A Thesis for the Degree of Ph.D. in Engineering

Resource allocation and channel estimation techniques for OFDMA systems

August 2010

Graduate School of Science and Technology
Keio University

Maduranga Liyanage

Acknowledgments

This dissertation is the end result of years of study in the Graduate School of Science and Technology of Keio University. Many people influenced and helped me, both directly and indirectly, in this journey so far. First and foremost I want to thank my supervisor Prof. Iwao Sasase. The five years under his guidance has taught me numerous aspects of the academic life, from how to write a technical paper to how a conference is organized and managed. I published my first international conference paper, my first journal paper and participated in my first international conference, all with Prof. Sasase's guidance. For that and everything else, I sincerely thank him.

Then I want to specially thank Prof. Yamanaka, Prof. Ohtsuki and Prof. Sanada. Not only they reviewed and commented on my dissertation which helped to improve its quality, but over the years their guidance on my work greatly helped shape the outcome of my work. It was a delight to have worked with them, not only on research, but also on international workshops which we participated in number of occasions.

There are far greater number of people I want to thank here than I possibly can. But I want to specifically mention some of my colleagues who helped in many ways and made my life at Keio a very memorable one. In no particular order, I heartily thank Bouk Safder, Koichi Adachi, Chinnapat Sertthin, Mamiko Inamori, San Maw, Motoki Shirasu, Junji Asada, Sho Shimizu and Kou Kikuta. I also want to specially acknowledge my colleague Oussama Souihli, who also influenced my academic success with his advice and constant encouragement. Then, my very good friend Alex Fung, without whom my life in Japan would have been a lot harder. He has helped me in every way possible from everyday life to research.

I want to specially acknowledge the Ministry of Education, Culture, Sports, Science and Technology (Monbukagakusho) of Japan for being the financial supporter of my studies at Keio University. Monbukagakusho scholarship is responsible for paving way for me

to study in Japan. I also want to thank Keio Leading-edge Laboratory of Science and Technology (KLL) grants which greatly helped in obtaining research material in my Ph.D years.

Lastly, my family, without whom this journey would not have been possible. To my father, who helped and encouraged me in every step of my education. It is because of him that I always wanted to be a telecommunications engineer. Then to my mother, I thank her for everything. And to my only sibling, my brother. Finally, I want to thank my wife for her love, support, encouragement and most of all her patience in these years. Thank you.

Maduranga Liyanage

August 2010

Abstract

Wireless communications are ubiquitous. People are no longer just checking email from wireless devices. Multimedia services are in very high demand, while the quality of multimedia services is increasing, the data transmission speeds are not advancing proportionately. This puts enormous pressure on the wireless service providers to constantly increase the throughput. The 4th generation wireless systems are been designed to address this throughput demand. Orthogonal-frequency-division-multiplexing (OFDM) technology is being chosen as one of the PHY/MAC layer of the 4th generation networks due to number of advantages it provides and high throughput being one of them. Although OFDM is able to provide increased throughput in theory, in practice it is difficult to achieve this limit. Resource allocation is the technique that allocates bandwidth, power, etc., to users, such that the system throughput is increased, while at the same time user quality-of-service (QoS) is maintained.

In this dissertation we focus our study mainly on two areas related to the OFDM technology: resource allocation and channel estimation. Resource allocation is a fundamental process that basically allocates data bits to subcarriers. This is an important process since the bandwidth and power is limited, resource allocation algorithm needs to allocate as much data as possible, at the same time satisfying different constraints, including time constraint. Next we study the process of channel estimation in OFDM systems, since channel estimations, or users frequency responses, are the most important information for the operations in the PHY, including resource allocation. Resource allocation algorithms require, among other information, accurate estimations of users channels to make proper resource allocations.

Chapter 1 presents an introduction to the OFDM system with its theoretical background. The resource allocation fundamentals are presented with some simulation results

showing the importance of proper resource allocation and also the nature of the complexity of the problem. We also introduce the channel estimation fundamentals of the OFDM system. Channel estimation is important in OFDM systems which use coherent demodulation, multiple-input-multiple-output (MIMO), antenna selection and many other techniques. Most of all, channel estimations are required for proper resource allocation. Resource allocation algorithms rely heavily on the channel estimations to allocate subcarriers, power, modulation and coding levels, etc., to achieve larger throughput and QoS. A discussion on importance of channel estimations and the complexity involved with the process is discussed here.

Chapter 2 presents our proposed low complexity resource allocation technique using a user ranking procedure to address the complexity issue of the channel estimation in OFDM systems,. This technique takes user channel characteristics into account and define several attributes. Fuzzy set theory is used on these attributes to prioritize the users. The proposed methodology is flexible in the sense that it can be used with different attributes to customize to the system parameters and required performance.

Chapter 3 explains the proposed steady-state Kalman filter for the channel estimation in OFDM systems. Conventional Kalman filters are used in OFDM systems for channel estimation due to their simplicity and the ability to operate in non-stationary environments. Nevertheless, Kalman filters are quite computationally complex due to a matrix inversion present in the calculation. In OFDM systems, channel estimations are required to perform in frequency domain and this could become a computational burden. The proposed steady-state Kalman filter uses channel and system characteristics to simplify the problem to a scalar level. In addition to reduced complexity, the steady-state Kalman filter is able to avoid the convergence period, and thus, providing better performance.

Chapter 4 presents a statistical analysis of the quantization noise present in an end-

to-end OFDM link. Quantization noise is present in every digital communication system, and although the higher resolutions of currently available quantizers are able to reduce the quantization noise to negligible levels, this higher resolution comes at a high power consumption cost. With newer systems integrating MIMO, the effect of power consumption of the quantizers are going to more severe as parallel radio-frequency (RF) links are required. Here we identify the effect of system model on the quantization noise. An analysis on how the signal PDF is changing from the transmitter to the receiver end, and how it affects the quantization noise is given. This study is motivated by the Kalman filter channel estimation, where the noise statistics are treated as an important parameter that needs to be known.

Chapter 5 concludes this dissertation with an overall discussion of the OFDM techniques discussed.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Generations of Wireless Communications

21^{st} century is seeing the dawning of the 4^{th} Generation (4G) of wireless communications systems. Wireless communications have come a long way since using analog techniques in the 1G networks. It migrated to the digital domain in the early 1990's which marked the 2G. Global System for Mobile Communication, or more popularly known as GSM, was first introduced in the 2G and (with added modifications) still is the most used mobile technology in the world. GSM was carried over to the 3G with added technology such as Enhanced Data rates for GSM Evolution (EDGE). Code Division Multiple Access (CDMA) technique is the new entry in to the 3G networks, with different flavors of it, such as CDMA2000 and Wideband direct sequence CDMA (WCDMA), deployed worldwide. In this dissertation I focus on the newest generation of the wireless communications, the 4-th generation, specifically the Orthogonal Frequency Division Multiplexing (OFDM),which is the underlying PHY technology. An introductory discussion about OFDM is given in the next section.

Fig. 1.1 shows the evolution of the wireless communications networks in respect

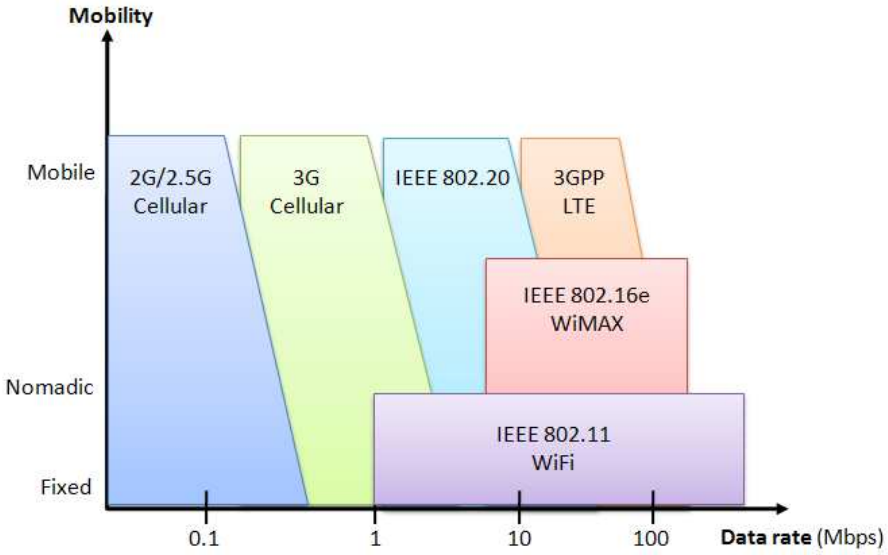

Figure 1.1: Generations of Wireless networks

to throughput and supported mobility. As evident from the figure, 2G networks had very limited throughput but mobility was well supported. Moving on to 3G networks saw an increase in throughput, but it varied upon the magnitude of the mobility. Finally, 4G networks surpassed the 100Mbps throughput mark with mobility receiving special attention, for example with the IEEE802.16e standard.

1.1.1 From 2G to 3G

Although marching towards the 4G, majority of the mobile subscribers are still using GSM-based 2G to 2.5G systems. For example, currently about 2.3 billion subscribers belong to GSM, while CDMA has 450 million [1]. 1G systems were analog and designed to support only voice services [2]. Frequency-division-duplex (FDD) is used in these systems. Advanced Mobile Phone Services (AMPS) was the popular 1G system, which also had competitors like Total Access Communications Standard (TACS) and Nordic Mobile

Telephone (NMT). These systems soon proved inadequate and wireless services entered to 2G in early 1990. GSM and CDMA-based IS-95 were the first technologies in this generation. In addition to being a digital system, these technologies also provided Simple Message Service (SMS) and data. The data rates were in the range of 9.6∼14.4kbps [3]. Over the years many technologies were included in to the 2G systems to increase its data rate. For example, General Packet Radio Services (GPRS) and High Speed Circuit Switched Data (HSCSD) helped 2G networks achieving data rates up to 384kbps.

3G services mainly focused on providing higher data rates. It was realized that packet access was the solution for the growing need of consumer requirements. 2G systems started as circuit-data systems, but GPRS and CDMAOne technologies (which are regarded as 2.5G) pushed the systems in to the packet data domain. So it was inevitable that 3G systems are all packet oriented. WCDMA saw the start of 3G systems. WCDMA, CDMA2000 and EDGE are the three main technologies in 3G [4]. EDGE is a technology built on top of GSM and is backward-compatible. Although built on top of GSM, EDGE belongs to 3G according to International Telecommunication Union (ITU) [5]. These networks initially supported speeds up to 2Mbps [3]. Like 2G, 3G also saw different enhancements added on to it, in particular, High Speed Downlink Packet Access (HSDPA) and Evolution Data Optimized (EV-DO). Data rates were substantially increased, with HSDPA taking the data rate up to a theoretical maximum of 14.4Mbps(quadrature-amplitude-modulation) [6]. Release-7 of 3GPP makes provision for up to 64-QAM in the downlink (release-6 supported only up to 16-QAM) which drives the data rate to about 21Mbps [7]. Original EV-DO supported data rates up to 2.4Mbps in a 1.25Mhz bandwidth, while Revision-A and Revision-B as increased the data rates to 3.1Mbps and 4.9Mbps, respectively. At this moment 3G has come quite a long way from its initial 2Mbps and the thus saw the migration in to the next-generation.

1.1.2 The Fourth Generation

As 2G was the point of entry in to the digital domain, 4G is the entry in to a native all IP domain. Main aspect of 3G networks is the multimedia services, but the recent explosion of multimedia services have proved too much to handle for the traditional 3G networks. With a clearly defined list of objectives, 4G networks are aiming to address the throughput problems of the existing networks. Among others, following objectives are the main priorities of the this newest generation [8]:

- Very high throughput.
- Usability: anytime, anywhere, with any technology
- Seamless connectivity across heterogeneous networks.
- Delivery of high quality multimedia contents.
- Compatibility with existing wireless standards.
- An all IP network.

There are three major candidates in the 4G technologies, IEEE802.16 (WiMAX and it's variants), Long Term Evolution (LTE) and Ultra Mobile Broadband (UMB) [1]. Each of these technologies are backed by different sets of industry giants and they aim to serve a specific market. 3GPP, the group responsible for GSM, is developing LTE, and has the advantage of easier migration to 4G for the existing GSM-based systems. UMB is from 3GPP-2, the consortium for CDMA-based systems. Hence, UMB provides easier migration for the CDMA-based systems to the next-generation. IEEE802.16 or WiMAX, on the other hand, does not provide an easier migration to the next-generation to existing operators and that seems to be hurting its' adoption. Nevertheless, number of WiMAX

systems are starting to be deployed throughout the world. Whichever is the technology, all these promise one major thing, high data rates. With the use of technologies like Multiple Input, Multiple Output (MIMO), the data rates are going to reach well over 100Mbps, with WiMAX II set to give 1Gbps for stationary users.

1.2 Introduction to OFDM

OFDM is in basic terms a form FDM. In the conventional FDM systems, each frequency band is separated from other frequency bands by a guard band placed in-between. This is depicted in the top figure of Fig. 1.2. The function of the guard band is to eliminate the Inter Carrier Interference (ICI), i.e. interference from neighboring frequency bands. In the context of OFDM, a frequency band is referred to as a *subcarrier*, so I will use it from hereon. OFDM, on the other hand, has its subcarriers overlapping (bottom of Fig. 1.2). This, evidently increases the spectrum efficiency, which is one of the primary advantages of OFDM. But since the subcarriers are overlapping, a special mechanism is needed to avoid this ICI. OFDM avoids ICI by using the orthogonality property. Orthogonality between an exponential set is defined as

$$\frac{1}{T_s}\int_{nT_s}^{(n+1)T_s} \texttt{exp}(j2\pi f_k t)\texttt{exp}(j2\pi f_n t) * dt = \delta_{nk} \tag{1.1}$$

where T_s is the symbol period. Therefore, by choosing the subcarrier frequencies to be multiples of $\frac{1}{T_s}$, interference between subcarriers can be completely avoided. The mechanism of modulating data symbols in such a way that the transmitting waveform will have orthogonal subcarriers can be referred to as OFDM. OFDM achieves this through Discrete Fourier Transform (DFT). DFT, one of the transformations in the Fourier Transformations (FT) family, is essentially the transformation from time-domain to the frequency-domain.

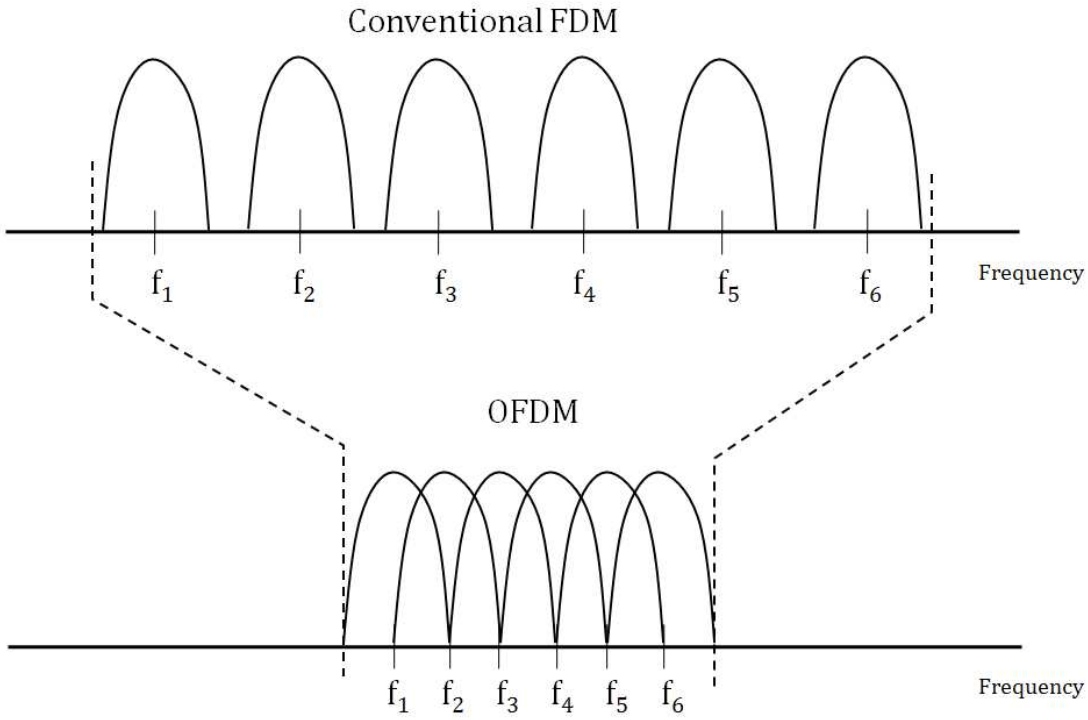

Figure 1.2: Conventional FDM vs. OFDM

Mathematically it is

$$\mathtt{X}(k) = \sum_{n=0}^{N-1} \mathtt{x}(n)\mathtt{exp}\left(-j2\pi\frac{nk}{N}\right) \qquad k = 0, 1, \ldots, N-1 \tag{1.2}$$

where x(n) is the time-domain values, which are sampled from the continuous time-domain signal x(t). In the case of OFDM transmission, frequency-domain values are put in to the time-domain. So the operation is the inverse, or Inverse DFT (IDFT),

$$\mathtt{x}(n) = \frac{1}{N}\sum_{k=0}^{N-1} \mathtt{X}(k)\mathtt{exp}\left(j2\pi\frac{nk}{N}\right) \qquad n = 0, 1, \ldots, N-1 \tag{1.3}$$

It can, therefore be seen that OFDM takes N complex data points and transmits them simultaneously in one symbol. I call $x(n), n = 0, 1, \ldots, N-1$ as a single OFDM symbol. Next section describes the main components of an OFDM transmitter through to the receiver.

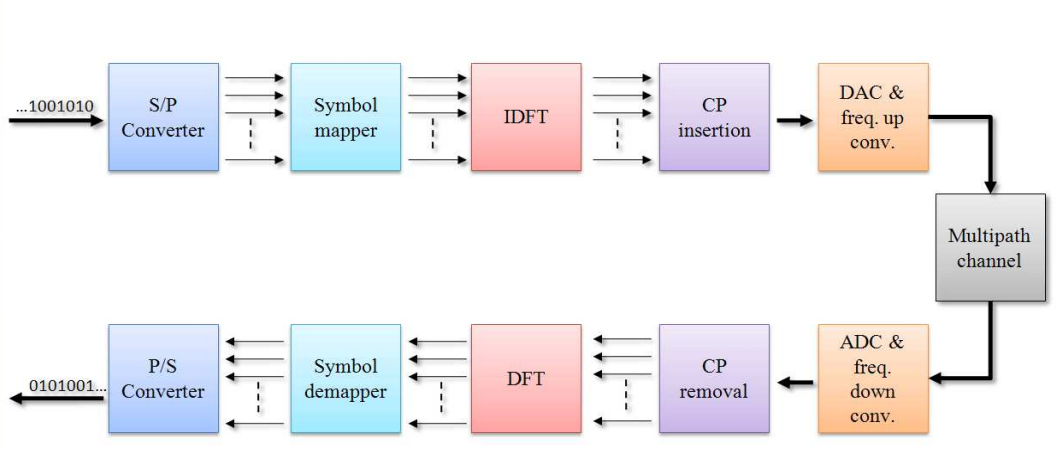

Figure 1.3: OFDM transceiver link

1.2.1 OFDM Link Components

Fig. 1.3 shows the main components of an OFDM transmitter-receiver link. Each component is described in the following.

Serial-to-Parallel conversion

The responsibility of the PHY of the Open System Interconnection (OSI) model is to transmit the data coming from the higher layers on to the transmission medium. Data coming in to the PHY is in the form of bits, i.e. in zeros and ones. PHY takes these bits and converts them in to a form that can be efficiently transmitted. This is modulation. The data are received in serial form, but OFDM transmits, or modulates, N data points at once. Therefore this serial bit stream is converted to N parallel data streams by the serial-to-parallel converter component. If the serial bit stream data rate is $R_S$bps, then the data rate of each parallel stream will be $R_P = \frac{R_S}{N}$.

Symbol mapping

The N parallel bit streams are sent to a symbol mapper. This component modulates the incoming bits in to symbols, according to a certain modulation scheme. In OFDM, adaptive modulation is possible, i.e. each parallel bit stream can be modulated with a different modulation scheme. OFDM usually employs QAM as its base modulation scheme, and uses 4-QAM (QPSK), 16-QAM and 64-QAM adaptively on each stream individually. The output of the symbol mapper, therefore, will be N complex values, corresponding to each of the N input streams. These N complex values are the data points which will be transmitted.

Inverse Discrete Fourier Transformation

This is the heart of the OFDM, the IDFT, transforming the N complex frequency-domain values on to the time-domain. The N complex values from the symbol mapper are fed in to the N inputs of the IDFT component. In OFDM terms, this is called putting the data on to subcarriers. The IDFT basically modulates the input N (complex) data-points in to N orthogonal subcarriers, so it can be transmitted without interference from other overlapping subcarriers. In the output of the IDFT, again N complex values are the time-domain values which will be transmitted. These N points will then be parallel-to-serial converted to be transmitted as a single OFDM symbol. The data points are spaced $\frac{T_s}{N}$ seconds apart.

Cyclic Prefix Insertion

Cyclic Prefix (CP) serves a very important purpose for OFDM and so a separate section is reserved for it later in the chapter. The basic functionality of the CP is to avoid ISI. When a signal is transmitted through a multipath channel, ISI is inherently present if

the symbols are not adequately spaced. In the case of OFDM this is more difficult because the OFDM symbol is much longer than the channel delay and the multipath delay. Deferring the explanation to the later section, CP insertion only will be briefly discussed here. CP Insertion is the simple process of taking the last N_{CP} ($> L-1$, where L is the number of multipaths) values of the the IDFT points and appending it to the beginning of the data stream.

Digital-to-Analog Conversion and Frequency Up Conversion

This is the last stage of the RF front end. The serial data stream will be Digital to Analog Converted (DAC) and then multiplied by a high frequency carrier to be transmitted. DAC can be performed by passing the data through a Low Pass Filter (LPF).

Discrete Fourier Transform

At the receiver end, the processes of the transmitter is reversed. DFT module transforms the time-domain data samples in to the frequency-domain. Assuming no distortion is caused during transmission, parallel-to-serial converter will output the original data stream of the transmitter.

Theory of Cyclic Prefix

In this section I present the mathematics behind the Cyclic Prefix. CP is famous as the technique of avoiding ISI in OFDM, but it also performs an important task of preserving the orthogonality of subcarriers when linearly convolved through the multipath channel, as discussed below.

Considering an N-point FFT system, a block of data points $\mathtt{X} = [x_0 \quad x_1 \quad \dots \quad x_{N-1}]$, is to be transmitted during a symbol time. The data vector $\mathtt{X}$ is sent to an IDFT module.

The operation of the DFT can be represented by the following matrix:

$$[\mathtt{F}]_{k,m} = \frac{1}{\sqrt{N}} e^{-j2\pi \frac{km}{N}}, k, m = 0, 1, \ldots . N-1, \tag{1.4}$$

where (k, m) denotes the $(k+1, m+1)^{th}$ entry in the DFT matrix $\mathtt{F}$.

Then the transmitted data vector after the IDFT operation is

$$\mathtt{x} = \mathtt{F}^H \mathtt{X}. \tag{1.5}$$

The normalization factor $\frac{1}{\sqrt{N}}$ takes care of the total bit energy as it is the same as the original vector $\mathtt{x}$ because $\mathtt{F}\mathtt{F}^H = \mathtt{F}^H\mathtt{F}$.

Assuming the channel length is L, the OFDM symbol requires a CP of length $N_{CP} \geq (L-1)$. Then the total symbol length is $\bar{N} = N + N_{CP}$.

After the (linear) convolution through the channel and removal of the CP, the received signal (at the input to the DFT module) can be written,

$$\mathtt{y} = \mathtt{H}\mathtt{x} + \mathtt{n} \tag{1.6}$$

where $\mathtt{n}$ is the AWGN noise vector. The channel matrix $\mathtt{H}$ is a circulant matrix due to the insertion of the CP in the transmitted data vector and can be written as,

$$\mathtt{H} = \begin{bmatrix} \mathtt{h}_0 & \mathtt{h}_{N-1} & \cdots & \mathtt{h}_1 \\ \mathtt{h}_1 & \mathtt{h}_0 & \cdots & \mathtt{h}_2 \\ \vdots & \vdots & \ddots & \vdots \\ \mathtt{h}_{N-1} & \mathtt{h}_{N-2} & \cdots & \mathtt{h}_0 \end{bmatrix} \tag{1.7}$$

Here $\mathtt{h}_i$ is the channel impulse response and $\mathtt{h}_i = 0$ for $L \leq i \leq (N-1)$.

Circulant matrices have an important property that the eigenvectors of a circulant matrix of a given size are the columns of the discrete Fourier transform matrix of the same size.

Therefore,

$$\mathtt{H} = \mathtt{F}^{H}\mathtt{H}_{\mathsf{Eig}}\mathtt{F} \tag{1.8}$$

Thus,

$$\mathtt{F}\mathtt{H}\mathtt{F}^{H} = \mathtt{H}_{\mathsf{Eig}} = \mathsf{diag}(H_0, H_1, \ldots, H_{N-1}) \tag{1.9}$$

where,

$$\mathtt{H}_n = \sum_{l=0}^{L-1} \mathtt{h}_l \mathtt{exp}\left(-j2\pi\frac{nl}{N}\right), n = 0, 1, \ldots . N-1. \tag{1.10}$$

Therefore, H_n is just the frequency response of the n^{th} subcarrier and hence, the eigenvalues of $\mathtt{H}$ are the frequency responses of the channel.

Then, going back to the received signal vector, $\mathtt{y} = \mathtt{H}\mathtt{x} + \mathtt{n}$, sending this received data vector through the DFT [9],

$$\begin{aligned} \mathtt{Y} &= \mathtt{F}\mathtt{y} \\ &= \mathtt{F}(\mathtt{H}\mathtt{x} + \mathtt{n}) \\ &= \mathtt{F}(\mathtt{H}\mathtt{F}^{H}\mathtt{X} + \mathtt{n} \\ &= \mathtt{F}\mathtt{H}\mathtt{F}^{H}\mathtt{X} + \mathtt{F}\mathtt{n} \\ &= \overline{\mathtt{H}}\mathtt{X} + \tilde{\mathtt{n}} \end{aligned} \tag{1.11}$$

Here $\overline{\mathtt{H}} = \mathtt{H}_{\mathsf{Eig}}$ and that the statistical properties of $\mathtt{n} = \mathtt{F}\mathtt{n} = \tilde{\mathtt{n}}$.

Therefore, the insertion of the CP helps OFDM receivers easily decode the transmitted date vector by a simple inverse DFT operation (single-tap equalization).

1.3 Resource Allocation in OFDM

Resource allocation is an important part of any wireless communication system, because bandwidth is a very expensive resource to any service provider. And it is very scarce.

There is only a certain bandwidth that could be used for consumer communications. On top of that, there is a strict limit imposed by regional regulatory committees, like the Federal Communications Commission (FCC) of the United States, on the maximum transmission power. This means that there is a strict upper bound on the SNR that can be provided by the base station to the end terminals. The throughput is hence upper bounded and base stations need to employ many complicated techniques to get to close to that upper bound. This is the challenge of resource allocation.

OFDMA has been chosen as the PHY/MAC technology for the next-generation broadband wireless communication systems [10]. It is selected on the promise of higher throughput for the next generation systems which are at present facing severe throughput demands. Recent explosion of smartphones has changed the traffic usage patterns of the existing networks that it has become difficult to provide the guaranteed QoS to the end users. Migration of traditional broadcast services like internet radio and TV into the IP domain has made things worse due to their high bandwidth consumption. It is our objective to address the efficient serving of broadcast services to end users of a multiuser system such as OFDMA, in a resource efficient manner.

OFDMA is capable of very high throughput at its' maximum theoretical performance. In practical deployments it is very difficult to achieve these performance levels due to limited RF bandwidth and transmission power restrictions and channel fading. Therefore, resource allocation in OFDMA networks has been a deeply researched area. There are many parameters to consider and the optimal resource allocation is a NP-hard problem that is mathematically difficult to solve [11], and hence researchers are resolving to sub-optimal schemes[12]. The main objectives of any resource allocation scheme must be:

- Maximizing system throughput

- Minimizing transmit power
- Maintaining QoS across all users

Earlier resource allocation research worked mainly in the PHY of OFDMA [12]. But it was realized that resource allocation techniques need to consider both PHY and MAC of OFDMA to provide a sustainable solution to the stringent QoS requirements of recent services which are not only bandwidth sensitive but also delay sensitive.

1.3.1 Complexity and Optimal Solution

Resource allocation in an OFDMA system is a highly complex task to be performed optimally. This is because of the many parameters in an OFDMA system which must be fine tuned to achieve the optimal solution. In principal these parameters are:

- Multiple users
- Time-varying channel conditions of each user
- Transmit power constraint
- Multiple modulation and coding levels
- Different QoS requirements

These parameters and constraints make the optimal resource allocation a highly non-linear optimization problem that is mathematically very complex. Furthermore, the resource allocation is a process which needs to be performed each OFDM symbol period, which makes it an even more difficult problem.

1.3.2 Optimal Bit Loading

The optimal bit loading strategy is the technique that would load the maximum possible number of bits on to a given set of subcarriers under a total power constraint. For an OFDM system, this technique is known as *water-filling*. The transmission power is not a linear function of the number of bits, therefore, when loading bits on to subcarriers, it must be done granularly, i.e. bit-by-bit, for each subcarrier while choosing the subcarrier which requires the least amount of power to load a bit. The algorithm for water-filling is[13] [14],

Water-filling algorithm

begin

Initialize

$c_k = 1, \Delta P_{k,c_k} = 0, P_{\mathsf{T}} = 0;$

while $P_{\mathsf{T}} \leq P_{\mathsf{MAX}}$ do

for $k := 1$ to K do

calculate $\Delta P_{k,c_k}$;

end

$k* = \mathsf{argmin}_k \Delta P_{k,c_k}$;

$c_{k*} = c_{k*} + 1;$

$P_{\mathsf{T}} = P_{\mathsf{T}} + \Delta P_{k,c_k};$

end

end

where $\Delta P_{k,c_k} = P_{k,c} - P_{k,c-1}$.

1.3.3 Water-Filling vs Greedy-Allocation

The water-filling algorithm takes each subcarrier and calculates how much additional power is required to load a bit on to each subcarrier, and then chooses the subcarrier which requires the least power and allocates a bit. The process is repeated until the maximum transmission power is reached. This algorithm is computationally intensive since the power needs to be calculated for each subcarrier in a per-bit basis. The computational time can be prohibitive for real-time processing with increased number of subcarriers and power levels. On the other hand, the easiest resource allocation method is the Greedy allocation method. There can be several alternatives of Greedy algorithms [14] [15], but as the name simply implies that a good subcarrier consumes more resources. The Greedy algorithm takes subcarriers in the order of their channel gains, better subcarriers first, and allocates the highest possible number of bits on to that subcarrier. An example subcarrier allocation scenario is depicted in Figs. 1.4, 1.5 and 1.6. The uniform power allocation is the case where the total transmission power is equally distributed among the subcarriers. In these simulations, total power is kept constant and the bit loading is done by modulations levels, instead of bit-by-bit. Modulation levels used in these are QPSK, 16-QAM and 64-QAM, which transmit 2, 4 and 6 bits, respectively. Fig. 1.4 shows the Greedy method and Fig. 1.5 shows the same method with power distributed among subcarriers equally. The Greedy method, as can be seen from Fig. 1.5 has allocated the highest number of bits to better subcarriers, while only few subcarriers are allocated with 2 and 4 bits. This is because more power is wasted in allocating the maximum amount of bits on to the subcarriers with better gain. Uniform power distribution scheme shows even worse performance. Only 2-bit allocations are possible in the best subcarriers available. This is because when the power is distributed uniformly, the fraction of the power available to each subcarrier is not enough for loading bits. The water-filling method, on the other hand, shows much higher bit load-

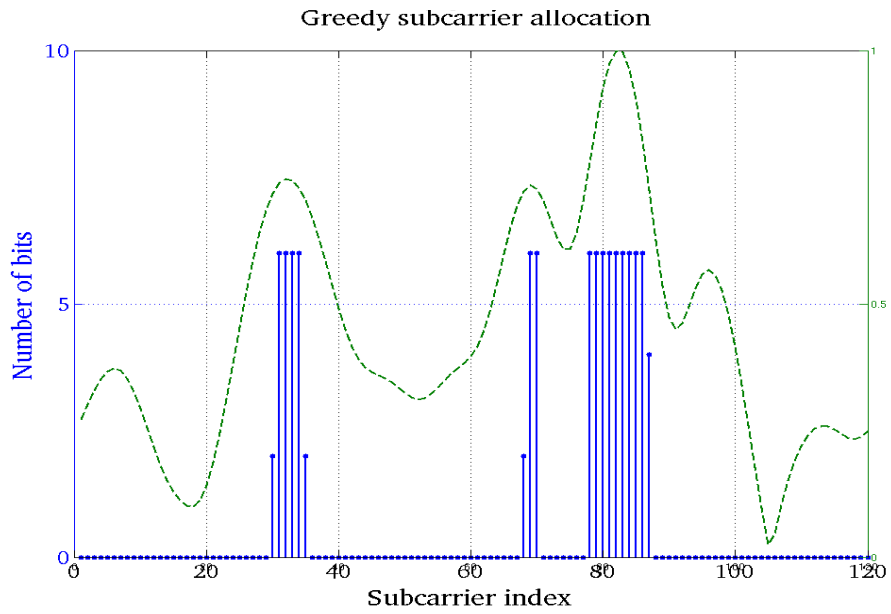

Figure 1.4: Greedy subcarrier allocation

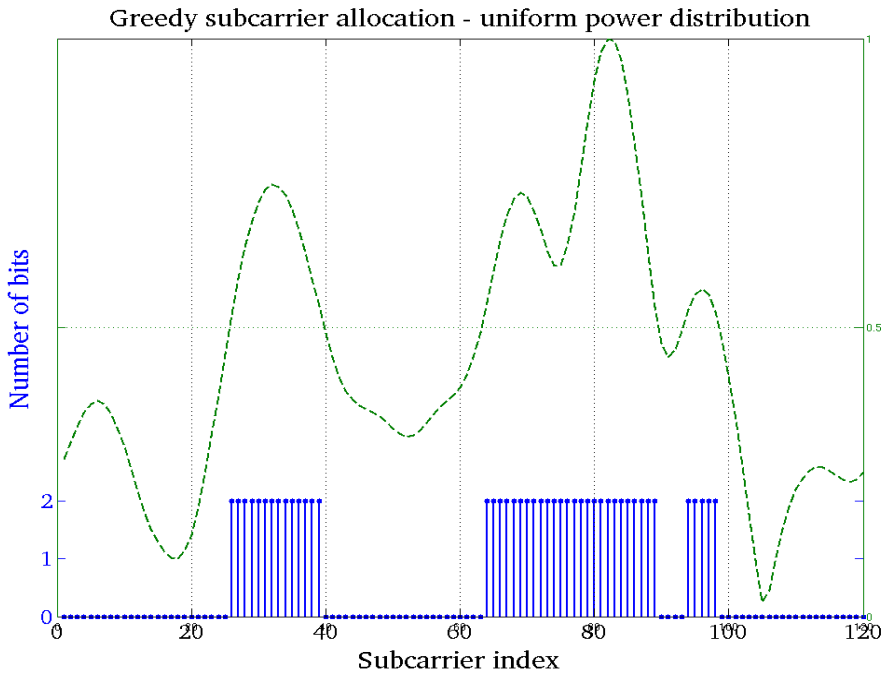

Figure 1.5: Greedy subcarrier allocation - uniform power distribution

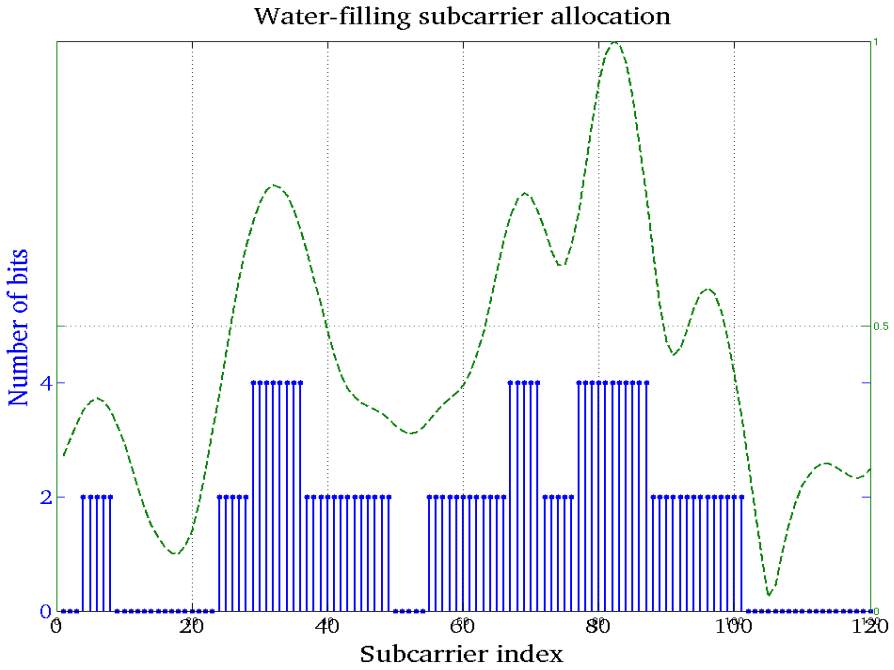

Figure 1.6: Water-filling subcarrier allocation

ing on to the subcarriers. Substantially higher number of subcarriers are allocated bits in this scheme. A point to note is that in the water-flling method, none of the subcarriers are assigned 6-bits. Instead, many subcarriers, except for ones in deep fade, are allocated with 2-bits. The stringent SNR requirement of higher-order modulation levels make it resource efficient to allocate many low-gain subcarriers with lower number of bits than allocating few good subcarriers with high number of bits.

1.3.4 OFDMA Resource Allocation - A Combinatorial Optimization Problem

The Water-filling method discussed in the previous section considers an OFDM system, or a single-user system. In an OFDMA system, where resources needs to be allocated to multiple users, the allocation problem becomes extremely complex. The optimal

solution of water-filling for the OFDM is no longer optimal in an OFDMA system because only one user can be allocated to a subcarrier at one symbol period. Furthermore, the optimal solution is not the allocation strategy that transmits more bits. When multiple users are present, the system needs to satisfy the QoS requirements of the users. Each user has a different QoS requirement. And the allocation strategy should try to provide the necessary bit allocations such that the QoS requirements are met. Resource allocation in an OFDMA system is a combinatorial optimization problem that could have different objectives and constraints. For example, in the water-filling algorithm of previous section, the objective was to maximize the number of bits with an upper bound on the transmission power. Another approach would be to minimize the transmission power with the constraint of satisfying user QOS requirements.

The allocation problem of minimizing the transmission power with the constraint of satisfying user data requirements can be stated as a Lagrangian optimization as follows [12]:

$$\begin{aligned} \text{Minimize} \quad & f(\mathbf{P}) = \sum_{n=1}^{N} \sum_{k=1}^{K} P_{k,n} \\ \text{s.t.} \quad & \sum_{n=1}^{N} c_{k,n} \geq R_k, \quad \forall k \in \{1, 2, \ldots, K\} \end{aligned} \tag{1.12}$$

Here R_k is the rate requirement of user k. The optimization state as above is difficult to analyze because the power constraint is a function of the number of bits on it and only one user is able to use a single subcarrier. To make the optimization mathematically tractable, a subcarrier usage indicator, $\rho_{k,n}$ needs to be introduced, which is defined

$$\rho_{k,n} = \begin{cases} 1 & \text{if} \quad c_{k,n} \neq 0 \\ 0 & \text{if} \quad c_{k,n} = 0 \end{cases} \tag{1.13}$$

provides a notion of time-sharing of a subcarrier among the users.

Then (1.12) can be restated as,

$$\begin{aligned} \text{Minimize} \quad & f(\mathbf{P}) = \sum_{n=1}^{N}\sum_{k=1}^{K} \rho_{k,n} P_{k,n} \\ \text{s.t.} \quad & \sum_{n=1}^{N} \rho_{k,n} c_{k,n} = R_k, \quad \forall k \in \{1, 2, \ldots, K\} \\ & \sum_{k=1}^{K} \rho_{k,n} = 1, \quad \forall n \in \{1, 2, \ldots, N\} \end{aligned} \tag{1.14}$$

where

$$P_{k,n} = \frac{1}{h_{k,n}^2} . \frac{N_0}{3} \left[Q^{-1} \left(\frac{P_e}{4} \right) \right]^2 \left(2^{c_{k,n}} - 1 \right), \tag{1.15}$$

is the transmission power on subcarrier k for user n. Here $\rho_{k,n}$ acts as a sharing factor among users for each subcarrier. But since subcarriers are utilized exclusively among users, $\rho_{k,n}$ is allowed to have values of 0 or 1 only. Then the problem comes down to the original problem of (1.12).

Then the Lagrangian is,

$$\mathcal{L}\left(\mathbf{P}, \boldsymbol{\lambda}, \boldsymbol{\sigma}\right) = \sum_{n=1}^{N}\sum_{k=1}^{K} \rho_{k,n} P_{k,n} - \sum_{n=1}^{N} \lambda_n \left(\sum_{k=1}^{K} \rho_{k,n} c_{k,n} - R_n \right) + \sum_{n=1}^{N} \sigma_k \left(\sum_{n=1}^{N} \rho_{k,n} - 1 \right), \tag{1.16}$$

where $\boldsymbol{\lambda}$ and $\boldsymbol{\sigma}$ are the vectors of Lagrangian multipliers. The conditions are then,

$$\frac{\partial \mathcal{L}}{\partial c_{k,n}} = \rho_{k,n} P'_{k,n} + \lambda_n \rho_{k,n} = 0. \tag{1.17}$$

$$\frac{\partial \mathcal{L}}{\partial \lambda_n} = \lambda_n \left(\sum_{k=1}^{K} \rho_{k,n} c_{k,n} - R_n \right) = 0. \tag{1.18}$$

$$\frac{\partial \mathcal{L}}{\partial \sigma_k} = \sigma_k \left(\sum_{n=1}^{N} \rho_{k,n} - 1 \right) = 0. \tag{1.19}$$

The equations given above is a framework for finding the optimal solution and modifications are necessary to find an optimal result. For example, the values of $c_{k,n}$ and $\rho_{k,n}$ are allowed to vary in the range $[0 \quad M]$ and $[0 \quad 1]$, respectively. But in the allocation, $\rho_{k,n}$ can only

get values 0 or 1, and $c_{k,n}$ can only have the discrete values of modulation levels used in the system.

Water-filling, greedy and Lagrangian methods described above are the resource allocation strategies that reside on the extremes of the complexities. For practical systems however, these schemes are not feasible and hence a plethora of research have gone and are still going in to introducing sub-optimal resource allocation schemes, which have less complexity to be able to implement in real-time systems and also provides acceptable performance. In chapter 2 I introduce the proposed low complexity resource allocation scheme for OFDMA systems.

1.4 Channel estimation in OFDM systems

Channel estimation is very important in any communication system. The importance of it is far more exceeded in the case of wireless communications because the channel often changes and accurate channel estimations are required for coherent demodulation of the data signal. Additionally, accurate channel estimations are required by a number of important technologies used in wireless communications, such as,

- Adaptive modulation
- Adaptive power control
- Adaptive antenna combining
- Space-time decoding

As with most every other major technique in OFDM, channel estimation is required for resource allocation too, but is often taken for granted. In the previous chapters, I allocated resources to users depending primarily on the subcarrier gains. In section 1.3.3 I presented

the superiority of the water-filling algorithm as the optimal bit-loading technique for OFDM systems. The water-filling algorithm is solely based on the subcarrier channel gains. If these values are not accurate, as we will see later in simulations results, that both power is wasted and BER performance is degraded. Therefore, in this chapter I introduce this importance technique of channel estimation.

1.4.1 Categories of Channel Estimation

Channel estimation can be broadly divided in to several categories as shown in Fig. 1.7. These categories show two main approaches taken in channel estimation techniques. These techniques are not limited to wireless situations and are employed in wired communications as well. The categorization to blind and non-blind estimations fundamentally refer to whether the estimator has access to pilot data or not. Pilot data are which sent by the transmitter for the use at the receiver for number of purposes. Among other uses, pilot data are used by the receiver for channel estimation, frequency synchronization and time synchronization. Although use of pilots are an efficient and easier way to estimate the channel, pilot data reduce spectrum efficiency since no user data can be sent in these symbols. Nevertheless, use of pilot data for channel estimation eases the estimation process as explained next.

Blind channel estimation

Blind channel estimation, as the name implies, estimates the channel *blindly*. This category of techniques relies primarily on the channel and signal statistics for the estimation of the channel. Therefore, a large number of data are required for a good estimation and the procedure is also mathematically complex. For these reasons, blind channel estimation methods are not used in practical mobile systems, because they are unable to provide

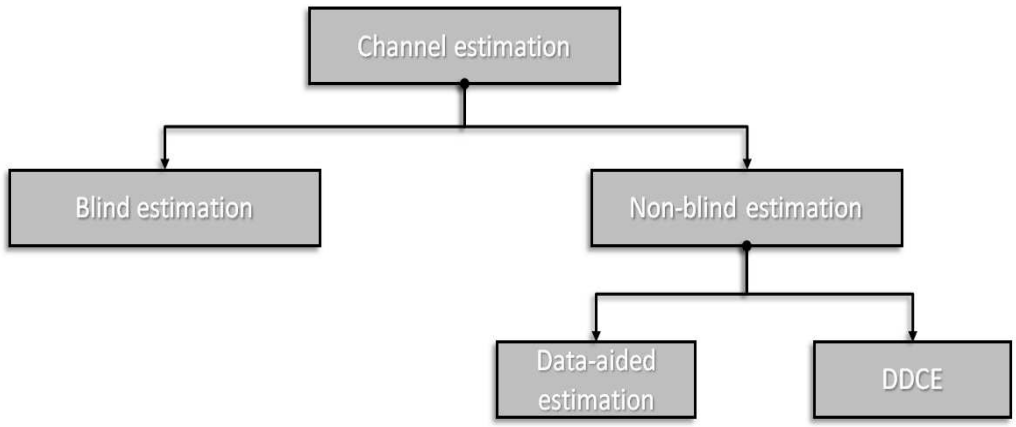

Figure 1.7: Channel estimation techniques

accurate estimations in fast-varying channel conditions. I will discuss non-blind channel estimation techniques in this chapter because they are used in practical systems, especially in WLAN and WiMAX systems.

Non-Blind channel estimation

Non-blind channel estimation, on the other hand, makes use of data in the received signal. In this category of channel estimators, the transmitter sends a sequence of symbols known at the receiver, called *pilot symbols*, and the receiver can estimate the channel by the state of the received pilot symbols. Non-blind channel estimation techniques can be further divided into two more sub-categories: Decision Directed Channel Estimation and Data Aided Channel Estimation [16].

Decision Directed Channel Estimation (DDCE) In DDCE, current symbol is decoded using a channel estimate derived from the previous symbol. The current channel is estimated using the newly decoded symbol, which was decoded using the channel estimates from the previous symbol. Therefore, DDCE inherently has the disadvantage of using an

outdated channel estimate. If the channel is stationary or very slowly-varying, then this procedure would not pose a problem, but in fast-varying cases, DDCE poses a disadvantage, as error propagation can occur. To circumvent this, number of methods are used. For example, DDCE requires an initial channel estimate. This usually is obtained from a pilot-sequence of subcarriers and a data-aided channel estimation technique (which I discuss next). One way to improve the DDCE performance is to send pilot-sequences periodically, so that the channel estimations can be updated accurately at regular intervals. In addition, channel coding and interleaving techniques can also be incorporated.

Data Aided Channel Estimation (DACE) DACE is the widely used technique in most practical systems. This category of techniques make use of pilot symbols, i.e. symbols which are known to the receiver. Other than that, pilot symbols can also have special properties such as having a constellation of constant modulus in order to simplify the estimation process. Pilot symbols or in the case of OFDM, pilot subcarriers, are further divided in to two categories:

- Block-type pilots
- Comb-type pilots

Block-type pilots are sent as a symbol of all pilots and comb-type pilots are uniformly distributed among the data subcarriers. These two types of pilot subcarriers are graphically depicted in Fig. 1.8.

Fig. 1.8 shows four types of pilot subcarriers defined in the IEEE802.16 standard [17]:

- **Preamble** - A symbol of all pilots sent at the beginning of each OFDM frame. In addition to channel estimation, this pilot symbol, the preamble, is also used for time

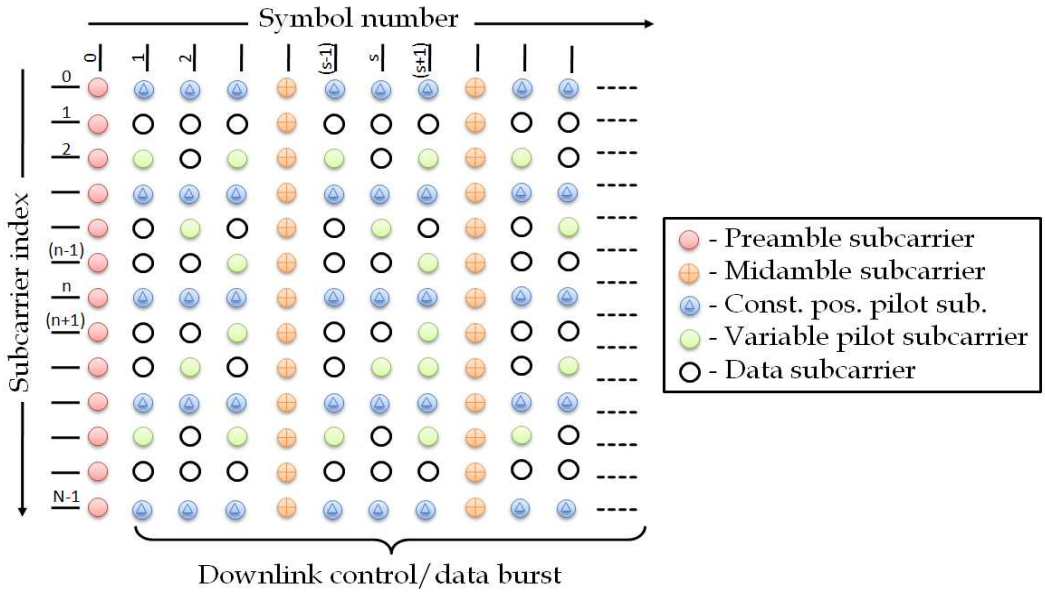

Figure 1.8: Pilot subcarrier types of IEEE802.16 standard

and frequency clock synchronizations.

- **Midamble** - Similar to the preamble, but is sent in the middle of the symbol.
- **Constant position pilots** - These pilots are sent with each symbol and are spread uniformly throughout the frequency spectrum. Receivers can use interpolation/extrapolation techniques to estimate the channel gain of the data subcarriers.
- **Variable position pilots** - These pilots are sent among the remaining subcarriers. The position of these pilots could be according to a certain pattern (ex. moving diagonally) or could be inserted to aid certain receivers to cope of different channel conditions

From the above four types of pilot subcarrier types, we can identify preambles and midambles as block-type pilots and constant positions pilots as comb-type pilots. It is evident that the insertion of pilot subcarriers hinders spectral efficiency since no data can be

sent in these subcarriers, but as the support for vehicular mobility with dense constellations are increased, pilots are unavoidable.

In the comb-type pilots, pilot subcarriers are inserted at some distance apart. The minimum distance of these pilot subcarriers is a function of the channel delay as is given by

$$D_{freq} \leq \frac{1}{\tau_{\mathsf{MAX}} \Delta f_d} \tag{1.20}$$

where τ_{MAX} is the maximum channel delay and Δf_d is the subcarrier spacing. This minimum distance needs to be maintained to capture the channel variations in the frequency-spectrum. Similarly, there is a maximum-delay between pilots which is

$$D_{time} \leq \frac{1}{2 f_{D_{\mathsf{MAX}}} T_s} \tag{1.21}$$

where $f_{D_{\mathsf{MAX}}}$ is the maximum Doppler spread and T_s is the OFDM symbol time.

1.4.2 Channel Estimation Techniques

Channel estimation can be performed using a number of different techniques, each with its' advantages and disadvantages. Nevertheless, these methods can be categorized in to two main types of techniques:

- Least Squares estimation (LS)
- Minimum Mean Squared Estimation (MMSE)

Least-squares estimation

LS is the easiest of the two techniques, and gives the worst performance. If received signal after DFT on the k^{th} subcarrier of the n^{th} symbol is

$$Y_{k,n} = X_{k,n} H_{k,n} + W_{k,n} \tag{1.22}$$

Then the LS estimate of $H_{k,n}$ is

$$\begin{aligned}\hat{H}_{k,n} &= \frac{Y_{k,n}}{X_{k,n}} \\ &= H_{k,n} + \bar{W}_{k,n},\end{aligned} \tag{1.23}$$

where $\bar{W}_{k,n} = \frac{W_{k,n}}{X_{k,n}}$. In matrix notation, the vector of the LS channel estimate can be written

$$\hat{\mathbf{H}}_{\mathsf{LS}} = \mathbf{X}^{-1}\mathbf{Y} \tag{1.24}$$

Here I have taken $\mathbf{X}$ as a diagonal matrix of dimensions $N \times N$. The above LS estimation is performed in the frequency-domain. The same procedure can be done in the time-domain as well. Using the DFT matrix $\mathbf{F}$ introduced in section 1.2.1, this can be written [16]

$$\mathbf{H} = \mathbf{F}\mathbf{h}, \tag{1.25}$$

where $\mathbf{h}$ is the $N \times 1$ vector of CIR. Then (1.22) can be written

$$\mathbf{Y} = \mathbf{X}\mathbf{F}\mathbf{h} + \mathbf{W}. \tag{1.26}$$

Then the time-domain LS estimate is given by

$$\hat{\mathbf{H}} = \mathbf{Q}_{\mathsf{LS}}\mathbf{F}^H\mathbf{X}^H\mathbf{Y}, \tag{1.27}$$

where

$$\mathbf{Q}_{\mathsf{LS}} = \left(\mathbf{F}^H\mathbf{X}^H\mathbf{X}\mathbf{F}\right)^{-1}. \tag{1.28}$$

When the channel length is not taken into consideration, both time and frequency domain LS estimates are the same. On the other hand, when the channel length L is known, this can be taken into account and the dimension of matrix $\mathbf{F}$ can be reduced and hence the noise performance is improved due to zeroing the unnecessary values. LS estimation

minimizes $(\mathbf{Y} - \mathbf{XFh})^H (\mathbf{Y} - \mathbf{XFh})$ [18]. It is apparent from the above discussion that the LS channel estimation does not consider channel noise at all and this is the main reason LS estimation performs poorly, especially in the low SNR regions. On the other hand, LS estimation requires no additional information, such as channel or signal statistics, therefore is suitable to use in mobile situations where these statistics frequently change or are not readily available [19].

Use of decision-directed channel estimation approach

Here, using the notation of LS estimation above, I briefly present the DDCE approach. Since block-type pilots are sent periodically, it is necessary to update the channel estimations of the subcarriers until the next block-pilot symbol. DDCE can be used with LS for this purpose. After the reception of the block-pilot, say the n^{th} symbol, the receiver estimates the channel gain of the k^{th} subcarrier as

$$\hat{H}_{k,n} = \frac{Y_{k,n}}{X_{k,n}}. \tag{1.29}$$

This is used as the initial channel estimate. Then, the next symbol, which is a data symbol, is decoded as follows:

$$\hat{X}_{k,n+1} = \frac{Y_{k,n+1}}{\hat{H}_{k,n}}. \tag{1.30}$$

I use the hat notation to indicate that this is an estimation of the data symbol. This estimation is then fed to a *symbol demapper*, where $\hat{X}_{k,n+1}$ is *mapped* to a constellation point. The demapper, for example, could use minimum distance criterion to choose the corresponding constellation point. Denoting this mapped value as $X^{\mathsf{MAP}}_{k,n+1}$, the channel estimation for the $(n+1)^{th}$ symbol can be updated as

$$\hat{H}_{k,n+1} = \frac{Y_{k,n}}{X^{\mathsf{MAP}}_{k,n+1}}. \tag{1.31}$$

Then $\hat{H}_{k,n+1}$ is used to estimate the next data symbol $\hat{X}_{k,n+2}$, and so on. It is clear that this DDCE technique can degrade performance if the channel changes during the symbol time. The symbol estimate $\hat{X}_{k,n+1}$ will be incorrect and the symbol demapper will map it to a different symbol than transmitted. Then the subsequent channel update will be incorrect, and hence sequence of data and channel updates will be erroneous. This error propagation makes DDCE unattractive in fast-varying channel conditions, and comb-type pilots with interpolation/extrapolation gives better performance [18].

Minimum Mean Squared Error Estimation (MMSE)

MMSE is considered to be the optimal, or in practical cases near-optimal channel estimation method. The technique used in OFDM channel estimation is the linear MMSE (LMMSE) and I briefly introduce it here since the proposed channel estimation method in chapter 3 is on the LMMSE-based Kalman filter. LMMSE is optimal in the sense of minimizing the Mean Squared Error (MSE) when AWGN noise is present. Unlike LS estimation, LMMSE uses second order statistics of the channel and the signal and also the operating SNR [16], and thus can be computationally complex as we see later in the section. Due its' importance in channel estimation, I present the definition of LMMSE here.

If $\mathbf{Y} = [y(1), y(2), \ldots, y(N)]$ is the received signal vector and we wish to estimate the vector $\boldsymbol{\theta} = [\theta(1), \theta(2), \ldots, \theta(P)]$, LMMSE finds the coefficients $a_i, i = 1, 2, \ldots, (N+1)$, such that

$$\hat{\theta}_i = \sum_{n=0}^{N-1} a_{i,n} y(n) + a_{i,N} \tag{1.32}$$

that minimizes the MSE

$$MSE\left(\hat{\theta}_i\right) = E\left[\left(\theta_i - \hat{\theta}_i\right)^2\right], \quad i = 1, 2, \ldots, P. \tag{1.33}$$

Then the LMMSE estimator is

$$\hat{\theta}_i = E(\theta_i) + \mathbf{C}_{\theta_i y}\mathbf{C}_{yy}^{-1}(\mathbf{y} - E(\mathbf{y})), \tag{1.34}$$

where $\mathbf{C}$ denotes the correlation function. The minimization over all the scalar θ_i gives the minimum MSE. Therefore, the LMMSE estimator for the vector $\boldsymbol{\theta}$ is

$$\begin{aligned}\hat{\boldsymbol{\theta}} &= \begin{pmatrix} E(\theta_1) \\ E(\theta_2) \\ \vdots \\ E(\theta_P) \end{pmatrix} + \begin{pmatrix} \mathbf{C}_{\theta_1 y}\mathbf{C}_{yy}^{-1}(\mathbf{y} - E(\mathbf{y})) \\ \mathbf{C}_{\theta_2 y}\mathbf{C}_{yy}^{-1}(\mathbf{y} - E(\mathbf{y})) \\ \vdots \\ \mathbf{C}_{\theta_P y}\mathbf{C}_{yy}^{-1}(\mathbf{y} - E(\mathbf{y})) \end{pmatrix} \\ &= E(\boldsymbol{\theta}) + \mathbf{C}_{\theta y}\mathbf{C}_{yy}^{-1}(\mathbf{y} - E(\mathbf{y})).\end{aligned} \tag{1.35}$$

Channel estimation with comb-type pilots

Here I present the mathematics of using LMMSE to estimate the channel with comb-type pilots. In the case of comb-type pilots, assuming the pilot subcarriers are equi-spaced, the received signal at the pilot subcarriers (after DFT) can be written

$$\mathbf{Y} = \mathbf{X}\mathbf{H}_p + \mathbf{W}. \tag{1.36}$$

I assume a total of P pilot subcarriers are present and the $\mathbf{Y}, \mathbf{H}_p$ and $\mathbf{W}$ are vectors of length P, while $\mathbf{X}$ is a $P \times P$ diagonal matrix. The objective of the LMMSE estimator is to estimate the $N \times 1$ vector of subcarriers $\mathbf{H}$. Using the fact that the channel gains are zero-mean Gaussian, I invoke (1.35) to give

$$\hat{\mathbf{H}}_{\mathsf{LMMSE}} = \mathbf{R}_{HY}\mathbf{R}_{YY}^{-1}\mathbf{Y} \tag{1.37}$$

Evaluating the correlation terms gives,

$$\begin{aligned}\mathbf{R}_{HY} &= E\left[(\mathbf{H} - E[\mathbf{H}])(\mathbf{Y} - E[\mathbf{Y}])^H\right] \\ &= E\left[\mathbf{H}(\mathbf{X}\mathbf{H}_p + \mathbf{W})^H\right] \\ &= E\left[\mathbf{H}\mathbf{H}_p^H\mathbf{X}^H + \mathbf{H}\mathbf{W}^H\right] \\ &= \mathbf{X}^H\mathbf{R}_{HH_p}\end{aligned} \tag{1.38}$$

$$\begin{aligned}\mathbf{R}_{YY} &= E\left[(\mathbf{X}\mathbf{H}_p + \mathbf{W})(\mathbf{X}\mathbf{H}_p + \mathbf{W})^H\right] \\ &= E\left[\mathbf{X}\mathbf{H}_p\mathbf{H}_p^H\mathbf{X}^H + \mathbf{X}\mathbf{H}_p\mathbf{W}^H + \mathbf{W}\mathbf{H}_p^H\mathbf{X}^H + \mathbf{W}\mathbf{W}^H\right] \\ &= \mathbf{X}\mathbf{X}^H\mathbf{R}_{H_pH_p} + \sigma_W^2\mathbf{I}\end{aligned} \tag{1.39}$$

Substituting and after some algebraic manipulations we get,

$$\hat{\mathbf{H}}_{\mathsf{LMMSE}} = \mathbf{R}_{HH_p}\left(\mathbf{R}_{H_pH_p} + \sigma_W^2\left(\mathbf{X}\mathbf{X}^H\right)^{-1}\right)^{-1}\hat{\mathbf{H}}_{\mathsf{LS}}, \tag{1.40}$$

where $\hat{\mathbf{H}}_{\mathsf{LS}} = \mathbf{X}^{-1}\mathbf{Y}$ is the LS estimation of the pilot subcarriers. We see in (1.40) that LMMSE estimation takes subcarrier frequency correlation and also noise statistics in to account. For this reason LMMSE gives better performance compared to LS estimation, especially in the low SNR region.

Complexity reduction of LMMSE

The superior performance of LMMSE over LS estimation comes at a computational cost. The matrix inversion $\left(\mathbf{X}\mathbf{X}^H\right)^{-1}$ of (1.40) is computationally prohibitive for number of reasons: 1) The matrix dimension increases with the increasing number of subcarriers, 2) the inversion needs to be performed each time $\mathbf{X}$ changes. A simple yet effective solution is presented in [20] for the block-type pilot case, where the expectation of $\left(\mathbf{X}\mathbf{X}^H\right)^{-1}$ is used instead. On the assumption that same constellation is used on all tones, which is a proper

assumption [17], the expectation of the inverse can be written

$$E\left[\left(\mathbf{X}\mathbf{X}^H\right)^{-1}\right] = E\left[|1/x_k|^2\right]\mathbf{I}, \tag{1.41}$$

where x_k is dependent on the chosen constellation. Then the modified LMMSE estimator is given by

$$\hat{\mathbf{H}}_{\mathsf{LMMSE}} = \mathbf{R}_{HH}\left(\mathbf{R}_{HH} + \frac{\beta}{SNR}\mathbf{I}\right)^{-1}\hat{\mathbf{H}}_{\mathsf{LS}}, \tag{1.42}$$

where

$$SNR = \frac{E\left[|x_k|^2\right]}{\sigma_W^2}, \tag{1.43}$$

and

$$\beta = E\left[|x_k|^2\right]E\left[|1/x_k|^2\right]. \tag{1.44}$$

As can be seen from (1.42), the LMMSE estimation is no longer a function of $\mathbf{X}$ and the channel correlation $\mathbf{R}_{HH}$ is a function of the channel power-delay-profile (PDP) [16] and thus changes very slowly. Therefore, the argument of the inversion does not change as long as the operating SNR stays constant. This modification brings about a substantial reduction in computational complexity. Nevertheless, the expression still needs to be recalculated whenever the channel PDP changes, or the more frequent case of change of operating SNR.

To further simplify the calculation, a singular-value-decomposition (SVD) based method can be used [20]. The SVD method is used on the realization that the number of significant singular values of the channel cross-correlation matrix is related the the number of significant channel taps of the channel impulse response. The number of significant channel taps are much less than the number of subcarriers and this fact has been used to simplify the estimation further. Writing the SVD of $\mathbf{R}_{HH}$ as

$$\mathbf{R}_{HH} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^H, \tag{1.45}$$

where $\mathbf{U}$ is a unitary matrix and $\mathbf{\Lambda}$ is the diagonal matrix of singular values $\lambda_0, \lambda_1, \ldots, \lambda_{N-1}$ in descending order, where N is the number of (pilot) subcarriers. Then the LMMSE estimator is shown to be,

$$\hat{\mathbf{H}}_{\mathsf{LMMSE}} = \mathbf{U}\mathbf{\Delta}\mathbf{U}^H\hat{\mathbf{H}}_{\mathsf{LS}}, \tag{1.46}$$

where $\mathbf{\Delta}$ is the diagonal matrix where the entries are given by

$$\delta_{i,i} = \begin{cases} \frac{\lambda_i}{\lambda_i + \frac{\beta}{SNR}} & i = 0, 1, 2, \ldots, p-1 \\ 0 & \imath = 0, p, p+1, \ldots, N-1 \end{cases} \tag{1.47}$$

Here p is the number of significant singular values. Since p is dependent on the number of long-term significant channel taps, this decomposition of the channel correlation function reduces the computation substantially due to the much less number of channel taps compared to the number of subcarriers.

1.4.3 Sequential LMMSE Channel Estimation for OFDM - Kalman Filtering

In case of OFDM, sequential estimation cannot be done in the conventional sense, since the receiver needs to wait for all the N samples to send to the DFT block. But channel estimation in OFDM can benefit from the sequential LMMSE procedure by realizing the channel variation as a Gauss-Markov model. The method is known as Kalman filtering. Kalman filers are an extension of the LMMSE filters but avoids the disadvantages of complex LMMSE estimators.

The first step is to model the channel, which is the unknown vector to be estimated, as a Gauss-Markov model as follows:

$$\mathbf{H}_n = \mathbf{A}\mathbf{H}_{n-1} + \mathbf{D}_n \tag{1.48}$$

Assuming we are estimating N subcarriers, $\mathbf{H}_n$ is a $N \times 1$ vector of subcarrier gains at time instant n, and $\mathbf{A}$ is $N \times N$ matrix which relates the subcarrier gains of the previous instant, $n-1$, to the current instant, n. This matrix is called the *state-transition* matrix, as it transits the previous state of the channel to the current state. The additive noise vector $\mathbf{D}_n$ are white Gaussing noise components with a PDF $D_n \sim \mathbf{N}(0, \mathbf{Q}_n)$. Q_n is its' covariance matrix and D_n is considered independent for each time instant. The received signal is

$$\mathbf{Y}_n = \mathbf{X}_n \mathbf{H}_n + \mathbf{W}_n. \tag{1.49}$$

The received vector $\mathbf{Y}_n$ is of dimension $N \times 1$ and $\mathbf{X}_n$ is the $N \times N$ diagonal matrix of pilot symbols and $\mathbf{W}_n$ is the $N \times N$ diagonal matrix of the AWGN noise with covariance matrix $\mathbf{R}_{WW} = \sigma_W^2 \mathbf{I}$. The process of estimating the channel by way of Kalman filtering is as follows:

- **Prediction** - Similar to the sequential LMMSE, Kalman filtering process starts by predicting the current vector to be estimated, $\mathbf{H}_n$.

  $$\hat{\mathbf{H}}_{n|n-1} = \mathbf{A}\hat{\mathbf{H}}_{n-1|n-1}. \tag{1.50}$$

  Here the transition matrix is used to predict the current values by the previously estimated values.

- **Minimum prediction MSE matrix** - This step involves predicting the minimum MSE from the previous value.

  $$\mathbf{M}_{n|n-1} = \mathbf{A}\mathbf{M}_{n-1|n-1}\mathbf{A}^T + \mathbf{Q}_n. \tag{1.51}$$

  MSE's are defined as

  $$\begin{aligned} \mathbf{M}_{n|n-1} &= E\left[\left(\mathbf{H}_n - \hat{\mathbf{H}}_{n|n-1}\right)^2\right] \\ \mathbf{M}_{n|n} &= E\left[\left(\mathbf{H}_n - \hat{\mathbf{H}}_{n|n}\right)^2\right] \end{aligned} \tag{1.52}$$

- **Kalman gain matrix** - The important calculation. The Kalman gain matrix is determined as

$$\mathbf{K}_n = \mathbf{M}_{n|n-1}\mathbf{X}_n^T \left(\sigma_W^2\mathbf{I} + \mathbf{X}_n\mathbf{M}_{n|n-1}\mathbf{X}_n^T\right)^{-1}. \tag{1.53}$$

  This step is similar to the gain correction of the sequential LMMSE estimator.

- **Correction** - This is the step where the values are estimated, i.e. the filter output. It is given by

$$\hat{\mathbf{H}}_{n|n} = \hat{\mathbf{H}}_{n|n-1} + \mathbf{K}_n\left(\mathbf{Y}_n - \mathbf{X}_n\hat{\mathbf{H}}_{n|n-1}\right). \tag{1.54}$$

  Recall from sequential LMMSE that, the value inside the parentheses, $\left(\mathbf{Y}_n - \mathbf{X}_n\hat{\mathbf{H}}_{n|n-1}\right)$, is the *innovation*. Since $\hat{\mathbf{H}}_{n|n-1}$ is the predicted channel gain vector, $\mathbf{X}_n\hat{\mathbf{H}}_{n|n-1}$ is the predicted current data, or $\hat{\mathbf{Y}}_{n|n-1}$. Therefore, the correction step is multiplying the innovation by the Kalman gain and then adding it to the predicted channel gains vector.

- **Minimum MSE matrix** - Second step of Kalman filtering is predicting the minimum MSE. In this final step, the predicted MSE of step two is used to calculate the current MSE.

$$\mathbf{M}_{n|n} = \left(\mathbf{I} - \mathbf{K}_n\mathbf{X}_n\right)\mathbf{M}_{n|n-1} \tag{1.55}$$

The five steps above are the Kalman filtering and it can be initialized by the values $\hat{\mathbf{H}}_{-1|-1} = E[\mathbf{H}]$ and $\mathbf{M}_{-1|-1} = \mathbf{R}_{HH}$, or simply initialize to zero. The initial value affects the convergence period of the filter, an issue I tackle when I discuss the proposed modified Kalman filter.

Things to note about the Kalman filter are,

- It does not need the channel covariance, $\mathbf{R}_{HH}$ (except for initialization, where zero can be used).
- Unlike LMMSE, matrix inversion is required for the Kalman gain calculation.
- An Auto Regressive (AR) model can be used to model the behavior of the channel by any order-p.
- Although channel covariance matrix is not required, variance of the driving-noise is needed. This could not be available at the filter and this issue is considered in the proposed Kalman filter in chapter 3

1.4.4 Channel Estimation and Resource Allocation in OFDM Systems

As mentioned in the beginning of this chapter, channel estimation is an important process for wireless communications, including resource allocation. In this last section I provide a brief discussion on the effects of channel estimation with simulated figures. Figs. 1.9- 1.15 show these results with figures obtained by simulating an example scenario.

Resource allocation, as I discussed in the earlier chapters, is allocating system resources, which includes not only power and subcarriers but also modulation and coding levels, such that the users in the system can be provided with their required QoS. If the number of users in the system is small or there is an abundance of transmission power, resource allocation is a very easy task and channel estimations would not be critical information. Unfortunately, this is never the case, especially in broadband wireless systems. Resources are scarce and the channel estimations are critical for a sustainable system.

As an example, Fig. 1.9 shows a simulation of Rayleigh fading channel for 500 symbol durations with a Doppler frequency of 100Hz. The dashed lines show the estimated channel from a conventional Kalman filter at a SNR of 15dB. Clearly the channel is not

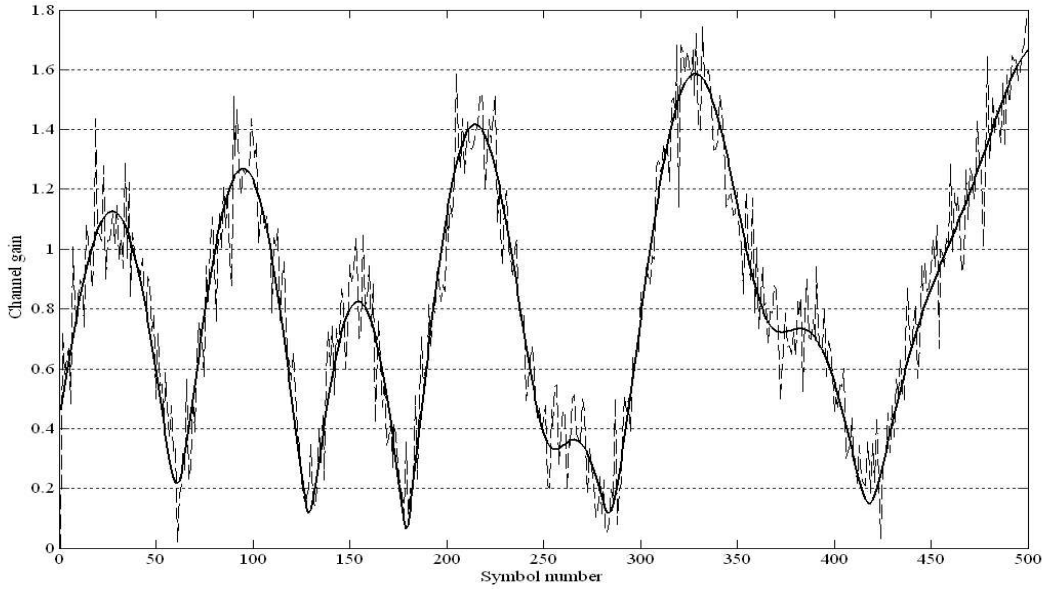

Figure 1.9: Channel estimation of a Rayleigh faded channel with Kalman filter

perfectly estimated. It is an impossible task. At some symbol instants the channel is estimated higher than its' real value, while at other times it is estimated to be less. It is important that the channel be estimated as close to the true channel as possible. As I discussed earlier in the section 1.3 on resource allocation techniques such as water-filling algorithm, the channel estimates are always considered to be perfect, which is hardly the case. The impact of these imperfect channel estimations are two-fold: First, when the channel is over-estimated, the resource allocation algorithm allocates a lower transmission power than required, or a higher modulation level than possible, resulting in increased BER at the receiver. Secondly, when the channel is under-estimated, more power than necessary is allocated to the subcarrier, wasting transmission power, which could be allocated on a different subcarrier to transmit more data.

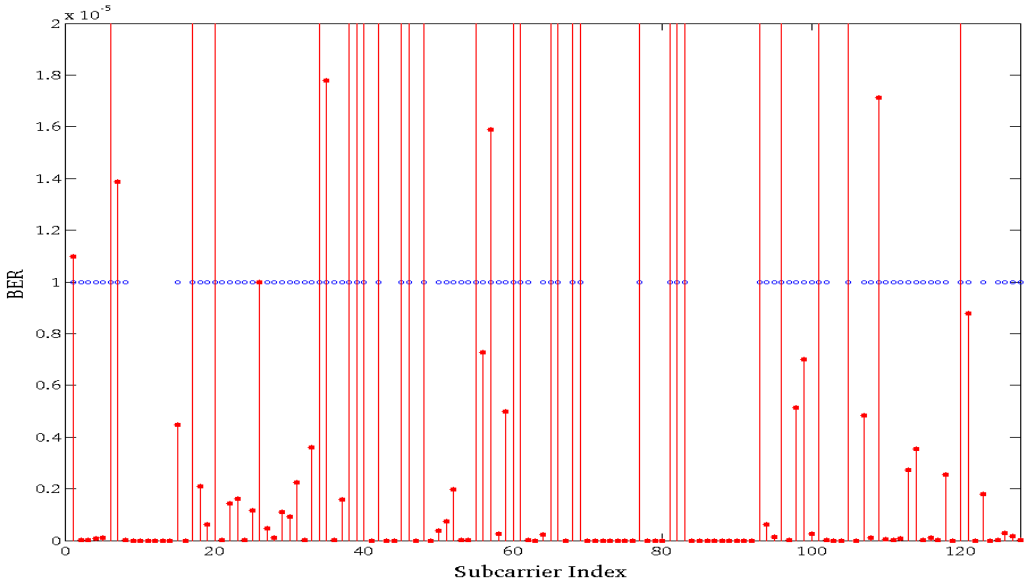

Figure 1.10: Received BER for a 128 subcarrier system under distorted channel estimations

1.4.5 Channel Estimation and BER

An example scenario is shown in Fig. 1.10 for a 128 subcarrier system with a required BER of 10^{-4}. The blue dots shows the allocated subcarriers and the required BER, while the red dots show the received BER. The resource allocation strategy used here is water-filling. It is evident that while some subcarriers received a better BER, others are worse than the expected BER. Depending on the application used, a worse than required BER would mean that the data packet will need to be re-transmitted, further wasting system resources.

Channel estimation is important for coherent detection and becomes crucial for higher modulation levels. Fig. 1.11 shows constellation captures for 512 subcarriers with 16-QAM. The signal is normalized for unity mean energy at the transmitter. Fig. 1.12 shows a similar constellation capture for the 64-QAM level. Both these modulation levels are used in the IEEE802.16 standard on the downlink [17].

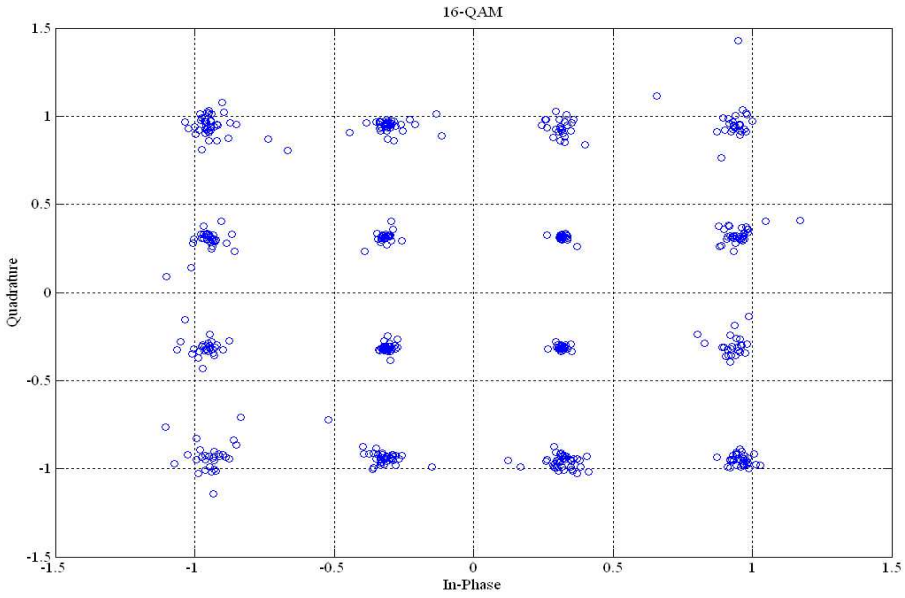

Figure 1.11: Received constellation of a 16-QAM under channel estimation errors

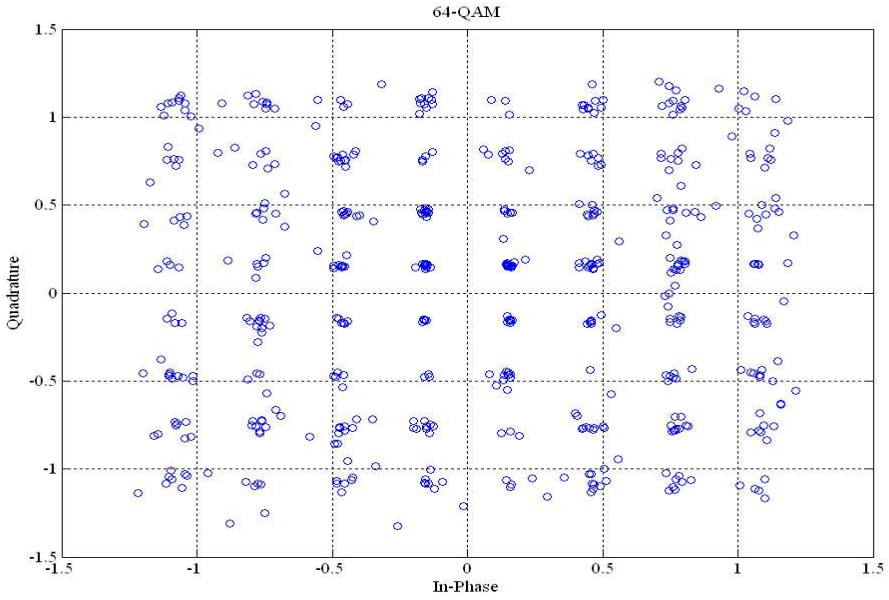

Figure 1.12: Received constellation of a 64-QAM under channel estimation errors

In both the cases the complex channels have been distorted with an additive Gaussian noise to reflect channel estimation errors. And it is apparent how accurate channel estimations are important, especially for higher modulation levels. The 64-QAM constellation capture shows that more received points spread in to neighbor decision regions, increasing BER.

1.4.6 Channel Estimation Complexity and Frequency

Channel estimation is a complex process. The complexity comes from both the amount of required statistical information, as is for the case with LMMSE, and also from the number of calculations to perform [21], such as matrix inversions. Channel estimation is a process that would be required to perform on each received symbol. On the downlink, depending on the Down Link Map (DL-MAP) [17], the receiver will need to estimate channels on the symbols/subcarriers destined for itself. For the IEEE802.16e standard, an OFDM symbol has a duration of $102.9\mu s$. It is extremely difficult to perform complex mathematics in this small time duration, especially because it is not possible to implement these mathematical functions in hardware. Therefore, channel estimations are usually carried out once per number of symbols, and the channel in-between the estimations are obtained through interpolations. Since our proposed scheme presented in chapter 3 is based on Kalman filtering, here I show two examples of using a conventional Kalman filter to estimate channel with 5 symbols apart.

Fig. 1.13 show a channel with a Doppler frequency of 10Hz while a 150Hz scenario is depicted in Fig. 1.15. The blue dashed line shows the true channel, red line shows the channel estimation through the Kalman filter in each symbol, and the black line is when the channel is estimated every 5 symbols. Between channel estimations, the previous channel estimation is used. As expected, the higher Doppler rate requires more frequent channel

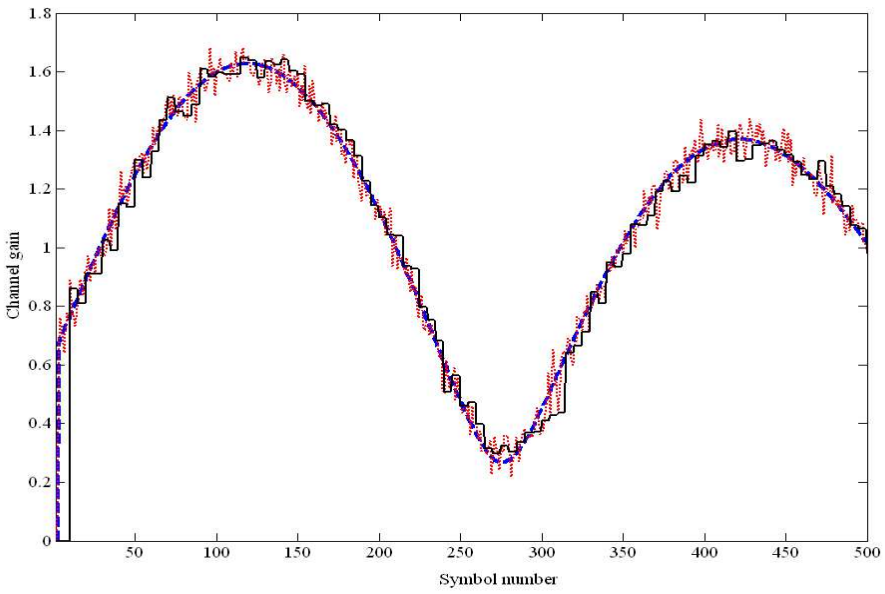

Figure 1.13: Channel estimation every 5 symbols with Kalman filter - 10Hz

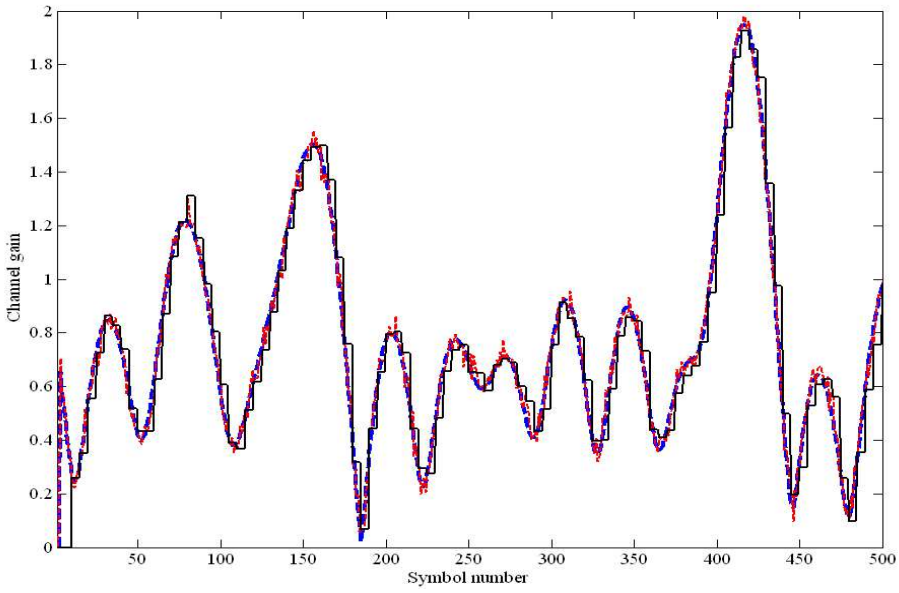

Figure 1.14: Channel estimation every 5 symbols with Kalman filter - 150Hz

estimations to track the fast variations of the channel, which is consistent with (1.21) introduced in sec. 1.4.1.

Therefore, it is important to find techniques which offer reduced complexity in the estimation process. Kalman filter is less complex in the fact that it requires less second-order statistics but it still requires a matrix inversion. In chapter 3 I propose a modified Kalman filter which takes into account certain channel and pilot signal properties to reduce the calculation complexity of the channel estimation, and in the process also increases the performance over the conventional Kalman filter.

1.5 Position of the Study

This section summarizes the position of the studies included in the dissertaion. It provides an overview of the motivation for the study, the proposed scheme and the performance of the scheme.

Table 1.1: Problems of existing schemes and the contribution of the proposed schemes

Chapter 2	Topic	Low complexity resource allocation algorithm by multiple attribute weighing and user ranking for OFDMA systems
	Problems of existing research	Existing schemes are either highly complex in calculations or iterative in nature which have a longer convergence period. These schemes can be practically not feasible to implement since resource allocation needs to be performed frame-by-frame.
	Proposed method	Users in the system are ranked according to a multi-attribute criterion which identifies users priority. The attributes can be defined according to the required performance metric.
	Effect of proposed scheme	Competitive results are obtained against other low-complexity resource allocation schemes.
Chapter 3	Topic	Steady-state Kalman filtering for channel estimation in OFDM systems for Rayleigh Fading Channels
	Problems of existing research	Conventional Kalman filters have a five-step process which needs to performed in each filtering iteration and the matrix inversion in the Kalman gain calculation step becomes a calculation burden.
	Proposed method	A steady-state Kalman filter is proposed, which eliminates the transient period and operates at the converged state, enhancing performance. Kalman gain calculation is performed only once for the duration of the channel stationarity.
	Effect of proposed scheme	The conventional five-step process is reduced to two. Kalman gain is calculated only once for the duration of the channel stationarity, which substantially reduces the number of calculations. Dynamic calculation of the channel driving noise keeps the filter operating at the optimal state.
Chapter 4	Topic	Analysis Quantization Noise in an End-to-End OFDM Link
	Problems of existing research	Quantization noise is often taken as having an uniformly distributed PDF. But theory shows this condition is only achieved when certain conditions are met and it is mainly a function of the input signal PDF.
	Proposed method	An analysis is performed to examine the statistical properties of the signal in an OFDM link, from transmitter to receiver. Signal statistics are obtained as functions of transceiver and channel parameters.
	Effect of proposed scheme	The theoretical derivation shows strong agreement to simulated results, meaning that the derived results can be used in a system to determine the quantization noise at the receiver as a function of the transceiver and channel parameters. Using the results, optimal receiver conditions can be chosen accordingly.

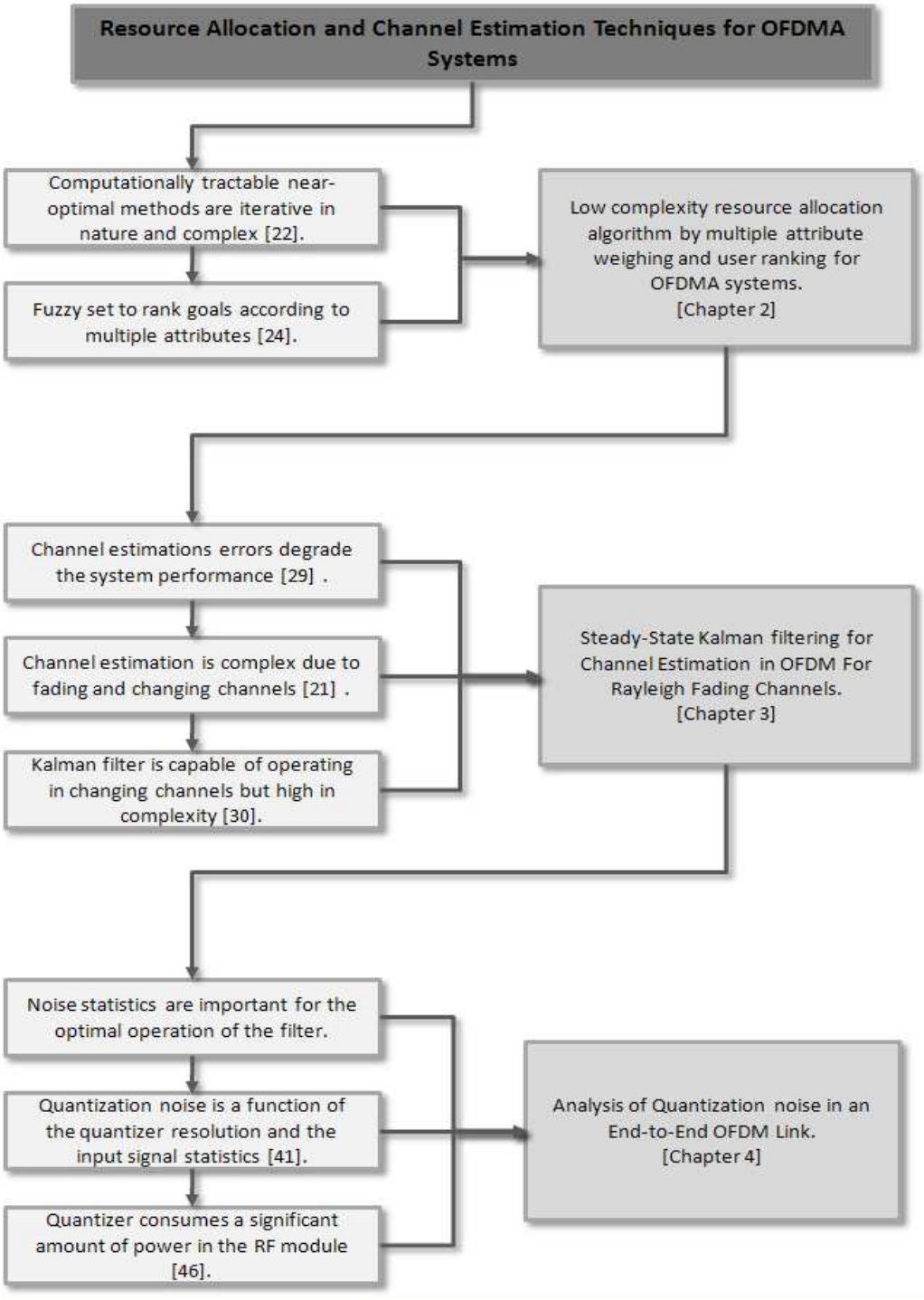

Figure 1.15: Chapter flow of the thesis

Chapter 2

Low complexity resource allocation algorithm by multiple attribute weighing and user ranking for OFDMA systems

Proposed low complexity resource allocation algorithm for OFDMA systems is introduced in this chapter. As discussed in the previous chapters, an optimal solution for resource allocation in OFDMA systems is mathematically heavy in complexity and hence is not feasible to use in practical systems. Our motivation in the proposed scheme is to develop an algorithm for subcarrier allocation in OFDMA systems without complex calculations and many iterative loops which makes the algorithms inappropriate real-time resource allocation. I make an entrance to the proposed scheme by discussing the channel transfer function characteristics, which will be the main consideration in our proposed scheme. Then I discuss about user ranking, different user attributes and multiple attribute

decision making. Finally I introduce our proposed resource allocation algorithms.

2.1 Introduction

Different resource allocation techniques have been proposed. In [12], Wong et al introduce the idea of using Lagrangian Relaxation (LR) technique to carrier allocation problem of the multiuser OFDM, which I discussed in section 1.3.4. The authors assume a constant data rate for users. Data rate is taken to be a constraint and the scheme is optimized for minimum power. This is a combinatorial minimization where the variables are relaxed to take values [0, 1] to make the calculations feasible. In [22] Ergen et al propose a fair resource allocation scheme together with two improvement modules. The subcarriers and bits are allocated by taking users in order and allocating the subcarrier which needs the least power to transmit data at the highest modulation level. The modulation levels are decremented according to the maximum power constraint. System load is not considered and user dropping is not mentioned explicitly in this algorithm. This allocation takes an iterative approach until the rate requirements are satisfied for the given power constraint. After the resources are allocated the scheme runs two improvements modules to swap bits and subcarriers between users to reduce power. These algorithms are iterative and become very exhaustive with increasing user or subcarrier numbers.

In practical systems, the approach taken is less complicated than the schemes mentioned above. In IEEE802.16 WirelessMAN standard two types of subcarrier allocation schemes are employed [23]: (1) Distributed subcarrier permutation and (2) Adjacent subcarrier permutation. In the Distributed subcarrier permutation allocation scheme, subcarriers are chosen according to a permutation algorithm and grouped together to form subchannels. The subcarriers are chosen to a subchannel in a manner that the channel

diversity is exploited. Therefore fast fading channel users can have advantage in distributed subcarrier permutation scheme. In the adjacent subcarrier permutation scheme, adjacent subcarriers are grouped together to form subchannels. This method is simple and effective for slow fading channel users. After subchannelization, these subchannels are assigned to different users in the system. These schemes give less computational overhead and make it feasible to adopt in real-time communications. Although these methods are simple, because of the reduced optimality, system performances such as throughput or power levels might be degraded.

Existing schemes, such as mentioned above, need complex iterations for the algorithms to converge. To avoid the iterative looping in the algorithm, it is desirable to introduce a ranking to the users depending on different characteristics of their channel transfer function and take users according to their rank for carrier allocation. Use of different channel characteristics (attributes as I will call it later) can be justified as follows: For example, one user will have a relatively small number of high gain subcarriers and large portion of average to low gain subcarriers, while another user will have slowly varying spectrum of relatively good subcarrier as shown in Fig. 2.1. And as depicted in the figure by the dashed lines, both two users have almost same average channel gain. Due to this effect of diversity, user with the low number of good subcarriers should be given priority to choose subcarriers, because the selection of subcarriers available to that user is less than that of the other user. This fact motivated us to exploit the channel characteristics more when determining the order of the users taken for subcarrier allocation. I take an effort to exploit these different attributes of users' channel characteristics, as explained later, and determine how important each user needs to get his best subcarriers to reduce the overall transmission power. Another target is the reduction of the complexity of the existing schemes. Most of the conventional proposed schemes need intensive search and swap algorithms to con-

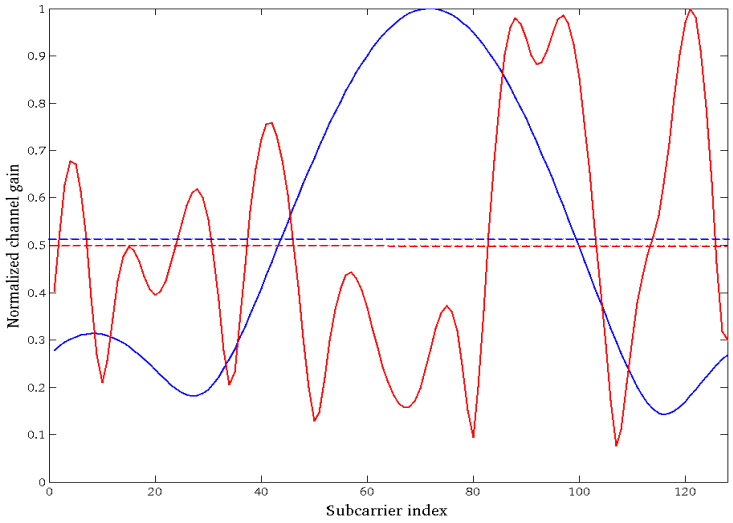

Figure 2.1: Diversity of channel characteristics

verge. The main reason for this is because when subcarriers are allocated to users, they are allocated according to either their average channel gain or highest channel gain or by taking users/subcarriers in an order. This requires the need for searching and swapping subcarriers between users to reduce the transmission power or provide the required QoS, because subcarriers are allocated in a simple manner in the first assignment. Therefore it is important to find a scheme which could assign subcarriers to users in a more efficient but yet simple manner.

I propose a computationally efficient sub-optimal subcarrier allocation algorithm for an OFDMA system by avoiding complex iterations and calculations used in classical water-filling algorithms. Our scheme adopts a user ranking method where I give a value of priority for each user in the system. The rank of a user indication of how important it is for that particular user to get priority over other users when obtaining subcarriers. I use characteristics such as carrier gain decrease rate and deviation from the mean channel gain of the system in determining rank for each user. Then I use a multiple attribute weighing

mechanism to combine the different values obtained for each user and arrive at a single value, the *rank*. Our proposed scheme is simulated in comparison with other mathematically efficient subcarrier allocation schemes as well as with the conventional greedy allocation schemes. It is shown that the proposed method demonstrates competitive results with the conventional schemes.

2.2 User ranking algorithm

Consider a system with M users and N subcarriers. $H_i(n)$ indicates channel gain of user i in subcarrier n. Channel gain $H_i(n)$ is indicated as the amplitude. Estimation of $H_i(n)$ can be determined by the received power, but in the scheme I assume that the base station is capable of predicting the frequency response in amplitude of each user ahead of scheduling transmissions or that the available channel estimation is valid. C_{max} denotes the maximum allowable number of bits on a subcarrier and C_i is the total required number of bits by user i.

2.2.1 User Attributes

As was discussed in the introduction, average channel gain of a user does not convey enough information. To understand a users' channel as possibly as I can, in this paper I define three attributes (goals):

- Average channel gain
- Carrier gain decrease rate
- Variation from mean system channel gain

By defining these three attributes of a users channel, I try to understand the channel of each user better, and determine how important each user needs to get priority in choosing

subcarriers.

Average channel gain - W_1

This measurement is chosen to give a fair overall indication of the users channel.

$$W_{i,1} = \frac{\sum_{n=1}^{N}}{N}, \quad i = 1, 2, \ldots, M. \tag{2.1}$$

Carrier gain decrease rate - W_2

In this attribute I try to get an indication of the rate of decrease of a users channel gain. Here, sort the users channel gains in decreasing order and check how much the channel gain drops from the highest gain carrier to some constant number of subcarriers. Here, I can get an indication of how many good carriers a user has. If the value is large, it indicates there is a large drop in gain and thus the user has only few good carriers, and if it is a low value that indicates the user has fair amount of high gain carriers. With this I am able to give preference to users who have only few good carriers, because these users need to be allocated to the available few good subcarriers.

begin

Initialize

$\forall i, \Omega \longleftarrow \{\}$

for $k := 1$ to N do

$\text{sort}_i(k) = \max H_i(n), \quad n \notin \Omega$

$\Omega \longleftarrow n$

end

end

$$W_{i,2} = \mathsf{sort}_i(k) = \mathsf{sort}_i(1) - \mathsf{sort}_i(\mathsf{const}), \quad \forall i, \tag{2.2}$$

where *const* is chosen in the simulation.

Variation from mean system channel gain - W_3

This attribute is defined to give a measurement of an individual users channel profile in relative to the channels of all the users in the system. As defined in (2.1), $W_{i,1}$ used in attribute W_1 indicates individual users average channel gain over N subcarriers at the base station.

Eq. 2.3 is the average channel gain of all the M users in the system. Therefore, I make use of attribute W_3 to determine which users have more deviation from ξ and give these users more priority. This is done by the non-linear shape of the bell function expressed in (2.4). Fig. 2.2 shows a graphical representation of the bell function in (2.4). As can be seen from the plotted curves, variable m controls the vertical steepness of the graph while n controls the horizontal flatness of the graph. I used values of -2 and -1 for m and n respectively in the simulations.

$$\xi = \frac{\sum_{i=1}^{M} W_{i,1}}{M} \tag{2.3}$$

$$W_{i,3} = \begin{cases} \left(1 + (W_{i,1} - \xi)^{-m}\right)^{-n} & \text{if} \quad W_{i,1} \leq \xi \\ -\left(1 + (W_{i,1} - \xi)^{-m}\right)^{-n} & \text{if} \quad W_{i,1} > \xi \end{cases} \tag{2.4}$$

Using a bell function as above and changing its curvature with provides the ability to place a certain users average channel relative to all the other users in the system. As discussed previously, a user's average frequency response does not provide an adequate indication of the users distribution of the channel gains. But when the average frequency response deviates considerably from the system average, it provides an important indication of that

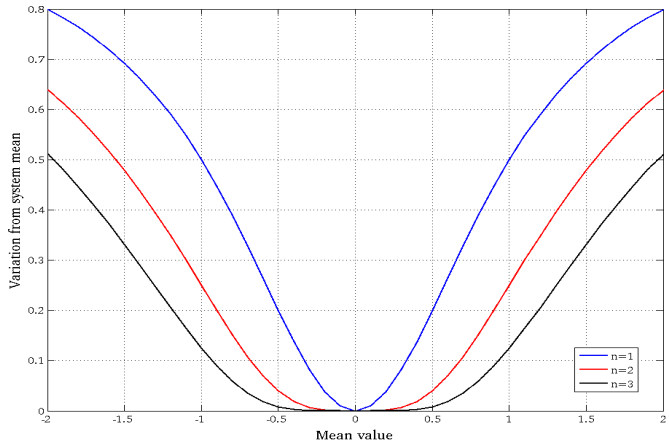

Figure 2.2: Graphical representation of the bell function

channel. If a users average frequency response deviates considerably to the either end, it could mean that the user has a very poor channel response, or the users has a very good channel response. The latter case, for example, could happen when a user is very close to the base station or a line-of-sight path exists from the transmitter. In these cases, those particular users must be given priority accordingly. If a user has a very poor channel response, he should be give priority over other users who have average channel response relative to the system. Similarly, for a user with a very good channel response compared to the system average, does not need to get priority because the subcarriers are on average much better than most of the users in the system. This differentiability is the function I seek by using the bell function in attribute W_3.

These attributes are then used to construct the $M \times 3$ *goal membership* value matrix G:

$$G_{i,j} = W_{i,j}, \quad i = 1, 2, \ldots, M, j = 1, 2, 3. \tag{2.5}$$

This matrix indicates how well or how weakly each user attains the goals.

2.2.2 User Information Feedback

The three attributes defined earlier are used to get a better understanding of the users channel frequency response. Although I am using three attributes, the algorithm only requires the information of each users channel frequency response. Then the attributes are calculated by using the frequency response. The users only feedback the channel frequency response only. The base station can then calculate all three attributes from the frequency response of every user.In the proposed scheme the users feedback their channel frequency response only. The attributes are calculated purely from each users frequency response. If more additional information can be sent, the algorithm can use this information for better allocation strategies by defining new attributes to utilize the new information. For example, if users send their mobility speeds, or the type of service in use (ex: VoIP, media streaming, browser data, etc.), the algorithm can make better choices by using these information. This also gives the opportunity for sustainable resource allocation method, for example, instead of allocating resources independently in each symbol, user QoS, and delay constraints, etc., can be utilized for a better resource allocation strategy, but increases the amount of feedback required from the users.

Quantization of feedback information and be performed to reduce the amount of user channel information sent to base station. In this proposed scheme quantization is not considered. As is the case with many resource allocation schemes, I assumed that the base station has perfect channel frequency response of all the users in the system. If user channel state information is quantized, the effect of it would be two-fold. First, since attributes are calculated using the frequency response, quantized frequency response values would not yield accurate rankings among users. Although quantized frequency responses will affect the attribute calculations, since every users' frequency response is mutually independent, provided the quantization resolution is high enough, the impact of the quantization would

not reflect strongly on the final rankings. Secondly, although the ranking would not be affected substantially by the quantization of channel state information, when data and power are allocated to the subcarriers, if accurate channel frequency response is not available, degradation could occur as I have demonstrated in sections 1.4.4 and 1.4.5 of the thesis. Therefore, in simulations it is important to isolate these two phenomena to obtain a proper quantification of the effect quantization of channel state information would have on the resource allocation scheme.

2.2.3 Multiple Attribute Decision Making and Goal Weighing

Multiple attribute decision making is a topic discussed in *fuzzy set theory*. This concept is used in situations to find an optimal alternative for a number of given goals [24] [25]. I consider utilizing the concept of multiple attribute decision making in our scheme. I use the attributes defined in section 2.2.1 as the multiple attributes and make a decision depending on the values; that is find a rank.

Another method incorporated in the scheme is to individually weigh the importance of each attribute. Saaty [24] proposes a scheme for the goal weighing and Yager [25] improves it to be used in multiple attribute decision making. The attributes considered in the scheme are not equally important. Some are important than others depending on different factors. By using Saatys method in our scheme, I am able to further improve the effectiveness by assigning different weights to each goal. Another reason for using the goal weighing scheme is that, the attributes used in the scheme can be changed to suit different situations (ex. different QoS policies). And by this goal weighing scheme, you can control different goals priority. For example, in our algorithm, our objective is to reduce transmission power. But in a different scheme, the objective would be to give priority to a special user class and can change the weights accordingly to obtain the necessary goals.

The weights therefore, are defined according to the attributes. They reflect the relative importance of attributes and not the user requirements. If satisfying user requirements are the final objective of the algorithm, then the attributes should be defined accordingly to reflect this fact and then weights are determined for the attributes. Therefore, the weights are not directly related to the user requirements, but are related through the system attributes. The goal weights are calculated as follows: First construct a 3-by-3 weight matrix WM which indicates the relative importance of each goal. Hence $WM_{i,j}$ indicates the importance of goal i compared to goal j and $WM_{j,i} = \frac{1}{WM_{i,j}}$. Therefore WM is a diagonal matrix with upper diagonal entries being the reciprocal of lower diagonal matrix. The value of $WM_{i,j}$ is taken between 1 to 9 with 9 being the most important. After obtaining WM, I construct the column matrix WM_I. WM_I is the column matrix of the eigenvector matrix of WM which corresponds to the highest eigenvalue. WM_I is then normalized to 1. After normalization, WM_I represents the weights of the three goals [25].

2.2.4 Data Commensuration

After constructing G, it is necessary to normalize the membership values because they are generally incommensurate. Different attributes have different value ranges, hence need to shift them to a common range of values. For data commensuration I use the method used in [26] for *benefit criterion*:

$$G_{i,j} = \frac{G_{i,j} - G_{\mathsf{max}}}{G_{\mathsf{max}} - G_{\mathsf{min}}}, \tag{2.6}$$

where G_{min} and G_{max} are minimum and maximum values of the matrix, respectively.

2.2.5 Rank Determination

Then move onto rank the users according to their membership values in matrix G. After commensuration of the data in matrix G, I need to take the power of each of these

membership values with the normalized weight matrix values in WM_I. Then to obtain the rank matrix R, intersect the values by row in matrix G: the minimum value on each row should be chosen. This is illustrated in Fig. 2.3 where R_i is a column matrix of M entries,

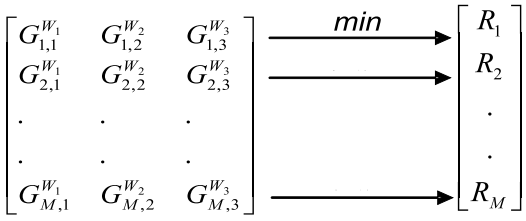

Figure 2.3: Rank determination of users

each value corresponding to a user. These entry values of R_i are the ranks of users.

$$R_i = \min G_{i,j}^{W_j}, \quad i = 1, 2, \ldots, M, \quad j = 1, 2, 3. \tag{2.7}$$

Higher membership value gives that user a higher rank or a higher priority. I will then arrange the rank values in descending order and call it $Ra_i, i = 1, \ldots, M$. Therefore Ra_1 has the highest rank value and Ra_M has the lowest rank value. Next the subcarrier allocation algorithm is described.

2.3 Algorithms for the Proposed Scheme

In our scheme, when the system is overloaded and needs to drop users, I drop users who have both low average channel gains and low data requirements. By dropping users with low average gain the overall system transmit power can be reduced, while dropping users with low data rate helps to effectively utilize the bandwidth. I use a similar approach to determining user drop ranks as was with R_i. The goals are channel gain and data rate, and the weights are taken as unity for both goals. I call the users drop rank as UD_i.

2.3.1 Proposed Subcarrier Allocation Algorithm

Figs. 2.4∼2.6 show the proposed subcarrier allocation algorithm. In the first phase, shown in Fig. 2.4, total system subcarrier requirement is calculated, and if the system is overloaded users are dropped according to their user drop rank. During the next phase (Fig. 2.5), subcarriers are allocated to users according to their rank and in the final phase (Fig. 2.6), available unallocated subcarriers are assigned to users.

Subcarrier allocation algorithm in Figs. 2.5 and 2.6 converges with a maximum number of iterations equal to the amount of system subcarriers. In the above subcarrier allocation algorithm, I take users in decreasing order of rank and assign the required carriers. The shape of the users channel frequency response depends on the multipath Rayleigh fading characteristic of the channel. When a users channel profile is changing slowly as shown in Fig 2.8, the user can request a continuous block of subcarriers since many high gain subcarriers tend to be grouped closer. When the number of users increases, channel amplitudes might be correlated more and many users may have good subcarriers in the same band. Thus, by allowing one user to take all the best subcarriers it needs, other users who have similar channel amplitudes will be left with no better subcarriers. Therefore, to give fairness to users, the subcarrier allocation to users is done in several iterations, i.e. in first iteration, take a user according to its rank, and allocate a fraction of the subcarriers it needs and then proceed to the next user. The optimum number of iterations depends on number of users and user channel conditions and in the simulations I used two iterations.

In the final part of the algorithm I allocate unassigned carriers to users. In our proposed scheme, to allocate the unallocated carriers, users are taken from the reverse order of ranks; they are the users who are taken from the lowest rank. Because I assign the required minimum subcarriers to users depending on the highest rank users, the users who are low ranked might have been unable to get most of their best carriers. This is more

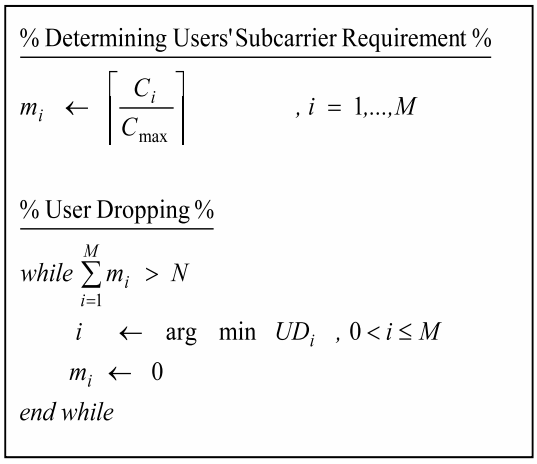

Figure 2.4: User dropping if system overloaded

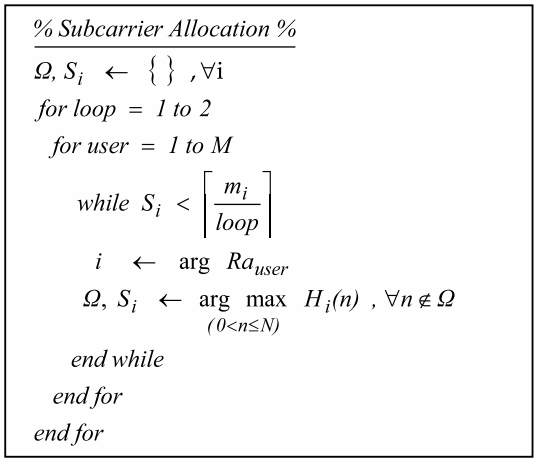

Figure 2.5: Subcarrier allocation

% Unallocated Subcarrier Assignment %

$$\textit{while} \sum_{i=1}^{M} m_i < N$$

for rank = 1 to M

$$i \leftarrow \arg\ Ra_{rank}$$

$$\Omega, S_i \leftarrow \arg \max_{(0<n\leq N)} H_i(n)\ ,\ \forall n \notin \Omega$$

$$m_i \leftarrow m_i + 1$$

end for

end for

Figure 2.6: Unallocated subcarrier allocation

significant when the number of users increases. Therefore, when allocating the unallocated carriers, I take users who were low ranked and assign carriers one by one. Hence, in this scheme I can avoid further calculations in the allocation loop, such as measuring the water filling value of each users subcarriers in each loop, that need to be performed iteratively until all subcarriers are assigned. Also the number of iterations for the proposed scheme is low and upper bounded by the number of subcarriers, as explained in the next section. A complete flow chart of the algorithm is given in Fig. 2.7.

2.3.2 Rank Re-Calculation under User Movement

The attributes are defined from the perspective of the system as whole, and they are aimed to achieve the final objective of the system. In our proposed system it is the reduction of transmission power. Therefore, I defined the attributes to reflect this objective. The weights adjust the importance of the attributes, and therefore the values of weights are only related to the attributes defined. The movements of the users are not directly related to the attributes (i.e. I did not define attributes which depends on user mobility).

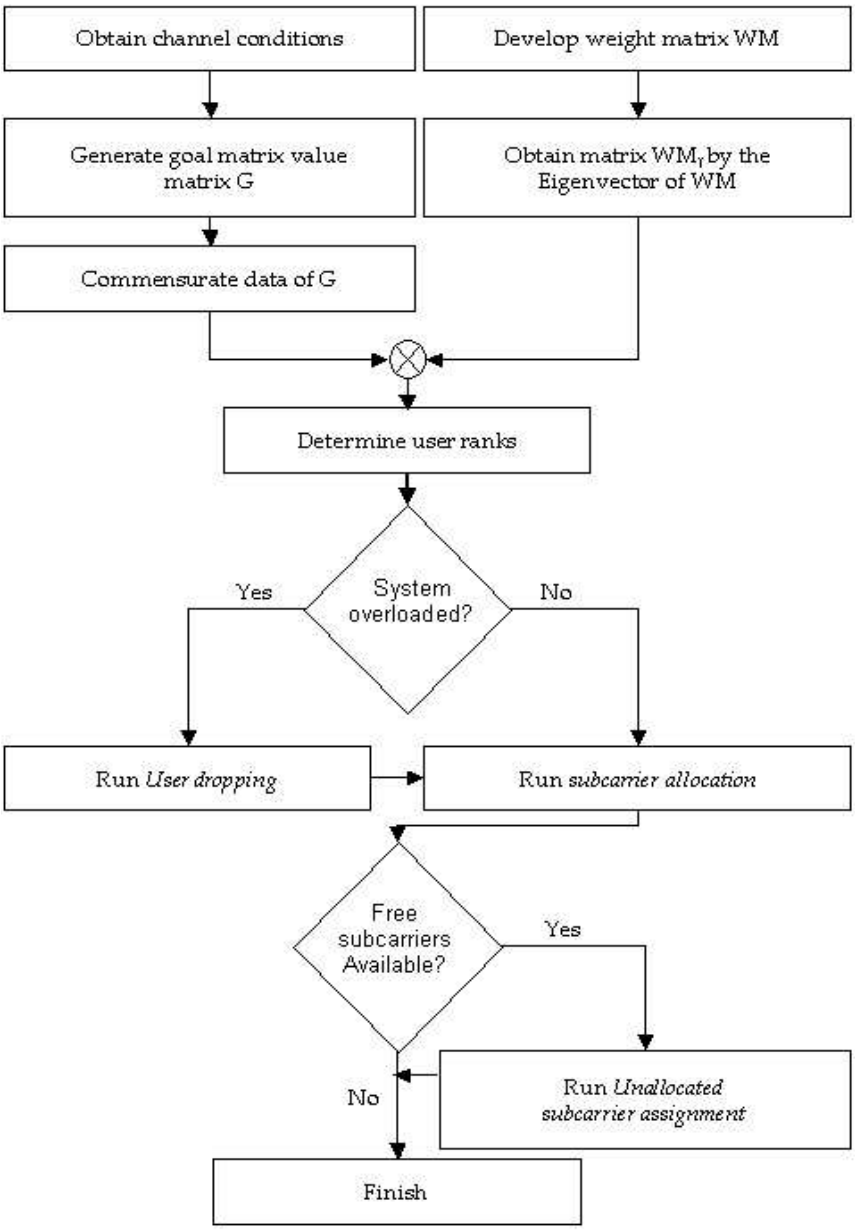

Figure 2.7: Flow-chart of the algorithm

Therefore, for the defined attributes of the proposed system, the mobility of the users does not affect the weights and they can be kept constant. In a practical system, some users maybe stationary, while other users are moving. In this case, the values of the rows corresponding to the moving users of the goal membership matrix changes, because the frequency responses change. In this case, the ranks of these particular moving users also change, effectively changing the ranks of all the users in the system. But in this case, the ranks of only the moving users need to be calculated again. This is because the rank of a user is a relative value only. Not an absolute value between 1 and M (if M is the number of users in the system). The absolute ranks are obtained by sorting the relative ranks. And in the next iteration of resource allocation, the algorithm assign resources based on the new absolute ranks.

2.4 Simulation Results and Discussion

This section shows the simulated results of the proposed scheme. The proposed scheme is simulated and compared with the Bandwidth-assignment-based-on-SNR + Rate-Craving-Greedy (BABS+RCG) and Bandwidth-assignment-based-on-SNR + Amplitude-Craving-Algorithm (BABS+ACG) algorithms proposed in [27]. BABS algorithm determines the number of subcarriers needed by a user to manage the subcarrier requirement of the whole system. If the system is overloaded, BABS algorithm drops users with lower subcarrier needs until the subcarrier requirement of the system is reduced to the supportable limit. On the other hand, if the system has extra subcarriers, BABS algorithm distributes these extra subcarriers to users by using a rate-power function. RCG/ACG algorithms assign the number of subcarriers to the users determined by the BABS algorithm. If the system has extra subcarriers, some users may have more subcarriers than they need. In this case

they can distribute their bits and reduce the transmission power. I choose BABS+RCG and BABS+ACG algorithms for our simulations because they show, although lower, but very competitive results with LR algorithm [12] and RC algorithms and also it has significantly low complexity compared to LR and RC algorithms. LR and RC algorithms can achieve near-optimal results. BABS+RCG algorithm is a sub-optimal rate-craving algorithm, while BABS+ACG is a simpler version of BABS+RCG. Rate-Craving (RC) algorithms involve finding the shortest path in an M-node graph by re-calculating the graph weights in each iteration [27]. The complexity of such algorithms is high and therefore BABS+RCG algorithm is introduced to simplify the search for a single-node. Table 2.1 shows the simulations parameters for the proposed scheme. $A_{i,j}$ is the relative weights of the attribute i over attribute j. Parameters given in Table 2.1 are the values used in the weight matrix described in section 2.2. Users are divided into three traffic types; Type A, B and C. Type A traffic is defined for users requiring a constant and high bandwidth of 1.5Mbps, while type B traffic users are assigned a constant bandwidth of 350 kbps. Type C users are assigned data rates uniformly distributed with a mean of 50 kbps. Type A is assigned 30% of the users while 30% is allocated to type B and the rest of the users are allocated to type C. I used these types of data traffics to keep in accordance with the reference scheme [27]. The data rate requirement of each user for each symbol period is determined by dividing the desired data rate by the symbol rate.

Resource allocation policy is as follows: Each user gets the minimum amount of subcarriers needed to transmit its' data. The minimum number of subcarriers is a function of the maximum modulation level. After allocating all the required subcarriers to the users in the system, if additional unallocated subcarriers are available, then those are distributed among the users (taking users by reverse rank). After the subcarriers are allocated to each user, then s suitable allocation method can be used to distribute the required bits

Table 2.1: Parameters used in the simulation

Parameter	**Values**
A_{12}	1/3
A_{13}	1/3
A_{23}	5
Propagation model	COST 231
Channel	Rayleigh Fading
Number of Subcarriers	512
Symbol rate	9.6 KSymbols/s
Highest modulation level	64-QAM
Number of users	10 100

on to the assigned subcarriers of the respective users. In the simulations, each user is assigned a fixed amount of data to be transmitted during each symbol. All of these data bits are then loaded on to the subcarriers allocated by the proposed resource allocation algorithm. I used the water-filling method to load the bits on to the subcarriers since it gives the lowest transmission power required. Same water-filling method is used to load bits in the simulated conventional schemes as well to maintain fairness in the results. Any other efficient bit-loading scheme can be used too. Since the purpose of the proposed scheme is to demonstrate the effect of ranking method and the resulting subcarrier allocation using the ranks of users, any bit loading algorithm can be used for comparative purposes.

Transmit power is calculated by the following equations [12]:

$$\begin{aligned} P(n,c,BER) &= \frac{f(c,BER)}{H(n)^2} \\ f(c,BER) &= \frac{N_0}{3}\left[Q^{-1}\left(\frac{P_e}{4}\right)\right]^2 (2^c-1)\,, \end{aligned} \tag{2.8}$$

where $Q(.)$ is the Gaussian tail function defined as:

$$Q(x) = \frac{1}{\sqrt{2\pi}}\int_x^\infty e^{(-\frac{t^2}{2})}dt. \tag{2.9}$$

Thermal noise, N_0, is taken as 10^{-5} and a target bit error rate of 10^5 is considered. Adaptive modulation of bits up to 6 bits (64-QAM) per subcarrier is considered. Since I implement

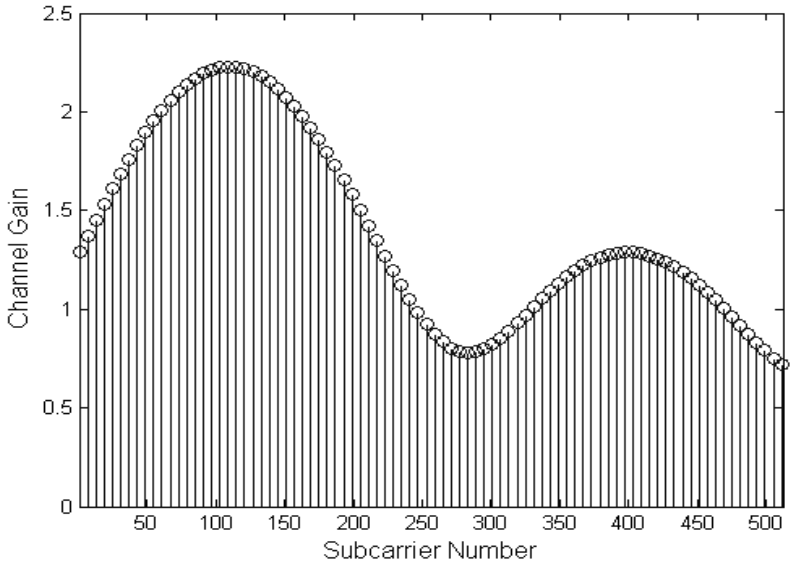

Figure 2.8: A slowly-fluctuating channel

bit-by-bit water-filling on subcarriers to minimize power, I assume that a subcarrier can be modulated with $\{1, 2, \ldots, 6\}$ bits adaptively by BPSK, QPSK, 8PSK, 16QAM, 32QAM and 64QAM. Our main objective is to reduce transmission power based on frequency response of the users and error rate is maintained at a constant level, therefore coding is not implemented in the scheme. When allocating bits to subcarriers, I use water-filling to allocate subcarriers bit-by-bit, by calculating which one of the assigned subcarriers needs the least power to carry one more bit and allocating that bit to the chosen subcarrier. This process is done until all the required bits are loaded onto the subcarriers. The required threshold SINR is calculated by (2.10) below, where c is the number of bits transmitting on the particular subcarrier.

$$\gamma = \left[Q^{-1}\left(\frac{P_e}{4}\right)\right]^2 \frac{(2^c - 1)}{3} \tag{2.10}$$

Figs. 2.8 and 2.9 show two channel profiles as examples of small-scale frequency-selective fading in our simulations. Fig. 2.8 shows a slowly-fluctuating channel, i.e. a

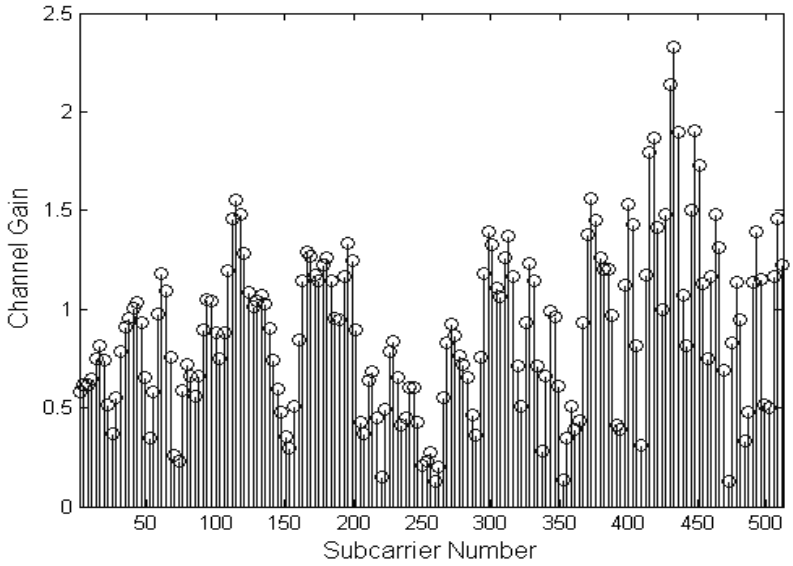

Figure 2.9: A rapidly-fluctuating channel

channel with a frequency response along the considered bandwidth changing slowly, while a rapidly-fluctuating frequency response is shown in Fig. 2.9. Slowly-fluctuating channel is created by using a maximum multipath delay of 5% of the input symbol duration, and for the rapidly-fluctuating channels a maximum multipath delay of 80% of the symbol duration is considered. And for the average channel frequency response, I create channels with frequency response with a maximum multipath delay of 25% of the symbol duration, and the number of multipaths and path gains are chosen according to each channel type to avoid channel pattern repetition.

Simulated results in Fig. 2.11 is obtained for a system with all users having similar channel frequency response shown in Fig. 2.8. For other simulations the users are assigned with average frequency responses. Propagation loss with a cell radius of 500m in COST 231 suburban model with isotropic antennas are considered in the simulation. All the parameters of the COST 231 model are taken in accordance with the IEEE802.16e standard [28].

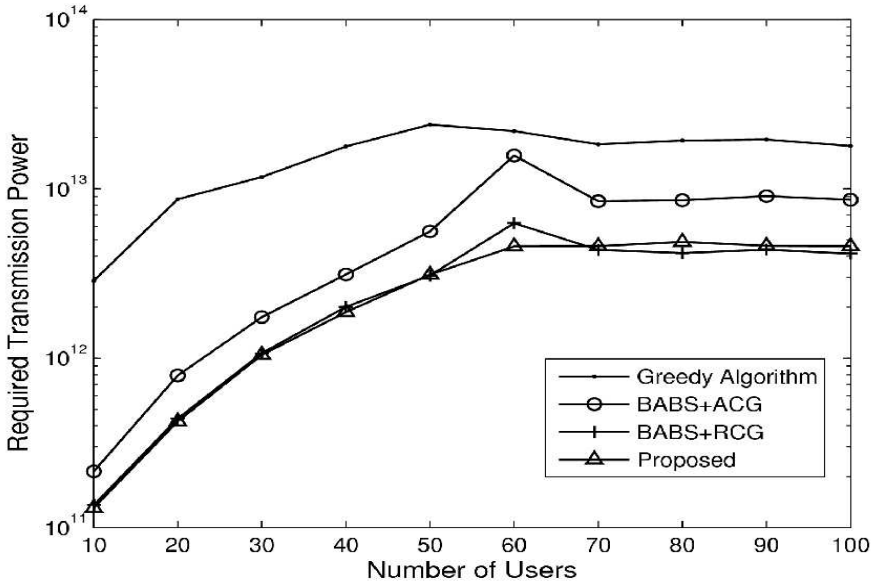

Figure 2.10: Required transmission power versus number of users

Users are distributed uniformly in the cell area and frequency-selective Rayleigh fading is considered. A system with 512 subcarriers and users who might have both slowly-varying and rapidly-varying channels are used. It is assumed in the simulation that all the subcarriers are used for data transfer.

Fig. 2.10 shows the transmission power needed by 4 schemes, the conventional greedy algorithm, BABS+RCG, BABS+ACG and the proposed scheme. The normalized channel gains of users and random carrier selection of the ACG algorithm outperform the Greedy algorithm by giving better chances to users with low channel gains and also avoid the channel correlation of users. Giving priority to users by only considering average channel values as well as the lack of ability to distribute extra subcarriers make the greedy algorithm give the worst performance with significantly higher required transmission power. BABS+ACG algorithm performs better than the greedy algorithm, but the proposed scheme and the BABS+RCG scheme give the best results. The proposed and BABS+RCG schemes

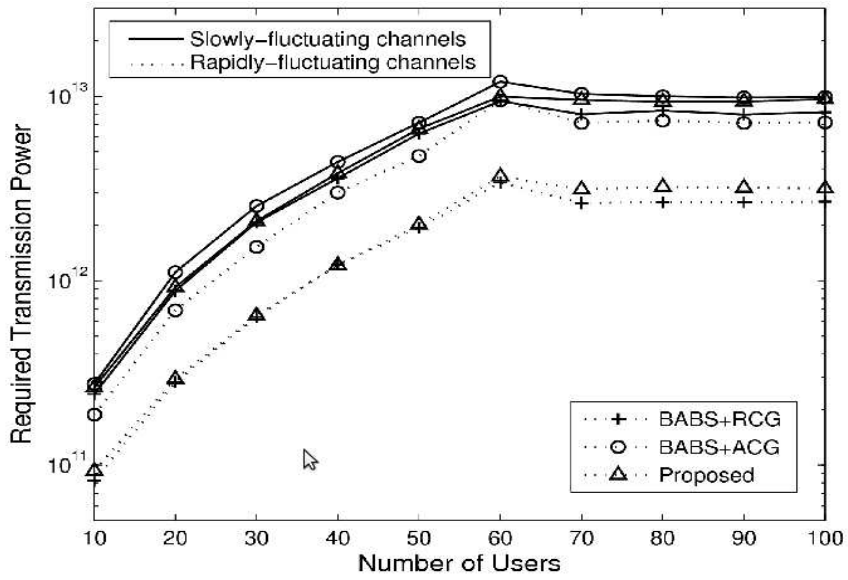

Figure 2.11: Required transmission power versus number of users for different channel profiles

gives identical performance in terms of average transmission power, while the BABS+ACG requires about 105% more transmission power on average than the former two schemes. It should be noted that these results can vary depending on the channel characteristics of the users and data rates used in the simulations.

Fig. 2.11 depicts the required power for the 3 main schemes, BABS+RCG, BABS+ ACG and the proposed scheme for the extreme channel cases, i.e., simulated with all users in the system having similar channel characteristics, either slowly-fluctuating channels (Fig. 2.8) or rapidly-fluctuating channels(Fig. 2.9). This is in contrast to simulation depicted in Fig. 2.10, where the simulated channels are of average fluctuations. As can be seen from the straight lines, which shows the performance for slowly-fluctuating channels, the required transmission power is higher than that for the rapidly-fluctuating channels, shown by dotted lines. Comparing with Fig. 2.10 it can be seen that the required transmission powers for rapidly-fluctuating channels are lower. The reason for this can be explained

as follows: When all the users in the system have slowly-fluctuating channels, there is a high probability that many users will have high correlation with each others channel amplitudes. Therefore, many users will have high and low gain channels in the same range. But since only one user is assigned to a single channel, few users will share the better carriers while other users will be allocated less gain carriers. Therefore higher power requirements should be expected when all the users have slowly-varying channels. As for the RCG algorithm, the neighbor search and swap are not effective enough to find better allocations, therefore some users get more favored over others. The reason for the transmission power to not increase after about 60 users is because the number of subcarriers is kept constant. In this case, there is an upper bound on the maximum transferable data throughput, and therefore some users are dropped when the total required throughput is higher than the maximum system throughput. For this reason, although number of users is increasing, the transmission power does not keep increasing with the number of users, because there is a maximum amount of data the system can transmit. This value of saturation depends on the number of users, number of subcarriers and data rate requirements of the users in the system.

The power requirements for different subcarrier numbers (64,128, 256, 512 and 1024) are simulated in Fig. 2.12. Here the number of users is chosen for each subcarrier number to fully load the system for each scheme. As can be seen from the figure, greedy algorithm gives the worst performance followed by the BABS+ACG algorithm. In this case BABS+RCG algorithm gives slightly better results than the proposed scheme. It should be noted that the performances can vary depending on the number of users and the user data traffic.

The transmission powers shown in the previous figures are higher than supported in a practical system. This is because in the simulations there was no limit imposed on the maximum transmission power. The objective of the simulations were to determine if

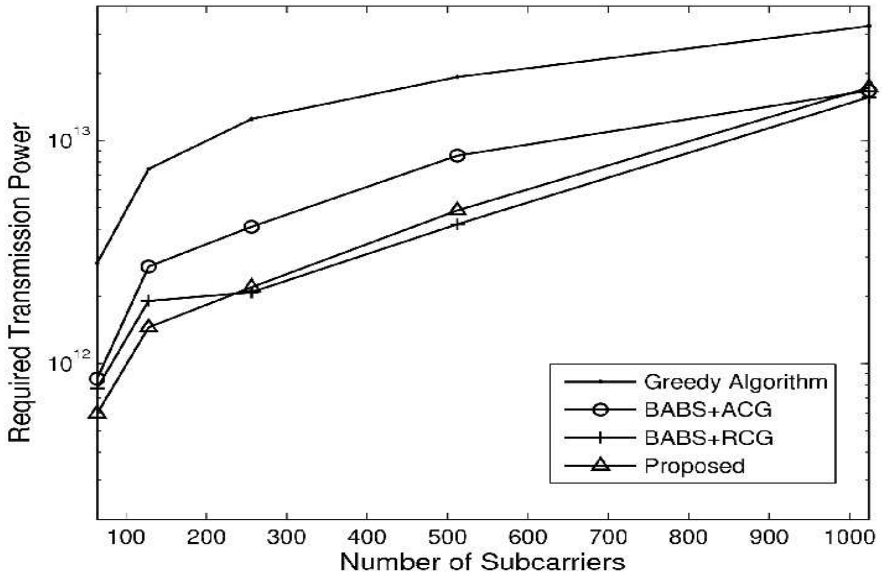

Figure 2.12: Required transmission power verses number of subcarriers

the use of ranking system would reduce the required transmission compared to the other conventional schemes simulated. Therefore, we calculated the 'total' required transmission power for a given set of users and their data rates. In the simulations, users were assigned three data rates. And each of the users, who are not dropped because of insufficient system capacity (in terms of number of subcarriers) get all of their data transmitted. Maximum number of bits on a subcarrier is limited to 6, and the system transmits up to this amount of bits on a subcarrier to fulfill user requirement. Therefore the transmission power shows a higher value. In a practical system there will be a maximum transmission power, system-basis or per-subcarrier-basis, and all the data of users will not be transmitted. Since the objective of the proposed system is to reduce the overall system transmission power, the value of transmission power in the simulations serves only as a relative value for the comparison purposes with the conventional schemes. Next, the individual performances of the three attributes are defined.

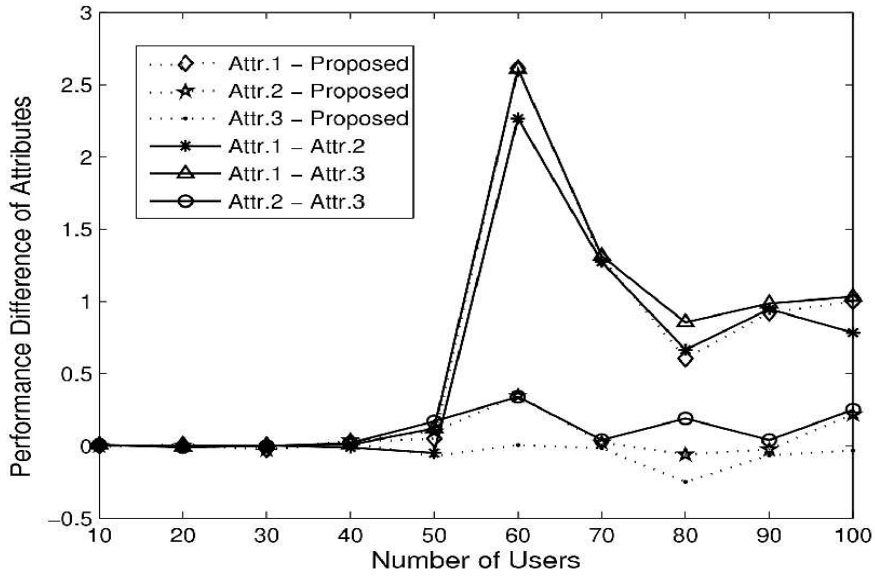

Figure 2.13: Performance difference of attributes verses number of users

Figs. 2.13 and 2.14 show the performance differences by using individual attributes. I take each attribute separately and determine the attribute values for each user. Since multiple attributes are not used in these figures, no weighing is performed. To clearly show the variation of the performances of each attribute, I take the difference of the results. For example, curve Attr.1-Attr.2 shows the required transmission power by attribute 1 subtracted by attribute 2.

Fig. 2.13 shows the power requirement differences for different user numbers in a 512 subcarrier system. The schemes perform similarly for smaller number of users and fluctuate when the numbers of users are increasing. Performance of attribute 1 decreases with the increasing number of users while attribute 2 and 3 shows similar performances. Attribute 1 is able to give competitive performance with small user numbers, but when the number of users is increasing, average channel gain of the user does not give good channel information, while attribute 2 and 3 are able to exploit the channels more efficiently and

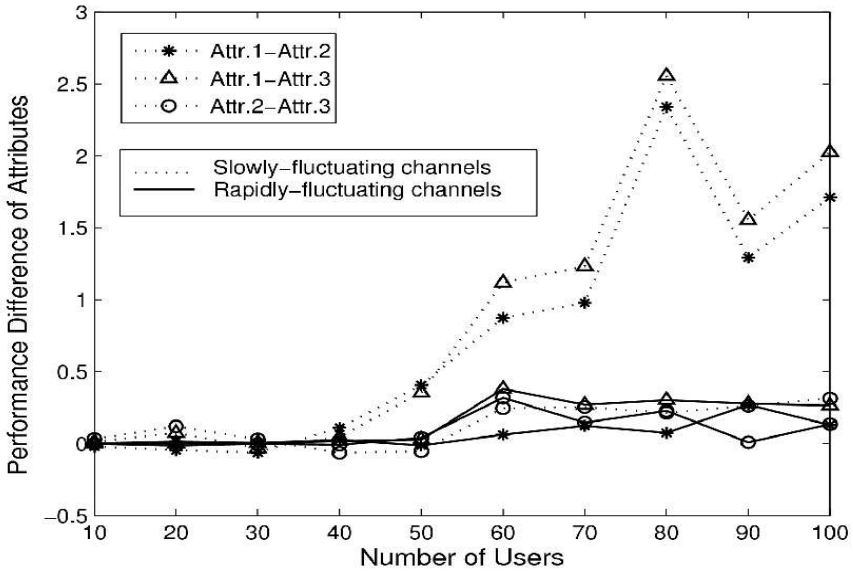

Figure 2.14: Performance difference of attributes verses number of users for different channel profiles

give higher performance. As can be seen from the dotted curves, by multiple attribute weighing, the proposed scheme is able to obtain an overall better result.

Fig. 2.14 shows the difference of attribute performances for the slowly-fluctuating and rapidly-fluctuating channel profiles. Dotted lines depict the slowly-fluctuating channels while straight lines show the results for rapidly-fluctuating channels. Important observations are that the performance difference is higher for the slowly-fluctuating channel, and that for lower number of users the attributes perform similarly. As number of users increases, the performance of attribute 1 diminishes, while the performances of attribute 2 and 3 remain similar throughout all the number of users. Also it can be seen that for the rapidly-fluctuating channel, all attributes perform in a similar manner. It is found from Figs. 2.13 and 2.14 that different attributes perform differently for varying system parameters and the appropriate weighing gives an efficient way to obtain an overall better result. When the users in the system have slowly-varying channels, average channel gain of the user does not give

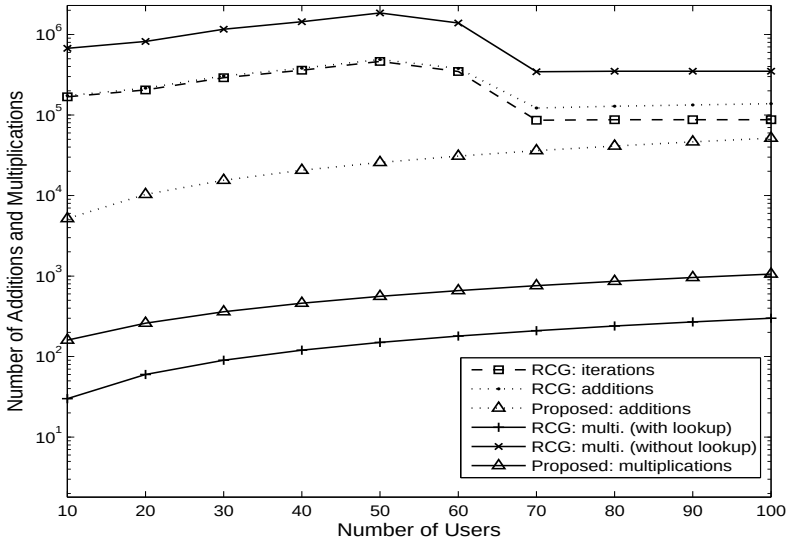

Figure 2.15: Number of additions and multiplications verses number of users

a good indication of the users channel, since channel amplitudes of users are correlated. Therefore, attribute 1 is not able to perform efficiently with increasing number of users, however, when users have rapidly-varying channels, the uncorrelated channel amplitudes of the users can be efficiently exploited and all the attributes give similar results.

Therefore, by multiple attribute weighing, the proposed scheme is able to obtain an overall good result. We see that using all attributes collectively with weight tuning gives an overall better result. The attributes and weights can be adjusted for the system to perform at a satisfactory performance level at all times, or they could be adjusted dynamically depending on the system state.

Fig. 2.15 shows the number of additions and multiplications needed by the RCG and the proposed algorithms and also the number of iterations for the RCG algorithm. Number of iterations shown in the figures denotes the number of iterations which the RCG algorithm has to loop itself until the convergence. RCG algorithm in the first step takes

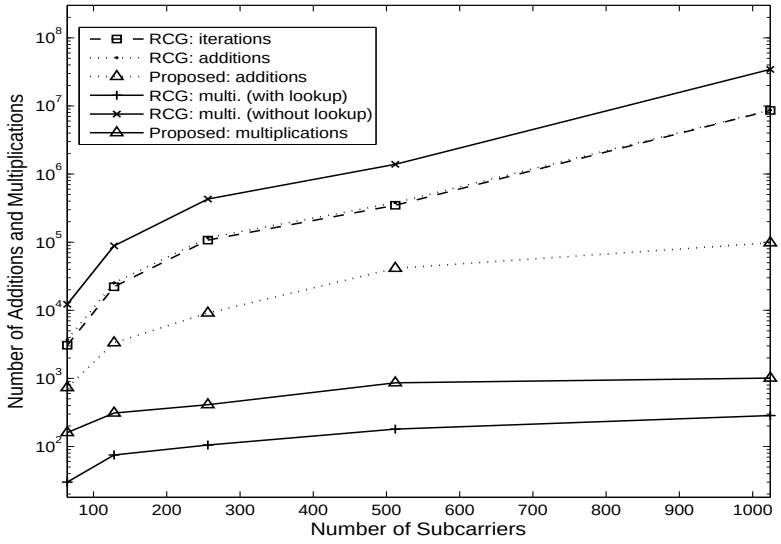

Figure 2.16: Number of addition and multiplications versus the number of subcarriers

the status of the subcarriers one by one and assigns each subcarrier to a user. In the first step, users with better channel conditions might be assigned more subcarriers than they are supposed to have, while some users will not be assigned the minimum number of subcarriers they need. Thus, in the second step, RCG algorithm takes extra subcarrier from users who have been allocated more subcarriers than needed in the first step, and re-assigns these extra subcarriers to users who have not been allocated sufficient number of subcarriers. RCG algorithm converges when all users are allocated their determined number of subcarriers. The amount of calculations in each iteration is a constant. On the other hand, the number of iterations for the convergence is not constant, since it depends on the channel response of the users in the system. Therefore, to show the complexity in terms of number of additions and multiplications required, I show the average number of additions and multiplications needed for the algorithm convergence verses number of users and number of subcarriers, respectively. Proposed algorithm only needs to iterate for the allocation of

unallocated subcarriers and it is comparatively negligible to the RCG algorithm, thus is not shown in the figure. The numbers of additions are large for the RCG algorithm, especially when the number of users is smaller. In the case of small number of users, lot of subcarriers are distributed among a fewer number of users and these subcarriers must be swapped back to other users. Two cases for the number of multiplications needed by the RCG algorithm are considered; with lookup and without lookup. As mentioned previously, RCG algorithm allocates and swaps subcarriers by a rate-power function which uses each users water-filling coefficient for the particular subcarrier. Thus, water-filling coefficients must be calculated for each user for each subcarrier. Therefore, to simplify the algorithm, each users water-filling coefficients are stored in a table and during each iteration the algorithm looks up the coefficient in the table instead of calculating it. This case is shown in the with lookup curve. The without lookup curve shows the number of multiplications it needs when the water-filling coefficients are not stored but calculated during each iteration. As can be seen from the without lookup curve, the RCG algorithm requires a large number of multiplications for the convergence. From the figure, it is observed that the number of multiplications needed by the RCG algorithm is smaller than that of the proposed scheme when the algorithm is run with table lookup. Few points are worth mentioning about this case: when implementing table lookup, it is necessary to calculate and store (MxN) values in a table, and then, during each iteration, the algorithm has to lookup the table and obtain the coefficients for two users and perform an addition. As the graph shows, RCG algorithm needs a large number of iterations, and the efficiency of the algorithm decreases when it is necessary to lookup and retrieve values from a table. The proposed algorithm needs to iterate only if there are unallocated subcarriers, and the allocation only needs to know the rank of the user and no further calculations are necessary.

Fig. 2.16 shows the similar case as Fig. 2.15 but with the increasing number of

subcarriers. I simulate for 64, 128, 256, 512 and 1024 subcarriers. Similar to Fig. 2.15, the number of calculations is larger for the RCG algorithm and it increases with the increasing number of users. Therefore, the efficiency of proposed algorithm over RCG algorithm is evident from the amounts of information to store, calculations and iterations. It should be mentioned that, the performance results of these algorithms depend on number of factors such as channel profiles of users, number of users in the system and required data rates of users.

Considering the number of calculations from a practical point of view, resource allocations need to be fast, and therefore if a hardware implementation is used, the amount of calculations can become a calculation burden. If, fixed-point arithmetic hardware is used opposed to a floating-point hardware (for cost reasons), then it can be more complex since register overflows, etc. need to be taken care of, and this adds additional delay to the calculations. On the other hand, if the with-lookup method is used, the water-filling coefficients are stored in memory. These memory values are then read in each iteration. As an example, Fig. 2.15 shows that the RCG algorithm requires about 10^5 iterations in each step. In a DSP for example, each memory read could take one cycle. The RCG algorithm requires four water-filling coefficients in each iteration, which could take up to four cycles of delay (assuming only one memory bank). Therefore, from a calculation and delay point of view, the conventional scheme seems to be high in complexity.

2.5 Conclusion

The proposed a user ranking technique which completely avoids complex computations and iterations. The scheme allows efficient resource allocation with reduced transmission power and increased bandwidth utilization. The employed multi-attribute technique

allows for defining suitable attributes for the required performance metric. The ability to weight these attributes gives extended flexibility in handling the importance of the various attributes. Simulation results show good performance compared to the two low complexity resource allocation algorithms used. Proposed scheme constantly gave better results than the BABS+ACG algorithm and gave similar results to that of BABS+RCG scheme. Although giving similar results to that of the BABS+RCG scheme, from the perspective of calculation complexity, the proposed scheme has a higher complexity. The proposed scheme avoids iterations and the number of calculations are bounded by the number of subcarriers and the users, unlike the RCG scheme which needs iterations for the convergence, and also the convergence rate is dependent on the user channel conditions. Therefore, the proposed scheme is able to achieve a resource allocation scheme with reduced complexity, and also provides the flexibility for system objective and attribute weighing.

Chapter 3

Steady-state Kalman filtering for OFDM systems in Rayleigh Fading Channels

3.1 Introduction

In section 1.4.4 I discussed the importance of channel estimation for OFDM systems, in particular for resource allocation algorithms and section 1.4.5 show an example of how incorrect channel estimations degrade the receiver performance. Section 1.3.1 discussed that resource allocation is a complex task and in chapter 2 I proposed a low complexity resource allocation scheme. Channel estimation too is a complex process as explained in sections 1.4.2 and 1.4.6 and the estimation accuracy affects the system performance [29]. In this chapter, I introduce a steady-state Kalman filter, which has reduced complexity compared to the conventional Kalman filter, and also gives better performance since the proposed filter avoids the convergence period.

Least Squares Estimation (LSE), Maximum Likelihood Estimation (MLE) and Minimum Mean Squared Estimation (MMSE) are the main techniques employed in channel estimation. Kalman filters can be categorized as an extension of the MMSE. Although MMSE gives superior performance over others, it is calculationally complex due to the matrix inversion in the calculation, as is the case with vector Kalman filters.

Kalman filters can be seen as a subset of MMSE filters. In fact, a Kalman filter behaves as a sequential MMSE filter. However, Kalman filter has the important feature of being able to handle non-stationary channels, in contrast to the conventional MMSE filters [30]. Apart from the ability to handle non-stationary channels, which makes Kalman filters very useful in mobile environments, it needs minimal channel statistics. This is an obvious advantage in practical situations where the advanced channel statistics are not easily available. Although Kalman filter is a better alternative to the MMSE, it still has a considerable amount of calculation complexity involved in calculating the Kalman gain.

In [31], the author has used the conventional Kalman filtering with an initial LSE to reduce estimator variance, and recursive calculations of noise variances are performed. The complexity of the Kalman filtering is thus increased over the conventional method by using recursive noise variance and Kalman gain calculations. In [32], the authors propose a modified Kalman filter for channel estimation and tracking of OFDM systems. A simplified estimation for driving noise covariance matrix is derived on the assumption that the channel has equal power, and a recursive estimator for the auto-regressive model fading parameter is obtained by partial differentiation that would minimize the MSE. The calculation still requires the matrix inversion as in the conventional Kalman filter and the computational complexity is further increased by the recursive calculations of parameters which need to be determined in each sample. Although Kalman filters are researched in the context of OFDM channel estimation, a steady-state Kalman filter approach could not be found in the

literature to the best of our knowledge. By exploiting the pilot subcarrier characteristics of OFDM, the calculation complexity of the conventional filtering approach can be greatly reduced as we can transform the vector Kalman filtering to scalar domain without losing general functionality. And this lets us easily derive the steady-state Kalman gain which is quite difficult for the case of conventional Kalman filtering. Furthermore, even though the channel statistic requirement is minimal in Kalman filters, it still needs the knowledge of AWGN noise variance and the channel driving noise variance. I will further extend our scheme to perform in the absence of knowledge of these two parameters by utilizing SNR, which is a value measurable easily by most receivers.

3.2 Conventional Kalman Filtering and Problem Formulation

I consider a OFDM system with N subcarriers and assume that all the subcarriers are used for data transmission, excluding the DC, null or guard subcarriers. A bit stream is serial-to-parallel multiplexed and modulated to $X_n = [x_{n,0}, x_{n,1}, \cdots, x_{n,N-1}]$ complex points. Subscripts $x_{n,k}$ denote the k^{th} data point of the n^{th} symbol. These N complex modulation points are then input to a inverse discrete-Fourier-transform (IDFT) block. The output of the IDFT block is then parallel-to-serial converted and the CP is added to mitigate the effects of ISI. Then it is digital to analog converted before transmitting over the channel. The time-domain signal received during the n^{th} symbol period, after the removal of guard interval, can be expressed as:

$$Y_n(t) = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} x_{n,k} h_{n,k} e^{\frac{j2\pi nk}{N}} + w_n(t). \tag{3.1}$$

Here $0 < t \leq (N-1)$ and $h_{n,k}$ denotes the Rayleigh fading complex channel gain of subcarrier k during symbol period n and is considered constant during the symbol period.

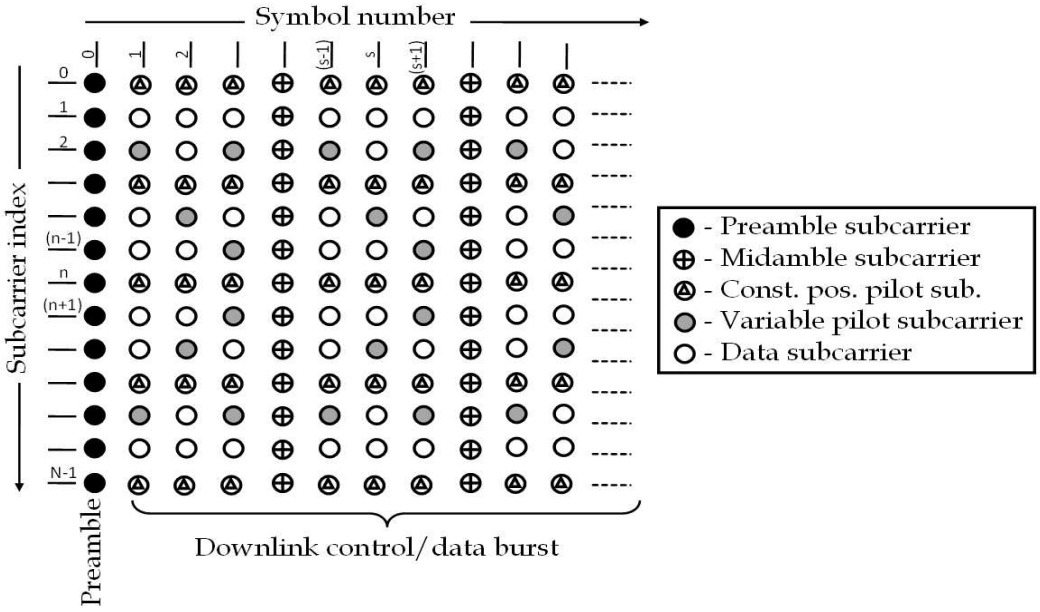

Figure 3.1: Time-frequency resource grid of a partial downlink frame

I assume a constant channel for the symbol period, and thus can disregard the Inter Carrier Inference (ICI) appearing after the DFT operation. With a properly defined CP we can assume the transmitted signal, after removal of the guard interval, is cyclically convoluted and thus the received signal after the DFT for the symbol period n can be expressed in matrix form as:

$$\tilde{Y}_n = \tilde{X}_n H_n + \tilde{w}_n. \tag{3.2}$$

Here $\tilde{X}_n$ is a NxN diagonal matrix with X_n on its diagonal. $\tilde{w}_n$ is AWGN noise vector with probability density function (PDF) $w_n \sim \mathbf{N}(0, C_n)$, where C_n is the covariance matrix. H_n is the channel transfer function defined as, $H_n = [h_{n,0}, h_{n,1}, \cdots, h_{n,N-1}]^T$. T denotes the matrix transpose. Kalman filtering is a non-blind estimation and thus the filtering is performed on signals with known information. I assume a system with N subcarriers and of which P subcarriers are used as pilot subcarriers in the downlink. Interpolation/extrapolation can then be used to measure the channel at other subcarriers from

the estimated pilot subcarriers. The set of pilot subcarriers which can be filtered with the Kalman filter is graphically depicted in Fig. 3.1. It shows part of a downlink frame of a typical OFDM system, where the colored circles show pilot subcarriers. Kalman filtering in general needs to perform the filtering on pilot samples received at regular intervals. Filtering is not straightforward when the received pilot positions are variable. Therefore, to arrive at a steady-state filter, I assume that the Kalman filtering is performed on a constant position pilot set of P subcarriers, thus enabling our scheme to be employed to filter preambles or constant pilots in the full usage of subcarriers (FUSC) permutation in the IEEE802.16e standard, since these pilots are sent at regular intervals. Furthermore, the proposed steady-state filter can easily be used with the preambles, midambles or pilot subcarriers of the IEEE802.16e standard which has fixed position pilots throughout the downlink frame [17]. Now (3.2) will be modified as $\tilde{Y}_n^P = \tilde{X}_n^P H_n^P + \tilde{w}_n^P$ to distinguish the received signals from the pilot subcarriers. Here $\tilde{X}_n^P = \left[x_{n,p_0}, x_{n,p_1}, \cdots, x_{n,p_{P-1}}\right]$ which denotes the complex data values of the pilot subcarriers at the corresponding positions and $H_n^P = \left[h_{n,p_0}, h_{n,p_1}, \cdots, h_{n,p_{P-1}}\right]$ is the corresponding channel transfer function of the pilot subcarriers, with $\tilde{X}_n^P \subseteq \tilde{X}_n$ and $H_n^P \subseteq \tilde{H}_n$. For example, if filtering on the preambles or midambles $\tilde{X}_n^P = \tilde{X}_n$ and $H_n^P = \tilde{H}_n$, otherwise $\tilde{X}_n^P$ and H_n^P will contain entries corresponding to a set of constant pilot subcarriers. The channel will be defined as a dynamic model using an order-1 AR process.

$$H_n^P = AH_{n-1}^P + d_n. \tag{3.3}$$

A is the $P \times P$ *state-transition* matrix which defines how the $(n-1)^{th}$ channel state is correlated with the n^{th} state of the channel. d_n is the Px1 *driving noise* vector with a PDF $d_n \sim \mathbf{N}(0, Q_n)$, where Q_n is the corresponding covariance matrix [30]. For clarity I will change (3.2) as $\tilde{Y}_n^P = \tilde{X}_n^P H_n^P + \tilde{w}_n$ to distinguish the received signals from the pilot

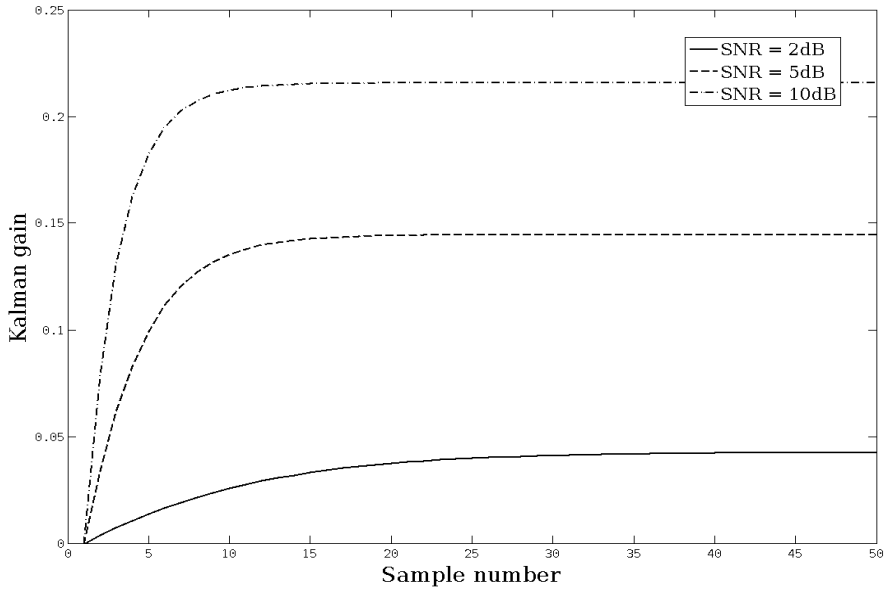

Figure 3.2: Convergence rates of Kalman gain for different SNR values

subcarriers.

Kalman filtering has five steps associated with it as explained earlier in section 1.4.3: 1) *prediction*, 2) *minimum prediction MSE calculation*, 3) *Kalman gain calculation*, 4) *correction* and 5) *minimum MSE calculation*. In the conventional Kalman filtering, these five steps must be performed for each data sample received, of which, three steps are for the calculation of the Kalman gain. Vector Kalman filtering involves a matrix inversion in the Kalman gain calculation as shown in (3.4) [30]:

$$K_n = \frac{M_{n|n-1}\tilde{X}_n^{P^H}}{C_n + \tilde{X}_n^P M_{n|n-1}\tilde{X}_n^{P^H}}. \tag{3.4}$$

Here H denotes the Hermitian transpose. K_n is a $P \times P$ diagonal matrix with its diagonal entries giving the Kalman gain of sample n and $M_{n|n-1}$ is the minimum prediction MSE (i.e. minimum MSE at n given minimum MSE at $(n-1)$). The calculation complexity

of (3.4) grows with the increasing matrix dimension and the need to calculate this matrix for each input data sample becomes a very high calculation burden, specifically with a matrix inversion involved. However, for the case of a stationary channel, the Kalman gain converges to a constant value, i.e. its $steady - state$ value. Fig. 3.3 shows the convergence rate of the Kalman gain for different SNR values. At this steady-state value Kalman filter gives the optimal estimations. Therefore we see that the benefits of the steady-state filtering are two-fold. First is the substantially reduced number of computations required, achieved by eliminating number of steps required from Kalman filtering process mentioned earlier in the section. Second advantage is that since the proposed steady-state filter is operating in the converged state of the filter, it always operates at the optimum condition, while the conventional filter needs to be operated through a convergence period to arrive at the steady-state. This gives an additional performance margin when filtering directly at the steady-state than that compared to the conventional Kalman filtering.

It can be seen that to achieve the convergence and to arrive at the steady-state, the Kalman filter needs to be operated in a stationary channel and also the filter parameters need to be constant for the duration of the filtering. When this requirement is not met such as with the variable pilots whose time and frequency positions are varying, Kalman filtering becomes complicated since it requires dynamic parameter variation from symbol-to-symbol which is not easily realized in practical systems. In this scenario the state transition matrix A and the driving noise covariance matrix Q_n vary for each filtering step. Due to this variation of filter parameters the Kalman filter will not enter a converged state even for a stationary channel. This is true for sequential filters in general. Therefore, to obtain a steady-state Kalman filter, we concentrate on filtering of pilots which are received at constant positions and intervals as shown in the Fig. 3.1, i.e. steady-state filtering can be performed on either of the preamble, midamble or constant pilot subcarrier sets. The

scheme can also be easily utilized to filter all three sets of pilot patters by defining separate steady-state Kalman gains for each pattern with the corresponding model parameters. I show that finding the steady-state Kalman gain is a trivial task. This does not hamper the complexity of symbol-by-symbol filtering because during each symbol, only one particular type of pilot is filtered.

Observing (3.4), we can see that the variability of Kalman gain is dependent on the variation of the minimum prediction MSE only, while other parameters remain constant for a stationary channel. Thus by finding the steady-state minimum prediction MSE we can get the steady-state Kalman gain by substitution. Minimum prediction MSE is in turn a function of minimum MSE, which is a function of Kalman gain. To find the steady-state value I substitute for minimum MSE and again substitute for Kalman gain. Then at steady-state we have, $M_{n+1|n} = M_{n|n-1} = \tilde{M}$. This results in an expression for steady-state minimum prediction MSE $\tilde{M}$:

$$\tilde{M} = A\tilde{M}A^H - A\tilde{M}\tilde{X}_n^{P^H}\left(C_n + \tilde{X}_n^P\tilde{M}\tilde{X}_n^{P^H}\right)^{-1}\tilde{X}_n^P\tilde{M}A^H + Q_n. \tag{3.5}$$

Here (3.5) is in the form of a *Riccati equation* [33] and therefore solving for $\tilde{M}$ becomes a problem of solving a Riccati equation. Solving Riccati equations is a complex process and different techniques have been implemented. For example, in [34] the authors have derived an eigenvector solution while the author of [35] has used a Schur vector approach instead of eigenvectors. Other methods include solving scalar polynomials and calculation by iterations [36]. Apart from the calculation complexity, these methods have the drawback of not solving $\tilde{M}$ for an explicit expression. Thus if we use any of the aforementioned methods to solve for $\tilde{M}$ we would have to stop our process at that because it is not possible to manipulate the solution further, and hence we cannot extend the solution for performing without the knowledge of the driving noise variance. Our objective in this paper is to find

the steady-state Kalman gain which can greatly reduce the calculation burden in filtering. In addition, operation without the driving noise variance knowledge makes the scheme very efficient in practical situations. Therefore, we need to simplify the equation such that the matrix inversion in (3.5) can be eliminated and we can obtain an explicit solution for $\tilde{M}$ in terms of $A, \tilde{X}_n^P, C_n$ and Q_n. In the next section I will consider the pilot subcarrier characteristics of the OFDM system and how it effects the behavior of the Kalman filtering process. Then I will show that the vector Kalman filtering can be reduced to a much simpler version, which will help us solve for the steady-state Kalman gain easily.

3.3 Proposed Steady-State Kalman-Filter

Previous section discussed how the Kalman filter converges to a steady-state in stationary channel conditions after an initial transient period. I showed how this Kalman gain can be directly determined by solving the Riccati equation in (3.5) for the steady-state minimum prediction MSE. Due to the computational complexities in solving (3.5), in this section I propose how this complexity can be avoided in the case of OFDM transmission, specifically to the IEEE802.16e standard by utilizing the characteristics of pilot subcarriers used in OFDM. Then I explain transforming the Kalman filter from the conventional vector domain to the scalar domain and still easily obtain the same functionality. In the scalar domain we can easily derive a closed-form expression for the steady-state minimum prediction MSE. Then obtaining the steady-state Kalman gain matrix becomes a trivial task. I then proceed to derive the steady-state Kalman gain with SNR which will omit the requirement of knowing the driving noise variance of the dynamic channel model. I will show in section3.5 that accurate value of driving noise variance is very important for the proper operation of the Kalman filter and that the proposed method can easily track the

variance and change the parameter dynamically.

In most communication systems pilot subcarriers have the characteristic of having a lower order modulation level. In this chapter I will focus on the IEEE802.16e standard. This standard proposes the use of QPSK as the modulation level for the pilot subcarriers, meaning having only $(1+j1), (1-j1), (-1+j1), (-1-j1)$ in the modulation alphabet for the pilot subcarriers. Consider the following facts:

- 1.The channel model is an AR-1. Then the state-transition matrix becomes a diagonal matrix, because current state of the channel is only dependent on the previous state of the channel. The state-transition value characterizes the rate of change from the previous symbol to the current symbol. Therefore, this value can be seen as the time-correlation of the subcarrier and thus depends on the Doppler frequency. Since all the subcarriers undergo the same Doppler frequency, the time-correlation of all the subcarriers are the same. For this reason, the state-transition value is the same constant for all the subcarriers.Thus, the state-transition matrix A will be a diagonal matrix, giving $A = \alpha I$, where α is a scalar value depicting the transition value and I is the identity matrix.

- The AWGN noise and the driving noise of the channel model are independent Gaussian noises, uncorrelated with each subcarrier. Also these noises are the same for all the subcarriers. Therefore, the covariance matrices of the AWGN noise and the driving noise are diagonal matrices with each matrix having the same noise value, which lets us write the covariance matrices Q_n and C_n as $Q_n = \sigma_u^2 I$ and $C_n = \sigma_n^2 I$. σ_u^2 and σ_n^2 are driving noise and AWGN noise variances, respectively.

- Therefore, we have $A, \tilde{X}_n^P, Q_n$ and C_n all as diagonal matrices, then from Kalman filter definition for the minimum prediction MSE matrix $M_{n|n-1}$, becomes a diagonal matrix. The diagonal entries of $M_{n|n-1}$ will correspond to the different pilot subcarrier modulation alphabet. Later I will see that for QPSK modulation, all the diagonal entries will have the same value at the steady-state condition.

Therefore, the state-transition matrix and the noise covariance matrices will be of the form:

$$\begin{pmatrix} a & 0 & \cdots & 0 \\ 0 & a & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a \end{pmatrix} \tag{3.6}$$

and

$$\begin{pmatrix} \sigma^2 & 0 & \cdots & 0 \\ 0 & \sigma^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma^2 \end{pmatrix} \tag{3.7}$$

When these three (state-transition, AWGN noise and driving noise) matrices are diagonal, the resulting Kalman gain matrix is diagonal too. And we see that each entry in the Kalman gain matrix diagonal is a function of the state-transition constant, AWGN noise variance, driving noise variance and pilot symbol. Since the state-transition value, AWGN and driving noise variances are same for all the subcarriers as discussed earlier; each value of the Kalman gain differs by the pilot symbol value. Therefore, we are able to simplify the conventional problem of finding the Kalman gain matrix by finding the individual Kalman gains for each subcarrier, which only depends on the data (pilot) symbol. Thus the original

vector problem is solved as a scalar problem. With the above facts we can deduce that the Kalman gain is a diagonal matrix with diagonal entries corresponding to the particular modulation alphabet, which for the case of QPSK is only four distinct values. Therefore, we can easily construct the Kalman gain matrix since the receiver has the complete knowledge of the pilot data sequence. Hence can greatly reduce the complexity of Kalman gain calculation by finding the entries in the matrix separately since we only need to determine few values (depending on the pilot subcarrier modulation format), regardless of the number of pilot subcarriers in the system.

AR-1 channel model and the state-transition value

A brief explanation about the AR-1 channel model and its' corresponding Kalman filter state-transition value is given here. All the constants of the state-transition matrix is taken as the same constant a as shown in eq. 3.6. This is due to the following reasoning: The conventional Kalman filter is built on the assumption that the transition model of the signal being measured is a Gauss-Markov model. Based on this model, the channel, i.e. the subcarrier in the proposed OFDM case, takes the form [30]

$$h_n = -\sum_{k=1}^{p} \alpha_k h_{n-k} + u_n \tag{3.8}$$

where subscript n denotes the current time instant and u is the driving noise. Driving noise is a zero-mean Gaussian variable which models the random error of the system. This model is a p^{th}-order Auto-Regressive (AR-p) model. In the proposed method, I assumed the channel to be an order-1 AR model. Therefore, the channel model becomes,

$$h_n = \alpha h_{n-1} + u_n \tag{3.9}$$

The purpose of the state-transition constant a is to define how much the current value will be similar to the previous value. In essence, this is the time-correlation between the

samples. Therefore, for example, in the case of a subcarrier, if the transmitter-receiver link is stationary, the time-correlation between the current and previous samples will be unity. On the other hand, if the transmitter and/or the receiver is moving, the subcarrier value will change from sample to sample. Then, this rate of change will be proportional to the magnitude of mobility. If it is fast, then there will be less correlation between samples and vice versa.

For a wide-sense stationary uncorrelated scattering (WSSUS) channel, this time-correlation between Rayleigh faded samples is given by the zeroth-order Bessel function of the first-kind [9],

$$E\left[h(t)h(t-\tau)\right] = J_0\left(2\pi f_d \tau\right) \tag{3.10}$$

Therefore, the correlation coefficient, or the state-transition constant, is a function of the product $f_d\tau$, the Doppler frequency and the delay between samples, respectively. For the case of OFDM, if we are estimating the channel each symbol, the correlation becomes

$$E\left[h_n h_{n-1}\right] = J_0\left(2\pi f_d T_s\right) \tag{3.11}$$

where T_s is the symbol duration. Considering the time of an OFDM symbol and maximum practical Doppler frequencies, it can be seen that correlation coefficient is a value greater than zero. Therefore, the value of the state-transition constant is a function of the Doppler frequency and the symbol duration, thus, is same for all the subcarriers.

In a multipath channel, the frequency-response is frequency-selective for an OFDM system due to the narrow bandwidth subcarriers. And for a mobile scenario, the channel frequency response, or the subcarrier gains, changes with time; from symbol to symbol. This change of values is what the Kalman filter is trying to track, or estimate. In this case, some subcarriers values will increase in the next symbol, while other subcarriers decrease. This increase and decrease is of course not independent from subcarrier to subcarrier as

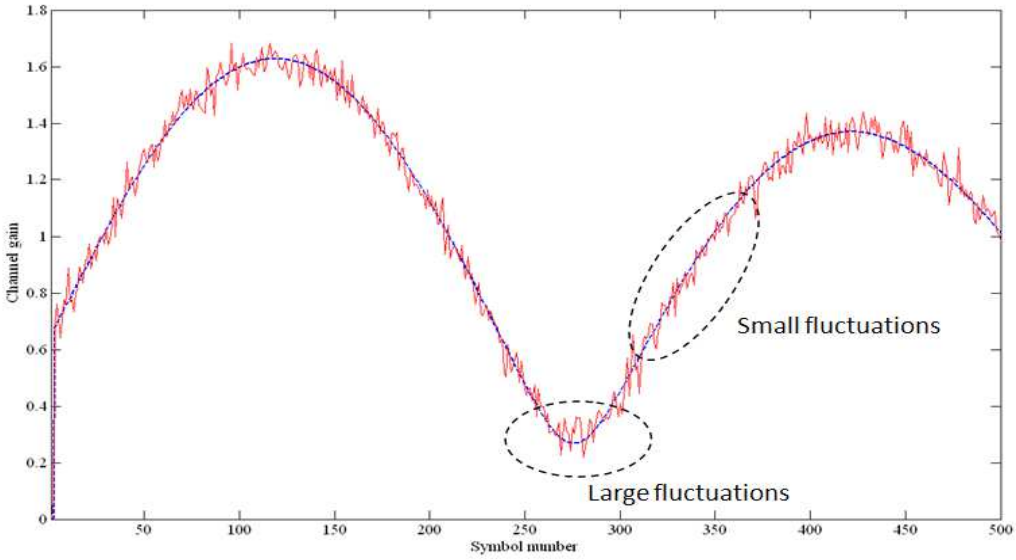

Figure 3.3: Effect of state-transition value on Kalman filtering

there is frequency-correlation between them. For the context of our study I do not consider any frequency-time correlation between subcarriers.

The state-transition constant, as discussed earlier, measures the correlation between samples, and does not indicate a positive or negative transition between samples. Tracking of this positive or negative transition is performed by the Kalman filter. Kalman filter, as most estimation filters such as least-mean squares (LMS) or recursive-least-squares (RLS), tracks the signal by a learning process, aided by the pilot data. Of the five steps involved with the Kalman filter, the first step is the prediction. Kalman filter uses the state-transition value to predict the current value. Then in the fourth step, which is the correction step, Kalman filter utilizes the Kalman gain and pilot data to correct the prediction. Figure below depicts tracking of a subcarrier using a Kalman filter. The blue line shows the true channel and the red dashed line depicts the Kalman filter estimation.

The property to note is the higher fluctuations at the turning points and lesser

fluctuations in the straight regions. When the subcarrier is either increasing or decreasing steadily, the filter is able to track it better, because the value follows the prediction closely. Therefore we see better estimations of the subcarrier. At turning points, there are higher fluctuations, i.e. the estimations are not correct. The pattern followed by the subcarrier (increasing or decreasing) is changed and the previous symbol value does not provide a correct prediction of the current value. Hence the estimations are erroneous and we see relatively large fluctuations. During this period the Kalman filter tries to learn and adjust. Then again when the channel settles to a linear period, the estimations are again accurate and fluctuations are minimal. Therefore, the state-transition constant serves as a prediction parameter for the current value from the previous value. From the prediction the Kalman filter refines its estimation by using the pilot data and the actual received value. Thus, the state-transition value is a constant for all the subcarriers in the OFDM system, and the increasing or decreasing nature of the channel is tracked and estimated by the Kalman filter using received signal and known information such as the pilot data and the noise variances.

3.3.1 Simplification to the scalar form

As I explained in the previous section, we can easily construct the Kalman gain matrix by finding the individual values of its diagonal. This lets us transform our Kalman filter problem from matrix domain to scalar domain and process the filtering considering a single subcarrier case. The scalar domain equations will be defined next which can be derived from transforming the vector Kalman equations. Subcarrier index is redundant and will be dropped from hereon, and filtering of a pilot subcarrier is implied.

Dynamic channel model

$$h_n = \alpha h_{n-1} + u_n. \tag{3.12}$$

Received signal

$$y_n = \beta h_n + q_n. \tag{3.13}$$

Prediction

$$\hat{h}_{n|n-1} = \alpha \hat{h}_{n-1|n-1}. \tag{3.14}$$

Minimum prediction MSE

$$m_{n|n-1} = \alpha^2 m_{n-1|n-1} + \sigma_u^2. \tag{3.15}$$

Kalman gain

$$k_n = \frac{\beta^* m_{n|n-1}}{\sigma_n^2 + |\beta|^2 m_{n|n-1}}. \tag{3.16}$$

Correction

$$\hat{h}_{n|n} = \hat{h}_{n|n-1} + k_n \left(y_n - \beta \hat{h}_{n|n-1} \right). \tag{3.17}$$

Minimum MSE

$$m_{n|n} = m_{n|n-1} \left(1 - \beta k_n \right). \tag{3.18}$$

Here $\tilde{h}_{n|n-1}$ depicts the *estimated* value of h at time n given estimation at time $(n-1)$. u_n is the driving noise and AWGN noise of the channel is given by q_n, both of which are independent variables to h_n. β is the complex pilot subcarrier symbol corresponding to either of the set $\{(1+j1), (1-j1), (-1+j1), (-1-j1)\}$ for QPSK and * depicts the complex conjugate. Rest of the notations are the scalar form of the previously defined parameters. Then we can derive the scalar form of (3.5) to find the steady-state minimum prediction MSE as:

$$\tilde{m} = \frac{\alpha^2 \sigma_n^2 \tilde{m}}{\sigma_n^2 + |\beta|^2 \tilde{m}} + \sigma_u^2. \tag{3.19}$$

Then Eq.(3.12) can be re-arranged to the following quadratic equation:

$$|\beta|^2\tilde{m}^2 + \left(\sigma_n^2\left(1-\alpha^2\right) - |\beta|^2\sigma_u^2\right)\tilde{m} - \sigma_n^2\sigma_u^2 = 0. \tag{3.20}$$

Solving for the roots:

$$\tilde{m} = \frac{-\left(\sigma_n^2\left(1-\alpha^2\right) - |\beta|^2\sigma_u^2\right)}{2|\beta|^2} \pm \frac{\sqrt{\left(\sigma_n^2\left(1-\alpha^2\right) - |\beta|^2\sigma_u^2\right) + 4|\beta|^2\sigma_n^2\sigma_u^2}}{2|\beta|^2}. \tag{3.21}$$

Here we have the steady-state minimum prediction MSE for a particular pilot subcarrier symbol of β. Inspecting the above equation further reveals that for the case of QPSK modulated symbols, steady-state minimum prediction MSE, $\tilde{m}$, is a constant for all the β. (3.21) gives two roots hence we need to find the one corresponding to the steady-state minimum prediction MSE. Although we perform complex channel filtering, examining the coefficients of the equation shows that the discriminant evaluates to a positive value, resulting in real roots. If we write (3.20) as $a'\tilde{m}^2 + b'\tilde{m} + c' = 0$, and assume the solutions of (3.21) are $\tilde{m}_1$ and $\tilde{m}_2$, they can be written as: $\tilde{m}_1 = \left(\frac{c'}{a'}\right)\frac{1}{\tilde{m}_2}$. Examining the coefficients of (3.20) we see that the ratio $\left(\frac{c'}{a'}\right) = \frac{-\sigma_n^2\sigma_u^2}{|\beta|^2}$ always takes a negative value and hence the roots $\tilde{m}_1$ and $\tilde{m}_2$ will always be of opposite signs. Considering MSE is a positive value, the desired solution from (3.21) is the one corresponding to the positive sign. Finally, the steady-state Kalman gain can be obtained by substituting $\tilde{m}$ in to $m_{n|n-1}$ of (3.16). The steady-state Kalman gain found will be equal to the Kalman gain of that of the conventional Kalman filter in the converged state.

3.3.2 Steady-State Kalman Filtering in the Absence of Noise Variance Statistics

Inspecting (3.21) we see that to solve for the steady-state minimum prediction MSE we need the knowledge of driving noise variance σ_u^2 and AWGN noise variance σ_n^2. Accurately determining the driving noise variance is difficult in a practical situation. In this

section I will implement calculating (3.21) without the explicit knowledge of σ_u^2 by showing that the channel driving noise variance can be realized from system parameters such as noise power and SNR. I assume the knowledge of σ_n^2. According to the IEEE802.16e standard, the receiver must have the capability to measure received-signal-strength-indicator (RSSI) and carrier-to-interference-noise-ratio (CINR). When requested by the base station, it is mandatory for the subscriber station to measure RSSI/CINR and report them via RES-RSP messages [37]. Also, CINR conveys more information than the RSSI, including interference and noise levels and the receiver can measure the noise figure from the downlink preamble [38]. Therefore, I can safely assume that the receiver has the ability to accurately measure σ_n^2 and received SNR and with this assumption we will proceed to calculate σ_u^2.

Determination of Steady-State Minimum Prediction MSE with σ_n^2 and SNR

To find an expression for the driving noise variance, I first determine the variance of the channel in the dynamic channel model given in (3.12):

$$E\left[h_n h_n^*\right] = E\left[\alpha^2 h_{n-1} h_{n-1}^* + u_n u_n^*\right]. \tag{3.22}$$

Since channel gain and driving noise are independent variables and channel can be modeled as zero-mean Gaussian, we can obtain the following expression for the channel variance:

$$\sigma_h^2 = \frac{\sigma_u^2}{1-\alpha^2}. \tag{3.23}$$

Similarly, calculating the variance of (3.13) we can get the expression:

$$E\left[|y|^2\right] = |\beta|^2 \sigma_h^2 + \sigma_n^2. \tag{3.24}$$

Since the received signal can be modeled as a stationary stochastic process, the power of the signal can be defined as the autocorrelation at the origin, $R_{yy}(0)$. Thus, we can take the power of the received signal as, $P_y = R_{yy}(0) = E\left[|y|^2\right]$.

Therefore, the received SNR can be defined as

$$SNR = \frac{\sigma_y^2}{\sigma_n^2}, \tag{3.25}$$

where σ_y^2 is the received signal variance. Substituting Eqs.(3.23) and (3.25) to (3.24) we can obtain the following expression for the driving noise variance:

$$\sigma_u^2 = \frac{\sigma_n^2 \left(1-\alpha^2\right)(SNR-1)}{|\beta|^2}. \tag{3.26}$$

Therefore, given accurate values of AWGN noise variance and received SNR, driving noise variance can be estimated from (3.26). It can be seen that σ_u^2 depends on SNR, and thus changes frequently with variables such as Doppler shift. Therefore, for optimal performance, Kalman filter needs to dynamically track the change of σ_u^2 for the estimator to perform optimally. Finally, substituting (3.26) in to (3.21) gives the steady-state minimum prediction MSE as follows:

$$\begin{aligned}\tilde{m}_{steady} = \sigma_n^2 \Bigg[& \frac{\left(\alpha^2-1\right)(2-SNR)}{2|\beta|^2} + \\ & \frac{\sqrt{\left[\left(1-\alpha^2\right)(2-SNR)\right]^2 + 4\left(1-\alpha^2\right)(SNR-1)}}{2|\beta|^2}\Bigg].\end{aligned} \tag{3.27}$$

Substituting $\tilde{m}_{steady}$ found above to Kalman gain calculation in (3.16) gives the steady-state Kalman gain. Hence, use of (3.27) gives the increased filter performance by operating in the steady-state and provides an efficient mechanism for dynamically tracking the driving noise variance changes. Finally, substituting for β in (3.16) we can find the corresponding Kalman gains individually and then easily construct the Kalman gain matrix, as will be further discussed in the following section.

3.4 Usage and Advantages of Steady-State Kalman gain

Kalman gain can be calculated *offline*, i.e. no need to observe data samples. Therefore, we can calculate the steady-state Kalman gain as discussed in the proposed

scheme once and use this value throughout the filtering process as long as the channel is stationary. A change in channel or SNR will only cause (3.27) to be re-calculated. In the case of QPSK modulated pilot subcarriers as in IEEE802.16e, constructing the steady-state Kalman gain matrix K_n can be easily done by calculating four scalar Kalman gains individually. Since the receiver has the complete knowledge of the pilot data sequence, constructing the Kalman matrix is a trivial task of placing the corresponding Kalman gain in the appropriate diagonal position of the matrix diagonal. This greatly reduces the calculation complexity of the channel estimator because the Kalman filtering steps of minimum prediction MSE calculation and minimum MSE calculation can be avoided altogether. And more importantly this method does not need to perform any matrix inversions, which can get very computationally demanding for large matrix dimensions.

Kalman filters give optimal performance in stationary channels under the steady-state condition. Therefore, our proposed method has the advantage of operating with better performance, given accurate values of σ_n^2 and SNR without having a convergence period. As (3.26) shows, the optimal value of σ_u^2 depends on parameters such as Doppler frequency and channel conditions. Thus, for proper operation of the filter, the value of σ_u^2 needs to be dynamically changed with changing parameters. Use of (3.27) presents an efficient method to track these changes easily with SNR and σ_n^2 and operates the filter at the desired condition. In the following section we will see how filter performance can be substantially degraded if we do not operate the filter at the proper value. A particular steady-state Kalman gain applies to a certain channel condition and when the existing channel condition changes, steady-state Kalman gain needs to be changed too. In the conventional Kalman filter, when the channel condition changes, i.e. when the channel is non-stationary, the filter needs to go through the transient period to converge to the steady-state Kalman gain corresponding to the new channel conditions. However, with the proposed steady-

state scheme, the scheme can directly calculate the steady-state Kalman gain without the need to go through the transient period to converge. Therefore, in non-stationary channel conditions, the proposed scheme has the added advantage of being able to filter with the optimal Kalman gain.

Finally, in the conventional vector Kalman filtering, Kalman gain at each data sample is calculated from the Kalman gain of the previous sample. In each of the multiplication/division operations, there is a degree of round-off error involved. When this continues for each sample, the errors produce a certain margin of error in the filter performance. This is eliminated in the proposed method by only calculating the Kalman gain once (for the duration of the stationarity).

3.5 Numerical Results and Discussion

This section discusses the simulation results. Simulation parameters are as shown in Table. 3.1. I simulate three schemes; conventional Kalman filter, the proposed scheme with steady-state Kalman filtering calculated from (3.21) (proposed-1) and proposed scheme with steady-state Kalman gain calculated from (3.27) (proposed-2) using the SNR to evaluate σ_u^2 for the filtering. In the figures proposed schemes are depicted with filled markers, while the markers corresponding to the conventional scheme are non-filled. Furthermore, in Fig. 3.5 10Hz Doppler is depicted with solid lines and dashed lines are used for the 100Hz Doppler. Results are obtained through Monte Carlo simulations and in each iteration of the simulation, I estimate channel for a duration of 500 symbols. A multipath channel is considered. The channel delay is kept constant while the number of multipaths and power delay profile were changed in each realization of the channel. Each subcarrier was modulated with a random QPSK data symbol and tracked independently as the scalar nature of

the simplified model can determine each Kalman gain individually. Each subcarrier is then estimated in the time-domain to calculate the average filter performance.

Performance of each method is calculated according to the following criteria:

$$\Delta(dB) = 10log_{10}\left[\frac{\sum_{i=0}^{P-1}|\hat{h}_i - h_i|^2}{\sum_{i=0}^{P-1}|h_i|^2}\right]. \tag{3.28}$$

Table 3.1: Simulation parameters

Parameter	Value
Initial parameters: $\hat{h}_{1\|-1}, m_{1\|-1}$	0
Symbol time	100μs
Doppler frequencies	10Hz, 100Hz
Propagation model	ITU-A Vehicular
Number of symbols per filtering	500

Here $\hat{h}_i$ denotes the *estimated* channel value while h_i is the corresponding *true* value. Fig. 3.4 shows the performance of conventional Kalman filter and the proposed-1 scheme for a range of σ_u^2 (by simulations I verify that $0.0001 \sim 0.05$ range for σ_u^2 includes the maximum performance point for our simulation parameters). We simulate at a SNR of 5dB and Doppler shifts of 10Hz and 100Hz. A pattern can be observed as to the filter performance with the varying values of σ_u^2. On the average I can see that the performance of the proposed-1 scheme is better than the conventional scheme for complete range of σ_u^2 values simulated. An important observation is that the point of maximum performance (minimum value) changes with the Doppler frequency. For the conventional scheme, the maximum performance is observed at σ_u^2=0.0095 for the 10Hz Doppler frequency and σ_u^2=0.0255 for the 100Hz case. As is evident from Fig. 3.4, filter performance is highly dependent on the value of σ_u^2, especially when the value is lower than the optimal point of operation, the performance degrades rapidly. In the proposed-1 scheme, the filter gives maximum performance at σ_u^2=0.0070 for the 10Hz Doppler case, while σ_u^2=0.0230 gives the maximum performance for the 100Hz situation. However, if the filtering for the 100Hz Doppler frequency signal is

performed at the σ_u^2 value corresponding to the 10Hz Doppler case for example, the filter performance degrades by 5.72dB from the maximum performance.Therefore it is necessary for the filter to dynamically track the signal and operate at the correct driving noise variance since a small offset can degrade the performance substantially. Fig. 3.4 further shows that the rate of performance degradation is different when the driving noise variance is decreased or increased from the optimal point. Numerical calculations show that the rate of degradation lies in close proximity for both the conventional and proposed schemes when the variance is either increased or decreased. It is clear from the figure that this rate of degradation is higher when σ_u^2 is reduced from the optimal point than it is increased. The channel is modeled as an AR-1 model (3.3) and the driving noise value represents the random noise-like behavior present in the time-varying channel that is inherent in a Rayleigh fading channel. When σ_u^2 approaches zero, it translates to predicting the minimum MSE(3.15) based only on the state transition value α and the random behavior is assumed to be non-existent. Thus the performance degrades rapidly when driving noise variance approach zero. Although the performance degrades when σ_u^2 is increased from the optimal point, the degradation is not rapid because the variance term present in the model helps account for the independent fluctuation present in the channel. Furthermore, we see that at very low values of σ_u^2 both the conventional and proposed schemes perform similarly.

Fig: 3.5 shows the performance of all three schemes against different received SNR's. For the conventional scheme and the proposed-1 scheme, the performances depicted in the graph are those of the best case, i.e. in the simulations I exhaustively searched for the maximum performance point for the range of σ_u^2 values. As depicted, both the proposed schemes give better filter performance than the conventional filter. For the 10Hz Doppler frequency there is an average performance margin of 2.30dB between the conventional scheme and the proposed-1 scheme, while for the 100Hz Doppler shift, the average

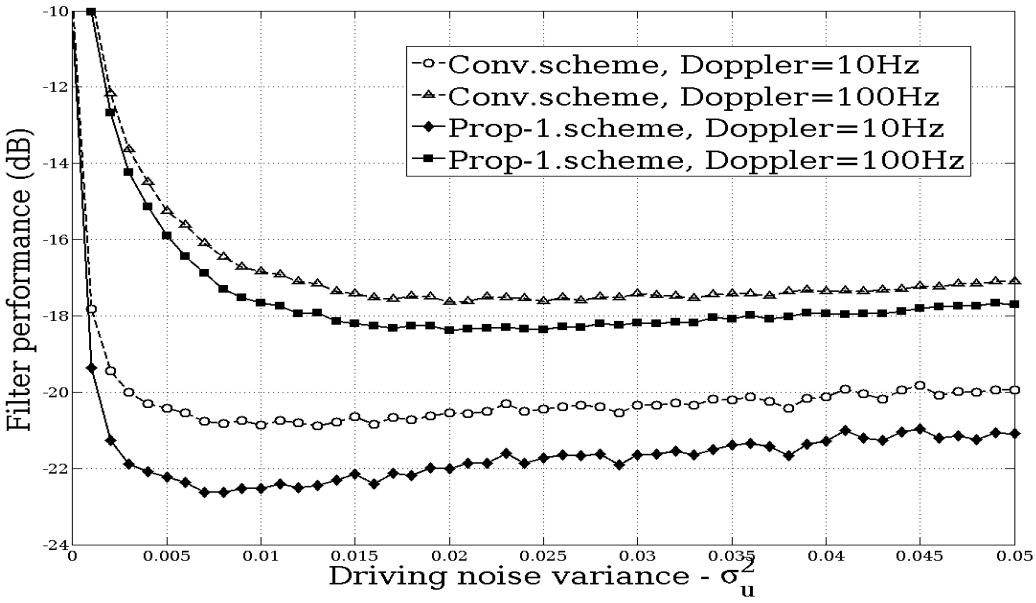

Figure 3.4: Filter performance for varying driving noise variances simulated at 10dB received SNR.

performance margin narrows to 0.98dB, for the SNR range of 2dB to 20dB. Proposed-1 and proposed-2 schemes give similar performance with the former giving slightly better results at higher received SNR's. Since the performance of proposed-1 scheme is obtained through exhaustive search, it can be taken as the optimal filter performance for the simulation parameters I used. Therefore, by observing that the proposed-2 scheme is giving very similar results to that of the proposed-1 scheme, we can deduce that the steady-state minimum prediction MSE found from (3.27) does indeed provide an accurate estimation. On average there is a performance gap of 0.11dB and 0.30dB for 10Hz and 100Hz Doppler frequencies, respectively. The likely reason for this slight reduction in performance is the usage of Eqs.(3.25) and(3.26). Although by definition I have assumed the channel and the received signal to be zero-mean variables but in the simulations, I determine received SNR by calculating the variance of the noise added signal as defined in (3.25) and filtering is performed on 500 symbol-time blocks of data in each iteration.The variance is calculated

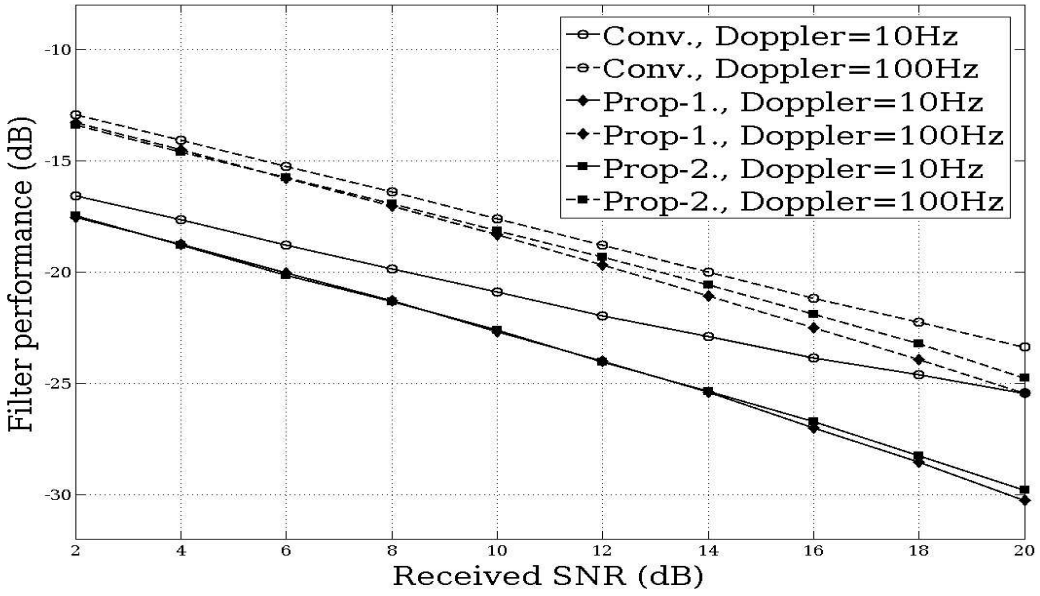

Figure 3.5: Filter performances for varying received SNRs.

as unbiased *sample variance*, which can have an offset from the true variance for a finite data block. Therefore, in simulation there is a certain degree of discrepancy present from the theoretical optimal value of σ_u^2 defined in (3.26). And as we saw in Fig. 3.4, a small deviation of σ_u^2 from the optimal operating point can degrade the performance. Although proposed-2 scheme performs slightly lower than proposed-1 scheme, it must be noted that the performances of the proposed-1 scheme (and conventional scheme) depicted in Fig: 3.5 is that of the best case as discussed above, which is obtained through simulating the schemes for σ_u^2 increments as small as 0.0005. Therefore, from a computational and performance point of view, proposed-2 scheme gives the best performance. The simulation results of Fig. 3.5 above are when the values of average received SNR and noise variance are known accurately at the receiver to calculate the driving noise variance. Although this method provides the accurate value of the driving noise variance, number of parameters such as average received SNR and noise variance σ_n^2 must be evaluated accurately. If the receiver is

unable to accurately determine these values, the optimal driving noise variance cannot be calculated and the filter will not be operating at the best operating state. Furthermore, in simulations I calculated the received SNR beforehand and used this value for the filtering. In practice the filtering needs to be performed in real-time. In such a case, if the previously measured average SNR is different from the signal samples currently receiving, there might be a performance degradation since the value of σ_u^2 is different from the current optimal value. However, it is possible to minimize this degradation by properly increasing the rate of average SNR calculation or operating with a margin on the value of driving noise variance to ensure that it will be greater than the optimal value, since a value less than the optimal value will degrade the performance considerably compared to a value greater than the optimal value as we saw in Fig. 3.4.

Finally, calculation complexity of the schemes is presented here since reducing the complexity is the major objective of the proposed schemes. For better understanding of the calculations involved in the conventional and proposed schemes, I determine the complexity in terms of the number of multiplications(divisions) and additions(subtractions). For the calculation of the square-root in Eqs.(3.21) and (3.27), Newton's method is used since it would allow us to explicitly determine the number of multiplications and additions required for the calculation. Also I consider 10 iterations for the Newton's method for the square-root calculation which would provide a good estimation. Furthermore, complex calculations are considered when determining the number of calculations. Table. 3.2 shows the number of multiplications and additions involved for each simulated method.

We see from the table that the matrix inversions, which requires a power of 3 term, dominates the computational complexity. Fig. 3.6 depicts the number of multiplications needed for the conventional scheme for varying FFT sizes of the IEEE802.16e standard. Values within the parentheses show the number of pilot subcarriers for each FFT size.

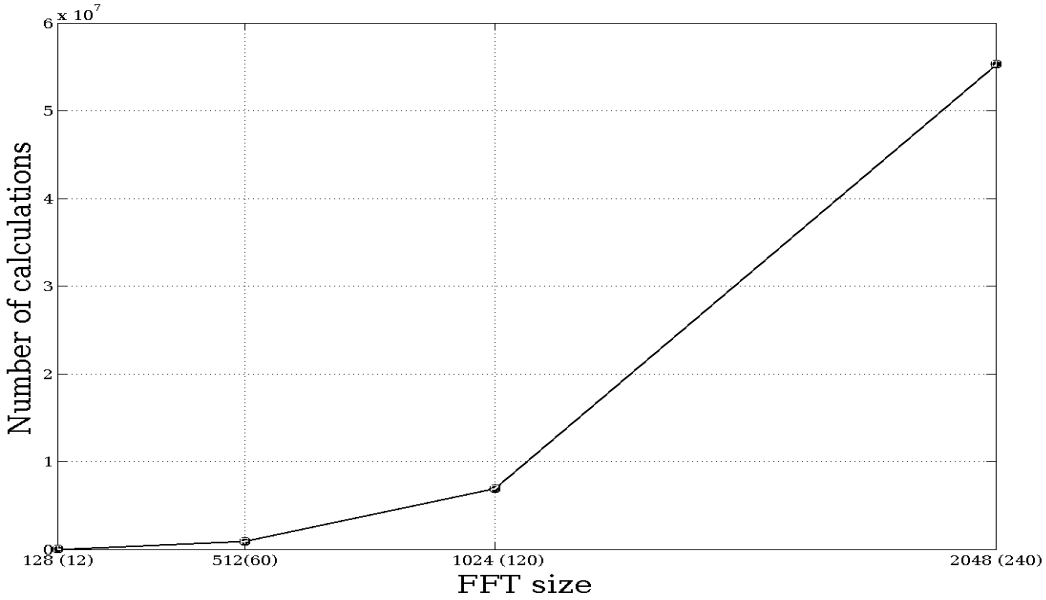

Figure 3.6: Number of multiplications for the conventional scheme for different number of pilot subcarriers.

Table 3.2: Calculation complexity of the conventional and proposed schemes

Scheme	Number of $\times$	Number of $+$	Calculation freq.
Conventional	$4P^3 + 24P$	$4P^3 + 19P$	Each data sample
Proposed-1	63	28	Once for the duration of the channel stationarity
Proposed-2	70	33	

It is clear from the Fig. 3.6 that conventional Kalman filtering needs a large amount of calculations to be performed during each filtering step. Thus, the proposed method provides better performance while requiring only a fraction of the calculations compared to the conventional Kalman filter.

3.6 Conclusion

Kalman filters give their optimum performance at the steady-state condition and lot of computational complexity can be eliminated by determining this steady-state Kalman gain. However, in the conventional vector Kalman filtering it involves solving a Riccati equation to be solved to obtain the steady-state Kalman gain, which is computationally complex. In this chapter, I have derived a steady-state Kalman filter for the OFDM systems by transforming the vector Kalman filtering into the scalar form. This is possible by realizing properties of the OFDM system, specifically for the IEEE802.16e standard. In this case steady-state Kalman gain can be easily determined. The calculation complexity is greatly reduced by calculating the Kalman gain only once in each period of the stationary channel, as opposed to in each data sample in the conventional filter. Furthermore, the value of channel driving noise variance σ_u^2 is also determined as a function of SNR which needs to be dynamically changed with the signal condition for the optimum filter performance. The proposed scheme operates with significantly less computation complexity and the simulations results show that it achieves a substantial performance gain over the conventional scheme.

Chapter 4

Analysis of Quantization Noise in an End-to-End OFDM Link

In sec. 1.4 channel estimation for OFDM is discussed, which indicated how the errors in channel estimations can effect the resource allocation process. In chapter 3 it is shown that the AWGN noise and the driving noise affects the channel estimation process. And all these noises affect the performance of the system. In this chapter I analyze another form of noise that is inherently present in any communication system: *quantization noise*.

Quantization noise arises due to quantization of the signal, which occurs mainly in the Analog to Digital Converter (ADC) and Digital to Analog Converters (DAC). Quantization noise can also occur in other components of the transceiver, including low-pass-filter [39] and the FFT module [40], but in this chapter I am considering the quantization noise arising from the DAC of the transmitter and the ADC of the receiver, since other quantization noises are not controllable to the designer. The objective of this study is to analyze how different components in an OFDM link affects the signal and their dependence on the resulting quantization noise. Quantization noise is often not regarded in the noise budget,

but if the resolution of the quantizer is not high enough, it can be shown that the quantization noise is a function not only of the quantizer resolution, but also of the channel. Since wireless transceivers are required to operate in different channel conditions, this study provides an analytical mean to determine the quantization noise depending on the operating system. Noise present at the receiver

Quantization noise is generally considered as an independent additive noise with an uniform distribution between $[-\frac{q}{2}, \frac{q}{2}]$, where q is the quantization step size. This is called the *pseudo quantization noise (PQN)* model. Although this model can be used to model the noise characteristics in certain situations it has a distinct difference in the output signal PDF. It has been shown in [41] that the PQN model is applicable only when the input signal Characteristic Function (CF) has a certain property, namely it has to satisfy the Quantization Theorem III (QT-III). When the input signal CF does not satisfy QT-III, the quantization noise is no longer uniformly distributed and differs from the PQN model. Furthermore, overflow conditions introduce clipping noise which is no longer restricted to the quantization step size.

Therefore, quantization noise has a strong dependence on the quantizing signal PDF. To properly determine and control the magnitude of the quantization noise, it is important to determine the statistics of the quantizing signal. Having knowledge of the quantizing signal statistics at the receiver provides with the ability to design a quantizer with required dynamic range and quantization step size, for a given performance metric. The quantizer is an integral part of the system and if the dynamic range and the quantization step size, which define the resolution or the number of bits of the quantizer, is not determined properly, the quantizer can have a profound affect on the system performance. In this chapter I analyze the signal flow from the transmitter to the receiver and determine its statistical properties and their dependence on the quantization noise. The results give an

analytical means to determine an optimal quantizer resolution.

In [42], Dardari analyzed the the joint effects of clipping and quantization for an OFDM receiver. Only ADC has been considered and the effects are characterized in the spectral domain. Similarly, [43] also performed a quantization noise analysis for only the ADC component. Here the clipping plus granular noise is evaluated for the receiver ADC for an arbitrary dynamic range and a quantization step size. Statistical properties are not derived in these studies. Analysis in [44] also calculates the required ADC resolution for a Discrete Multitone (DMT) system, again considering only the receiver end of the link.

Quantizer performance is measured by its resolution, which is the number of bits, and the sampling rate. The amount power consumed is proportional to both the resolution and sampling rate [45]. For example, in [46] it is shown that a IEEE802.11a transceiver operating at 54Mbps uses about 21% of the power on the digital-to-analog (DAC) circuitry, while the ADC consumes up to 47% of the power. Generally, a ADC used in such receivers consume power in the order of 1-3.7mW/MHz [47]. It is apparent, thus, the quantizer drains a significant portion of the device power. Furthermore, the *effective* resolution of the quantizer is less than the stated resolution. This is because other noise sources such as circuit noise, aperture noise and comparator ambiguity are present in addition to the quantization noise [48], which further increases the required resolution of the quantizer. Therefore, from a power consumption point of view it is important to carefully choose the quantizer resolution which will give an optimal trade-off between performance and power consumption.

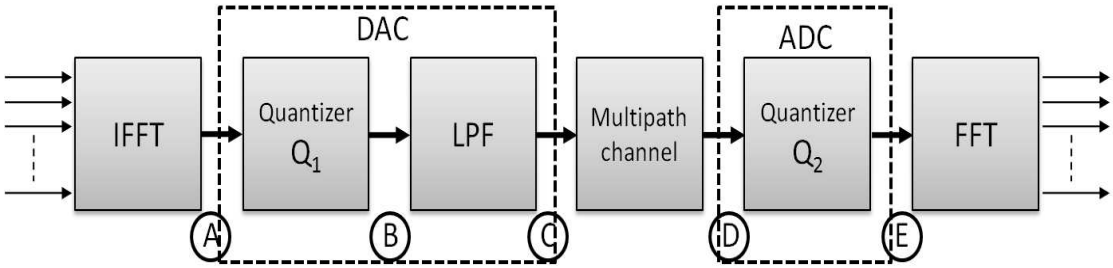

Figure 4.1: End-to-end signal flow of a typical OFDM system

4.1 Signal Statistics Analysis

The signal reaching the receiver goes through a number of operations and transformations, and finally through a multipath fading channel, before arriving at the receiver. The signal is transformed and it's statistical properties are changed during this process and noises are added to the signal which degrade the SNR. Quantization noise contributes to the noise present in a received signal and it is our objective to theoretically obtain the quantization noise power present at the receiver.

Fig. 4.1 shows a schematic of the signal flow from the IFFT module at the transmitter to the FFT module at the receiver. In order to determine the quantization noise, I analyze the statistics of the signal at point A through point D shown in the figure. It should be noted that the analysis is performed on the baseband signal. At points A and E there are quantization noise components arising from the FFT modules [40], but they are not considered in this analysis. It is also important to note that the signal at consideration is complex-valued, but quantization needs to be performed separately for the real and imaginary parts of this signal. Hence, the throughput analysis will consider only one stream of the signal (real or imaginary). The statistics are the same for both.

The IFFT output signal is fed to a DAC to be transmitted through the channel.

The DAC is modeled as a cascade of a quantizer and a low-pass filter (LPF) [41]. The LPF produces a smooth signal to be transmitted. The quantization noise at B arises from the quantizer $\mathtt{Q}_1$. Filter coefficient quantization is present at the LPF but I will not consider it here since it manifests itself as changes to the filter response such as passband ripples and stopband attenuations, rather than a source of additive noise [39].

At this point it is required to determine the statistics of the quantized output signal from the DAC. From a mathematical point of view, the order of the cascade of the quantizer and the LPF can be interchanged without changing the statistical properties of the resulting DAC signal. A LPF followed by a quantizer arrangement simplifies the derivation of output signal statistics. Thus, for analytical purpose I assume that in the DAC, the signal first passes through the LPF and then quantized.

Input to DAC is the IFFT output, which is a set of N random (complex) numbers to be transmitted at sampling intervals of T_s seconds. IFFT input is a zero-mean (non-Gaussian) signal and from Central Limit theorem (CLT) that the output is a zero-mean Gaussian vector $\mathbf{X}_{\mathtt{IFFT}} = [X_0, X_1, \ldots, X_{N-1}]$. LPF is a linear transformation of the input signal and therefore, the output signal will be Gaussian. I need to determine the transformation of the signal mean and the covariance, which can be derived from [49]:

$$\begin{aligned} &E\left\{Y\left(t\right)\right\} = m_X \int h\left(\tau\right) d\tau = m_X H\left(0\right) \\ &E\left\{Y\left(t\right) Y\left(t+\tau\right)\right\} \\ &= \int\int h\left(s\right) h\left(r\right) R_X\left(\tau + s - r\right) dsdr, \end{aligned} \tag{4.1}$$

where $Y(t)$ is the output signal, $h(t)$ is the time-domain filter response with it's frequency transfer function $H(f)$ and $R_X(\tau)$ is the autocorrelation function of the input signal. It is clear from (4.1) that low-pass filtered signal will have zero mean but the autocorrelation is not directly determined. Since the signal is manipulated in its discrete form at sampling

instants, it is easier to determine the statistical properties of the signal autocorrelation when transformations are represented in the discrete form. I define the discretization parameter $\eta = \frac{T_s}{\Delta}$ with $\Delta \to \infty$ corresponding to the continuous case. Let IFFT output be arranged as $\mathbf{X} = [\mathbf{X}_0, \mathbf{X}_1, \ldots, \mathbf{X}_{N-1}]^T$, then the LPF transformation can be written,

$$\mathbf{Y}_{\texttt{LPF}} = \mathbf{H}\mathbf{X}, \tag{4.2}$$

where $\mathbf{H}$ is the $M \times N\Delta$ convolution matrix of the LPF and $\mathbf{Y}_{\texttt{LPF}}$ is the M element output vector. Assuming the filter spans $[-PT_s, PT_s]$ in duration for a positive integer P, we have $M = (2P\Delta + N)$. Transformation matrix $\mathbf{H}$ can be defined by a Toeplitz structure as follows:

$$\mathbf{H} = \begin{pmatrix} \mathbf{0}_0 & \mathbf{0}_1 & \cdots & \mathbf{0}_k & \cdots & \mathbf{0}_{N-1} \\ \mathbf{h} & \mathbf{h} & \cdots & \mathbf{h} & \cdots & \mathbf{h} \\ \mathbf{0}_{N-1} & \mathbf{0}_{N-2} & \cdots & \mathbf{0}_{N-(k+1)} & \cdots & \mathbf{0}_0 \end{pmatrix} \tag{4.3}$$

where $\mathbf{0}_k$ is a $k \times 1$ vector of zeros and

$$\mathbf{h} = [h(-P\Delta\eta), \cdots, h((P\Delta - 1)\eta), h(P\Delta\eta)]^T, \tag{4.4}$$

is the $(2P\Delta+1)$ element vector of the filter response where $h(\cdot)$ is the filter impulse response defined in (4.1). The IFFT output can be interleaved by sampling times by defining $\mathbf{X}_k = [X_k \quad \mathbf{0}_{\Delta-1}^T]$, where X_k is the k-th IFFT output.

Defining the LPF transformation as above, I am able to determine the PDF of $\mathbf{Y}_{\texttt{LPF}}$ as [50],

$$\mathbf{f}_{\mathbf{Y}_{\texttt{LPF}}}(\mathbf{y}) = \mathbf{f}_{\mathbf{X}}(\mathbf{H}^{-1}\mathbf{y}). \tag{4.5}$$

Since $\mathbf{X}$ is a zero-mean random vector,

$$\mathbf{f}_{\mathbf{Y}_{\texttt{LPF}}}(\mathbf{y}) = \frac{1}{(2\pi)^{\frac{M}{2}}|\mathbf{C}_{\mathbf{X}}|^{\frac{1}{2}}} \exp\left[-\frac{1}{2}\mathbf{y}^T(\mathbf{H}\mathbf{C}_{\mathbf{X}}\mathbf{H}^T)^{-1}\mathbf{y}\right] \tag{4.6}$$

Therefore, the LPF output is a zero-mean Gaussian random vector with covariance $\mathbf{C}_{\mathbf{Y}_{\mathtt{LPF}}} = (\mathbf{H}\mathbf{C}_{\mathbf{X}}\mathbf{H}^T)$, where $\mathbf{C}_{\mathbf{X}}$ is the $N\Delta \times N\Delta$ covariance matrix of $\mathbf{X}$. If required $\mathbf{H}$ can be made square by appending zero row vectors. Since $\mathbf{X}$ is a zero-interleaved vector of zero-mean independent Gaussian random variables $\mathbf{X}_{\mathtt{IFFT}}$, $\mathbf{C}_{\mathbf{X}}$ is a diagonal matrix with its entries given by,

$$\mathrm{diag}(\mathbf{C}_{\mathbf{X}}) = [\sigma^2_{X_0}\, \mathbf{0}_{\Delta-1}\, \sigma^2_{X_1}\, \mathbf{0}_{\Delta-1} \cdots \sigma^2_{X_{N-1}}\, \mathbf{0}_{\Delta-1}], \tag{4.7}$$

with $\sigma^2_{X_k}$ the variance of X_k and I assume $\sigma^2_{X_0} = \sigma^2_{X_1} = \cdots = \sigma^2_{X_{N-1}} = \sigma^2_{\mathbf{X}}$.

Next step is the quantization. As was shown earlier, the quantized signal PDF of a Gaussian input is not Gaussian, rather a *Gaussian-shaped* impulsive PDF given by [41],

$$\mathbf{f}_{\mathbf{Y}_{\mathtt{Q}_1}}(y) = \sum_{m=-\infty}^{\infty} \delta(y - mq_1) \int_{mq_1-\frac{q_1}{2}}^{mq_1+\frac{q_1}{2}} \mathbf{f}_{\mathbf{Y}_{\mathtt{LPF}}}(y)dy, \tag{4.8}$$

where $\mathbf{f}_{\mathbf{Y}_{\mathtt{LPF}}}(y)$ is the Gaussian input PDF and q_1 is the quantization step size of $\mathtt{Q}_1$. The $\mathtt{Q}_1$ output is clearly not Gaussian distributed. Top of Fig. 4.2 shows the histogram of a single realization of a 1024-point IFFT output corresponding to a 64-QAM input data. The figure below shows the histogram of the corresponding quantized IFFT output with a quantization step size of $q_1 = 0.1\sigma_{\mathbf{X}}$. It is evident that the histogram is not continuous in its range because of the quantizing intervals. Furthermore, the frequencies of the histogram of the quantized data are increased due to a range of values stacking up in to a single bin. Although the distribution of the quantized data is not Gaussian, we see later that only the knowledge of the mean and the variance of this distribution is adequate for the analysis. To find these quantities characteristic functions (CF) are used. CF of $\mathbf{f}_{\mathbf{Y}_{\mathtt{Q}_1}}(y)$ can be expressed as

$$\Phi_{\mathbf{Y}_{\mathtt{Q}_1}}(\omega) = \sum_{l=-\infty}^{\infty} \Phi_{\mathbf{Y}_{\mathtt{LPF}}}(\omega + l\Psi)\mathrm{sinc}\left(\frac{q_1(\omega + l\Psi)}{2}\right), \tag{4.9}$$

where $\Psi = \frac{2\pi}{q_1}$. From (4.6) we have, for the vector case with zero-mean,

$$\Phi_{\mathbf{Y}_{\mathtt{LPF}}}(\boldsymbol{\omega}) = \exp\left(-\frac{1}{2}\boldsymbol{\omega}^T \mathbf{C}_{\mathbf{Y}_{\mathtt{LPF}}}\boldsymbol{\omega}\right). \tag{4.10}$$

For the Gaussian input, [41] has shown that the PQN model is very closely satisfied and therefore the moments of $\mathbf{f}_{\mathbf{Y}_{\mathtt{Q}_1}}(y)$ is well approximated by

$$\frac{d^r \Phi_{\mathbf{Y}_{\mathtt{Q}_1}}(\omega)}{d^r \omega} = \frac{d^r \Phi_{\mathbf{Y}_{\mathtt{LPF}}(\omega)}}{d^r \omega} + M_r, \tag{4.11}$$

where M_r is the *moment difference*. From (4.11) we have $M_1 = 0$ and $M_2 = \frac{q_1^2}{12}$ giving,

$$E\left\{\mathbf{Y}^2_{\mathtt{Q}_1}\right\} = E\left\{\mathbf{Y}^2_{\mathtt{LPF}}\right\} + \frac{q_1^2}{12}. \tag{4.12}$$

Hence, the DAC output signal is zero-mean with covariance matrix

$$\mathbf{C}_{\mathbf{Y}_{\mathtt{Q}_1}} = (\mathbf{H}\mathbf{C}_{\mathbf{X}}\mathbf{H}^T) + \frac{q_1^2}{12}\mathbf{I}_{M\times N}. \tag{4.13}$$

Therefore, although the input signal to DAC is a set of independent random variables, the output is a jointly distributed set of random variables with covariance matrix $\mathbf{C}_{\mathbf{Y}_{\mathtt{Q}_1}}$.

The next stage of the signal flow is the transmission. A wideband Rayleigh fading channel with sample-spaced paths is considered. Each received signal component at the receiver is a complex-number with the real and imaginary components being zero-mean Gaussian random variables. This is due to the superposition of multiple independent random variables, or the CLT. But since our transmission signal is jointly distributed, direct application of the CLT to determine statistical properties of the received signal is not possible. Since signals are processed in their discrete form at the receiver, I analyze the signal at its sampling instants.

I consider a low-pass Nyquist filter in the LPF stage with the impulse response characteristics $h(0) = 1$ and $h(lT_s) = 0, l = \pm 1, \pm 2, \ldots$. Then, from (4.3) and (4.7), the covariance matrix $\mathbf{C}_{\mathbf{Y}_{\mathtt{Q}_1}}$ has the following properties:

$$\mathbf{C}_{\mathbf{Y}_{\mathtt{Q}_1}}(l\Delta, l\Delta) = \sigma^2_{\mathbf{X}} + \frac{q_1^2}{12}, \quad l = 0, 1, \ldots, N-1, \tag{4.14}$$

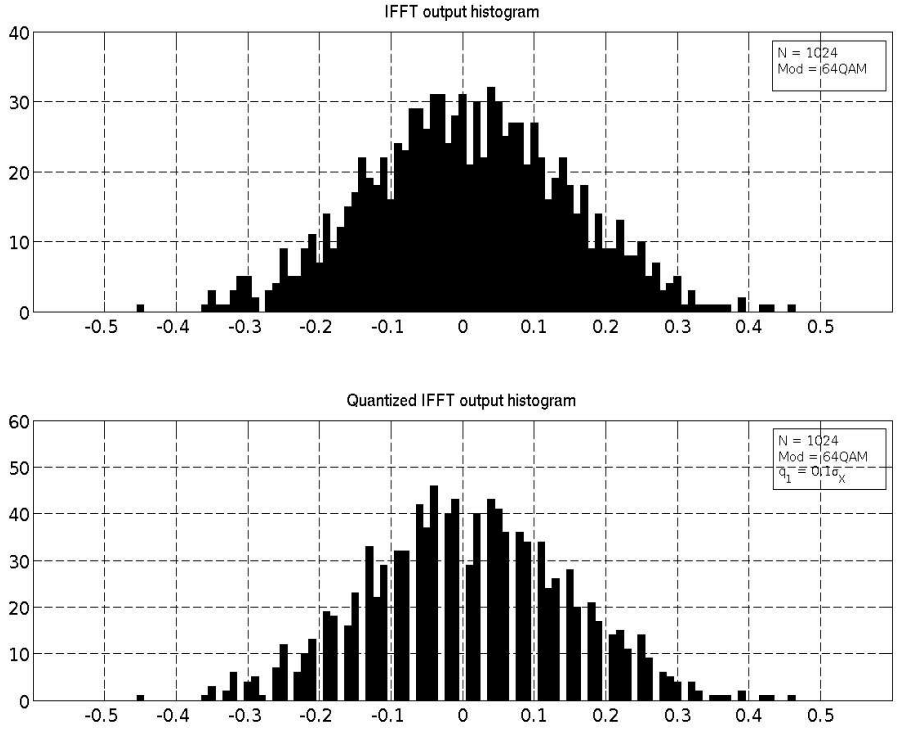

Figure 4.2: Histograms of an IFFT output (above) and quantized version of it (below).

$$\mathbf{C}_{\mathbf{Y}_{Q_1}}(k\Delta, l\Delta) = 0, \ k \neq l, k, l = 0, 1, \ldots, N-1, \tag{4.15}$$

where $\mathbf{C}_{\mathbf{Y}_{Q_1}}(k,l)$ is the (k,l)-th element of $\mathbf{C}_{\mathbf{Y}_{Q_1}}$ starting from zero. Thus, we have that the transmitting signal is an independent random variable at the sampling instants with variance $\sigma^2_{\mathbf{Y}_{Q_1}} = \sigma^2_{\mathbf{X}} + \frac{q_1^2}{12}$. It should be noted here that the transmitting signal statistics are altered from the quantization process and it depends on the quantization step size.

4.1.1 Multipath Propagation

This section determines the change of signal statistics during the next stage of the signal flow, multipath propagation. The propagation over the wideband Rayleigh fading

channel can be expressed as [51],

$$y_{\mathtt{RX}}(t) = \sum_{k=0}^{L-1} A_k e^{(-j2\pi f_0 \tau_k)} y_{\mathtt{TX}}(t-\tau_k), \tag{4.16}$$

where $y_{\mathtt{RX}}(t)$ is the received complex baseband signal, $y_{\mathtt{TX}}(t)$ is the transmitted signal and is complex with its real and imaginary components coming from separate LPF stages. Multipath component's complex fading coefficient and delay are given by A_k, where A_k is Rayleigh distributed and τ_k, respectively. Passband frequency is f_0 and L is the number of multipath components. Omitting the phase rotation, after sampling the received signal of (4.16) at the proper intervals, a discrete-time sample can be decomposed to a combination of complex terms as

$$\begin{aligned} y_{\mathtt{BB}}(lT_s) &= \sum_{k=0}^{L-1} \left(a_{r,k} + j a_{i,k}\right)\left(y_{r,k} + j y_{i,k}\right) \\ &= \underbrace{\left(\sum_k a_{r,k} y_{r,k} - \sum_k a_{i,k} y_{i,k}\right)}_{I_y(lT_s)} \\ &\quad + j \underbrace{\left(\sum_k a_{r,k} y_{i,k} + \sum_k a_{i,k} y_{r,k}\right)}_{Q_y(lT_s)}, \end{aligned} \tag{4.17}$$

where subscripts r and i denote the corresponding real and imaginary components, respectively. The received signal points $y_{\mathtt{BB}}(lT_s)$ is complex with in-phase and quadrature components given by $I_y(lT_s)$ and $Q_y(lT_s)$. As with the transmitter, I consider either the real or the imaginary part for the quantization noise analysis. To apply CLT to each summation I first need to find the mean and variance of the product terms $a_k x_k$. Each of the four products has the same statistics. Mean is given by,

$$\begin{aligned} E\left\{AY_{\mathtt{Q}_1}\right\} &= \int ay\, \mathbf{f}_{\mathbf{A}\mathbf{Y}_{\mathtt{Q}_1}}(a,y) da dy \\ &= \int ay\, \mathbf{f}_{\mathbf{A}}(a) \mathbf{f}_{\mathbf{Y}_{\mathtt{Q}_1}}(y) da dy = E\left\{A\right\} E\left\{Y_{\mathtt{Q}_1}\right\} = 0, \end{aligned} \tag{4.18}$$

where I use the independence of the variables. Similarly the variance can be calculated as,

$$\begin{aligned} E\left\{A^2 Y_{\mathtt{Q}_1}^2\right\} &= \int a^2 y^2 \mathbf{f}_{\mathbf{A}\mathbf{Y}_{\mathtt{Q}_1}}(a,y) da dy \\ &= E\left\{A^2\right\} E\left\{Y_{\mathtt{Q}_1}^2\right\} = \sigma_{\mathtt{chan}}^2 \sigma_{\mathbf{Y}_{\mathtt{Q}_1}}^2, \end{aligned} \tag{4.19}$$

where $E\left\{A^2\right\} = \sigma_{\mathtt{chan}}^2$ is the variance of the channel. Therefore, from CLT we have that the $\sum_k a_k y_k$ terms are zero-mean with variance $\sigma_\Sigma^2 = \mathrm{L}\sigma_{\mathbf{Y}_{\mathtt{Q}_1}}^2 \sigma_{\mathtt{chan}}^2$. Next, to find the statistics of $I_y(t)$ and $Q_y(t)$, we need to determine the statistics of difference and summation of Gaussian variables, respectively. We have as the CF of the $\sum_k a_k y_k$ terms [52],

$$\Phi_\Sigma(\omega) = \exp\left(-\frac{1}{2}\sigma_\Sigma^2 \omega^2\right). \tag{4.20}$$

To find the CF of the difference of two such variables, from the definition of CF

$$\Phi_{\mathtt{I}_\mathtt{y}}(\omega) = E\left\{e^{j\omega(\Sigma - \Sigma)}\right\}. \tag{4.21}$$

Using the independence of the terms,

$$\Phi_{\mathtt{I}_\mathtt{y}}(\omega) = \Phi_\Sigma(\omega)\Phi_\Sigma(-\omega) = \left[\exp\left(-\frac{1}{2}\sigma_\Sigma^2 \omega^2\right)\right]^2. \tag{4.22}$$

Similarly, for the summation of terms, the CF is

$$\Phi_{\mathtt{Q}_\mathtt{y}}(\omega) = \left[\exp\left(-\frac{1}{2}\sigma_\Sigma^2 \omega^2\right)\right]^2 = \Phi_{\mathtt{I}_\mathtt{y}}(\omega). \tag{4.23}$$

Therefore, the statistics of $I_y(t)$ and $Q_y(t)$ components are identical. Their mean is found from the CF to be

$$E\left\{\Phi_{\mathtt{I}_\mathtt{y}}(\omega)\right\} = \frac{1}{j}\frac{d\Phi_{\mathtt{I}_\mathtt{y}}(\omega)}{d\omega}\bigg|_{\omega=0} = 0. \tag{4.24}$$

Finally, variance is found from

$$\begin{aligned} E\left\{\Phi_{\mathtt{I}_\mathtt{y}}^2(\omega)\right\} &= \frac{1}{j^2}\frac{d^2\Phi_{\mathtt{I}_\mathtt{y}}(\omega)}{d^2\omega}\bigg|_{\omega=0} \\ &= 2\sigma_\Sigma^2 = 2\mathrm{L}\sigma_{\mathbf{Y}_{\mathtt{Q}_1}}^2 \sigma_{\mathtt{chan}}^2. \end{aligned} \tag{4.25}$$

The mean and variance statistics obtained in (4.24) and (4.25) for difference of products in $I_y(t)$, are applied identically to the summation of products in $Q_y(t)$.

Finally, the signal is corrupted with i.i.d additive white Gaussian noise (AWGN). AWGN is zero-mean Gaussian and it is added to the multipath component which I found earlier to be zero-mean Gaussian. This addition of two independent Gaussian random variables results in a received Gaussian distributed signal vector $\mathbf{Y}_{\mathtt{RX}}$ with the following PDF:

$$\mathbf{f}_{\mathbf{Y}_{\mathtt{RX}}}(\mathbf{y}) = \frac{1}{(2\pi)^{\frac{M}{2}}|\mathbf{C_{RX}}|^{\frac{1}{2}}}\exp\left[-\frac{1}{2}\mathbf{y}^T\mathbf{C}_{\mathbf{RX}}^{-1}\mathbf{y}\right], \tag{4.26}$$

where $\mathbf{C_{RX}}$ is the covariance matrix of $\mathbf{Y}_{\mathtt{RX}}$ given by

$$\mathbf{C_{RX}} = \left(2\mathrm{L}\sigma^2_{\mathtt{cha}}\left(\sigma^2_{\mathbf{X}} + \frac{q_1^2}{12}\right) + \sigma^2_{\mathtt{n}}\right)\mathbf{I}_{N\times N}, \tag{4.27}$$

with $\sigma^2_{\mathtt{n}}$ denoting AWGN noise variance and $\mathbf{I}_{N\times N}$ is a $N \times N$ identity matrix.

4.2 Quantization Noise Analysis

Quantization noise introduced at the ADC quantizer is determined here. Granular noise only and clipping (overflow) plus granular noise cases are considered separately.

Quantization noise differs from the PQN model described earlier if the CF of the

input signal does not satisfy Quantization theorem-III, which states the following [41]:

QuantizationTheorem – III

if

$$\Phi_x(l\Psi) = 0, \quad l = 1, 2, \ldots$$

then the quantization noise would have the CF

$$\Phi_v(l\Psi) = sinc\left(\frac{q\omega}{2}\right),$$

and its PDF would be

$$f_v(x) = \begin{cases} \frac{1}{q} & \text{if } \mathsf{x} \leqslant q/2 \\ 0 & \text{elsewhere.} \end{cases}$$

Therefore, the input signal to the DAC does not comply with the requirements of QT-III although they are shown to exhibit very small deviation from the PQN model. Nevertheless, to make our analysis general, I will consider the general form of the quantization noise power which is given by

$$E\left\{v^2\right\} = \frac{q^2}{12} + \frac{q^2}{\pi^2}\sum_{l=1}^{\infty} \Re\left\{\Phi(l\Psi)\right\} \frac{(-1)^l}{l^2}, \tag{4.28}$$

where $E\left\{v^2\right\}$ is the quantization noise variance and q is the quantization step size. $\Re\left\{\Phi(\bullet)\right\}$ is the real part of the CF of the input signal.

Fortunately, as I determined in the earlier section, both the input signals to the quantizers $\mathtt{Q_1}$ and $\mathtt{Q_2}$ (Fig. 4.1), $\mathbf{X}$ and $\mathbf{Y}_{\mathtt{RX}}$, are Gaussian signals which have a CF of the kind

$$\Phi_{\mathtt{Gauss}}(t) = \exp\left(-\frac{1}{2}\sigma_x^2 t^2\right) \tag{4.29}$$

for the zero-mean case. Due to the symmetry and the fast decay of the Gaussian PDF, the quantization noise power expression can be simplified to

$$E\left\{v^2\right\} = \frac{q^2}{12} - \frac{q^2}{\pi^2}\exp\left(-2\pi^2\left(\frac{\sigma_x^2}{q}\right)^2\right). \tag{4.30}$$

The uniform quantization noise therefore deviates from the PQN model depending on the magnitude of the ratio of input signal variance to the quantization step size. Smaller quantization step size can make the quantization noise power similar to the PQN model.

4.2.1 Quantization noise at the transmitter

For the zero-mean Gaussian DAC input signal $\mathbf{X}$, we have its CF

$$\Phi_{\mathbf{X}}(t) = \exp\left(-\frac{1}{2}\sigma_{\mathbf{X}}^2 t^2\right), \tag{4.31}$$

and the power of the quantization noise introduced at the DAC, v_{Q_1}, is then given by,

$$E\left\{v_{\mathtt{Q}_1}^2\right\} = \frac{q_1^2}{12} + -\frac{q_1^2}{\pi^2}\exp\left(-2\pi^2\left(\frac{\sigma_{\mathbf{X}}^2}{q_1}\right)^2\right). \tag{4.32}$$

4.2.2 Quantization noise at the Receiver

Signal $y_{\mathtt{RX}}(t)$ is received with the PDF given in (4.26) and is sampled and then quantized at quantizer $\mathtt{Q}_2$. Following types of quantization noises for the ADC $\mathtt{Q}_2$ are considered: 1) uniform quantization noise, $v_{\mathtt{Q}_2,\mathtt{UQ}}$, and 2) floating-point quantization noise, $v_{\mathtt{Q}_2,\mathtt{FQ}}$ and 3) uniform quantization noise with overflow. It should be reminded that in the analyses of quantization noise in the transmitter, uniform quantization with no overflow was considered. From (4.28) we can calculate the uniform quantization noise to be

$$E\left\{v_{\mathtt{Q}_2,\mathtt{UQ}}^2\right\} = \frac{q_2^2}{12} - \frac{q_2^2}{\pi^2}\exp\left(-2\pi^2\left(\frac{\sigma_{\mathbf{Y}_{\mathtt{Rx}}}^2}{q_2^2}\right)\right), \tag{4.33}$$

where q_2^2 is the quantization step size of $\mathtt{Q}_2$ and $\sigma_{\mathbf{Y}_{\mathtt{RX}}}^2$ is the scalar quantity of $\mathbf{C_{RX}}$, which is the received signal variance. Under the conditions that the input signal is zero-mean Gaussian, [41] states the floating point quantization noise to be given by

$$E\left\{v_{\mathtt{Q}_2,\mathtt{FQ}}^2\right\} = (2.16)E\left\{v_{\mathtt{PQN}}^2\right\}E\left\{\frac{\mathbf{Y}_{\mathtt{Rx}}^2}{\Theta^2}\right\}, \tag{4.34}$$

where $E\left\{v_{\mathtt{PQN}}^2\right\} = \frac{q_2^2}{12}$ is the PQN model quantization noise. By definition $\Theta \triangleq 2^p q_2$ and p is the number of bits of the mantissa. Then (4.34) can be simplified to give

$$\begin{aligned} E\left\{v_{\mathtt{Q}_2,\mathtt{FQ}}^2\right\} &= (0.180)2^{-2p} E\left\{\mathbf{Y}_{\mathtt{Rx}}^2\right\} \\ &= (0.180)2^{-2p}\left(2\mathrm{L}\sigma_{\mathtt{cha}}^2\left(\sigma_{\mathbf{X}}^2 + \frac{q_1^2}{12}\right) + \sigma_{\mathtt{n}}^2\right). \end{aligned} \tag{4.35}$$

Thus, we see that the uniform quantization noise and floating-point quantization noise is a dependent on both transceiver and channel parameters.

Quantization noise with overflow

In (4.33) I assumed a quantizer with no overflow. But from the PDF given in (4.26) it can be seen that the received signal has much higher variance and it is possible that overflowing will occur which will give additional clipping noise over the granular noise I considered. From the equations derived earlier, it is possible to design an ADC for a certain dynamic range that clips a particular percentage of the PDF tails.

We can find the dynamic range $[-R_{\mathtt{ADC}}, R_{\mathtt{ADC}}]$ that will clip a tolerable fraction of the received signal PDF tails and quantizes a $\mathtt{Q}_{\mathtt{FRAC}}$ fraction of the signal. The range can then be found by using the symmetry of the PDF,

$$\int_{-R_{\mathtt{ADC}}}^{R_{\mathtt{ADC}}} \mathbf{f}_{\mathbf{Y}_{\mathtt{RX}}}(y) = P(-R_{\mathtt{ADC}} \leq y \leq R_{\mathtt{ADC}}) = \mathtt{Q}_{\mathtt{FRAC}}. \tag{4.36}$$

Since $\mathbf{F}_{\mathbf{Y}_{\mathtt{RX}}}(y)$, CDF of $\mathbf{f}_{\mathbf{Y}_{\mathtt{RX}}}(y)$, is an odd function, we have

$$\int_{-R_{\mathtt{ADC}}}^{R_{\mathtt{ADC}}} \mathbf{f}_{\mathbf{Y}_{\mathtt{RX}}}(y) = 2\mathbf{F}_{\mathbf{Y}_{\mathtt{RX}}}(R_{\mathtt{ADC}}) = 1 + \mathrm{erf}\left(\frac{R_{\mathtt{ADC}}}{\sigma_{\mathbf{Y}_{\mathtt{RX}}}\sqrt{2}}\right). \tag{4.37}$$

Above equation can be solved for a required $R_{\mathtt{ADC}}$. The resulting quantization noise differs from the PQN model, even for a CF satisfying QT-III, due to the clipping. The noise can be found the following way.

$$E\left\{v_{\mathtt{Q}_2}^2\right\} = \int_{-\infty}^{\infty} \left(x - \mathtt{Q}_2(x)\right)^2 \mathbf{f}_{\mathbf{Y}_{\mathtt{RX}}}(x)dx. \tag{4.38}$$

Using symmetry of the PDF, we can decompose the above to include the dynamic range as,

$$E\left\{v_{\mathtt{Q}_2}^2\right\} = \underbrace{2\int_{R_{\mathtt{ADC}}}^{\infty}\left(x-R_{\mathtt{ADC}}\right)^2 \mathbf{f}_{\mathbf{Y}_{\mathtt{RX}}}(x)dx}_{N_{\mathtt{clip}}} + \underbrace{2\sum_{n=0}^{Q_L-1}\int_{x_n}^{x_{n+1}}\left(x-\mathtt{Q}_2(x)\right)^2 \mathbf{f}_{\mathbf{Y}_{\mathtt{RX}}}(x)dx}_{N_{\mathtt{gran}}}, \tag{4.39}$$

where $Q_L = R_{\mathtt{ADC}}/q_2$ is the number of quantization levels in each half of the dynamic range. Considering a *midriser*-type quantizer, $\mathtt{Q}_2(x) = \left(\frac{2n+1}{2}\right) q_2$, for $x_n \leq x \leq x_{n+1}$. I divide the quantization noise in to granular noise and clipping noise components, given respectively by, $N_{\mathtt{gran}}$ and $N_{\mathtt{clip}}$. Expanding (4.39) and with a bit of manipulations I obtain the following expressions for the quantization noises.

$$\begin{aligned} N_{\mathtt{clip}} = 2R_{\mathtt{ADC}}\Bigg\{ & \frac{R_{\mathtt{ADC}}}{2}\left(1-\operatorname{erf}\left(\frac{R_{\mathtt{ADC}}}{\sigma_{\mathbf{Y}_{\mathtt{RX}}}\sqrt{2}}\right)\right) + \chi(R_{\mathtt{ADC}}) \\ & + R_{\mathtt{ADC}}^2\,\chi(R_{\mathtt{ADC}}) + \frac{\sigma_{\mathbf{Y}_{\mathtt{RX}}}^2}{2}\left(1-\operatorname{erf}\left(\frac{R_{\mathtt{ADC}}}{\sigma_{\mathbf{Y}_{\mathtt{RX}}}\sqrt{2}}\right)\right)\Bigg\}, \end{aligned} \tag{4.40}$$

$$\begin{aligned} N_{\mathtt{gran}} = 2\sum_{n=0}^{Q_L-1}\Bigg\{ & \left(\frac{2n+1}{2}\right)^2 q_2^2\left[\mathbf{F}_{\mathbf{Y}_{\mathtt{RX}}}(x_{n+1}) - \mathbf{F}_{\mathbf{Y}_{\mathtt{RX}}}(x_n)\right] \\ & - (2n+1)\,q_2\chi(\varphi)\Big|_{x_n}^{x_{n+1}} - \varphi^2\,\chi(\varphi)\Big|_{x_n}^{x_{n+1}} \\ & + \sigma_{\mathbf{Y}_{\mathtt{RX}}}^2\left[\mathbf{F}_{\mathbf{Y}_{\mathtt{RX}}}(x_{n+1}) - \mathbf{F}_{\mathbf{Y}_{\mathtt{RX}}}(x_n)\right]\Bigg\}, \end{aligned} \tag{4.41}$$

with $\chi(\varphi)$ defined as

$$\chi(\varphi) = \frac{\sigma_{\mathbf{Y}_{\mathtt{RX}}}^2}{\sqrt{2\pi\sigma_{\mathbf{Y}_{\mathtt{RX}}}^2}}\left(e^{-\frac{\varphi^2}{2\sigma_{\mathbf{Y}_{\mathtt{RX}}}^2}}\right). \tag{4.42}$$

The quantization step size q_2 can be derived from evaluating (4.41) numerically. Therefore, following the above procedure, it is possible to design a receiver ADC according to the transceiver and propagation channel characteristics and a required receiver performance such as the BER. Also it provides an analytical method to determine the effect of quantization noise for a given set of device and link parameters.

4.3 Simulation results and performance evaluation

Simulations are performed to first determine the validity of the theoretical model derived in section 4.1 with simulated transmission scenarios. Then the effect of quantization noise on BER is simulated under different system parameters.

4.3.1 Validity of the model

Fig. 4.3 shows results of the received signal variance for FFT sizes of 512, 1024 and 2048 points with 64-QAM modulation level obtained through Monte Carlo simulations. Quantization step size q_1 and AWGN noise are kept constant. The dashed lines show the theoretical value I derived for the received signal variance in (4.27) and the markers show the corresponding values obtained from the simulations. A good agreement between the theoretical values and simulated results can be observed. It is further seen that the discrepancy with the theoretical value reduces with the increasing number of data points (large FFT).

Fig. 4.4 shows the deviation of the simulated results for the theoretical model as a function of the quantization step size. Simulations of Fig. 4.3 used a constant transmitter quantization step size of $q_1 = 0.5\sigma_{\mathbf{X}}$ and it showed a good agreement between theoretical and simulated values. Here I change q_1 from $0.1\sigma_{\mathbf{X}}$ to $3.0\sigma_{\mathbf{X}}$ and repeat the simulations while keeping other parameters constant. The solid lines show the theoretical value and the dashed lines show the simulated results. Curves show that the theoretical values are tightly held until the quantization step size is increased to about $2.0\sigma_{\mathbf{X}}$, after which point the simulated results start to deviate from the theoretical limit. The larger quantization step size effectively reduces the number of quantizer bins and the output PDF starts to deviate more from the Gaussian shape. This effect is much more apparent by observing that the rate of deviation is larger for the smaller FFT size. At a particular step size, the

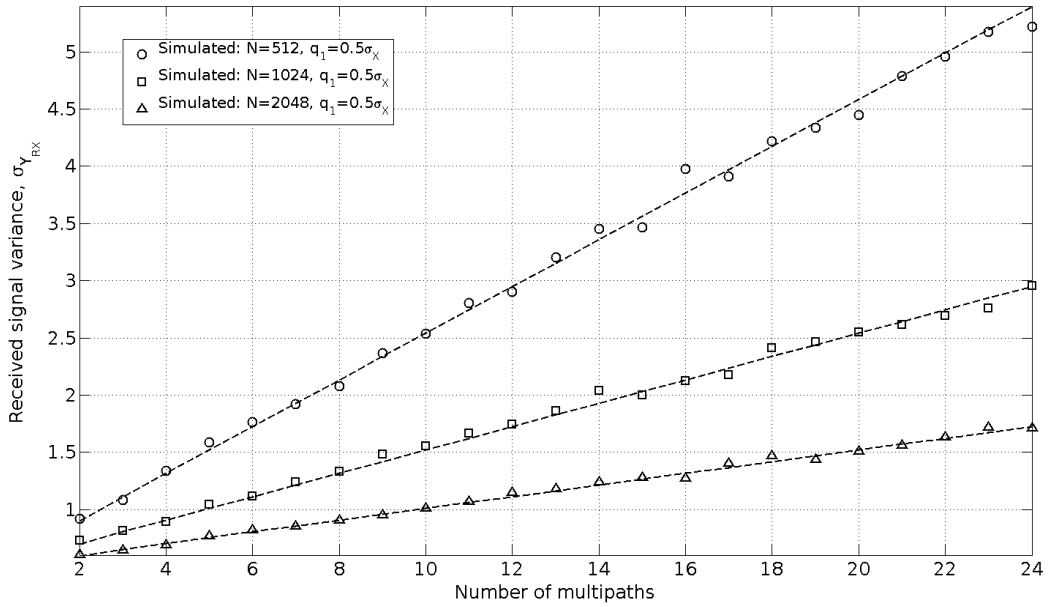

Figure 4.3: Comparison of the theoretical model for $\sigma^2_{\mathbf{Y}_{\mathrm{RX}}}$ against the results of simulated transmissions

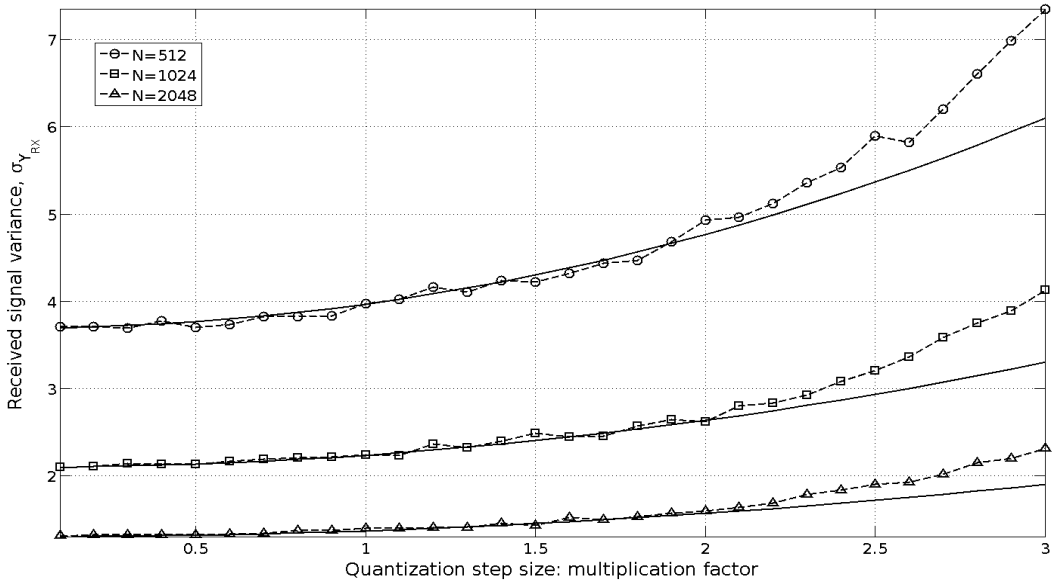

Figure 4.4: Deviation of the simulated results from the theoretical model at larger quantization step sizes

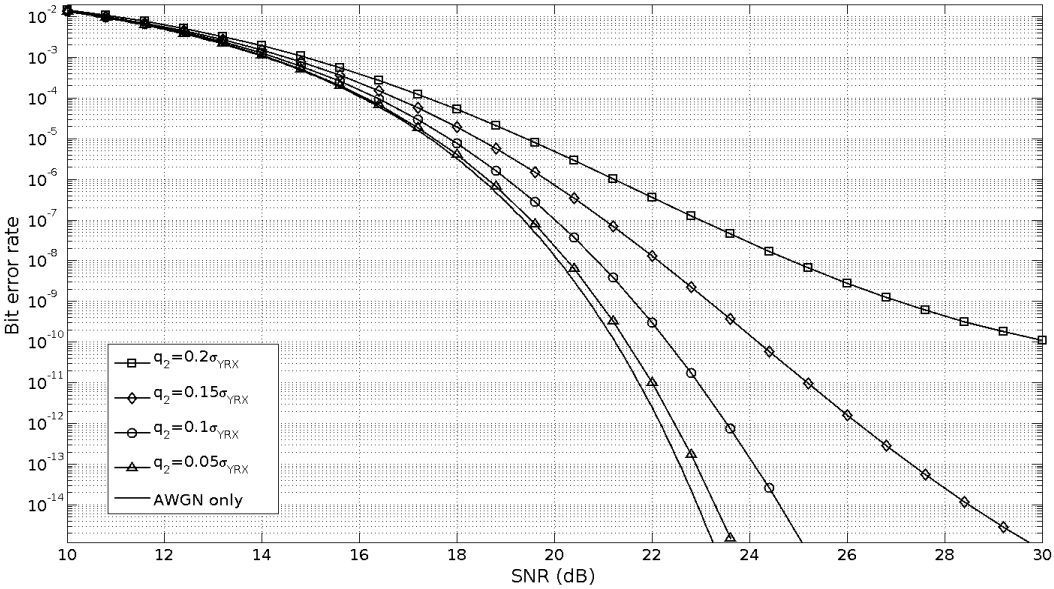

Figure 4.5: BER performance of the uniform quantizer for varying quantizer resolutions

deviation is higher for the 512-point FFT compared to the 1024- or 2048-point FFTs. It should be noted, that although simulated, step sizes as large as $2.0\sigma_{\mathbf{X}}$ are not practical quantizer resolutions as it introduces a large quantization noise that results in undesirable error rates as can be seen from the following section. Considering Figs. 4.3 and 4.4, we can assume that the theoretical results derived can model a practical transmission accurately.

4.3.2 Effect on BER performance

The following simulations show the quantization effect on the BER performance. A 1024-point FFT with 64-QAM is used in the simulations with the transmitter quantization step size kept constant at $q_1 = 0.5\sigma_{\mathbf{X}}$. Fig. 4.5 shows the effect of receiver quantization step size q_2 on the BER for an uniform quantizer without overflow. The step size q_2 is decreased from $0.2\sigma_{\mathbf{Y}_{\mathrm{RX}}}$ to $0.05\sigma_{\mathbf{Y}_{\mathrm{RX}}}$. The curves show the effect of increasing the quantizer resolution. At a BER level of 10^{-8}, increasing q_2 from $0.2\sigma_{\mathbf{Y}_{\mathrm{RX}}}$ to $0.1\sigma_{\mathbf{Y}_{\mathrm{RX}}}$ gives a SNR gain of 4dB,

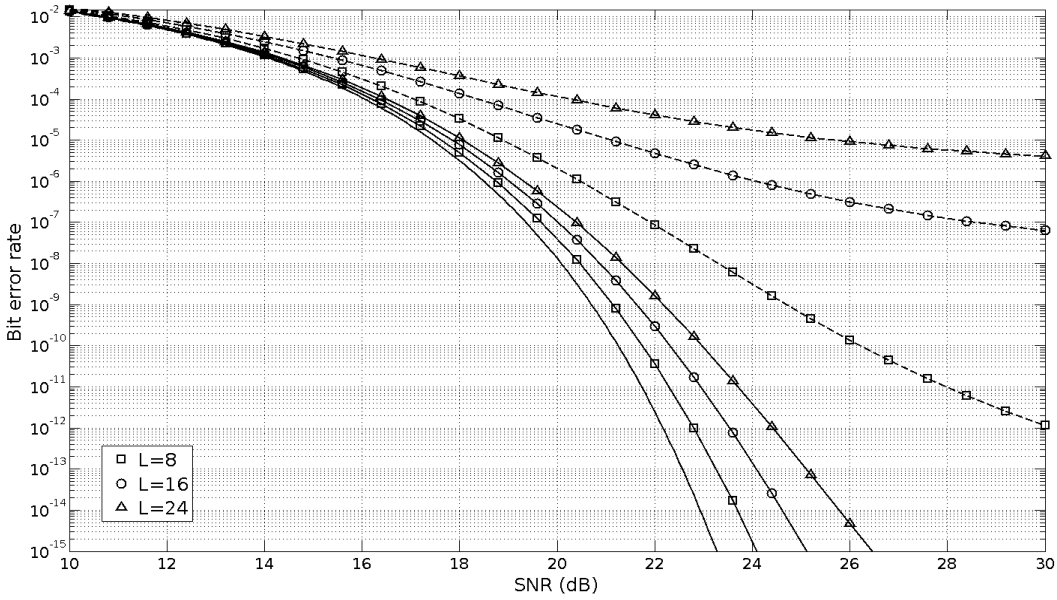

Figure 4.6: BER performance of the uniform quantizer for $q_2 = 0.25\sigma_{\mathbf{X}}$ and $q_2 = 0.1\sigma_{\mathbf{X}}$ for different number of multipaths

while further doubling the resolution to $0.05\sigma_{\mathbf{Y}_{\mathrm{RX}}}$ only improves the SNR by about 0.6dB. The effect of quantizer resolution on BER is thus nonlinear and the figure shows that it also depends on the operating SNR. Therefore, the quantizer resolution can be set to the required receiver performance using the results of the previous section without unnecessarily increasing the resolution. Fig. 4.6 shows the effect of number of channel multipaths on the uniform quantization noise. The BER curves are shown for $q_2 = 0.25\sigma_{\mathbf{Y}_{\mathrm{RX}}}$ (denoted by dashed lines) and $q_2 = 0.1\sigma_{\mathbf{Y}_{\mathrm{RX}}}$ (denoted by solid lines) for values of L=8, 16 and 24. The AWGN only case is shown in solid line without markers for comparison. Results show how the increasing number multipaths has a degrading effect on the BER performance through the increase of $\sigma_{\mathbf{Y}_{\mathrm{RX}}}$ in (4.27). Effect of number of multipaths is higher for the larger quantization step size compared to $q_2 = 0.1\sigma_{\mathbf{Y}_{\mathrm{RX}}}$. For example, at a BER level of 10^{-8}, 0.4dB of SNR is lost when the number of multipaths is increased from 8 to 16 for the step size of $0.1\sigma_{\mathbf{Y}_{\mathrm{RX}}}$ and when it is increased to $0.25\sigma_{\mathbf{Y}_{\mathrm{RX}}}$, an irreducible error floor is introduced

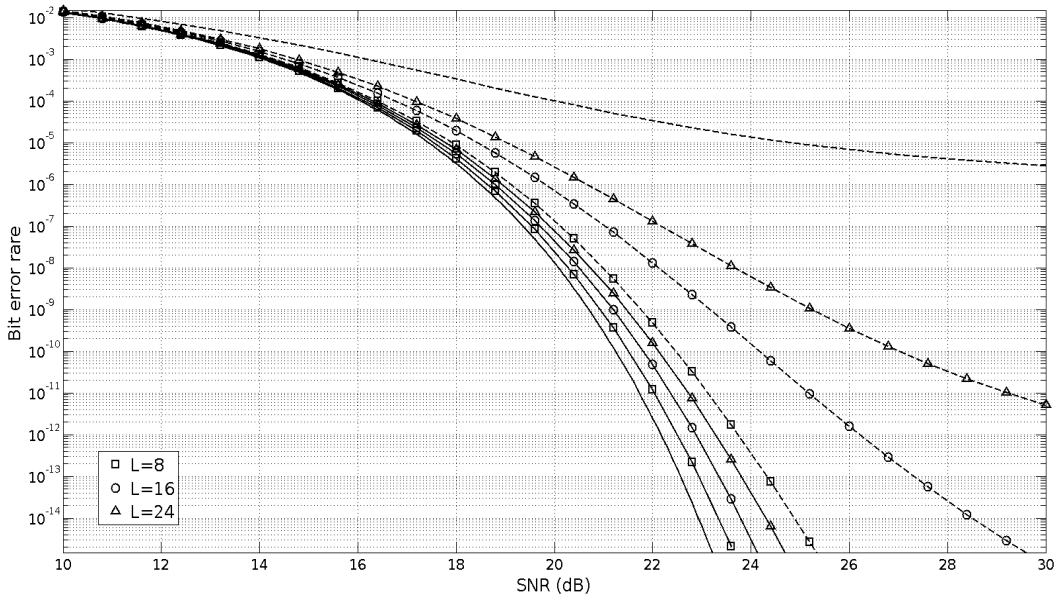

Figure 4.7: BER performance of the floating-point quantizer for b=3 bits and b=4 bits resolutions

and is not even able to achieve the specified BER. Also it is apparent that this multipath effect on BER is higher in the higher SNR regime than at the lower values. Hence, the quantizer resolution can be chosen for the propagation channel and the operating SNR. Finally, Fig. 4.7 shows a similar case as Fig. 4.6 but for a floating-point quantizer at the receiver for different resolutions. Dashed lines show a 3-bit mantissa and a 4-bit mantissa is shown by solid lines. A similar performance pattern as the uniform quantizer is observed here. BER performance degrades with the increasing number of multipaths as expected and this effect is higher for the lower resolution 3-bit mantissa. For the same BER level when number of multipaths is increased from 8 to 16 for the 3-bit mantissa, 1.2dB of SNR is lost. This degradation is reduced to 0.3dB when the quantizer resolution is increased by 1-bit. To depict the BER performance as a function of the mantissa size, the figure also includes the performance of the 2-bit mantissa for L=16 shown in the dashed line without markers. Similar to the previous uniform quantizer, the 2-bit resolution results in an irreducible error

floor. It is also apparent that the BER performance is again nonlinear with the quantizer resolution. Therefore, similar to the uniform quantizer, a better operating resolution for the floating-point quantizer can be chosen for the system parameters.

An increased resolution quantizer can suppress the effect of quantization noise to negligible levels but as discussed earlier, it affects the sampling rate and power consumption, and also the resolution has a direct contribution to the device complexity, heat dissipation, die size, signal bus width, etc., which are important considerations in a small-scale device [53]. It is more critical when these devices employ MIMO systems which require several RF links operating in parallel. With the very close approximations obtained from the theoretical model, the results presented in section 4.1 can be used effectively to find an optimal quantizer resolution for the systems' operational parameters and the required receiver performance.

4.4 Conclusion

Statistics of the signal at various stages of the transmission in an OFDM link and determined how the properties of the transceivers and the channel affect the quantization noise is analyzed. Quantizer resolution affects the receiver BER performance and at the same time is proportional to the power consumption. Therefore, operating the quantizer at an optimal resolution can maintain the required receiver performance without consuming device power unnecessarily. The presented analysis is from a theoretical point of view and various other external factors could alter the statistics of the signal as well. But the detailed process provides an initial estimate of the quantization noise at the receiver for fundamental system parameters, and can be used as lower bound on the quantization noise. This allows for an estimation of the BER at the receiver.

Chapter 5

Conclusion

In this dissertation I looked in two of the main processes involved in an OFDM system: *resource allocation* and *channel estimation*. In the first chapter, an introduction in to OFDM is given, where I discussed the functionality of the main components of an OFDM link. It was followed by an introduction to the problem of resource allocation.

Resource allocation is an important functionality since it is the process which is responsible to provide the required QoS to the end users. Resource allocation techniques need to consider a large number of system and user parameters in the process, and hence an optimal solution is often mathematically intractable. For this reason, resource allocation schemes in OFDMA systems are always sub-optimal. The reason for sub-optimality is not only of the mathematical complexity, it is also due to the amount of calculations that need to be performed for every OFDM frame. Due to the high frequency that the resource allocation algorithm requires to be run, often simple schemes are used in practice. Chapter 2 presented our proposed low-complexity resource allocation scheme. It uses a multi-attribute decision making criterion used in Fuzzy set theory to rank users accordingly. This ranking process is able to determine which users should be given priority in resource allocation. The

weighting process used adds more flexibility to the scheme since it is able to prioritize the attributes. The proposed scheme is able to easily adapt to different system parameters and performance criterion easily by change of attributes and weights.

In chapter 3 discussed the proposed steady-state Kalman filter for channel estimation in OFDM systems in Rayleigh fading channels. Kalman filters are efficient channel estimators such that they are a subset of LMMSE estimators, sequential and requires less second-order statistics. This is highly desirable in mobile situations where the channel is constantly changing the statistical information are not readily available. Although Kalman filters do not require many statistical information, the calculation complexity is still rather high due to a matrix inversion present in the algorithm. Since the channel estimations are required to be performed frequently, this complexity can become prohibitive, especially when the number of subcarriers are large. To alleviate this problem, I identified that for a stationary channel, it is possible to avoid the complex calculations and the iterations by realizing system properties of the OFDM system. Specifically for the IEEE802.16e standard, the proposed method transforms the matrix problem to a scalar problem and then calculations becomes trivial. Not only the number of calculations are substantially reduced, the proposed method is able to increase performance by avoiding the convergence period present in conventional Kalman filters. Extending the scheme a step further, I designed it to utilize the operational SNR and derive the driving noise variance, which is required for the filtering process. This further simplifies the algorithm and at the same time provides an effective method to dynamically change the system to optimal operational status.

Chapter 4 is a theoretical analysis of the quantization noise in an OFDM link. Quantization noise is often disregarded in analyses, but I show that depending on the transmitter/receiver quantizer resolution and the propagating channel, the quantization noise will not be negligible. Quantization noise is not only a function of the resolution but

depends also on the input signal statistics. The statics of the signal is transformed during the transmission from the transmitter to the receiver. Here I study these transformations and determine how they affect the quantization noise at the receiver end. Quantizers consume a substantial amount of power and it is proportional to the quantizer resolution. If the system and channel are known, the theoretical study provides the ability to find the optimum quantizer resolution for a required receiver performance. The simulation results show that the derived theoretical model agrees strongly with a simulated scenario. As Kalman filtering requires noise statistics, this analysis provides a method to determine the quantization noise in the receiver and take it in to account in the channel estimation process.

As concluding remarks, the future work of this line of research will be a cross-layered approach to resource allocation. The resource allocation methods discussed in this report do not consider the factors such as the packet delay and broadcast services. These factors are going to play a considerable role in the next-generation traffic and sustainable resource allocation schemes need to be proposed. The demand for throughput will keep increasing while the wireless access will not be able to keep up and become a bottleneck. Therefore, new trends of user traffic needs to be identified and incorporated in to the resource allocation schemes.

Bibliography

[1] S. Ortiz Jr., "4G Wireless Begins to Take Shape," Computer, vol. 40, no. 11, pp. 18-21, Oct. 2007, doi:10.1109/MC.2007.369.

[2] W. Mohr and W. Konhuser, "Access Network Evolution Beyond Third Generation Mobile Communications," IEEE Communications Magazine, Vol. 38, No. 12, pp. 122-133, 2000.

[3] E. Halepovic, C. Williamson, M. Ghaderi, "Wireless Data Traffic: A Decade of Change," IEEE Network, Vol. 23, No. 2, pp. 20-26, March 2009.

[4] J. Wang, Broadband Wireless Communications - 3G, 4G and Wireless LAN, Springer, Jul. 2001.

[5] Cellular and Mobile Broadband Access for the 21st Century, International Mobile Telecommunications (IMT). (http://www.itu.int/ITU-D/imt-2000/DocumentsIMT2000/IMT-2000.pdf)

[6] J. Gonzalvez; , "Moving Beyond 3G [Mobile Radio]," Vehicular Technology Magazine, IEEE , vol.1, no.2, pp.48-55, 2006.

[7] J. Bergman, M. Ericson, D. Gerstenberger, B. Gransson, J. Peisa, S. Wager, "HSPA Evolution Boosting the performance of mobile broadband access," Ericsson Review, no.1, 2008.

[8] S. Y. Hui, K. H. Yeung, "Challenges in the migration to 4G mobile systems," IEEE Communications Magazine, vol. 41, no. 12, pp. 54-59, Dec. 2003.

[9] J. Choi, Adaptive and Iterative Signal Processing in Communications, Cambridge University Press, Dec. 2006.

[10] R. Prasad, OFDM for Wireless Communications Systems, ArchTech House 2004.

[11] G. Li, H. Liu, "Dynamic resource allocation with finite buffer constraint in broadband OFDMA networks," IEEE Wireless Communications and Networking Conference 2003, Vol.2, pp.1037-1042, March 2003.

[12] W. C. Y. Wong, R. S. Cheng, K. B. Letaif, R. D. Murch, "Multiuser OFDM with adaptive subcarrier, bit, and power allocation", IEEE J. Selec. Areas Comm., Vol 17, pp 1747-1758, Oct. 1999.

[13] P.S. Chow, J.M. Cioffi, J.A.C.Bingham, "A practical discrete multitone transceiver loading algorithm for data transmission over spectrally shaped channels," IEEE Transactions on Communications, vol.43, no.234, pp.773-775, Feb/Mar/Apr 1995.

[14] L. Kaiming, Y. Feifei, W. Wenbo, L. Yuan'an, "An efficient greedy loading algorithm for adaptive modulation in OFDM systems," Personal, Indoor and Mobile Radio Communications, 2005. PIMRC 2005. IEEE 16th International Symposium on , vol.2, no., pp.1130-1134 Vol. 2, 11-14 Sept. 2005.

[15] "A greedy-based resource allocation algorithm for multicast and unicast services in OFDM system," International Conference on Wireless Communications and Signal Processing, WCSP 2009, pp.1-5, 13-15 Nov. 2009.

[16] M. K. Ozdemir and H. Arslan, "Channel estimation for wireless OFDM systems, "IEEE Communication Surveys, Vol. 2, No. 2, pp 18-48, Secong quarter 2007.

[17] C. Eklund, R. B. Marks, S. Ponnuswamy, K. L. Stanwood and N. J. M. V. Waes, WirelessMAN: Inside the IEEE802.16 Standard for Wireless Metropolitan Area Networks, IEEE Standards Information Network/IEEE Press, 20006.

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

[19] Y. Shen, E. Martinez, "Channel Estimation in OFDM Systems," Freescale Semiconductor, AN3059, rev.0, Jan. 2006.

[20] O. Edfors, M. Sandell, J.-J. van de Beek, S.K. Wilson, P.O. Borjesson, "OFDM channel estimation by singular value decomposition," , IEEE Transactions on Communications, vol.46, no.7, pp.931-939, Jul. 1998.

[21] S. G. Kang, Y. M. Ha, E. K. Joo, "A comparative investigation on channel estimation algorithms for OFDM in mobile communications, "IEEE Trans. on Broadcasting, Vol. 49, No. 2, pp 142-149, Jun. 2003. I

[22] M. Ergen, S. Coleri, P Varaiya, "QoS aware adaptive resource allocation techniques for fair scheduling in OFDMA based broadband wireless access systems," IEEE Trans. Broadcasting, Vol 49, pp 362-370, Dec. 2003.

[23] H. Yaghoobi, "Scalable OFDMA physical layer in IEEE 802.16 WirelessMAN", Intel Technology Journal, Vol 8, Issue 3, pp 201-212, 2004.

[24] H.-J. Zimmermann, Fuzzy Set Theory- and Its Applications: Second, Revised Edition, Kluwer Academic Publishers 1991.

[25] R.R.Yager, "Fuzzy decision making using unequal objectives", Fuzzy Sets and Systems 1, pp 87-95, 1978.

[26] J. Ma, Q. Zhang, Z. Fan, J. Liang, D Zhou, "An approach to multiple attribute decision making based on preference information on alternatives", Proc. 34th Hawaii Int. Conf. System Sciences, pp 1-7, Jan. 2001.

[27] D. Kivanc, G. Li, H Liu, "Computationally efficient bandwidth allocations and power control for OFDMA," IEEE Trans. Wireless Comm., Vol 2, pp 1150-1158, Nov. 2003.

[28] Mobile WiMax - Part 1: A technical overview and performance evaluations, August 2006, WiMax Forum.

[29] H. Cheon, D. Hong, "Effect of channel estimation error in OFDM-based WLAN, "IEEE Communications letters, Vol. 6, No. 5, pp 190-192. May 2002.

[30] S. M. Kay, Fundamentals of Statistical Processing, Volume I: Estimation Theory, Prentice Hall PTR, 1993.

[31] Z. Yuanjin, "A novel channel estimation and tracking method for wireless OFDM systems based on pilots and Kalman filtering, "IEEE Trans. on Consumer Electronics, Vol. 49, No. 2, pp 275-283, May 2003.

[32] K-Y. Han, S-W. Lee, J-S. Lim and K-Mo Sung, "Channel estimation for OFDM with fast fading channels by modified Kalman filter, "IEEE Trans. on Consumer Electronics, Vol. 50, No. 2, pp 443-449, May 2004.

[33] L. Kleeman, "Understanding and applying Kalman filtering, "Second Workshop on "Perceptive Systems ", Curtin University of Technology, Australia, Jan. 1996.

[34] D. Vaughan, "A nonrecursive algebraic solution for the discrete Riccati equation, "IEEE Trans. on Automatic Control, Vol. 15, No. 5, pp 597-599, Oct. 1970.

[35] A. J. Laub, "A Schur method for solving algebraic Riccati equations, "IEEE Trans. on Automatic Control, Vol. AC-24, No. 6, Dec. 1979.

[36] R. Leland, "An alternate calculation of the discrete-time Kalman filter gain and Riccati equation solution, "IEEE Trans. on Automatic Control, Vol. 41, No. 12, Dec. 1996.

[37] Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems, IEEE Standard for Local and metropolitan area networks.

[38] J. G. Andrews, A. Ghosh and R. Muhamed, Fundamentals of WiMAX: Understanding Broadband Wireless Networking, Prentice Hall PTR, 2007.

[39] T-D. Chiueh, P-Y. Tsai, OFDM Baseband Receiver Design for Wireless Communications, Wiley, Dec. 2007.

[40] J. G. Proakis, D. K. Manolakis, Digital Signal Processing (4th Edition), Prentice Hall, Apr. 2006.

[41] B. Widrow, I. Kollar, Quantization Noise: Roundoff Error in Digital Computation, Signal Processing, Control and Communications, Cambridge University Press, Jul 2008.

[42] D. Dardari, "Joint clip and quantization effects characterization in OFDM receivers, "IEEE Trans. on Circuits and Systems I:, Vol. 53, No. 8, pp 1741-1748, Aug. 2006.

[43] H. Ehm, S. Winter, R. Weigel, "Analytic quantization modeling of OFDM signals using normal Gaussian distribution, "Asia-Pacific Microwave Conference 2006, pp 847-850, Dec. 2006.

[44] D. J. G. Mestdagh, "Calculation of ADC resolution for DMT modulation, "Electronics Letters, Vol. 31, No. 16, pp 1315 - 1316, Aug. 1995.

[45] M. Casamayor, C. Croke, "How to Save Power in Battery Applications Using the Power-Down Mode in an ADC. "Analog Devices, Vol.37, Sep. 2003.

[46] J. Thomson, et al., "An integrated 802.11 a baseband and MAC processor, "in 2002 IEEE International Solid-State Circuits Conference (ISSCC), Vol. 2, Feb. 2002.

[47] J. Arias, V. Boccuzzi, L. Quintanilla, L. Enriquez, D. Bisbal, M. Banu, J. Barbolla, "Low-power pipeline ADC for wireless LANs, "IEEE Jour. of Solid-State Circuits, Vol. 39, No. 8, pp 1338 - 1340, Aug. 2004.

[48] R.H. Walden, "Analog-to-digital converter survey and analysis, "IEEE Jour. on Selected Areas in Communications, Vol. 17, No.4, pp 539 - 550, April 1999.

[49] A. Leon-Garcia, Probability and Random Processes for Electrical Engineering (Second edition), Addison-Wesley publishing, May 1994.

[50] S. M. Kay, Intuitive Probability and Random Processes using MATLAB, Springer, Nov. 2005.

[51] F. P. Fontan, P. M. Espineira, Modeling the Wireless Propagation Channel: A simulation approach with Matlab, Wiley, Sep. 2008.

[52] M. K. Simon, Probability Distributions Involving Gaussian Random Variables: A Handbook for Engineers and Scientists, Springer, May. 2002.

[53] B. M. Murray, H. Suzuki, S. Reisenfeld, "Optimal quantization of OFDM at digital IF, "IEEE Region 10 Conference TENCON 2008, pp 1-5, Nov. 2008.