping to subchannels and OFDMA symbols.

2.3.2 Preamble Structure and Modulation

The first symbol of the downlink transmission is the preamble. Fig. 2.8 shows a downlink transmission period. There are three types of preamble carrier-sets, which are defined by allocation of different subcarriers. The subcarriers are modulated using a boosted BPSK

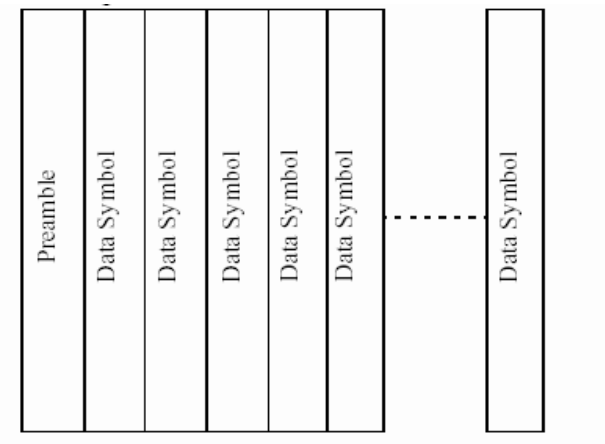


Figure 2.8: Downlink transmission basic structure (from [8]).

modulation with a specific pseudo-noise (PN) code. The PN series modulating the pilots in the preamble can be found in [8, pp. 553–562].

The preamble carrier-sets are defined as

$$PreambleCarrierSet_n = n + 3 \cdot k, \tag{2.4}$$

where:

- $PreambleCarrierSet_n$ specifies all subcarriers allocated to the specific preamble,
- n is the number of the preamble carrier-set indexed 0, 1, 2, and
- k is a running index 0,. . . ,567.

Each segment uses one type of preamble out of the three sets in the following manner: For the preamble symbol, there will be 172 guard band subcarriers on the left side and the right side of the spectrum. Segment i uses preamble carrier-set i, where $i = 0, 1, 2$. The DC subcarrier will not be modulated at all and the appropriate PN will be discarded. Therefore, DC subcarrier shall always be zeroed.

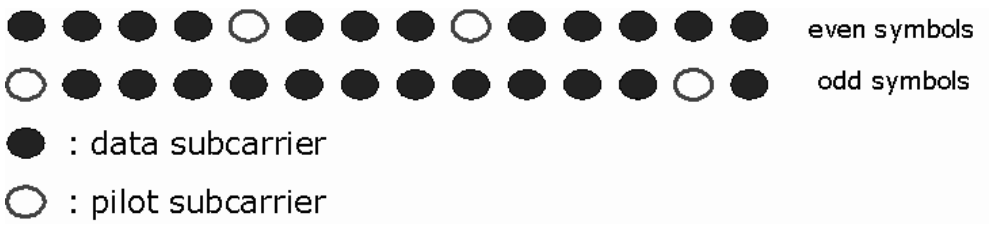


Figure 2.9: Cluster structure (from [9]).

The pilots in downlink preamble shall be modulated as

$$\begin{aligned} \Re\{PreamblePilotsModulated\} &= 4 \cdot \sqrt{2} \cdot (\frac{1}{2} - w_k), \\ \Im\{PreamblePilotsModulated\} &= 0. \end{aligned} \tag{2.5}$$

2.3.3 Subcarrier Allocations

The OFDMA symbol structure is constructed using pilot, data and zero subcarriers. The symbol is first divided into basic clusters and zero carriers are allocated. The pilot tones are allocated first; what remains are data subcarriers, which are divided into subchannels that are used exclusively for data. Pilots and data carriers are allocated within each cluster.

Figure 2.9 shows the cluster structure with subcarriers from left to right in order of increasing subcarrier index. For the purpose of determining PUSC pilot location, odd and even symbols are counted from the beginning of the current zone. The first symbol in the zone is even. The preamble shall not be counted as part of the first zone. Table 2.1 summarizes the parameters of the OFDMA PUSC symbol structure.

The allocation of subcarriers to subchannels is performed using the following procedure:

1. Divide the subcarriers into a number ($N_{clusters}$) of physical clusters containing 14 adjacent subcarriers each (starting from carrier 0).

Table 2.1: OFDMA Downlink Subcarrier Allocation Under PUSC [8], [9]

Parameter	**Value**	**Comments**
Number of DC subcarriers	1	Index 1024 (counting from 0)
Number of guard subcarriers, left	184	
Number of guard subcarriers, right	183	
Number of used subcarriers (N_{used})	1681	Number of all subcarriers used within a symbol, including all possible allocated pilots and the DC carrier
Number of subcarriers per cluster	14	
Number of clusters	120	
Renumbering sequence	1	Used to renumber clusters before allocation to subchannels: 6,108,37,81,31,100,42,116,32,107,30,93,54,78, 10,75,50,111,58,106,23,105,16,117,39,95,7, 115,25,119,53,71,22,98,28,79,17,63,27,72,29, 86,5,101,49,104,9,68,1,73,36,74,43,62,20,84, 52,64,34,60,66,48,97,21,91,40,102,56,92,47, 90,33,114,18,70,15,110,51,118,46,83,45,76,57, 99,35,67,55,85,59,113,11,82,38,88,19,77,3,87, 12,89,26,65,41,109,44,69,8,61,13,96,14,103,2, 80,24,112,4,94,0
Number of data subcarriers in each symbol per subchannel	24	
Number of subchannels	60	
Basic permutation sequence 12 (for 12 subchannels)	12	6,9,4,8,10,11,5,2,7,3,1,0
Basic permutation sequence 8 (for 8 subchannels)	8	7,4,0,2,1,5,3,6

2. Renumber the physical clusters into logical clusters using the following formula:

$$LogicalCluster = \begin{cases} RenumberingSequence(PhysicalCluster), & \text{first DL zone,} \\ RenumberingSequence\big((PhysicalCluster + 13 \cdot DL\_PermBase) \bmod N_{clusters}\big), & \text{otherwise.} \end{cases}$$

3. Dividing the clusters into six major groups. Group 0 includes clusters 0–23, group 1 clusters 24–39, group 2 clusters 40–63, group 3 clusters 64–79, group 4 clusters 80–103 and group 5 clusters 104–119. These groups may be allocated to segments. If a segment is being used, then at least one group shall be allocated to it. (By default group 0 is allocated to segment 0, group 2 to segment 1, and group 4 to segment 2) .

4. Allocate subcarriers to subchannels in each major group separately for each OFDMA symbol by first allocating the pilot subcarriers within each cluster and then taking all remaining data subcarriers within the symbol. The exact partitioning into subchannels is according to the equation below, called a permutation formula:

$$subcarrier(k,s) = N_{subchannels} \cdot n_k + \big\{p_s[n_k \ mod \ N_{subchannels}] + DL\_PermBase\big\} \bmod N_{subchannels}$$

where:

- $subcarrier(k,s)$ is the subcarrier index of subcarrier k in subchannel s,
- s is the index number of a subchannel, from the set $[0..N_{subchannels} - 1]$,
- $n_k = (k + 13 \cdot s) \bmod N_{subcarriers}$, where k is the subcarrier-in-subchannel index from the set $[0..N_{subcarriers} - 1]$,
- $N_{subchannels}$ is the number of subchannels (for PUSC use number of subchannels in the currently partitioned group),

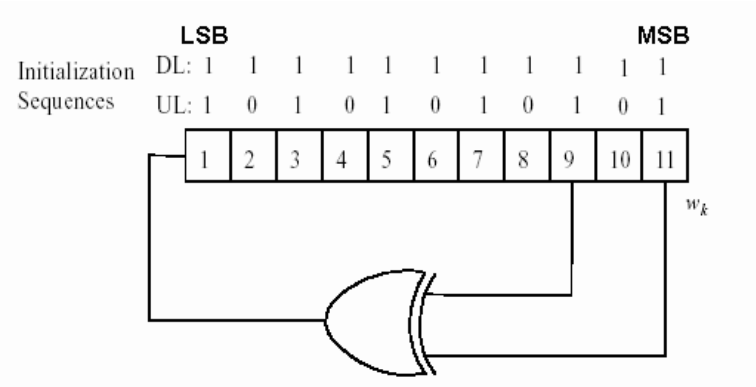


Figure 2.10: PRBS generator for pilot modulation (from [8] and [9]).

- $p_s[j]$ is the series obtained by rotating basic permutation sequence cyclically to the left s times,
- $N_{subcarriers}$ is the number of data subcarriers allocated to a subchannel in each OFDMA symbol, and
- $DL\_PermBase$ is an integer from 0 to 31.

2.3.4 Pilot Modulation

Pilot subcarriers shall be inserted into each data burst in order to constitute the symbol. The PRBS (pseudo-random binary sequence) generator depicted in Fig. 2.10 shall be used to produce a sequence w_k.

Each pilot shall be transmitted with a boosting of 2.5 dB over the average non-boosted power of each data tone. The pilot subcarriers shall be modulated according to

$$\Re\{c_k\} = \frac{8}{3}\left(\frac{1}{2} - w_k\right), \quad \Im\{c_k\} = 0. \tag{2.6}$$

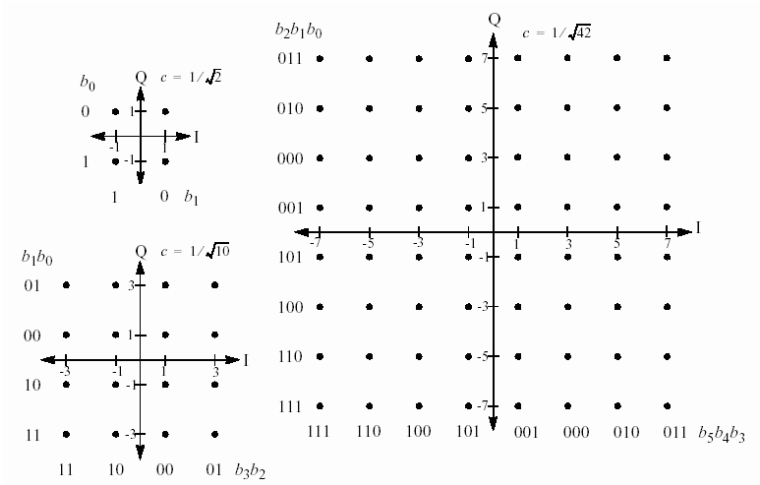


Figure 2.11: QPSK, 16-QAM, and 64-QAM constellations (from [8]).

2.3.5 Data Modulation

As shown in Fig. 2.11, for downlink transmission, gray-mapped QPSK and Gray-mapped 16QAM shall be supported, whereas the support of 64QAM (also Gray-mapped) is optional.

Chapter 3

Introduction to IEEE 802.16m OFDMA

3.1 Basic OFDMA Symbol Structure in IEEE 802.16m [10]

The Advanced Air Interface defined by IEEE 802.16m is designed for nonline-of-sight (NLOS) operation in the licensed frequency bands below 6 GHz. The Advanced Air Interface supports time-division-duplexing (TDD) and frequency-division-duplexing (FDD) duplex modes, including half FDD (H-FDD) mobile station (MS) operation. Unless otherwise specified, the frame structure attributes and baseband processing are common for all duplex modes.

The Advanced Air Interface uses OFDMA as the multiple access scheme in the downlink and uplink. The material of this is taken from [10].

3.1.1 OFDMA Basic Terms

We introduce some basic terms appeared in the OFDMA physical layer (PHY) of IEEE 802.16m. These definitions help us understand the concepts of subcarrier allocation and transmission in IEEE 802.16m OFDMA.

Table 3.1: PRU Structure for Different Types of Subframes

Subframe Type	Number of Subcarriers	Number of Symbols
Type-1	18	6
Type-2	18	7
Type-3	18	5

- Physical and logical resource unit: A physical resource unit (PRU) is the basic physical unit for resource allocation. It comprises P_{sc} consecutive subcarriers by N_{sym} consecutive OFDMA symbols. P_{sc} is 18 subcarriers and N_{sym} is 6 OFDMA symbols for type-1 subframes, 7 OFDM symbols for type-2 subframes, and 5 OFDMA symbols for type-3 subframes. Table 1.1 illustrates the PRU's sizes for different subframe types. A logical resource unit (LRU) is the basic logical unit for distributed and localized resource allocations. An LRU is $P_{sc} \cdot N_{sym}$ subcarriers for type-1, type-2, and type-3 subframes. The LRU includes the pilots that are used in a PRU. The effective number of subcarriers in an LRU depends on the number of allocated pilots.

- Distributed resource unit: A distributed resource unit (DRU) contains a group of subcarriers which are spread across the distributed resource allocations within a frequency partition. The size of DRU equals the size of PRU, i.e., P_{sc} subcarriers by N_{sym} OFDMA symbols.

- Contiguous resource unit: The localized resource unit, also known as contiguous resource unit (CRU), contains a group of subcarriers which are contiguous across the localized resource allocations. The size of CRU equals the size of PRU, i.e., P_{sc} subcarriers by N_{sym} OFDMA symbols.

3.1.2 Frequency Domain Description

The frequency domain description includes the basic structure of an OFDMA symbol. An OFDMA symbol is made up of subcarriers, the number of which determines the DFT size used. There are several subcarrier types:

- Data subcarriers: for data transmission.
- Pilot subcarriers: for various estimation purposes.
- Null subcarriers: no transmission at all, for guard bands and DC subcarrier.

The purpose of the guard bands is to help enable proper bandlimiting.

3.1.3 Primitive Parameters

Four primitive parameters characterize the OFDMA symbols:

- BW: the nominal channel bandwidth.
- N_{used}: number of used subcarriers (which includes the DC subcarrier).
- n: sampling factor. This parameter, in conjunction with BW and N_{used}, determines the subcarrier spacing and the useful symbol time. This value is given in Figs. 3.1 and 3.2 for each nominal bandwidth.
- G: This is the ratio of CP time to "useful" time, i.e., T_{cp}/T_s. The following values shall be supported: 1/16, 1/8, and 1/4.

3.1.4 Derived Parameters

The following parameters are defined in terms of the primitive parameters.

- N_{FFT}: smallest power of two greater than N_{used}.
- Sampling frequency: $F_s = \lfloor n \cdot BW/8000 \rfloor \times 8000$.
- Subcarrier spacing: $\triangle f = F_s/N_{FFT}$.
- Useful symbol time: $T_b = 1/\triangle f$.
- CP time: $T_g = G \times T_b$.
- OFDMA symbol time: $T_s = T_b + T_g$.
- Sampling time: T_b/N_{FFT}.

3.1.5 Frame Structure

The advanced air interface basic frame structure is illustrated in Fig. 3.3. Each 20 ms superframe is divided into four 5-ms radio frames. When using the same OFDMA parameters as in Figs. 3.1 and 3.2 with channel bandwidth of 5, 10, or 20 MHz, each 5-ms radio frame further consists of eight subframes for $G = 1/8$ and $1/16$. With channel bandwidth of 8.75 or 7 MHz, each 5-ms radio frame further consists of seven and six subframes, respectively for $G = 1/8$ and $1/16$. In the case of $G = 1/4$, the number of subframes per frame is one less than that of other CP lengths for each bandwidth case. A subframe shall be assigned for either downlink (DL) or uplink (UL) transmission. There are four types of subframes:

- Type-1 subframe consists of six OFDMA symbols.
- Type-2 subframe consists of seven OFDMA symbols.
- Type-3 subframe consists of five OFDMA symbols.

The nominal channel bandwidth, BW (MHz)			5	7	8.75	10	20
Sampling factor, n			28/25	8/7	8/7	28/25	28/25
Sampling frequency, F_s (MHz)			5.6	8	10	11.2	22.4
FFT size, N_{FFT}			512	1024	1024	1024	2048
Subcarrier spacing, Δf (kHz)			10.94	7.81	9.77	10.94	10.94
Useful symbol time, T_b (μs)			91.4	128	102.4	91.4	91.4
CP ratio, $G = 1/8$	OFDMA symbol time, T_s (μs)		102.857	144	115.2	102.857	102.857
	FDD	Number of OFDMA symbols per 5ms frame	48	34	43	48	48
		Idle time (μs)	62.857	104	46.40	62.857	62.857
	TDD	Number of OFDMA symbols per 5ms frame	47	33	42	47	47
		TTG + RTG (μs)	165.714	248	161.6	165.714	165.714
CP ratio, $G = 1/16$	OFDMA symbol time, T_s (μs)		97.143	136	108.8	97.143	97.143
	FDD	Number of OFDMA symbols per 5ms frame	51	36	45	51	51
		Idle time (μs)	45.71	104	104	45.71	45.71
	TDD	Number of OFDMA symbols per 5ms frame	50	35	44	50	50
		TTG + RTG (μs)	142.853	240	212.8	142.853	142.853

Figure 3.1: OFDMA parameters (Table 647 in [10]).

- Type-4 subframe consists of nine OFDMA symbols. This type shall be applied only to UL subframe for the 8.75 MHz channel bandwidth when supporting the WirelessMAN-OFDMA frames.

The basic frame structure is applied to FDD and TDD duplexing schemes, including H-FDD MS operation. The number of switching points in each radio frame in TDD systems shall be two, where a switching point is defined as a change of directionality, i.e., from DL to UL or from UL to DL.

CP ratio, $G = 1/4$	OFDMA symbol time, T_s (μs)		114.286	160	128	114.286	114.286
	FDD	Number of OFDMA symbols per 5ms frame	43	31	39	43	43
		Idle time (μs)	85.694	40	8	85.694	85.694
	TDD	Number of OFDMA symbols per 5ms frame	42	30	38	42	42
		TTG + RTG (μs)	199.98	200	136	199.98	199.98
Number of Guard Sub-Carriers		Left	40	80	80	80	160
		Right	39	79	79	79	159
Number of Used Sub-Carriers			433	865	865	865	1729
Number of Physical Resource Unit (18x6) in a type-1 sub-frame.			24	48	48	48	96

Figure 3.2: More OFDMA parameters (Table 647 in [10]).

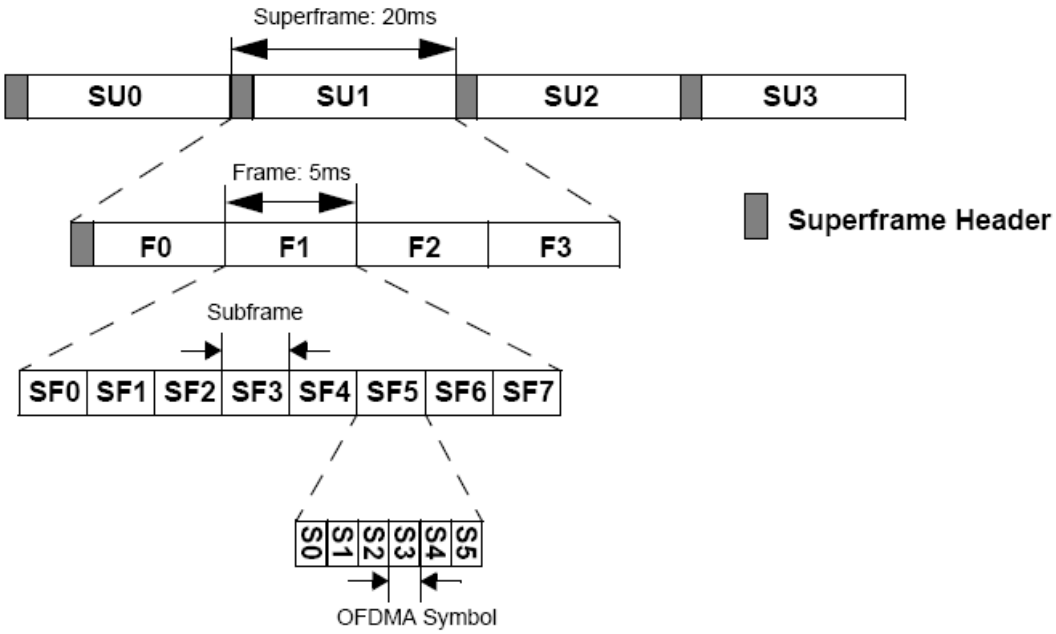


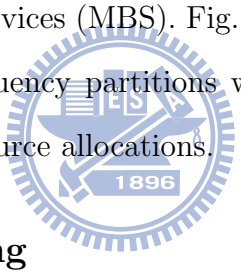
Figure 3.3: Basic frame structure for 5, 10 and 20 MHz channel bandwidths (Fig. 430 in [10]).

A data burst shall occupy either one subframe (i.e., the default transmission time interval [TTI] transmission) or contiguous multiple subframes (i.e., the long TTI transmission). The

long TTI in FDD shall be 4 subframes for both DL and UL. The long TTI in TDD shall be the whole DL (UL) subframes for DL (UL) in a frame. Every superframe shall contain a superframe header (SFH). The SFH shall be located in the first DL subframe of the superframe and shall include broadcast channels.

3.2 Downlink Transmission in IEEE 802.16m OFDMA[10]

Again this section is mainly taken from [10]. Each DL subframe is divided into 4 or fewer frequency partitions; each partition consists of a set of physical resource units across the total number of OFDMA symbols available in the subframe. Each frequency partition can include contiguous (localized) and/or non-contiguous (distributed) physical resource units. Each frequency partition can be used for different purposes such as fractional frequency reuse (FFR) or multicast and broadcast services (MBS). Fig. 3.4 illustrates the downlink physical structure in an example of two frequency partitions with frequency partition 2 including both contiguous and distributed resource allocations.



3.2.1 Subband Partitioning

The PRUs are first subdivided into subbands and minibands where a subband comprises N_1 adjacent PRUs and a miniband comprises N_2 adjacent PRUs, where $N_1 = 4$ and $N_2 = 1$. Subbands are suitable for frequency selective allocations as they provide a contiguous allocation of PRUs in frequency. Minibands are suitable for frequency diverse allocation and are permuted in frequency.

The number of subbands reserved is denoted by K_{SB}. The number of PRUs allocated to subbands is denoted by L_{SB}, where $L_{SB} = N_1 \cdot K_{SB}$, depending on system bandwidth. A 4 or 3-bit field called Subband Allocation Count (SAC) determines the value of K_{SB}. The remainder of the PRUs are allocated to minibands. The number of minibands in an

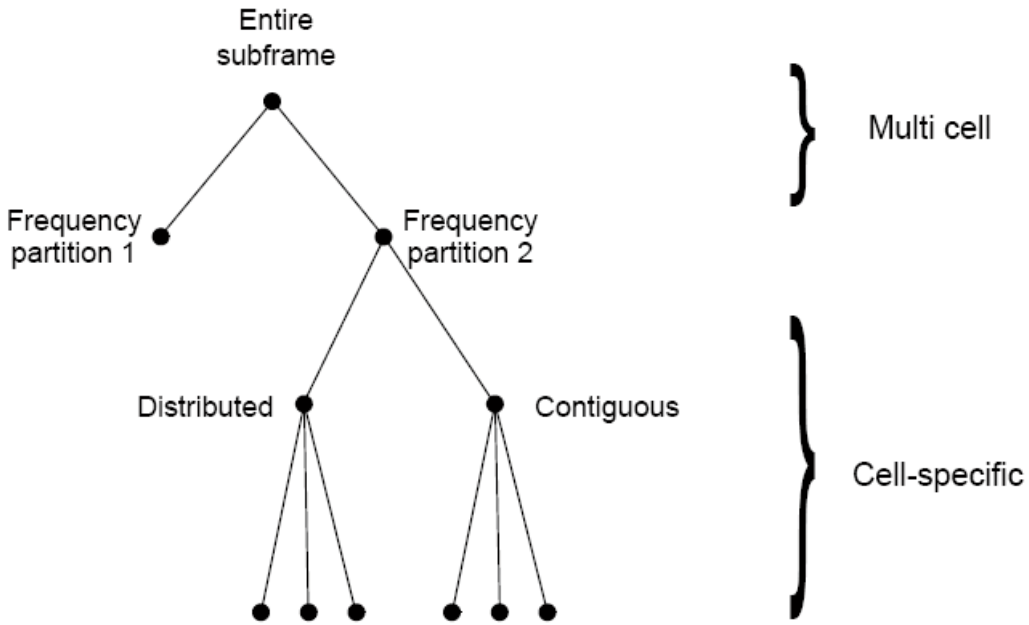


Figure 3.4: Example of downlink physical structure (Fig. 449 in [10]).

allocation is denoted by K_{MB}. The number of PRUs allocated to minibands is denoted by L_{MB}, where $L_{MB} = N_2 \cdot K_{MB}$. The total number of PRUs is denoted $NPRU$ where $NPRU = L_{SB} + L_{MB}$. The maximum number of subbands that can be formed is denoted N_{sub} where $N_{sub} = NPRU/N_1$.

Figs. 3.5 and 3.6 show the mapping between SAC and K_{SB} for the 10 and 20 MHz bands and the 5 MHz band, respectively.

PRUs are partitioned and reordered into two groups called subband PRUs and miniband PRUs and denoted PRU_{SB} and PRU_{MB}, respectively. The set PRU_{SB} is numbered from 0 to $L_{SB} - 1$,and the set PRU_{MB} is numbered from 0 to $L_{MB} - 1$. Equation 3.1 defines the mapping of PRUs to PRU_{SB}, and 3.3 the mapping of PRUs to PRU_{MB}. Fig. 3.7 illustrates the PRU to PRU_{SB} and PRU_{MB} mappings for a 5 MHz bandwidth with K_{SB} equal to 3.

$$PRU_{SB}[j] = PRU[i], \quad j = 0, 1, ..., L_{SB} - 1 \tag{3.1}$$

SAC	# of subbands allocated (K_{SB})	SAC	# of subbands allocated (K_{SB})
0	0	8	10
1	1	9	12
2	2	10	14
3	3	11	16
4	4	12	18
5	5	13	20
6	6	14	22
7	8	15	24

Figure 3.5: Mapping between SAC and K_{SB} for 10 or 20 MHz band (Table 649 in [10]).

where

$$i = N_1 \cdot \{\{\lceil \frac{N_{sub}}{K_{SB}} \rceil \cdot \lfloor \frac{j}{N_1} \rfloor + \lfloor \lfloor \frac{j}{N_1} \rfloor \cdot \frac{GCD(N_{sub}, \lceil \frac{N_{sub}}{K_{SB}} \rceil)}{N_{sub}} \rfloor \} \mod N_{sub}\} + j \cdot N_1 \quad (3.2)$$

with $GCD(x, y)$ being the greatest common divisor of x and y.

$$PRU_{MB}[k] = PRU[i], \quad k = 0, 1, ..., L_{MB} - 1 \quad (3.3)$$

where

$$i = \begin{cases} N_1 \cdot \{\lceil \frac{N_{sub}}{K_{SB}} \cdot \rceil \cdot \lfloor \frac{k+L_{SB}}{N_1} \rfloor + \lfloor \lfloor \frac{k+L_{SB}}{N_1} \rfloor \cdot \frac{GCD(N_{sub}, \lceil \frac{N_{sub}}{K_{SB}} \rceil)}{N_{sub}} \rfloor \} \mod N_{sub} \\ +(k + L_{SB}) \mod N_1, & K_{SB} > 0 \\ i = k, & K_{SB} = 0. \end{cases} \quad (3.4)$$

SAC	# of subbands allocated (K_{SB})	SAC	# of subbands allocated (K_{SB})
0	0	4	4
1	1	5	5
2	2	6	6
3	3	7	N.A

Figure 3.6: Mapping between SAC and K_{SB} for 5 MHz band (Table 650 in [10]).

3.2.2 Miniband Permutation

The miniband permutation maps the PRU_{MB}s to Permuted PRU_{MB}s ($PPRU_{MB}$s) to ensure that frequency diverse PRUs are allocated to each frequency partition. Fig. 3.8 shows an example. The following equation provides a mapping from PRU_{MB} to $PPRU_{MB}$:

$$PRU_{MB}[j] = PRU[i], \quad j = 0, 1, ..., L_{MB} - 1, \tag{3.5}$$

where

$$i = (q(j) \mod D) \cdot P + \lfloor \frac{q(j)}{D} \rfloor, \tag{3.6}$$

$$P = \min(K_{MB}, N_1/N_2), \tag{3.7}$$

$$r(j) = \max(j - ((K_{MB} \mod P) \cdot D), 0), \tag{3.8}$$

$$q(j) = j + \lfloor \frac{r(j)}{D-1} \rfloor, \tag{3.9}$$

$$D = \lfloor \frac{K_{MB}}{P} + 1 \rfloor. \tag{3.10}$$

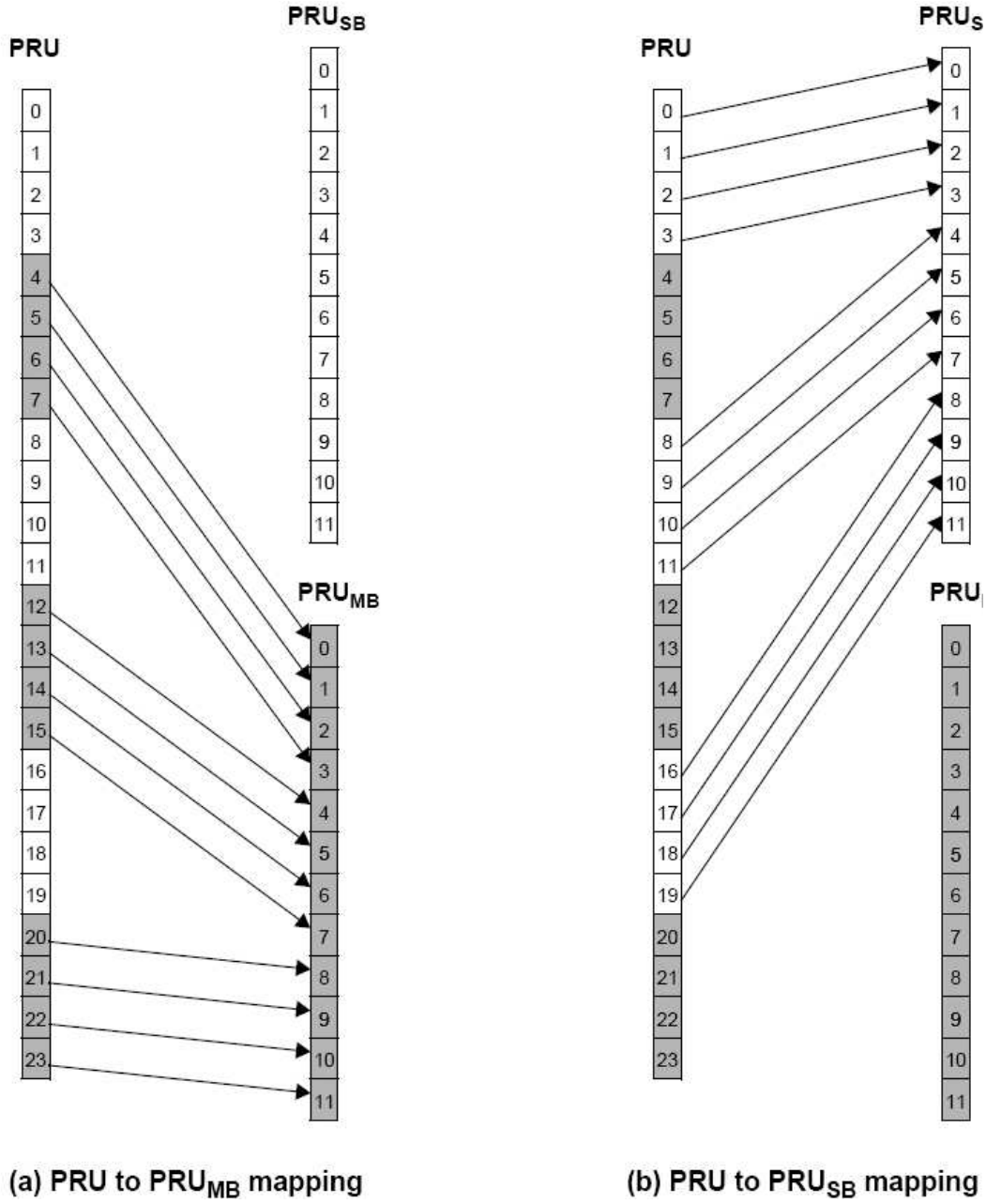


Figure 3.7: PRU to PRU_{SB} and PRU_{MB} mapping for BW = 5 MHz, and K_{SB}=3 (Fig. 450 in [10]).

3.2.3 Frequency Partitioning

The PRU_{SB} and $PPRU_{MB}$ are allocated to one or more frequency partitions. The frequency partition configuration is transmitted in the SFH in a 4 or 3-bit called the Downlink

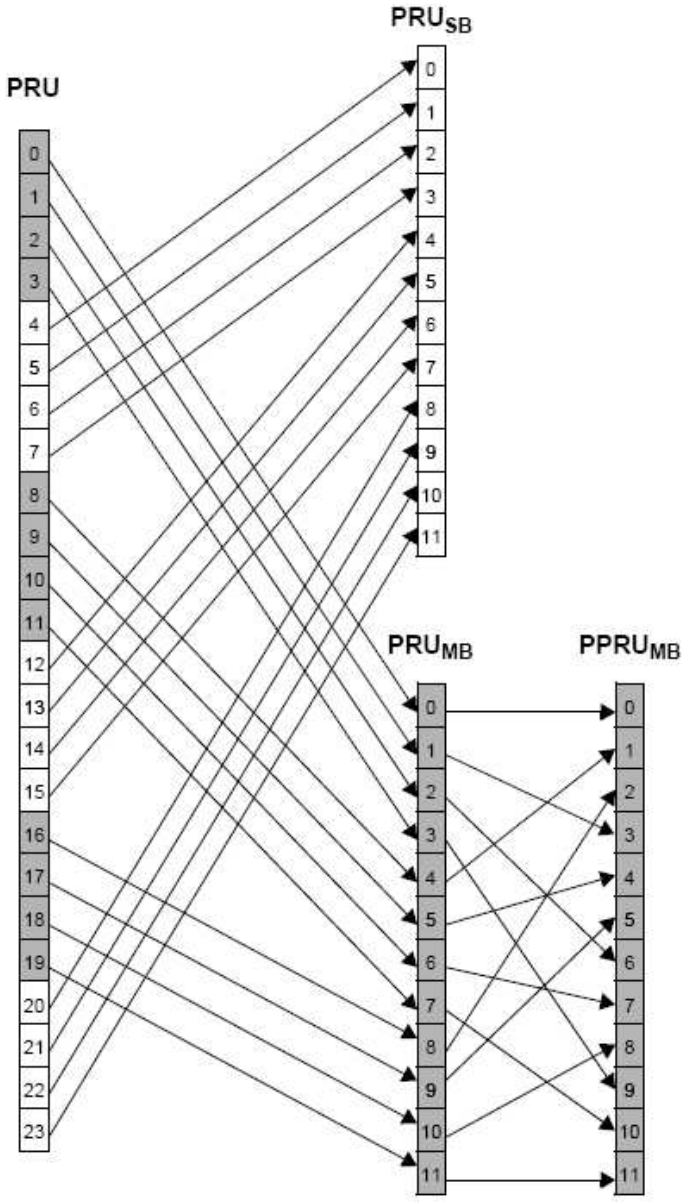


Figure 3.8: Mapping from PRUs to PRU_{SB} and $PPRU_{MB}$ for BW = 5 MHz and $K_{SB} = 3$ (Fig. 451 in [10]).

Frequency Partition Configuration (DFPC) depending on system bandwidth. Frequency Partition Count (FPCT) defines the number of frequency partitions. Frequency Partition Size (FPSi) defines the number of PRUs allocated to FP_i. FPCT and FPSi are determined

DFPC	Freq. Partitioning (FP_0:FP_1:FP_2:FP_3)	FPCT	FPS_0	FPS_i (i>0)
0	1 : 0 : 0 : 0	1	N_{PRU}	0
1	0 : 1 : 1 : 1	3	0	N_{PRU} * 1/3
2	1 : 1 : 1 : 1	4	N_{PRU} * 1/4	N_{PRU} * 1/4
3	3 : 1 : 1 : 1	4	N_{PRU} * 1/2	N_{PRU} * 1/6
4	5 : 1 : 1 : 1	4	N_{PRU} * 5/8	N_{PRU} * 1/8
5	9 : 1 : 1 : 1	4	N_{PRU} * 9/12	N_{PRU} * 1/12
6	9 : 5 : 5 : 5	4	N_{PRU} * 3/8	N_{PRU} * 5/12
7-15	*Reserved*			

Figure 3.9: Mapping between DFPC and frequency partitioning for 10 or 20 MHz band (Table 651 in [10]).

FPC	Freq. Partitioning (FP0:FP1:FP2:FP3)	FPCT	FPS_0	FPS_i (i>0)
0	1 : 0 : 0 : 0	1	N_{PRU}	0
1	0 : 1 : 1 : 1	3	0	N_{PRU} * 1/3
2	1 : 1 : 1 : 1	4	N_{PRU} * 1/4	N_{PRU} * 1/4
3	3 : 1 : 1 : 1	4	N_{PRU} * 1/2	N_{PRU} * 1/6
4-7	*Reserved*			

Figure 3.10: Mapping between DFPC and frequency partitioning for 5 MHz band (Table 652 in [10]).

from FPC as shown in Figs. 3.9 and 3.10. A 3, 2, or 1-bit parameter called the Downlink Frequency Partition Subband Count (DFPSC) defines the number of subbands allocated to FP_i, $i > 0$. Fig. 3.11 continues the examples in Fig. 3.7 and 3.8 and shows how PRU_{SB} and $PPRU_{MB}$ can be mapped to frequency partitions.

The number of subbands in ith frequency partition is denoted by K_{SB,FP_i}. The number of minibands is denoted by K_{MB,FP_i}, which is determined by FPS and FPSC fields. The

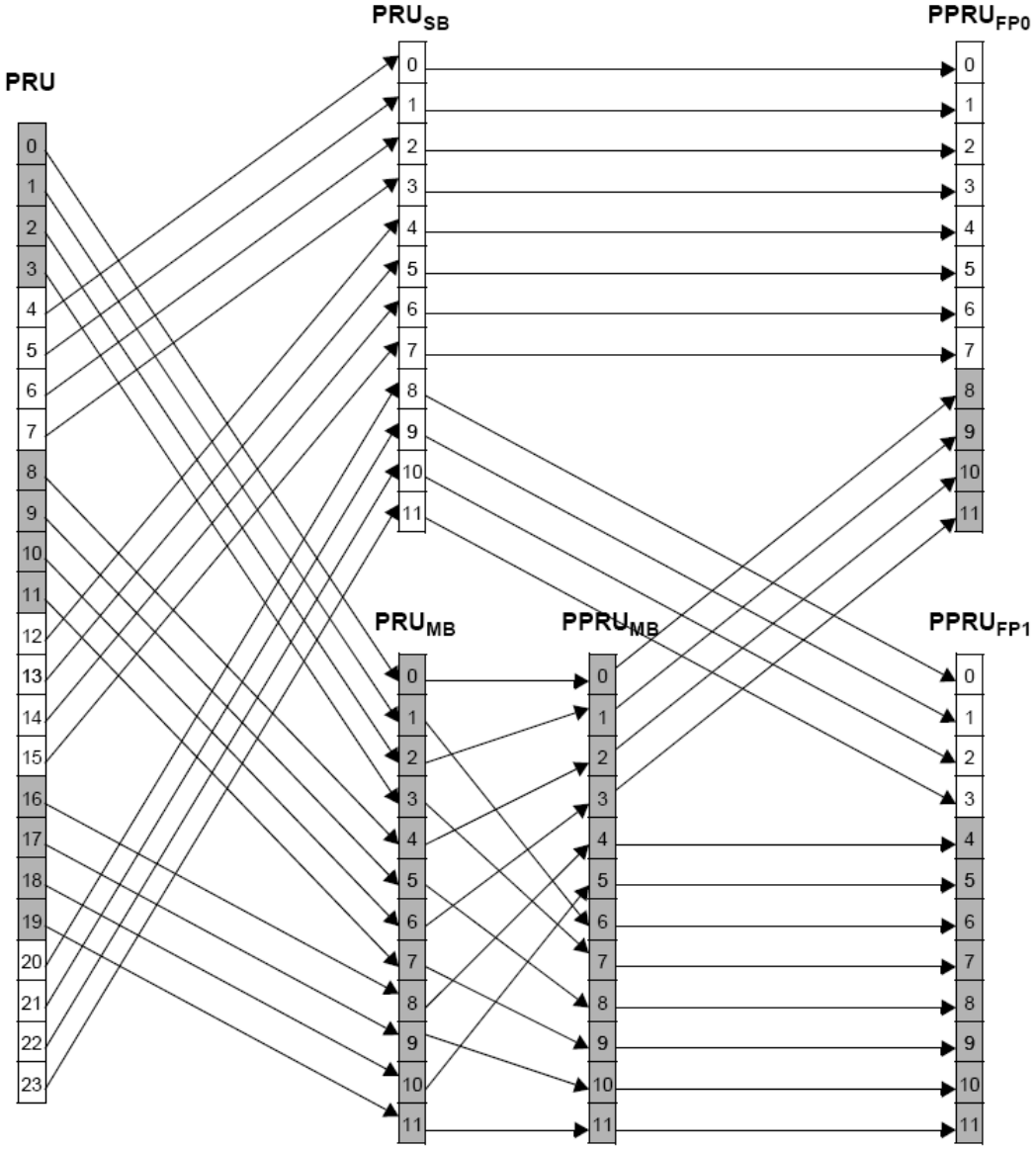


Figure 3.11: Frequency partitioning for BW = 5 MHz, $K_{SB} = 3$, FPCT = 2, FPS = 12, and FPSC = 1 (Fig. 452 in [10]).

number of subband PRUs in each frequency partition is denoted by $L_{SB,FPi}$, which is given by $L_{SB,FP_i} = N_1 \cdot K_{SB,FPi}$. The number of miniband PRUs in each frequency partition is denoted by $L_{MB,FPi}$, which is given by $L_{MB,FP_i} = N_2 \cdot K_{MB,FPi}$.

$$K_{SB,FPi} = \begin{cases} K_{SB}, & i = 0, \\ FPSC, & i > 0, \end{cases} \tag{3.11}$$

$$K_{MB,FPi} = (FPS_i - K_{SB,FPi} \cdot N_1)/N_2, \quad 0 \leq i < FPCT. \tag{3.12}$$

The mapping of subband PRUs and miniband PRUs to the frequency partition is given by

$$PRU_{FPi}(j) = \begin{cases} PRU_{SB}(k_1), & 0 \leq j < L_{SB,FPi}, \\ PPRU_{MB}(k_2), & L_{SB,FPi} \leq j < (L_{SB,FPi} + L_{MB,FPi}), \end{cases} \tag{3.13}$$

where

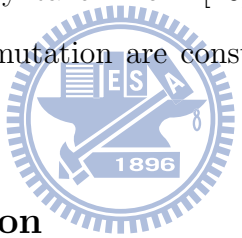
$$k_1 = \sum_{m=0}^{i-1} L_{SB,FPm} + j \tag{3.14}$$

and

$$k_2 = \sum_{m=0}^{i-1} L_{SB,FPm} + j - L_{SB,FPi}. \tag{3.15}$$

3.3 Cell-Specific Resource Mapping[10]

The content of this section is mainly taken from [10]. PRU_{FPi}s are mapped to LRUs. All further PRU and subcarrier permutation are constrained to the PRUs of a frequency partition.



3.3.1 CRU/DRU Allocation

The partition between CRUs and DRUs is done on a sector specific basis. A 4 or 3-bit Downlink subband-based CRU Allocation Size ($DCAS_{SBi}$) field is sent in the SFH for each allocated frequency partition. $DCAS_{SBi}$ indicates the number of allocated CRUs for partition FP_i in unit of subband size. A 5, 4 or 3-bit Downlink miniband-based CRU Allocation Size ($DCAS_{MB}$) is sent in the SFH only for partition FP_0 depending on system bandwidth, which indicates the number of allocated miniband-based CRUs for partition FP_0. The number of CRUs in each frequency partition is denoted $L_{CRU,FPi}$, where

$$L_{CRU,FPi} = \begin{cases} CAS_{SBi} \cdot N_1 + CAS_{MB} \cdot N_2, & i = 0, \\ CAS_{SBi} \cdot N_1, & 0 < i < FPCT. \end{cases} \tag{3.16}$$

The number of DRUs in each frequency partition is denoted $L_{DRU.FPi}$, where $L_{DRU,FPi} = FPS_i - L_{CRU,FPi}$ for $0 \leq i < FPCT$ and FPS_i is the number of PRUs allocated to FP_i. The mapping of PRU_{FPi} to CRU_{FPi} is given by

$$CRU_{FPi}[j] = \begin{cases} PRU_{FPi}[j], & 0 \leq j < CAS_{SBi} \cdot N_1, \quad 0 \leq i < FPCT, \\ PRU_{FPi}[k + CAS_{SBi} \cdot N_1], & CAS_{SBi} \cdot N_1 \leq j < L_{CRU,FPi}, \quad 0 \leq i < FPCT. \end{cases} \tag{3.17}$$

where $k = s[j - CAS_{SBi} \cdot N_1]$, with s[ ] being the CRU/DRU allocation sequence defined as

$$s[j] = \{PermSeq(j) + DL\_PermBase\} \mod \{FPS_i - CAS_{SBi} \cdot N_1\} \tag{3.18}$$

where $PermSeq()$ is the permutation sequence of length $(FPS_i - CAS_{SBi} \cdot N_1)$ and is determined by $SEED = IDcell \cdot 343 \mod 2$, $DL\_PermBase$ is an interger ranging from 0 to 31, which is set to preamble IDcell. The mapping of PRU_{FPi} to DRU_{FPi} is given by

$$DRU_{FPi}[j] = PRU_{FPi}[k + CAS_{SBi} \cdot N_1], \quad 0 \leq j < L_{DRU,FPi} \tag{3.19}$$

where $k = s^c[j]$, with $s^c[\,]$ being the sequence which is obtained by renumbering the remainders of the PRUs which are not allocated for CRU from 0 to $L_{DRU,FPi} - 1$.

3.3.2 Subcarrier Permutation

The subcarrier permutation defined for the DL distributed resource allocations within a frequency partition spreads the subcarriers of the DRU across the whole distributed resource allocations. The granularity of the subcarrier permutation is equal to a pair of subcarriers.

After mapping all pilots, the remainders of the used subcarriers are used to define the distributed LRUs. To allocate the LRUs, the remaining subcarriers are paired into contiguous tone-pairs. Each LRU consists of a group of tone-pairs.

Let $L_{SC,l}$ denote the number of data subcarriers in lth OFDMA symbol within a PRU, i.e., $L_{SC,l} = P_{SC} - N_l$, where n_l denotes the number of pilot subcarriers in the lth OFDMA

symbol within a PRU. Let $L_{SP,l}$ denote the number of data subcarrier-pairs in the lth OFDMA symbol within a PRU and is equal to $L_{SC,l}/2$. A permutation sequence PermSeq() is defined by (TBD) to perform the DL subcarrier permutation as follows. For each lth OFDMA symbol in the subframe:

1. Allocate the n_l pilots within each DRU as described in Section (TBD). Denote the data subcarriers of $DRU_{FPi}[j]$ in the lth OFDMA symbol as

$$SC^{FPi}_{DRU,j,l}[k], \quad 0 \le j < L_{DRU,FPi}, \qquad 0 \le k < L_{SC,l} \tag{3.20}$$

2. Renumber the $L_{DRU,FPi} \cdot L_{SC,l}$ data subcarriers of the DRUs in order, from 0 to $L_{DRU,FPi} \cdot L_{SC,l} - 1$. Group these contiguous and logically renumbered subcarriers into $L_{DRU,FPi} \cdot L_{SP,l}$ pairs and renumber them from 0 to $L_{DRU,FPi} \cdot L_{SP,l} - 1$. The renumbered subcarrier pairs in the lth OFDMA symbol are denoted as

$$RSP_{FPi,l}[u] = \{SC^{FPi}_{DRUj,l}[2v], SC^{FPi}_{DRUj,l}[2v+1]\}, \quad 0 \le u < L_{DRU,FPi}L_{SP,l}, \tag{3.21}$$

   where $\quad j = \lfloor u/L_{SP,l} \rfloor, \quad v = \{u\} \mod (L_{SP,l})$.

3. Apply the subcarrier permutation formula to map $RSP_{FPi,l}$ into the sth distributed LRU, s=0,1,...,$L_{DRU,FPi}$-1, where the subcarrier permutation formula is given by

$$SC^{FPi}_{LRUs,l}[m] = RSP_{FPi,l}[k], \quad 0 \le m \le L_{SP,l}, \tag{3.22}$$

   where

$$k = L_{DRU,FPi} \cdot f(m,s) + g(PermSeq(), s, m, l, t). \tag{3.23}$$

In the above,

1. $SC^{FPi}_{LRUs,l}[m]$ is the mth subcarrier pair in the lth OFDMA symbol in the sth distributed LRU of the tth subframe;

2. m is the subcarrier pair index, 0 to $L_{SP,l} - 1$;

3. l is the OFDMA symbol index, 0 to $N_{sym} - 1$;

4. s is the distributed LRU index, 0 to $L_{DRU,FPi} - 1$;

5. t is the subframe index with respect to the frame;

6. $PermSeq()$ is the permutation sequence of length $L_{DRU,FPi}$ and is determined by $SEED = \{IDcell * 1367\} \mod 2^{10}$; and

7. $g(PermSeq(), s, m, l, t)$ is a function with value from the set [0,$L_{DRU,FPi}$-1], which is defined according to

$$g(PermSeq(), s, m, l, t) = \{PermSeq[\{f(m, s) + s + l\} \mod \{L_{DRU,FPi}\}] + DL\_PermBase\} \mod L_{DRU,FPi} \quad (3.24)$$

where $DL\_PermBase$ is an integer ranging from 0 to 31(TBD), which is set to preamble IDcell, and $f(m, s) = (m + 13 * s) \mod L_{SP,l}$.

3.3.3 Random Sequence Generation

The permutation sequence generation algorithm with 10-bit SEED $(S_{n-10}, S_{n-9}, ..., S_{n-1})$ shall generate a permutation sequence of size M according to the following process:

- Initialization

  1. Initialize the variables of the first order polynomial equation with the 10-bit seed, SEED. Set $d_1 = \lfloor SEED/2^5 \rfloor + 1$ and $d_2 = SEED \mod 2^5$.

  2. Initialize the maximum iteration number, $N = 4$.

3. Initialize an array A with size M to contents $0, 1, \ldots, M-1$ (i.e., $A[i] = i$, for $0 \leq i < M$.

4. Initialize the counter i to $M-1$.

5. Initialize x to -1.

- Repeat the following steps if $i > 0$

1. Initialize the counter j to 0.

2. Loop as follows:

   (a) Increment x and j by 1.

   (b) Calculate the output variable of $y = \{(d_1 \cdot x + d_2) \mod 1031\} \mod M$.

   (c) Repeat the above steps (a) and (b), if $y \leq i$ and $j < N$.

   (d) If $y \leq i$, set $y = y \mod i$.

   (e) Swap the ith and the yth elements in the array, i.e., perform the steps $Temp = A[i]$, $A[i] = A[y]$, and $A[y] = Temp$.

   (f) Decrement i by 1.

Then $PermSeq[i] = A[i]$, where $0 \leq i < M$.

3.4 Test Case Generation

We set some system parameters to build a frame as a test case. It is also used in the later channel estimation study. The parameters are as given below. We will walk through some of the derived mappings in subsequent subsections.

3.4.1 System Parameters

- NPRU = 48

- SAC = 6
- $K_{SB} = 6$
- $K_{MB} = 24$
- $L_{SB} = N_1 * K_{SB} = 24$
- $N_{sub} = 12$
- FPC = 1
- FPCT = 3
- FP0:FP1:FP2:FP3 = 0:1:1:1
- FPSC = 2
- ID_Cell = 1
- SEED = 343
- DL_PermBase = 0

3.4.2 Subband Partitioning

The 48 PRUs map to the subbands according the formulas described in Section 2.3.1. Fig. 3.12 illustrates the PRU to PRU_{SB} mapping for a 10 MHz bandwidth with K_{SB} equal to 6. Table 3.2 shows the mapping between the PRU index and the PRU_{SB} index.

3.4.3 Miniband Partitioning

The remainder of the PRUs are allocated to minibands according the formulas given previously. Fig. 3.13 illustrates the PRU to PRU_{MB} mapping for a 10 MHz bandwidth with

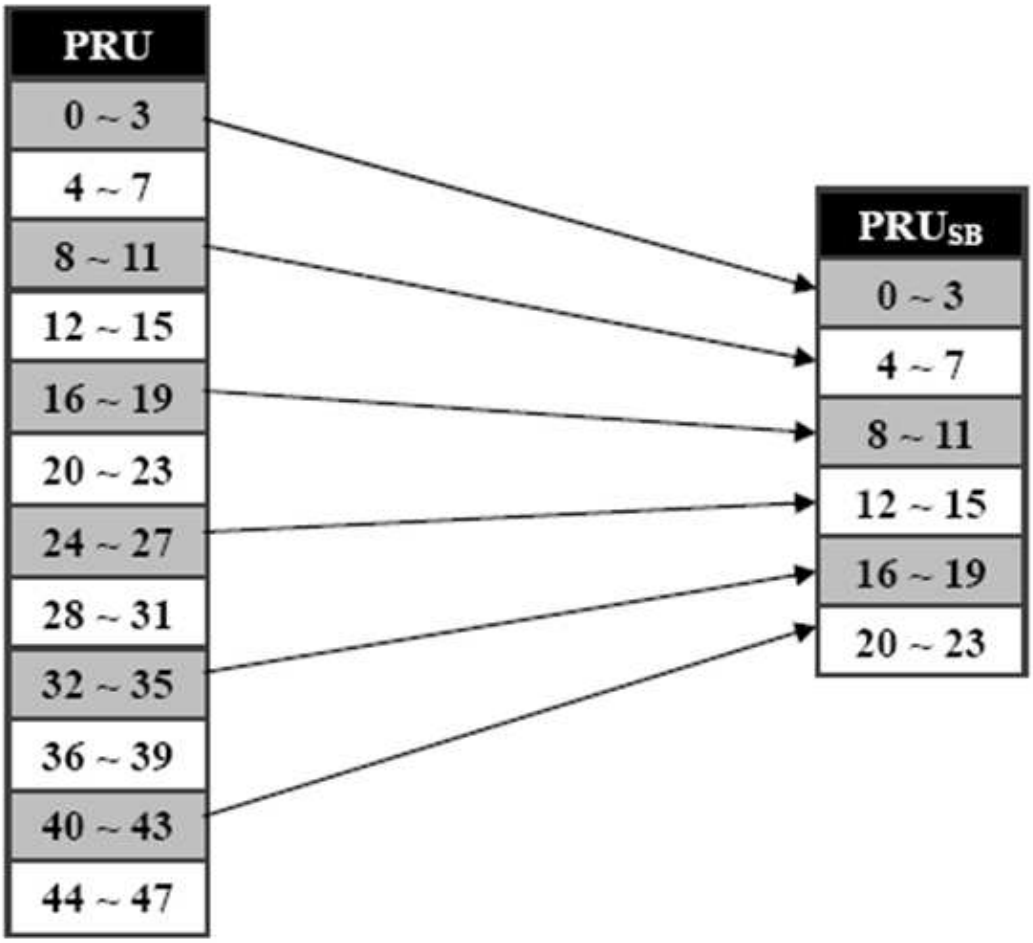


Figure 3.12: PRU to PRU_{SB} mapping.

Table 3.2: Mapping Between PRU Index and PRU_{SB} Index

PRU_{SB} Index	0	1	2	3	4	5	6	7	8	9	10	11
PRU Index	0	1	2	3	8	9	10	11	16	17	18	19
PRU_{SB} Index	12	13	14	15	16	17	18	19	20	21	22	23
PRU Index	24	25	26	27	32	33	34	35	40	41	42	43

K_{MB} equal to 24. Table 3.3 shows the mapping between the PRU index and the PRU_{MB} index.

Table 3.3: Mapping Between PRU Index and $PPRU_{MB}$ Index

PRU_{MB} Index	0	1	2	3	4	5	6	7	8	9	10	11
PRU Index	4	5	6	7	12	13	14	15	20	21	22	23
PRU_{MB} Index	12	13	14	15	16	17	18	19	20	21	22	23
PRU Index	28	29	30	31	36	37	38	39	44	45	46	47

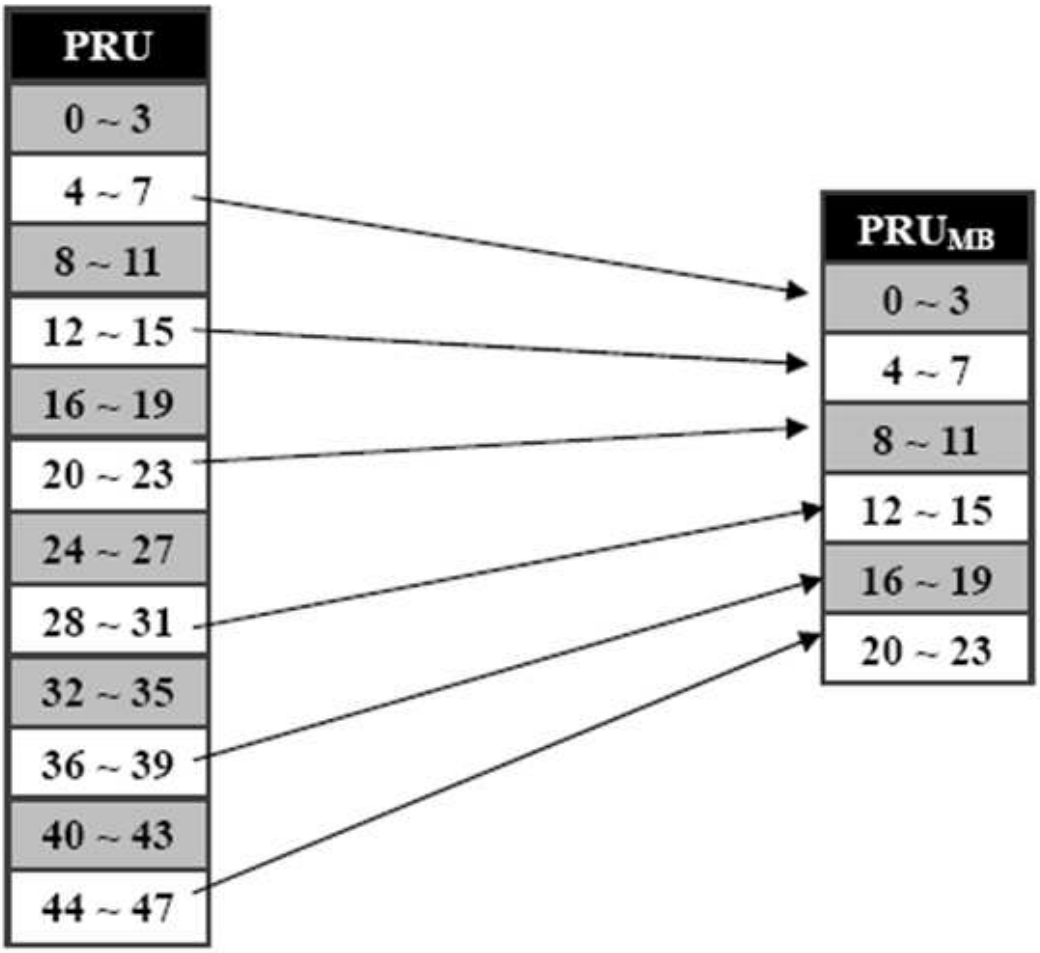


Figure 3.13: PRU to PRU_{MB} mapping.

3.4.4 Miniband Permutation

The miniband permutation maps the PRU_{MB}s to Permuted PRU_{MB}s ($PPRU_{MB}$s) to ensure that frequency diverse PRUs are allocated to each frequency partition. The mapping rule

Table 3.4: Mapping Between PRU_{MB} Index and $PPRU_{MB}$ Index

$PPRU_{MB}$ Index	0	1	2	3	4	5	6	7	8	9	10	11
PRU_{MB} Index	0	4	8	12	16	20	1	5	9	13	17	21
$PPRU_{MB}$ Index	12	13	14	15	16	17	18	19	20	21	22	23
PRU_{MB} Index	2	6	10	14	18	22	3	7	11	15	19	23

Table 3.5: Mapping Between PRU_{SB} Index, $PPRU_{MB}$ Index, and PRU_{FP1} Index

PRU_{FP1} index	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
PRU_{SB} index	0	1	2	3	4	5	6	7	x	x	x	x	x	x	x	x
$PPRU_{MB}$ index	x	x	x	x	x	x	x	x	0	1	2	3	4	5	6	7

Table 3.6: Mapping Between PRU_{SB} Index, $PPRU_{MB}$ Index, and PRU_{FP2} Index

PRU_{FP2} index	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
PRU_{SB} index	8	9	10	11	12	13	14	15	x	x	x	x	x	x	x	x
$PPRU_{MB}$ index	x	x	x	x	x	x	x	x	8	9	10	11	12	13	14	15

is as described previously. Fig. 3.14 illustrates the PRU_{MB} to $PPRU_{MB}$ mapping for a 10 MHz bandwidth with K_{MB} equal to 24. Table 3.4 shows the mapping between the PRU_{MB} index and the $PPRU_{MB}$ index.

3.4.5 Frequency Partitioning

The PRU_{SB} and $PPRU_{MB}$ are allocated to the frequency partitions. There are 3 frequency partitions used because FPCT equals to 3. The PRU_{SB} and $PPRU_{MB}$ map to frequency partitions 1, 2, and 3 according the formulas given previously. Fig. 3.15 illustrates the PRU_{SB} and $PPRU_{MB}$ to frequency partitions mapping for a 10 MHz bandwidth. Tables 3.5, 3.6 and 3.7 show the mapping between the PRU_{SB} index, $PPRU_{MB}$ index, and PRU_{FPi} index for $1 \leq i \leq 3$.

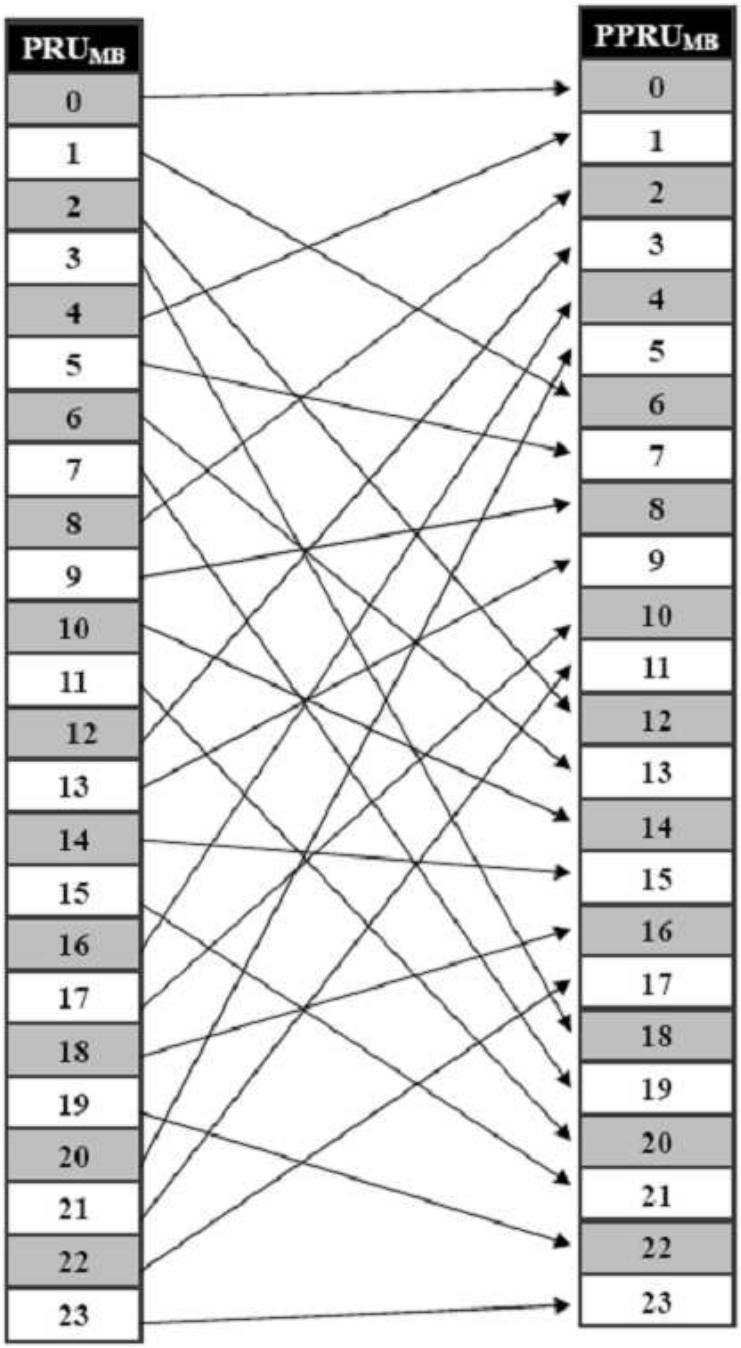


Figure 3.14: PRU_{MB} to $PPRU_{MB}$ mapping.

3.4.6 Random Sequence

We set some parameters to generate a random sequence. We set $ID\_Cell = 1$, $DL\_PermBase = 0$, then $SEED = \{ID\_Cell * 1367\} \mod 2^{10} = 343$. The random sequence is generated ac-

Table 3.7: Mapping Between PRU_{SB} Index, $PPRU_{MB}$ Index, and PRU_{FP3} Index

PRU_{FP1} index	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
PRU_{SB} index	16	17	18	19	20	21	22	23	x	x	x	x	x	x	x	x
$PPRU_{MB}$ index	x	x	x	x	x	x	x	x	16	17	18	19	20	21	22	23

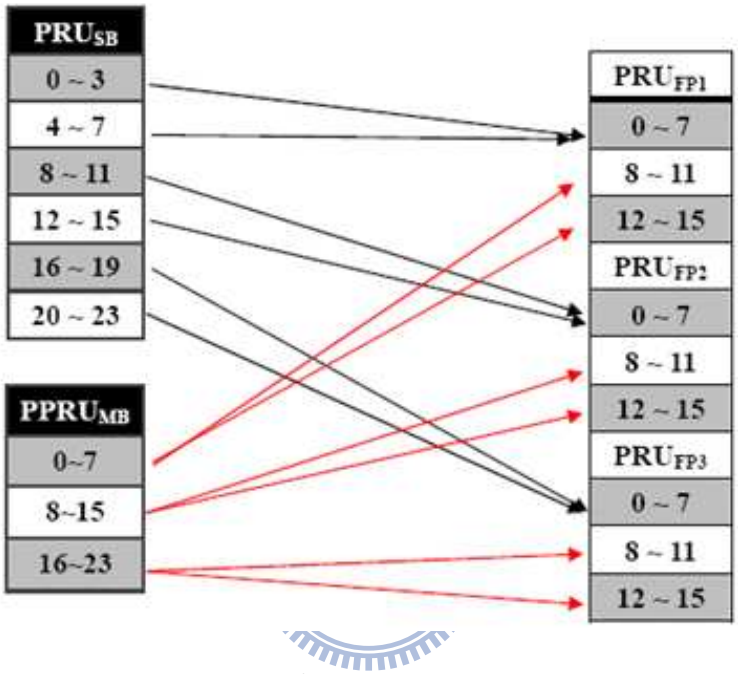


Figure 3.15: PRU_{SB} and $PPRU_{MB}$ to PRU_{FP1} mapping.

Table 3.8: Random Sequence

k	0	1	2	3	4	5	6	7
PermSeq[k]	7	6	4	1	3	0	5	2

cording to the formulas given previously. Table 3.8 shows the random sequence.

Table 3.9: Mapping Between PRU_{FP1} Index and CRU_{FP1}/DRU_{FP1} Index

PRU_{FP1} index	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
CRU_{FP1} index	0	1	2	3	4	5	6	7	x	x	x	x	x	x	x	x
DRU_{FP1} index	x	x	x	x	x	x	x	x	5	3	7	4	2	6	1	0

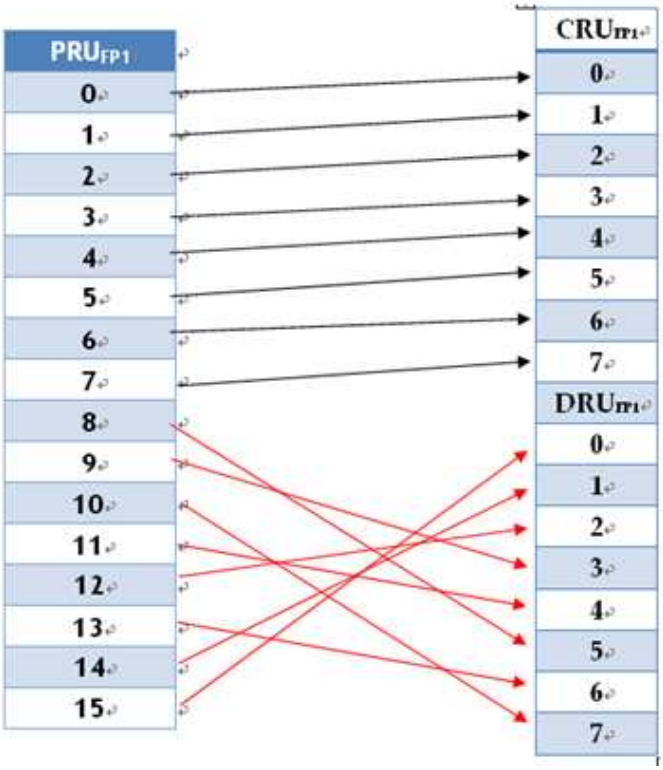


Figure 3.16: The PRU_{FP1}s mapping to CRU/DRU.

3.4.7 CRU/DRU Allocation

The PRU_{FP1}s map to the CRUs and DRUs according to the formulas given previously. Fig. 3.16 illustrates the PRU_{FP1} to CRU/DRU mapping for a 10 MHz bandwidth. Table 3.9 show the mapping between the PRU_{FP1} index and the CRU/DRU index.

Table 3.10: Mapping Between $RSP[k]$ and DRU_{FP1} Index When $l = 0$

DRU0	0	1	2	3	4	5	6	7
$RSP_{l=0}[k]$	7	14	20	25	35	40	53	58
DRU1	0	1	2	3	4	5	6	7
$RSP_{l=0}[k]$	45	50	63	6	12	17	27	32
DRU2	0	1	2	3	4	5	6	7
$RSP_{l=0}[k]$	19	24	37	42	55	62	4	9
DRU3	0	1	2	3	4	5	6	7
$RSP_{l=0}[k]$	60	1	11	16	29	34	47	54
DRU4	0	1	2	3	4	5	6	7
$RSP_{l=0}[k]$	39	46	52	57	3	8	21	26
DRU5	0	1	2	3	4	5	6	7
$RSP_{l=0}[k]$	13	18	31	8	44	49	59	0
DRU6	0	1	2	3	4	5	6	7
$RSP_{l=0}[k]$	51	56	5	10	23	30	36	41
DRU7	0	1	2	3	4	5	6	7
$RSP_{l=0}[k]$	28	33	43	48	61	2	15	22

3.4.8 Subcarrier Permutation

After mapping all pilots, the remainders of the used subcarriers are used to define the distributed LRUs. To allocate the LRUs, the remaining subcarriers are paired into contiguous tone-pairs. Each LRU consists of a group of tone-pairs. A permutation sequence PermSeq() is defined to perform the DL subcarrier permutation according to the formulas given previously. Fig. 3.17 illustrates how the data subcarriers in the DRU are paired into contiguous tone-pairs. Tables 3.10 to 3.15 show the mapping between $RSP[k]$ and DRU_{FP1} index when l equals to 0 to 5 respectively.

Table 3.11: Mapping Between $RSP[k]$ and DRU_{FP1} Index When $l = 1$

DRU0	0	1	2	3	4	5	6	7
$RSP_{l=1}[k]$	6	12	17	27	32	45	50	63
DRU1	0	1	2	3	4	5	6	7
$RSP_{l=1}[k]$	42	55	62	4	9	19	24	37
DRU2	0	1	2	3	4	5	6	7
$RSP_{l=1}[k]$	16	29	34	47	54	60	1	11
DRU3	0	1	2	3	4	5	6	7
$RSP_{l=1}[k]$	57	3	8	21	26	39	46	52
DRU4	0	1	2	3	4	5	6	7
$RSP_{l=1}[k]$	38	44	49	59	0	13	18	31
DRU5	0	1	2	3	4	5	6	7
$RSP_{l=1}[k]$	10	23	30	36	41	51	56	5
DRU6	0	1	2	3	4	5	6	7
$RSP_{l=1}[k]$	48	61	2	15	22	28	33	43
DRU7	0	1	2	3	4	5	6	7
$RSP_{l=1}[k]$	25	35	40	53	58	7	14	20

Table 3.12: Mapping Between $RSP[k]$ and DRU_{FP1} Index When $l = 2$

DRU0	0	1	2	3	4	5	6	7
$RSP_{l=2}[k]$	4	9	19	4	37	42	55	62
DRU1	0	1	2	3	4	5	6	7
$RSP_{l=2}[k]$	47	54	60	1	11	16	29	34
DRU2	0	1	2	3	4	5	6	7
$RSP_{l=2}[k]$	21	26	39	46	52	57	3	8
DRU3	0	1	2	3	4	5	6	7
$RSP_{l=2}[k]$	59	0	13	18	31	38	44	49
DRU4	0	1	2	3	4	5	6	7
$RSP_{l=2}[k]$	36	41	51	56	5	10	23	30
DRU5	0	1	2	3	4	5	6	7
$RSP_{l=2}[k]$	15	22	28	33	43	48	61	2
DRU6	0	1	2	3	4	5	6	7
$RSP_{l=2}[k]$	53	58	7	14	20	25	35	40
DRU7	0	1	2	3	4	5	6	7
$RSP_{l=2}[k]$	27	32	45	50	63	6	12	17

Table 3.13: Mapping Between $RSP[k]$ and DRU_{FP1} Index When $l = 3$

DRU0	0	1	2	3	4	5	6	7
$RSP_{l=3}[k]$	1	11	16	29	34	47	54	60
DRU1	0	1	2	3	4	5	6	7
$RSP_{l=3}[k]$	46	52	57	3	8	21	26	39
DRU2	0	1	2	3	4	5	6	7
$RSP_{l=3}[k]$	18	31	38	44	49	59	0	13
DRU3	0	1	2	3	4	5	6	7
$RSP_{l=3}[k]$	56	5	10	23	30	36	41	51
DRU4	0	1	2	3	4	5	6	7
$RSP_{l=3}[k]$	33	43	48	61	2	15	22	28
DRU5	0	1	2	3	4	5	6	7
$RSP_{l=3}[k]$	14	20	25	35	40	53	58	7
DRU6	0	1	2	3	4	5	6	7
$RSP_{l=3}[k]$	50	63	6	12	17	27	32	45
DRU7	0	1	2	3	4	5	6	7
$RSP_{l=3}[k]$	24	37	42	55	62	4	9	19

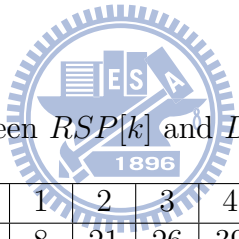


Table 3.14: Mapping Between $RSP[k]$ and DRU_{FP1} Index When $l = 4$

DRU0	0	1	2	3	4	5	6	7
$RSP_{l=4}[k]$	3	8	21	26	39	46	52	57
DRU1	0	1	2	3	4	5	6	7
$RSP_{l=4}[k]$	44	49	59	0	13	18	31	38
DRU2	0	1	2	3	4	5	6	7
$RSP_{l=4}[k]$	23	30	36	41	51	56	5	10
DRU3	0	1	2	3	4	5	6	7
$RSP_{l=4}[k]$	61	2	15	22	28	33	43	48
DRU4	0	1	2	3	4	5	6	7
$RSP_{l=4}[k]$	35	40	53	58	7	14	20	25
DRU5	0	1	2	3	4	5	6	7
$RSP_{l=4}[k]$	12	17	27	32	45	50	63	6
DRU6	0	1	2	3	4	5	6	7
$RSP_{l=4}[k]$	55	62	4	9	19	24	37	42
DRU7	0	1	2	3	4	5	6	7
$RSP_{l=4}[k]$	29	34	47	54	60	1	11	16

Table 3.15: Mapping Between $RSP[k]$ and DRU_{FP1} Index When $l = 5$

DRU0	0	1	2	3	4	5	6	7
$RSP_{l=5}[k]$	0	13	18	31	38	44	49	59
DRU1	0	1	2	3	4	5	6	7
$RSP_{l=5}[k]$	41	51	56	5	10	23	30	36
DRU2	0	1	2	3	4	5	6	7
$RSP_{l=5}[k]$	22	28	33	43	48	61	2	15
DRU3	0	1	2	3	4	5	6	7
$RSP_{l=5}[k]$	58	7	14	20	25	35	40	53
DRU4	0	1	2	3	4	5	6	7
$RSP_{l=5}[k]$	32	45	50	63	6	12	17	27
DRU5	0	1	2	3	4	5	6	7
$RSP_{l=5}[k]$	9	19	24	37	42	55	62	4
DRU6	0	1	2	3	4	5	6	7
$RSP_{l=5}[k]$	54	60	1	11	16	29	34	47
DRU7	0	1	2	3	4	5	6	7
$RSP_{l=5}[k]$	26	39	46	52	57	3	8	21

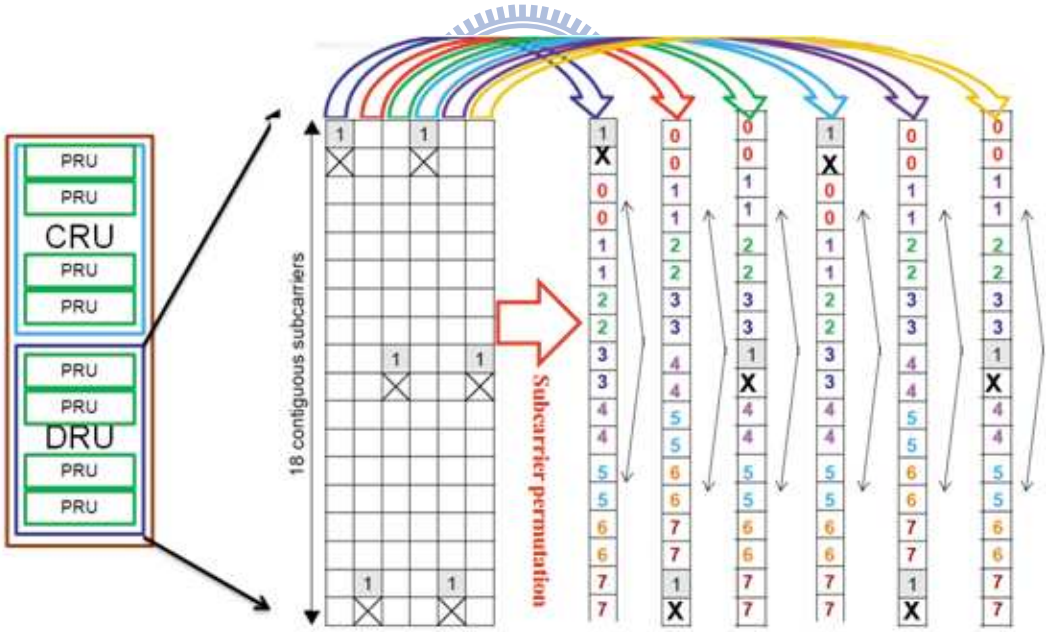


Figure 3.17: How the data subcarriers in the DRU are paired.

Chapter 4

Channel Estimation Techniques

Channel estimators in OFDMA system usually need pilot information as reference. A fading channel requires constant tracking, so pilot information has to be transmitted continuously. In general, the fading channel can be viewed as a two-dimensional (2-D) signal (time and frequency), whose values are sampled at pilot positions.

We consider three topics in this chapter, which are channel estimation at pilot subcarriers, interpolation schemes and minimum mean-square error (LMMSE) estimation. In the final proposal design, we use the least-squares (LS) technique to estimate the channel response at pilots, use linear interpolation in the time domain to estimate the frequency response at nonpilot subcarriers, and use LMMSE channel estimation in the frequency domain to estimate the frequency response at nonpilot subcarriers. These building-block techniques are introduced separately in the following subsections.

4.1 The Least-Squares (LS) Estimator

Based on the a priori known data, we can estimate the channel information at pilot carriers roughly by the least-squares (LS) estimator. An LS estimator minimizes the squared error

[16]

$$||\mathbf{Y} - \hat{\mathbf{H}}_{LS}\mathbf{X}||^2 \tag{4.1}$$

where $\mathbf{Y}$ is the received signal and $\mathbf{X}$ is a priori known pilots, both in the frequency domain and both being $N \times 1$ vectors where N is the FFT size. $\hat{\mathbf{H}}_{LS}$ is an $N \times N$ matrix whose values are 0 except at pilot locations m_i where $i = 0, \cdots, N_p - 1$:

$$\hat{\mathbf{H}}_{LS} = \begin{bmatrix} H_{m_0,m_0} & \cdots & 0 & \cdots & 0 & \cdots & 0 \\ 0 & \cdots & H_{m_1,m_1} & \cdots & 0 & \cdots & 0 \\ 0 & \cdots & 0 & \cdots & H_{m_2,m_2} & \cdots & 0 \\ 0 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\ 0 & \cdots & 0 & \cdots & 0 & \cdots & H_{m_{N_p-1},H_{m_{N_p-1}}} \end{bmatrix}. \tag{4.2}$$

Therefore, (4.1) can be rewritten as

$$[Y(m) - \hat{H}_{LS}(m)X(m)]^2, \text{ for all } m = m_i. \tag{4.3}$$

Then the estimate of pilot signals, based on only one observed OFDMA symbol, is given by

$$\hat{H}_{LS}(m) = \frac{Y(m)}{X(m)} = \frac{X(m)H(m) + N(m)}{X(m)} = H(m) + \frac{N(m)}{X(m)} \tag{4.4}$$

where $N(m)$ is the complex white Gaussian noise on subcarrier m. We collect $H_{LS}(m)$ into $\hat{\mathbf{H}}_{\mathbf{p,LS}}$, an $N_p \times 1$ vector where N_p is the total number of pilots, as

$$\begin{aligned} \hat{\mathbf{H}}_{p,LS} &= [H_{p,LS}(0)\ H_{p,LS}(1)\ \cdots H_{p,LS}(N_p - 1)]^T \\ &= [\tfrac{Y_p(0)}{X_p(0)}, \tfrac{Y_p(1)}{X_p(1)}, \ldots, \tfrac{Y_p(N_p-1)}{X_p(N_p-1)}]^T. \end{aligned} \tag{4.5}$$

The LS estimator is a simplest channel estimator one can think of.

4.2 Linear Interpolation

After obtaining the channel response estimate at the pilot subcarriers, we use interpolation to obtain the response at the rest of the subcarriers. Linear interpolation is a commonly considered scheme due to its low complexity. It does the interpolation between two known

data. That is, we use the channel information at two pilot subcarriers obtained by the LS estimator to estimate the channel frequency response information at the data subcarriers between them. We also use linear extrapolation to estimate the response as the data subcarriers beyond the outermost pilot subcarriers.

Linear interpolation may be done in frequency or in time. In frequency domain linear interpolation, the channel estimate at data subcarrier k, $mL < k < (m+1)L$, using linear interpolation is given by [13]

$$H_e(k) = H_e(mL + l) = [H_p(m+1) - H_p(m)]\frac{l}{L} + H_p(m) \tag{4.6}$$

where $H_p(k)$, $k = 0, 1, \cdots, N_p$, are the channel frequency responses at pilot subcarriers, L is the pilot subcarriers spacing, and $0 < l < L$.

4.3 LMMSE Channel Estimation

The material in this section is mainly taken from [14].

4.3.1 Channel Modeling for Channel Estimation

Consider a discrete-time equivalent lowpass channel impulse response

$$h(n) = \sum_{l=0}^{L-1} \alpha_l \delta(n - l) \tag{4.7}$$

where n and l are integers in units of the sampling period T_s and α_l is the complex gain of path l. The mean delay and the RMS delay spread are given by, respectively,

$$\tau_\mu = \frac{\sum_{l=0}^{L-1} E(|\alpha_l)|^2 l}{\sum_{l=0}^{L-1} E(|\alpha_l|^2)} \tag{4.8}$$

and

$$\tau_{rms} = \sqrt{\frac{\sum_{l=0}^{L-1} E(|\alpha_l|^2)(l - \tau_\mu)^2}{\sum_{l=0}^{L-1} E(|\alpha_l|^2)}}. \tag{4.9}$$

One question here is how the expectation $E(|\alpha_l|^2)$ should be defined. As our purpose is channel estimation, suppose one channel estimation is performed for K OFDM symbols. Then the expectation should be an average that is taken over these symbols. In the extreme case of $K = 1$, no average should be taken, but the instantaneous channel response in that symbol period should be used to compute τ_μ and τ_{rms}.

A common design is to put channel estimation after carrier frequency and OFDM symbol timing synchronization. Since typical symbol timing synchronizers may yield some error, for simplicity and practicality we assume that the channel estimator input contains no carrier frequency error, but the PDP can have a nonzero initial delay τ_0, although conventional definition of the PDP usually zero out the initial path delays.

Fourier transforming the PDP gives the corresponding frequency autocorrelation function. For an exponential PDP with possibly nonzero initial delay τ_0, we have

$$\frac{R_f(k)}{R_f(0)} = \frac{e^{-j2\pi\tau_0 k/N}}{1 + j2\pi\tau_{rms}k/N} \tag{4.10}$$

where $\tau_0 = \tau_\mu - \tau_{rms}$ and N is the discrete Fourier transform (DFT) size used in the multicarrier system. For a uniform PDP of width T with a possibly nonzero initial delay τ_0,

$$\frac{R_f(k)}{R_f(0)} = \frac{e^{-j2\pi\tau_0 k/N}\sin(\pi Tk/N)}{\pi Tk/N} \tag{4.11}$$

where $\tau_\mu = \tau_0 + T/2$ and $T = \sqrt{12\tau_{rms}}$.

4.3.2 Estimation of Channel Delay Parameters

The frequency response of the channel (4.7) is given by

$$H(f) = \sum_{l=0}^{L-1} \alpha_l e^{-j2\pi lf/N} \tag{4.12}$$

where the division by N in the exponent normalizes the period of $H(f)$ in f to N. If we advance the channel response by τ (arbitrary) time units, then the frequency response

becomes

$$H_a(f) = e^{j2\pi\tau f/N} H(f) = \sum_{l=0}^{L-1} \alpha_l (l-\tau) e^{-j2\pi(l-\tau)f/N}. \tag{4.13}$$

Differentiating $H_a(f)$ with respect to f, we get

$$\frac{dH_a(f)}{df} = \frac{-j2\pi}{N} \sum_{l=0}^{L-1} \alpha_l (l-\tau) e^{-j2\pi(l-\tau)f/N}. \tag{4.14}$$

Applying Parsevals theorem, we get

$$J(\tau) \triangleq \left\langle |\frac{dH_a(f)}{df}|^2 \right\rangle = \frac{4\pi^2}{N^2} \sum_{l=0}^{L-1} |\alpha_l|^2 (l-\tau)^2 \tag{4.15}$$

where $< \cdot >$ denotes frequency averaging. Hence

$$\overline{J}(\tau) \triangleq \left\langle |\frac{dH_a(f)}{df}|^2 \right\rangle = \frac{4\pi^2}{N^2} \sum_{l=0}^{L-1} E(|\alpha_l|^2)(l-\tau)^2. \tag{4.16}$$

The above equations show that $\overline{J}(\tau)$ is minimized when $\tau = \tau_\mu$. In addition,

$$\tau_{rms}^2 = \frac{N^2 \min \overline{J}(\tau)}{4\pi^2 \sum_{l=0}^{L-1} E(|\alpha_l|^2)}. \tag{4.17}$$

We can estimate τ_μ and τ_{rms} by this way, and it is suitable for typical pilot-transmitting OFDM systems.

Consider a system where one out of every F_s subcarriers is a pilot. Later, we will see that F_s can be made equal to 4 and 8 for IEEE 802.16e and 802.16m. We can approximate $dH_a(f)/df$ by first-order difference, say, $[H_a(f+F_s) - H_a(f)]/F_s$, and substitute it into (4.16). Then, we obtain

$$\overline{J}(\tau) \approx \frac{1}{F_s^2} E\left\langle |e^{j\phi} H(f+F_s) - H(f)|^2 \right\rangle_p \tag{4.18}$$

where $\phi = 2\pi\tau F_s/N$, f takes values only over pilot frequencies (which are integers by the earlier normalization of frequency in (4.12), and $< \cdot >$ denotes averaging over pilot subcarriers. Then we modify the approximation by taking circular differencing over f rather than linear differencing. Therefore, we approximate $\overline{J}(\tau)$ by

$$\overline{J}(\tau) \approx \frac{1}{F_s^2} E|\left\langle |e^{j\phi} H(f+F_s)\%N - H(f)|^2 \right\rangle_p \tag{4.19}$$

where % denotes modulo operation, $< \cdot >$ now averages over the full number of pilot subcarriers, and we have assumed that $(f + F_s)\%N$ is a pilot subcarrier. Now let R_i be the (instantaneous) frequency-domain autocorrelation of the channel response:

$$R_i = \left\langle H((f + iF_s\%N)H^*(f) \right\rangle_p. \tag{4.20}$$

Then from (4.19) we have

$$\overline{J}(\tau) \approx \frac{2}{F_s^2}[E(R_0 - \Re\{e^{j\phi}E(R_1))\}]. \tag{4.21}$$

Then (4.21) gives an approximation of $\overline{J}(\tau)$ defined in (4.16). According to (4.21), τ_μ and τ_{rms} can be estimated in the following way:

1. estimate the channel responses at the pilot subcarriers,
2. estimate R_i $(i = 0, 1)$,
3. estimate $\overline{J}(\tau)$,
4. find the value of τ that minimizes $\overline{J}(\tau)$, and
5. substitute the result into (4.17) to estimate τ_{rms}^2.

Step 1 can be achieved using the LS method. Then, in step 2, R_0 and R_1 can be estimated via

$$\hat{R}_0 = \left\langle |\hat{H}(f)|^2 \right\rangle_p - \hat{\sigma}_n^2, \quad \hat{R}_1 = \left\langle \hat{H}(f + F_s\%N)\hat{H}^*(f) \right\rangle_p. \tag{4.22}$$

where $\hat{H}(f)$ denotes the estimated channel response at pilot subcarrier f and, we may estimate σ_n^2 from the received power in the null subcarriers of the system. Thus, for step 3, $\overline{J}(\tau)$ can be estimated using

$$\hat{J}_{Av}(\tau) \triangleq \frac{2}{F_s^2}\left[Av(\hat{R}_0) - R\{e^{j\phi}Av(\hat{R}_1)\}\right] \tag{4.23}$$

where Av denotes time averaging, i.e., averaging over OFDM symbols. If one performs a channel estimation over K OFDM symbols, then the averages should be taken over these K symbols. If $K = 1$, then the instantaneous values should be used instead of averages. For step 4, we may estimate the mean delay as

$$\hat{\tau}_\mu \triangleq \arg\min \hat{J}_{Av}(\tau) = -\frac{N\angle Av(\hat{R}_1)}{2\pi F_s}, \tag{4.24}$$

which also yields $\min \hat{J}_{Av}(\tau) = 2[Av(\hat{R}_0) - |Av(\hat{R}_1)|]/F_s^2$. Finally, for step 5, in view of (4.17) and that $R_0 =< |H(f)|^2 >_p$, we may estimate τ_{rms} as

$$\hat{\tau}_{rms} = \frac{N}{2\pi F_s}\sqrt{2\Big[1 - \frac{|Av(\hat{R}_1)|}{Av(\hat{R}_0)}\Big]}. \tag{4.25}$$

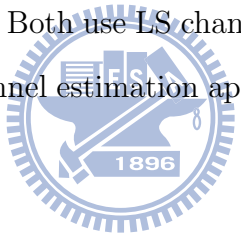
4.3.3 LMMSE Filtering

To complete LMMSE channel estimation, the above estimates of delay parameters, namely, $\hat{\tau}_\mu$ and $\hat{\tau}_{rms}$, can be substituted into proper places in (4.10) or (4.11) depending on the choice of PDP model. Then the resulting autocorrelation function of channel frequency response can be used in LMMSE channel estimation as outlined in chapter 1.

Chapter 5

Downlink Channel Estimation Simulation for IEEE 802.16e

In this chapter, we use two methods to do the channel estimation in downlink transmission for IEEE 802.16e. The channel estimation techniques are advanced 4-point linear interpolation [18] and LMMSE channel estimation. Both use LS channel estimation at pilot positions. We evaluate the performance of each channel estimation approach mainly via mean square error (MSE) and symbol error rate (SER).



5.1 System Parameters and Channel Model

Table 5.1 gives the primitive and derived parameters used in our simulation work. In our system, we let the preamble be followed by 24 data symbols. In addition to AWGN, we use SUI-2 to SUI-6 to do simulation. Their profiles are introduced in Tables 5.2–5.4.

5.1.1 Simulation Channel Model

Erceg *et al.* [15] published a total of 6 different radio channel models for type G2 (i.e, LOS and NLOS) MMDS BWA systems in three terrain categories. The three types in suburban area are:

Table 5.1: OFDMA Downlink Parameters

Parameters	Values
Bandwidth	10 MHz
Central frequency	3.5 GHz
N_{used}	841
Sampling factor n	28/25
G	1/8
N_{FFT}	1024
Sampling frequency	11.2 MHz
Subcarrier spacing	10.94 kHz
Useful symbol time	91.43 μs
CP time	11.43 μs
OFDMA symbol time	102.86 μs
Sampling time	44.65 ns

- A: hilly terrain, heavy tree,
- C: flat terrain, light tree, and
- B: between A and C.

The correspondence with the so-called SUI channels is:

- C: SUI-1, SUI-2,
- B: SUI-3, SUI-4, and
- A: SUI-5, SUI-6.

In the above, SUI-1 and SUI-2 are Ricean multipath channels, whereas the other four are from Hari and are Rayleigh multipath channels. The Rayleigh channels are more hostile and exhibit a greater RMS delay spread. And the SUI-2 represents a worst-case link for terrain type C. We employ SUI-2 and SUI-3 model in our simulation, but we use Rayleigh fading to model all the paths in these channels. The channel characteristics are as shown in Tables 5.2–5.4.

Table 5.2: Channel Profile of SUI-2 [15]

SUI – 2 Channel				
	Tap 1	**Tap 2**	**Tap 3**	**Units**
Delay	0	0.4	1.1	μs
Power (omni ant.)	0	-12	-15	dB
90% K-fact. (omni)	2	0	0	
75% K-fact. (omni)	11	0	0	
Power (30° ant.)	0	-18	-27	dB
90% K-fact. (30°)	8	0	0	
75% K-fact. (30°)	36	0	0	
Doppler	0.2	0.15	0.25	Hz
Antenna Correlation: $\rho_{ENV} = 0.5$ **Gain Reduction Factor:** GRF = 2 dB **Normalization Factor:** $F_{omni} = -0.3930$ dB, $F_{30°} = -0.0768$ dB			**Terrain Type:** C **Omni antenna:** $\tau_{RMS} = 0.202$ μs, overall K: K = 1.6 (90%); K = 5.1 (75%) **30° antenna:** $\tau_{RMS} = 0.069$ μs, overall K: K = 6.9 (90%); K = 21.8 (75%)	

5.2 Channel Estimation Methods

The first symbol of a downlink subframe is preamble, and we assume that 24 data symbols follow in the subframe. Pilots in the preamble appear every 3 subcarriers. We consider both advanced 4-point linear interpolation and LMMSE channel estimation and compare their performance.

5.2.1 Advanced Four-Point Cluster Linear Interpolation for IEEE 802.16e

The channel may be modeled as linearly varying in a short time period. This can be used to yield a predicted channel response at future OFDMA symbol instants, for example,

$$H_k(t+1) = H_k(t) + [H_k(t) - H_k(t-1)]. \tag{5.1}$$

Table 5.3: Channel Profiles of SUI-3 and SUI-4 [15]

SUI – 3 Channel				
	Tap 1	**Tap 2**	**Tap 3**	**Units**
Delay	0	0.4	0.9	μs
Power (omni ant.)	0	-5	-10	dB
90% K-fact. (omni)	1	0	0	
75% K-fact. (omni)	7	0	0	
Power (30° ant.)	0	-11	-22	dB
90% K-fact. (30°)	3	0	0	
75% K-fact. (30°)	19	0	0	
Doppler	0.4	0.3	0.5	Hz
Antenna Correlation: $\rho_{ENV} = 0.4$ **Gain Reduction Factor:** GRF = 3 dB **Normalization Factor:** $F_{omni} = -1.5113$ dB, $F_{30°} = -0.3573$ dB			**Terrain Type:** B **Omni antenna:** $\tau_{RMS} = 0.264$ μs, overall K: K = 0.5 (90%); K = 1.6 (75%) **30° antenna:** $\tau_{RMS} = 0.123$ μs, overall K: K = 2.2 (90%); K = 7.0 (75%)	

SUI – 4 Channel				
	Tap 1	**Tap 2**	**Tap 3**	**Units**
Delay	0	1.5	4	μs
Power (omni ant.)	0	-4	-8	dB
90% K-fact. (omni)	0	0	0	
75% K-fact. (omni)	1	0	0	
Power (30° ant.)	0	-10	-20	dB
90% K-fact. (30°)	1	0	0	
75% K-fact. (30°)	5	0	0	
Doppler	0.2	0.15	0.25	Hz
Antenna Correlation: $\rho_{ENV} = 0.3$ **Gain Reduction Factor:** GRF = 4 dB **Normalization Factor:** $F_{omni} = -1.9218$ dB, $F_{30°} = -0.4532$ dB			**Terrain Type:** B **Omni antenna:** $\tau_{RMS} = 1.257$ μs overall K: K = 0.2 (90%); K = 0.6 (75%) **30° antenna:** $\tau_{RMS} = 0.563$ μs overall K: K = 1.0 (90%); K = 3.2 (75%)	

If the receiver latency is not a concern, time-domain interpolation can be performed. A simplest way of time-domain interpolation is, of course, linear interpolation, such as [17]:

$$H_k(t) = \frac{1}{2}[H_k(t-1) + H_k(t+1)]. \tag{5.2}$$

We take the pilots in the previous and the next symbols as reference. As shown in

Table 5.4: Channel Profiles of SUI-5 and SUI-6 [15]

SUI – 5 Channel				
	Tap 1	**Tap 2**	**Tap 3**	**Units**
Delay	0	4	10	µs
Power (omni ant.)	0	-5	-10	dB
90% K-fact. (omni)	0	0	0	
75% K-fact. (omni)	0	0	0	
50% K-fact (omni)	2	0	0	
Power (30° ant.)	0	-11	-22	dB
90% K-fact. (30°)	0	0	0	
75% K-fact. (30°)	2	0	0	
50% K-fact. (30°)	7	0	0	
Doppler	2	1.5	2.5	Hz
Antenna Correlation: $\rho_{ENV} = 0.3$ **Gain Reduction Factor:** GRF = 4 dB **Normalization Factor:** F_{omni} = -1.5113 dB, $F_{30°}$ = -0.3573 dB			**Terrain Type:** A **Omni antenna:** $\tau_{RMS} = 2.842$ µs overall K: K = 0.1 (90%); K = 0.3 (75%); K = 1.0 (50%) **30° antenna:** $\tau_{RMS} = 1.276$ µs overall K: K = 0.4 (90%); K = 1.3 (75%); K = 4.2 (50%)	

SUI – 6 Channel				
	Tap 1	**Tap 2**	**Tap 3**	**Units**
Delay	0	14	20	µs
Power (omni ant.)	0	-10	-14	dB
90% K-fact. (omni)	0	0	0	
75% K-fact. (omni)	0	0	0	
50% K-fact. (omni)	1	0	0	
Power (30° ant.)	0	-16	-26	dB
90% K-fact. (30°)	0	0	0	
75% K-fact. (30°)	2	0	0	
50% K-fact. (30°)	5	0	0	
Doppler	0.4	0.3	0.5	Hz
Antenna Correlation: $\rho_{ENV} = 0.3$ **Gain Reduction Factor:** GRF = 4 dB **Normalization Factor:** F_{omni} = -0.5683 dB, $F_{30°}$ = -0.1184 dB			**Terrain Type:** A **Omni antenna:** $\tau_{RMS} = 5.240$ µs overall K: K = 0.1 (90%); K = 0.3 (75%); K = 1.0 (50%) **30° antenna:** $\tau_{RMS} = 2.370$ µs overall K: K = 0.4 (90%); K = 1.3 (75%); K = 4.2 (50%)	

Fig. 5.1, this would result in four pilots in a cluster to estimate the other data subcarriers response instead of its only two original pilots. We do linear interpolation in the frequency

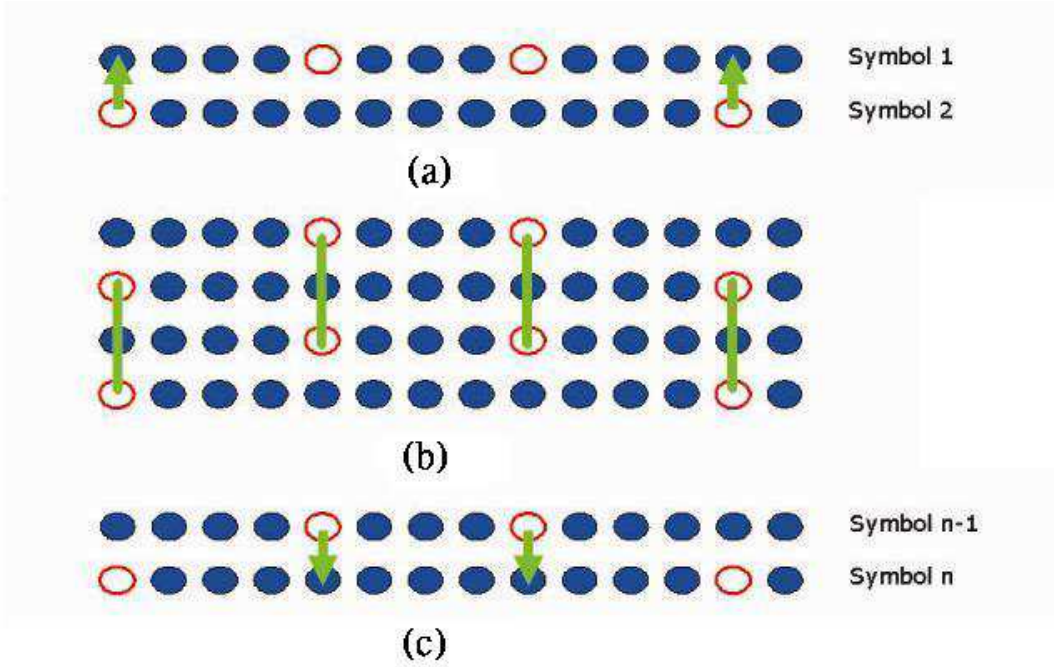


Figure 5.1: Advanced four-point cluster linear interpolation. (a) First data symbol. (b) Second to $(n-1)$th data symbols. (c) Last (nth) data symbol. (From [18]).

domain within the cluster afterwards. The detailed steps for each symbol are as follows:

1) First data symbol:

- Estimate the channel response at each pilot location by using the LS technique.
- Take the pilots in the next symbol as reference (see Fig. 5.1(a)) to yield four "virtual" pilots in a cluster.
- Use linear interpolation to estimate the data subcarrier responses from the four estimated "virtual" pilot responses.

2) Second to $(n-1)$th data symbols:

- Estimate the channel response at each pilot location using the LS technique.

- Take the pilots in the previous and the next symbols as reference (see Fig 5.1(b)) to yield 4 "virtual" pilots in a cluster.
- Use linear interpolation to estimate the data subcarrier responses from the 4 estimated "virtual" pilot responses.

3) Last data symbol:

- Estimate the channel response at each pilot location using the LS technique.
- Take the pilots in the previous symbol as reference (see Fig 5.1(c)) to yield 4 "virtual" pilots in a cluster.
- Use linear interpolation to estimate the data subcarrier responses from the 4 estimated "virtual" pilot responses.

This method gives our system a symbol time latency. For example, if we want to get the channel response of the first symbol, we must wait until we receive the second data symbol. In the last symbol, we get not only the previous symbol information but also the last one's.

5.2.2 LMMSE Channel Estimation for IEEE 802.16e

We employ the technique of [14], where the following observation is made. By the pseudo-random frequency distribution, two closest clusters may not be next to each other in frequency, but may be widely apart to the point well beyond channel coherence bandwidth. Hence it may not pay to perform LMMSE channel estimation across cluster boundaries. Thus LMMSE filtering is done within each cluster separately. Although there are only two pilots per cluster in any OFDM symbol, there are four frequencies where a pilot may appear over time. Hence we try to have 4-tap filters. The LMMSE channel estimation for IEEE 802.16e downlink proceeds as follows.

1. Do LS channel estimation at pilot subcarriers of all used clusters.

2. Linearly interpolate in time to acquire two additional channel estimates per cluster, as advanced four-point cluster linear interpolation illustrated in Fig. 5.1.

3. Estimate R_0 and R_1.

4. Estimate τ_μ and τ_{rms}.

5. Find the autocorrelation function associated with the exponential power delay profile (PDP) as

$$\frac{R_f(k)}{R_f(0)} = \frac{e^{-j2\pi\tau_0 k/N}}{1 + j2\pi\tau_{rms}k/N}. \tag{5.3}$$

   where $\tau_0 = \tau_\mu - \tau_{rms}$ and N is the discrete Fourier transform (DFT) size used in the multicarrier system. ($N = 1024$ in our system.)

6. Base on the above autocorrelation function, do LMMSE filtering to estimate the data subcarrier responses as where the exact meaning of $\hat{\sigma}_n^2\mathbf{I}$ will be putted below.

$$\underline{w}_d = (\hat{R}_p + \hat{\sigma}_n^2\mathbf{I})^{-1}\underline{r}_{dp}, \tag{5.4}$$

$$\hat{H}_d = \underline{w}_d^H \underline{\hat{H}}_p. \tag{5.5}$$

Let $\hat{H}(f)$ denote the resulting channel estimate at pilot subcarrier f from step 1. Let σ_n^2 be the variance of additive white gaussian noise (AWGN) at the pilot locations. After linear interpolation from two pilots in step2, the noise variance at the interpolated locations becomes $\sigma_n^2/2$. For step 3, R_0 and R_1 can be estimated via

$$\hat{R}_0 = \left\langle |\hat{H}(f)|^2 \right\rangle_p - \hat{\sigma}_n^2, \quad \hat{R}_1 = \left\langle \hat{H}(f + F_s\%N)\hat{H}^*(f) \right\rangle_p, \tag{5.6}$$

where $\left\langle |\hat{H}(f)|^2 \right\rangle$ is the averaged magnitude-squares of estimated pilot subcarrier channel responses, $\hat{\sigma}_n^2$ is the estimated noise variance at pilot subcarriers, $F_s = 4$ (there is one

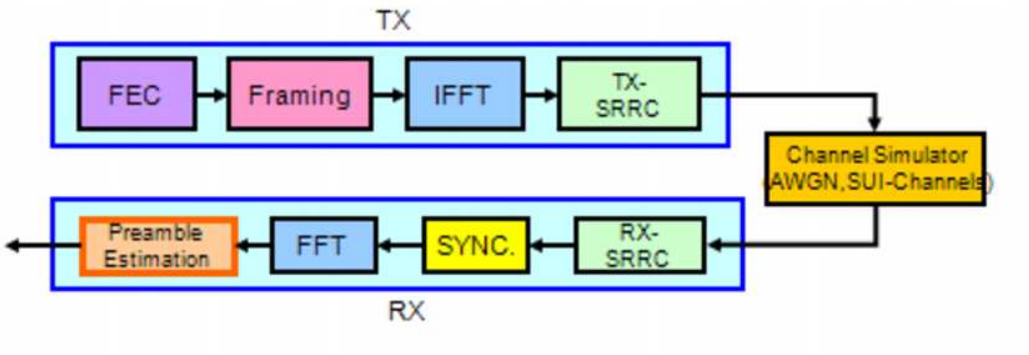
pilot subcarrier of every 4 subcarriers), and $N = 1024$. According the signal structure for IEEE 802.16e, $\hat{\sigma}_n^2$ here should equal to $3/4 \times \sigma_n^2$ in the 2nd–23th data symbols of a DL subframe. This is because there are two pilot responses in a cluster that we obtained by linear interpolation in time. Hence the noise variance at the interpolated pilot locations becomes $1/2 \times \hat{\sigma}_n^2$. The average of the noise variance in a cluster thus becomes $(1+1+1/1+1/2)/4 = 3/4 \times \sigma_n^2$. In the first and the last symbols, because we do no time-domain linear interpolation, the noise variance does not need to be scaled. In summary, we have to modify the scale of the noise variance according to the signal structure. For step 4, we may estimate the mean delay as

$$\hat{\tau}_\mu = -\frac{N \angle Av(\hat{R}_1)}{2\pi F_s}. \tag{5.7}$$

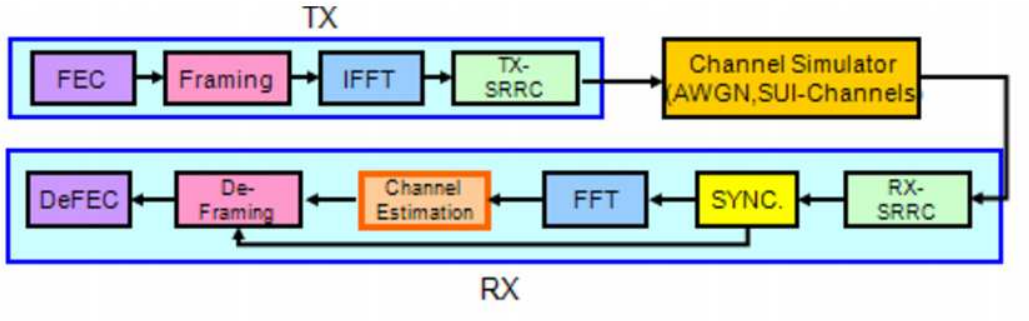
And we may estimate the RMS delay spread as

$$\hat{\tau}_{rms} = \frac{N}{2\pi F_s}\sqrt{2\left[1 - \frac{|Av(\hat{R}_1)|}{Av(\hat{R}_0)}\right]}, \tag{5.8}$$

where Av denotes time averaging, i.e., averaging over OFDM symbols. In our present work, we do delay estimation based on only one OFDM symbol for simplicity and for letter performance in a time-varying channel. Hence we estimate $\hat{R}_0$ and $\hat{R}_1$ for each OFDM symbol separately. For step 5, we use the exponential PDP rather than the uniform PDP, because the performance with exponential PDP is better than uniform PDP [14]. Finally, for step 6, the $\hat{\sigma}_n^2$ in diagonal elements of the matrix $(\mathrm{R}_p + \sigma_n^2 \mathrm{I})$ should be scaled property as discussed abovee. That is the channel response at the pilot location is interpolated in time, the $\hat{\sigma}_n^2$ at the associated element of the matrix should be multiplied by $1/2$, else we should do no modification.



(a)



(b)

Figure 5.2: Downlink transmission simulation flow. (a)Preamble. (b)Data symbols.

5.3 Simulation Results for IEEE 802.16e

5.3.1 Simulation Flow

Figure 5.2 illustrate the block diagrams of our simulated system. We also assume perfect synchronization and omit it in our simultion. Because the preamble is all pilots, there is no need to do DeFraming and DeFEC. For data transmission simulation, we do not do FEC and DeFEC. After channel estimation, we calculate the channel MSE between the true channel and the estimated one, where the average is taken over all the subcarriers. The symbol error rate (SER) can also be obtained after demapping.

5.3.2 Validation with AWGN Channel

Before considering multipath channels, we do simulation with an AWGN channel to validate the simulation model. We validate this model by MSE and SER curves resulting from simulation. We simulate a system where FFT size = 1024, bandwidth = 10 MHz, TDD frame length = 5 ms, DL subframe size = 1 preamble + 24 OFDM symbols, and 12 clusters (forming 6 subchannels) are used in DL transmission.

In Figure 5.3, the theoretical symbol error rate (SER) curve versus E_s/N_0 for uncoded QPSK are plotted together with the SER curve resulting from the simulation. We can see LMMSE channel estimation performs better than advanced 4-point linear interpolation, and LMMSE channel estimation approaches the theory curve. Figure 5.3 shows the mean square error (MSE) curve resulting from the simulation versus E_s/N_0, and LMMSE performs better than advanced 4-point linear interpolation about 2 dB. Here, the E_s/N_0 means the data power divided by the noise power. For LMMSE channel estimation , because the noise power is measured by pilots, $\hat{\sigma}_n^2$ should be multiplied by a scale factor 9/16. In addition, σ_n^2 has to be multiplied by a scale factor 1/2 when using linear interpolation in time to produce two additional channel estimates in each cluster at the pilot positions of temporally adjacent OFDM symbols. In summary, if the pilot response is interpolated, the σ_n^2 becomes $1/2*9/16$; if not, the σ_n^2 becomes 9/16. When averaging the $\hat{\sigma}_n^2$ with each OFDM symbol, there are two cases. In one case, we just let $\hat{\sigma}_n^2$ be $9/16 \times \sigma_n^2$ for the first and the last symbol. In the other, we multiply by $[(1+1+1/2+1/2)/4] \times 9/16$ which is the case for the 2nd–23th symbols of a DL subframe frame. We can see LMMSE estimation maintains a close to -1 slope in MSE throughout the simulated range of SNR. Figure 5.4 shows the performance of LMMSE channel estimation before and after doing noise variance modification in AWGN channel. We can see the performance is similar between before and after doing noise variance modification, but there are different results in multipath channels.

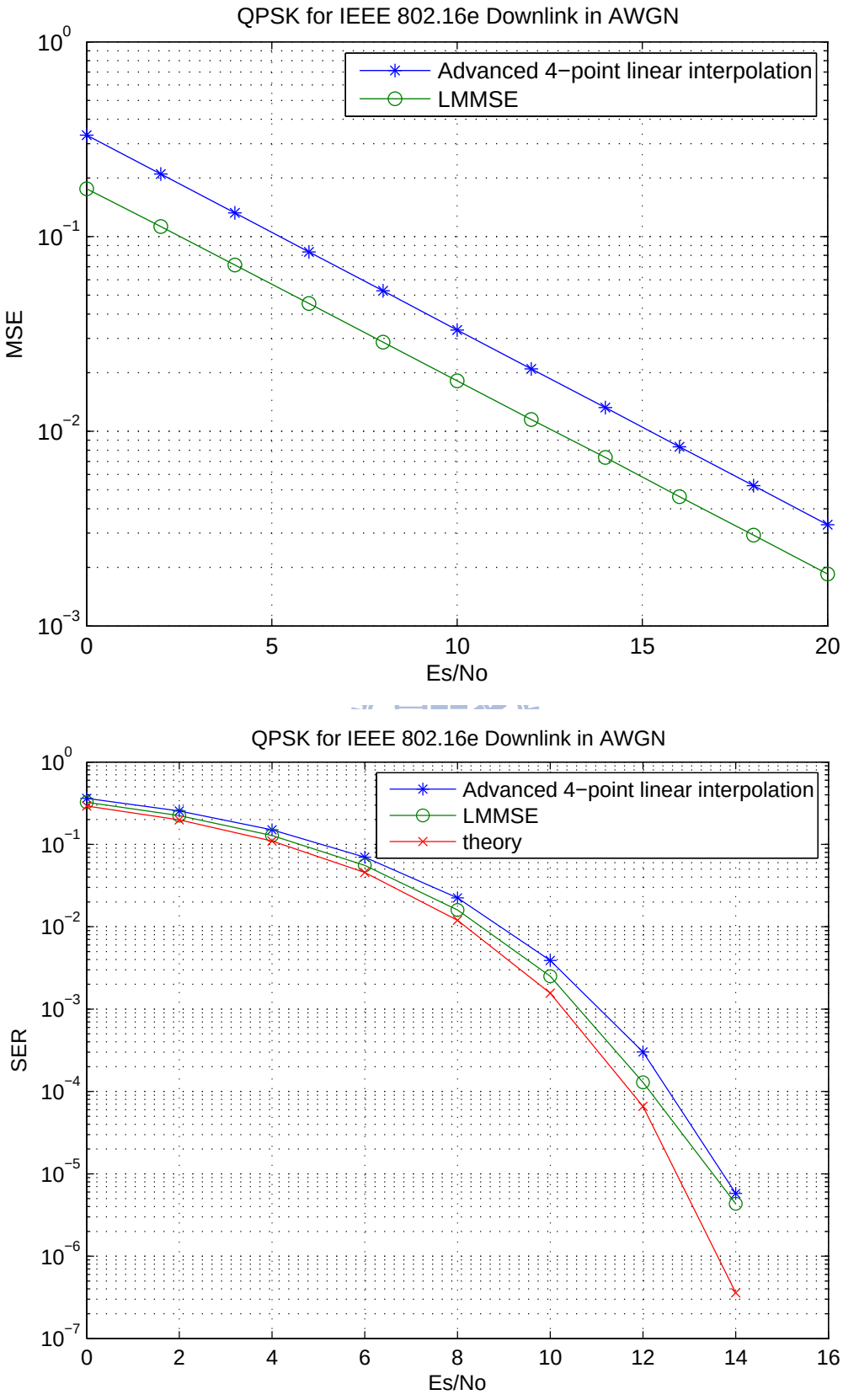


Figure 5.3: MSE and SER after channel estimation for QPSK in AWGN for IEEE 802.16e downlink.

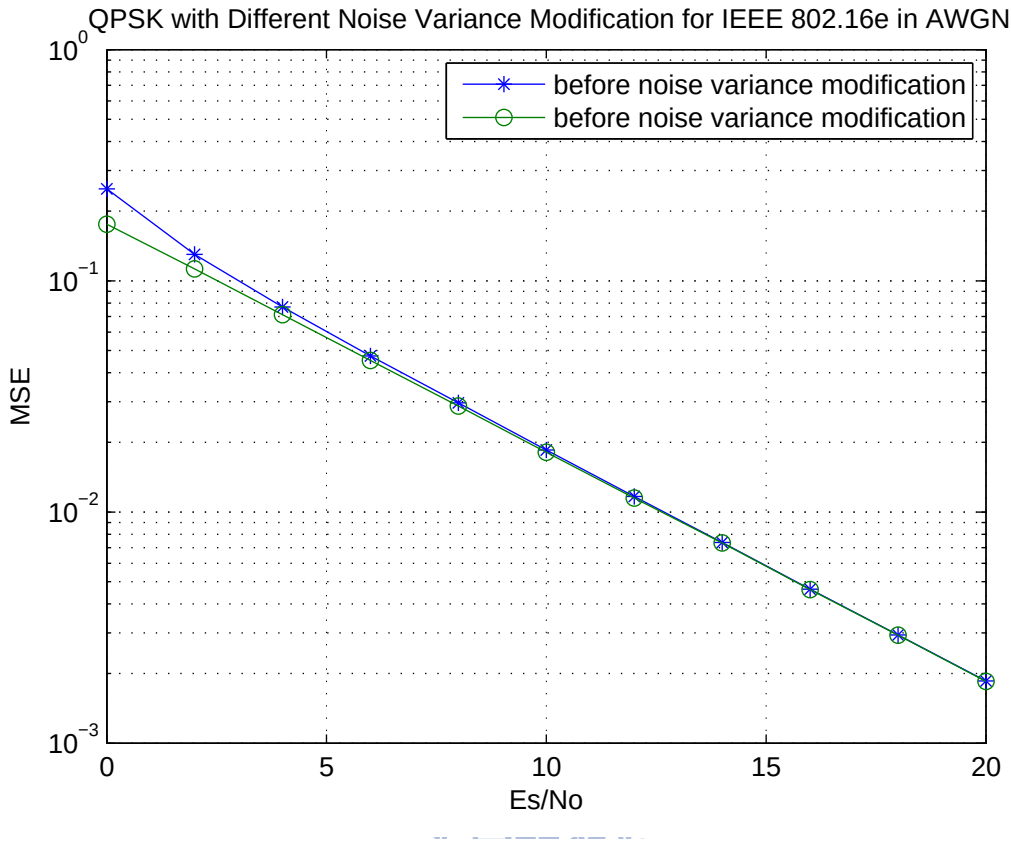


Figure 5.4: Channel estimation MSE and SER for QPSK with different noise variance modification in AWGN channel for IEEE 802.16e downlink.

5.3.3 Simulation Results for Multipath Channels

Figures 5.5–5.9 show the performance of different channel estimation methods with different velocities in SUI-2–SUI-6. We can see that LMMSE channel estimation performs better than advanced 4-point linear interpolation. For SUI-2 channel (type-C, for flat terrain, light tree, as introduced in suction 5.1.1), the delay spread is smaller, the SER performance is also more similar to AWGN channel. If the velocity becomes higher, the channel varies faster in time domain. Therefore, the performance with a higher velocity is worse. For SUI-5 and SUI-6 (type-A, for hilly terrain and heavy tree), the delay spread is larger than other channels, so the performance becomes worse. But the LMMSE channel estimation performs better than advanced 4-point linear interpolation significantly. That is because the LMMSE method does 4-point interpolation in the frequency domain, but the advanced 4-point linear interpolation only does 2-point interpolation. For SUI-3 and SUI-4 (type-B), the delay spread is between type-A and type-C, so the performance is also between in. Finally, Figures 5.10–5.11 show the performance of LMMSE channel estimation before and after doing noise variance modification in SUI-2 and SUI-4 channels. We can see the better performance after doing noise variance modification.

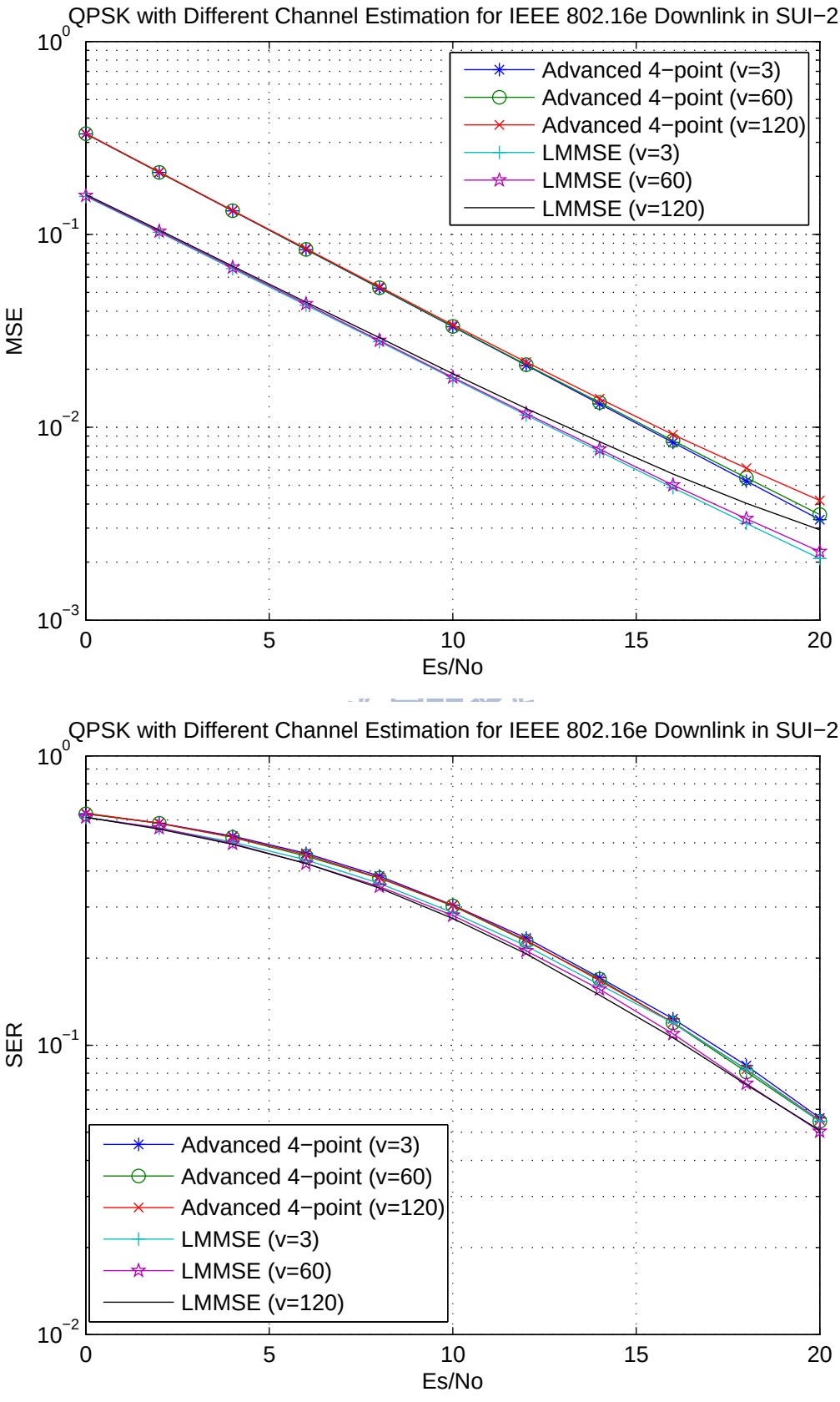
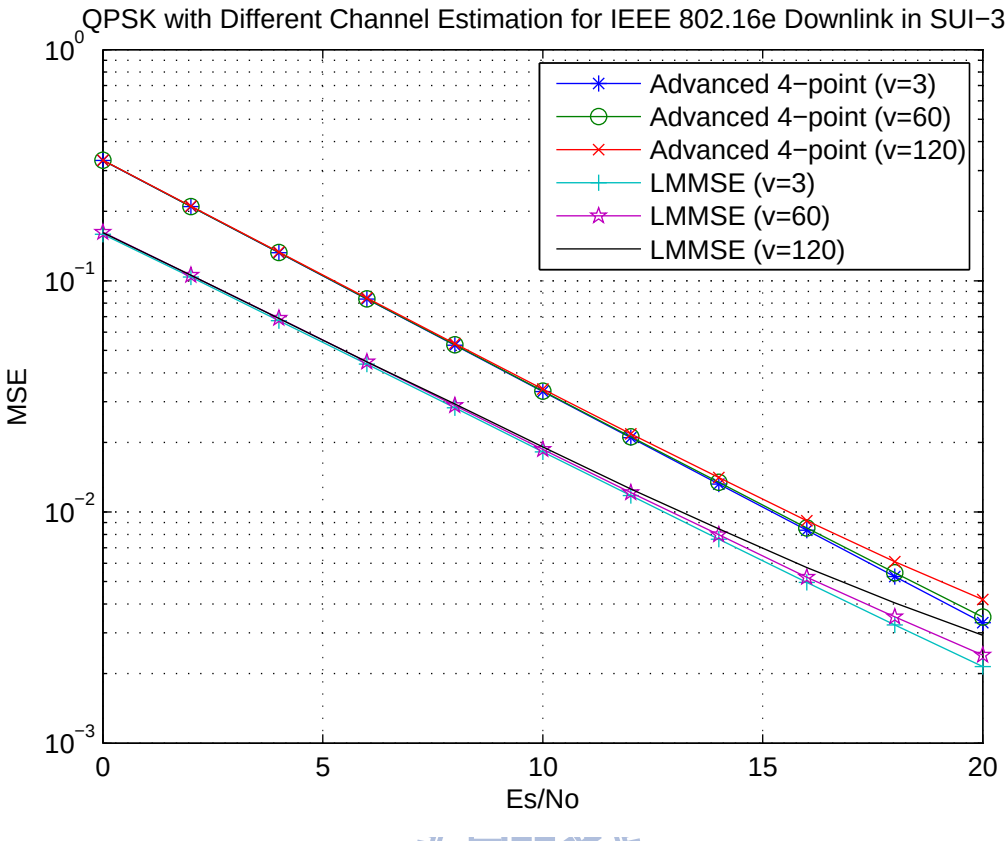


Figure 5.5: Channel estimation MSE and SER for QPSK at different velocities in SUI-2 channel for IEEE 802.16e downlink.



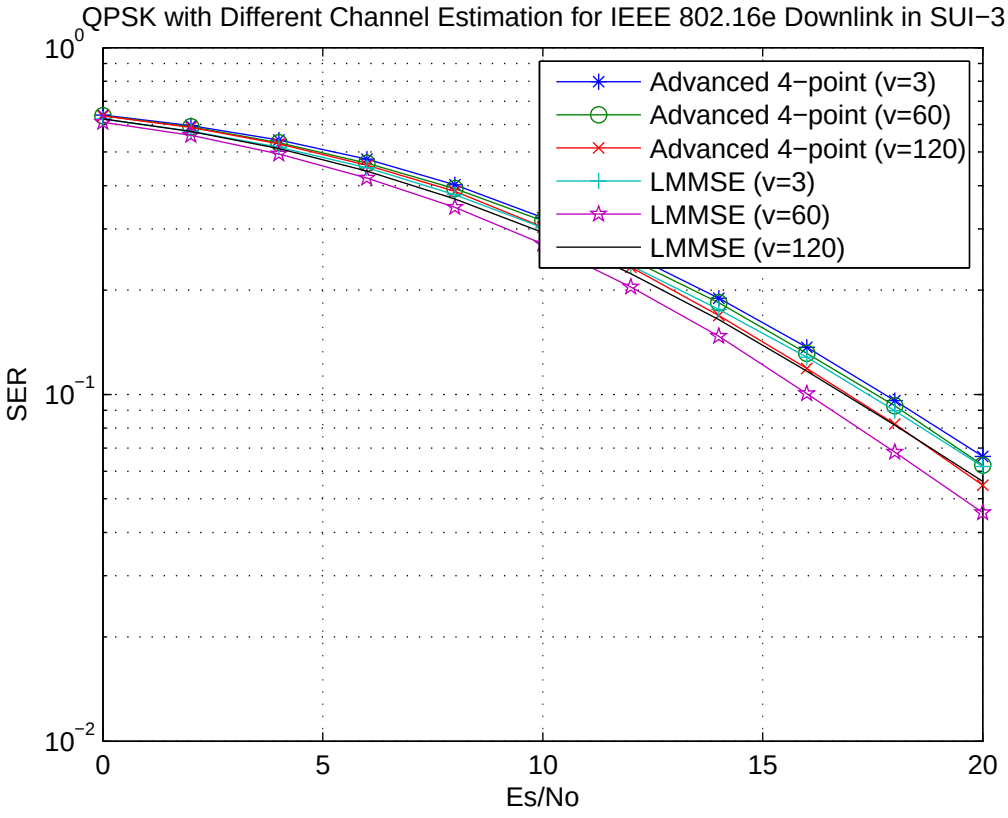
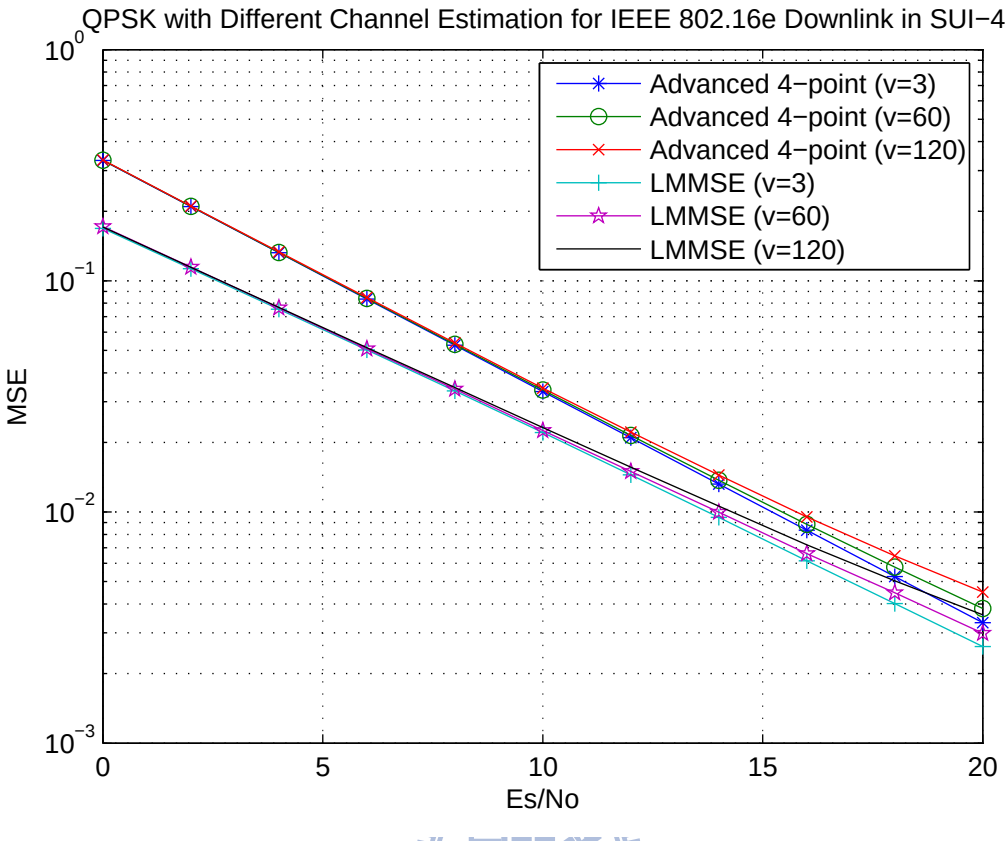


Figure 5.6: Channel estimation MSE and SER for QPSK at different velocities in SUI-3 channel for IEEE 802.16e downlink.



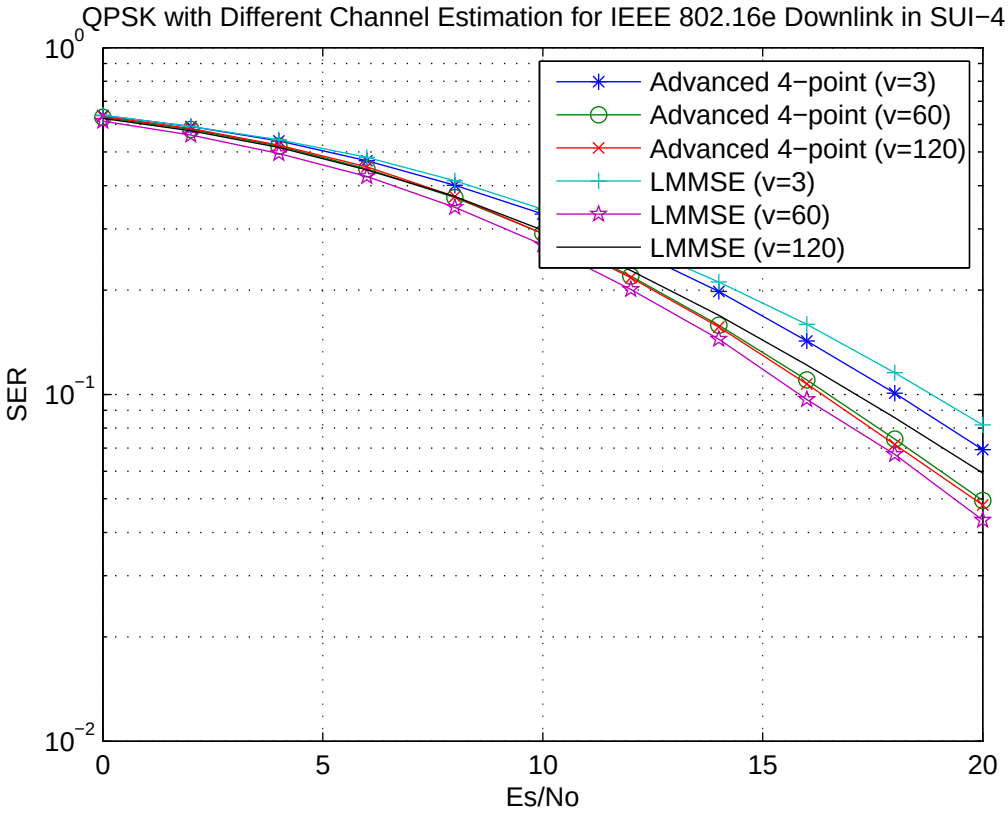
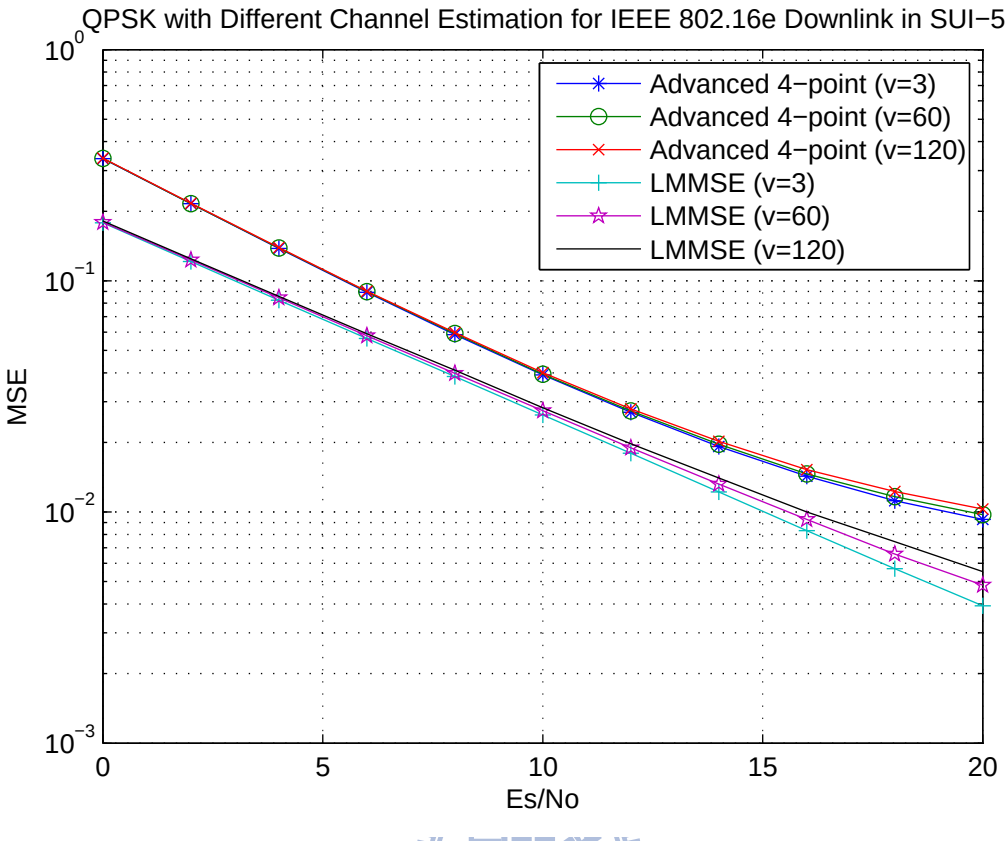


Figure 5.7: Channel estimation MSE and SER for QPSK at different velocities in SUI-4 channel for IEEE 802.16e downlink.



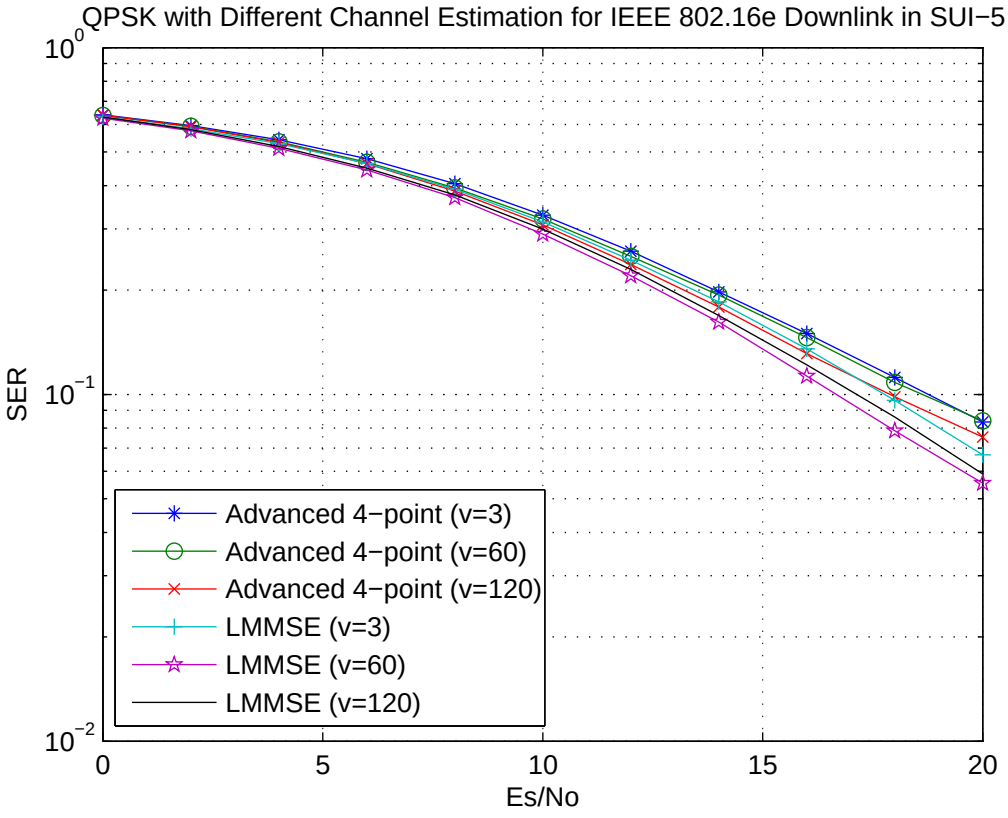
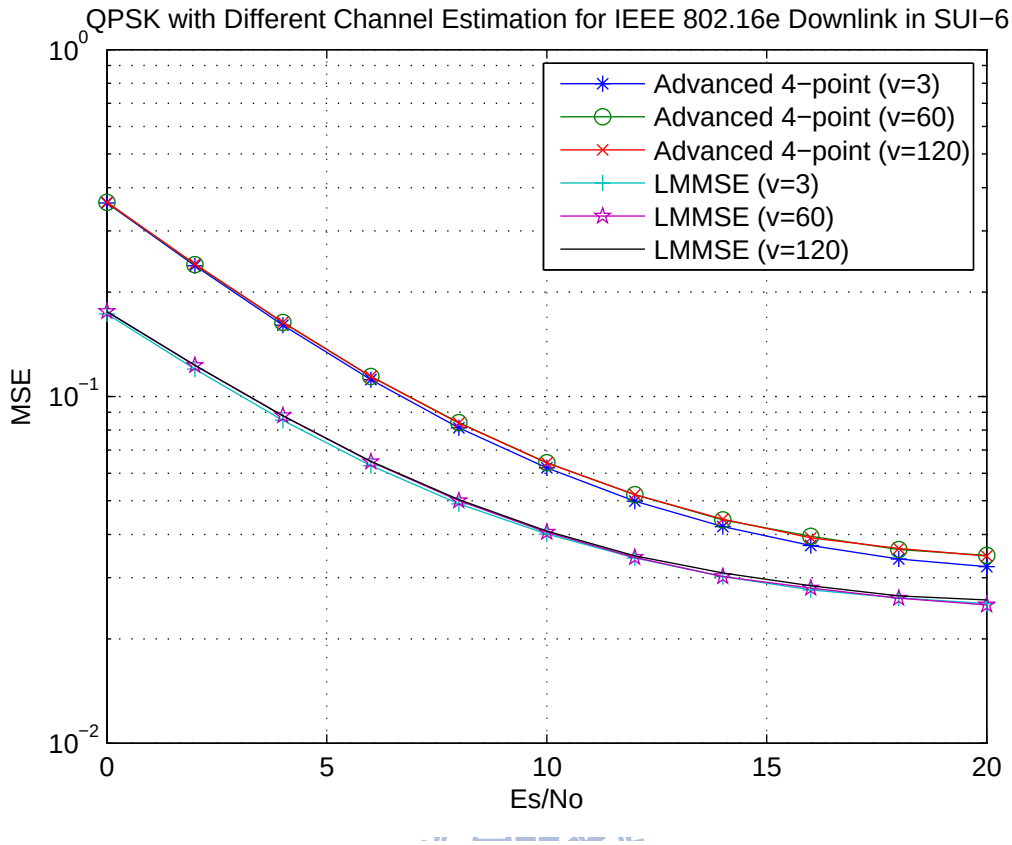


Figure 5.8: Channel estimation MSE and SER for QPSK at different velocities in SUI-5 channel for IEEE 802.16e downlink.



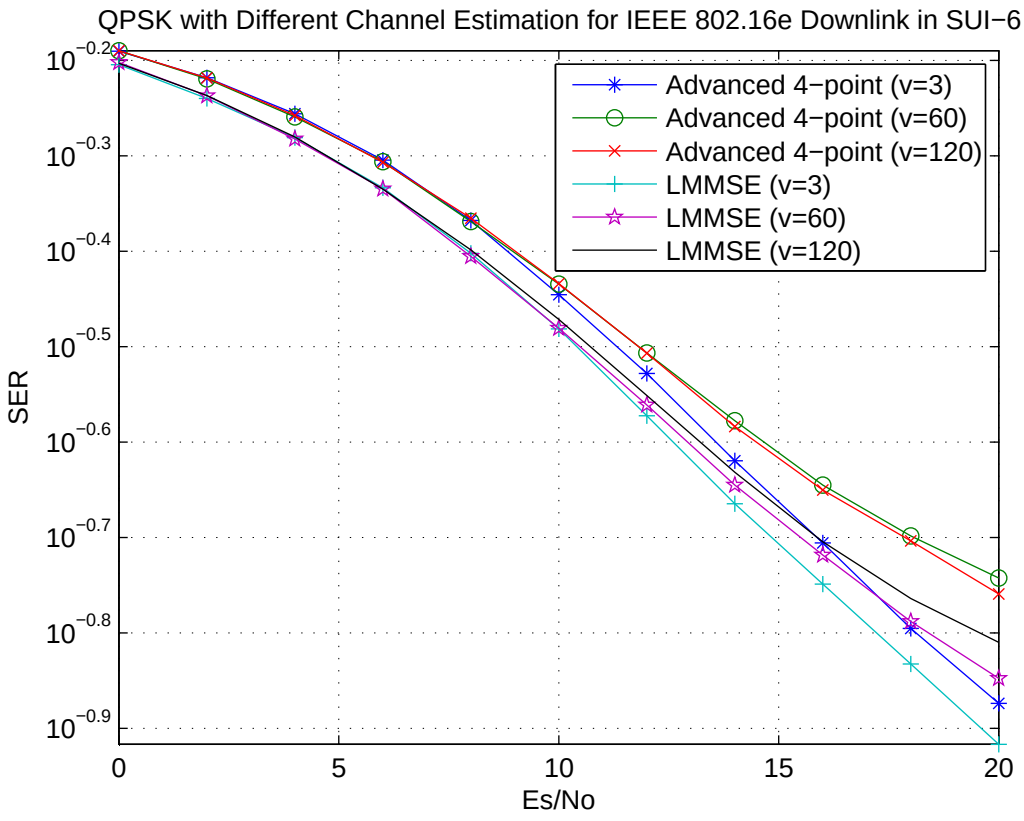
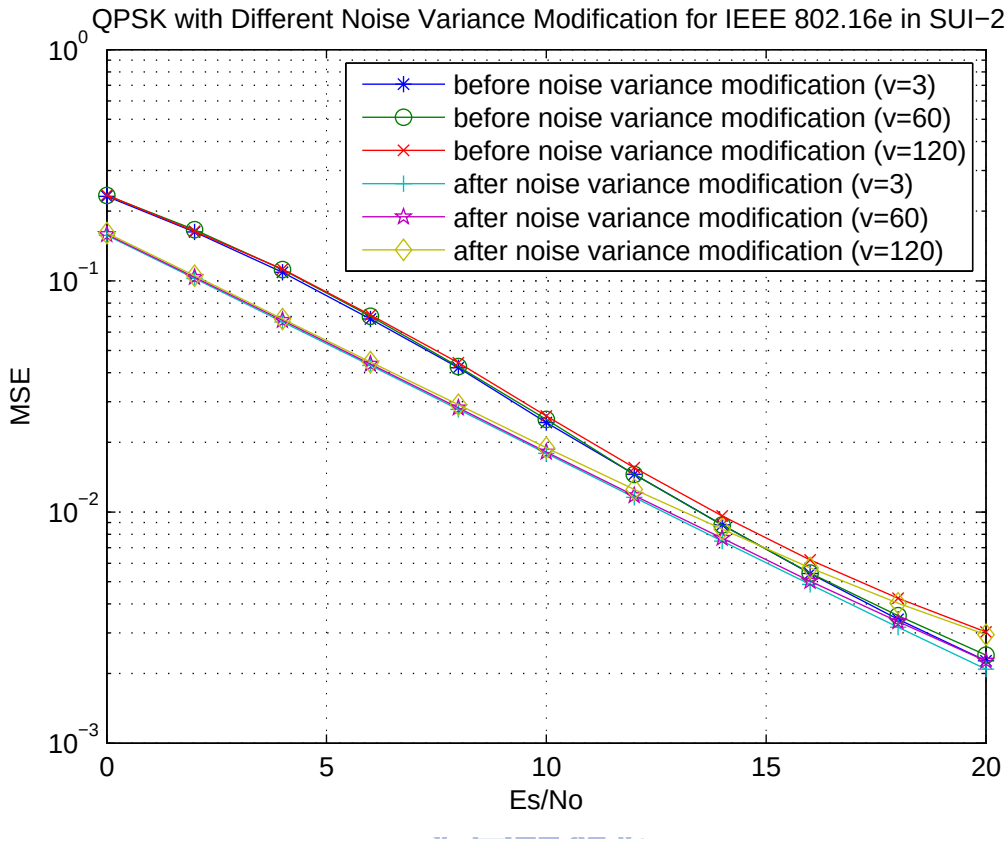


Figure 5.9: Channel estimation MSE and SER for QPSK at different velocities in SUI-6 channel for IEEE 802.16e downlink.



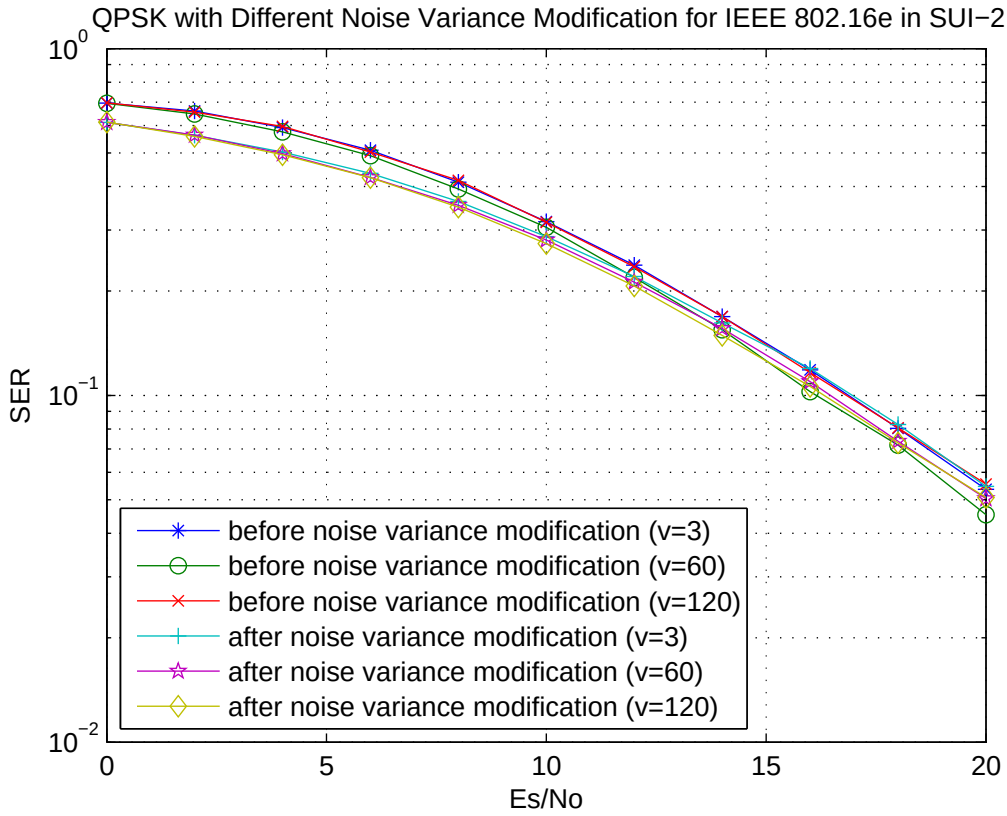


Figure 5.10: Channel estimation MSE and SER for QPSK with different noise variance modification in SUI-2 channel for IEEE 802.16e downlink.

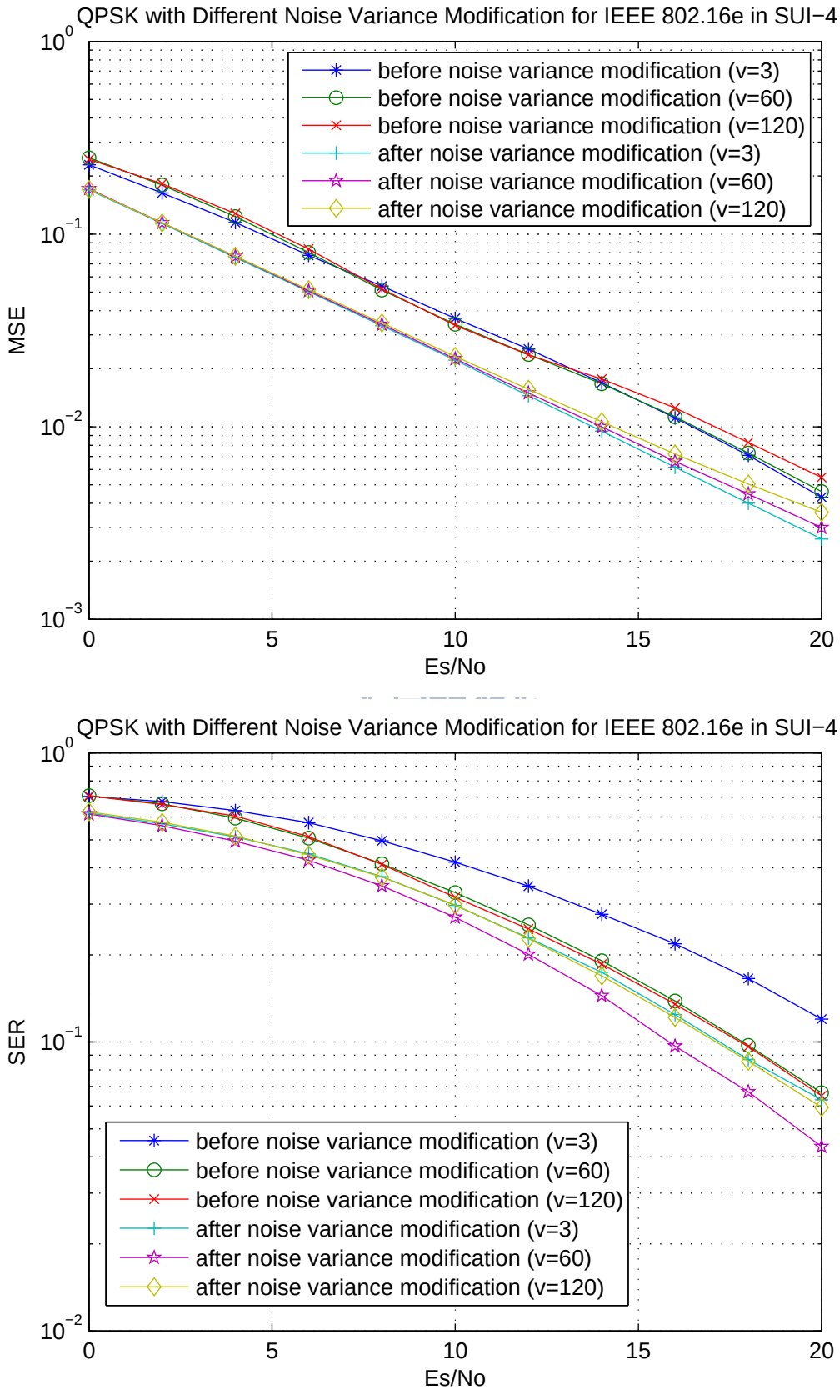


Figure 5.11: Channel estimation MSE and SER for QPSK with different noise variance modification in SUI-4 channel for IEEE 802.16e downlink.

Chapter 6

Downlink Channel Estimation Simulation for IEEE 802.16m

In this chapter, we use the LMMSE method to do the channel estimation in downlink transmission of IEEE 802.16m. We evaluate the performance mainly via mean square error (MSE) and symbol error rate (SER).

6.1 System Parameters and Channel Model

Table 6.1 gives the primitive and derived parameters used in our simulation work. In our system, we let the preamble be followed by 28 data symbols. In addition to AWGN, we use SUI-2 to SUI-6 and the Vehicular A to do simulation. The channel characteristics are decided in section 5.1.1. Their profiles are introduced in Tables 5.2–5.4.

6.2 LMMSE channel Estimation for IEEE 802.16m

The first symbol of a downlink subframe is preamble, and 28 data symbols follow. We design the LMMSE channel estimation according to the signal structure of IEEE 802.16m, and study the performance at different velocities in different channel models.

Table 6.1: OFDMA Downlink Parameters

Parameters	Values
Bandwidth	10 MHz
Central frequency	3.5 GHz
N_{used}	865
Sampling factor n	28/25
G	1/8
N_{FFT}	1024
Sampling frequency	11.2 MHz
Subcarrier spacing	10.94 kHz
Useful symbol time	91.43 μs
CP time	11.43 μs
OFDMA symbol time	102.86 μs
Sampling time	44.65 ns

6.2.1 Linear Interpolation in Time for IEEE 802.16m

The results discussed in the last chapter indicate that in the case of the IEEE 802.16m signal structure (shown in Fig. 6.1), LMMSE channel estimation should yield better performance than simple linear interpolation. However, because each OFDM symbol contains at most one pilot subcarrier in each PRU, we use time-domain linear interpolation to get several more pilot subcarrier channel estimates to facilitate the LMMSE channel estimation.

The detailed steps for each symbol are as follows:

1) First to third data symbols:

- Estimate the channel response at each pilot location using the LS technique.
- Take the pilots in the nearest next symbols as reference.

2) Fourth to $(n-3)$th data symbols:

- Estimate the channel response at each pilot location by using the LS technique.

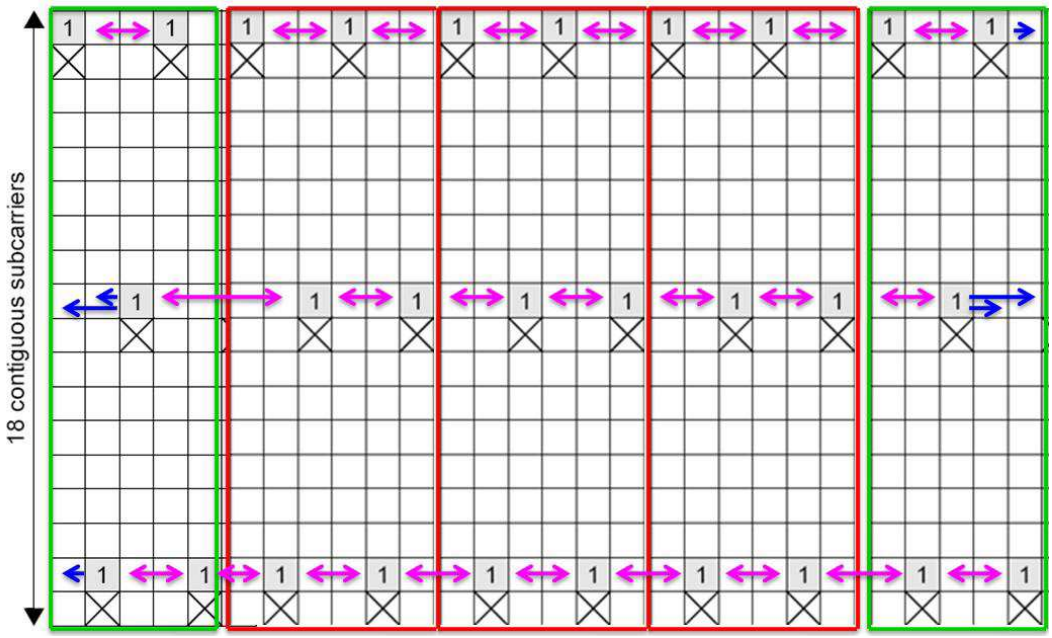


Figure 6.1: Linear interpolation in time domain.

- Use linear interpolation in time to estimate channel response at each pilot location of a nearest previous and a nearest next OFDM symbols. If there is no channel response at the pilot locations in previous symbol time to do linear interpolation in time, we copy the channel response to these pilot locations in previous time.

3) $(n-2)$th to the last data symbols:

- Estimate the channel response at each pilot location using the LS technique.
- Use the estimated channel responses at pilot locations of nearest previous symbols as reference.

This method gives our system a three-symbol time latency. For example, if we want to get the channel response of the first symbol, we must wait until we receive the fourth data symbol. In the last three symbols, we get not only the previous symbol information but also the channel response at each pilot location of the current symbol.

6.2.2 LMMSE Channel Estimation for IEEE 802.16m

The LMMSE channel estimation for IEEE 802.16m downlink proceeds in a way similar to that introduced in section 5.2.2 for IEEE 802.16e downlink, except that the time-domain interpolation is performed differently (as discussed in the last subsection) and that the noise variances in the autocorrelation matrix have different scaling factors. Because the signal structure for IEEE 802.16m is more complex than 802.16e, there are cases in scaling of the noise variances. In one case, there is one subcarrier response interpolated by two adjacent pilot responses in time. The $\hat{\sigma}_n^2$ at the subcarrier should be $1/2 \times \hat{\sigma}_n^2$. In a second case, there are two temporally consecutive subcarrier responses interpolated by one previous and one succeeding pilot responses in time. The $\hat{\sigma}_n^2$ at these two subcarrier locations is $5/9 \times \hat{\sigma}_n^2$. In the last case, there are four temporally consucutive subcarrier responses interpolated by one previous and one succeeding pilot responses in time. The $\hat{\sigma}_n^2$ at the first and the 4th subcarrier locations should be $17/25 \times \hat{\sigma}_n^2$, and at the 2nd and the 3rd subcarrier locations should be $13/25 \times \hat{\sigma}_n^2$. Moreover, in the calculation of $\hat{R}_1$, $\hat{\tau}_\mu$ and $\hat{\tau}_{rms}$, We have $F_s = 8$.

6.3 Simulation Results for IEEE 802.16m

6.3.1 Simulation Flow

Figure 5.2 illustrate the block diagram of our simulated system. We also assume perfect synchronization and omit it in our simultion. For data transmission simulation, we don't do FEC and DeFEC. After channel estimation, we calculate the channel MSE between the true channel and the estimated one, where the average is taken over all the subcarriers. The symbol error rate (SER) can also be obtained after demapping.

6.3.2 Validation with AWGN Channel

Before considering multipath channels, we again do simulation with an AWGN channel to validate the simulation model. We validate this model by MSE and SER curves resulting from simulation. We simulate a system where FFT size = 1024, bandwidth = 10 MHz, TDD frame length = 5 ms, DL subframe size = 1 preamble + 28 OFDM symbols, and 48 PRUs are used in DL transmission.

In Figure 6.2, the theoretical symbol error rate (SER) curve versus E_s/N_0 for uncoded QPSK are plotted together with the SER curve resulting from the simulation. We can see the SER curve of the LMMSE channel estimation approaches the theory curve. Figure 6.2 shows the mean square error (MSE) curve resulting from the simulatin versus E_s/N_0, and we can see LMMSE estimation maintains a close to -1 slope in MSE throughout the simulated range of SNR. Here, the E_s/N_0 means the data power divided by the noise power. For LMMSE channel estimation, because the noise power is measured by pilots, $\hat{\sigma}_n^2$ should be multiplied by a scale factor 9/16. In addition, $\hat{\sigma}_n^2$ has to be multiplied by a scale factor if the corresponding pilot subcarrier response estimate is obtained by linear interpolation in times discussed in the last section. Figure 6.3 shows the performance of LMMSE channel estimation before and after doing noise variance modification in AWGN channel. We can see the performance is similar between before and after doing noise variance modification, but there are different results in multipath channels. In summary, the average noise variance for each of the 28 data symbols of the DL subframe is as follows.

- 1st symbol: $(9/16 + 9/16 + 9/16)/3 \times \hat{\sigma}_n^2$,
- 2nd symbol: $(1 + 1 + 5/9)/3 \times 9/16 \times \hat{\sigma}_n^2$,
- 3rd symbol: $(1 + 5/9 + 5/9)/3 \times 9/16 \times \hat{\sigma}_n^2$,

- 4th symbol: $(2 + 5/9)/3 \times 9/16 \times \hat{\sigma}_n^2$,
- 5th symbol: $(13/25 + 1/2 + 1)/3 \times 9/16 \times \hat{\sigma}_n^2$,
- 6th symbol: $(13/25 + 1/2 + 1)/3 \times 9/16 \times \hat{\sigma}_n^2$,
- 7th symbol: $(5/9 + 1 + 17/25)/3 \times 9/16 \times \hat{\sigma}_n^2$,
- 8th symbol: $(1 + 5/9 + 5/9)/3 \times 9/16 \times \hat{\sigma}_n^2$,
- 9th to 25th symbols: $(1 + 5/9 + 5/9)/3 \times 9/16 \times \hat{\sigma}_n^2$,
- 26th symbol: $(1 + 5/9 + 5/9)/3 \times 9/16 \times \hat{\sigma}_n^2$,
- 27th symbol: $(1 + 1 + 5/9)/3 \times 9/16 \times \hat{\sigma}_n^2$,and
- 28th symbol: $(9/16 + 9/16 + 9/16)/3 \times \hat{\sigma}_n^2$.

This validates the simulation (we use C/C++ programming language).

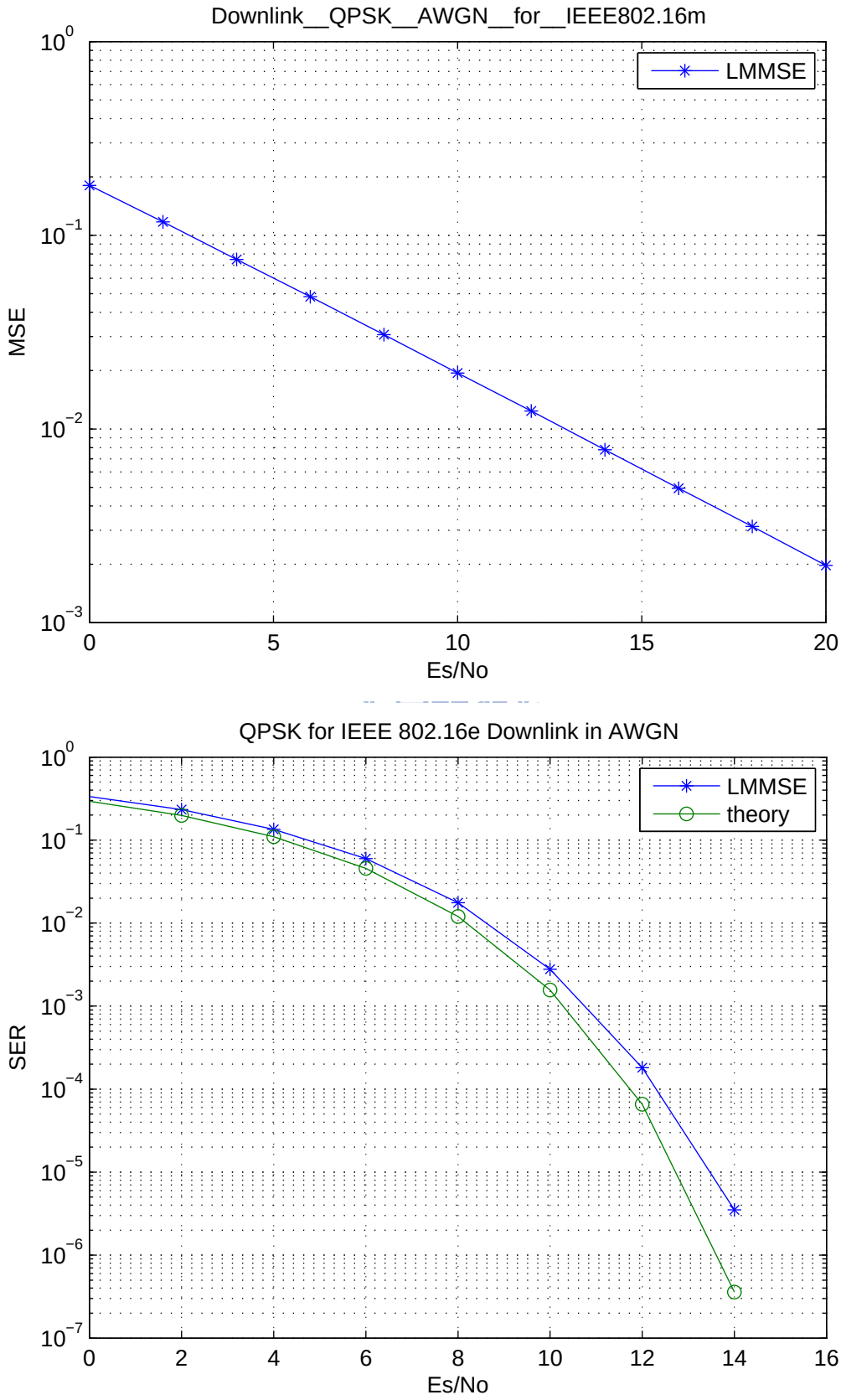


Figure 6.2: MSE and SER after channel estimation for QPSK in AWGN for IEEE 802.16m downlink downlink.

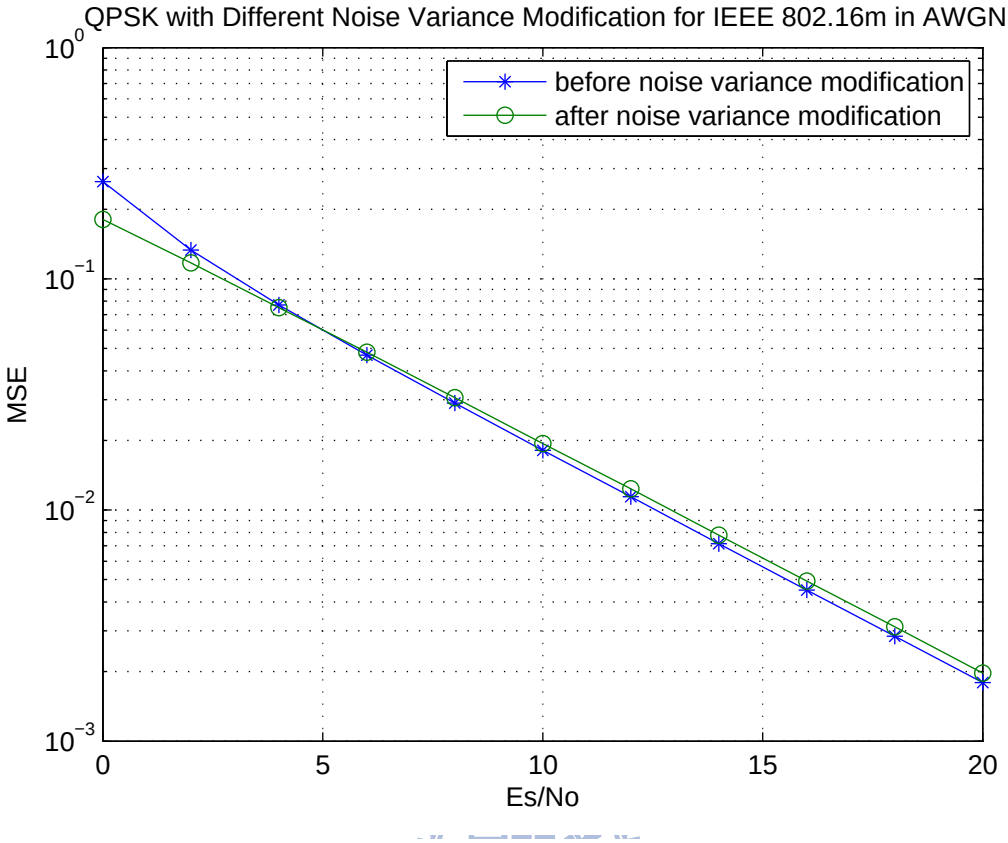


Figure 6.3: Channel estimation MSE and SER for QPSK with different noise variance modification in AWGN channel for IEEE 802.16m downlink.

6.3.3 Simulation Results for Multipath Channels

Figures 6.4–6.5 show the distributions of the estimated mean delays and RMS delay spreads at different SNR values in AWGN. Figures 6.6–6.13 show the distributions of the estimated mean delays and RMS delay spreads at different velocities and SNR values in SUI-2 and SUI-5 channel. Figures 6.14 shows the performance of LMMSE channel estimation with different velocities in SUI-2. We can see that the MSE and SER performances are relatively close at different speeds. This is because the SUI-2 channel model has a relatively small delay spread (type C). Figures 6.15–6.16 show the performance of LMMSE channel estimation at different velocities in SUI-5 and SUI-6 channels. We can see worse performance compared to SUI-2, because SUI-5 and SUI-6 are have greater delay spreads (type A). Figures 6.17–6.18 show the performance of LMMSE channel estimation at different velocities in SUI-3 and SUI-4. The delay spreads of SUI-3 and SUI-4 are between type A and C. The performance in SUI-3 and SUI-4 is also in between. Finally, Figures 6.19–6.20 show the performance of LMMSE channel estimation before and after doing noise variance modification in SUI-2 and SUI-4 channels. We can see the better performance after doing noise variance modification.

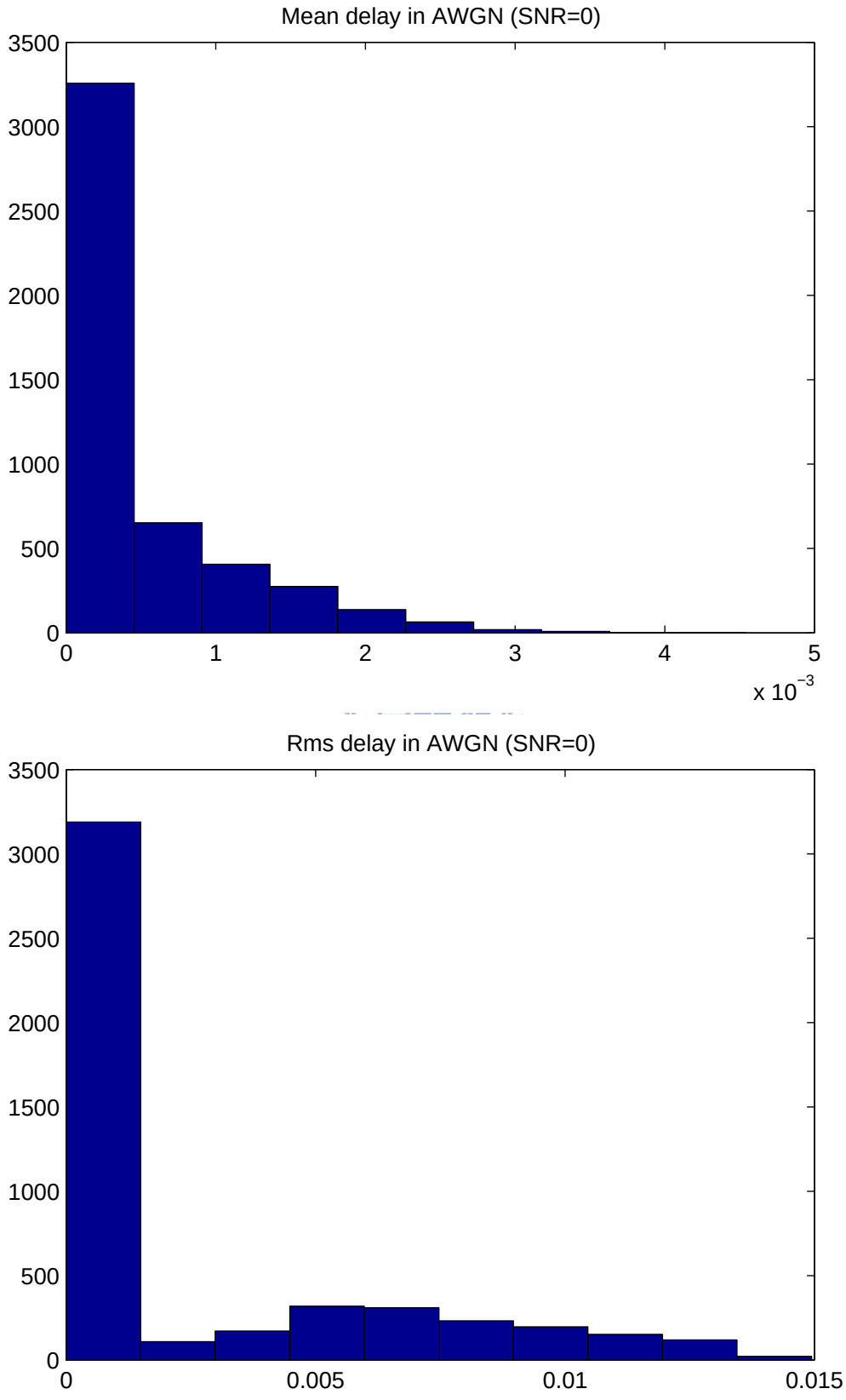


Figure 6.4: Distributions of estimated mean delays and RMS delay spreads at SNR = 0 dB in AWGN.

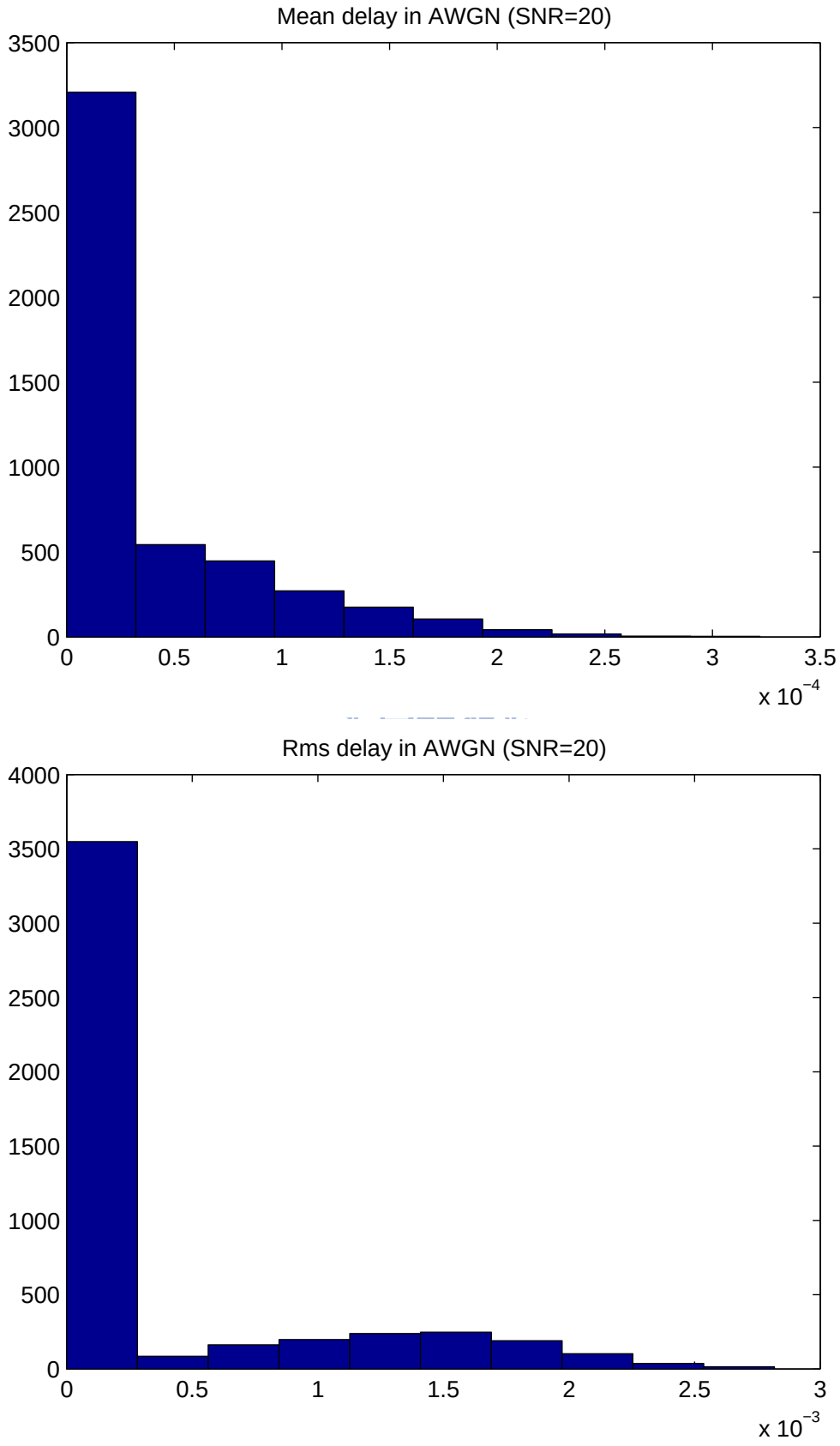


Figure 6.5: Distributions of estimated mean delays and RMS delay spreads at SNR = 20 dB in AWGN.

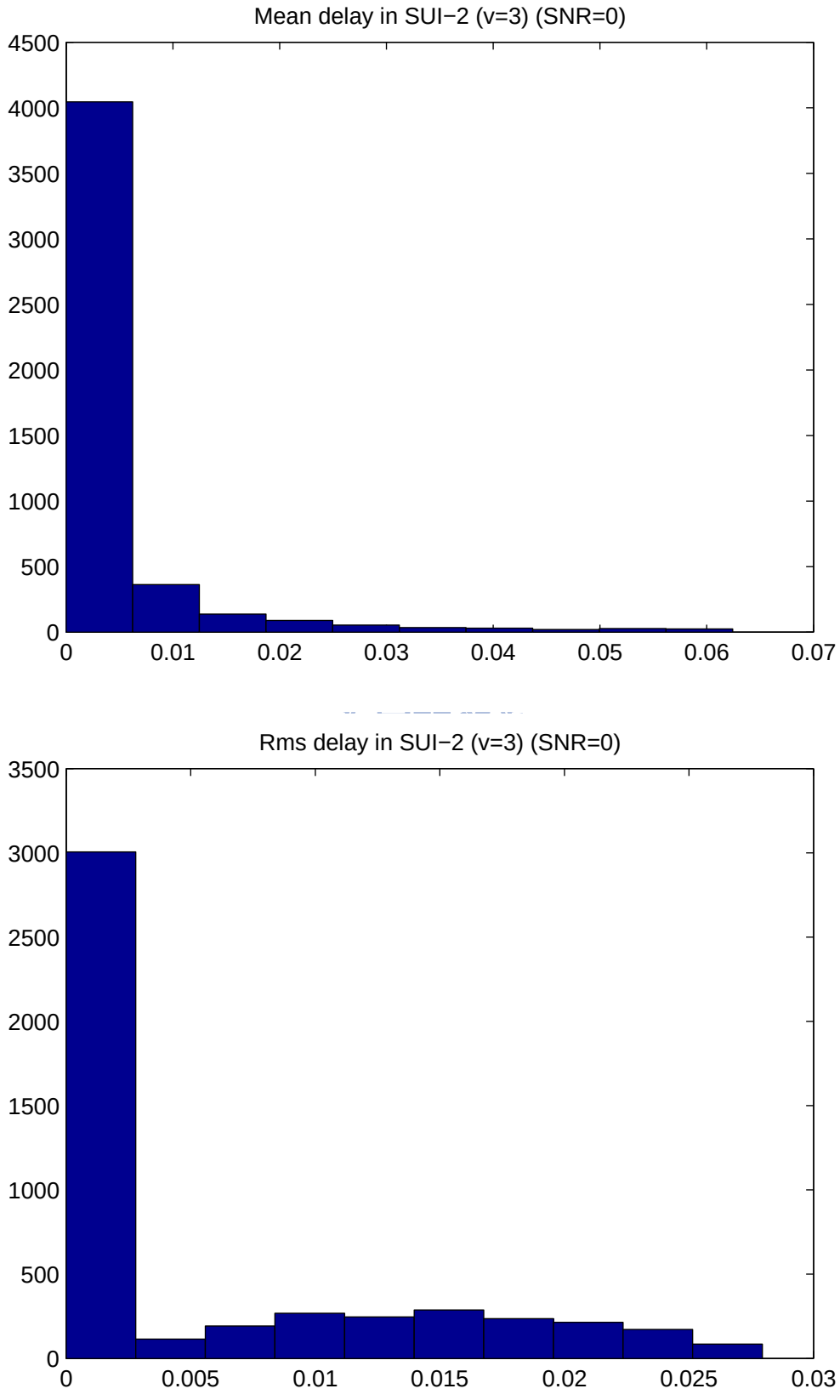


Figure 6.6: Distributions of estimated mean delays and RMS delay spreads at velocity = 3 km/hr and SNR = 0 dB in SUI-2.

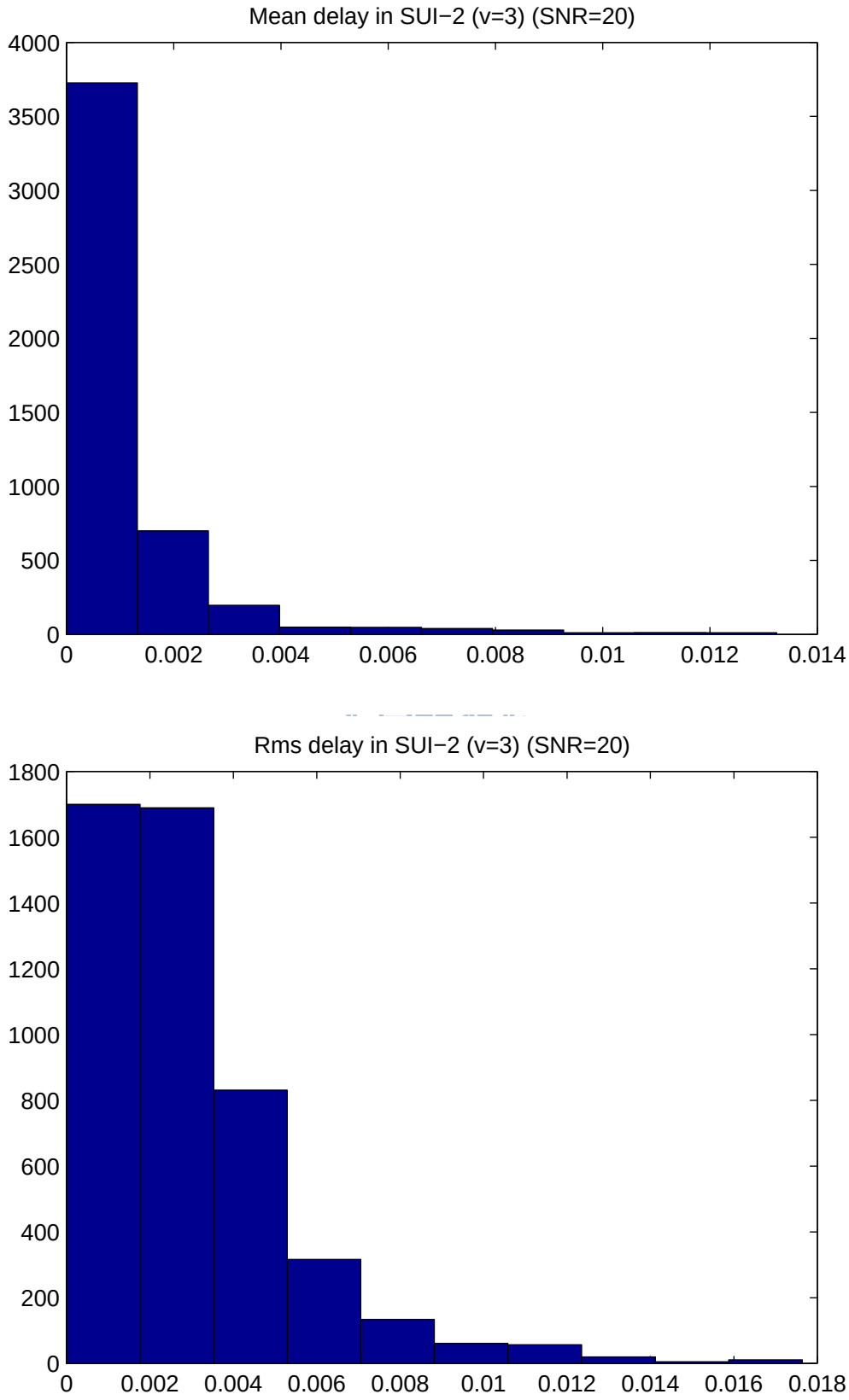


Figure 6.7: Distributions of estimated mean delays and RMS delay spreads at velocity = 3 km/hr and SNR = 20 dB in SUI-2.

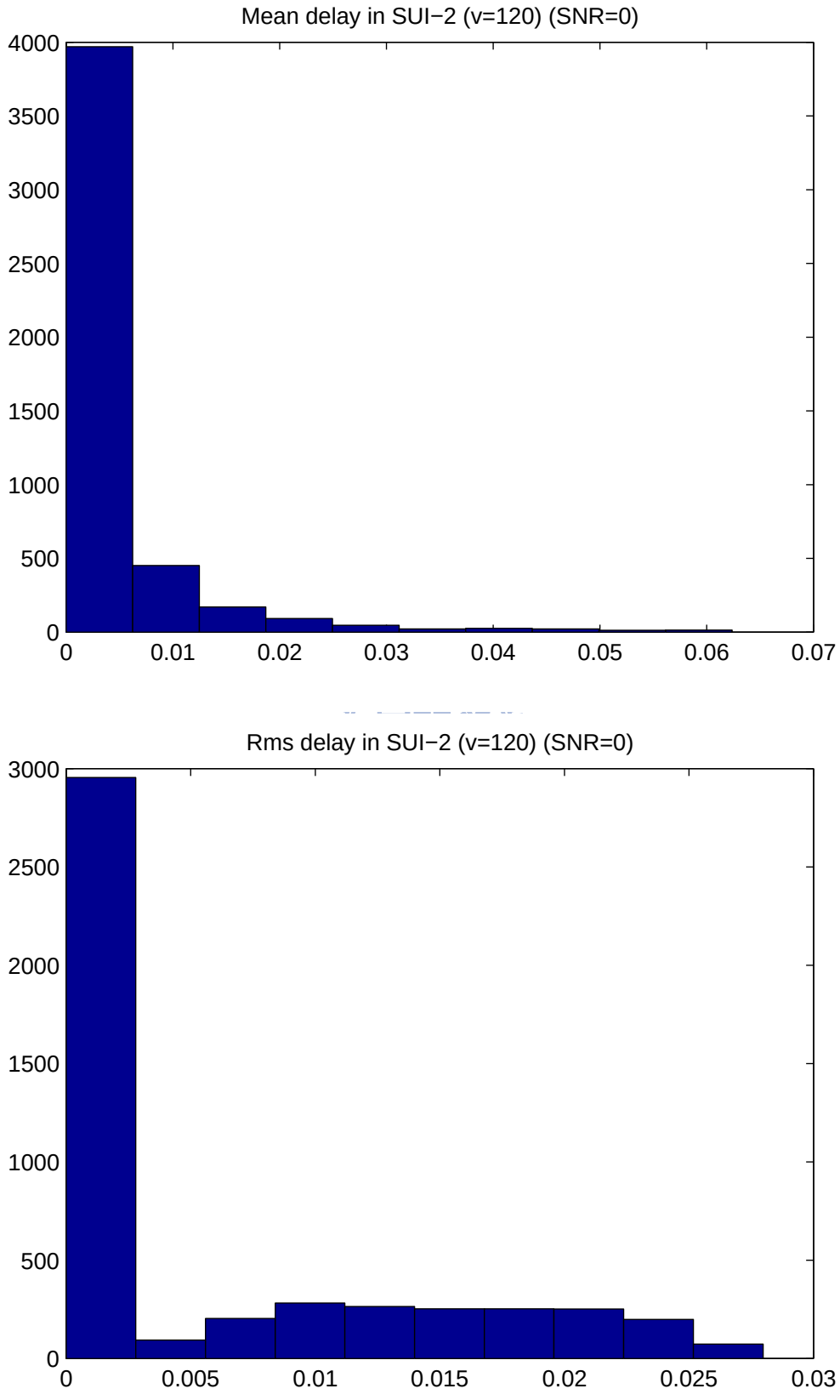


Figure 6.8: Distributions of estimated mean delays and RMS delay spreads at velocity = 120 km/hr and SNR = 0 dB in SUI-2.

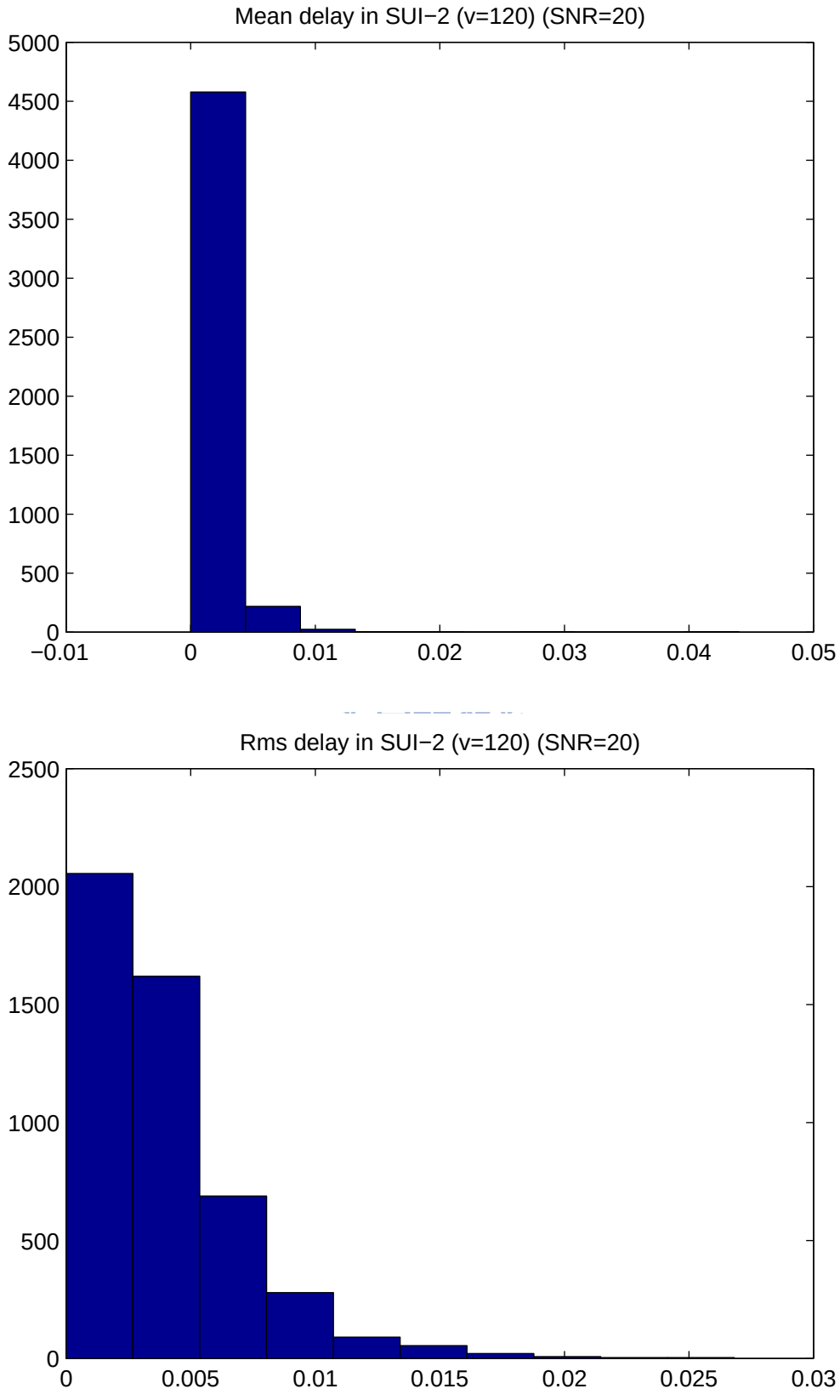


Figure 6.9: Distributions of estimated mean delays and RMS delay spreads at velocity = 120 km/hr and SNR = 10 dB in SUI-2.

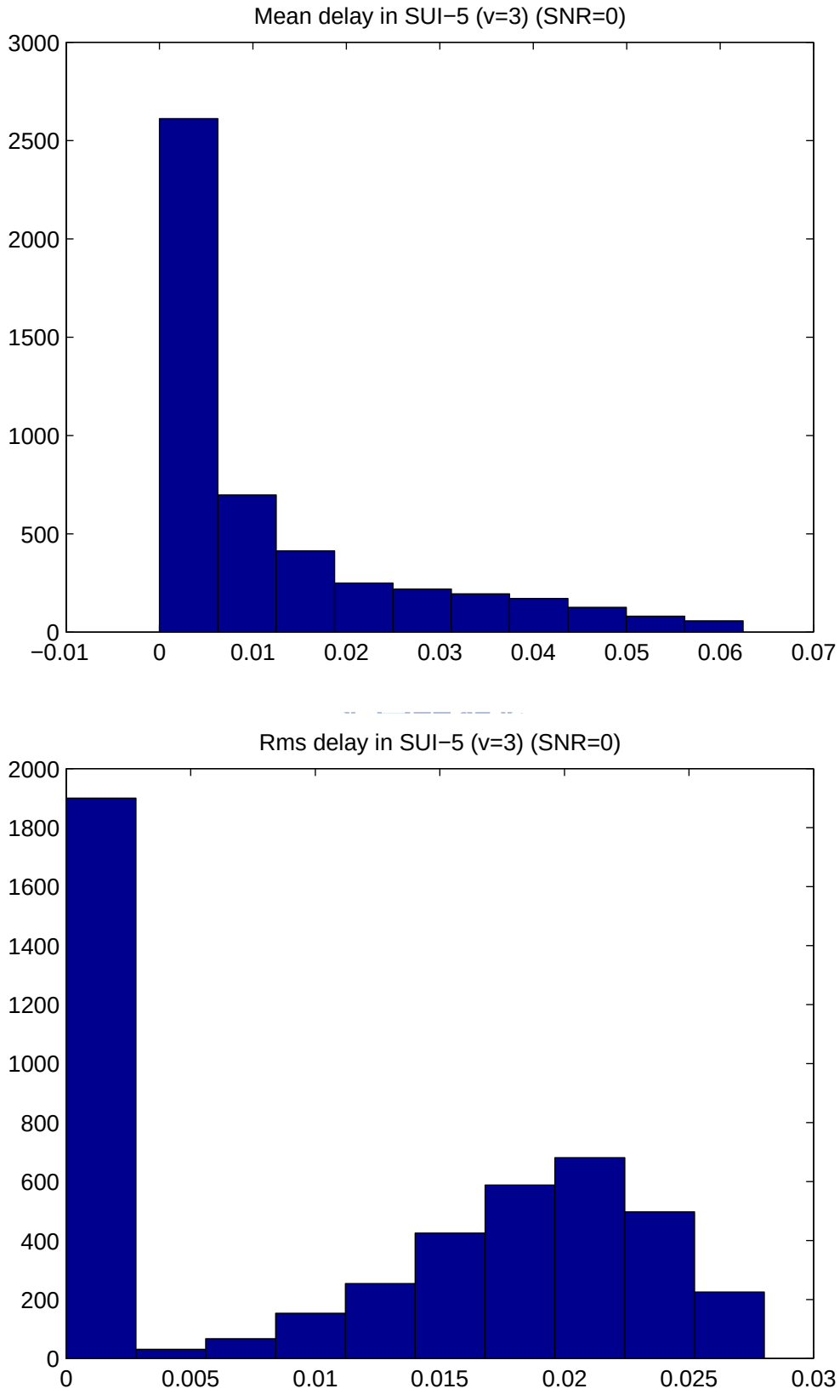


Figure 6.10: Distributions of estimated mean delays and RMS delay spreads at velocity = 3 km/hr and SNR = 0 dB in SUI-5.

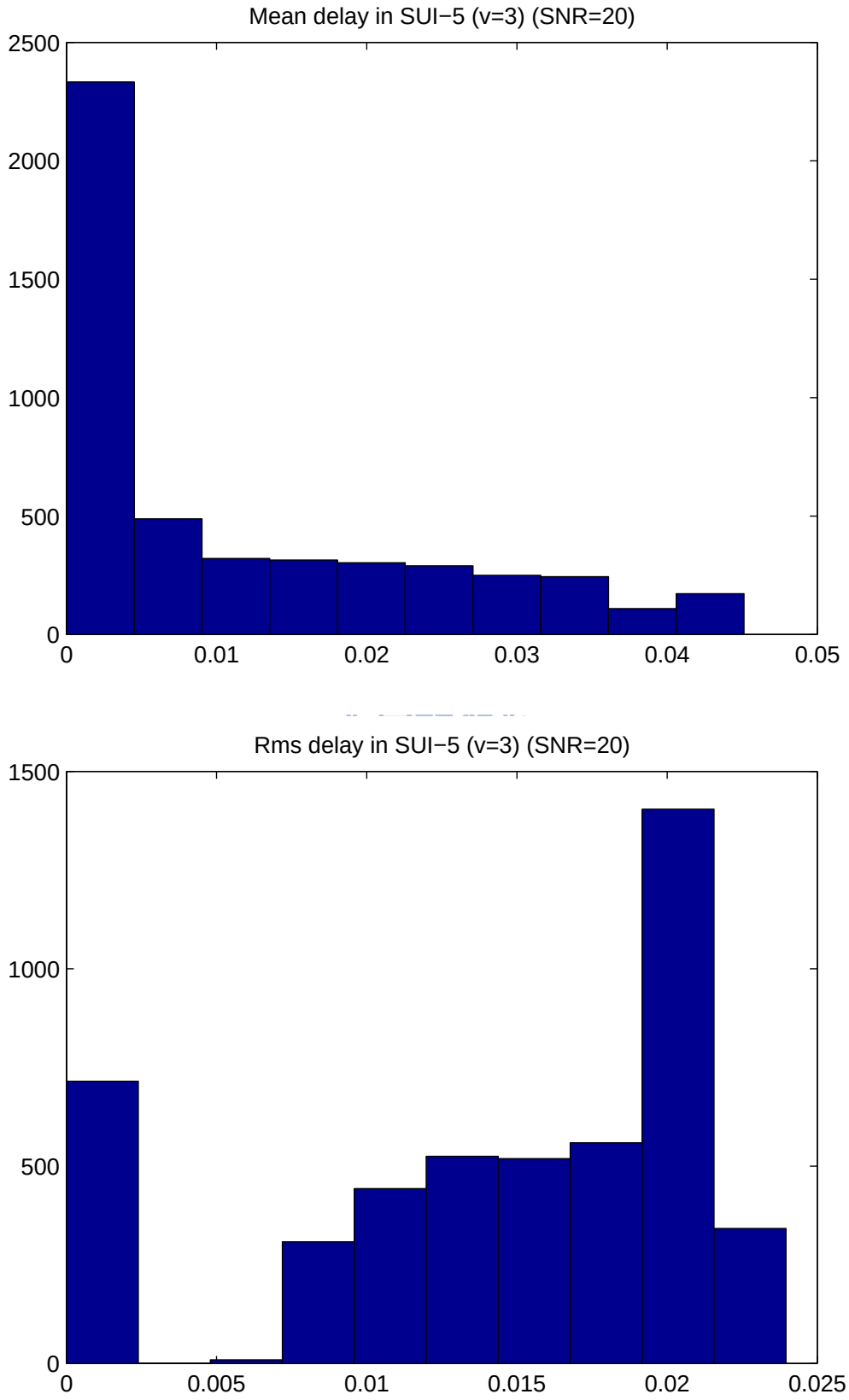


Figure 6.11: Distributions of estimated mean delays and RMS delay spreads at velocity = 3 km/hr and SNR = 20 dB in SUI-5.

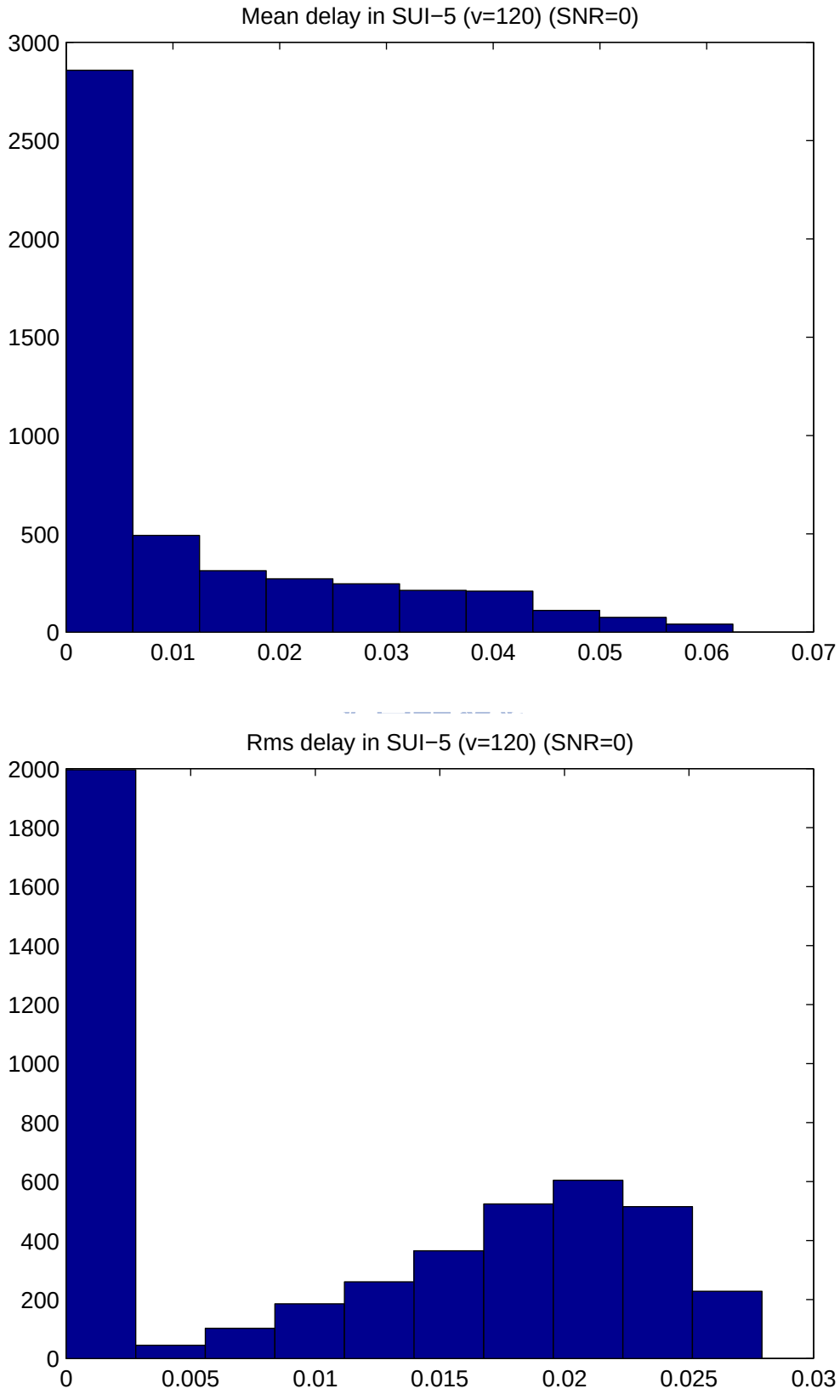


Figure 6.12: Distributions of estimated mean delays and RMS delay spreads at velocity = 120 km/hr and SNR = 0 dB in SUI-5.

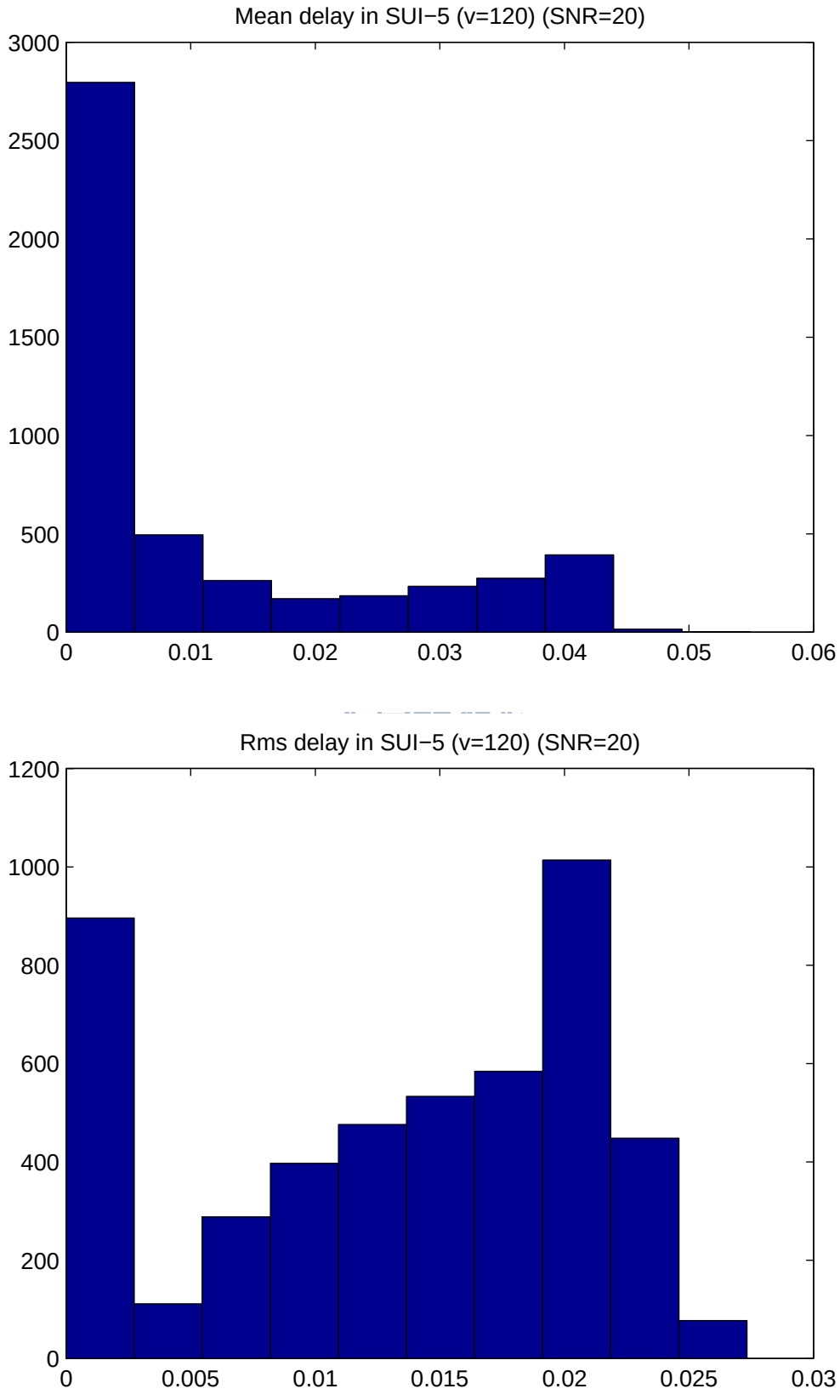


Figure 6.13: Distributions of estimated mean delays and RMS delay spreads at velocity = 120 km/hr and SNR = 20 dB in SUI-5.

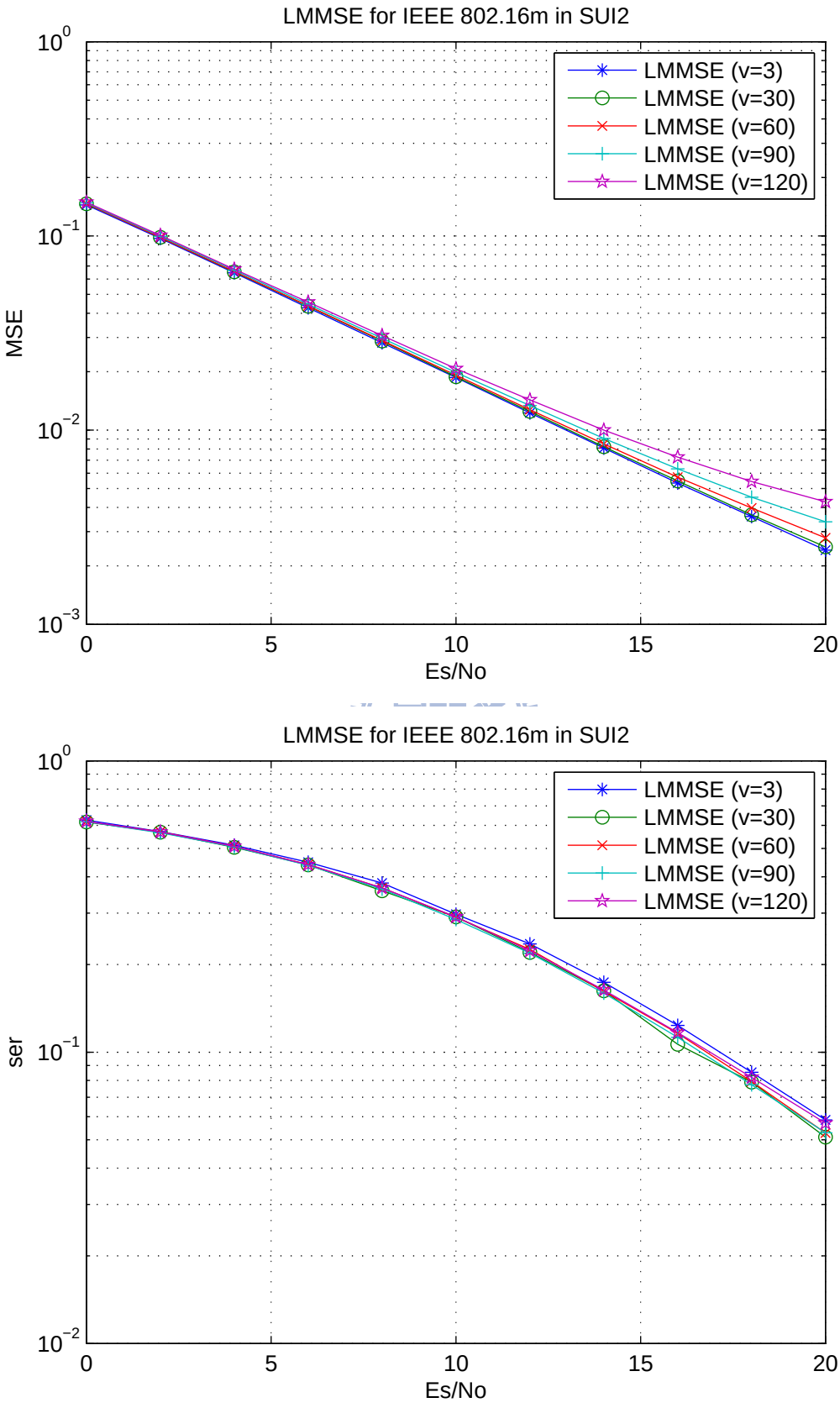


Figure 6.14: Channel estimation MSE and SER for QPSK at different velocities in SUI-2 channel for IEEE 802.16m downlink.

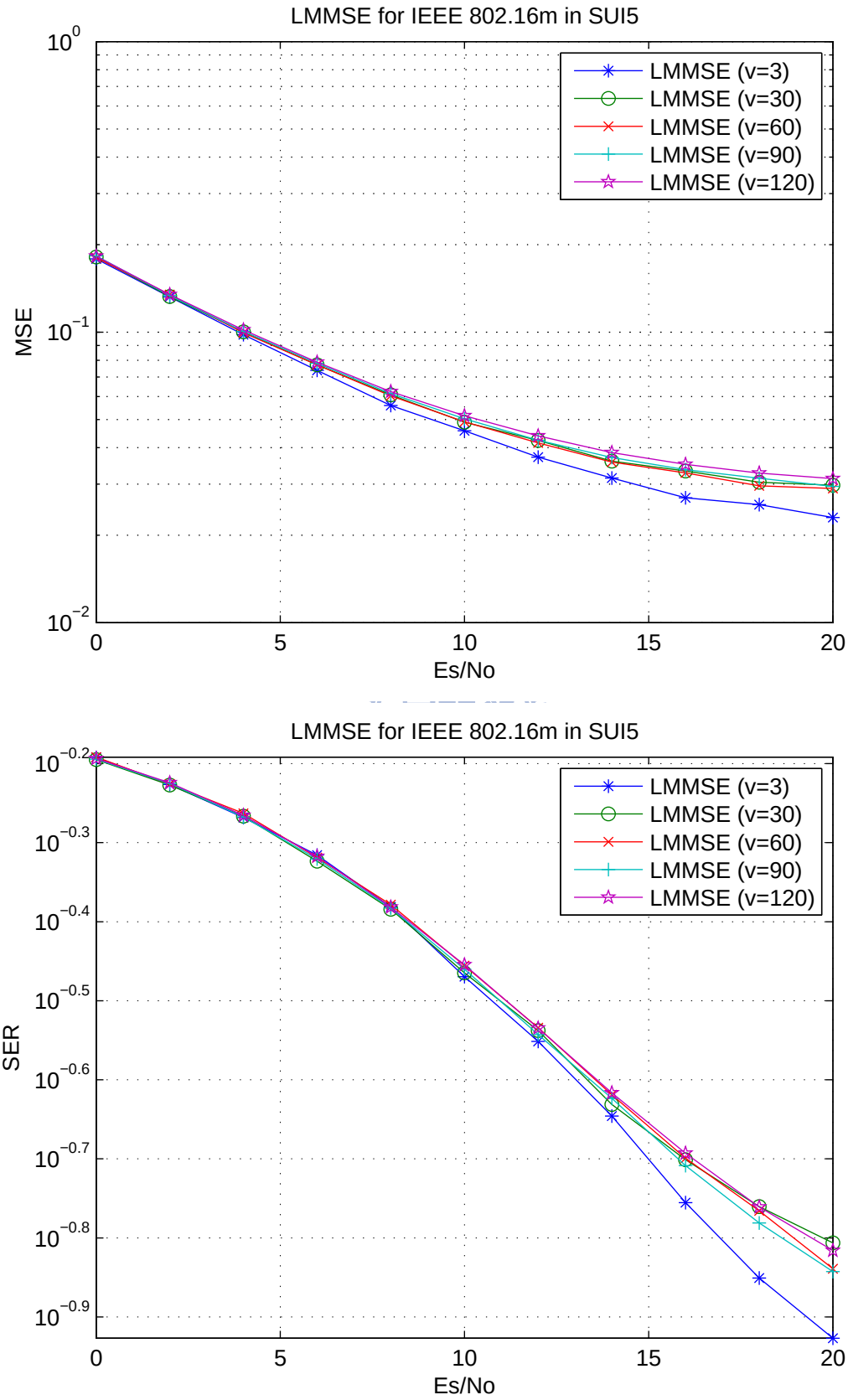


Figure 6.15: Channel estimation MSE and SER for QPSK at different velocities in SUI-5 channel for IEEE 802.16m downlink.

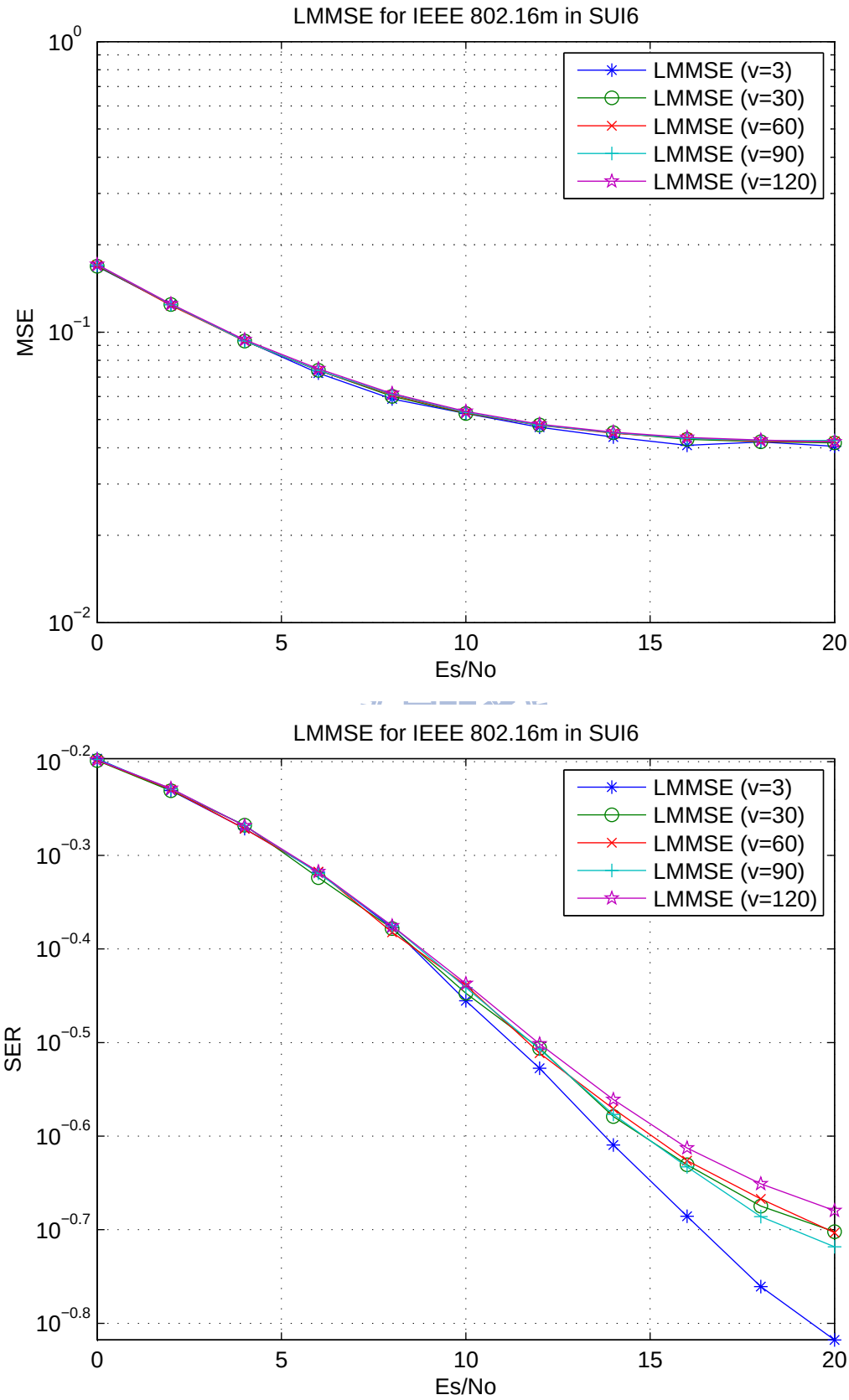


Figure 6.16: Channel estimation MSE and SER for QPSK at different velocities in SUI-6 channel for IEEE 802.16m downlink.

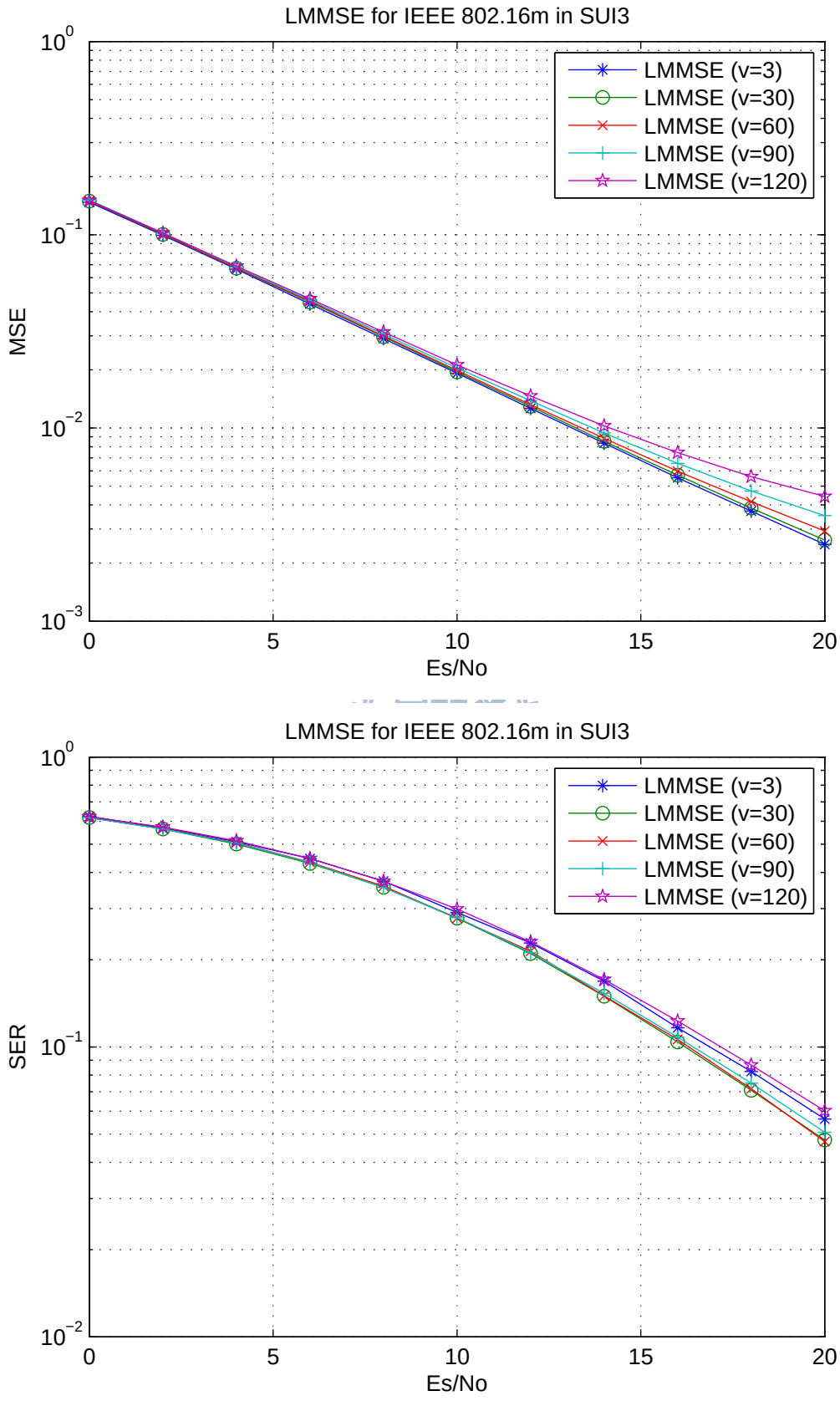


Figure 6.17: Channel estimation MSE and SER for QPSK at different velocities in SUI-3 channel for IEEE 802.16m downlink.

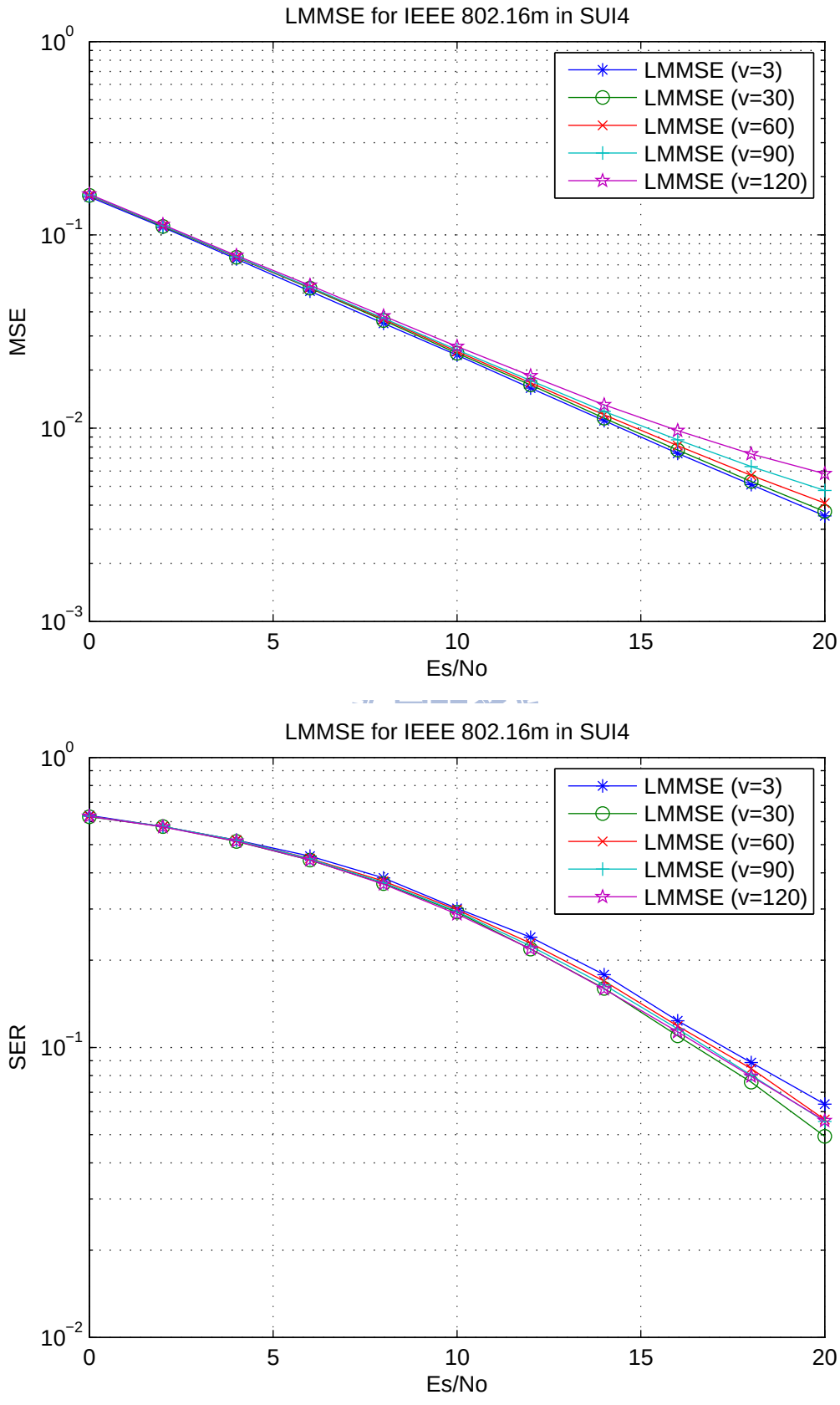
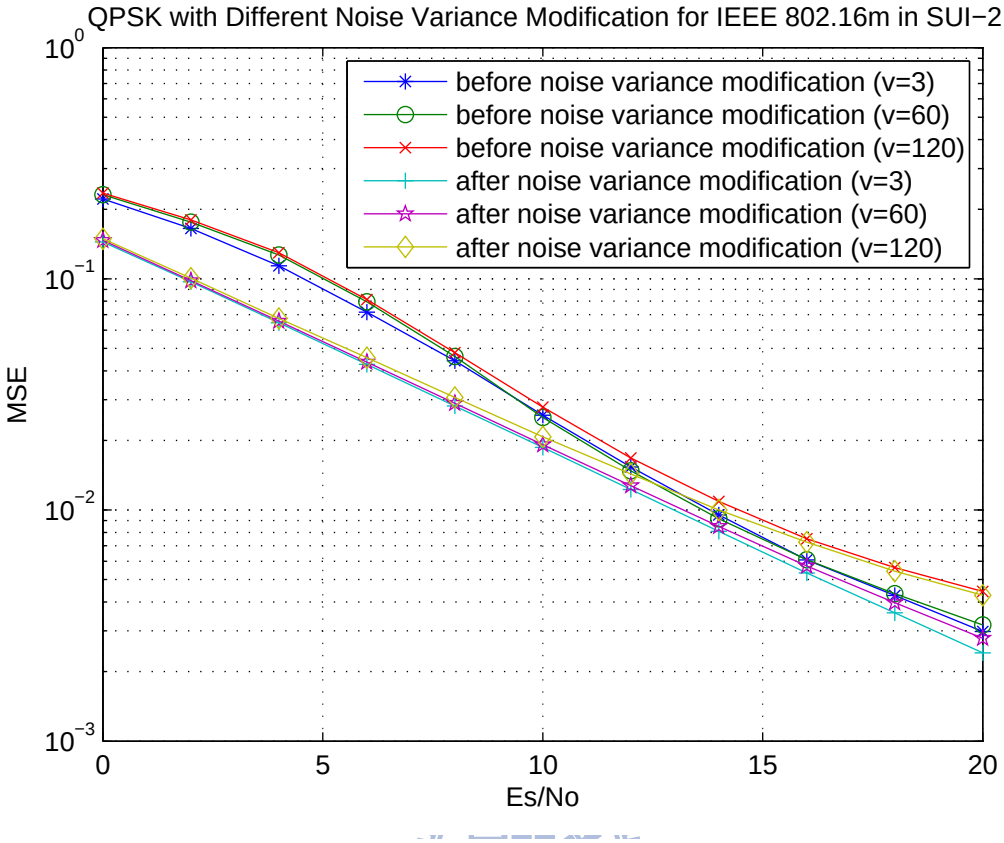


Figure 6.18: Channel estimation MSE and SER for QPSK at different velocities in SUI-4 channel for IEEE 802.16m downlink.



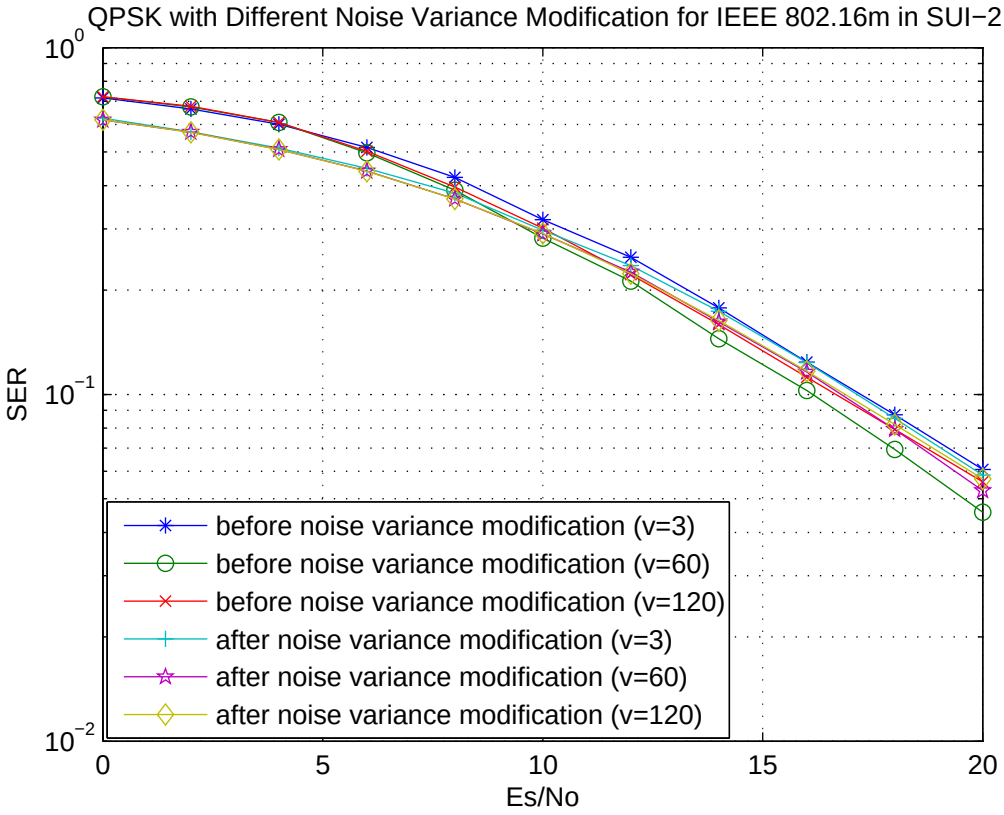


Figure 6.19: Channel estimation MSE and SER for QPSK with different noise variance modification in SUI-2 channel for IEEE 802.16m downlink.

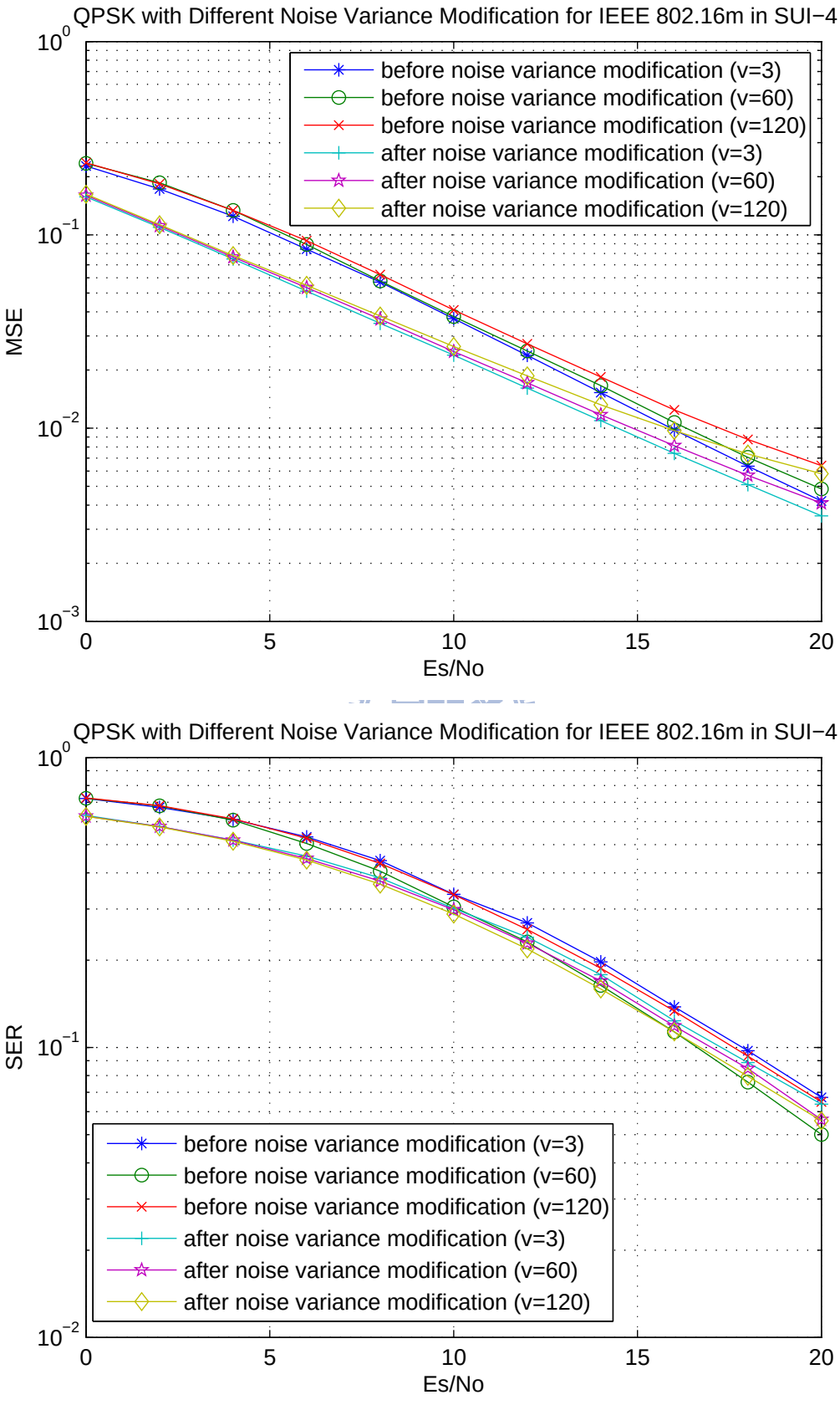


Figure 6.20: Channel estimation MSE and SER for QPSK with different noise variance modification in SUI-4 channel for IEEE 802.16m downlink.

Chapter 7

Conclusion and Future Work

7.1 Conclusion

In this thesis, we presented several channel estimation methods for OFDMA downlink for IEEE 802.16e and 802.16m. There are the steps to do the advanced 4-point linear interpolation:

- First, use LS estimator to estimate the channel frequency response on the pilot subcarriers.
- Second, interpolate the channel response in the middle of the pilot location in time domain.
- Third, interpolate the data subcarrier response in frequency domain.

There are the steps to do the LMMSE channel estimation:

- First, perform LS estimation of the pilot subcarrier responses
- Second, estimate the rms delay and mean delay
- Third, find the autocorrelation function associated with the approximate PDP

- Fourth, base on this autocorrelation function, do LMMSE filtering to estimate the data subcarrier responses

For IEEE 802.16e, we compared the performance of the two channel estimation methods, and the LMMSE channel estimation performed better than advanced 4-point linear interpolation. For IEEE 802.16m, we presented the LMMSE method to do channel estimation, and we got good performance with different channel models.

7.2 Potential Future Work

There are several possible extensions for our research:

- enhance performance in the long delay spread channels
- Add MIMO to the system for IEEE 802.16m.
- Optimize the performance on DSP.
- Try other kinds of techniques to estimate channel response on pilots for less estimated errors.
- In this thesis, we do not consider the influence of intercarrier interference. The simulation can be involved in the future.

Bibliography

[1] P. Hoeher, S. Kaiser, and P. Roberson, "Two-dimensional pilot-symbol-aided channel estimation by Wiener filtering," *in Proc. IEEE Int. Conf Acoust. Speech Signal Processing*, vol. 3, 1997, pp. 1845–1848.

[2] M. Morelli and U. Mengali, "A comparsion of pilot-aided channel estimation methods for OFDM systems," *IEEE Trans. Signal Processing*, vol. 49, no. 12, pp. 3065–3073, Jan. 1998.

[3] O. Edfors, M. Sandell, J.-J. van de Beek, S. K. Wilson, and P. O. Borjesson, "OFDM channel estimation by singular value decomposition," *IEEE Trans. Commun.*, vol. 46, no. 7, pp.931–939, July 1998.

[4] Y. Li, "Pilot-symbol-aided channel estimation for OFDM in wireless systems," *IEEE Trans. Vehic. Technol.*, vol. 49, no. 4, pp. 1207–1215, July 2000.

[5] H. Stark and J. W. Woods, *Probability and Random Processes, 3rd ed.* Upper Saddle River, New Jersey: Prentice-Hall, 2002.

[6] R. van Nee and R. Prasad, *OFDM for Wireless Multimedia Communications.* Boston: Artech House, 2000.

[7] L. Wilhelmsson, B. Bernhardsson, and L. Andersson, "Channel estimation by adaptive interpolation," U.S. patent 7,433,433, Oct. 7, 2008.

[8] IEEE Std 802.16-2004, *IEEE Standard for Local and Metropolitan Area Networks—Part 16: Air Interface for Fixed Broadband Wireless Access Systems*. New York: IEEE, June 24, 2004.

[9] IEEE Std 802.16e-2005 and IEEE Std 802.16-2004/Cor 1-2005, *IEEE Standard for Local and metropolitan area networks—Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems—Amendment 2: Physical and Medium Access Control Layers for Combined Fixed and Mobile Operation in Licensed Bands and Corrigendum 1*. New York: IEEE, Feb. 28, 2006.

[10] IEEE 802.16 Task Group m, *Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems — Advanced Air Interface (working document)*. Doc. no. IEEE 802.16m-09/0010r2, June 1, 2009.

[11] Man-On Pun, Michele Morelli, and C.-C. Jay Kuo, "Maximum-likelihood synchronization and channel estimation for OFDMA uplink transmissions," *IEEE Trans. Commun.*, vol. 54, no. 4, pp. 726–736, Aprl. 2006.

[12] Lior Eldar, M. R. Raghavendra, S. Bhashyam, Ron Bercovich, and K. Giridhar, "Parametric channel estimation for pseudo-random user-allocation in uplink OFDMA," *IEEE Int. Conf. Commun.*, 2006, vol. 7, pp. 3035–3039.

[13] S. Coleri, M. Ergen, A. Puri, and A. Bahai, "Channel estimation techniques based on pilot arrangement in OFDM systems," *IEEE Trans. Broadcasting,* vol. 48, no. 3, pp. 223–229, Sep. 2002.

[14] K.- C. Hung and D. W. Lin, "Pilot-based LMMSE channel estimation for OFDM systems with power-delay profile approximation," *IEEE Trans. Veh. Technology*, to appear.

[15] V. Erceg *et al.*, "Channel models for fixed wireless applications," IEEE 802.16.3c-01/29r4, July 2001.

[16] M.-H. Hsieh, "Synchronization and channel estimation techniques for OFDM systems," Ph.D. dissectation, Department of Electronics Engineering, National Chiao Tung University, Hsinchu, Taiwan, R.O.C., May 1998.

[17] Tien-Hsiang Lo, Kun-Chien Hung and David W. Lin, "Role of channel estimation in physical layer protocol design of OFDM wireless systems and relay-type cooperative communication," in *Proc. Workshop Wireless Ad Hoc Sensor Networks*, Chungli, Taiwan, ROC, Aug. 2006, pp. 301–308.

[18] Yi-Ling Wang, "Research in and DSP implementation of channel estimation techniques for IEEE 802.16e OFDMA uplink and downlink," M.S. thesis, Dept. Electronics Eng., National Chiao Tung Unversity, Hsinchu, Taiwan, R.O.C., June 2007.