A novel OFDM Blind Equalizer: Analysis and Implementation

David E. Gonzalez Fitch

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Master of Science
in
Electrical Engineering

Tamal Bose, Chair
Ashwin E. Amanna
Timothy Pratt

July 13, 2012
Blacksburg, Virginia

Keywords: OFDM, blind channel estimation, blind equalization, BER prediction, link adaptation

A novel OFDM Blind Equalizer: Analysis and Implementation

David E. Gonzalez Fitch

(ABSTRACT)

Link adaptation is important to guarantee robust and reliable wireless communications without wasting valuable radio resources. This technique has become more feasible with the recent appearance of Software Defined Radios (SDRs), which allow easy reconfiguration of their parameters via software. As the environment changes over time, the transmitter needs to be able to effectively estimate its performance under different radio input parameters to be able to find a close to optimal solution. In most wireless communications, an equalizer is implemented at the receiver to estimate the channel impulse response. This estimate can be fed back to the transmitter via a feedback channel, which can in turn help generate a sub-optimal transmission solution for the current situation.

In this thesis, a link adaptation method is proposed that uses Orthogonal Frequency-Division Multiplexing (OFDM) in conjunction with blind channel estimation. With the use of OFDM, it can be assumed that the frequency fading at each subcarrier is approximately flat. In addition, under the assumption that the channel is quasi-stationary, the Bit Error Rate (BER) at each subcarrier can be estimated by using the well-known BER formulas for an Additive White Gaussian Noise (AWGN) channel. However, the effect of imperfect channel estimation must also be taken into account.

A novel OFDM blind channel estimator is developed. Finally, both simulations and real over-the-air results are presented.

This work received support from the Federal Rail Association through Grant # 1234.

Acknowledgments

I would like to acknowledge my thesis committee for dedicating so much of their time and energy in guiding me through my research: to Dr. Tamal Bose for the many meetings that were invaluable in steering me in the right direction, and for helpful advice at critical times when I needed to find a way forward; and to Dr. Ashwin Amanna and Dr. Timothy Pratt for serving on my committee. Ashwin also provided me with the radios and laptops to run my tests, and was an outstanding supervisor in all respects.

I would also like to thank Dr. Joe Gaeddert for his work in wireless communications and for his development and support of liquid. Joe dedicated far too much of his time explaining how liquid works and how to modify it, and was always willing to answer my questions on wireless communications. My sincerest thanks for his support and guidance.

Thanks are also due to all those who worked on the FRA project, especially Daniel Ali, whose patient willingness to help me with myriad programming problems proved invaluable.

Finally, I would like to acknowledge everyone who works in the MPRG lab, and the Bradley Department of Electrical and Computer Engineering in general, who made all of this possible.

All photos by author, 2012

Dedication

This thesis is dedicated to my parents, who made it possible for me to come this far.

List of Abbreviations

8-PSK 8-Phase Shift Keying

16-QAM 16-Quadrature Amplitude Modulation

32-SQAM 32-Square Quadrature Amplitude Modulation

64-QAM 64-Quadrature Amplitude Modulation

arb. units arbitrary units

AWGN Additive White Gaussian Noise

BER Bit Error Rate

BPSK Binary Phase Shift Keying

BW Bandwidth

CBR Case Based Reasoner

CE Cognitive Engine

CIC Cascaded Intergrator-Comb

CIR Channel Impulse Response

CP Cyclic Prefix

CR Cognitive Radio

CSI Channel State Information

DFT Discrete Fourier Transform

FIR Finite Impulse Response

FPGA Field-Programmable Gate Array

GA Genetic Algorithm

ICI Inter-Carrier Interference

IDFT Inverse Discrete Fourier Transform

ISI Inter-Symbol Interference

ISM Industrial, Scientific and Medical

LOS Line of Sight

MAP Maximum A Posteriori

ML Maximum Likelihood

MLSD Maximum Likelihood Sequence Detection

MMSE Minimum Mean Square Error

M-PSK M-Phase Shift Keying

MSE Mean Square Error

NMSE Normalized Mean Square Error

OC One-Cycle

OFDM Orthogonal Frequency Division Multiplexing

QAM Quadrature Amplitude Modulation

QPSK Quadrature Phase Shift Keying

RMSE Root Mean Square Error

SDR Software Defined Radio

SER Symbol Error Rate

SINR Signal-to-Noise-and-Interference Ratio

SIR Signal-to-Interference Ratio

SNR Signal-to-Noise Ratio

TC Two-Cycle

USB Universal Serial Bus

USRP1 Universal Software Radio Peripheral 1

USRP2 Universal Software Radio Peripheral 2

Contents

List of Abbreviations **v**

1 Introduction **1**

1.1 Motivation 1

1.2 Background on link adaptation 1

1.3 Contributions 2

2 OFDM Blind Equalization **4**

2.1 Introduction to multipath fading 4

2.2 Introduction to OFDM 7

2.3 General equalization techniques 11

2.4 Overview of OFDM specific blind channel estimation 12

3 OFDM blind equalizer **19**

3.1 Algorithm 19

3.2 Simulation results 25

4 System design **28**

4.1 System model implementation 28

4.1.1 Description of *liquid* 31

4.1.2 Modifications to *liquid* 34

4.2 Performance estimation 36

4.2.1 BER estimation for an uncoded transmission 36

4.2.2 BER estimation for a FEC coded transmission 37
4.2.3 BER estimation with imperfect channel estimate 39

5 Over-the-air tests **42**
5.1 Issues to be considered for over-the-air transmissions 42
5.2 System setup 45
5.3 Results 47

6 Conclusions and future work **56**

References **58**

A Simulation plots **62**
A.1 Received signal constellations 62
A.2 Channel estimates 69

List of Figures

2.1 Illustration of a multipath channel example 5

2.2 Multipath channel example 6

2.3 OFDM subbands 7

2.4 OFDM adjacent subcarriers 8

2.5 Baseband OFDM system model, where DFT is the discrete Fourier transform, IDFT is the inverse discrete Fourier transform, CP is the cyclic prefix, P/S means parallel to serial and S/P means serial to parallel 9

3.1 Signal constellation before and after precoding for BPSK modulation 20

3.2 Signal constellation before and after precoding for QPSK modulation 20

3.3 Signal constellation before and after precoding for 8-PSK modulation 21

3.4 Signal constellation before and after precoding for 16-QAM modulation . . . 21

3.5 Signal constellation before and after precoding for 32-SQAM modulation . . 22

3.6 Signal constellation before and after precoding for 64-QAM modulation . . . 22

3.7 Comparison of the resulting time signal amplitudes after precoding 24

3.8 Normalized mean square error (NMSE) of the channel estimate vs. SNR . . 26

3.9 BER for a random channel of length $L = 3$ 27

3.10 BER for a random channel of length $L = 6$ 27

4.1 Transmitter node (on the right) and receiver node (on the left) 29

4.2 Antenna used on the transmitter and receiver (VERT900) 29

4.3 Transmitter daughter card (RFX900) 30

4.4 Receiver daughter card (WBX) 30

4.5 Packet structure, where S_0 and S_1 are the two different types of preamble symbols, H_m are the header symbols and P_m are the data symbols 32

4.6 Phase rotation ($\Delta\varphi(k)$) due to residual carrier frequency/phase offsets ($\Delta\omega$ and $\Delta\theta$) and timing offset (Δt), and i and k are the time and frequency indexes respectively 33

4.7 System model, where FEC means forward error correcting, $Q_{i,k}$ and A are the precoder parameters, S/P means serial to parallel, IDFT is the inverse discrete Fourier transform, CP is the cyclic prefix, J is the smoothing factor, S_0 and S_1 are the preamble symbols, ADC stands for analog to digital converter, CIR is the channel impulse response, AWGN stands for additive white Gaussian noise, DAC means digital to analog converter and IDFT is the inverse discrete Fourier transform 35

4.8 Measured and estimated BER for the six tested modulation schemes 40

4.9 Error between the estimated and measured BER values for the six tested modulation schemes 40

5.1 General shape of the variation of the error in the estimation of the carrier frequency offset ($\varepsilon(\Delta\omega)$) with the carrier frequency offset ($\Delta\omega$) 43

5.2 Typical example of the general shape of the channel response gain in the frequency domain (ω) caused by the CIC filter of the FPGA, which is responsible for "drooping" effect on both ends of the allocated bandwidth, where BW stands for the 3 dB bandwidth 44

5.3 BER for a signal that is clipped at higher SNR values (for 64-QAM) 45

5.4 Subcarrier allocation automatically asigned by *liquid* for a system with 64 subcarriers, which is independent of the center frequency or bandwidth. There are 44 data subcarriers, 6 pilot subcarriers and 14 null subcarriers 46

5.5 Channel estimate 47

5.6 Received constellation for BPSK 49

5.7 Received constellation for QPSK 49

5.8 Received constellation for 8-PSK 50

5.9 Received constellation for 16-QAM 50

5.10 Received constellation for 32-SQAM 51

5.11 Received constellation for 64-QAM 51

5.12 Measured and estimated BER for uncoded data, for the six tested modulation schemes and $J = 100$ 53

5.13 Error between the estimated and measured BER for uncoded data, for the six tested modulation schemes and $J = 100$ 53

5.14 Measured and estimated BER for uncoded data, for the six tested modulation schemes and $J = 50$ 54

5.15 Error between the estimated and measured BER for uncoded data, for the six tested modulation schemes and $J = 50$ 54

5.16 Measured and estimated BER for data that was encoded with Hamming (7,4), for the six tested modulation schemes and $J = 100$ 55

5.17 Error between the estimated and measured BER for data that was encoded with Hamming (7,4), for the six tested modulation schemes and $J = 100$ . . 55

A.1 Received signal constellation before and after equalizing and de-precoding for BPSK modulation for a channel generated randomly of length $L = 3$ 63

A.2 Received signal constellation before and after equalizing and de-precoding for QPSK modulation for a channel generated randomly of length $L = 3$ 63

A.3 Received signal constellation before and after equalizing and de-precoding for 8-PSK modulation for a channel generated randomly of length $L = 3$ 64

A.4 Received signal constellation before and after equalizing and de-precoding for 16-QAM modulation for a channel generated randomly of length $L = 3$ . . . 64

A.5 Received signal constellation before and after equalizing and de-precoding for 32-SQAM modulation for a channel generated randomly of length $L = 3$ . . 65

A.6 Received signal constellation before and after equalizing and de-precoding for 64-QAM modulation for a channel generated randomly of length $L = 3$ . . . 65

A.7 Received signal constellation before and after equalizing and de-precoding for BPSK modulation for a channel generated randomly of length $L = 6$ 66

A.8 Received signal constellation before and after equalizing and de-precoding for QPSK modulation for a channel generated randomly of length $L = 6$ 66

A.9 Received signal constellation before and after equalizing and de-precoding for 8-PSK modulation for a channel generated randomly of length $L = 6$ 67

A.10 Received signal constellation before and after equalizing and de-precoding for 16-QAM modulation for a channel generated randomly of length $L = 6$ . . . 67

A.11 Received signal constellation before and after equalizing and de-precoding for 32-SQAM modulation for a channel generated randomly of length $L = 6$ . . 68

A.12 Received signal constellation before and after equalizing and de-precoding for 64-QAM modulation for a channel generated randomly of length $L = 6$ . . . 68

A.13 Amplitude and phase of an estimated channel of length $L = 3$ and smoothing factor $J = 100$ 69

A.14 Amplitude and phase of an estimated channel of length $L = 6$ and smoothing factor $J = 100$ 70

A.15 Amplitude and phase of an estimated channel of length $L = 3$ and smoothing factor $J = 200$ 70

A.16 Amplitude and phase of an estimated channel of length $L = 6$ and smoothing factor $J = 200$ 71

List of Tables

2.1 Referenced OFDM blind channel estimator comparison 18

5.1 System parameters 48

Chapter 1

Introduction

1.1 Motivation

The region of the electromagnetic frequency spectrum used for radio communications is becoming a scarce and ever more valuable resource. An increasing number of technologies are relying on wireless communications, leading to saturation of the radio frequency spectrum, thus it is essential to use it more efficiently. In this context, link adaptation is an innovative technique that provides robust and reliable wireless transmissions while not wasting valuable radio resources. This has become more feasible with the recent appearance of software defined radios (SDR), which allow easy reconfiguration of radio parameters via software. As the environment changes over time, the transmitter needs to have the capability to effectively estimate performance under different radio input parameters so it can find a close to optimal solution. The main objective of this thesis was to develop a fast and effective method of estimating the performance of these wireless communication systems for link adaptation.

1.2 Background on link adaptation

Link adaptation refers to the different techniques that permit the adjustment of radio parameters, such as modulation or coding rate, to adapt to the changing channel conditions.

The majority of link adaptation techniques modify only one or two parameters, and they do so by trial-and-error procedures such as a backoff algorithm. An overview of different methods designed to adapt either data rate or packet size can be found in section 2.2 of Ref. [1]. Most data rate adaptation techniques rely on some form of success rate or failure rate - a statistical approach to determine whether the data rate should be changed. Other techniques use a measured signal-to-noise ratio (SNR) approach or a hybrid mechanism between these two approaches. The methods proposed for packet size adjusting are based

on either determination of the success rate or estimating the bit error rate (BER), using a Kalman filter.

A method of adapting both modulation and power is also presented in Ref. [1]. It consists of a simple algorithm where, if the target BER is not achieved, then the modulation order is progressively reduced. If the minimum modulation order still does not enable the BER to reach its desired target, then power is adjusted. The decision of how much to increase the power is made by estimating the interference power and the signal-to-interference-plus-noise (SINR), after which the power is increased to reach a target SINR. An overview of other techniques to adapt modulation and power is also given in Ref. [1]. A Rayleigh block-fading channel is estimated by using pilots and adapts the modulation scheme to maximize the throughput for a target BER [2]. In order to do so, it is necessary to use a training phase and a data phase.

Some link adaptation methods allow more than two parameters to be changed. In Das *et al.* different techniques are compared to adapt power, modulation, and coding, as well as adaptation interval, sub-band size and choice of bit-power loading algorithm [3]. These methods also require channel state information (CSI) and use iterative proceedures to perform the link adaptation.

Optimization of multiple radio parameters has also been discussed by Rondeau [4]. The optimization is achieved with the help of a cognitive engine (CE), which uses a hybrid architecture combining a genetic algorithm (GA) and a case base reasoner (CBR), in the context of cognitive radios (CR).

A different type of CE randomly tries different parameter configurations, and their resulting performances are stored in a CBR [5, 6, 7, 8]. As good solutions are found, the CE balances what is known as "exploration vs exploitation", which is the ratio to which the system keeps looking for a better solution while at the same time using one that is known to work.

Following the same hybrid arquitecture proposed by Rondeau (GA plus CBR), further development has been done in Ref. [9]. In our case, performance estimations were based on additive white Gaussian noise (AWGN) formulas, which will lead to overly optimistic solutions unless these formulas are adjusted to take into account other channel impairments.

1.3 Contributions

The main contributions of this thesis are in two areas related to performance evaluation and adaptation of wireless radio systems: a) blind channnel estimation and b) accurate BER estimation for orthogonal frequency division multiplexing (OFDM).

Most blind methods find an estimate of the channel up to a complex constant. In such cases, it is common to resolve this ambiguity by using one subcarrier as a pilot. A new blind channel estimator, based on the results of Petropolu *et al.* [10] is presented here, that

solves this issue. This new estimator finds the channel estimate without any ambiguity, and therefore does not need a pilot, making the algorithm truly blind. The algorithm requires previous knowledge of the channel's impulse response length.

The second contribution of the presented work is in the development of an analytical model that is derived from the estimated channel information. This model is then used to predict the BER. This BER prediction could be used by different link adaptation techniques to adapt the radio's transmission parameters.

Chapter 2

OFDM Blind Equalization

2.1 Introduction to multipath fading

In a typical radio environment, multiple objects can block or reflect the transmitted electromagnetic signal. Consequently, the signal that arrives at the receiver is a superposition of multiple waves that have passed through (shadowing) or were reflected, diffracted or scattered by different obstacles, coming in from multiple directions after following completely different paths [11, 12]. Each received wave, or multipath component, will have a different delay, attenuation and phase shift. The delay depends on the path length and the speed at which it propagates in the corresponding media; while the attenuation and phase shift is due to free space propagation and also depends on the path length, as well as on the objects the rays encountered along the way.

The effect of the multipath components will depend on the incident phase of each wave, where their superposition can be constructive or destructive. For instance, if a signal has two multipath components which arrive with a phase difference of 180 °, they will negate each other, whereas if they have the same phase, the signal will add constructively. Furthermore, receiving multiple copies of the information-bearing signal at different instances will cause time dispersion.

Generally, two types of multipath channels can be distinguished, depending on whether or not there is a "line of sight" (LOS). A LOS path goes directly from the transmitter to the receiver without any obstacles. This component is usually the least attenuated one and has the shortest delay. In the case of no-LOS, all multipath components will have been reflected or shadowed once or multiple times. In this second scenario, there is rarely a component that would dominate the rest as the LOS would do, therefore the no-LOS scenario is typically a harsher environment. An example of a simple multipath channel is depicted in Fig. 2.1, where there are four paths. There is a LOS path following a straight line that goes directly from the transmitter (on the left side of the figure) to the receiver (on the right side) and

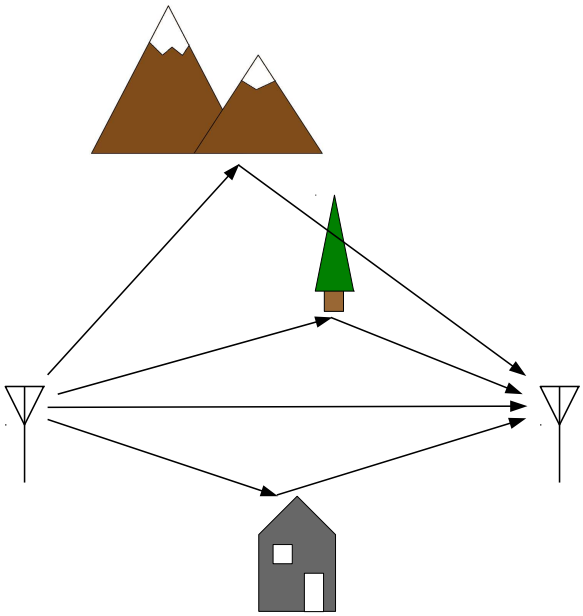


Figure 2.1: Illustration of a multipath channel example

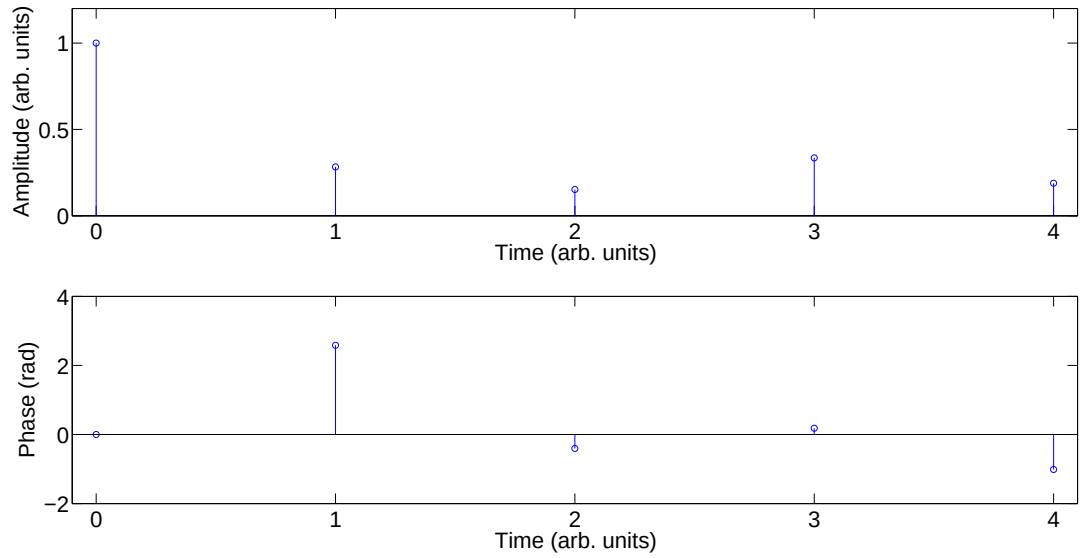
three paths along which the signal experiences reflection in a mountain, a tree and a house. Additionally, one reflected signal is also shadowed by one of the other obstacles.

The impulse response of a time-invariant multipath channel that has L paths can be modelled as follows [11]:

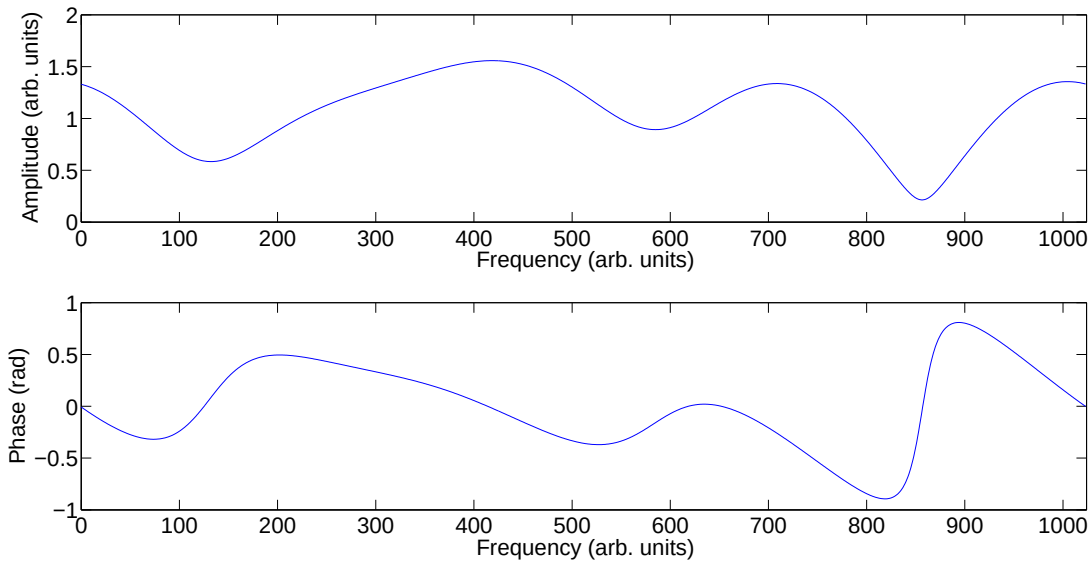
$$h(t) = \sum_{l=0}^{L-1} a_l e^{j\theta_l} \delta(t - \tau_l) \tag{2.1}$$

where l is the path index, a_l is the path gain, θ_l is the path phase shift and τ_l is the path delay. An example of a multipath channel is shown in Fig. 2.2, where 2.2(a) represents the channel impulse response and 2.2(b) is the channel frequency response, for a channel of length $L = 5$, where $h(t) = 1\delta(t) + (-0.24 + 0.15j)\delta(t-1) + (0.14 - 0.06j)\delta(t-2) + (0.33 + 0.06j)\delta(t-3) + (0.10 - 0.16j)\delta(t-4)$. The magnitudes of voltage amplitude, time and frequency are given in arbitrary units (arb. units). It can be seen that the response of the channel is not the same at every frequency, making it a frequency-selective channel. The sections of the frequency response that have the lowest gain are referred to as fades, where the signal is attenuated the most.

The time delay between the first and last multipath components is defined as the delay spread $\tau = \tau_{L-1} - \tau_0$, which is constant for static channels. For quasi-static channels or



(a) Multipath channel impulse response



(b) Multipath channel frequency response

Figure 2.2: Multipath channel example

other types of time-variant channels, the root mean square delay spread τ_{RMS} is defined, and its exact mathematical definition can be found in Refs. [11] and [12].

Frequency-selective fading can be characterized by the *coherence bandwidth* B_c, which is the average bandwidth over which the channel in the frequency domain is correlated. It is defined as the frequency difference for which the channel's autocorrelation function has decreased by 3 dB. The ratio of the coherence bandwidth and the bandwidth of the information-bearing signal (BW) determines how the fading affects the communication. If $BW << B_c$, then the channel response can be considered constant over the information signal bandwidth, therefore the fading is flat. If not, the signal will experience frequency selective-fading.

2.2 Introduction to OFDM

The conventional way of transmitting information over a wireless link was to serially send data by modulating the signal over a single carrier frequency at a fixed baud rate. However, in a multipath fading channel, the receiver will get different copies of the transmitted symbols via different paths, causing them to spread into the subsequent symbols. This effect is known as inter-symbol interference (ISI). A complex equalizer is then needed to compensate for the channel distortion [11].

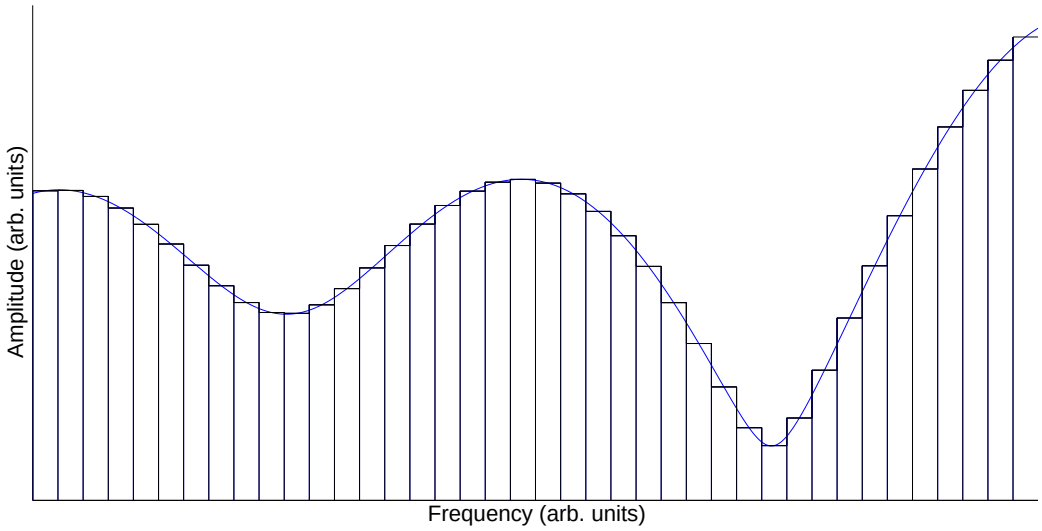


Figure 2.3: OFDM subbands

Another approach, known as multicarrier modulation, is to divide the available bandwidth into different subbands. Then, the data is modulated onto each subband with the corresponding subcarrier. This process effectively converts one fast serial data stream into mul-

tiple slow serial ones. If the bandwidth of each subband is sufficiently narrow (significantly smaller than the coherence bandwidth defined in the previous section), then the channel can be considered flat at each subcarrier, as shown in Fig. 2.3. However, to ensure that subbands do not interfere with each other, guard bands need to be inserted, which diminishes spectral efficiency.

To achieve high spectral efficiency, OFDM takes multicarrier modulation one step further. For each symbol, the waveform of the subcarrier is limited to a time window whose duration is the inverse of the baud rate. This windowing results in a convolution with a sinc function in the frequency domain, which will lead to a waveform that has zero-crossings at fixed intervals. Therefore, in OFDM the subbands are spaced in such a way that the peak of each subcarrier overlaps with the nulls of the rest, as shown in Fig. 2.4. If each subcarrier is sampled at its peak, there will be no interference from the adjacent ones. Not limiting the spectrum of each subcarrier in the frequency-domain reduces the need for more complex filters, and it allows the symbols to be limited in time [12].

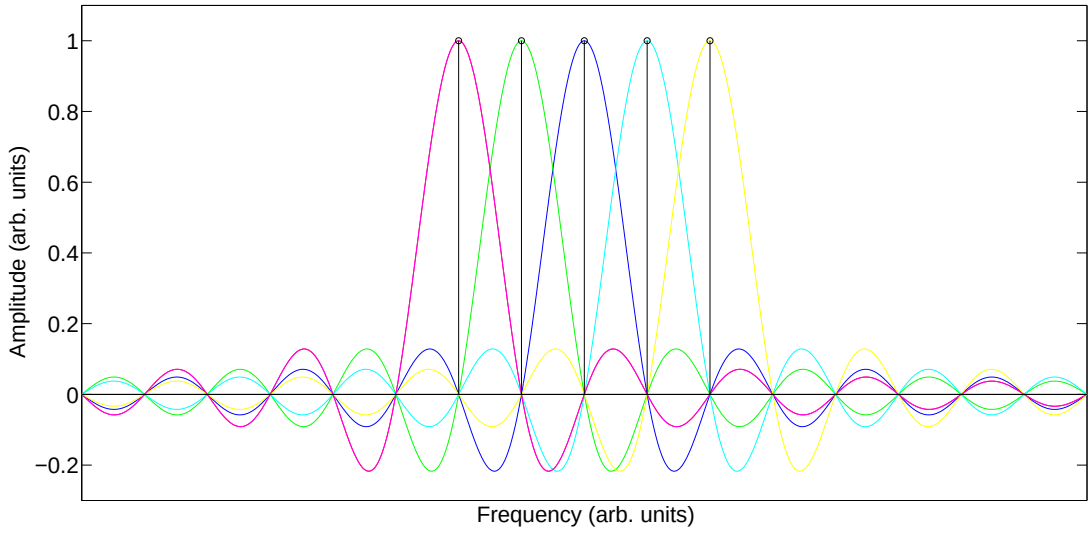


Figure 2.4: OFDM adjacent subcarriers

In an OFDM system, a time-dispersive channel will cause ISI as well as inter-carrier interference (ICI). To overcome these issues, the last part of the OFDM symbol is copied and added to the front of the symbol, resulting in what is called a cyclic prefix (CP). The CP's length must be chosen such that it is larger than the maximum delay spread of the channel. The insertion of the CP acts as guard interval, protecting the symbol from ISI. In addition, it converts the linear convolution of the signal with the channel impulse response (CIR) into a circular convolution, which is the equivalent of a multiplication in the frequency domain, so the subcarriers remain orthogonal, thus preventing ICI [11].

To implement an OFDM system, the symbols to be transmitted on each subcarrier are generated in the frequency domain. An inverse discrete Fourier transform (IDFT) operation then yields the signal in the time domain, to which the CP is added. The resulting symbol is modulated and sent over the wireless link. On the receiver side, the system first domodulates the signal, removes the CP and then performs the discrete Fourier transform (DFT) operation. Such a system model is represented in Fig. 2.5.

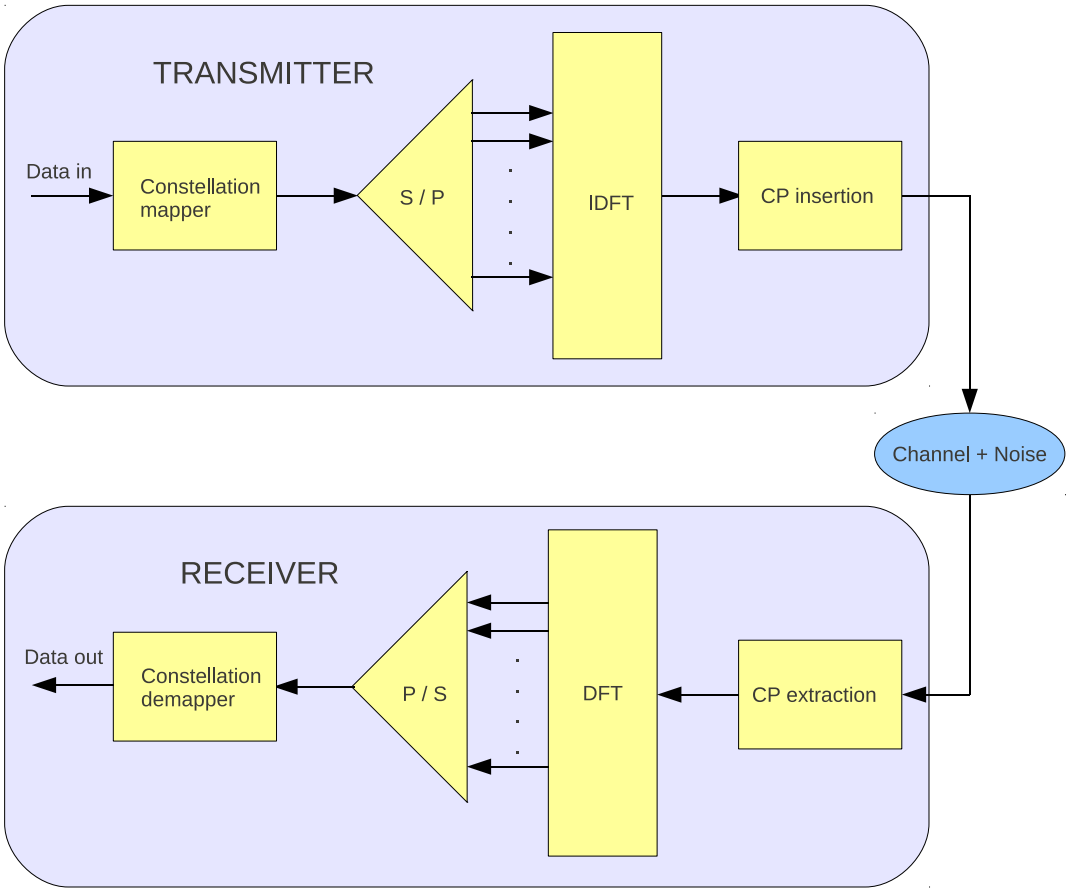


Figure 2.5: Baseband OFDM system model, where DFT is the discrete Fourier transform, IDFT is the inverse discrete Fourier transform, CP is the cyclic prefix, P/S means parallel to serial and S/P means serial to parallel

This model can be formulated mathematically by assuming that a regular discrete time-domain signal $u[n]$ goes through a channel $h[n]$ of length L and Gaussian noise $v[n]$ is added. Thus, the received discrete sequence would be

$$r[n] = \sum_{l=0}^{L-1} u[n-l] + v[n] \tag{2.2}$$

Next, an OFDM transmission is defined that uses N subcarriers and a CP of length $CP > L$. The OFDM symbol in the frequency-domain is represented as $\mathbf{s}[i] = [s[(i-1)N], s[(i-1)N+1], ..., s[(i-1)N+N-1]]^T$, where $s[k]$ are the symbols that are modulated onto each subcarrier and ^{T} is the transpose operation. The i^{th} OFDM symbol in the time-domain after CP insertion is defined as $\mathbf{u}[i] = [u[(i-1)P], u[(i-1)P+1], ..., u[(i-1)P+P-1]]^T$, where P is the length of the OFDM symbol ($P = N+CP$). The vector $\mathbf{u}[i]$ can be generated from $\mathbf{s}[i]$ as follows:

$$\mathbf{u}[i] = \mathbf{T}_{CP}\mathbf{F}_N^H\mathbf{s}[i] \tag{2.3}$$

where $\mathbf{T}_{CP} := \left[\mathbf{I}_{CP}^T \mathbf{I}_N^T\right]^T$ is a CP inserting matrix with $\mathbf{T}_{CP}$ being the last CP rows of the $N \times N$ identity matrix $\mathbf{I}_N$, $\mathbf{F}_N$ is an $N \times N$ DFT matrix, and $\mathbf{F}_N^H$ is an $N \times N$ IDFT matrix obtained by taking the Hermitian of $\mathbf{F}_N$.

According to [12], using 2.2, it can be shown that

$$\mathbf{r}[i] = \mathbf{H}_0\mathbf{u}[i] + \mathbf{H}_1\mathbf{u}[i-1] + \mathbf{v}[i] \tag{2.4}$$

where $\mathbf{r}[i] = [r[(i-1)P], r[(i-1)P+1], ..., r[(i-1)P+P-1]]^T$ and $\mathbf{v}[i] = [v[(i-1)P], v[(i-1)P+1], ..., v[(i-1)P+P-1]]^T$. Since $P > L$, the $P \times P$ channel matrices $\mathbf{H}_0$ and $\mathbf{H}_1$ are given by

$$\mathbf{H}_0 = \begin{pmatrix} h[0] & 0 & 0 & \cdots & 0 \\ \cdots & h[0] & 0 & \cdots & 0 \\ h[L-1] & \cdots & \ddots & \cdots & 0 \\ \vdots & \ddots & \cdots & \ddots & 0 \\ 0 & \cdots & h[L-1] & \ddots & h[0] \end{pmatrix}_{P \times P} \tag{2.5}$$

and

$$\mathbf{H}_1 = \begin{pmatrix} 0 & \cdots & h[L-1] & \cdots & h[1] \\ \vdots & \ddots & 0 & \ddots & \vdots \\ 0 & \cdots & \ddots & \cdots & h[L-1] \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \vdots & 0 \end{pmatrix}_{P \times P} \tag{2.6}$$

It should be emphasized that ISI from the immediately preceding symbol $\mathbf{u}[i-1]$ occurs only because $P > L$.

At the receiver, the operation to remove the CP and convert the signal to the frequency domain can be expressed as:

$$\mathbf{y}[i] = \mathbf{F}_N \mathbf{R}_{CP} \mathbf{r}[i] = \mathbf{F}_N \mathbf{R}_{CP} \mathbf{H}_0 \mathbf{T}_{CP} \mathbf{F}_N^H \mathbf{s}[i] + \mathbf{F}_N \mathbf{R}_{CP} \mathbf{v}[i] \tag{2.7}$$

Here, $\mathbf{R}_{CP} := [\mathbf{0}_{N \times CP} \mathbf{I}_N]$ removes the CP from the received OFDM symbol; the ISI is also removed because the channel matrix $\mathbf{H}_1$ becomes zero since $\mathbf{R}_{CP}$ removes the first CP rows, and $CP > L$. The first factor $\mathbf{H} = \mathbf{F}_N \mathbf{R}_{CP} \mathbf{H}_0 \mathbf{T}_{CP} \mathbf{F}_N^H$ simplifies to a diagonal matrix which, if it exists, can be easily inverted. Since $\mathbf{H}$ is a diagonal matrix, the relationship between the transmitted and received i^{th} symbol modulated onto subcarrier k ($s_{i,k}$ and $y_{i,k}$ respectively) is

$$y_{i,k} = H[k] s_{i,k} + v_{i,k} \tag{2.8}$$

where $H[k]$ is the k^{th} element of the principal diagonal $\mathbf{H}$ and is defined as

$$H[k] = \sum_{n=0}^{L-1} h[n] e^{-j2\pi k \frac{n}{N}} \tag{2.9}$$

which is the DFT of $h[n]$.

2.3 General equalization techniques

Equalization is the process of eliminating the effects of the channel on the received signal. For single carrier communications, there are several techniques that can be used to estimate the channel and perform equalization; detailed descripition of each can be found in [13].

One such technique is known as zero-forcing decision feedback equalization, where past symbol decisions are used to remove ISI from previous symbols. Assuming the detected values are correct, ISI from the past symbols is also avoided. However, it has a potential for catastrophic behavior due to error propagation. Zero-forcing means that the equalizer tries to completely inverse the channel $h(t)$ by finding a $g(t)$ such that $h(t) * g(t) = \delta(t - t_0)$. The zero-forcing process, however, can add the noise components of the other adjacent symbols. Another possibility is to minimize the sum of ISI and noise. This is referred to as the minimum mean-square error (MMSE) strategy.

In linear equalization, a symbol is detected by forming a weighted combination of received values. The idea is to compensate for ISI by subtracting the contributions of the adjacent

symbols, but these value are not known. However, copies of these symbols are present in other received values. By multiplying these received values by certain weights and adding them to the matched filter value, the interference from adjacent symbols can be cancelled. Again, linear equalization can be realized following two strategies: zero-forcing or MMSE. Similar to MMSE, another way to compensate for ISI would be to use the maximum likelihood (ML) estimator.

Maximun likelihood sequence detection (MLSD) involves determining the symbol sequence that best explains the received signal. It provides a way to account for ISI rather than removing it. This method minimizes the sequence error rate. Another possibility would be to minimize the symbol error rate instead. This approach is known as maximum a posteriori (MAP) symbol detection.

2.4 Overview of OFDM specific blind channel estimation

Regular equalization techniques for single carrier communications require complex methods to compensate for the ISI of the adjacent symbols that were transmitted previously, by estimating the multiple channel taps. One of the main advantages of OFDM is that it converts the channel matrix into a circular one. This means that, instead of having multiple taps, each subcarrier now "sees" a one tap channel with a certain gain. Compensating for these channel effects is easy and only requires inversion of the diagonal channel matrix (assuming it exists). But to be able to do so, the channel first needs to be estimated. One way to do this is by inserting pilot symbols. The pilots, which can be added in the time or frequency domain, allow estimation of the channel at the positions of the pilots. At the other locations, the channel can be estimated by interpolation. However, the use of pilot schemes reduces the overall useful throughput. For this reason, blind channel estimators, which do not require the use of pilots, have been investigated. In this section, an overview of OFDM specific blind channel estimators and equalizers is presented.

Blind channel estimators can be categorized as either statistical or deterministic. The former ones require multiple received OFDM symbols to be capable of estimating the CIR, making them only suitable for quasi-static fading channels; while the latter ones can estimate the channel for each received symbol, but require very high computational complexity.
The most common way of statistical blind estimation is the subspace method, where the autocorrelation matrix of the received signal is decomposed into the signal and noise subspaces. The CIR can be estimated because the noise subspace is orthogonal to the signal subspace, which is linearly related to the CIR. An example of such a method is given in Ref. [14]. For OFDM systems without a CP the estimation requires either time oversampling at the receiver or multiple receiver antennas [15], or oversampling of the frequency domain in order to reduce complexity [16].

Some of the statistical techniques exploit the cyclostationarity induced by the CP to the transmitted signal [17, 18]. In both cases, the equalizer coefficients are estimated using a second-order least square fitting, in combination with the constant modulus algorithm, to enhance performance. Muquet *et al.* exploit the redundancy introduced by the addition of the CP by evaluating the auto-correlation matrix of the received signals [19]. Heath *et al.* introduced a subspace approach for blind channel identification using cyclic correlations at the OFDM receiver together with impulse response shortening. These authors use two different approaches called one-cycle (OC) and two-cycle (TC) [20]. Wensheng *et al.* designed a OC blind channel estimation algorithm based on cyclostationarity properties of OFDM signals. The channel is uniquely identified from the single frequency obtained after analyzing the z-transform of the delay variable [21].

Other types of statistical methods precode the transmitted signal and exploit its induced auto-correlation properties at the receiver to estimate the CIR. Most of these methods precode the signal without adding any redundancy, so the overall throughput of the system is not affected [10, 22, 23, 24].

Banani *et al.* developed a statistical iterative blind channel estimation technique in which a decision algorithm first makes primary symbol estimates of the data on each subcarrier based on a constrained linear MMSE criterion [25]. These primary data estimates are then used for the MMSE channel estimation. The technique is appropriate for open-loop systems, which means there is no adaptive power control at the transmitter, making the method unsuitable for link adaptation. Another iterative method is presented in Ref. [26], which is based on Independent Component Analysis, but it is for zero padding OFDM systems which transmit no information during the guard interval. A method based on the Hilbert transform for causal signals, where the amplitude and phase are related, is given in Ref. [27]. The phase-frequency response is derived from the estimation of the amplitude-frequency response.

Many studies have used Kalman filters to estimate and track the channel. However, in most cases they require training symbols. One method that does not need these training symbols, and is therefore blind, is reported in Ref. [28]. It was proposed to use a Mixture Kalman filter for each subchannel to explore the time domain correlation of the channel, and subsequently an MMSE combiner to further refine the channel estimates, by exploring the frequency domain correlation.

Most deterministic methods for channel estimation are based on the finite alphabet property of the transmitted signal. These methods typically suffer from very high computational complexity. One example is provided in Ref. [29], where it takes into account specific properties of M-phase shift keying (M-PSK) and quadrature amplitude modulation (QAM) signals. The method of Necker and Stuber is based on the maximum likelihood principle, but it requires constant modulus modulations, otherwise it becomes too computationally complex. In addition, different modulation schemes are required for the adjacent subcarriers [30]. Wei *et al.* were able to lower the complexity by reducing the searching range for the maximum likelihood sequence [31]. Similarly, a reduced complexity minimum distance algorithm, which

exploits constraints on the unwrapped phase of finite impulse response (FIR) systems, results in significant reductions of the numerical complexity, and in many cases eliminates the exhaustive search for the maximum likelihood sequence [32]. A different approach is based on the fact that the transmitted signal is confined to an allocated bandwidth, and therefore, if sampled at a sufficiently fast rate, the resulting discrete-time signal is band-limited and exhibits a smooth waveform, so the channel can be estimated via interpolation [33]. The algorithm is complex but most of its computations can be calculated beforehand.

The statistical and deterministic methods for blind channel estimation are summarized in table 2.1, which shows their performance and the conditions under which they were tested. It is important to note that the channel models used in each work are different, and performance is hard to compare when different algorithms are tested under different scenarios. The objective of this comparison is to give the reader an initial idea of how they perform. The results presented are for after the algorithms converged and have been extracted from the references previously cited. The four different parameters in the results section of the table are BER, symbol error rate (SER), normalized mean square error (NMSE), mean square error (MSE) and root mean square error (RMSE).

Type	**References**	**Testing parameters**			**Results**						
		Modulation	*Number of subcarriers*	*Number of multipath components*	*SNR or* E_b/N_0		*BER*	*SER*	*NMSE*	*MSE*	*RMSE*
Statistical	[14]	QPSK / 64-QAM	64	9	SNR	10 dB	$5 \cdot 10^{-2}$ / $2 \cdot 10^{-1}$	-	-	$3 \cdot 10^{-1}$ / $3 \cdot 10^{-1}$	-
						20 dB	10^{-2} / $7 \cdot 10^{-2}$	-	-	$3 \cdot 10^{-2}$ / $3 \cdot 10^{-2}$	-
	[15]	BPSK	120	3	SNR	10 dB	-	-	-	-	10^{-1}
						20 dB	-	-	-	-	$1.5 \cdot 10^{-2}$
	[16]	16-QAM	32	5	SNR	10 dB	-	$4.5 \cdot 10^{-1}$	-	-	-
						20 dB	-	$5 \cdot 10^{-2}$	-	$4 \cdot 10^{-2}$	-
	[17, 18]	QPSK	20	4	SNR	10 dB	$3 \cdot 10^{-2}$	-	-	-	-
						20 dB	$3 \cdot 10^{-3}$	-	-	-	-
	[19]	QPSK	128	30	SNR	10 dB	-	-	-	$1.5 \cdot 10^{-2}$	-
						20 dB	-	-	-	-	-

[20]	16-QAM	15	4	SNR	10 dB	$9 \cdot 10^{-2}$	-	-	-	$0.25(OC)$ / $0.015(TC)$
					15 dB	$1.5 \cdot 10^{-2}$	-	-	-	$0.25(OC)$ / $0.15(TC)$
[21]	QPSK	32	5	SNR	10 dB	$4 \cdot 10^{-3}$	-	-	$1.5 \cdot 10^{-3}$	-
					15 dB	$3 \cdot 10^{-5}$	-	-	$1.2 \cdot 10^{-3}$	-
[22]	QPSK	64	3	SNR	10 dB	-	-	$5.5 \cdot 10^{-4}$	-	-
					20 dB	-	-	$3 \cdot 10^{-4}$	-	-
[23]	QPSK (Data encoded with 1/2 convolutional code)	16	4	E_b/N_0	10 dB	$5 \cdot 10^{-2}$	-	$3.3 \cdot 10^{-3}$	-	-
					20 dB	$1.8 \cdot 10^{-3}$	-	$3 \cdot 10^{-4}$	-	-
[10]	16-QAM	64	3	SNR	10 dB	$2 \cdot 10^{-1}$	-	10^{-3}	-	-
					20 dB	$2 \cdot 10^{-2}$	-	$2 \cdot 10^{-4}$	-	-
[24]	QPSK	64	9	SNR	10 dB	$8 \cdot 10^{-2}$	-	$3 \cdot 10^{-3}$	-	-
					20 dB	10^{-2}	-	$6 \cdot 10^{-4}$	-	-

	[25]	QPSK	64	8	SNR	10 dB 20 dB	$6 \cdot 10^{-2}$ 7^{-3}	- -	- $2 \cdot 10^{-4}$	- -	- -
	[26]	QPSK	64	-	SNR	10 dB 20 dB	$4 \cdot 10^{-2}$ $4 \cdot 10^{-3}$	- -	- -	- -	- -
	[27]	QPSK	256	4	SNR	10 dB 20 dB	- -	- -	$4 \cdot 10^{-3}$ $2 \cdot 10^{-4}$	- -	- -
	[28]	BPSK	16	2	SNR	10 dB 20 dB	$3 \cdot 10^{-2}$ $4 \cdot 10^{-3}$	- -	$4 \cdot 10^{-2}$ $6 \cdot 10^{-3}$	- -	- -
Deterministic	[29]	64-QAM	16	2	E_b/N_0	10 dB 20 dB	- -	- -	$7 \cdot 10^{-4}$ -	- -	- -
	[30]	QPSK / 3-PSK combined (Data encoded with 1/2 convolutional code)	1512	50 (Results for a typical urban channel)	E_b/N_0	10 dB 20 dB	$2.5 \cdot 10^{-4}$ -	- -	- -	$6 \cdot 10^{-3}$ -	- -

	[32]	4-DPSK	64	4	SNR	10 dB	-	$1.7 \cdot 10^{-1}$	-	$2 \cdot 10{-}1$	-
						20 dB	-	$2.5 \cdot 10^{-2}$	-	$9 \cdot 10^{-3}$	-
	[31]	QPSK	512	8 (For ITU-VA and exponential distribution channel profile)	E_b/N_0	10 dB	$4 \cdot 10^{-2}$ / $3 \cdot 10^{-2}$	-	-	-	-
						20 dB	$2 \cdot 10^{-3}$ / 10^{-7}	-	-	-	-
	[33]	QPSK	512	12	SNR	10 dB	-	-	10^{-3}	-	-
						20 dB	-	-	$4 \cdot 10^{-5}$	-	-

Table 2.1: Referenced OFDM blind channel estimator comparison

Chapter 3

OFDM blind equalizer

3.1 Algorithm

In this chapter, a channel estimator is presented that modifies the simple linear precoding estimation introduced by Petropulu *et al.* [10]. The estimator belongs to the statistical class. It precodes the OFDM symbols and exploits the induced correlation properties at the receiver to make the estimation. The algorithm transforms the i^{th} OFDM block of N information symbols $d_{i,k}$, where k is the subcarrier index and its range is $\{k = 0, ..., N-1\}$, according to

$$s_{i,k} = \frac{1}{\sqrt{1+|A|^2}} \left(d_{i,k} + (-1)^k A q_{i,k} \right) \tag{3.1}$$

where A is a pure imaginary number with $|A| \leq 1$ and $q_{i,k}$ is a pseudo-randomly generated binary phase shift keying (BPSK) symbol $\{q_{i,k} = -1, 1\}$. Both A and $q_{i,k}$ are predefined and assumed to be known to both the transmitter and receiver. According to Ref. [10], the precoding scheme has the following properties:

- It introduces no redundancy to the transmitted data, which makes the approach bandwidth efficient.
- It preserves the transmission power on each subcarrier.
- It maintains the zero-mean of the signal transmitted on each subcarrier.
- It maintains zero DC offset on each block.
- It introduces a correlation structure in signals transmitted over different subcarriers, which can be explored for channel estimation.

The effect of this precoding scheme on the signal constellation is represented in Figs. 3.1 to 3.6, for the six different modulation schemes that are tested: BPSK, quadrature phase shift keying (QPSK), 8-phase shift keying (8-PSK), 16-quadrature amplitude modulation (16-QAM), 32-square quadrature amplitude modulation (32-SQAM) and 64-quadrature amplitude modulation (64-QAM). These figures show that the precoding scheme creates two image constellations, one of them shifted up and the other one shifted down, of the original constellation. How much these images are shifted will depend on the value of the imaginary number A.

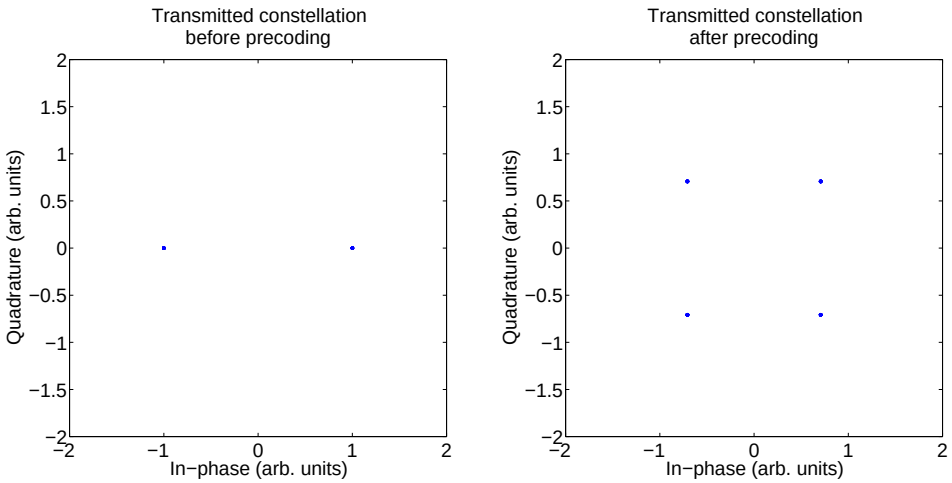


Figure 3.1: Signal constellation before and after precoding for BPSK modulation

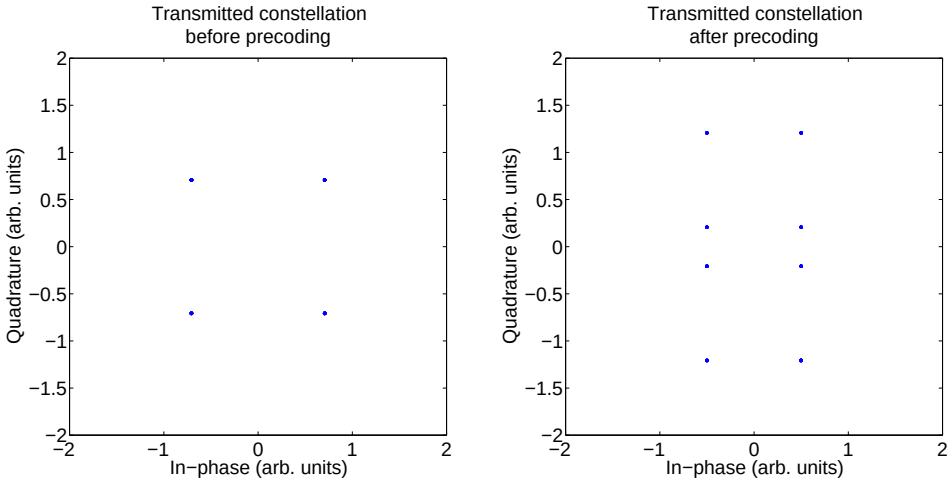


Figure 3.2: Signal constellation before and after precoding for QPSK modulation

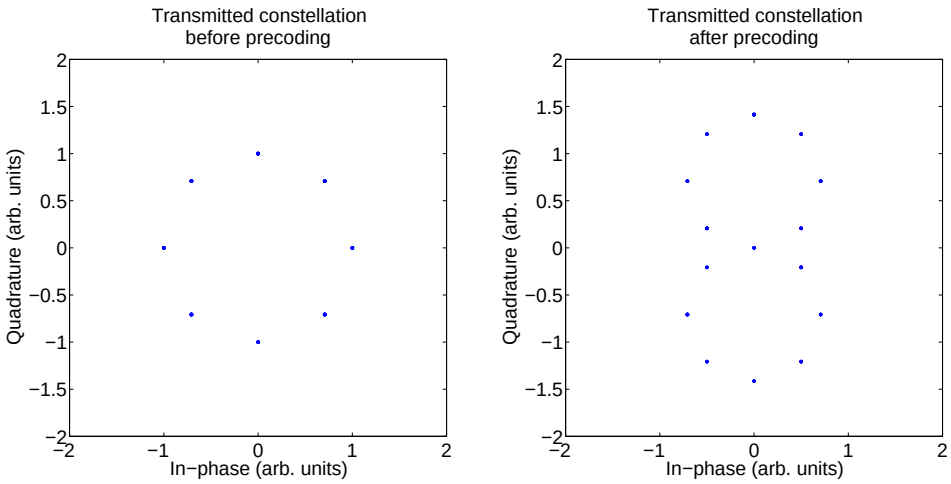


Figure 3.3: Signal constellation before and after precoding for 8-PSK modulation

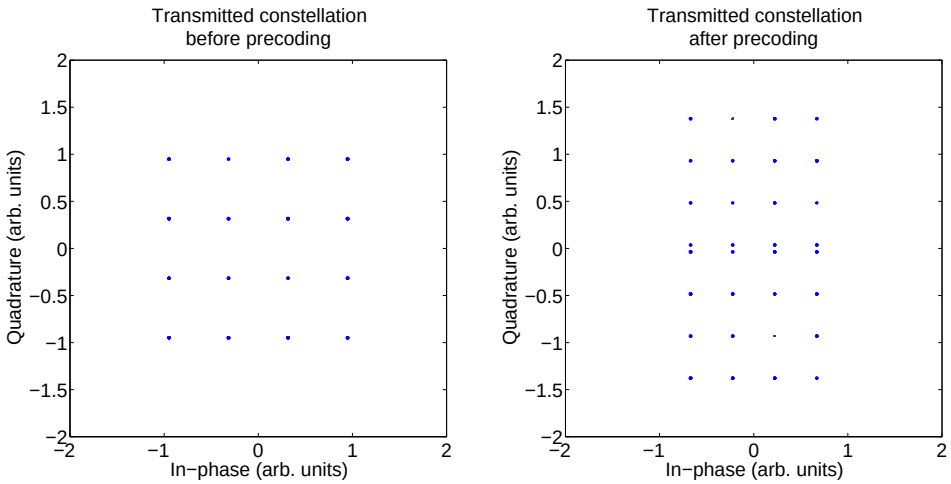


Figure 3.4: Signal constellation before and after precoding for 16-QAM modulation

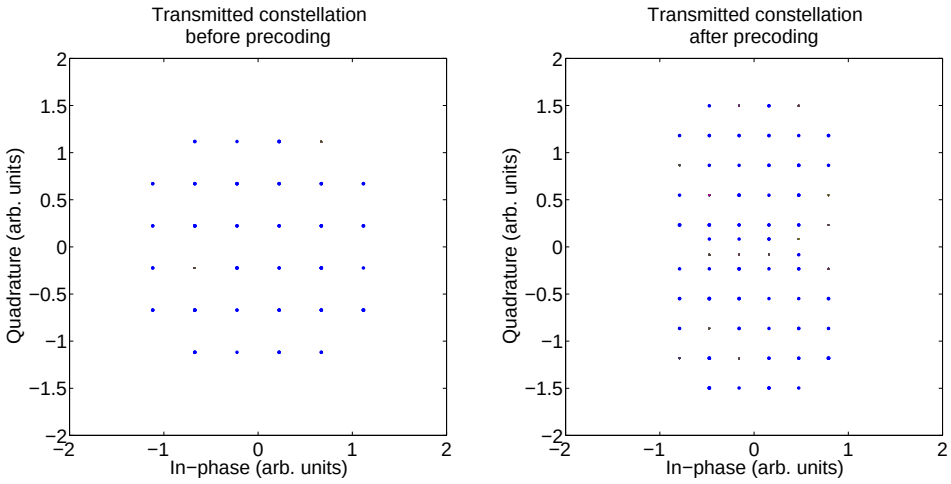


Figure 3.5: Signal constellation before and after precoding for 32-SQAM modulation

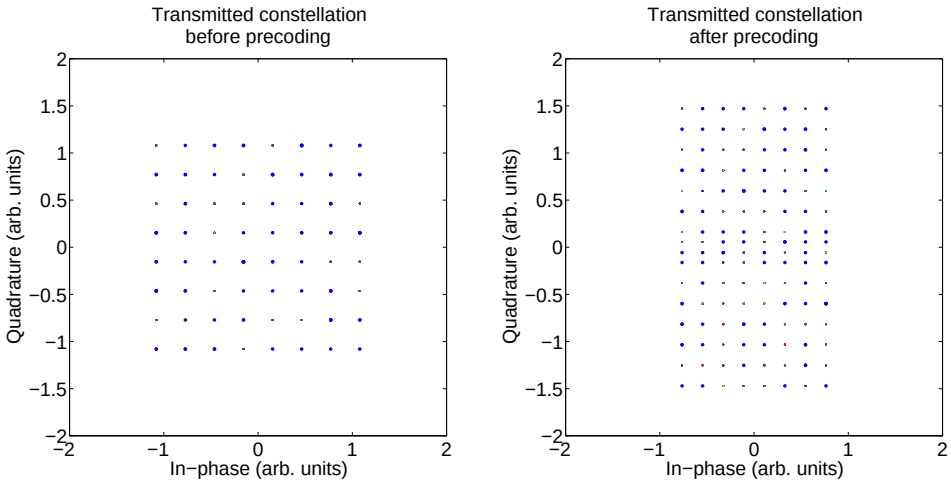


Figure 3.6: Signal constellation before and after precoding for 64-QAM modulation

The scheme presented in Ref. [10] precodes the symbols by replacing $q_{i,k}$ with the symbol of subcarrier T. Thus the precoding method can be written as:

$$s_{i,k} = \frac{1}{\sqrt{1+|A|^2}} \left(d_{i,k} + (-1)^k A d_{i,T} \right) \tag{3.2}$$

In the above equation a constant ($d_{i,T}$) is added to the subcarriers as a result of which the time signal amplitude shows very high peaks. These high peaks would require the quantizer to vary within a very large dynamic range, especially at high transmission powers. If the dynamic range is not wide enough, the signal will be distorted due to clipping. By using pseudo-random symbols that are different on each subcarrier, the precoded symbols derived from Eq. 3.1 will not suffer from this shortcoming. Fig. 3.7 compares the time OFDM signals that have been precoded with both equations.

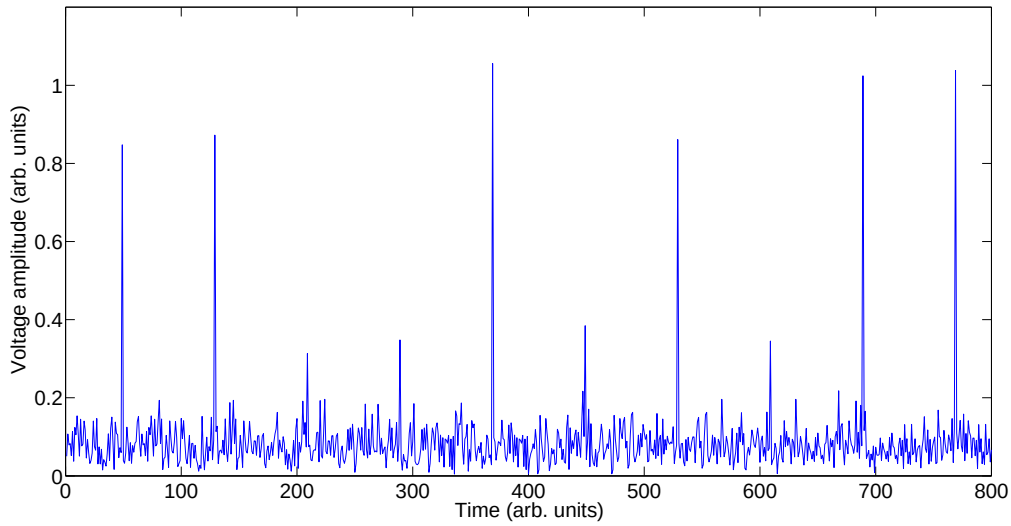
After the precoding, the symbols go through the regular OFDM system model. At the receiver, after the CP removal and the DFT steps, the i^{th} OFDM block is:

$$y_{i,k} = H(k)s_{i,k} + v_{i,k} = \frac{1}{\sqrt{1+|A|^2}} H(k) \left(d_{i,k} + (-1)^k A q_{i,k} \right) + v_{i,k} \tag{3.3}$$

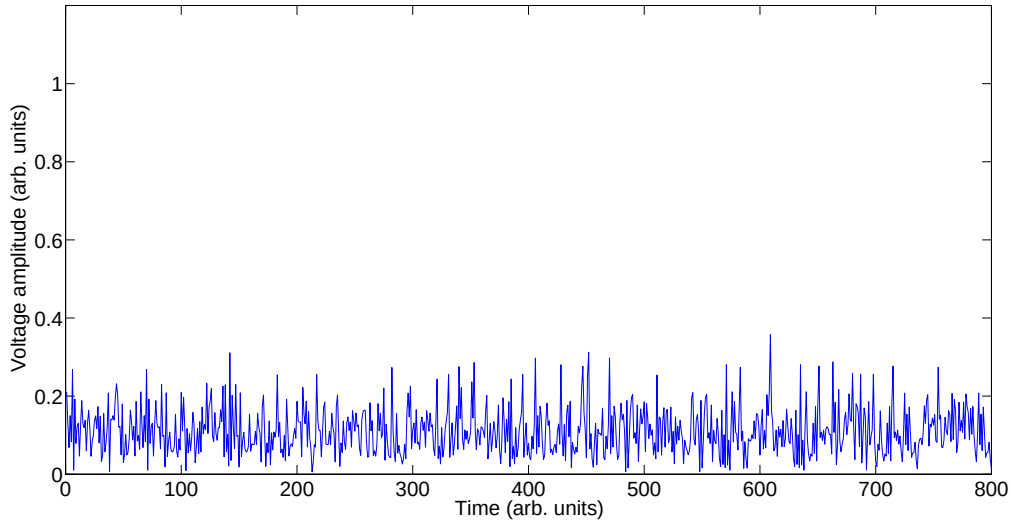
where $H(k)$ is the complex gain of the k^{th} subcarrier, and $v_{i,k}$ is the noise, modeled as a zero-mean complex Gaussian random variable with variance σ_v^2. The channel is modeled as a FIR filter of length L with tap coefficients $h(l), l = 0, ..., L-1; H(k) = \sum_{l=0}^{L-1} h(l) e^{-j2\pi \frac{kl}{N}}$. The channel does not change over the duration of at least one block, and is quasistationary between blocks.

The precoding induces a known correlation of each received subcarrier $y_{i,k}$ with the pseudo-random symbols $q_{i,k}$:

$$\begin{aligned}
z_k &= \mathrm{E}\left[y_{i,k} q_{i,k}^* \right] \\
&= \mathrm{E}\left[\left(\frac{1}{\sqrt{1+|A|^2}} H(k) \left(d_{i,k} + (-1)^k A q_{i,k} \right) + v_{i,k} \right) q_{i,k}^* \right] \\
&= \frac{1}{\sqrt{1+|A|^2}} H(k) \mathrm{E}\left[d_{i,k} q_{i,k}^* + (-1)^k A q_{i,k} q_{i,k}^* \right] + \mathrm{E}\left[v_{i,k} q_{i,k}^* \right] \\
&= \frac{1}{\sqrt{1+|A|^2}} H(k) \left(\mathrm{E}\left[d_{i,k} \right] \mathrm{E}\left[q_{i,k}^* \right] + (-1)^k A \mathrm{E}\left[|q|^2 \right] \right) \\
&= \frac{A}{\sqrt{1+|A|^2}} \sigma_q^2 H(k)
\end{aligned} \tag{3.4}$$



(a) Time signal amplitude obtained following the precoding method of Ref. [10]



(b) Time signal amplitude obtained following the precoding method proposed herein

Figure 3.7: Comparison of the resulting time signal amplitudes after precoding

where σ_q^2 is the variance of $q_{i,k}$. Therefore, the channel can be estimated from the correlation:

$$\hat{H}(k) = \frac{\sqrt{1+|A|^2}}{\sigma_q^2 A} z_k \tag{3.5}$$

In Ref. [10] the channel could be estimated only up to a complex constant; to resolve this ambiguity, they propose inserting one pilot symbol. This new estimator does not suffer from this limitation. Equation 3.5 completely estimates the channel, making it a totally blind method.

The channel estimate in Eq. 3.5 can be further improved if the length of the CIR is known, by performing the IDFT operation on $\hat{H}(k)$, setting the last $N-L$ samples to zero and then performing an N-point DFT to convert back to the frequency domain. This is known as denoising [10]:

$$\hat{H}_o(k) = \frac{1}{N} \sum_{l=0}^{L-1} \sum_{n=0}^{N-1} \hat{H}(n) e^{j2\pi \frac{ln}{N}} e^{-j2\pi \frac{kl}{N}} \tag{3.6}$$

To estimate the channel, the correlation can be calculated from a finite number of samples:

$$\hat{z}_k = \frac{1}{J} \sum_{i=1}^{J} y_{i,k} q_{i,k}^* \tag{3.7}$$

where J is referred to as the smoothing factor

3.2 Simulation results

The performance of the proposed method is compared to that of Ref. [10]. The OFDM system has $N = 64$ subcarriers and a cyclic prefix length $CP = 16$. The simulated channel is an FIR filter. The method was tested under two different channel lengths: $L = 3$ and $L = 6$. Each tap was modeled as a Rayleigh random variable and then kept constant for the whole simulation. Three thousand OFDM blocks were simulated and divided into groups of J symbols each. Two different smoothing factors were tested: $J = 100$ and $J = 200$. Best results were obtained for $A = j$, where j is the imaginary unit. To measure performance, the NMSE is calculated:

$$\text{NMSE} = \frac{\sum_{k=0}^{N-1} \left| \hat{H}_o(k) - H(k) \right|^2}{\sum_{k=0}^{N-1} |H(k))|^2} \tag{3.8}$$

Figure 3.8 depicts a plot of the NMSE versus the SNR for 64-QAM, although it should be noted that the modulation scheme does not influence the NMSE. Both the present method and the one reported in Ref. [10] have approximately the same NMSE for high SNR values. However, at low SNR, the method proposed herein achieves better results. Both methods have a higher NMSE for channels with longer impulse responses. The reason for this is that the non-zero components of the channel, after the denoising step, are estimates, while the null components are exactly zero, so the longer the channel, the noisier the estimate.

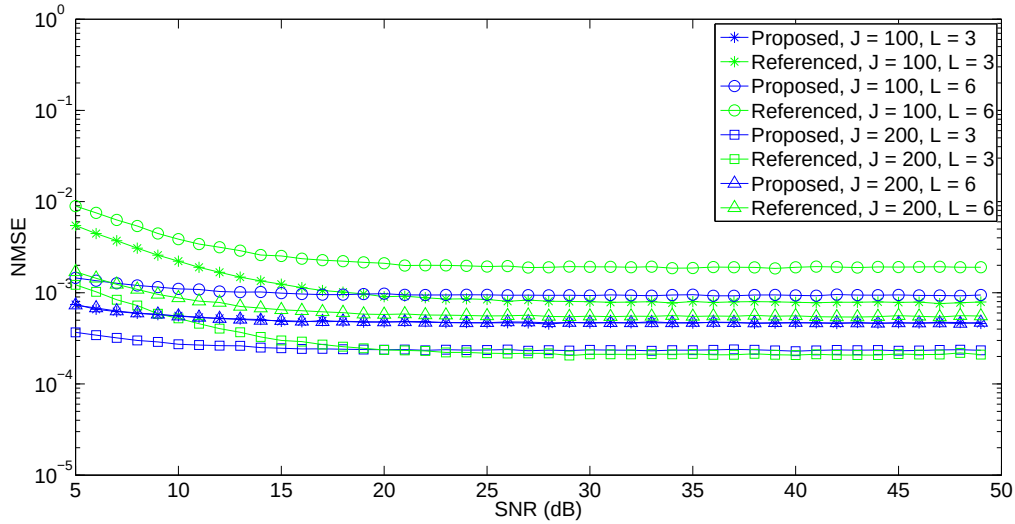


Figure 3.8: Normalized mean square error (NMSE) of the channel estimate vs. SNR

It was noted that the NMSE of a channel of length L is approximately L times the NMSE of a channel of length one, i.e.:

$$\text{NMSE}(L) \approx L \cdot \text{NMSE}(L = 1) \tag{3.9}$$

This is an interesting observation that will be exploited in the next chapter for the estimation of the BER.

Figures 3.9 and 3.10 compare the BER curves for channels of length $L = 3$ and $L = 6$ for six different modulation shcemes: BPSK, QPSK, PSK8, 16-QAM, 32-SQAM and 64-QAM. The BER curves reach a plateau where even with higher SNR values the BER will not decrease. This is due to the imperfect channel estimate, which also reaches a plateau, as shown in Fig. 3.8. The mentioned plateau is not evident in some of the BER curves of Figs. 3.9 and 3.10. This is because the plateau occurs outside of the range shown in these figures.

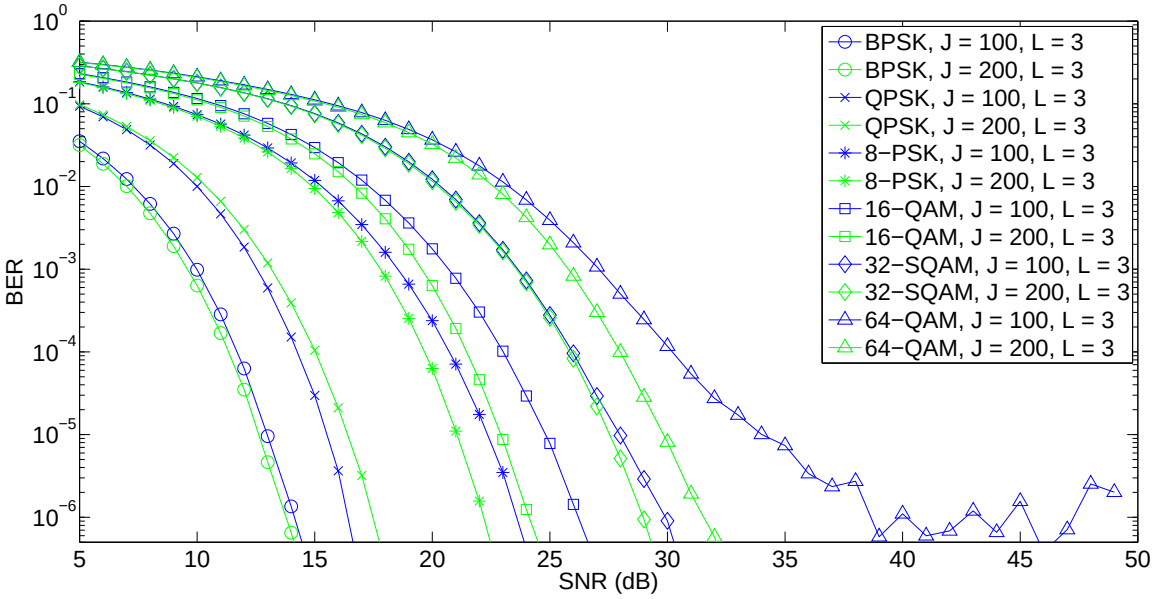


Figure 3.9: BER for a random channel of length $L = 3$

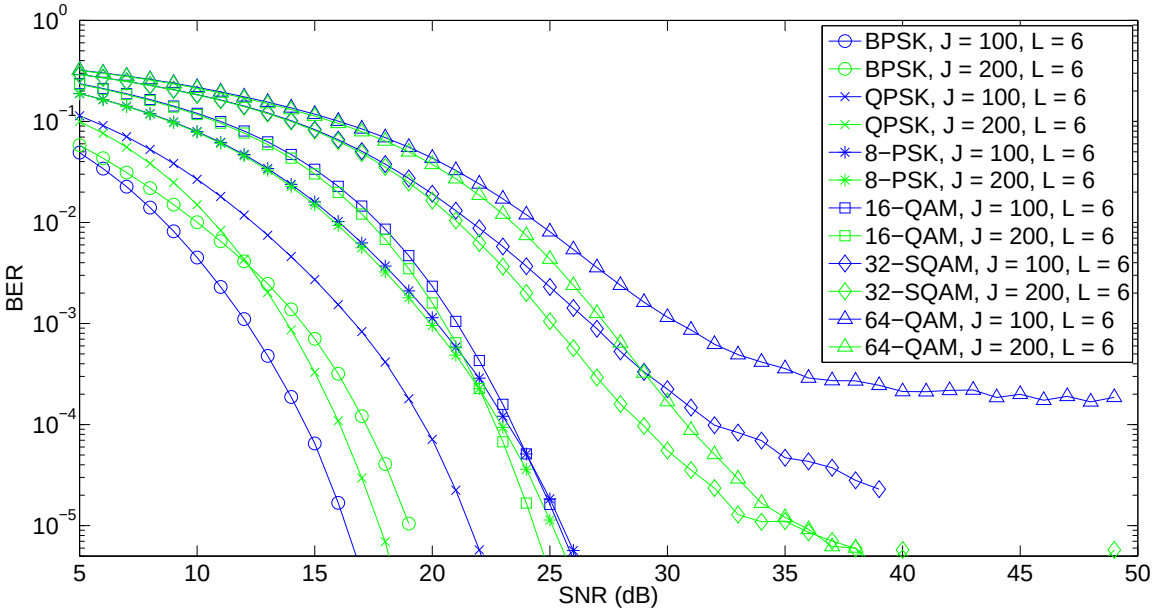


Figure 3.10: BER for a random channel of length $L = 6$

Chapter 4

System design

4.1 System model implementation

The system is designed as a single link between two nodes, the transmitter and the receiver, and is fully described in Ref. [3]. Each node consists of a Universal Software Peripheral Device 1 (USRP1) connected to a laptop. The USRP1 is used as a flexible and affordable SDR platform often used in academic research. Ettus Research [34] is well known for producing the USRP which provides a flexible RF hardware front end on which to send and receive data. There are several generations of USRP devices, each with a different bus speed for user data processing. For the present system, the USRP1 generation was chosen, which is defined by the types of daughter cards supported along with its universal serial bus (USB) interface. The USRP contains a single-core processor. On top of the USRP board itself, a daughter card must be used to specifically define the RF parameters that it supports. Different daughter cards will have different characteristics. In this case, the transmitter used the RFX900 daughter card and the receiver used the WBX daughter card. This selection was made because the RFX900 card is a high-performance transreceiver designed specifically for the 900 MHz band, while the WBX card is a wide bandwidth transreceiver and provides access to different bands within the range of 50 MHz to 2.2 GHz. The RFX900 was used at the transmitter because it provided a higher output power, while the WBX was used at the receiver because it has a lower noise figure. Both radios used the VERT900 antenna, which is designed for the 824 to 960 MHz and 1710 to 1990 MHz bands, and is an omni-directional vertical antenna. Pictures of the radios with their laptops, the used antennas and the two daughter boards are provided in Figs. 4.1, 4.2, 4.3 and 4.4.

Liquid-USRP is used as a high level interface to communicate with the USRP1s, in conjunction with *liquid-DSP*. The latter has already implemented the basic blocks for a wireless communication, and introduces a framing structure that puts them all together to send and receive data. The full tutorial on *liquid* can be found in Refs. [35, 36]. This section will

Figure 4.1: Transmitter node (on the right) and receiver node (on the left)

Figure 4.2: Antenna used on the transmitter and receiver (VERT900)

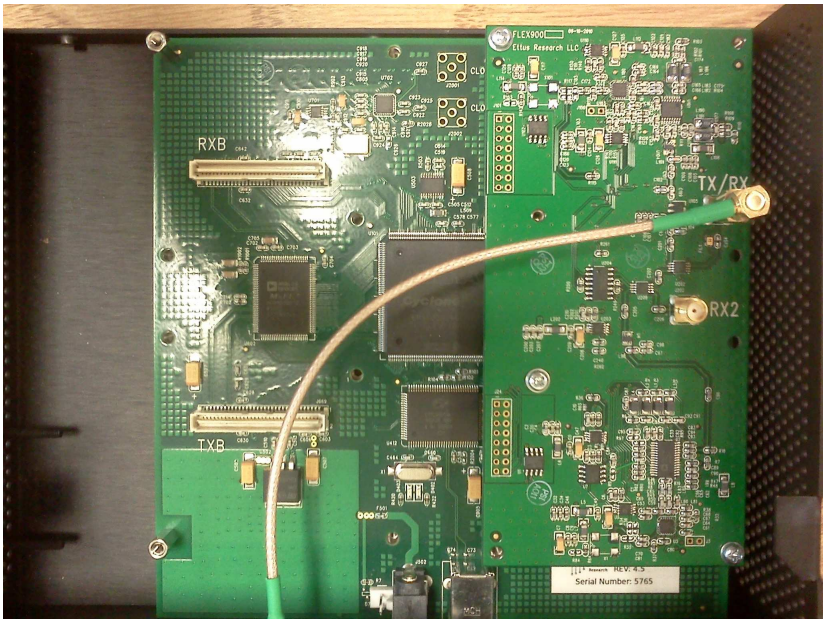


Figure 4.3: Transmitter daughter card (RFX900)

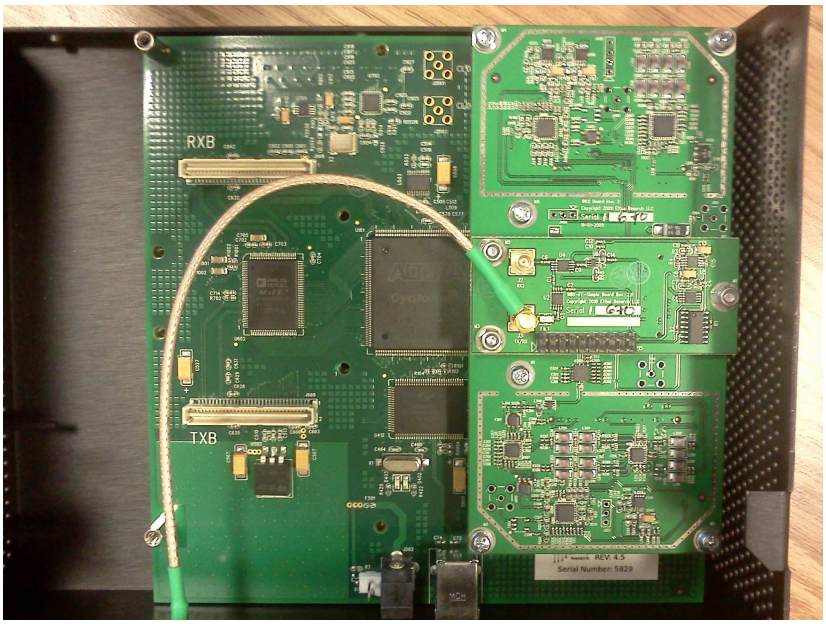


Figure 4.4: Receiver daughter card (WBX)

describe some of the more important blocks and also how some of them had to be modified to implement the previously described OFDM blind channel estimator.

4.1.1 Description of *liquid*

For an OFDM system, *liquid* defines three different types of subcarriers, which are called null, pilot and data subcarriers:

- *Null subcarriers*: This option disables the subcarrier. Its main purpose is to create guard bands on both ends of the allocated signal bandwidth to prevent aliasing during up-conversion/interpolation and to avoid interfering with adjacent bands. However, any subcarrier can be defined as null, allowing for spectral notching.
- *Pilot subcarriers*: These are used for tracking slowly-varying channel effects such as carrier frequency/phase offsets and timing frequency offset that are due to a residual error from the initial estimation made by the preamble. The pilots are pseudo-random BPSK symbols generated by a linear feedback shift register, and are different for each pilot subcarrier and from one OFDM symbol to the next. The exact pilot sequence is known by both the transmitter and receiver. They allow for coherent demodulation.
- *Data subcarriers*: These are used for carrying the payload, modulated with the desired scheme.

On the transmitter side, packets are generated that encapsulate an 8-byte user-defined header and a payload of variable length that contains the data to be sent. The user can define multiple physical-layer parameters, such as the number of subcarriers and their type, the cyclic prefix length, the type of forward error-correcting (FEC) coding, the modulation scheme and payload length, among others. The receiver can compensate for carrier phase and frequency offset, timing offset and can also equalize the received signal to compensate for multi-path fading.

Multi-carrier systems, such as OFDM, are significantly more sensitive to carrier frequency offsets and Doppler shifts, which cause ICI, so it is especially important for these systems to have good synchronization. To help in the synchronization process, the transmitter appends to each packet some special preamble symbols which help to estimate the carrier frequency offset, recover the symbol timing, and estimate the channel.

The structure of a packet has three main components:

- *Preamble*: The preamble consists of two types of symbols that are also known by the receiver, called S_0 and S_1 symbols. The transmitter generates multiple S_0 symbols and only one S_1 symbol. The S_0 sequences are used to detect the packet, and provide an

initial estimate of the carrier frequency and timing offset. They do not have a cyclic prefix to ensure continuity between the S_0 symbols to eliminate the ISI. The S_1 sequence, which does have a cyclic prefix, makes it possible to fine-tune these estimates, synchronizing the receiver with the incoming signal, and making an estimation of the channel which will then be used to equalize the rest of the symbols of the packet. In this manner, the receiver knows when the header starts and samples at the correct instances to avoid ICI.

- *Header*: The header is made up of fourteen bytes of information. The total amount of OFDM symbols used by the header will depend on the number of subcarriers used and their allocation of nulls, pilots and data. Part of the header (six bytes) is reserved to be used internally by *liquid*, and the remaining eight bytes can be defined by the user. The internal header contains information on the type of modulation and FEC coding used by the payload, as well as its length and the payload data validation scheme. The header is encoded and uses a robust modulation scheme (such as BPSK or QPSK) that is known beforehand by both the transmitter and receiver.
- *Payload*: The payload contains the information to be sent from the transmitter to the receiver. The length of the payload in bytes is defined by the user. The exact number of OFDM symbols to be sent will depend on the number of subcarriers and their allocation.

The structure of the packets previously described is shown in Fig. 4.5, where H_m are the header symbols and P_m are the data symbols.

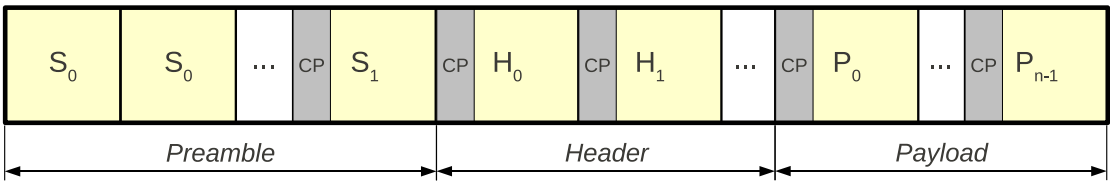


Figure 4.5: Packet structure, where S_0 and S_1 are the two different types of preamble symbols, H_m are the header symbols and P_m are the data symbols

After the carrier frequency/phase offset and timing offset are initially estimated and compensated for with the received preamble, there will still be residual offsets due to imperfect

estimation. The model of the received signal with these residual offsets can be represented as:

$$y_{i,k} = s_{i,k} \cdot H(k) \cdot e^{j(\Delta\theta + \Delta\omega \cdot i + \Delta t \cdot k)} + v_{i,k} \tag{4.1}$$

where i and k are the time and frequency indexes respectively; and $\Delta\theta$ is the residual phase offset, $\Delta\omega$ is the residual carrier frequency offset and Δt is the residual timing offset. After equalizing each received OFDM symbol with the channel estimate $\hat{H}(k)$, the phase of the pilot subcarriers will be rotated by $\varphi(k)$ due to the offsets. This rotation varies linearly with the frequency, as shown in Fig. 4.6. Therefore, *liquid* measures the rotation at each pilot subcarrier and fits these measurements to a straight line to obtain an estimate of the offsets, and then rotates each data subcarrier back to compensate for the offsets.

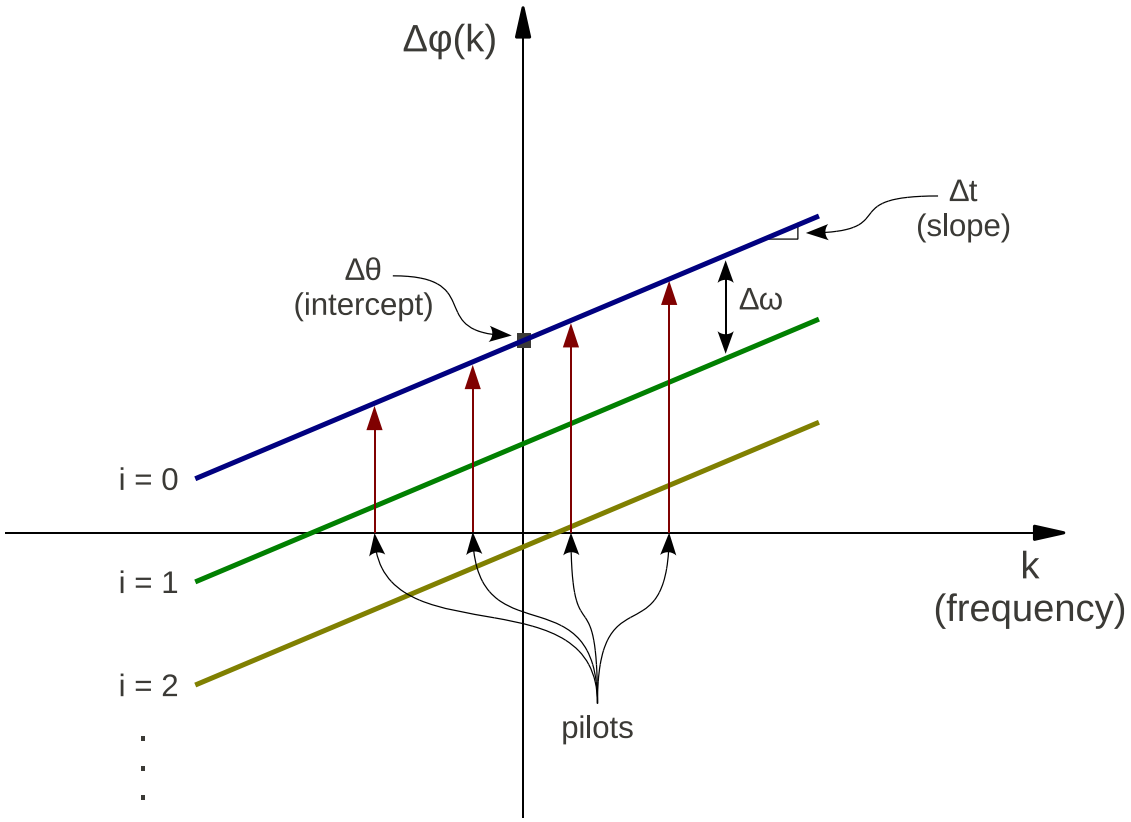


Figure 4.6: Phase rotation ($\Delta\varphi(k)$) due to residual carrier frequency/phase offsets ($\Delta\omega$ and $\Delta\theta$) and timing offset (Δt), and i and k are the time and frequency indexes respectively

4.1.2 Modifications to *liquid*

As described in the previous subsection, upon the detection of a packet, the system follows the general procedure described below:

1. Initial estimation of carrier frequency/phase offsets and timing offset using the S_0 and S_1 symbols, and then compensation for these effects.
2. Estimation of the channel from the S_1 symbol.
3. Use of the pilots in the header and payload and the channel estimate to refine the estimation of the carrier frequency/phase and timing offsets and compensate for them.
4. Equalization of each OFDM symbol of the header and payload.

However, this process flow can no longer be followed when implementing the proposed blind channel estimator. The system first requires an estimate of the channel to be able to refine and compensate for the carrier frequency/phase and timing offsets, but the blind equalizer first needs to receive J symbols with no offsets to estimate the channel. To overcome this problem, the channel estimator implemented by *liquid* is used to equalize only the phase of the pilot subcarriers to next estimate the offsets. The data subcarriers are then compensated for the offsets, and, finally, the blind channel estimator can be used to equalize the data subcarriers. For a real system, this procedure would be totally impractical. However, one of the goals of the present work is to measure the performance of the blind channel estimator with real over-the-air results assuming that the other channel impairments have already been compensated for.

It should be noted that the phase and timing offsets in Eq. 4.1 are independent of the time index. Therefore both parameters would induce a constant rotation on each received OFDM symbol belonging to the same packet, as opposed to the frequency offset, which would increasingly rotate the received OFDM symbols. The proposed blind equalizer needs the channel to be quasi-static so that the J symbols can be correctly estimated. Therefore, the effects of the phase and timing offsets could be considered to be part of the CIR, and only the frequency offset would need to be compensated for before estimating the channel with the proposed method. In this case, one possible way of estimating the frequency offset would be to calculate the difference in phase between the current received pilot subcarriers and those received previously, and then average over all pilot subcarriers. When testing this method with over-the-air transmissions, however, it was found to produce poor results.

The general system model is graphically represented in Fig. 4.7.

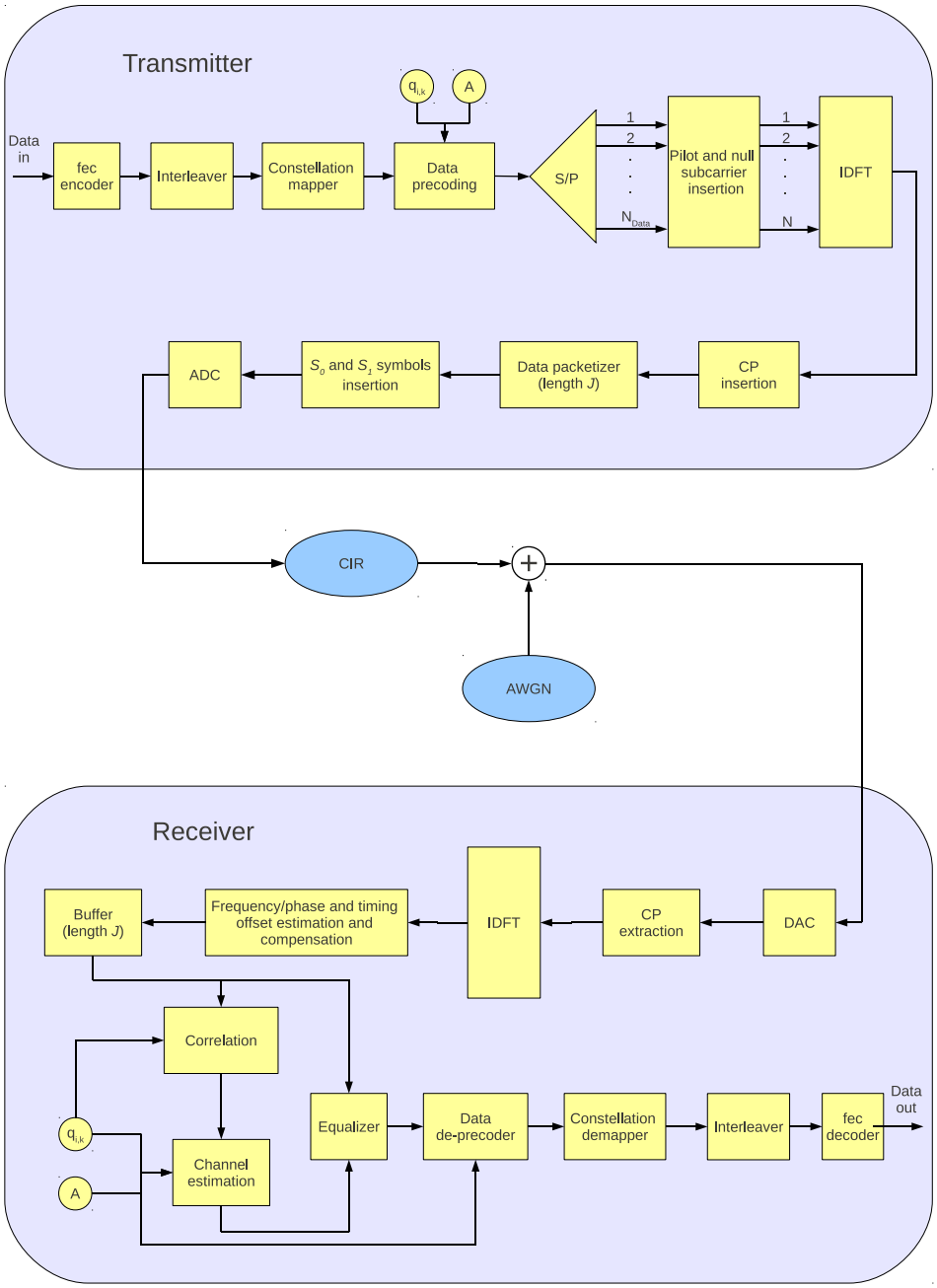


Figure 4.7: System model, where FEC means forward error correcting, $Q_{i,k}$ and A are the precoder parameters, S/P means serial to parallel, IDFT is the inverse discrete Fourier transform, CP is the cyclic prefix, J is the smoothing factor, S_0 and S_1 are the preamble symbols, ADC stands for analog to digital converter, CIR is the channel impulse response, AWGN stands for additive white Gaussian noise, DAC means digital to analog converter and IDFT is the inverse discrete Fourier transform

4.2 Performance estimation

One possible way to achieve link adaptation would be for the transmitter to estimate its performance whenever its transmission parameters are modified. A theoretical approach could be used to make this estimation. The method described here to estimate the BER improves on the one previously reported in Ref. [3].

4.2.1 BER estimation for an uncoded transmission

To determine the overall BER of the system, the channel response at each subcarrier is assumed to be flat and quasi-static over time. In this situation, each subcarrier can be modelled as if it were to go through an AWGN channel with a different gain. Therefore, to estimate the BER per data subcarrier ($P_{e_{subcarrier}}$), the system uses the theoretical formulas described in Ref. [4] for AWGN channels, coherent demodulation and the different modulation schemes used:

- BPSK:

$$P_{e_{subcarrier}} = \frac{1}{2}\text{erfc}\left(\sqrt{(E_b/N_0)_{subcarrier}}\right) \tag{4.2}$$

- M-PSK ($M > 2$):

$$P_{e_{subcarrier}} = \frac{1}{\log_2(M)}\text{erfc}\left(\sin\left(\frac{\pi}{M}\right)\sqrt{\log_2(M)(E_b/N_0)_{subcarrier}}\right) \tag{4.3}$$

- M-QAM:

$$P_{e_{subcarrier}} = \left(\frac{2}{\log_2(M)}\right)\left(\frac{\sqrt{M}-1}{\sqrt{M}}\right)\text{erfc}\left(\sqrt{\frac{3\log_2(M)}{2(M-1)}(E_b/N_0)_{subcarrier}}\right) \tag{4.4}$$

where M is the number of bits per symbol and $(E_b/N_0)_{subcarrier}$ is the energy per bit to noise power spectral density ratio per subcarrier. The parameter $(E_b/N_0)_{subcarrier}$ can be calculated from the signal to noise ratio per subcarrier ($(SNR)_{subcarrier}$), M, the system's symbol rate (R_S) and the bandwidth (BW):

$$(E_b/N_0)_{subcarrier} = (SNR)_{subcarrier} + 10\log_{10}\left(\frac{BW}{R_S \cdot \log_2(M)}\right) \tag{4.5}$$

In *liquid*, the ratio of $\frac{BW}{R_S}$ is set automatically to $\frac{1}{2}$.

The $(SNR)_{subcarrier}$ can be calculated from the overall system's SNR by using the estimated channel in the frequency domain at each data subcarrier. First, the channel estimate needs

to be normalized so the mean power of the channel estimate over all the data subcarriers equals one:

$$\hat{H}_{o_{norm}}(k) = \frac{\hat{H}_o(k)}{\sqrt{\frac{1}{N_{data}} \sum_{k \epsilon N_{data}} \left| \hat{H}_o(k) \right|^2}} \tag{4.6}$$

where N_{data} is the number of data subcarriers. Therefore, $(SNR)_{subcarrier}$ can be calculated as:

$$(SNR)_{subcarrier} = SNR \cdot \left| \hat{H}_{o_{norm}}(k) \right|^2 \tag{4.7}$$

where SNR is in linear units.

4.2.2 BER estimation for a FEC coded transmission

Reference [37] describes how to calculate the word error probability (P_{E_c}) for a t-error correcting code (n, k) from the digit error probability of the uncoded case (P_{e_u}). It is assumed that the useful throughput (data rate of just information-bearing bits) for both the coded and uncoded data streams is the same, so a correction factor for the extra bits is added to E_b/N_0. However, for this system, what remains constant is the symbol rate and not the data bit rate. Therefore, for coded data, to calculate the word error probability, this correction factor is no longer needed. Thus, the digit error probability of the coded (P_{e_c}) data equals the digit error probability of the uncoded data, so $P_{e_u} = P_{e_c}$.

For data coded with a t-error correcting code (n, k), the coding scheme can correct a received word that has t errors or less. Therefore, only words with more than t errors in n bits will be received incorrectly. If $P(j, n)$ is the probability of j errors in n digits, then

$$P_{E_c} = \sum_{j=t+1}^{n} P(j, n) \tag{4.8}$$

There are $\binom{n}{j}$ ways in which j errors can occur in n digits

$$P(j, n) = \binom{n}{j} (P_{e_c})^j (1 - P_{e_c})^{n-j} \tag{4.9}$$

Therefore

$$P_{E_c} = \sum_{j=t+1}^{n} \binom{n}{j} (P_{e_c})^j (1 - P_{e_c})^{n-j} \tag{4.10}$$

For error correcting codes that use a small n, such as Hamming (7,4), and if $P_{e_c} \ll 1$, the most likely event for a word received incorrectly is the occurence of $t + 1$ errors.

$$P_{E_c} \approx \binom{n}{t+1} (P_{e_c})^{t+1} \tag{4.11}$$

The previous approximation is valid because the system uses an interleaver after the FEC encoding, so on the receiver side, even if burst errors occur, the deinterleaving process will spread them out.

If the received word is in error, then the decoded word of length k will also contain errors. To calculate the probability of error of the decoded data P_{e_d}, the law of total probability can be used:

$$P_{e_d} = P(\text{errors}_{\text{decoded}}) = \sum_{j=0}^{n} P(\text{errors}_{\text{decoded}} | \text{errors}_{\text{coded}} = j) \cdot P(\text{errors}_{\text{coded}} = j) \tag{4.12}$$

Making the same assumptions that were described previously in this subsection, Eq. 4.12 can be approximated to:

$$P_{e_d} \approx P(\text{errors}_{\text{decoded}} | \text{errors}_{\text{coded}} = 2) \cdot P(\text{errors}_{\text{coded}} = 2) \tag{4.13}$$

The system was tested for data encoded with the Hamming (7,4) scheme. In order to determine the probability of error of the decoded data if the coded word contains two errors ($P(\text{errors}_{\text{decoded}} | \text{errors}_{\text{coded}} = 2)$) a simulation was run to calculate it deterministically. This simulation generated 10^7 uncoded words of length 4 bits and encoded them using the Hamming (7,4) scheme. Two different random locations were generated within each coded word and its bits were flipped, so that each word always contained exactly two errors. Finally, the words were decoded and compared to the original uncoded words. It was found that the average amount of bit errors was 2.285. Therefore, the probability of error of the decoded data if the coded word contains two errors can be estimated to be $P(\text{errors}_{\text{decoded}} | \text{errors}_{\text{coded}} = 2) = \frac{2.285}{4}$. Finally, the BER per data subcarrier after it was decoded ($P_{e_{d_{sub}}}$) needs to be calculated. This is done by using the BER per data subcarrier for uncoded data ($P_{e_{u_{sub}}}$), which is a function of E_b/N_0 and was calculated in Eqs. 4.2, 4.3 and 4.4. For the case where the data was encoded with Hamming (7,4):

$$P_{e_{d_{sub}}} \approx \frac{2.285}{4}\binom{7}{2}\left(P_{e_{u_{sub}}}(E_b/N_0)\right)^2 \tag{4.14}$$

4.2.3 BER estimation with imperfect channel estimate

The steps in the previous two subsections to estimate performance assume that the system has perfect knowledge of the channel. In reality, the system has only an estimate, and this imperfect estimation degrades the performance. The work of Ref. [38] describes this degradation and makes it possible to quantify the signal-to-interference ratio (SIR) that it produces. The total interference (I) due to imperfect channel identification and additive Gaussian noise on the k^{th} subchannel is:

$$I = \left(\frac{H_k - H_k^{est}}{H_k^{est}}\right) X_k + n_k \tag{4.15}$$

where H_k^{est} is the estimated subchannel gain, H_k is the real subchannel gain, X_k is the transmitted sample and n_k is the noise of the k^{th} subchannel with power σ^2. Assuming the error in the channel estimation follows a Gaussian distribution, the total interference power is:

$$\sigma_k^2 = \sigma^2 + \left\|\frac{H_k - H_k^{est}}{H_k^{est}}\right\|^2 \epsilon_k \approx \sigma^2 + \left\|\frac{H_k - H_k^{est}}{H_k}\right\|^2 \epsilon_k = \sigma^2 + MSE \cdot \epsilon_k \tag{4.16}$$

where ϵ_k is the signal energy and MSE is the Mean Squared Error of the channel estimate, which is defined as:

$$MSE = \left\|\frac{H_k - H_k^{est}}{H_k}\right\|^2 \tag{4.17}$$

Therefore, if $SNR = \frac{\epsilon_k}{\sigma^2}$, the resulting SNR is

$$SIR = \frac{\epsilon_k}{\sigma_k^2} = \frac{\epsilon_k}{\sigma^2 + MSE \cdot \epsilon_k} = \frac{SNR}{1 + MSE \cdot SNR} \tag{4.18}$$

Accordingly, the BER formulas can be written as a function of SIR instead of SNR.

Next, some simulations were run to measure how well the system could predict its performance. The simulated system used the same parameters that were described in section 3.2. The channel length L was set to 3 and $J = 100$.

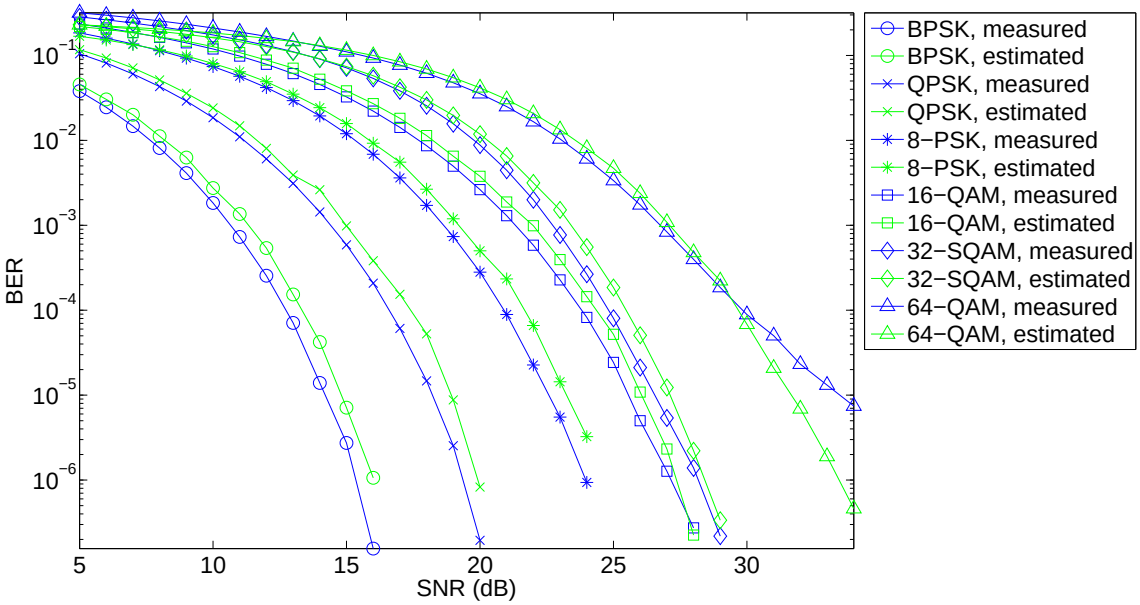


Figure 4.8: Measured and estimated BER for the six tested modulation schemes

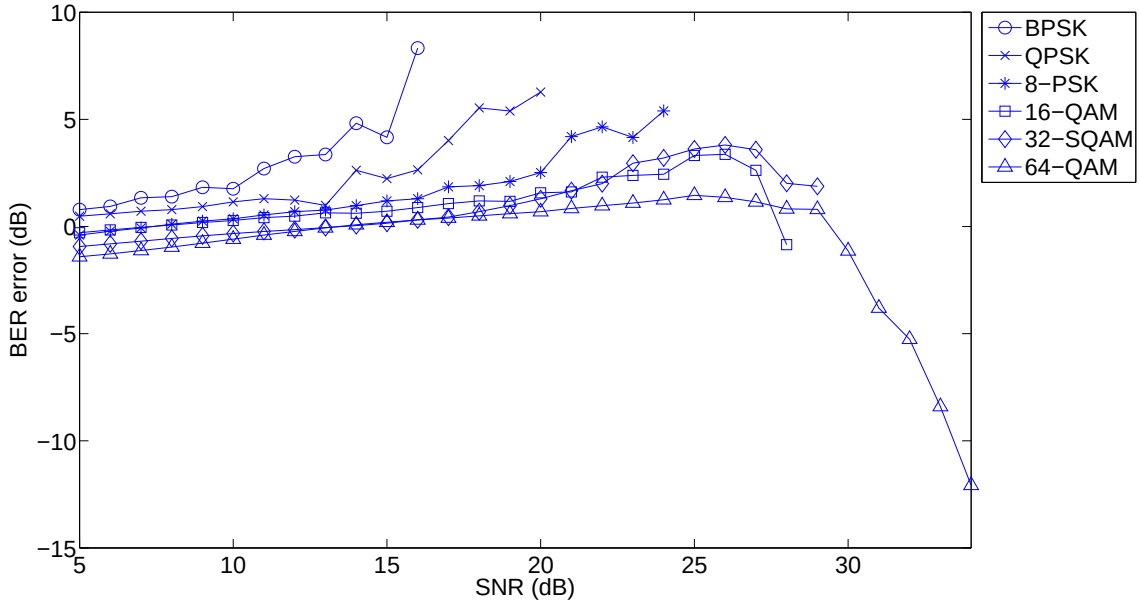


Figure 4.9: Error between the estimated and measured BER values for the six tested modulation schemes

In Fig. 4.8, the blue curves represent the BER measured for each of the six different modulation schemes that were tested. The green curves show what the system estimated the BER would be using Eq. 4.18 to adjust the SNR for the different modulation schemes characterized by Eqs. 4.2, 4.3 and 4.4. Figure 4.9 represents the proportional error between the estimated and measured BER, in dB:

$$\varepsilon_{BER} = 10 \log_{10} \left(\frac{BER_{estimated}}{BER_{measured}} \right) \tag{4.19}$$

The reason for representing the BER error in dB is to be able to adequately compare the difference between the measured and estimated BER when the estimated BER is overestimating or underestimating the real BER. Therefore, the BER error will be positive when the estimated BER is higher than its measured value, negative when the estimated BER is lower than the measured one, and zero if they are equal. Figure 4.9 indicates that the error is almost always within the range of -10 dB and 10 dB. This means that the error in the estimate is smaller than the order of magnitude of BER. The reason for this error between the estimate and the simmulated BER is due to the assumption that the error in the channel estimation follows a Gaussian distribution. This is not necessarily true. However, the asumpution is good enough that for the simmulated BER between 1 and 10^{-6}, the difference between the simulated and estimated BER is usually between -5 dB and 5 dB.

Chapter 5

Over-the-air tests

5.1 Issues to be considered for over-the-air transmissions

Many variables that are hard to model and thus are not considered in simulations can be responsible for undesirable effects in over-the-air transmissions, degrading performance. For instance, the hardware used by the USRP1s has some deficiencies and limitations. The oscillators at the transmitter and receiver, when set at the same frequency, do not operate at exactly that frequency, but will each have a different small offset. In theory, this should not be an issue because the receiver can estimate the frequency offset and compensate for it. However, the error in the estimation of the frequency offset $\varepsilon(\Delta\omega)$ increases with increasing frequency offset $\Delta\omega$ in a manner similar to what is represented in Fig. 5.1. Therefore, to reduce this error, the center frequency of the receiver was manually adjusted to achieve a frequency offset as close as possible to zero. The center frequency of the transmitter was set to 910.0000 MHz, and the center frequency of the receiver was set to 910.0037 MHz.

Another problem can occur if the transmitter laptop feeds data to the USRP1 at too slow a rate; this will cause the radio to emit "junk" data, where it will transmit anything it has in its outdated buffer, generating what are known as underruns. Alternatively, if the USRP1 on the receiver side feeds data to the laptop at too fast a rate to proccess, then some of the data gets dropped and lost, producing what are known as overruns. Both situations will cause a higher BER than what will be predicted by the system. To make sure these situations do not occur, the bandwidth needs to be reduced, because it is well known that, for a constant spectral efficiency, a narrower bandwidth means a slower throughput, so the USRP1s will need fewer samples, lowering the burden on the laptops. On the other hand, ODFM systems with narrower bandwidths are more sensitive to a frequency offset, so a compromise needs to be found. It was determined experimentally that an adequate bandwidth for the system was 100 kHz.

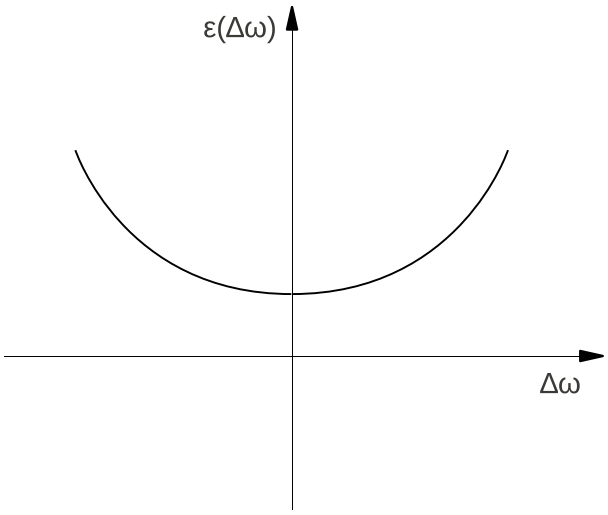


Figure 5.1: General shape of the variation of the error in the estimation of the carrier frequency offset ($\varepsilon(\Delta\omega)$) with the carrier frequency offset ($\Delta\omega$)

For an OFDM system with a bandwidth $BW = 100$ kHz and with $N = 64$ subcarriers, the bandwidth per subcarrier would be $BW_{sub} = \frac{BW}{N} = 1.5625$ kHz. Therefore, since electromagnetic waves propagate at the speed of light $c = 3 \cdot 10^8$ m/s, for the channel to exhibit multipath characteristics, the separation between multipath components should be at least $\frac{c}{BW_{sub}} = 192$ km. At such a large distance, the multipath components will be so attenuated that they will not have any discernible effect. This means that for the chosen bandwidth, the channel is effectively flat. However, the USRP1 hardware will deform this flat channel response. More specifically, the part of the digital circuitry that has the main effect on this distortion is the cascaded integrator-comb (CIC) filter in the field-programmable gate array (FPGA), causing the channel response in the frequency domain to have a shape similar to that shown in Fig. 5.2. An IDFT of the channel yields a value of approximately three for the CIR length L.

It is well known that OFDM signals have a high peak-to-average power ratio due to the overlap of multiple sinusoids [12]. The transmit power amplifier needs to be linear across the whole signal range, otherwise it will clip the peaks of the signal, causing distortion. This clipping problem is especially apparent at higher transmission powers where the range of the amplifier needs to be wider. Therefore, the transmitter USRP1 will clip the signal at higher powers, causing the BER to increase instead of decrease. Also, OFDM symbols modulated at higher modulation orders have a higher peak-to-average ratio, so they suffer from more distortion caused by clipping for the same transmission power. An example of this deterioration of the BER is shown in Fig. 5.3 for 64-QAM modulation, where the BER starts to increase with the SNR above a certain SNR value. For results that will be shown

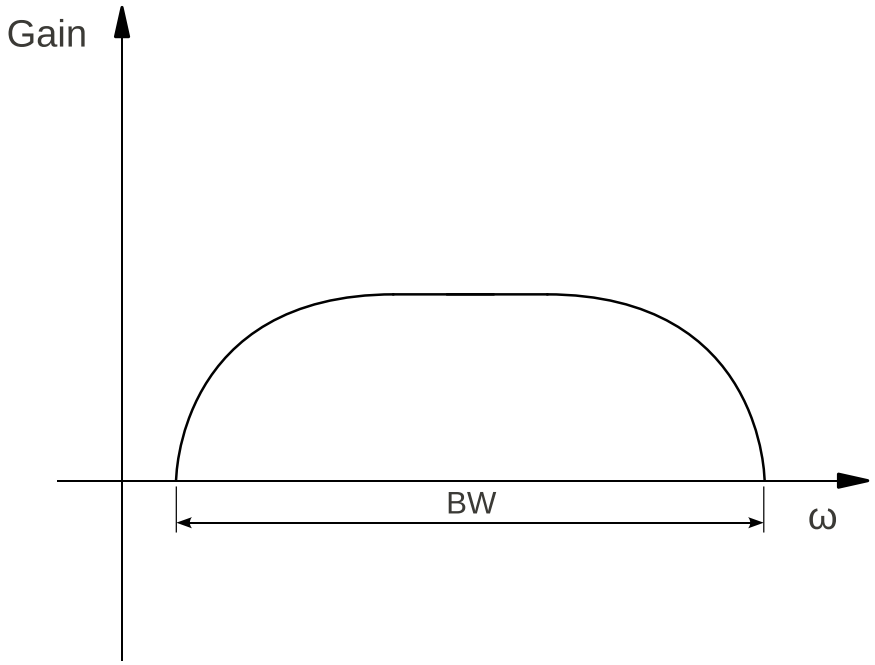


Figure 5.2: Typical example of the general shape of the channel response gain in the frequency domain (ω) caused by the CIC filter of the FPGA, which is responsible for "drooping" effect on both ends of the allocated bandwidth, where BW stands for the 3 dB bandwidth

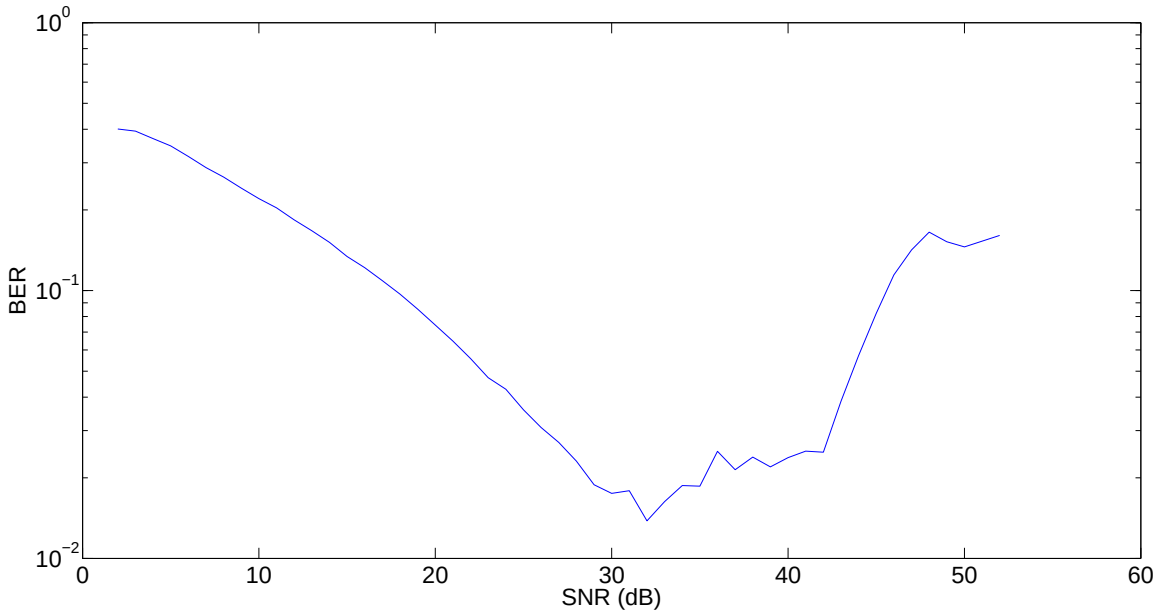


Figure 5.3: BER for a signal that is clipped at higher SNR values (for 64-QAM)

later on in this chapter, this clipping region will be omitted.

Finally, it was observed that electric charge was building up in the hardware and then discharging when the receiver was started; this was probably caused by the capacitators of the switches of the USRP1 not having enough time to discharge. The solution to this problem was to ignore the first 500 ms of data after the receiver hardware was started. This ensured sufficient time for the discharge to dissipate.

5.2 System setup

The OFDM system was tested at a carrier frequency of 910 MHz, which belongs to the industrial, scientific and medical (ISM) band, therefore a license is not required to transmit at this frequency. The allocated bandwidth was 100 kHz. It uses 64 subcarriers, of which 44 are data subcarriers, 6 are pilot subcarriers and 14 are null subcarriers. The exact allocation of each type of subcarrier is shown in Fig. 5.4. It should be noted that the subcarrier allocation is independent of both the frequency and bandwidth of the signal. The nulled subcarriers are located at each end of the allocated bandwidth to prevent aliasing during up-conversion/interpolation. The subcarrier located in the center of the signal bandwidth is also nulled to avoid interference from the leakage of the local oscillator. Six different types of modulation schemes were used: BPSK, QPSK, 8-PSK, 16-QAM, 32-SQAM and 64-QAM. In addition, the results were obtained for data that were not encoded with any FEC scheme

and for data that were encoded with the Hamming (7,4) scheme. The equalizer was tested for two different smoothing factor J values: 100 and 200.

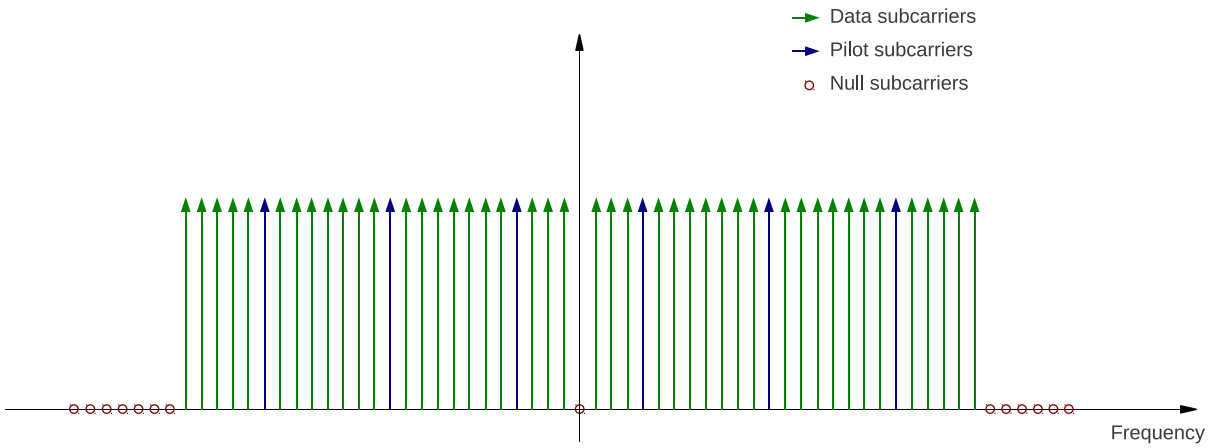


Figure 5.4: Subcarrier allocation automatically asigned by *liquid* for a system with 64 subcarriers, which is independent of the center frequency or bandwidth. There are 44 data subcarriers, 6 pilot subcarriers and 14 null subcarriers

The denoising step that was described in Chapter 3 to refine the channel estimate can no longer be performed when adding the null subcarriers. The reason for this is that the system does not have an estimate of the channel gain at the location of the null subcarriers, so it can no longer perform the IDFT operation on the frequency domain channel estimate. Accordingly, for the over-the-air experiments, to denoise the channel estimate, the corresponding amplitude at each subcarrier was fitted to a polynomial curve of order four.

The transmitter gain was randomly varied for each packet with a uniform distribution within a dynamic range of 50 dB, and the signal power at the receiver was stored in combination with the BER. This method ensured that enough samples were received to obtain an accurate calculation of the BER for the whole power range that was being measured without having to synchronize the transmitter and receiver. To calculate the SNR, the noise floor was measured previously, resulting in a value of -88.5 dBm. The SNR at the receiver ranged between approximately 0 dB and 60 dB. The distance between the transmitter and receiver radios was about 3 ft. This short distance was chosen in an attempt to mitigate the clipping distortion as much as possible. Since this distortion was found to occur predominantly at the transmitter, the radios needed to be near each other to avoid having the transmitter use higher gains while still ensuring good SNR values.

For the proposed equalizer to work correctly, the pseudo-randomly generated symbols $q_{i,k}$ used to precode the information-bearing symbols needed to be known by both the transmitter and the receiver. To do so, the sequence of symbols $q_{i,k}$ was always the same and would restart from the beginning for each packet. The used sequence had a length of 17 symbols:

$\{1, 1, -1, 1, 1 - 1, 1, 1, -1, 1, -1, -1, -1, 1, -1, 1, -1\}$.

The system parameters are summarized in table 5.1.

5.3 Results

This section describes the achieved performance of the system in terms of BER using the proposed equalizer, and compares it to the predicted performance of the system. Figure 5.5 shows an example of the channel estimate. It compares the initial channel estimate, in blue, with the smoothed estimate, in green. The frequencies of the null and pilot subcarriers are denoted with red circles. At these locations, the channel is not estimated because the received signal does not have the expected correlation charateristics that would be required.

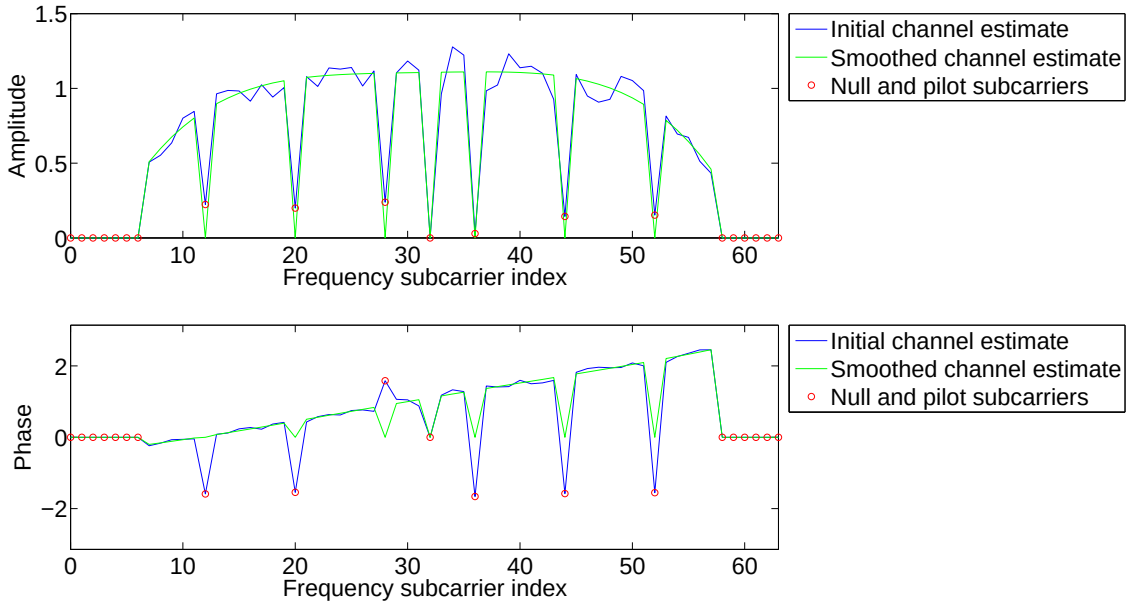


Figure 5.5: Channel estimate

Next, the received signal's normalized constellation for a high SNR is represented in Figs. 5.6 to 5.11, for before (on the left) and after (on the right) equalization and de-precoding. It should be noted that the received constellation shape before equalization is due to both the effect of the channel and the precoding scheme.

For the BER prediction, the SNR needs to be adjusted as described in Eq. 4.18. From the simmulations presented in Chapter 3, it is known that the NMSE, and therefore the MSE, is constant, and independent of the channel. Therefore, the MSE defined in Eq. 4.17 is adjusted manually. As mentioned previously, it was also noted that the effective channel

System parameters	
Transmitter center frequency	910.0000 MHz
Receiver center frequency	910.0037 MHz
Bandwidth	100 kHz
Dynamic range of transmitter power	50 dB
Number of subcarriers	64
- Data subcarriers	44
- Pilot subcarriers	6
- Null subcarriers	14
Tested modulation schemes	BPSK, QPSK, 8-PSK, 16-QAM, 32-SQAM, 64-QAM
Tested FEC coding schemes	None, Hamming (7,4)
Tested smoothing factors J	50, 100
Distance between transmitter and receiver	$\approx$ 3 ft.

Table 5.1: System parameters

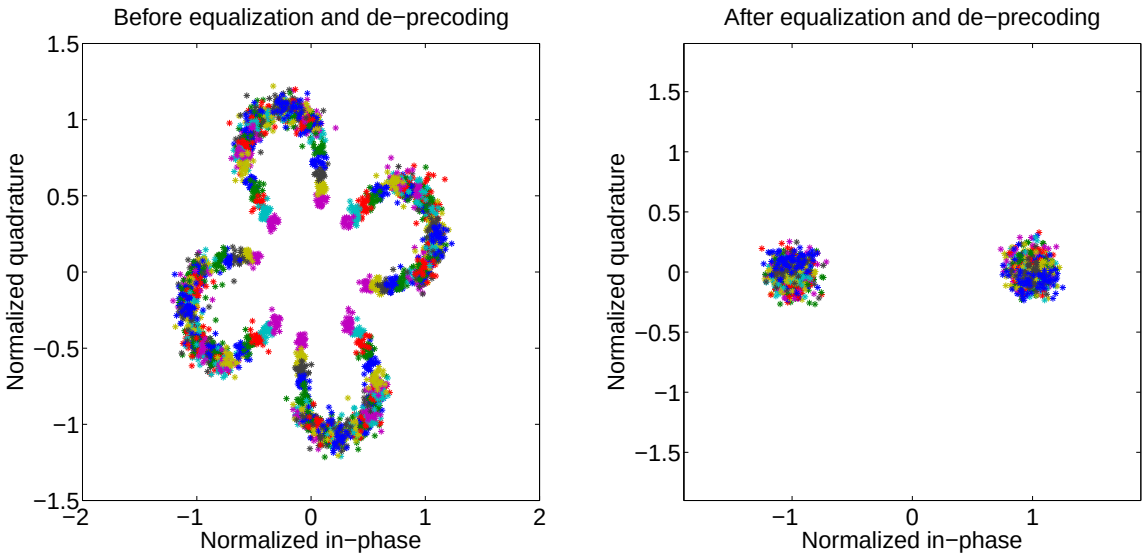


Figure 5.6: Received constellation for BPSK

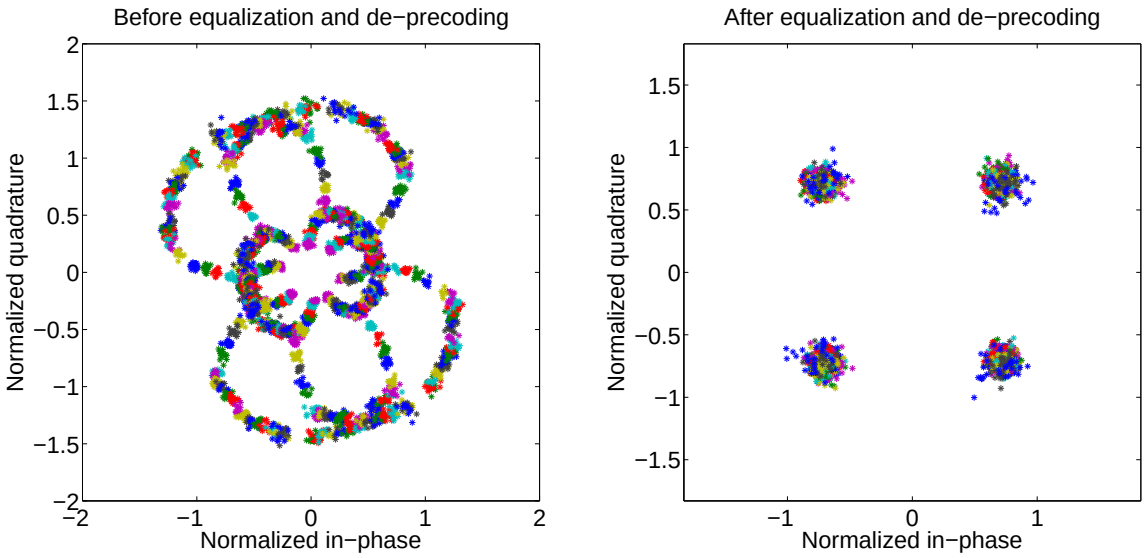


Figure 5.7: Received constellation for QPSK

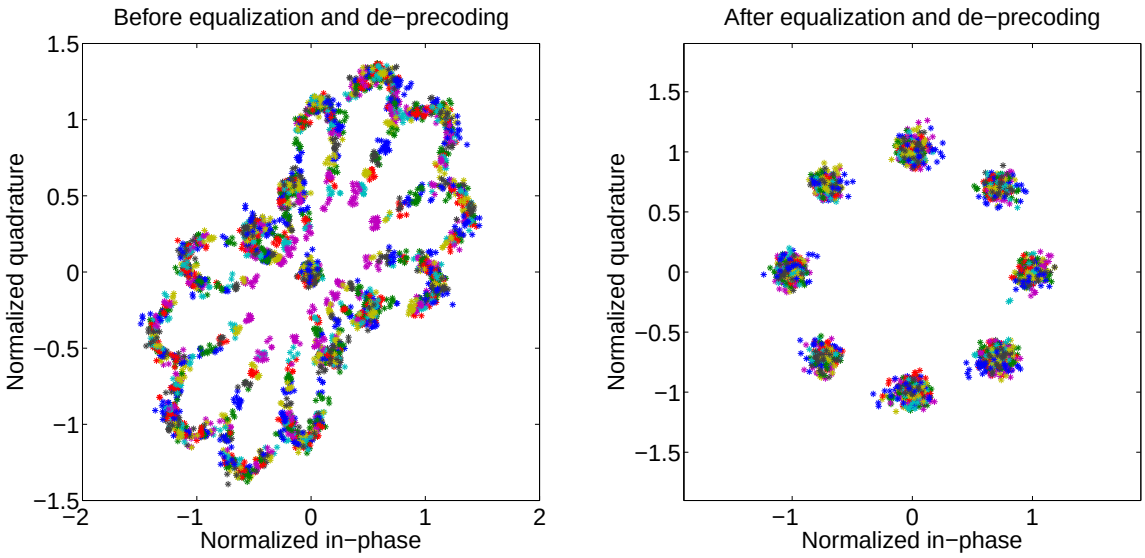


Figure 5.8: Received constellation for 8-PSK

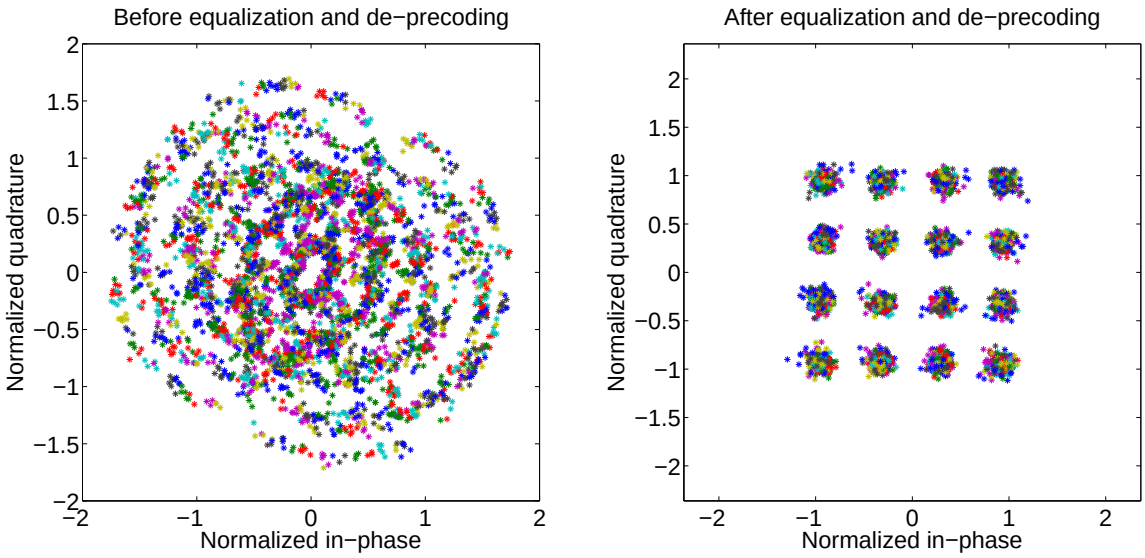


Figure 5.9: Received constellation for 16-QAM

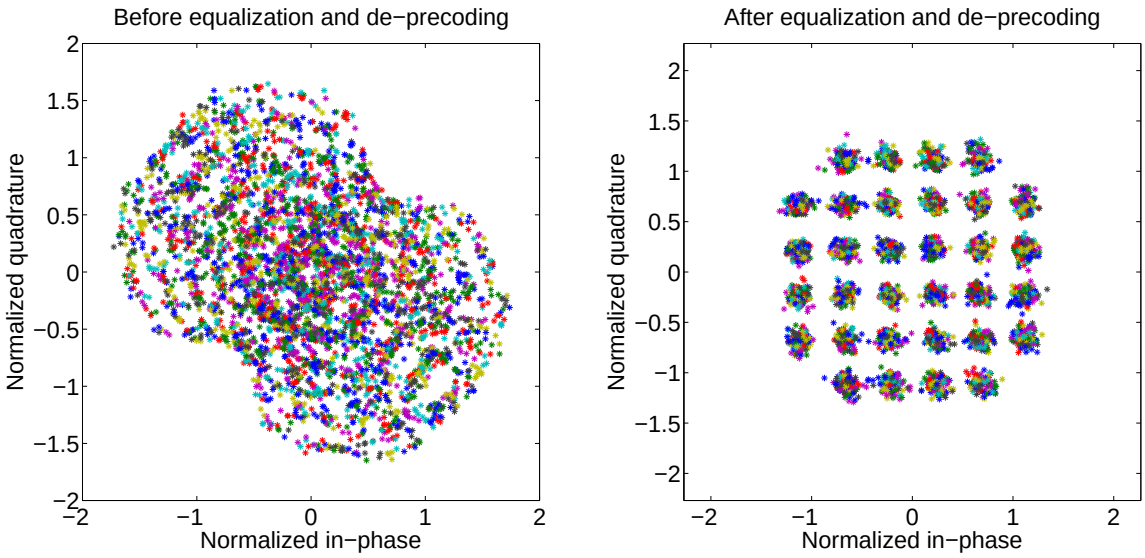


Figure 5.10: Received constellation for 32-SQAM

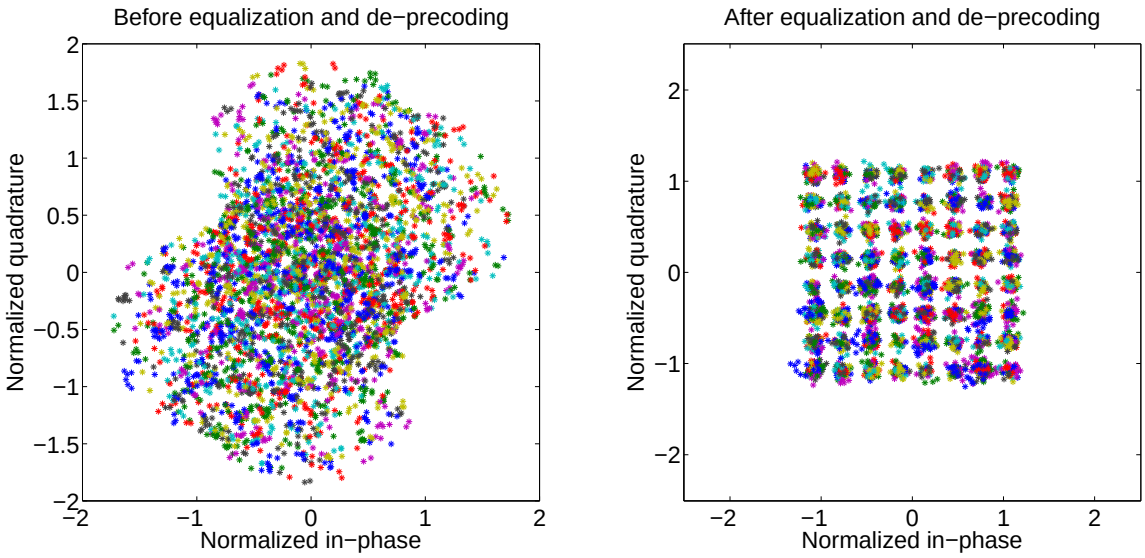


Figure 5.11: Received constellation for 64-QAM

length was $L = 3$. Thus, to adjust the MSE, the MSE($L = 1$) was found manually, and the total MSE for any value of L was then calculated following a procedure similar to that used to determine NMSE (see Eq. 3.9). The system was tested for different values of J, and it was found that, contrary to the simulation results, the system performance was virtually the same for values of $J = \{100, 200, 500\}$. Therefore, the results presented in this section are only for $J = 100$. In addition, the system was tested for $J = 50$. In both cases, the MSE was adjusted as described previously, and the resulting values were 0.013 and 0.015, respectively.

The blue curves in Figs. 5.12, 5.14 and 5.16, depict the measured BER for each of the six different modulations schemes that were tested, while the green curves show the BER estimated by the system using Eq. 4.18 to adjust the SNR in accordance with Eqs. 4.2, 4.3 and 4.4. The proportional error between the estimated and measured BER, in dB, as defined by Eq. 4.19, is shown in Figs. 5.13, 5.15 and 5.17. As seen in Fig. 5.13, the error is generally within the range of -5 dB and 5 dB, and almost always within the range of -10 dB and 10 dB. This implies that the error in the estimate is smaller than the order of magnitude of BER. In general, the predictions work better for the M-PSK schemes than for the QAM schemes. This could be due to the larger peak-to-average ratio of QAM, which could cause some clipping effects to occur.

The measured BER for a fixed SNR for each modulation scheme is ordered as expected, that is, the BER is lowest for BPSK, followed in increasing order by QPSK, 8-PSK, 16-QAM, 32-SQAM and 64-QAM. However, in Fig. 5.14, the BER curve for 16-QAM does not follow this trend after an SNR of approximately 30 dB, where it exhibits an abnormally high BER (higher than for 32-SQAM). The reason for this anomalous behavior is unclear. A possible explanation for these high BER values could be related to the combination of two phenomena: a worse channel estimate due to $J = 50$ together with the fact that the 16-QAM scheme has a higher peak-to-average ratio than 32-SQAM, so clipping would be more severe. Further simulation and over-the-air work is necessary in order to clarify this conjecture.

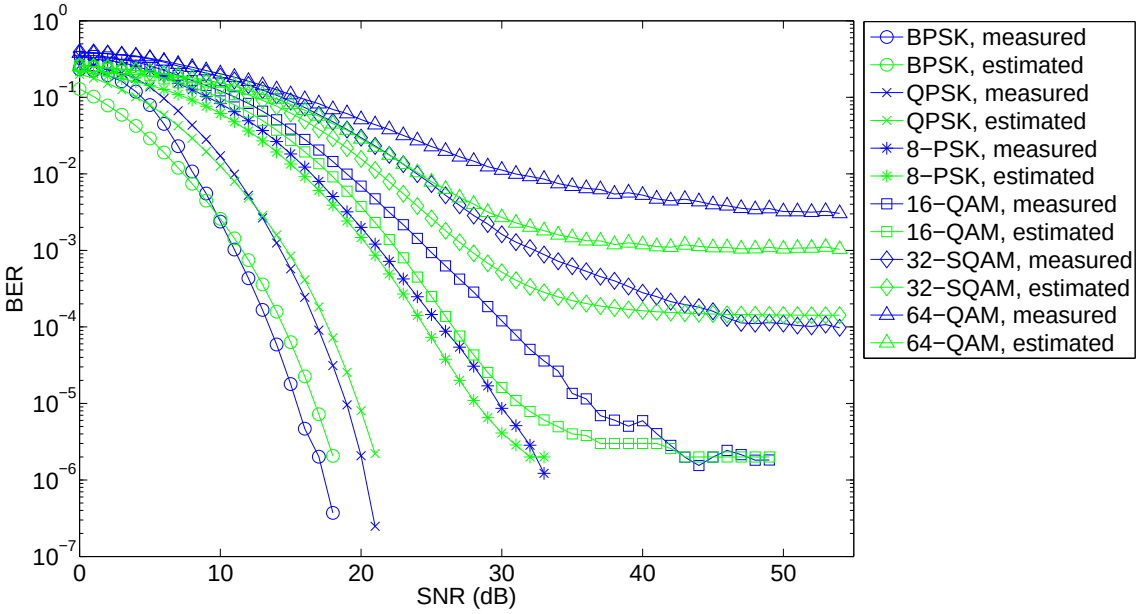


Figure 5.12: Measured and estimated BER for uncoded data, for the six tested modulation schemes and $J = 100$

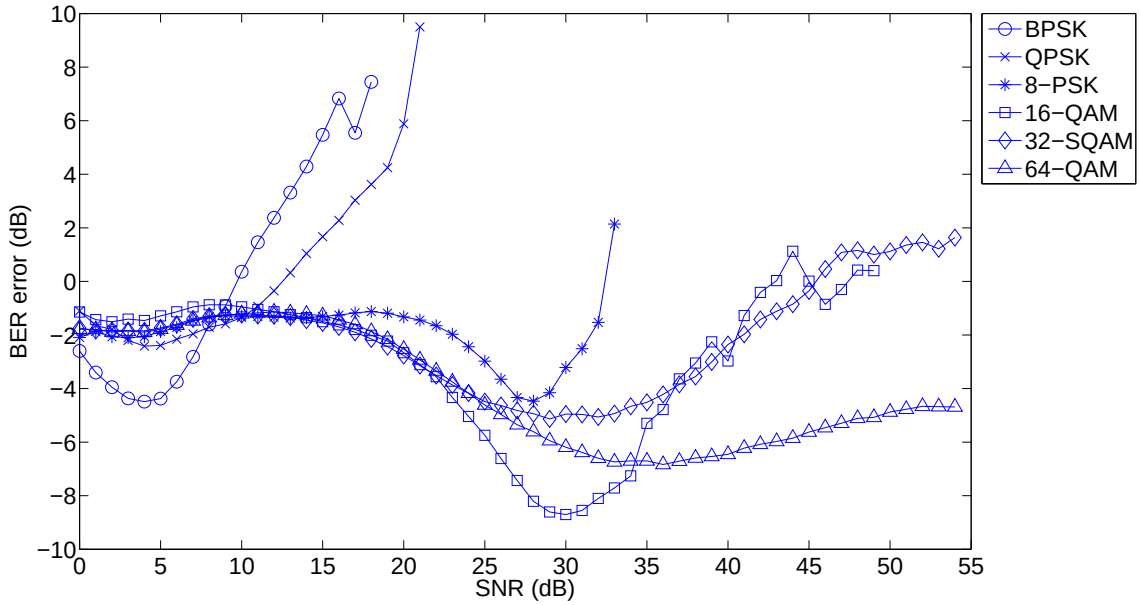


Figure 5.13: Error between the estimated and measured BER for uncoded data, for the six tested modulation schemes and $J = 100$

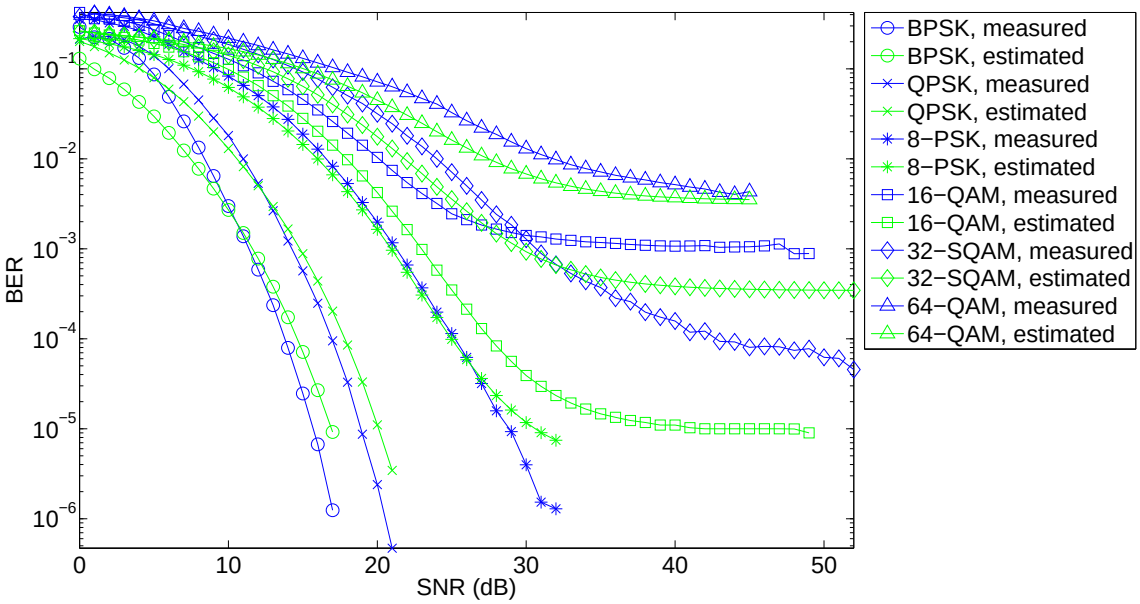


Figure 5.14: Measured and estimated BER for uncoded data, for the six tested modulation schemes and $J = 50$

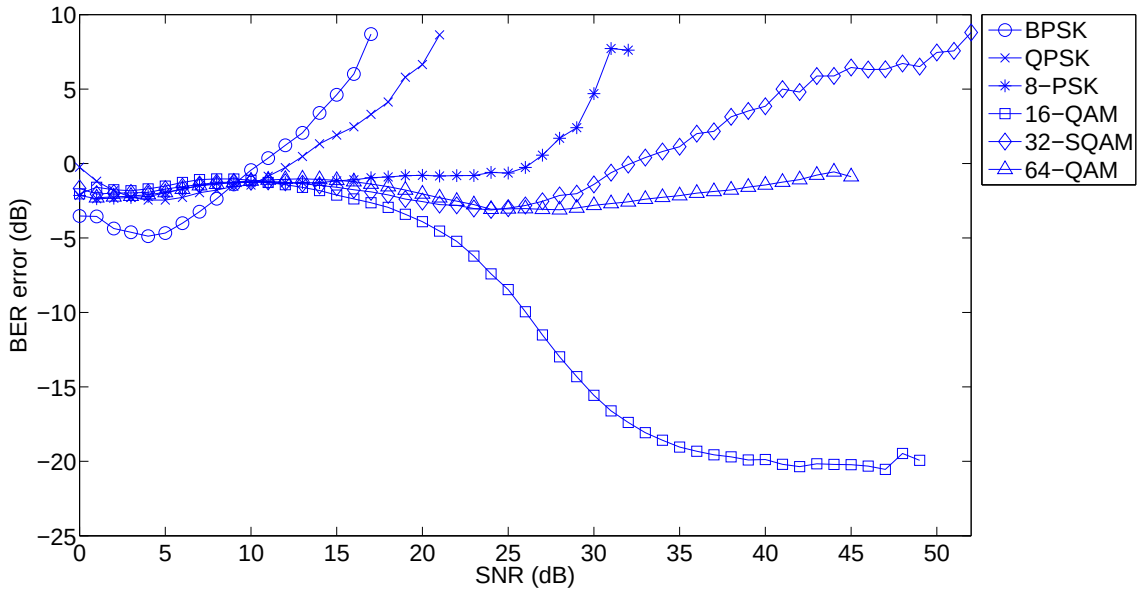


Figure 5.15: Error between the estimated and measured BER for uncoded data, for the six tested modulation schemes and $J = 50$

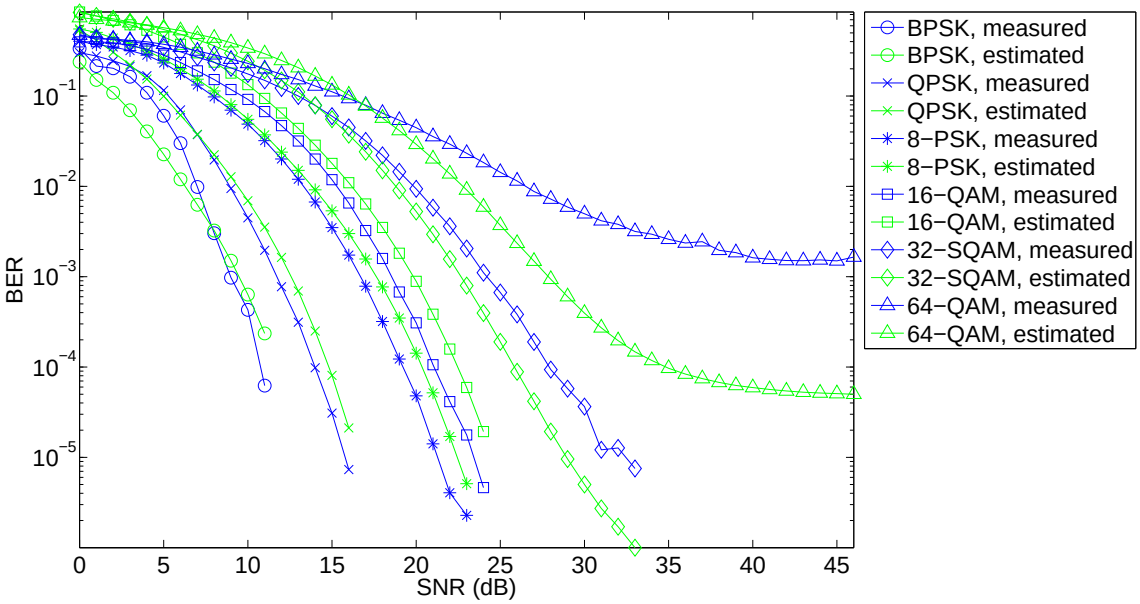


Figure 5.16: Measured and estimated BER for data that was encoded with Hamming (7,4), for the six tested modulation schemes and $J = 100$

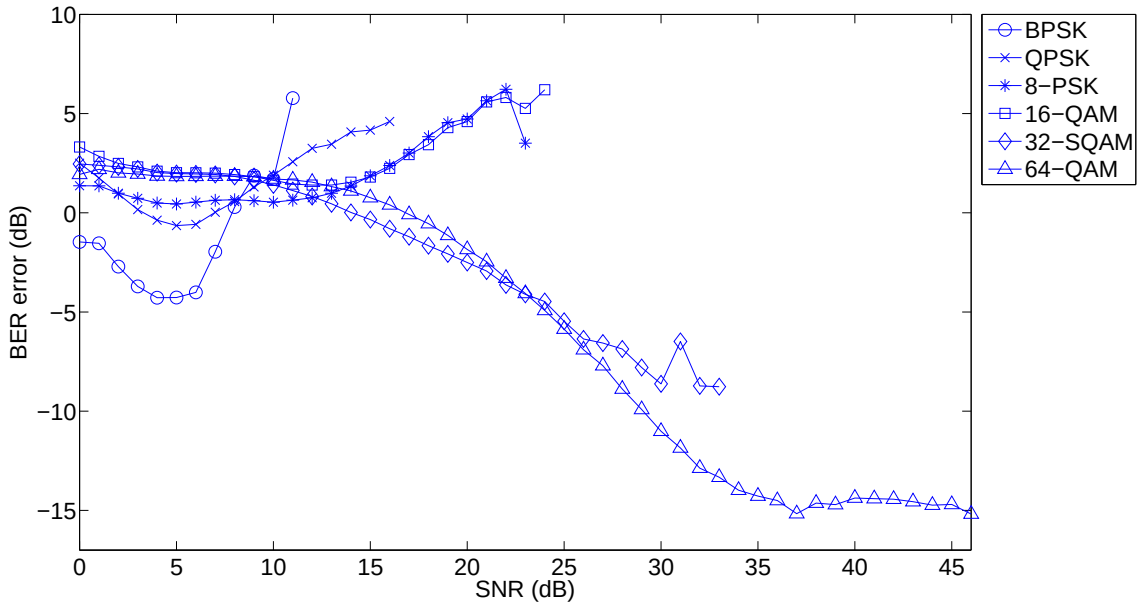


Figure 5.17: Error between the estimated and measured BER for data that was encoded with Hamming (7,4), for the six tested modulation schemes and $J = 100$

Chapter 6

Conclusions and future work

The work described herein presents a novel blind channel estimator using a statistical method that improves on the one previously introduced by Petropulu *et al.* [10]. It is a simple method that is not difficult to implement. However, it requires extra computation processing at the receiver to buffer the OFDM symbols and then to equalize them all at once when the system has a channel estimate.

In contrast with many other studies that simply run simulations, here, over-the-air tests were performed and real experimental results were obtained. The performance, measured in terms of the BER, was satisfactory and opens the way to possible improvements in the future. The equalizer was also found to work well in a real world environment.

In addition, a method has been proposed to predict how the system will perform, given that it knows the SNR and it has an imperfect estimate of the communication channel for data that is either uncoded or coded with a FEC block code, such as the Hamming (7, 4) scheme. It was found that the predictions, for BER of as low as 10^{-6}, were generallly within the same order of magnitude for all modulation schemes. These predictions could be used in combination with a Cognitive Engine to perform link adaptation, optimizing the transmission parameters controlled by the system.

While the proposed equalizer is blind, some other issues that are present in real world communications, such as the frequency/phase and timing offsets, require the presence of pilot subcarriers. To make the proposed system truly blind, however, methods to estimate the frequency/phase and timing offsets without the presence of pilots should be investigated. Other blind channel equalizers, based on either statistical or deterministic methods, could also be implemented and tested over-the-air to compare their performance.

The hardware used for the tests presented many limitations, especially in the case of an OFDM transmission, which is known to have a high peak-to-average ratio, thus the radios could have been distorting the signal due to clipping or other non linear effects. The system should be tested with better hardware, such as USRP-2 (the second generation of USRPs)

or other radios, to measure if performance is improved.

In summary, this thesis has presented an overview of OFDM specific blind channel estimators and equalizers; a blind channel estimator for OFDM was developed by improving on previous work; a method was implemented for the system to predict its own BER; and real over-the-air tests and results are presented for both the channel estimator and the system's predictions.

References

[1] K Jayanthi. *Some investigations on quality improvement using link adaptation techniques in cellular mobile networks*. PhD thesis, Pondicherry University, 2010.

[2] A. Soysal, S. Ulukus, and C. Clancy. Channel estimation and adaptive m-qam in cognitive radio links. In *Communications, 2008. ICC '08. IEEE International Conference on*, pages 4043 –4047, may 2008.

[3] S.S. Das, M.I. Rahman, Yuanye Wang, F.B. Frederiksen, and R. Prasad. Hybrid strategies for link adaptation exploiting several degrees of freedom in ofdm based broadband wireless systems. In *Vehicular Technology Conference, 2007. VTC-2007 Fall. 2007 IEEE 66th*, pages 1807 –1811, 30 2007-oct. 3 2007.

[4] Thomas W. Rondeau. *Application of Artificial Intelligence to Wireless Communications*. PhD thesis, Virginia Tech, Blacksburg, VA, 2007.

[5] H.I. Volos and R.M. Buehrer. Cognitive engine design for link adaptation: An application to multi-antenna systems. *Wireless Communications, IEEE Transactions on*, 9(9):2902 –2913, september 2010.

[6] Haris I Volos, Chris I Phelps, and R Michael Buehrer. *Initial Design of a Cognitive Engine for MIMO Systems*. Number 0520418. 2007.

[7] H.I. Volos, C.I. Phelps, and R.M. Buehrer. Physical layer cognitive engine for multi-antenna systems. In *Military Communications Conference, 2008. MILCOM 2008. IEEE*, pages 1 –7, nov. 2008.

[8] Haris I Volos and R Michael Buehrer. Robust training of a link adaptation cognitive engine. *Trials*, pages 1318–1323, 2010.

[9] David Gonzalez Fitch Ashwin Amanna, Daniel Ali and Jeffrey H. Reed. Hybrid experiential-heuristic cognitive radio engine architecture and implementation. *Journal of Computer Networks and Communications*, page 15, 2012.

[10] A. Petropulu, Ruifeng Zhang, and R. Lin. Blind ofdm channel estimation through simple linear precoding. *Wireless Communications, IEEE Transactions on*, 3(2):647 – 655, march 2004.

[11] M. Engels. *Wireless Ofdm Systems: How to Make Them Work?* The Kluwer International Series in Engineering and Computer Science. Kluwer Academic Publishers, 2002.

[12] Mahesh K Banavar Narasimhamurthy, Adarsh B and Cihan Tepedelenliolu. *OFDM Systems for Wireless Communications*. Morgan & Claypool Publishers, San Rafael, Calif. (1537 Fourth Street, San Rafael, CA 94901 USA), 2010.

[13] Gregory E. Bottomley. *Channel Equalization for Wireless Communications: From Concepts to Detailed Mathematics*. Hoboken : John Wiley & Sons, 2011.

[14] B. Muquet, M. de Courville, and P. Duhamel. Subspace-based blind and semi-blind channel estimation for ofdm systems. *Signal Processing, IEEE Transactions on*, 50(7):1699 –1712, jul 2002.

[15] Chengyang Li and S. Roy. A subspace blind channel estimation method for ofdm systems without cyclic prefix. In *Vehicular Technology Conference, 2001. VTC 2001 Fall. IEEE VTS 54th*, volume 4, pages 2148 –2152 vol.4, 2001.

[16] Qinghua Shi, Y. Karasawa, and Ying Wang. Blind channel and frequency offset estimation for ofdm via frequency-domain oversampling. In *Wireless Communications Networking and Mobile Computing (WiCOM), 2010 6th International Conference on*, pages 1 –5, sept. 2010.

[17] F.O. Alayyan, K. Abed-Meraim, and A.M. Zoubir. Blind equalization in ofdm systems exploiting guard interval redundancy. In *Signals, Systems and Computers, 2005. Conference Record of the Thirty-Ninth Asilomar Conference on*, pages 697 –700, 28 2005-nov. 1 2005.

[18] F.O. Alayyan, K. Abed-Meraim, and A.M. Zoubir. Blind equalization and frequency offset estimation in ofdm systems exploiting guard interval redundancy. In *Signal Processing and Its Applications, 2005. Proceedings of the Eighth International Symposium on*, volume 1, pages 135 – 138, 28-31, 2005.

[19] B. Muquet and M. de Courville. Blind and semi-blind channel identification methods using second order statistics for ofdm systems. In *Acoustics, Speech, and Signal Processing, 1999. Proceedings., 1999 IEEE International Conference on*, volume 5, pages 2745 –2748 vol.5, 1999.

[20] Jr. Heath, R.W. and G.B. Giannakis. Exploiting input cyclostationarity for blind channel identification in ofdm systems. *Signal Processing, IEEE Transactions on*, 47(3):848 –856, mar 1999.

[21] Zhu Wensheng, Li Youming, Zhou Xinxing, and Yu Jianding. Blind channel estimation algorithm with simplified implementation for ofdm system. In *Networks Security Wireless Communications and Trusted Computing (NSWCTC), 2010 Second International Conference on*, volume 2, pages 274 –277, april 2010.

[22] S. Ghadrdan, M. Ahmadian, S. Salari, and M. Heydarzadeh. An improved blind channel estimation algorithm for ofdm systems. In *Telecommunications (IST), 2010 5th International Symposium on*, pages 421 –425, dec. 2010.

[23] Zhou Jing, Chang Yongyu, Zhe Chen, and Dacheng Yang. Joint blind channel estimation and turbo equalization for ofdm systems. In *Vehicular Technology Conference Fall (VTC 2010-Fall), 2010 IEEE 72nd*, pages 1 –5, sept. 2010.

[24] F.J. Simois, J.J. Murillo-Fuentes, R. Boloix-Tortosa, and L. Salamanca. Near the cramer-rao bound precoding algorithms for ofdm blind channel estimation. *Vehicular Technology, IEEE Transactions on*, 61(2):651 –661, feb. 2012.

[25] S.A. Banani and R.G. Vaughan. Ofdm with iterative blind channel estimation. *Vehicular Technology, IEEE Transactions on*, 59(9):4298 –4308, nov. 2010.

[26] R. Boloix-Tortosa, F.J. Simois-Tirado, and J.J. Murillo-Fuentes. Blind adaptive channel estimation for ofdm systems. In *Signal Processing Advances in Wireless Communications, 2009. SPAWC '09. IEEE 10th Workshop on*, pages 191 –195, june 2009.

[27] Chen Wei. Fast blind channel estimation based on discrete hilbert transform in ofdm system. In *Communications and Mobile Computing (CMC), 2010 International Conference on*, volume 3, pages 21 –23, april 2010.

[28] R. Zhang and W. Chen. A mixture kalman filter approach for blind ofdm channel estimation. In *Signals, Systems and Computers, 2004. Conference Record of the Thirty-Eighth Asilomar Conference on*, volume 1, pages 350 – 354 Vol.1, nov. 2004.

[29] Shengli Zhou and G.B. Giannakis. Finite-alphabet based channel estimation for ofdm and related multicarrier systems. *Communications, IEEE Transactions on*, 49(8):1402 –1414, aug 2001.

[30] M.C. Necker and G.L. Stuber. Totally blind channel estimation for ofdm on fast varying mobile radio channels. *Wireless Communications, IEEE Transactions on*, 3(5):1514 – 1525, sept. 2004.

[31] Li Wei, Chen Ming, Shixin Cheng, and Haifeng Wang. A complexity reduced blind channel equalization scheme for ofdm systems. In *Personal, Indoor and Mobile Radio Communications, 2006 IEEE 17th International Symposium on*, pages 1 –4, sept. 2006.

[32] Seongwook Song and Andrew C. Singer. Blind ofdm channel estimation using fir constraints: Reduced complexity and identifiability. *Information Theory, IEEE Transactions on*, 53(3):1136 –1147, march 2007.

[33] H. Murakami. Deterministic blind channel estimation for a block transmission system using fractional sampling and interpolation. *Signal Processing, IEEE Transactions on*, 55(10):4969 –4978, oct. 2007.

[34] M. Ettus. Ettus research. http://www.ettus.com/, 2012.

[35] Joseph D. Gaeddert. liquid-dsp. https://github.com/jgaeddert/liquid-dsp, June 2012.

[36] Joseph D. Gaeddert. liquid-usrp. https://github.com/jgaeddert/liquid-usrp, June 2012.

[37] Bhagwandas Pannalal Lathi. *Modern Digital and Analog Communication Systems*. Oxford University Press, 3rd ed. edition, 1998.

[38] A. Leke and J.M. Cioffi. Impact of imperfect channel knowledge on the performance of multicarrier systems. In *Global Telecommunications Conference, 1998. GLOBECOM 1998. The Bridge to Global Integration. IEEE*, volume 2, pages 951 –955 vol.2, 1998.

Appendix A

Simulation plots

This appendix contains some extra figures from the simulations that were described in chapter 3.

A.1 Received signal constellations

Figures A.1 through A.12, shown in this section, represent the received signal's constellation, for all six tested modulation schemes (BPSK, QPSK, 8-PSK, 16-QAM, 32-SQAM and 64-QAM), for two simulated channels of length three and six, where the taps were generated randomly for an SNR of 40 dB. The shape of the received constellation before the equalizing and de-precoding is due to both the channel effect and the precoding scheme.

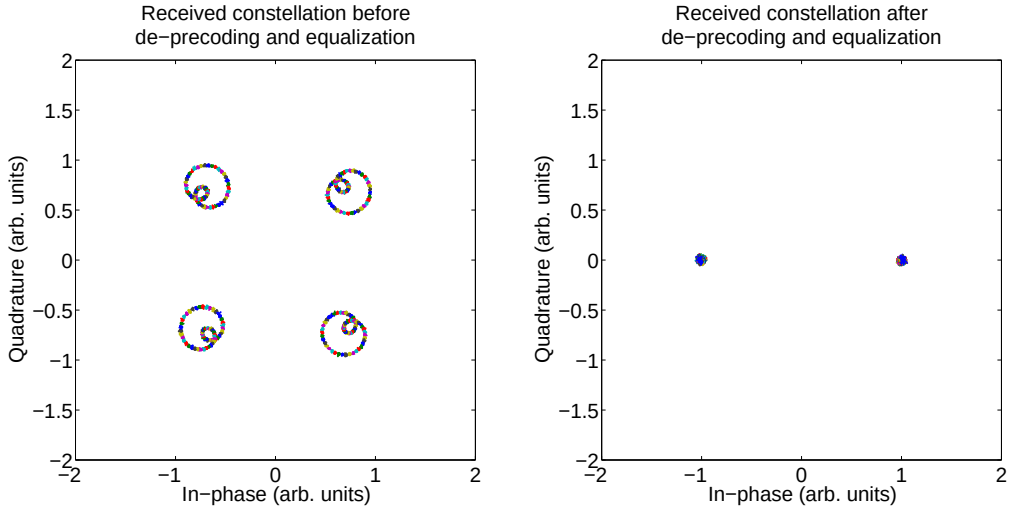


Figure A.1: Received signal constellation before and after equalizing and de-precoding for BPSK modulation for a channel generated randomly of length $L = 3$

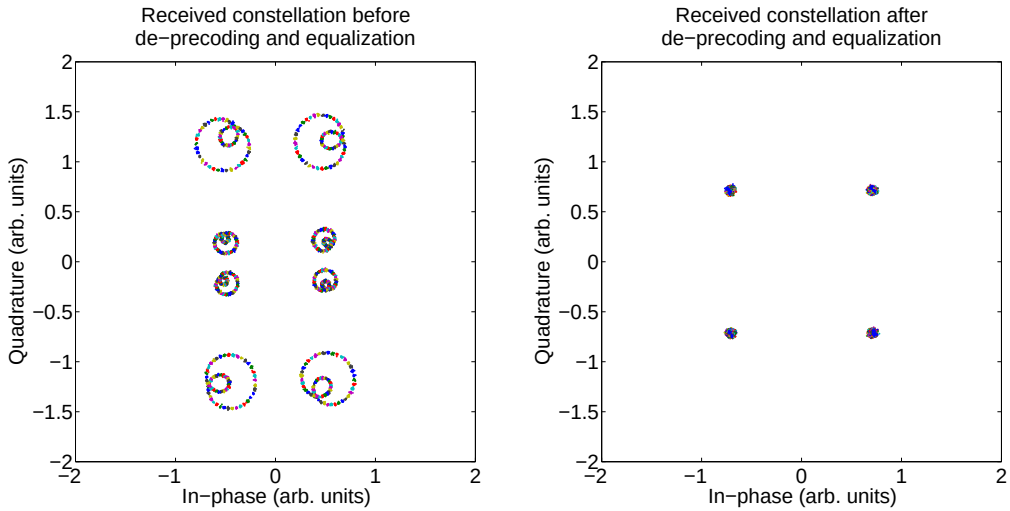


Figure A.2: Received signal constellation before and after equalizing and de-precoding for QPSK modulation for a channel generated randomly of length $L = 3$

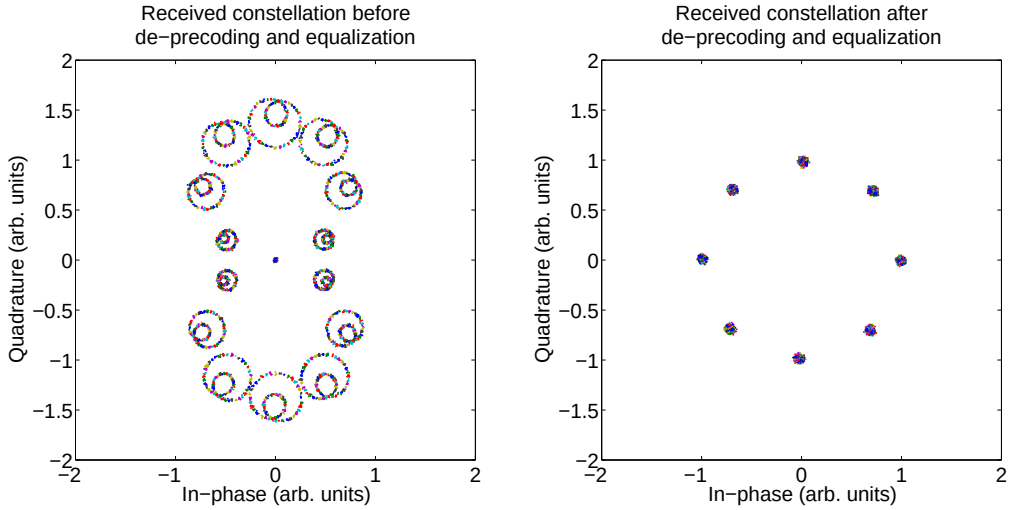


Figure A.3: Received signal constellation before and after equalizing and de-precoding for 8-PSK modulation for a channel generated randomly of length $L = 3$

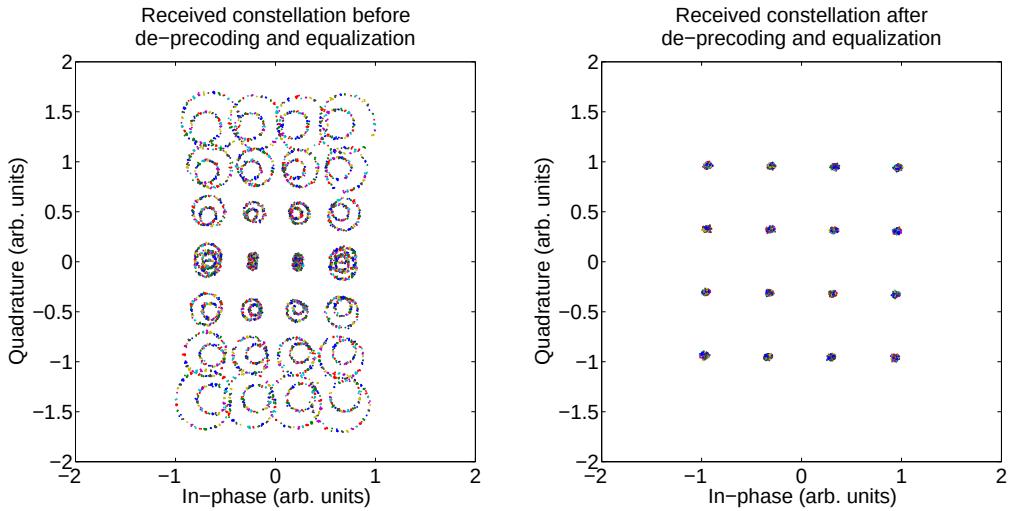


Figure A.4: Received signal constellation before and after equalizing and de-precoding for 16-QAM modulation for a channel generated randomly of length $L = 3$

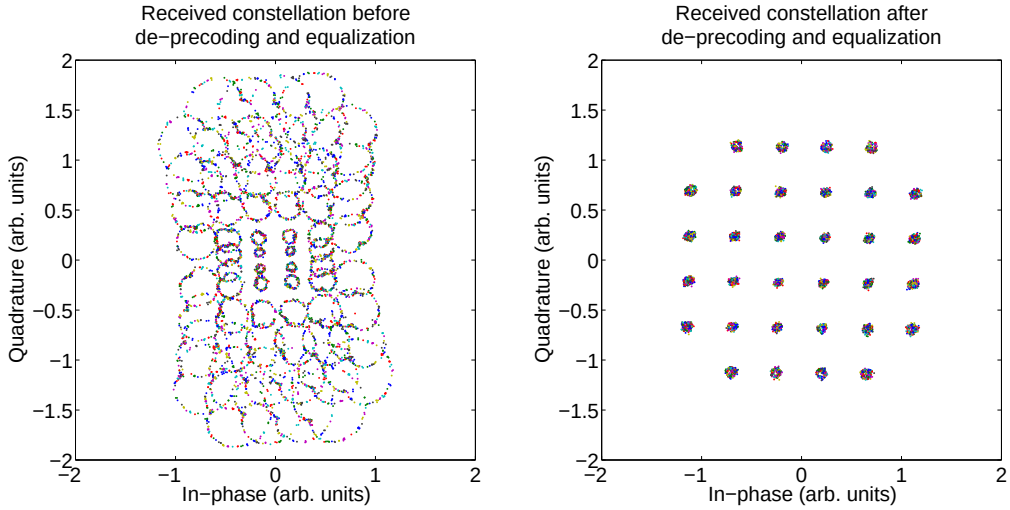


Figure A.5: Received signal constellation before and after equalizing and de-precoding for 32-SQAM modulation for a channel generated randomly of length $L = 3$

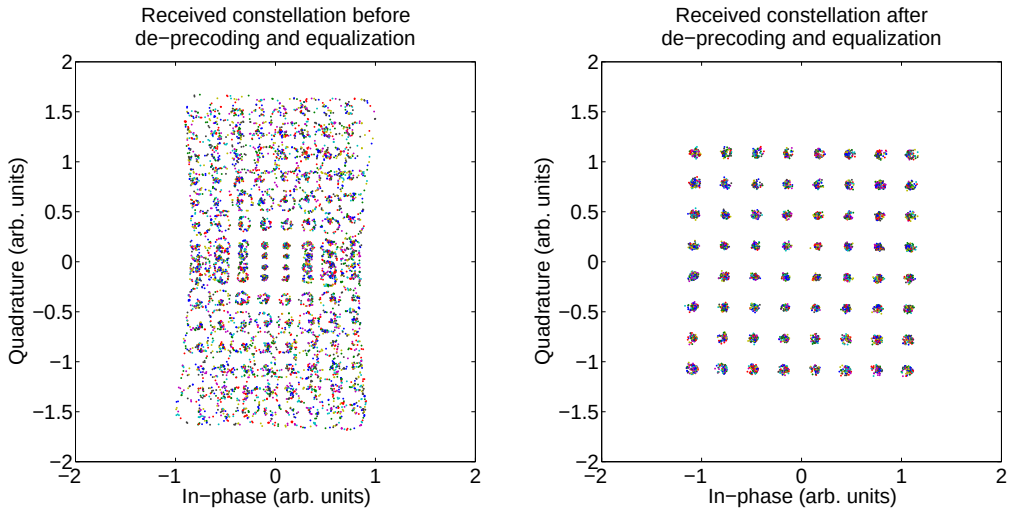


Figure A.6: Received signal constellation before and after equalizing and de-precoding for 64-QAM modulation for a channel generated randomly of length $L = 3$

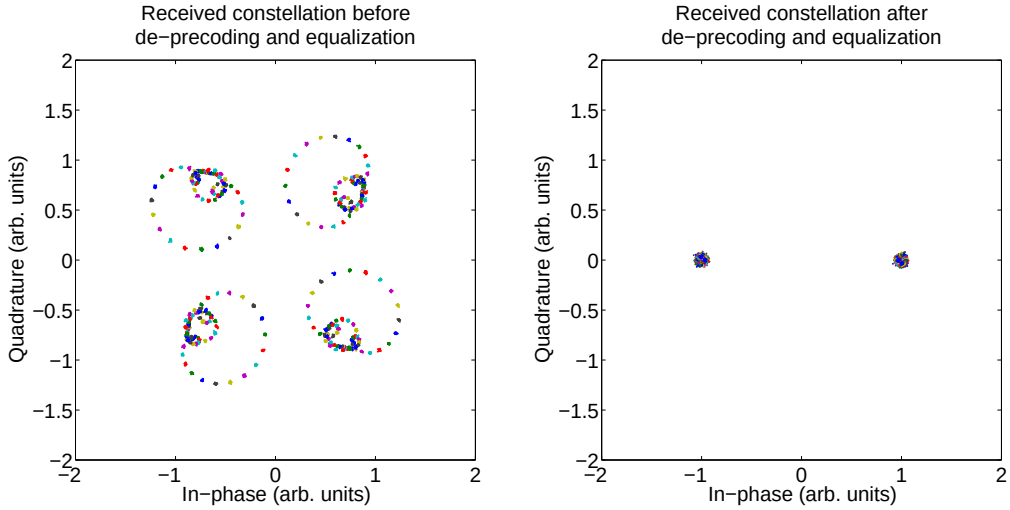


Figure A.7: Received signal constellation before and after equalizing and de-precoding for BPSK modulation for a channel generated randomly of length $L = 6$

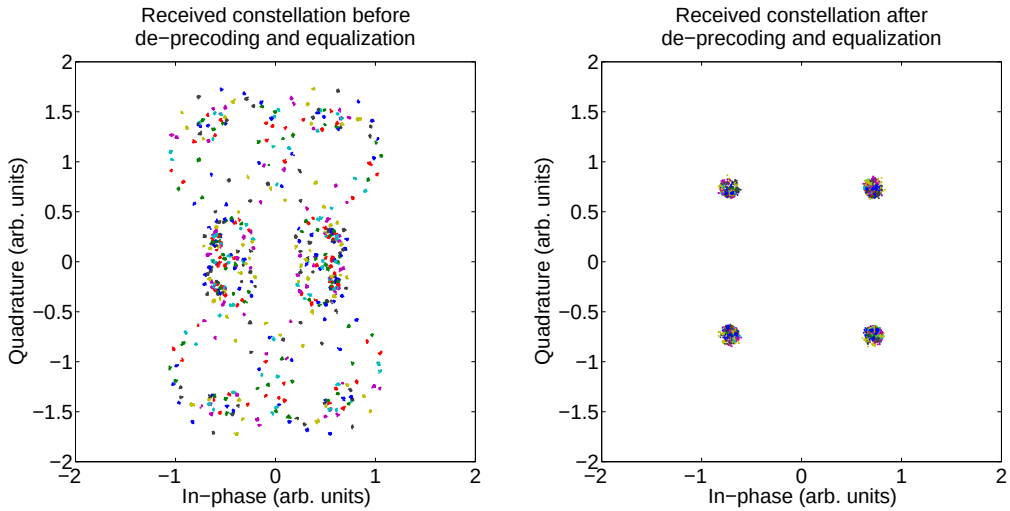


Figure A.8: Received signal constellation before and after equalizing and de-precoding for QPSK modulation for a channel generated randomly of length $L = 6$

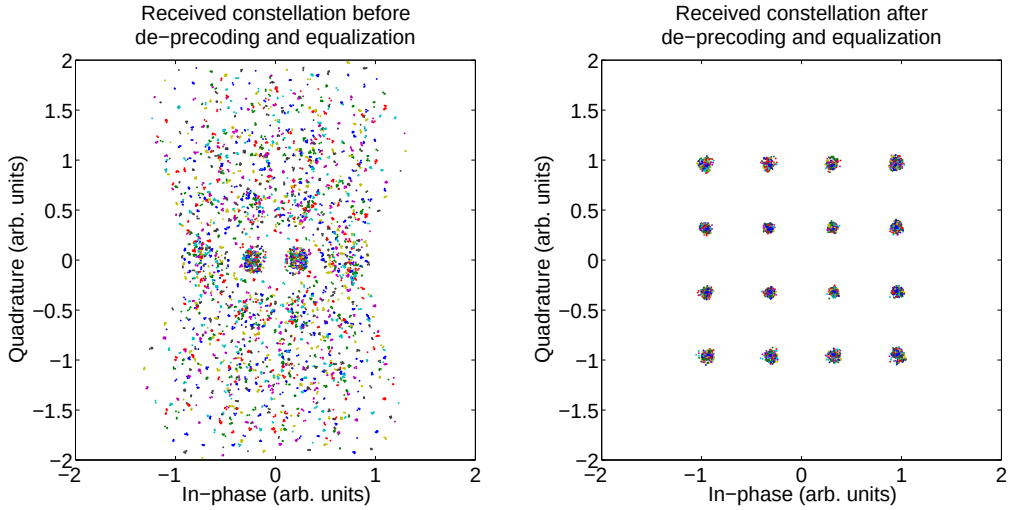


Figure A.9: Received signal constellation before and after equalizing and de-precoding for 8-PSK modulation for a channel generated randomly of length $L = 6$

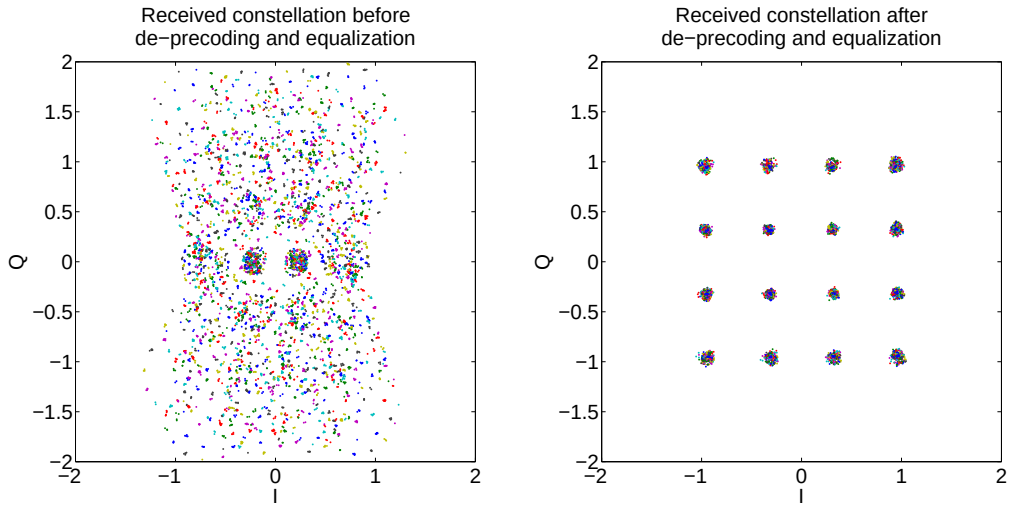


Figure A.10: Received signal constellation before and after equalizing and de-precoding for 16-QAM modulation for a channel generated randomly of length $L = 6$

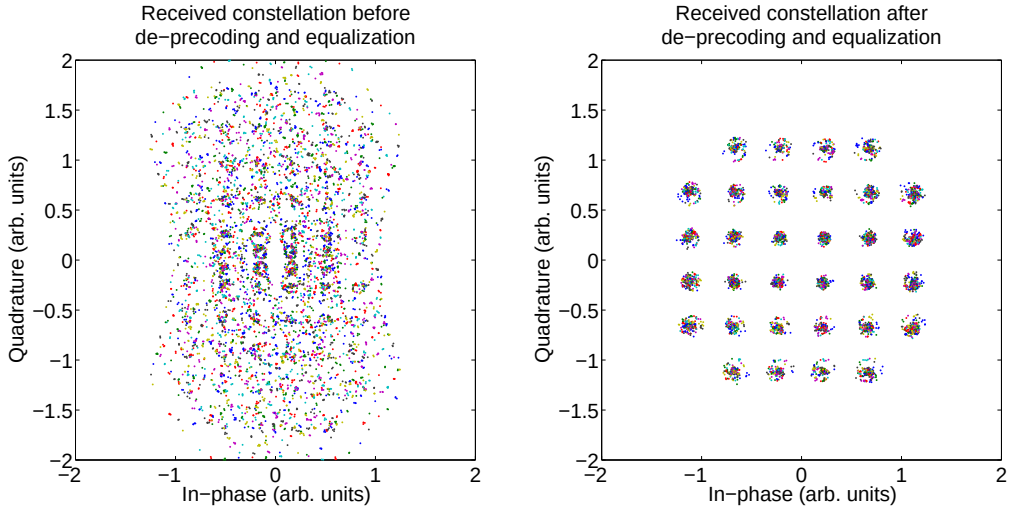


Figure A.11: Received signal constellation before and after equalizing and de-precoding for 32-SQAM modulation for a channel generated randomly of length $L = 6$

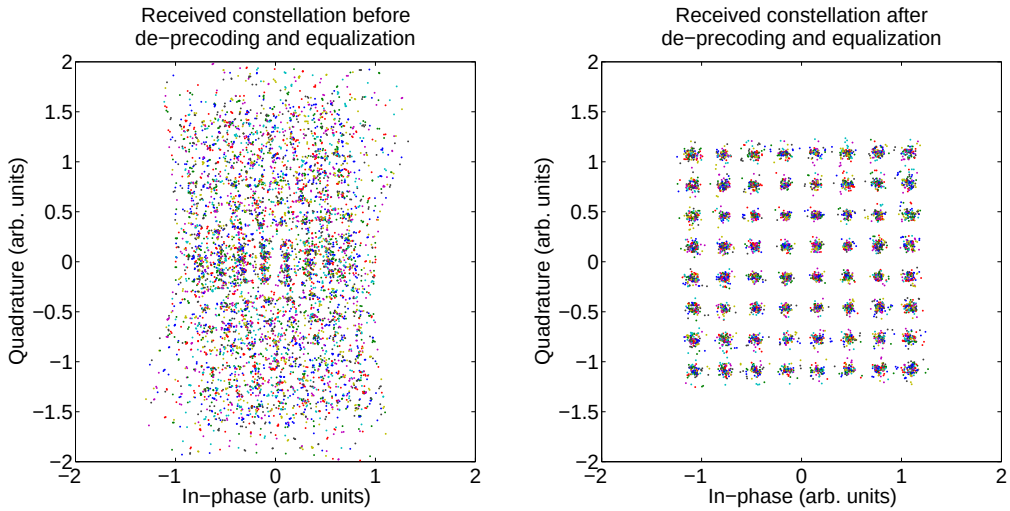


Figure A.12: Received signal constellation before and after equalizing and de-precoding for 64-QAM modulation for a channel generated randomly of length $L = 6$

A.2 Channel estimates

Figures A.13 through A.16 represent examples of the estimated channel for channel lengths L of three and six, and for two different smoothing factors J of 100 and 200, where the red curve is the initial estimate, the blue curve is the denoised estimate and the green curve is the original simulated channel.

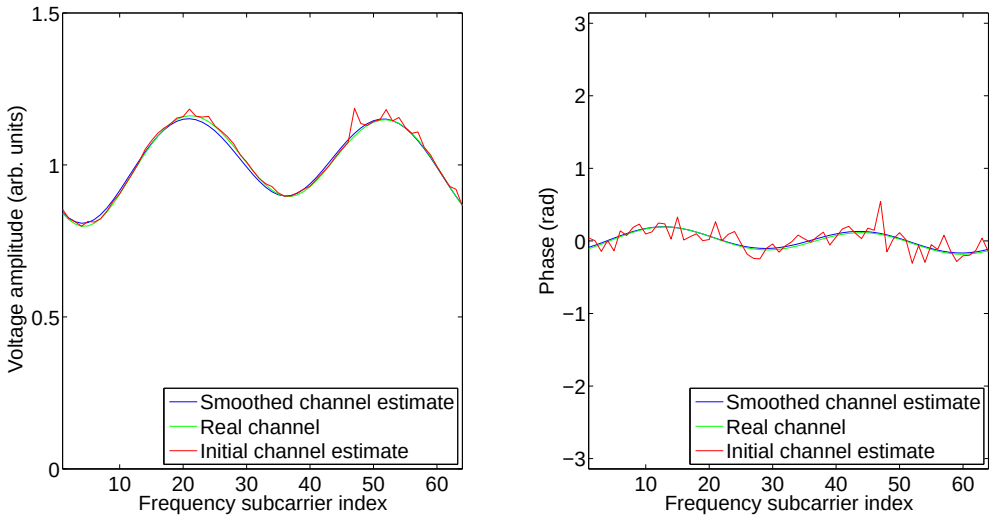


Figure A.13: Amplitude and phase of an estimated channel of length $L = 3$ and smoothing factor $J = 100$

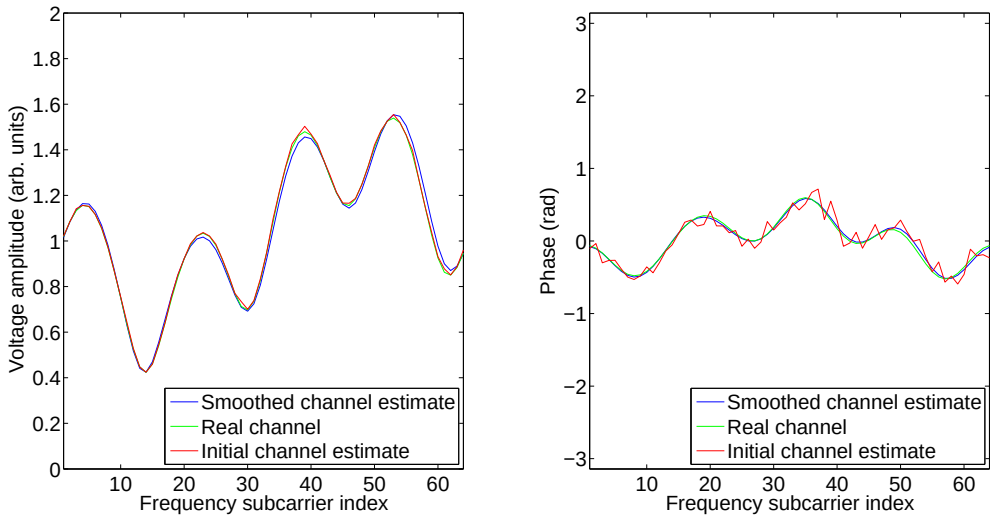


Figure A.14: Amplitude and phase of an estimated channel of length $L = 6$ and smoothing factor $J = 100$

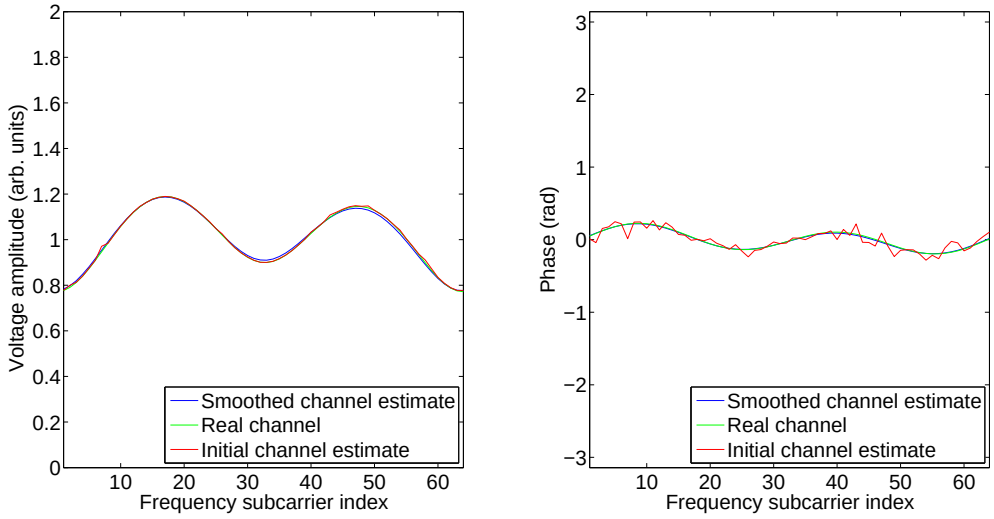


Figure A.15: Amplitude and phase of an estimated channel of length $L = 3$ and smoothing factor $J = 200$

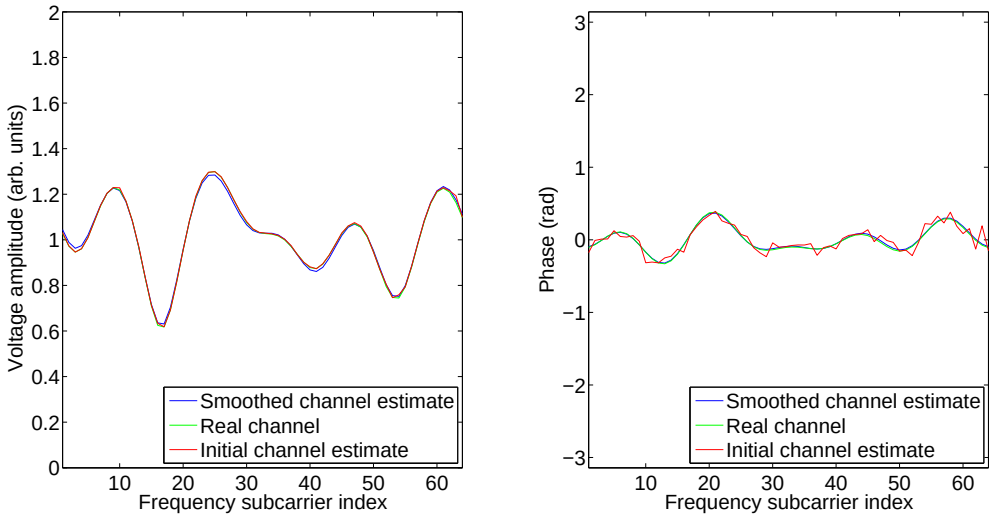


Figure A.16: Amplitude and phase of an estimated channel of length $L = 6$ and smoothing factor $J = 200$