THESIS FOR THE DEGREE OF LICENTIATE OF PHILOSOPHY

On Wavelets for Digital Communication

ANDERS SILANDER

CHALMERS | GÖTEBORG UNIVERSITY

Department of Mathematics
CHALMERS UNIVERSITY OF TECHNOLOGY
AND GÖTEBORG UNIVERSITY
Göteborg, Sweden 1999

On Wavelets for Digital Communication
ANDERS SILANDER

Preprint no 2000:16
ISSN 0347-2809
Department of Mathematics
Chalmers University of Technology and Göteborg University
SE-412 96 Göteborg
Sweden
Telephone + 46(0)31-7721000

Matematiskt Centrum
Göteborg, Sweden 2000

Abstract

We treat the subject of using wavelets in digital communication. The transmitted signal in a digital communication system can be constructed as a linear combination of scaling functions, wavelets or wavelet packets. These have the property that they are orthogonal to each other and orthogonal to unit translates, in $L^2(\mathbb{R})$. In most examples presented in the report the Meyer wavelet is used.

First, we consider communication in an additive white noise channel. The subject of mapping a binary sequence to the transmitted linear combination of wavelets (or wavelet packets) is treated here. Also the inverse mapping from a received signal to a binary sequence, which is done by the receiver, is discussed. The error probability is calculated for this type of channel.

Second, we discuss other types of channels, such as additive coloured noise- and fading channels. An erroneous receiver in the way that its frequency is not tuned correctly is also considered.

Third, we discuss some implementation aspects, such as sampling the received signal and the use of discrete wavelet transform in the transmitter and the receiver.

Fourth, an estimate of the numerically calculated Meyer scaling function is done. We also discuss the numerical calculation of the Meyer wavelet packets used in the Matlab toolbox, version 1.2.

Keywords: wavelets, digital communication, modulation

AMS 2000 Mathematics subject classification: 42C40, 94A11

Preface

This report

This licentiate thesis is written to conclude a five semester ECMI (European Consortium for Mathematics in Industry) programme in applied mathematics. This programme includes a block of core courses covering several areas of applied mathematics and computing science and a block of specialisation courses within a selected field. The final part of the programme is to work with a mathematical problem that emanates from the industry.

Acknowledgements

I wish to express my gratitude to my advisor Jöran Bergh, at the Department of Mathematics at Chalmers. He has given me many suggestions on the report and urging me forward with the work.

I would also like to thank Krister Boman and Göran Lidström at Ericsson Microwave Systems AB, for our discussions and meetings and for introducing me to the subject.

Contents

List of acronyms

AWGN	Additive White Gaussian Noise
BE	Bandwidth Efficiency
BER	Bit Error Rate
BW	BandWidth
DFT	Discrete Fourier Transform
DWT	Discrete Wavelet Transform
FSK	Frequency Shift Keying
ICI	Inter Channel Interference
IDFT	Inverse Discrete Fourier Transform
ISI	Inter Symbol Interference
IDWT	Inverse Discrete Wavelet Transform
MRA	Multi Resolution Analysis
OFDM	Orthogonal Frequency Division Multiplexing
PSK	Phase Shift Keying
QAM	Quadrature Amplitude Modulation
SER	Symbol Error Rate
SNR	Signal to Noise Ratio
TDMA	Time Division Multiple Access

Chapter 1

Introduction to digital communication

Communication is concerned about transmitting information from a source to a recipient. Digital communication means that the information is in digital form, which can be represented as a binary sequence. The picture below shows a block diagram of a digital communication system.

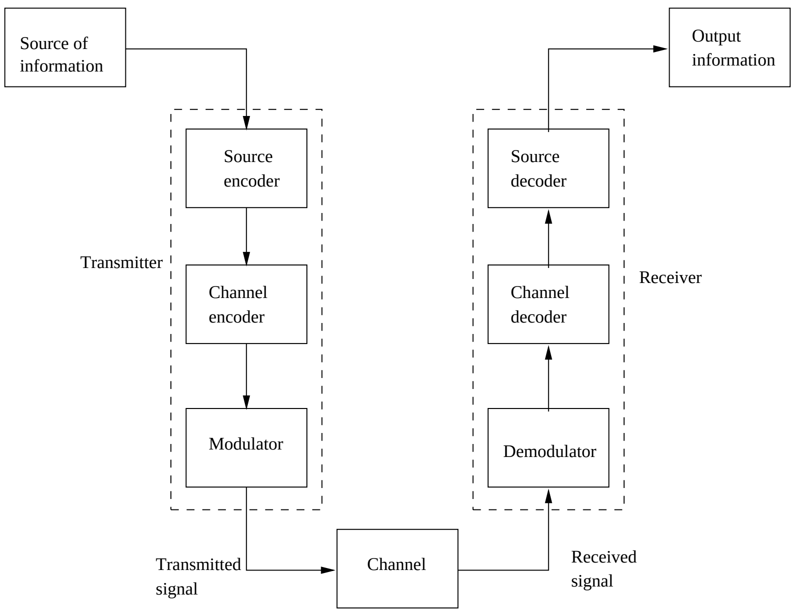

Figure 1.1: Digital communication system

The source of information could be in analog form, so the task for the *source encoder* is to convert the information into a binary sequence. This could be done for example by sampling the signal or by representing the signal by some wavelet coefficients.

The sequence is passed to the *channel encoder*, which groups the sequence into blocks and adds some redundancy and/or training bits to each block. A training sequence is a short part of the block which is known by the receiver and can be used to estimate the channel conditions and can also be used for time synchronisation. By adding bits to a block the receiver is able to correct some misinterpreted bits. The simplest such redundancy is for each bit add two redundant bits, e.g $0 \mapsto 000$ and $1 \mapsto 111$. If the receiver decodes the sequence 001, it should be interpreted as 000 or 111 since 001 is never transmitted. Either the first two bits are wrong or the last bit is wrong. The probability that one bit is wrong is much higher so the receiver corrects it to 000. A more sophisticated error correction takes k information bits and maps them to a n-bit sequence called a code word. k/n is called the code rate.

The next instance is the *digital modulator* which converts the binary sequence into an electric signal which can be transmitted over the channel. For example one could use two frequencies ω_0 and ω_1, coding a zero as $\cos\omega_0 t$ and one as $\cos\omega_1 t$ for some duration T. If $\omega_k = 2\pi n_k/T,\ n_k \in \mathbb{N},\ n_0 \neq n_1$ the two signals will be orthogonal in $L^2[0,T]$. This technique exists and is called Frequency Shift Keying (FSK). More generally the encoder uses a set of signals $S = \{f_0, \dots, f_{n-1}\}$ and is then able to encode $m = \log_2 n$ bits at a time. The group of these m bits which are encoded together is called a *symbol*. The duration T is the time it takes to transmit one symbol and $1/T$ is called the symbol rate and m/T is called the bit rate, i.e *bits/s*. The main object of this report is to consider choices of these sets of signals. The set should be chosen in such a way that the probability for the receiver to make a wrong interpretation is low. The signals often has to be band limited. A high bit rate is also desirable. In the FSK example above we had a set with two signals and were able to encode one bit. This is called *binary modulation*. If we enlarge the set S from $\{\cos\omega_0 t, \cos\omega_1 t\}$ to $S = \{\pm\cos\omega_0 t, \pm\cos\omega_1 t\}$ we get $\mathrm{card}(S) = 4$. We are now able to encode 2 bits into a symbol. If $m \geq 2$ we talk about 2^m-ary modulation, where $2^m = \mathrm{card}(S)$. More generally we start with a d-dimensional subspace $Y = \mathrm{span}\{g_0, \dots, g_{d-1}\}$ of $L^2(\mathbb{R})$. Then we choose the set S as n elements in the unit ball in Y. Often they are chosen on the unit sphere, which means that full energy is used for each symbol.

The modulated signal is then passed to the *channel*, which in the case of a microwave link channel is transmitting antenna, receiving antenna and air, ground and other obstacles between them. In telephony the channel would be the copper wire or an optical fibre. The channel will distort the signal, and the channel cause of bit error in the receiver. Other causes might be filters, non linear amplifiers etc. For example the ground and obstacles might reflect the signal (*multi-path propagation, fading*). Other transmitters might be nearby interfering with our transmission (*coloured noise*) and thermal noise is added in the receiver (*white noise*). A frequently used model of the channel is the Additive White Gaussian Noise-channel (*AWGN-channel*), which is a channel that adds a white noise term to the transmitted signal.

The next block in the figure is the *demodulator*. This is basically a hypothesis test. What is the probability that the current signal is received given that s_i was transmitted? If we have a set of two signals $S = \{s_0, s_1\}$ and receive the signal r, we interpret the bit as a zero if $P(r = s_0 + n) > P(r = s_1 + n)$, where n is the noise term. Since r might not be in S or Y but an unknown element in L^2_{loc} it is easier if we make use of a set of functionals, Λ_i, in the dual space Y'. In other words we can do an orthogonal projection of r on Y. For example in the case S contains two signals and Y is two dimensional, we create $R \subset Y'$ containing two functionals and interpret a zero if $\Lambda_0(r) > \Lambda_1(r)$. The choice of R is made in such a way that the probability of bit error is minimised.

The binary sequence is passed on to the *channel decoder*. If error correction is used, that is if we added redundant bits to the original binary sequence the decoder will make use of the redundancy to correct possible errors. In other words it will determine the code word closest to an n-bit sequence and do the inverse mapping of that code word taking n bits and map them to k bits.

The final part is the *source decoder* which tries to construct the original signal, given the sequence from the channel decoder.

1.1 Modulation and demodulation

Given a set of channels we want to construct the set S of signals and the receiver. The set of channels means that we don't just want to construct signals for one channel, but for many types of channels, which are within some specified limits. A simple but often used channel is the channel which just add white noise to the transmitted signal. This channel is called Additive White Gaussian Noise channel (AWGN). This channel can be visualised as in figure 1.2, where the output y equals the input x plus the noise n.

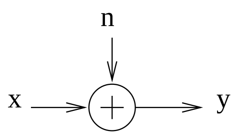

Figure 1.2: AWGN-channel

In the later chapters the analysis is in baseband. What is the connection between baseband and the actually transmitted signal in passband? Suppose we have a waveform φ that we want to use for modulating the signal, where $\varphi : \mathbb{R} \to \mathbb{R}$ or $\mathbb{C}$. Consider the vector space $Y = \{\alpha\varphi : \alpha \in \mathbb{R} \text{ or } \mathbb{C}\}$, where

Y is a subspace of the Hilbert space L^2 containing signals with finite energy. A finite number of elements in Y composes the signal set S. The number of elements should be a dyadic number, say 2^m. In this way we can find a bijective map from a binary sequence of length m to the set S. Figure 1.3 shows an example of a map from a binary sequence of length two onto the set S. In this case S contains elements in a complex vector space. The mappings will be discussed more in later chapters.

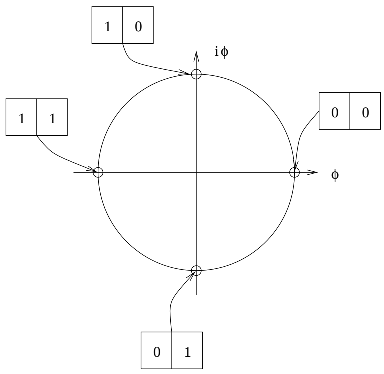

Figure 1.3: Map from a two-bit binary sequence onto the set S

The baseband transmitted signal would be $s^b(t) = c\varphi(t)$, where $c\varphi$ is one of the elements in the set S. The passband signal, using the modulation frequency ω_c, would be $s^p(t) = \text{Re }(c\varphi(t)\sqrt{2}e^{i\omega_c t})$. $\hat{\varphi}$ is assumed to be localised around zero and $\hat{\varphi}(\omega_c) = 0$. If Y is a complex vector space and $c = a + ib$ this is $s^p(t) = \varphi(t)(a\sqrt{2}\cos\omega_c t - b\sqrt{2}\cos\omega_c t)$, assuming that φ is a real function. If Y is a real vector space, this is $s^p(t) = \varphi(t)c\sqrt{2}\cos\omega_c t$. In the real case the receiver multiplies the signal with $\sqrt{2}\cos\omega_c t$. The resulting signal is $c\varphi(t)2\cos^2\omega_c t = c\varphi(t)(1 + \cos 2\omega_c t)$. Using a low pass filter on this the resulting baseband signal is $r^b(t) = c\varphi(t)$. In the complex case the receiver multiplies the signal with $\sqrt{2}e^{-i\omega_c t})$. The resulting signal is $\varphi(t)(a+ib+(a-ib)\cos 2\omega_c t-(b+ia)\sin 2\omega_c t)$. Again the signal could be low pass filtered resulting in $r^b = (a+ib)\varphi(t)$, which was the transmitted signal. If noise is added by the channel, φ would be exchanged for $\varphi + n$, where n is the noise, i.e $r^b = (a+ib)\varphi(t)+n(t)$. Thus the analysis could equally well be done on the base band signal s^b, which hereafter will be referred to as just s, without superscript. Let φ be a unit vector, the receiver can calculate $\langle r, \varphi\rangle = c$ in the complex case. The inverse mapping, i.e. the mapping from S to a binary sequence in done by the channel decoder. If the received signal contains noise, prior to the inverse mapping , we first need to determine the element in S closest to the the received element. We can also see that if we are only interested in the result of the scalar product, $\langle r, \varphi\rangle$, then we don't need to low pass filter when the support of $\hat{\varphi}$ is sufficiently small.

The next m-bit sequence is mapped in the same way, but with $\varphi(t)$ exchanged for $\varphi(t-1)$ and so on. The received total signal is then $\sum_j c_j\varphi(t-j)$. From this signal we need to determine the coefficients c_k, in order to perform the inverse mapping. If φ is orthogonal to its unit translates, we can calculate c_k from, $c_k = \langle \sum_j c_j\varphi(\cdot - j), \varphi(\cdot - k)\rangle$. The energy of the signal $E = \|s\|^2$ is one in the text above, also the symbol rate is one. This can of course be changed by $s(t) \mapsto \sqrt{E}\frac{1}{\sqrt{T_s}}s(\frac{t}{T_s})$, where T_s is the symbol rate.

In the text above, the space Y was one dimensional. The space can be enlarged to a n dimensional vector space $Y = \text{span}\{g_0, \dots, g_{n-1}\}$ over $\mathbb{R}$ or $\mathbb{C}$. As we will see it is desirable that the functions in Y are orthogonal to each other and as before we would also like the functions to be orthogonal to unit translates. In other words we would like $\langle g_i(\cdot - k), g_j(\cdot - l)\rangle = \delta_{i,j}\delta_{k,l}$. In this case we can write $Y = Y_0 \oplus \ldots \oplus Y_{n-1}$, where $Y_j = \{b_j g_j : b_j \in \mathbb{R}\}$. The signal set is $S = S_0 + \ldots + S_{n-1}$, where S_j contains a finite number of elements in the space Y_j. Writing it this way we see that the signal set is adapted to multi-user communications, by assigning each user to one or a couple of the S_js.

Before studying wavelet transmission, let us mention some common types of modulation. We are using the same notations as in the text above.

Phase Shift Keying (PSK), here one uses the space $Y = \text{span}\{\varphi\}$, where Y is vector space over $\mathbb{C}$. The basis function φ is the Haar scaling function, i.e $\varphi = \chi_{[0,1)}$.

Another common type is Frequency Shift Keying (FSK). In this case Y is a n dimensional vector space over $\mathbb{R}$ and $Y = \text{span}\{g_0, \dots, g_{n-1}\}$, where $g_j(t) = \chi_{[0,1)}e^{iq_jt}$ and $\{q_j\}$ are chosen so that the g_j are orthogonal to each other.

The next type of modulation is Orthogonal Frequency Division Multiplexing (OFDM). This is FSK but Y is a vector space over $\mathbb{C}$. In other words the signal space $Y = \text{span}\{g_0, \dots, g_{n-1}\}$ is a vector space over $\mathbb{C}$. $g_j(t) = \chi_{[0,1)}e^{iq_jt}$ and $\{q_j\}$ are chosen so that the g_j are orthogonal to each other. This is often used in multi-user applications, where each user is assigned to one g_j. The signal to be transmitted can be created by using inverse discrete Fourier transform and a D/A converter. Assume $q_j = 2\pi j$. The transmitted signal is translations of

$$s(t) = \sum_{j=0}^{N-1} a_j g_j(t) = \chi_{[0,1)}(t)\sum_{j=0}^{N-1} a_j e^{i2\pi t} \Rightarrow$$

$$s(n/N) = \sum_{j=0}^{N-1} a_j e^{i2\pi jn/N} = N \cdot \text{IDFT}[a](n/N),\ n = 0, \dots, N-1$$

where $a_j \in \mathbb{C}$. In resemblance with this, we will later see that inverse discrete wavelet transform can be used for wavelet communication.

1.2 Error causes

A general channel can be visualised as in figure 1.4, where L and K^* are linear operators and n is noise.

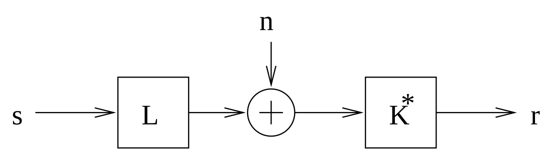

Figure 1.4: A general channel acting on the transmitted signal s

In a AWGN channel L and K^* are identity operators and n is white Gaussian noise. In a multi-path channel the signal is assumed to travel along several paths from the transmitter to the receiver. This is caused by buildings, trees and other obstacles, which will reflect the transmitted signal s. $L = \sum_j a_j \tau_{\gamma_j}$, where τ is the translation operator and $a_j \in \mathbb{C}$ and γ_j are stochastic, varying slowly in time. The receiver might not be tuned correctly to the modulation frequency ω_c, i.e the receiver uses the frequency ω_r, where ω_r might differ from ω_c. The receiver frequency could be either stochastic or deterministic. The frequency shift operator K in the receiver can be exchanged for the adjoint operator K^* acting on the channel. Instead of receiving the transmitted signal s, we receive the stochastic signal $r = K^*(Ls + n)$. The noise might be other than white especially if other communication systems are operating nearby.

Since the received signal differs from $s \in S$, we need to determine which point in S that is closest to r in order to perform the inverse mapping from S to the binary sequence. This is where the error can occur. The probability that an error will occur is called *Symbol Error Rate* (SER) and the probability that one bit is decoded incorrectly is called the *Bit Error Rate* (BER).

Chapter 2

Wavelet communication

In the last chapter we saw that orthogonal carriers have the property that the signals don't interfere with each other in an AWGN-channel. We will now study the case when the signal set is built from wavelets, scaling functions or wavelet packets. One difference with this and the other common modulation types mentioned is that previously $s_k \perp s_l,\ k \neq l$ because they have disjoint support. The subscript denotes the translation. Wavelets are inherently orthogonal to its unit translates, even though their support might not be disjoint. It is possible to use a number of wavelets $\psi_{jk} \in W_j$, for transmission at time instant k by varying j. Changing j means a dyadic dilation by a factor j. This leads to different time scales for the the different wavelets. Another possibility is to use $\varphi_{0k} \in V_0$ and $\psi_{0k} \in W_0$, which have the same time scales. The next possibility is to use the wavelet packets $\{\psi_{0k}^0, \ldots, \psi_{0k}^{2^m-1}\}$ associated to some wavelet, where $\psi_{jk}^l \in W_j^l$. These also lay in the same time scale. If the signal set should be used for multi-user communication the latter would be preferable. Though it's possible to proceed as in the TDMA case by limiting the number of ks for each user and use φ_{0k} and ψ_{0k}. Note that when we use a space Y with several dimensions it will only affect the mapping, since $V_1 = V_0 \oplus W_0$ and $V_L = W_0^0 \oplus \ldots \oplus W_0^{2^L-1}$. Thus the transmitted signal is a sum of scaling functions on level L. In the one dimensional case we also transmit a linear sum of scaling functions, but the map is done locally as will be explained in the coming chapters.

In this chapter we will investigate these different methods. It is always assumed that noise is present. Later we'll study what happens when the receiver is not tuned correctly and also what happens when multi path is present. The energy of the signal is normalised, that is $E = \int |s_k|^2 dt = 1$. The symbol rate is also assumed to be normalised, i.e $s_k = s(t-k),\ k \in \mathbb{Z}$.

2.1 One-dimensional communication

First we start with an one dimensional vector space $Y = \text{span}\{\varphi\}$ over a field $\mathbb{K}$ (,which is $\mathbb{R}$ or $\mathbb{C}$) and also the space $V_0 = \text{span}_{k\in\mathbb{Z}}\varphi(\cdot - k) = \text{span}_{k\in\mathbb{Z}}\varphi_k$, where φ is a scaling function. We also assume an AWGN-channel. In later chapters we will study more general channels. The signal set, S, is a finite set of scalars times the basis. Take the signal set to be $S = \cup_j b_j\varphi$, where b_j are elements from the coordinate space or the field $\mathbb{K}$. Let the number of elements in the set S be a dyadic number, say $2^m,\ m > 0$, which means that it is possible to transmit m bits on each symbol. The transmitted total signal is $s = \sum s_k \in V_0$ and $s_k = a_k\varphi_k$ is the symbol. First we assume that the field is $\mathbb{R}$. At the end of the chapter the case $\mathbb{C}$ is studied.

This type of signalling is obviously not orthogonal, i.e the elements in S are not orthogonal, since $\langle a_i\varphi, a_j\varphi\rangle = a_i a_j$. This leads to *Inter Channel Interference* (ICI). The signalling is free from *Inter Symbol Interference* (ISI), because $\varphi_k \perp \varphi_l \Leftrightarrow s_k \perp s_l,\ k \neq l$ and this can be exploited by the receiver. If $m = 1$, i.e binary communication, we can use antipodal carriers. The transmitted signal is $s = \sum s_k$. The received signal, in an AWGN channel, is $r = s+n = \sum s_k+n$. The term n is the white noise added by the channel. We use a correlated receiver $\lambda_k = \Lambda_k(r) = \langle r, \varphi_k\rangle = \langle \sum_l s_l, \varphi_k\rangle + \langle n, \varphi_k\rangle$. If we have orthogonality between the $\{s_k\}$, then $\lambda_k = \Lambda_k(r) = \Lambda_k(r_k)$, where $r_k = s_k + n$. Then $\Lambda_k(r) = \langle s_k, \varphi_k\rangle + \langle n, \varphi_k\rangle = a_k + n_k$. When $\Lambda_k(r) \neq \Lambda_k(r_k)$ it is called Inter Symbol Interference (ISI). The term n_k comes from a normal distribution $N(0, \sigma^2)$, where $\sigma^2 = N_0/2$, which is the spectral density of the noise. The channel condition is often described in term of Signal to Noise Ratio (SNR), where SNR $= E/N_0$. The variable $\lambda_k \sim N(a_k, \sigma^2)$ is called the decision variable.

As an example, suppose $m = 1$. We need to select $2^m = 2$ elements in $\mathbb{R}$, $\{b_0, b_1\}$, s.t $S = \cup b_j\varphi$ and that the elements of S has energy less or equal to one. Choose $b_0 = -1$ and $b_1 = 1$. If the kth bit is zero then the encoder puts $a_k = b_0$ and if the bit is one the encoder puts $a_k = b_1$. The transmitted signal is $s = \sum s_k = \sum a_k\varphi_k$. The receiver calculates $\lambda_k = \Lambda_k(r) = \Lambda_k(r_k) = a_k + n_k$. We need to determine if $a_k = b_0$ or $a_k = b_1$. If we have a good source encoder then a priori $P(a_k = b_0) = P(a_k = b_1)$. The hypothesis test is then

$$\begin{aligned} H_0 :&\quad \lambda_k = b_0 + n_k \\ H_1 :&\quad \lambda_k = b_1 + n_k \end{aligned}$$

In other words we need to determine from which distribution λ_k comes from. It is either $N(b_0, \sigma^2)$ or $N(b_1, \sigma^2)$. In our symmetric constellation, the decision boundary is 0, so H_1 is selected if $\lambda_k > 0$ and H_0 is selected if $\lambda_k < 0$.

If $m \geq 2$ the real line can be divided into 2^m intervals and $S = \cup_j b_j\varphi$. In other words, we have:

Received signal:

$$r(t) = \sum_k a_k \varphi_k(t) + n(t)$$

Decision variable:

$$\lambda_k = \Lambda_k(r) = \langle r, \varphi_k \rangle = a_k + \langle n, \varphi_k \rangle = a_k + n_k \sim N(a_k, \frac{N_0}{2})$$

The expectation value of the decision variable is:

$$\mathrm{E}[\lambda_k] = a_k,$$

and the variance is:

$$\mathrm{Var}[\lambda_k] = \mathrm{Var}[n_k] = N_0/2 = \sigma^2$$

The receiver calculates λ_k. The problem is then to determine a_k, when we know that there exist a j such that $a_k = b_j$. Figure 2.1 shows the Symbol Error Rate (SER) for different m, where $b_j,\ j = 0, \ldots 2^m - 1$ are placed uniformly in the interval $[-1, 1]$, i.e. $b_j = 2j/(2^m - 1) - 1$. The probability for symbol error (P_{se}) is then the probability that λ_k is outside the correct interval.

$$\begin{aligned}
&P_{se} = 1 - P_c = \\
&1 - \textstyle\sum_j P\left(\lambda_k \in \left(\frac{b_j + b_{j-1}}{2}, \frac{b_j + b_{j+1}}{2}\right) \middle| a_k = b_j\right) P(a_k = b_j) = \\
&1 - (1 - 2^{1-m}) P\left(X \in \left(-\frac{1}{2^m - 1}, \frac{1}{2^m - 1}\right)\right) - 2^{1-m} P\left(Y > 1 - \frac{1}{2^m - 1}\right)
\end{aligned}$$

Here $X \sim N(0, N_0/2)$ and $Y \sim N(1, N_0/2)$. The special case $m = 1$ gives:

$$\begin{aligned}
&P_{be} = P_{se} = 1 - P(Y > 0) = P(Y < 0) = \tfrac{1}{\sqrt{2\pi}\sigma} \textstyle\int_{-\infty}^{0} e^{-\frac{(x-1)^2}{2\sigma^2}} dx = \\
&\tfrac{1}{2}\left(1 - \mathrm{erf}(\tfrac{1}{\sqrt{2}\sigma})\right) = \tfrac{1}{2}\left(1 - \mathrm{erf}(\sqrt{\mathrm{snr}})\right)
\end{aligned}$$

In this case a symbol is the same as a bit. Here $\mathrm{erf}(x)$ is the error function and $\mathrm{erf}(x) = 2/\sqrt{\pi} \int_0^x e^{-t^2} dt$. (In communication another similar function is often used, the Q-function, $\mathrm{Q}(x) = 1/2(1 - \mathrm{erf}(x/\sqrt{2}))$.)

If the Meyer scaling function is used the bandwidth is $\mathrm{BW} = |\mathrm{supp}\hat{\varphi} \cap \mathbb{R}_+|/(2\pi) = |[0, 2/3]| = 2/3$ Hz and bandwidth efficiency: $\mathrm{BE} = m/BW = 3m/2$ bits/s 1/Hz. Needless to say the wavelet ψ can be used instead of the scaling function φ. The Meyer wavelet has the bandwidth $\mathrm{BW} = |\mathrm{supp}\hat{\psi} \cap \mathbb{R}_+|/(2\pi) = |[1/3, 4/3]| = 1$ Hz and bandwidth efficiency: $\mathrm{BE} = m/BW = m$ bits/s 1/Hz, where 2^m is the number of elements in S.

If we take Y as a vector space over $\mathbb{C}$. The signal set is as before some elements in Y, i.e $S = \cup_j b_j \varphi$, where $b_j \in \mathbb{C}$. In this case the decision variable

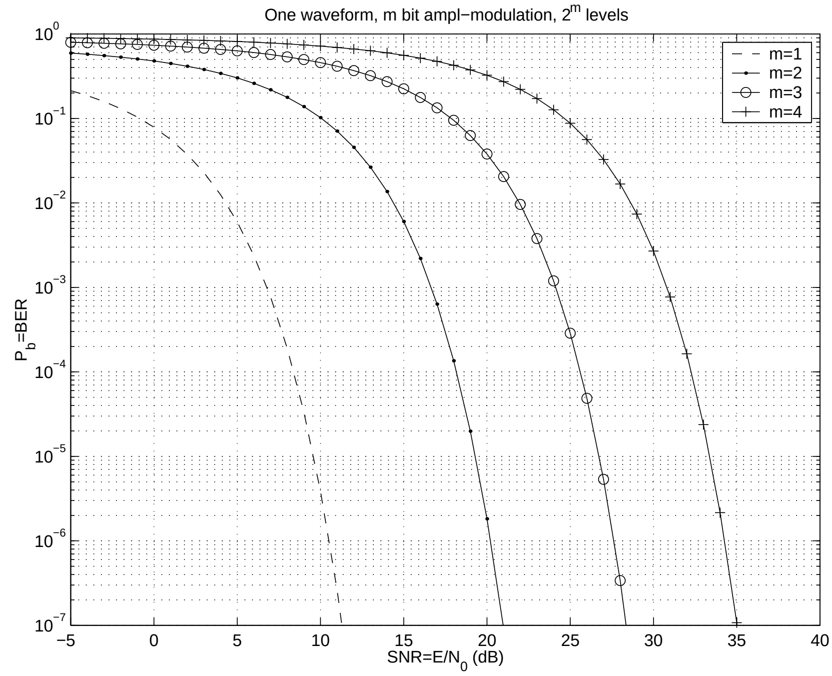

Figure 2.1: Symbol error rate for one dimensional communication, with amplitude modulation

λ_k is complex. Regarding the isometric isomorphism $\mathbb{C} \cong \mathbb{R}^2$, this turns into the two dimensional real case studied in the next chapter. In other words the bit error rate is the same as for two dimensional real communication. The bandwidth and bandwidth efficiency is the same as in the one dimensional real case. This case, $(\mathbb{C}, \operatorname{span}\varphi)$, can be compared to the common modulation scheme PSK. PSK is just the special case when φ is chosen as the Haar scaling function. Thus the the bit error rate is the same in an AWGN channel.

2.2 Two-dimensional communication

Next we consider two-dimensional signalling. Start with a space spanned by two functions. Then the signal set is a finite set of elements in this space. In our case we consider the space spanned by a scaling function and a wavelet, i.e $Y = \operatorname{span}\{\varphi, \psi\}$. As before we can regard Y as a vector space over $\mathbb{R}$ or $\mathbb{C}$. In this chapter we will study the real case. In the complex case, the same comments applies as those made in the last chapter about the one dimensional complex case, i.e it turns into the four dimensional real case. The elements in S could be chosen as points on the unit circle in

Y, which would mean that that all elements in S have unit energy. Then one would like to choose the points so that the minimum distance between all pair of points is as large as possible. If we want to transmit an integer number of bits on each symbol we need to choose a dyadic number of points. Taking $S = \{\varphi, \psi\}$ would mean binary orthogonal communication. If we take $S = \{+\varphi, -\varphi, +\psi, -\psi\}$ we have 4-ary antipodal communication. For 2^m-ary communication we can take the 2^m points as 2^m roots of unity. Another possibility is to take points not only on the unit circle, but on the unit disc. This is called Quadrature Amplitude Modulation (QAM). As before the points should be as distant as possible from each other. The next picture, figure 2.2, shows 8 points on the unit circle and on the unit disc, chosen according to this principle. With the same notation as before, select 2^m points, $\{b_j\}_{j=0}^{2^m-1}$, in the coordinate space $\mathbb{R}^2$. Let $v = (\varphi, \psi)^T$. Then $S = \cup_j v^T b_j$.

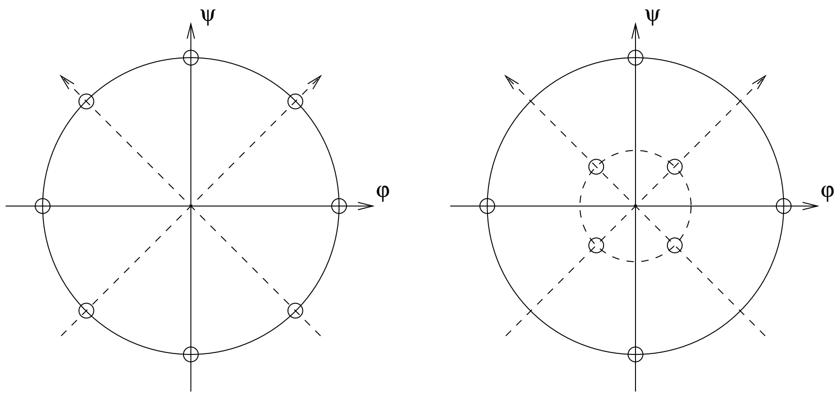

Figure 2.2: Two examples of coding constellations in two dimensions, when S contains 8 points.

If Meyer wavelet and scaling function is used, then this constellation has the bandwidth: BW $= |(\text{supp}\hat{\varphi} \cup \text{supp}\hat{\psi}) \cap \mathbb{R}_+|/(2\pi) = 4/3$ Hz and bandwidth efficiency: BE $= m/BW = 3m/4$ bits/s 1/Hz.

As an example, consider the case $m = 2$. With the same notation as before, select four points, $\{b_j\}_{j=0}^{3}$, in the coordinate space $\mathbb{R}^2$.
Transmitted signal:

$$s = \sum s_k = \sum a_k^T v_k$$

where

$$
\begin{aligned}
v_k &= \begin{pmatrix} \phi_k \\ \psi_k \end{pmatrix} \\
a_k &\in \left\{ b_0 = \begin{pmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \end{pmatrix}, b_1 = \begin{pmatrix} -1/\sqrt{2} \\ 1/\sqrt{2} \end{pmatrix}, \right. \\
&\qquad \left. b_2 = \begin{pmatrix} -1/\sqrt{2} \\ -1/\sqrt{2} \end{pmatrix}, b_3 = \begin{pmatrix} 1/\sqrt{2} \\ -1/\sqrt{2} \end{pmatrix} \right\} = \cup_{j=0}^{3} \{b_j\}
\end{aligned}
$$

Received signal:

$$
r = \sum_k s_k + n
$$

Decision variable:

$$
\begin{aligned}
\lambda_k &= \begin{pmatrix} \langle r, \varphi_k \rangle \\ \langle r, \psi_k \rangle \end{pmatrix} = \begin{pmatrix} \langle \sum v_k^T a_k, \varphi_k \rangle \\ \langle \sum v_k^T a_k, \psi_k \rangle \end{pmatrix} + \begin{pmatrix} \langle n, \varphi_k \rangle \\ \langle n, \psi_k \rangle \end{pmatrix} \\
&= a_k + n_k \sim N(a_k, \sigma^2 I)
\end{aligned}
$$

Or expressed as an orthogonal projection on Y: $y_\lambda^k = \mathcal{P}_Y^k r = \mathcal{P}_Y r(\cdot - k) = a_k^T v_k + n_k^T v_k$. Now divide the space Y into four regions Y_i.

$$
Y_i = \left\{ y \in Y : \|y - v^T b_i\|_2 \leq \min_{j \neq i} \|y - v^T b_j\|_2 \right\}
$$

With our selection of b_j, Y_i are the four quadrants. If $\lambda_k \in Y_i$ the interpretation is then that $a_k = b_i$. The probability of symbol error is:

$$
\begin{aligned}
P_s = \sum_i P(\lambda \notin Y_i | a_k = b_i) P(a_k = b_i) = \\
1 - \sum_i P(\lambda \in Y_i | a_k = b_i) P(a_k = b_i) = \\
1 - \sum_i \frac{1}{2\pi\sigma^2} \int_{Y_i} e^{-\frac{((x_1, x_2) - b_i) \cdot ((x_1, x_2) - b_i)}{2\sigma^2}} dx P(a_k = b_i)
\end{aligned}
$$

Now we assume a symmetric constellation, i.e where the points are placed on the unit circle and all symbols are equally probable. Then the equation reduces to:

$$
P_s = 1 - P(\lambda \in Y_0 | a_k = b_0) = 1 - \frac{1}{2\pi\sigma^2} \int_{Y_0} e^{\frac{((x_1, x_2) - b_0) \cdot ((x_1, x_2) - b_0)}{2\sigma^2}} dx
$$

When $m = 2$ this can be calculated analytically and expressed in terms of the error function.

$$
P_s = \frac{1}{4}(1 - \text{erf}(\frac{1}{2\sigma}))(3 + \text{erf}(\frac{1}{2\sigma})) = \frac{1}{4}(1 - \text{erf}(\sqrt{\frac{\text{snr}}{2}}))(3 + \text{erf}(\sqrt{\frac{\text{snr}}{2}}))
$$

The probability of bit error depends on how the bit sequence is encoded to a symbol, i.e how we map the bit sequence onto S. Generally it should be done in such a way that the difference in the bit sequence between two neighbours in Y is as small as possible, since if the symbol is decoded incorrectly it is most likely that the symbol is interpreted as the neighbour. In other words the map should in some sense preserve distances, i.e sequences with large Hamming distance should be mapped to points with large L^2 distance, and small Hamming distance should be mapped to points with small L^2 distance. The Hamming distance between two sequences is equal to the number of positions in which they differ.

As an example, if $m = 2$ then 11 should be placed at the opposite side of 00 on the unit circle and 01 and 10 as neighbours to 00. In this case the following BER is achieved.

$$\mathrm{BER} \equiv P_{be} = P(\text{bit error}|\text{symbol error})P(\text{symbol error}) = \frac{1}{2}(1 - \mathrm{erf}(\sqrt{\frac{\mathrm{snr}}{2}}))$$

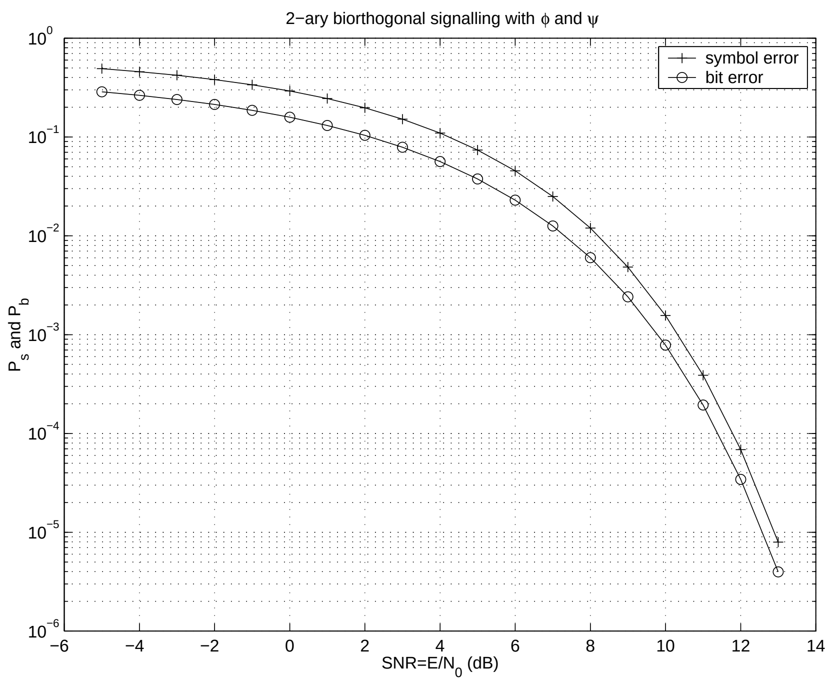

Figure 2.3: Symbol and bit error rate for two dimensional communication, with bit rate $m = 2$

Figure 2.3 shows the symbol- and bit error rate plotted against SNR= $E/N_0 = 2E_b/N_0$, when $m = 2$. Here E_b is the bit energy and $E = 2E_b$ is the symbol energy. The symbol error for $m = 2$ can also be extracted from figure 2.1, since in this case it is essentially two one-dimensional communications in parallel. We have $P_s = 1 - (1-p)^2$, where p is the bit error for one-dimensional communication, when $m = 1$. The snr in figure 2.1 would be shifted since the energy will be shared by the two systems in parallel, i.e. snr $\curvearrowright$ snr/2.

When $m > 2$ it is difficult to get a closed expression of the symbol error for two dimensional communication, but it is possible to get tight upper- and lower bounds of the symbol error. Again suppose that the points are placed equally spaced on the unit circle and all symbols are equally probable.

Assume $b_0 = (1,0)^T$ was transmitted, which gives $E[\lambda] = (1,0)^T$. The space $\mathbb{R}^2$ is divided as before into different decision regions Y_j. Define the following subsets of $\mathbb{R}^2$

$$\begin{aligned}
&Y_0 = \{(x,y) \in \mathbb{R}^2 : x > 0,\ -\alpha_0 < \arctan \tfrac{y}{x} < \alpha_0\}, \quad \text{where } \alpha_0 = 2^{-m}\pi \\
&\tilde{Y_0} = \{(x,y) \in \mathbb{R}^2 : x < 0,\ -\alpha_0 < \arctan \tfrac{y}{x} < \alpha_0\} \\
&A(\alpha) = \{(x,y) \in \mathbb{R}^2 : y\cos\alpha > x \sin\alpha\} \\
&C = \{(x,y) \in \mathbb{R}^2 : x < 0, |x| > |y|\}
\end{aligned}$$

The symbol is decoded correctly if $\lambda \in Y_0$ otherwise a symbol error has occurred. Thus we need to calculate $P_{se} = P(\lambda \notin Y_0) = 1 - P(\lambda \in Y_0)$. $\tilde{Y_0}$ is the region on the opposite side of the unit circle from Y_0.

$$\begin{aligned}
&P(\lambda \in A(\alpha)) = \tfrac{1}{\sqrt{2\pi}\sigma} \int_0^\infty e^{-\frac{(x+\sin\alpha)^2}{2\sigma^2}} dx = \tfrac{1}{2}(1 - \operatorname{erf}(\tfrac{\sin\alpha}{\sqrt{2}\sigma})) \\
&P(\lambda \in C) = \tfrac{1}{2\pi\sigma^2} \int_0^\infty \int_0^\infty e^{-\frac{(x+1/\sqrt{2})^2+(y+1/\sqrt{2})^2}{2\sigma^2}} dx dy = \\
&\left(\tfrac{1}{\sqrt{\pi}} \int_{1/(2\sigma)}^\infty e^{-z^2} dz\right)^2 = \tfrac{1}{4}(1 - \operatorname{erf}(\tfrac{1}{\sqrt{2}\sigma}))^2
\end{aligned}$$

Using the above we have the following equality and inequality.

$$\begin{aligned}
&P(\lambda \in Y_0) - P(\lambda \in \tilde{Y_0}) = P(\lambda \in A(-2^{-m}\pi)) - P(\lambda \in A(2^{-m}\pi)) \\
&P(\lambda \in \tilde{Y_0}) \le 2^{2-m} P(\lambda \in C)
\end{aligned}$$

With equality in the last expression for $m = 2$ and strict inequality for $m > 2$. Now we can estimate the symbol error P_{se}, by first rewriting the symbol error as

$$\begin{aligned}
P_{se} &= 1 - P(\lambda \in Y_0) = 1 - [P(\lambda \in Y_0) - P(\lambda \in \tilde{Y_0})] - P(\lambda \in \tilde{Y_0}) = \\
&1 - \left[P(\lambda \in A(-2^{-m}\pi)) - P(\lambda \in A(2^{-m}\pi))\right] - P(\lambda \in \tilde{Y_0})
\end{aligned}$$

Finally put $\sigma^{-2} = 2\,\mathrm{snr}$ and after some elementary calculations, we get the following upper and lower bound for the symbol error.

$$\begin{aligned}
2I(m) - 2^{-m}c \le P_{se} &< 2I(m), \qquad \text{where} \\
I(m) &= \frac{1}{2}(1 - \operatorname{erf}(\sin(\pi/2^m)\sqrt{\mathrm{snr}})) \\
c &= (1 - \operatorname{erf}(\sqrt{\mathrm{snr}/2}))^2
\end{aligned}$$

The left inequality is an equality for $m = 2$ and a strict inequality for $m > 2$. These bounds are shown in the next picture, figure 2.4. As we can see the upper and lower bounds are quite close to each other. The leftmost curve ($m = 2$) correspond to the symbol error curve in figure 2.3.

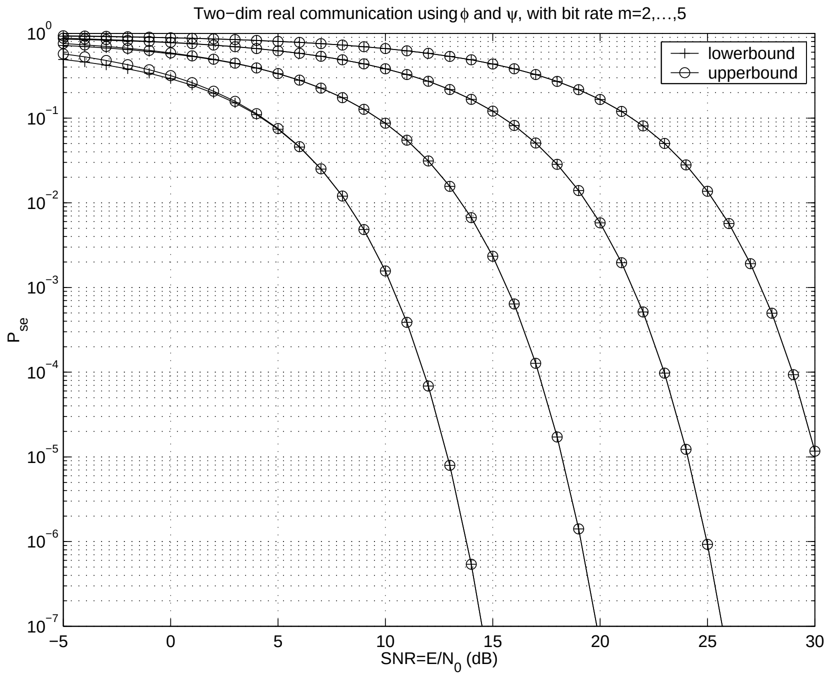

Figure 2.4: Symbol error rate for two dimensional communication, with bit rate $m \geq 2$

2.3 Many-dimensional communication

Next we consider many-dimensional communication, i.e. with dimension greater than two. The natural extension from wavelet and scaling function is to wavelet packets. Consider the d-dimensional vector space spanned by the wavelet packet functions $Y = \text{span}\{\psi^0, \ldots, \psi^{d-1}\}$, where $\{\psi^i\}_i$ are the wavelet packet functions. We need to choose a number of points b_j in the coordinate space $\mathbb{R}^d$, s.t $S = \cup_j\{b_j^T v\}$, where $v^T = (\psi^0, \ldots, \psi^{d-1})$. If wavelet packets are used, d is a dyadic number, if we include all packages on a certain level. The selection of points can be done in many different ways. In the complex case the points are chosen in $\mathbb{C}^d$ instead. Using the isometric isomorphism $\mathbb{C} \cong \mathbb{R}^2$, the complex case turns into $2d$-dimensional real communication.

The first strategy, which is the same as before, select 2^m points on the unit sphere in $\mathbb{R}^d$, s.t the minimum distance between all pair of points is as large as possible. Then m bits can be sent each time.

A second strategy can be used, which is natural if the system would be for multi-user communication. For n users, divide the space in to smaller subspaces $\mathbb{R}^d = \mathbb{R}^{d_0} \times \mathbb{R}^{d_1} \times \ldots \times \mathbb{R}^{d_{n-1}}$, where $d = \sum d_i$. In $\mathbb{R}^{d_i}$, 2^{m_i} points are selected using the first strategy. The points in $\mathbb{R}^d$ are then the Cartesian product of these points. This is essentially n communications in parallel. This strategy selects $\prod_{i=0}^{n-1} 2^{m_i}$ points on the unit sphere in $\mathbb{R}^d$. Thus we are able to transmit $\sum_{i=0}^{n-1} m_i$ bits at each time instant. An example if d is dyadic, $d_i = 2$, $\forall i$ and $n = d/2$. Then $\mathbb{R}^d$ can be written as $\mathbb{R}^d = \mathbb{R}^2 \times \mathbb{R}^2 \times \ldots \times \mathbb{R}^2$. Select m points in $\mathbb{R}^2$ using the first strategy, which in this case is the same as in the last chapter, refer to figure 2.2. Then $md/2$ bits can be transmitted on each symbol. If d_i and m_i does not depend on i then this second strategy yields points which would be chosen according to the first strategy.

As an example suppose $Y = \operatorname{span}\{\psi^0, \psi^1, \psi^2, \psi^3\}$. We need to select 2^m points in $\mathbb{R}^4$. Using the first strategy we could for example choose the points on the coordinate axes crossing the unit sphere, that is $8 = 2^3$ points. This means $S = \{\pm\psi^i\}_{i=0}^3$. Another possibility is to choose the $16 = 2^4$ points $(\pm1, \pm1, \pm1, \pm1)/2$, which are the corner points of a hyper cube in $\mathbb{R}^4$. The first has 3 as bit rate and the second 4. As said before this is the same as using strategy 2 and writing $\mathbb{R}^4 = \mathbb{R}^2 \times \mathbb{R}^2$ and choosing the four points $(\pm1, \pm1)/\sqrt{2}$ in $\mathbb{R}^2$. The points in $\mathbb{R}^4$ are then $(\pm1, \pm1, \pm1, \pm1)/2$, which has been normalised to have unit norm.

When the points are chosen on the hyper cube the bit error rate, P_b, can be extracted form previous calculations. The BER is equal to the one dimensional antipodal bit error rate, with the snr exchanged for snr$/d$, since the energy of the symbol is shared shared between several bits. The symbol error in this case is $P_{se} = 1 - (1 - P_b)^d$.

Figure 2.5 shows the symbol error rate for d-dimensional communication using the 2^d points, $2(\pm1, \ldots, \pm1)/\sqrt{d}$, which gives d as bit rate.

2.4 General channel

In the past chapters we have assumed an AWGN-channel. Now we will study more general channels, refer to the chapter about error causes and figure 1.4. The received signal is thus:

$$r = K^*(Ls + n)$$

The operators K and L are linear. K is the frequency shift induced by the receiver and L is the operator induced by the channel. L is usually time varying. It could for example model the multi-path scenario. We will also consider noise (n) other than white, i.e coloured or correlated noise. The

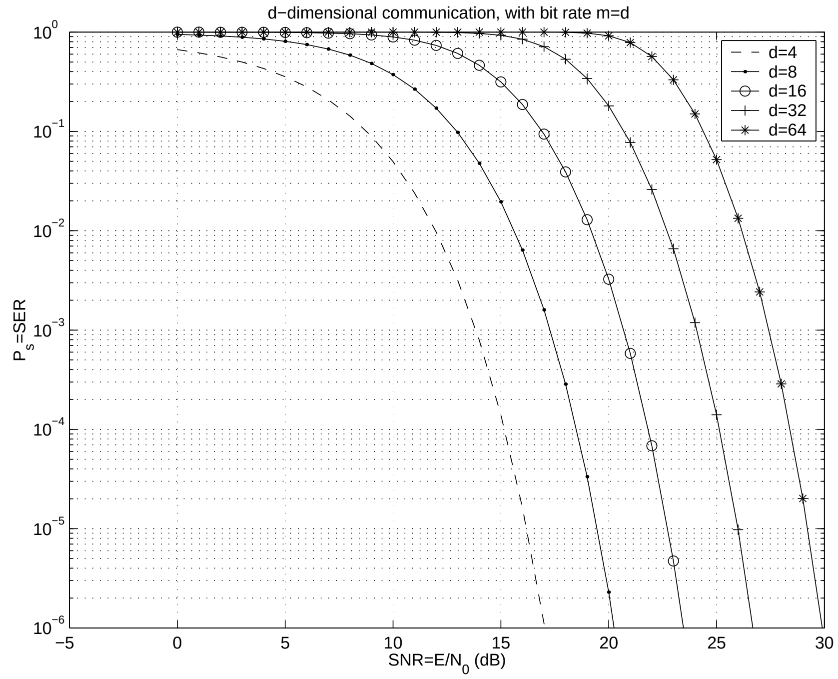

Figure 2.5: Symbol error rate for d dimensional communication, with dyadic d and bit rate $m = d$

channel is time varying, but usually the variation of L is slow w.r.t. the symbol rate. Thus we regard L as stationary over a symbol. In this case L can be represented by a kernel h, which is called the impulse response, i.e $Ls = h * s$. Both L and K give rise to ISI and ICI, which are unwanted effects. Which influence do these operators have on our signals and the error probability? What can be done to reduce these unwanted effects caused by K and L? In this chapter we will try to address these questions.

Let $x = (x_1, \dots, x_N)$ be the transmitted elements from S and $y = (y_1, \dots, y_N)$ the elements in S decoded by the receiver. Then the mutual information is defined as, see reference [Pro95]

$$\begin{aligned} I(X,Y) &= \int_{\mathbb{R}^N} \int_{\mathbb{R}^N} p(y|x)p(x) \log \frac{p(y|x)}{p(y)} dxdy = \\ \text{where: } p(y|x) &= \prod_{i=1}^{N} p(y_i|x_i) = \prod_{i=1}^{N} \frac{1}{\sqrt{\pi N}} e^{-(x_i - y_i)} \end{aligned}$$

In the second row the outputs y_i are assumed to be uncorrelated. The channel capacity is defined as $C = \max_{p(x)} I(X,Y)$. Shannon showed that the channel capacity for an AWGN-channel with band width W is

$$C = \max_{p(x)} I(X,Y) = W \log_2(1 + \frac{P_{av}}{WN_0})$$

where P_{av} is the available power.

Let ϕ_n denote the the autocorolation of a stationary random process n. Φ_n is the spectral density of the process n: $\Phi_n = \int_{\mathbb{R}} \phi_n(t) e^{-2\pi i f t} dt$. Assume now that we have a multi-carrier system with bandwidth Δf on each channel. The received power on the ith channel is $\Delta f P(f_i) |\hat{h}(2\pi f_i)|^2$ and the noise spectral density on this channel is $\Phi_n(f_i)$, thus

$$C = \sum_{i=1}^{N} C_i = \Delta f \sum_{i=1}^{N} \log_2 \left[1 + \frac{P(f_i)|\hat{h}(2\pi f_i)|^2}{\Phi_n(f_i)} \right]$$

Let $\Delta f \to 0$, and we obtain the channel capacity as

$$C = \int \log_2 \left[1 + \frac{P(f)|\hat{h}(2\pi f)|^2}{\Phi_n(f)} \right] df$$

We maximise this quantity with the constraint that $\int P(f) df = P_{av}$, by assuming there exist such a maximiser P_0. Put $P = P_0 + \epsilon P_1$, where P_1 is an arbitrary function. Let Q be as follows

$$Q = \int \log_2 \left[1 + \frac{(P_0(f) + \epsilon P_1(f))|\hat{h}(2\pi f)|^2}{\Phi_n(f)} \right] - \mu (P_0(f) + \epsilon P_1(f)) df$$

Now find P_0 by solving $\partial Q / \partial \epsilon |_{\epsilon=0} = 0$.

$$\frac{\partial}{\partial \epsilon}\bigg|_{\epsilon=0} \int \log_2 \left[1 + \frac{(P_0(f) + \epsilon P_1(f))|\hat{h}(2\pi f)|^2}{\Phi_n(f)} \right] - \mu (P_0(f) + \epsilon P_1(f)) df = 0$$

Which gives if we substitute P_0 for P again.

$$\frac{1}{|\hat{h}(2\pi f)|^2 P(f) + \Phi_n(f)} + \mu = 0$$

Thus for this type of channel we need to determine $P(f)$ such that we in a given frequency band have:

$$P(f) = \Phi_s(f) = \max \left(0, \lambda - \frac{\Phi_n(f)}{|\hat{h}(2\pi f)|^2} \right), \qquad \text{where} \int P(f) df = P_{av}$$

If we use wavelets localised in frequency, such as the Meyer wavelet, we can build up the carriers from a linear combination of some wavelet packets to approximate a given spectral density.

We will now consider the three situations separately. First a few words about coloured noise, i.e $\hat{R}_n$ is not constant in the communication band. Then operator K, i.e the situation when the receiver frequency differs from the carrier frequency. Then the fading case when $h \neq \delta$.

When the noise is not white, we have already seen how the power spectrum should be chosen.

As an example with coloured noise. We can study binary communication and determine which receiver to use.

Assume we have the one dimensional signal space $Y = \{a\varphi : a \in \mathbb{R}\}$ and zero mean noise. Which receiver $|g\rangle$ should we use? The receiver calculates the decision variable λ.

$$
\begin{aligned}
\lambda &= \langle s, g\rangle + \langle n, g\rangle \\
E[\lambda] &= E[\langle s, g\rangle] = \langle s, g\rangle \\
\mathrm{Var}[\lambda] &= E[\langle n, g\rangle^2] = \int\int g(x)g(y)\mathrm{E}[n(x)n(y)]dxdy = \\
&\int\int g(x)g(y)\phi_n(x-y)dxdy = \langle g, g * \phi_n\rangle
\end{aligned}
$$

The signal to noise ratio is proportional to $\alpha = E[\lambda]^2/\mathrm{Var}[\lambda]$. In the case of white noise this quantity is $2E/N_0$. We need to determine g to maximise this quantity with the restriction $\|g\|_2 = 1$, i.e minimise the denominator in α. Assume there exist such a g and call it g_0 and put $g = g_0 + \epsilon g_1$ for any g_1. Then we will find g_0 by maximising Q, i.e. solving $\partial Q/\partial\epsilon|_{\epsilon=0} = 0$.

$$
\begin{aligned}
\frac{\partial Q}{\partial \epsilon}\Big|_{\epsilon=0} &= \frac{\partial}{\partial \epsilon}\Big|_{\epsilon=0}\langle g_0 + \epsilon g_1, (g_0 + \epsilon g_1) * \phi_n\rangle - \mu\langle s, g_0 + \epsilon g_1\rangle = \\
&\int\left[\int g_0(x)R_n(y-x)dx - \mu s(y)\right] g_1(y)dy = 0, \ \forall g_1
\end{aligned}
$$

Thus $\hat{s} = \mu^{-1}\hat{g}\hat{\phi}_n = \mu^{-1}\hat{g}\Phi_n(\cdot/(2\pi))$ and we can take $\mu = 1$. So $g = \mathcal{F}^{-1}(\hat{s}/\hat{\phi}_n)$. For this choice of receiver we will get

$$
\sup_g E[\lambda]^2/\mathrm{Var}[\lambda] = \langle s, g\rangle = \frac{1}{2\pi}\langle \hat{s}, \frac{\hat{s}}{\hat{\phi}_n}\rangle
$$

2.5 Frequency instable receiver/transmitter

In this chapter we will study the effects that arise when the receiver frequency, ω_r differs from the transmitter frequency ω_c. Assume the baseband transmitted signal, with $c_k = (a+ib)_k \in \mathbb{C}$ and that φ is real, is.

$$
s_k = c_k\varphi_k
$$

Applying frequency shift and taking the real part of the signal gives the pass band signal.

$$
s_k^p(t) = \mathrm{Re}[c_k\varphi_k\sqrt{2}e^{i\omega_c t}] = a_k\varphi_k\sqrt{2}\cos\omega_c t - b\varphi_k\sqrt{2}\sin\omega_c t
$$

The receiver applies negative frequency shift.

$$\begin{aligned}r_k(t) = s_k^p(t)\sqrt{2}e^{-i\omega_r t} = \\ &2(a_k\varphi_k\cos\omega_c t - b_k\varphi_k\sin\omega_c t)(\cos\omega_r t - i\sin\omega_r t) = \\ &\varphi_k(t)[a_k(\cos(\omega_c-\omega_r)t + \cos(\omega_c+\omega_r)t) - \\ &b_k(\sin(\omega_c-\omega_r)t + \sin(\omega_c+\omega_r)t) - \\ &ia_k(\sin(\omega_r-\omega_c)t + \sin(\omega_c+\omega_r)t) + \\ &ib_k(\cos(\omega_c-\omega_r)t - \cos(\omega_c+\omega_r)t)]\end{aligned}$$

Let $\Delta\omega$ denote the difference between the receiver and transmitter frequency, i.e $\Delta\omega = \omega_r - \omega_c$, and applying a low pass filter, we get the signal

$$\begin{aligned}\varphi_k(t)[a_k\cos\Delta\omega t + b_k\sin\Delta\omega t - ia_k\sin\Delta\omega t + ib_k\cos\Delta\omega t] = \\ \varphi_k(t)(a+ib)_k e^{-i\Delta\omega t} = c_k\varphi_k(t)e^{-i\Delta\omega t}\end{aligned}$$

Scalar product with a real function g_l.

$$c_k\langle\varphi_k, e^{i\Delta\omega\cdot}g_l\rangle = c_k\int_{\mathbb{R}}\varphi_k(t)g_l(t)e^{-i\Delta\omega t}dt = c_k\widehat{\varphi_k g_l}(\Delta\omega) =$$

$$\frac{c_k}{2\pi}\hat{\varphi}_k * \hat{g}_l(\Delta\omega) = \frac{c_k}{2\pi}\int_{\mathbb{R}}\hat{\varphi}_k(\xi)\hat{g}_l(\Delta\omega-\xi)d\xi = \frac{c_k}{2\pi}\int_{\mathbb{R}}\hat{\varphi}_k(\xi)\overline{\hat{g}_l}(\xi-\Delta\omega)d\xi$$

Ignoring the noise for a moment, let the transmitted signal be $s = \sum_k s_k$, where $s_k = \sum_j c_k^j\psi_k^j$. The received signal is $r(t) = s(t)e^{-i\Delta\omega t}$. Now we want to find c_p^q. Using the same calculations as in previous chapters, we have

$$\tilde{c_p^q} = \langle s, e^{i\Delta\omega\cdot}\psi_p^q\rangle = \sum_k\sum_j c_k^j\langle\psi_k^j e^{i\Delta\omega\cdot}\psi_p^q\rangle = \sum_{k,j}c_k^j\frac{1}{2\pi}\hat{\psi_k^j} * \hat{\psi_p^q}(\Delta\omega)$$

We know $\tilde{c_p^q} = c_p^q$, when $\Delta\omega = 0$. Now we would like that the error doesn't become too large when $\Delta\omega$ is around zero, i.e $\tilde{c_p^q} \approx c_p^q$, when $\Delta\omega$ is around zero. This implies that $\frac{1}{2\pi}\hat{\psi_k^j} * \hat{\psi_p^q}(\Delta\omega)$ should be approximately one when both $k = p$ and $j = q$, otherwise it should be approximately zero.

The next set of plots, figure 2.6, shows $\int f_0(t)\overline{g_k(t)}e^{-i\Delta\omega t}dt$, where f_0 is the transmitted signal (φ or ψ). The receiver is $g_k(t)e^{i\Delta\omega t}$, where $g = \varphi$ or ψ, $g_k = g(\cdot - k)$ and $k = 0, 1$. Here φ and ψ are the Meyer scaling function and wavelet respectively.

Note that the angular frequencies are normalised. The normalised carrier angular frequency is achieved by $\omega_c = \omega_c^u\mathrm{BW}/\mathrm{BW}^u$, where the superscript u means unnormalised.

As an example, suppose only the Meyer scaling function is used, with a carrier frequency of 1 GHz and bandwidth 100 MHz. Then $\Delta\omega = 1$ in the above plots means a frequency shift of $\omega_c^u = 1\cdot\mathrm{BW}^u/\mathrm{BW} = 100/(2/3) =$ 150 MHz and $\int\varphi(t)\overline{\varphi(t)}e^{-i\Delta\omega t}dt = 0.89$ for this shift.

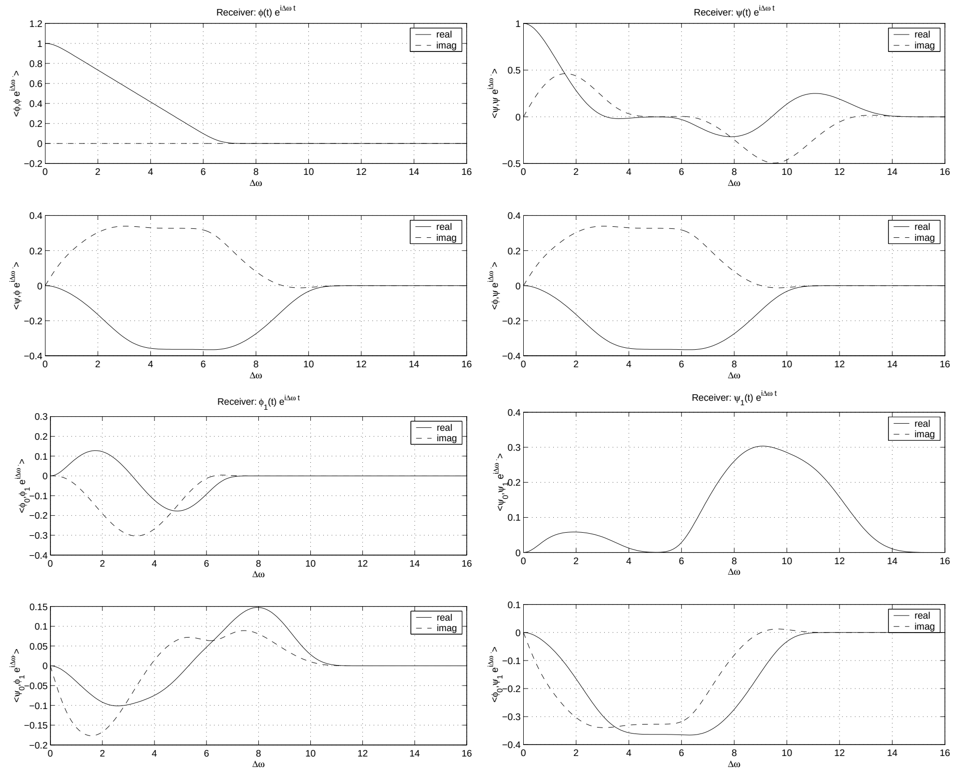

Figure 2.6: Different receivers, and different received signals.

The next question one would ask is: How is the bit error or the symbol error affected by this frequency shift? Assume that the Meyer family is used and $\Delta\omega \neq 0$. We get different results depending on the dimension of our signal. When the signal space Y is a vector space over $\mathbb{R}$, only the real parts of the plots are interesting. The imaginary parts is the interference between the real and the imaginary functions in Y when Y is a vector space over $\mathbb{C}$. We would like the real parts of the upper two plots to be one and the rest to be zero. Everything different from zero means some sort of interference.

Let us first study the one dimensional case where only the scaling function is used. In this case only plot 1 and 3 in the left column of plots in figure 2.6 are relevant to us. As seen from the pictures $\langle \varphi_k, \varphi_k e^{i\Delta\omega\cdot} \rangle < 1$, which effectively means a degradation of SNR. One can also see that $\langle \varphi_k, \varphi_k e^{i\Delta\omega\cdot} \rangle \in \mathbb{R}$, which means that if complex coefficients are used, i.e Y is a vector space over $\mathbb{C}$, then the real and imaginary parts of the coefficients won't interfere with each other. The orthogonality between symbols is also lost, which can be seen from the plot of $\langle \varphi_k, \varphi_{k+1} e^{i\Delta\omega\cdot} \rangle$. This leads to ISI. The result of this scalar product is complex.

In the two dimensional case all eight plots in figure 2.6 are interesting.

We can see from the plots that most things will interfere with each other, but mainly the real and imaginary parts. So if a substantial frequency shift is expected, then a real vector space would be recommended, in this particular case.

Let us consider a concrete example. Assume $S = \{\sum_{j=0}^{7} a_j \psi_{00}^j : a_j \in \{-1, 1\}\}$, where $\{\psi_{00}^j\}_{j=0}^7$ are the eight first Meyer wavelet packets. This will give a bit rate of 8. Assume we can clock the beginning of a symbol at the receiver, which means that the phase error at in the beginning of a symbol is zero. The received signal can in this case be written as

$$r = \sum_{j,k} s_k \chi_{[0,1)}(\cdot - j) e^{-i\Delta\omega_j(\cdot - j)} + n$$

where $\sum s_k$ is the transmitted signal and $s_k(\cdot + k) \in S$. The noise is assumed to be white with spectral density $N_0/2$. A Monte Carlo simulation of this was done in Matlab, with 20000 transmitted symbols, which means 160000 bits. Figure 2.7 shows the result of this. The bit error plotted against the bit-SNR for six different frequency shifts.

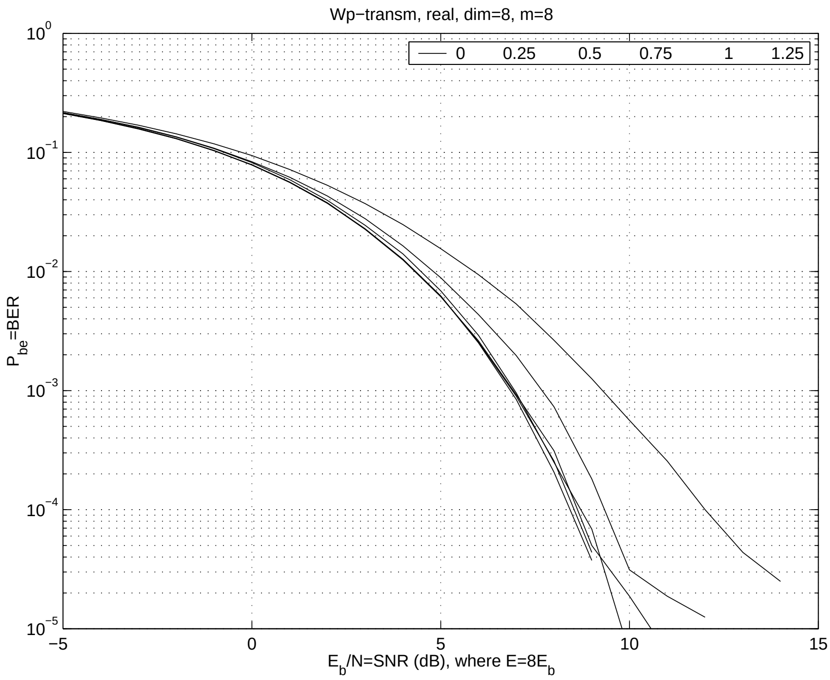

Figure 2.7: Bit error rate for communication using 8 Meyer wavelet packets, with bit rate $m = 8$. The receiver frequency is different from the transmitter frequency. Six different $\Delta\omega$. Note that the x-axis is $E_b/N_0 = E/(8N_0)$.

We can see from this plot that the signalling is quite resistant to the frequency error.

2.6 Multi-path and fading

In this chapter we will study the effects of a fading- or a multi-path channel. In a multi-path type of channel several copies of the transmitted signal are received by the receiver with different amplitudes and phases. Let s and s^b be the low- and band pass- transmitted signal respectively, i.e $s^b(t) = \mathrm{Re}[s(t)e^{i\omega_c t}]$. The received band pass signal could be written as

$$r^b(t) = \mathrm{Re}\left[\sum_n [\alpha_n e^{-i\omega_c \gamma_n(t)} s(t-\gamma_n(t))] e^{i\omega_c t}\right]$$

The low pass equivalent as

$$r(t) = \sum_n \alpha_n e^{-i\omega_c \gamma_n(t)} s(t-\gamma_n(t))$$

where α_n is the attenuation for the nth path and the γ_n is the propagation delay for this path. Both are stochastic. When the channel varies slowly in time, the multi-path operator L can be expressed as a convolution with an impulse response h. We will consider two cases, flat fading and frequency selective fading.

In a flat fading channel or a non frequency selective channel. One assumes that all frequencies of a symbol are attenuated equally. For Rayleigh fading the attenuation (α) of the signal follows a Rayleigh distribution. Let r be the received low pass signal then

$$r(t) = \alpha e^{-i\phi} s(t) + z(t)$$

where z represents complex white Gaussian noise. The phase shift can be estimated by the receiver hence we ignore it. Let η denote the SNR, then $\eta = \alpha^2 E/N_0$. The error probability in this channel can be expressed as

$$P_{fade} = \int_0^\infty P_{se}(\eta)p(\eta)d\eta$$

For Rayleigh fading α is Rayleigh distributed and it follows that η is exponentially distributed.

$$p(\eta) = \frac{1}{\eta_0} e^{-\eta/\eta_0}$$

with the expectation value $\mathrm{E}[\eta] = \eta_0$. Rice distribution models line of sight communication. The channel filter is $h = (a_0 + a)\delta$, where a_0 is constant and models the direct signal and a, which is a superposition of the reflected rays follows the Rice distribution.

$$p(a) = \mathrm{I}_0\left(\frac{aa_0}{\sigma^2}\right)\frac{a}{\sigma^2} e^{-\frac{a^2+a_0^2}{2\sigma^2}}$$

As an example we can consider the one dimensional antipodal case. Here we saw that the bit error rate in an AWGN channel was.

$$P_{be} = \frac{1}{2}(1 - \mathrm{erf}(\sqrt{\mathrm{snr}}))$$

In a Rayleigh fading channel, we would get the symbol error probability as

$$\begin{aligned} P_{fade} &= \frac{1}{2}\int_{\eta=0}^{\infty}\left(1 - \frac{2}{\sqrt{\pi}}\int_{t=0}^{\sqrt{\eta}} e^{-t^2}dt\right)\frac{1}{\eta_0}e^{-\eta/\eta_0}d\eta = \\ & \frac{1}{2}\left(1 - \frac{2}{\eta_0\sqrt{\pi}}\int_0^{\infty} 2xe^{-x^2/\eta_0}\int_0^x e^{-t^2}dtdx\right) = \\ & \frac{1}{2}\left(1 - \sqrt{\frac{\eta_0}{1+\eta_0}}\right) \end{aligned}$$

where we obtain the last equality by one partial integration.

Another interesting channel case is the frequency selective fading. In this case the channel will distort our transmitted signal. Different frequencies will experience different attenuations. In the article [Rum79] by Rummler, the following transfer function was used.

$$\hat{h}(\omega) = \alpha(1 - \beta e^{-i(\omega-\omega_0)\gamma})$$

Which is equivalent to

$$\begin{aligned} h &= \alpha(1 - \beta e^{i\omega_0\gamma}\delta_\gamma) \Rightarrow \\ L &= \alpha(I - \beta e^{i\omega_0\gamma}\tau_\gamma) =: \alpha(I - Q) \end{aligned}$$

where τ denotes translation. Here α, β and γ are stochastic and one of them is dependent on the other two. The model was achieved through measurements and is a three path model for microwave line of sight communication. We have already seen in a previous chapter how we should choose the power density spectrum of the transmitted signal. If we are able to estimate the channel conditions in the receiver we could try to find the inverse of L. If $|\beta| < 1$, we could express the inverse as a Neumann series

$$L^{-1} = \alpha^{-1}\sum_{k=0}^{\infty} Q^k$$

This could be implemented as a recursive net shown in figure 2.8. If $\|Q\|$ is large the noise term will be amplified and $\mathrm{Var}[\lambda]$ will become quite large. Thus the signal should be filtered before the inverse is calculated.

Let us study an example with this type of transfer function, in which we use the eight first Meyer wavelet packets. The spectrum of the transmitted signal has not been shaped and the signal set is $S = \{\sum_{j=0}^{7} a_j\psi^j \,:\, a_j \in$

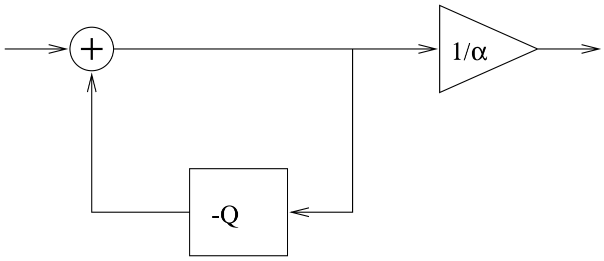

Figure 2.8: The inverse channel feed-back diagram.

$\{-1, 1\}\}$. The channel applies the operator $L = \alpha(I + \beta\tau_\gamma)$. No inverse is calculated in the receiver. Two examples are shown in figure 2.9 and figure 2.10 respectively.

We can see that the error rate decays slowly w.r.t the SNR when $|\beta|$ is large. The carriers do not only interfere with themselves but also with each other. In other words we have ICI in our signalling. This can be avoided to the cost of bandwidth or bit rate if we remove half of the packets. If we study figure 3.2, we see that the support of the Fourier transform of a wavelet packet is mainly only shared by its neighbours. If we go from low to high frequency and remove every second wavelet packet ICI could be avoided.

2.7 Digital implementation

So far we have assumed the receiver to be analog. The receiver performs the scalar product in $L^2(\mathbb{R})$. Since the Meyer wavelet (packets) are band limited we could take advantage of this fact. We shall see that we will get the same result in an AWGN channel if we use $l^2(\mathbb{Z})$ instead of $L^2(\mathbb{R})$. The Fourier transform of the Meyer scaling function is

$$\hat{\varphi}(\xi) = \begin{cases} 1 & \text{if } |\xi| \leq \frac{2\pi}{3} \\ \cos(\frac{\pi}{2}\nu(\frac{3|\xi|}{2\pi} - 1)) & \text{if } \frac{2\pi}{3} < |\xi| < \frac{4\pi}{3} \\ 0 & \text{otherwise} \end{cases}$$

where $\nu(x) = x^4(35 - 84x + 70x^2 - 20x^3)$. It should be noted that this is not the only choice of ν. This particular choice gives a function $\hat{\varphi} \in C^3(\mathbb{R})$. The regularity of $\hat{\varphi}$ is coupled to the asymptotic decay of φ. Let the low pass filter be $m_0(\xi) = \sum_{k\in\mathbb{Z}} \alpha_k e^{ik\xi}$. Using the MRA property and the above, we get

$$\hat{\varphi}(2\xi) = m_0(\xi)\hat{\varphi}(\xi) = m_0(\xi)\chi_{[-\pi,\pi]}(\xi)$$

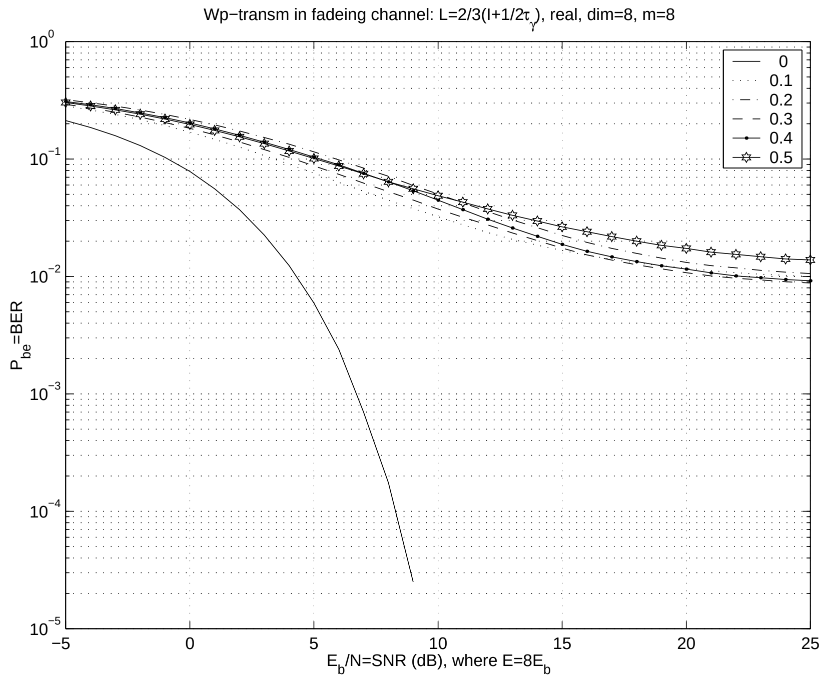

Figure 2.9: Bit error rate for communication using 8 Meyer wavelet packets, with bit rate $m = 8$. The channel is $L = 2/3(I + 1/2\tau_\gamma)$. Six different, γ. Note that on the x-axis is $E_b/N_0 = E/(8N_0)$.

The low pass coefficients are calculated from

$$\begin{aligned}\alpha_k =& \frac{1}{2\pi}\int_{-\pi}^{\pi} m_0(\xi)e^{-ik\xi}d\xi = \frac{1}{2\pi}\int_{-\pi}^{\pi} \hat{\varphi}(2\xi)e^{-ik\xi}d\xi = \\ & \frac{1}{2\pi}\int_{\mathbb{R}} \frac{1}{2}\hat{\varphi}(\xi)e^{i(-k/2)\xi}d\xi = \frac{1}{2}\varphi(-\frac{k}{2})\end{aligned} \tag{2.1}$$

The wavelet packets are calculated from

$$\begin{aligned}\hat{\psi}^{2n}(2\xi) &= m_0(\xi)\hat{\psi}^n(\xi) \\ \hat{\psi}^{2n+1}(2\xi) &= m_1(\xi)\hat{\psi}^n(\xi)\end{aligned}$$

where m_1 is the high pass filter,

$$m_1(\xi) = \sum_{k\in\mathbb{Z}} \beta_k e^{ik\xi}, \text{ where } \beta_k = (-1)^{k+1}\overline{\alpha_{1-k}}$$

Inverse Fourier transformation of this gives

$$\psi^{2n}(x) = 2\sum_{k\in\mathbb{Z}} \alpha_k \psi^n(2x+k) \tag{2.2}$$

$$\psi^{2n+1}(x) = 2\sum_{k\in\mathbb{Z}} \beta_k \psi^n(2x+k) \tag{2.3}$$

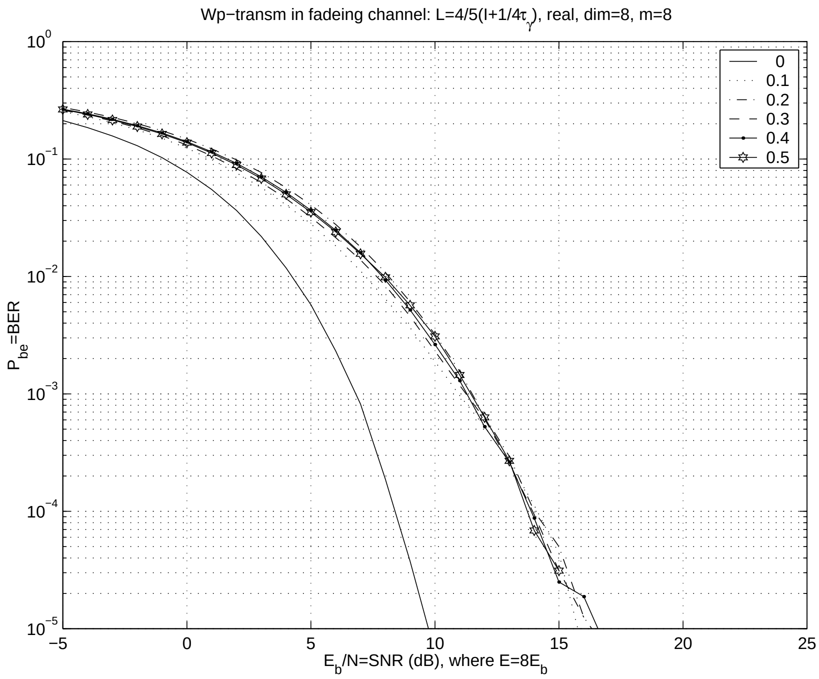

Figure 2.10: Bit error rate for communication using 8 Meyer wavelet packets, with bit rate $m = 8$. The channel is $L = 4/5(I + 1/4\tau_\gamma)$. Six different, γ. Note that on the x-axis is $E_b/N_0 = E/(8N_0)$.

We have the following conditions on the filter coefficients.

$$\sum_{l\in\mathbb{Z}} \alpha_{l-2k}\overline{\alpha_l} = \frac{1}{2}\delta_{k,0} \qquad \forall k \in \mathbb{Z} \tag{2.4}$$

$$\sum_{l\in\mathbb{Z}} \beta_{l-2k}\overline{\beta_l} = \frac{1}{2}\delta_{k,0} \qquad \forall k \in \mathbb{Z} \tag{2.5}$$

$$\sum_{l\in\mathbb{Z}} \alpha_{l-2k}\overline{\beta_l} = 0 \qquad \forall k \in \mathbb{Z} \tag{2.6}$$

If the received signal is a linear combination of wavelet packets. We have in the previous chapters for the decision variable calculated the scalar product $\langle \psi^n(\cdot - k), \psi^m \rangle_{L^2}$. Now we want to show that this scalar product is equal to $\langle \psi^n_{N+b,k}, \psi^m_{N+b,0} \rangle_{l^2}$, $\forall b \in \mathbb{N}$, when $n, m \in \{0, \ldots, 2^{N-1} - 1\}$. Let $A(N)$ denote the statement

$$A(N): \ \langle \psi^n_{N,k}, \psi^m_{N,0} \rangle_{l^2} = \delta_{k,0}\delta_{n,m}, \ n, m \in \{0, \ldots, 2^{N-1} - 1\}$$

where $\{\psi^n\}_n$ are the Meyer wavelet packets.

Proposition 1. *$A(N)$ is true* $\forall N \in \mathbb{N} \setminus \{0\}$

Proof. (i) Claim $A(1)$ is true.
Proof of claim: The statement reads $A(1):\ \langle \psi^0_{1,k}, \psi^0_{1,0}\rangle_{l^2} = \delta_{k,0}$

$$\langle \psi^0_{1,k}, \psi^0_{1,0}\rangle_{l^2} = \frac{1}{2}\sum_{l\in\mathbb{Z}} \psi^0(l/2-k)\overline{\psi}^0(l/2) = \{(2.1)\} =$$
$$2\sum_{l\in\mathbb{Z}} \alpha_{l-2k}\overline{\alpha}_l = \{(2.4)\} = \delta_{k,0}$$

(ii) Assume $A(N)$ is true. Show $A(N) \Rightarrow A(N+1)$. The statement reads

$$A(N+1):\ \langle \psi^n_{N+1,k}, \psi^m_{N+1,0}\rangle_{l^2} = \delta_{k,0}\delta_{n,m},\ \ n,m \in \{0,\dots,2^N-1\}$$

Let $n,m \in \{0,\dots,2^N-1\}$ and let $\gamma^{(m)}_p$ denote α_p if m is even and β_p if m is odd, then

$$\begin{aligned}
&\langle \psi^n_{N+1,k}, \psi^m_{N+1,0}\rangle_{l^2} = \\
&\frac{1}{2^{N+1}}\sum_{l\in\mathbb{Z}} \psi^n\left(\frac{l}{2^{N+1}}-k\right)\overline{\psi}^m\left(\frac{l}{2^{N+1}}\right) = \{(2.2),(2.3) \text{ and Fubini}\} = \\
&\frac{1}{2^{N-1}}\sum_{p\in\mathbb{Z}}\sum_{q\in\mathbb{Z}} \gamma^{(n)}_p\overline{\gamma}^{(m)}_q \sum_{l\in\mathbb{Z}} \psi^{\lfloor n/2\rfloor}\left(\frac{l}{2^N}-2k+p\right)\overline{\psi}^{\lfloor m/2\rfloor}\left(\frac{l}{2^N}+q\right) = \\
&2\sum_{p,q\in\mathbb{Z}} \gamma^{(n)}_p\overline{\gamma}^{(m)}_q \delta_{p-2k,q}\delta_{\lfloor n/2\rfloor,\lfloor m/2\rfloor} = 2\delta_{\lfloor n/2\rfloor,\lfloor m/2\rfloor}\sum_{q\in\mathbb{Z}} \gamma^{(n)}_{q+2k}\overline{\gamma}^{(m)}_q
\end{aligned} \tag{2.7}$$

The equality between the third and fourth row comes from $A(N)$, since $\lfloor n/2\rfloor, \lfloor m/2\rfloor \in \{0, 2^{N-1}-1\}$. Fubini is appropriate since $\{\gamma^{(n)}_p \psi^{\lfloor n/2\rfloor}(\frac{l}{2^N} - 2k+p)\}_p$, $\{\psi^{\lfloor n/2\rfloor}(\frac{l}{2^N}-2k+p)\}_p \in l^1(\mathbb{Z})$. $\delta_{\lfloor n/2\rfloor,\lfloor m/2\rfloor} = 0$, when $n \notin \{m-1, m, m+1\}$. If $\lfloor n/2\rfloor = \lfloor m/2\rfloor$ and $n \neq m$ then $\gamma^{(m)}_p\overline{\gamma}^{(n)}_q$ is either $\alpha_p\overline{\beta}_q$ or $\beta_p\overline{\alpha}_q$. It follows from (2.6) that (2.7) is zero in this case. If $n = m$ then $\gamma^{(m)}_p\overline{\gamma}^{(n)}_q$ is either $\alpha_p\overline{\alpha}_q$ or $\beta_p\overline{\beta}_q$. It follows from (2.4) and (2.5) that (2.7) is equal to $\delta_{k,0}$ in this case.
(iii) Thus by the induction axiom

$$\forall N \in \mathbb{N}\setminus\{0\}:\ A(N)$$

$\square$

If the received signal is a linear combination of $\psi^0,\dots,\psi^{2^N-1}$, then the receiver can without loss of information sample the received signal at the points $\{\frac{l}{2^{N+1+b}}\}_{l\in\mathbb{Z}}$, $\forall b \in \mathbb{N}$ and calculate the scalar product in $l^2(\mathbb{Z})$. The decision variable λ, will follow the same distribution in an AWGN channel.

2.8 An implementation with DWT

In this chapter we consider an implementation aspect for the transmitter and the receiver. When we discussed two or many-dimensional communication before, we used the wavelet packets. In the discussions below we need a finite length low-pass filter. Let the filter length be dyadic. Let Y be the vector space, over $\mathbb{R}$ or $\mathbb{C}$ spanned by the first 2^L wavelet packets at level 0, i.e $Y = \text{span}\{\psi_0^0, \ldots, \psi_0^{2^L-1}\}$. The signal set, S, is again some some dyadic number of points in Y. The number of points in S should at least be 2^{L+1}, say 2^m. The mapping onto S is done as before. The wavelet packets on level 0 can be written as a sum of functions on level L, since $W_L^0 = W_0^0 \oplus \ldots \oplus W_0^{2^L-1}$. Thus $Y \subset W_L^0 \equiv V_L$. So the transmitter needn't generate the 2^L different waveforms in Y, but a single waveform $\varphi_{L,0} \in V_0$, with translates. The receiver need only one waveform to perform its scalar product, $\langle r, \varphi_{L,k}\rangle$. The transmitter use the composition formula.

$$
\begin{aligned}
c_{j,l} = \langle s, \psi_{j,l}^n \rangle = \sqrt{2} \sum_{k \in \mathbb{Z}} [c_{j-1,k}\alpha_{2k-l} + d_{j-1,k}\beta_{2k-l}] = \\
\sqrt{2} \sum_{k \in \mathbb{Z}} [\langle s, \psi_{j-1,k}^{2n}\rangle \alpha_{2k-l} + \langle s, \psi_{j-1,k}^{2n+1}\rangle \beta_{2k-l}]
\end{aligned}
$$

The receiver applies the decomposition formulas.

$$
\begin{aligned}
c_{j-1,k} &= \langle r, \psi_{j-1,k}^{2n}\rangle = \sqrt{2} \sum_{m \in \mathbb{Z}} [\bar{\alpha}_m c_{j,2k-m}] = \sqrt{2} \sum_{m \in \mathbb{Z}} [\bar{\alpha}_m \langle r, \psi_{j,2k-m}^n\rangle] \\
d_{j-1,k} &= \langle r, \psi_{j-1,k}^{2n+1}\rangle = \sqrt{2} \sum_{m \in \mathbb{Z}} [\bar{\beta}_m c_{j,2k-m}] = \sqrt{2} \sum_{m \in \mathbb{Z}} [\bar{\beta}_m \langle r, \psi_{j,2k-m}^n\rangle]
\end{aligned}
$$

So all the time we need only transmit the scaling function independently of how many dimensions Y has. The difference is in the map from the information sequence to the signal we are transmitting. In the one dimensional case we are mapping locally to $\varphi_{0,k}$ and in the many dimensional case we are mapping to a linear combination of $\{\varphi_{L,k}\}_k$.

2.9 Comments and comparison with OFDM

We have discussed how wavelets can be used for communication. In most examples we have used the Meyer wavelets. These have the feature that they are localised in frequency. This means that we can shape the spectrum of our carriers by using a linear combination of some Meyer wavelet packets. Which wavelet to use depends on the particular situation. Alternatively one can use biorthogonal wavelets. The signal space is then $Y = \text{span}\{\varphi, \psi\}$ and the receiver receiver projects the received signal onto $\tilde{Y} = \text{span}\{\tilde{\varphi}, \tilde{\psi}\}$.

OFDM is described in the chapter called "Modulation and demodulation". In an AWGN channel OFDM and wavelet packet modulation will have the same symbol error rate. If we use frequency localised wavelet packets such as Meyer the bandwidth efficiency of WP-transmission is much higher than OFDM. OFDM can use other carriers than $\chi_{[0,1)}(t)e^{i\omega_j t}$ to achieve higher band width efficiency, to the cost of bit rate or SNR. For example $g(t)e^{i\omega_j t}$, where $g(t) = \chi_{[a,b)}(t)$ for $t \in [a, b]$, $0 < a < b < 1$. Outside $[a, b]$, g decays smoothly to zero and $g(t) = 0$ when $t \notin [0, 1]$.

OFDM is often implemented using a set of trigonometric functions orthogonal on an interval $[a, 1]$, where $0 \leq a < 1$. Then a cyclic extension is performed, i.e $s(t) = s(t + 1 - a)$ for $t \in [0, a]$. This reduces the multi-path effects, since only the interval $[a, 1]$ is used by the receiver. Any multi-path copy of s_k arriving at a time shorter than a will only interfere with s_k and not s_l, $l \neq k$, thus reducing ISI in a multi-path channel, but it will still experience ICI. If we want to use this idea for wavelets or wavelet packets we could for example remove all packets with j odd in $\psi^k_{i,j}$. We have said before that for localised wavelets ICI can be reduced by removing $\psi^k_{i,j}$ for some specific k. OFDM is sometimes used together with DFT, see chapter on "Modulation and demodulation". Wavelets or wavelet packets can be used in a similar way by using DWT.

OFDM is quite sensitive for frequency errors, see figure 2.11 and figure 2.12. The graphs were calculated in a similar fashion as the simulation made for wavelet packet transmission in chapter "Frequency instable receiver", i.e. a Monte Carlo simulation with 20000 transmitted symbols. Figure 2.11 shows a simulation when the signal set is

$$S = \{\chi_{[0,1)}(t) \sum_j a_j e^{i\omega_j t} : a_j \in \{\pm 1, \pm i\}, \omega_j = 2\pi j\}$$

Let s denote the transmitted signal, $s = \sum_k s_k$, where

$$s_k = \chi_{[0,1)}(t) \sum_j c_j e^{i\omega_{j-1} t}$$

Let λ^j_k denote the jth component of the m-dimensional decision variable λ_k calculated by the receiver.

$$\mathrm{E}[\lambda^j_k] = \sum_{n=0}^{m-1} \frac{c_n}{i(\Delta\omega + \omega_j - \omega_n)}(1 - e^{-i(\Delta\omega + \omega_j - \omega_n)})$$

In figure 2.12 the signal set is

$$S = \{\chi_{[0,1)}(t) \sum_j a_j e^{i\omega_j t} : a_j \in \{\pm 1\}, \omega_j = 2\pi j\}$$

Figure 2.13, which shows WP-transmission, is the same as figure 2.7 and is added for comparison. For zero frequency error we can see that OFDM will give the same BER as in WP-transmission.

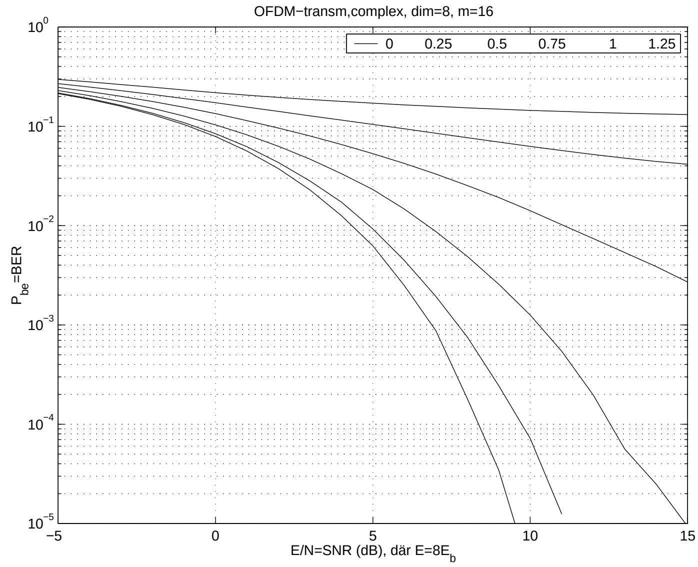

Figure 2.11: Bit error rate for OFDM communication with bit rate $m = 16$, dimension $d = 8$, (complex). The receiver frequency differs from the transmitter frequency. P_{be} for six different frequency shifts are plotted. Note that the x-axis is $E_b/N_0 = E/(16N_0)$.

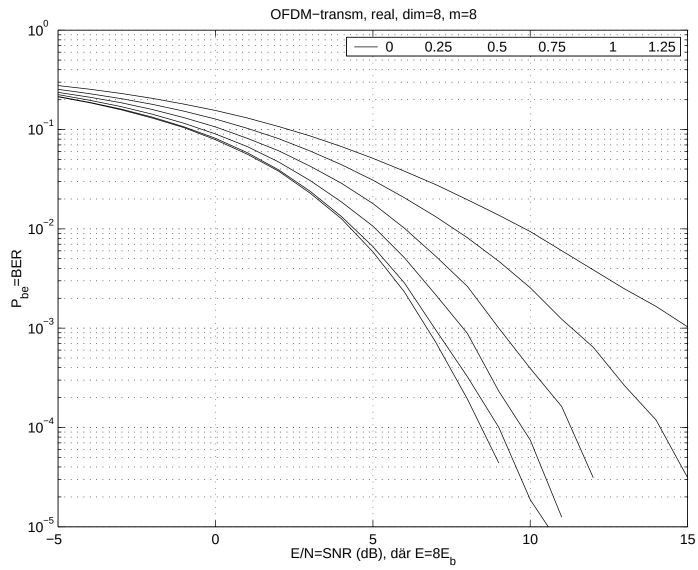

Figure 2.12: Bit error rate for OFDM communication with $m = 8$, $d = 8$, (real). The receiver frequency differs from the transmitter frequency. P_{be} for six different $\Delta\omega$ are plotted. Note that the x-axis is $E_b/N_0 = E/(8N_0)$.

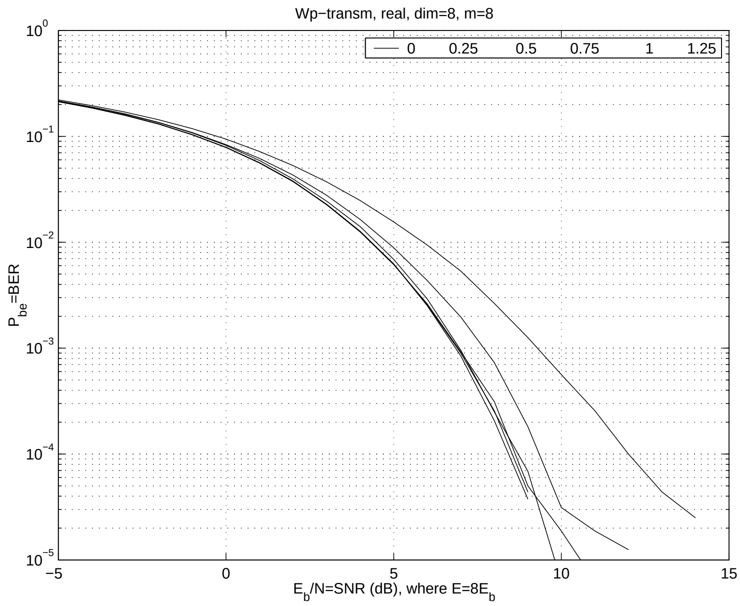

Figure 2.13: Bit error rate for communication using 8 Meyer wavelet packets, with bit rate $m = 8$. The receiver frequency is different from the transmitter frequency. Six different $\Delta\omega$. Note that the x-axis is $E_b/N_0 = E/(8N_0)$.

Chapter 3

Numerical computations with the Meyer wavelet

During the work an error was discovered in Matlab's Wavelet toolbox regarding Meyer wavelets. In the toolbox the wavelet packets are calculated recursively by using the filter coefficients.

The Meyer scaling function is given as closed expression on the transform side. The scaling function can be calculated either by DFT of this closed expression or by a number of convolutions of the low pass filter coefficients. The number of coefficients are infinitely many, which means that we have to truncate the sum somewhere. In this chapter we will estimate the error of the numerically calculated Meyer scaling function. At the end of the chapter there are some comments about the toolbox.

Let us first see what the Meyer wavelet packets look like. Figure 3.1 shows the first eight packets and figure 3.2 shows the absolute value of the Fourier transformed packets.

The Fourier transform of the Meyer scaling function is given by:

$$\hat{\varphi}(\xi) = \begin{cases} 1 & \text{if } |\xi| \le \frac{2\pi}{3} \\ \cos(\frac{\pi}{2}\nu(\frac{3|\xi|}{2\pi} - 1)) & \text{if } \frac{2\pi}{3} < |\xi| < \frac{4\pi}{3} \\ 0 & \text{o.w} \end{cases}$$

Where ν is a function on the interval $[0, 1]$, s.t

$$\nu(x) + \nu(1 - x) = 1 \ \forall x \in [0, 1] \ \text{ and } \ \nu(0) = 0, \ \nu(1) = 1$$

We will focus on the most common choice of ν, namely

$$\nu(x) = x^4(35 - 84x + 70x^2 - 20x^3)$$

This particular choice gives a function $\hat{\varphi} \in C^3(\mathbb{R})$. The regularity of $\hat{\varphi}$ is coupled to the asymptotic decay of φ. Let m_0 denote the Meyer low pass

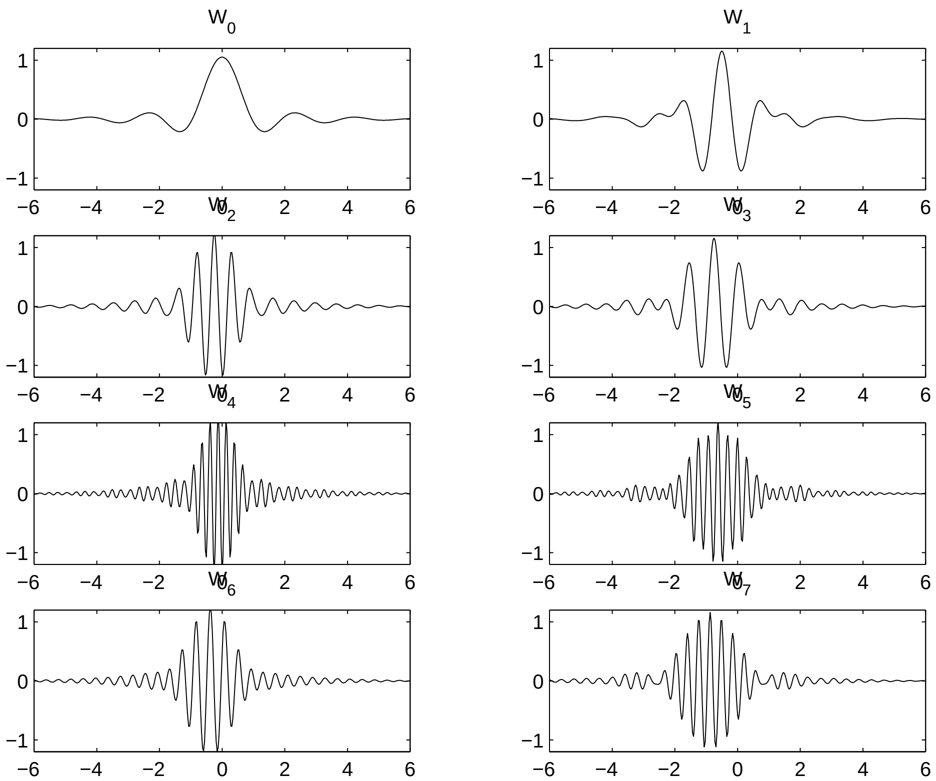

Figure 3.1: The first eight Meyer wavelet packets.

filter. Since m_0 is 2π-periodic it could be expressed as

$$m_0(\xi) = \sum_{k\in\mathbb{Z}} \alpha_k e^{ik\xi}$$

Figure 3.3 shows the low pass filter m_0, which is equal to $\varphi(2\cdot)$ in the interval $[-\pi, \pi]$.

Using the MRA property and the above, we get

$$\hat{\varphi}(2\xi) = m_0(\xi)\hat{\varphi}(\xi) = m_0(\xi)\chi_{[-\pi,\pi]}(\xi)$$

Thus

$$m_0(\xi) = \begin{cases} 1 & \text{if } |\xi| \leq \frac{\pi}{3} \\ \cos(\frac{\pi}{2}\nu(\frac{3|\xi|}{\pi} - 1)) & \text{if } \frac{\pi}{3} < |\xi| < \frac{2\pi}{3} \\ 0 & \text{if } \frac{2\pi}{3} \leq |\xi| \leq \pi \end{cases}$$

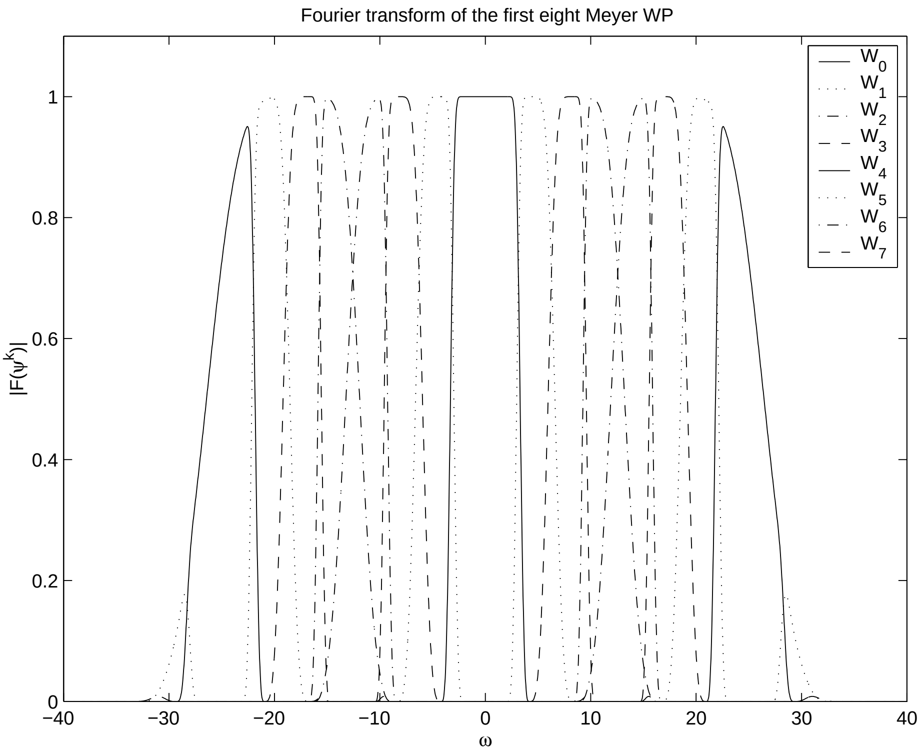

Figure 3.2: Absolute value of the Fourier transform of the first eight Meyer wavelet packets.

The low pass coefficients are calculated from

$$\alpha_k = \frac{1}{2\pi}\int_{-\pi}^{\pi} m_0(\xi)e^{-ik\xi}d\xi = \frac{1}{2\pi}\int_{-\pi}^{\pi}\hat{\varphi}(2\xi)e^{-ik\xi}d\xi = \frac{1}{2\pi}\int_{\mathbb{R}}\frac{1}{2}\hat{\varphi}(\xi)e^{i(-k/2)\xi}d\xi = \frac{1}{2}\varphi(-\frac{k}{2})$$

$\hat{\varphi}$ has compact support which means that φ doesn't have compact support. In other words we need infinitely many coefficients to specify the Fourier series of m_0. Inverse transform of the equation above gives

$$\frac{1}{2}\varphi(x/2) = \check{m}_0 * \varphi(x) = \check{m}_0 * \frac{\sin(\pi\cdot)}{\pi\cdot}(x) \Leftrightarrow$$
$$\varphi(x/2) = 2\sum_{k\in\mathbb{Z}}\alpha_k\varphi(x+k) = 2\sum_{k\in\mathbb{Z}}\alpha_k\frac{\sin(\pi(x+k))}{\pi(x+k)}$$

Suppose that we only know finitely many coefficients, say $\{\alpha_k\}_{k=-N}^{N}$.

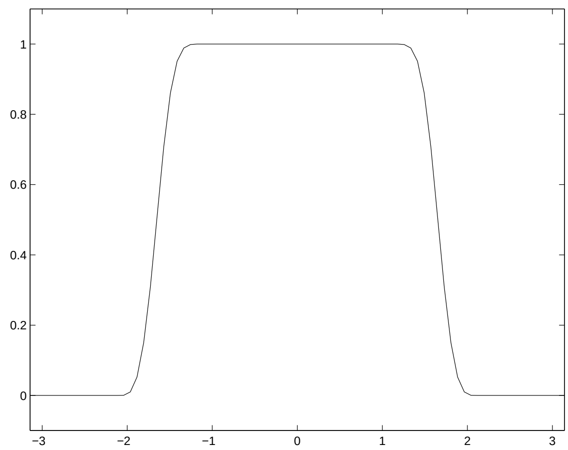

Figure 3.3: The Meyer low pass filter in the interval $[-\pi, \pi]$

What error do we get in our calculation of φ? Let $n \in \mathbb{Z}$. Then we have.

$$\varphi(n/4) = 2\sum_{k\in\mathbb{Z}} \alpha_k \varphi(\frac{n+2k}{2}) = 4\sum_{k\in\mathbb{Z}} \alpha_k \alpha_{-n-2k}$$

If we are going to truncate this sum, we need an estimation of the α_ks that we are lacking. We have for $k > 0$, $0 < p \leq 4$

$$\alpha_k = \frac{1}{2\pi}\int_{-2\pi/3}^{2\pi/3} m_0(\xi)e^{-ik\xi}d\xi = \frac{1}{2\pi(ik)^p}\int_{-2\pi/3}^{2\pi/3} m_0^{(p)}(\xi)e^{-ik\xi}d\xi$$

This gives an estimate for $k > 0$, $0 < p \leq 4$, we have:

$$|\alpha_k| \leq \frac{1}{\pi k^p}\int_{\pi/3}^{2\pi/3} |m_0^{(p)}(\xi)|d\xi := C_p/k^p$$

where we have used that $m_0^{(p)}(\pi/3) = m_0^{(p)}(2\pi/3) = 0$ for $p \in \{0,1,2,3\}$ and m_0 has continuous derivatives in the open interval $(\pi/3, 2\pi/3)$ and m_0 is even. In this way we can estimate $|\alpha_k|$ up to $p = 5$, where we get.

$$|\alpha_k| \leq \frac{1}{\pi k^5}\left(|m_0^{(4)}(2\pi/3)|\max_k |\sin(k2\pi/3)| + \int_{\pi/3}^{2\pi/3} |m_0^{(5)}(\xi)|d\xi\right) := C_5/k^5$$

where the second term has an asymptotic decay as $1/k^6$. So for $k \in 3\mathbb{N}$ the decay will be as $1/k^6$. Note that this means that the decay $1/k^5$ can not be improved. In this way we get the estimates shown in table 3.1 for $|\alpha_k| \leq C_p/|k|^p$, valid $\forall k$ if $p = 0$ and $\forall k \neq 0$ if $p > 0$.

These approximations are shown in figure 3.4 together with the numerically calculated coefficients.

p	C_p	best interval
0	2/3	$\{0\}$
1	$1/\pi$	$\{1,\dots,5\}$
2	1.68	$\{6,7,8\}$
3	13.47	$\{9,10\}$
4	139.22	$\{11,\dots,14\}$
5	1995.33	$\mathbb{N}\setminus\{0,\dots,14\}$

Table 3.1: Coefficients for estimation of the decay of $|\alpha_k|$

Let $\tilde{\varphi}$ denote the estimation of φ, where the sum is taken by finitely many α_ks. We have, for $-N \le n \le N$.

$$\tilde{\varphi}(n/4) = 4 \sum_{k=-\lfloor \frac{N+n}{2} \rfloor}^{\lfloor \frac{N-n}{2} \rfloor} \alpha_k \alpha_{-n-2k} =$$
$$\varphi(n/4) - 4 \sum_{k=-\infty}^{-\lfloor \frac{N+n}{2} \rfloor - 1} \alpha_k \alpha_{-n-2k} - 4 \sum_{k=\lfloor \frac{N-n}{2} \rfloor + 1}^{\infty} \alpha_k \alpha_{-n-2k} :=$$
$$\varphi(n/4) + R(n,N)$$

Since φ is symmetric, it suffices to estimate $R(n,N)$ for $n \ge 0$. For $0 \le n \le N-1$ and $0 \le p \le 5$, we get the following estimate of $R(n,N)$.

$$|R(n,N)| \le 4 \left(\sum_{k=\lfloor \frac{N+n}{2} \rfloor + 1}^{\infty} |\alpha_k||\alpha_{2k-n}| + \sum_{k=\lfloor \frac{N-n}{2} \rfloor + 1}^{\infty} |\alpha_k||\alpha_{2k+n}| \right) \le$$
$$2^{2-p} C_p^2 \left(\sum_{k=\lfloor \frac{N+n}{2} \rfloor + 1}^{\infty} |k|^{-p} |k - n/2|^{-p} + \sum_{k=\lfloor \frac{N-n}{2} \rfloor + 1}^{\infty} |k|^{-p} |k + n/2|^{-p} \right) \le$$
$$2^{2-p} C_p^2 \left(\int_{\lfloor \frac{N+n}{2} \rfloor - n/2}^{\infty} \frac{dk}{k^p (k + n/2)^p} + \int_{\lfloor \frac{N-n}{2} \rfloor}^{\infty} \frac{dk}{k^p (k + n/2)^p} \right)$$

We can use $k+n/2 \ge k$ and replace $k+n/2$ with k. With this replacement we get

$$|R(n,N)| \le 2^{2-p} C_p^2 \left(\int_{\lfloor \frac{N+n}{2} \rfloor - n/2}^{\infty} \frac{dk}{k^{2p}} + \int_{\lfloor \frac{N-n}{2} \rfloor}^{\infty} \frac{dk}{k^{2p}} \right) =$$
$$2^{2-p} C_p^2 \left(\left(\left\lfloor \frac{N+n}{2} \right\rfloor - \frac{n}{2} \right)^{1-2p} + \left\lfloor \frac{N-n}{2} \right\rfloor^{1-2p} \right)$$

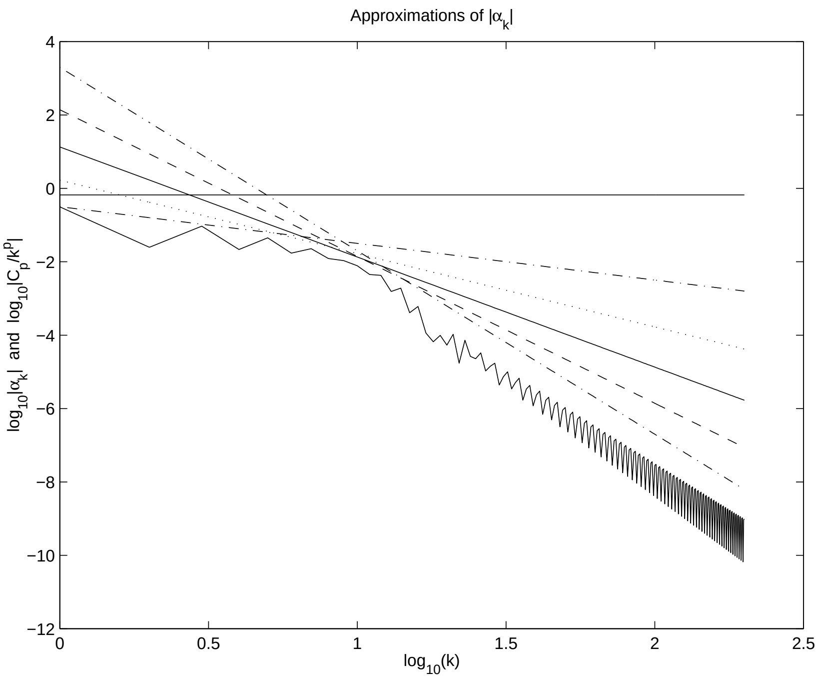

Figure 3.4: Approximations of $|\alpha_k|$

The error for this replacement will be small if n is small. Otherwise we calculate the integral and for $1 \leq p \leq 5$ we get

$$|R(n,N)| \leq 2^{2-p} C_p^2 \left(\int_{\lfloor \frac{N+n}{2} \rfloor - n/2}^{\infty} \frac{dk}{k^p (k+n/2)^p} + \int_{\lfloor \frac{N-n}{2} \rfloor}^{\infty} \frac{dk}{k^p (k+n/2)^p} \right) =$$

$$2^{2-p} C_p^2 \left(\frac{2}{n}\right)^{2p-2} \left[\sum_{\substack{j=0 \\ j \neq p-1}}^{2p-1} \binom{2p-2}{j} \frac{(-1)^j}{p-j-1} \cdot \left\{ \right. \right.$$

$$\left. \left(1 + \frac{n}{2\left(\lfloor \frac{N+n}{2} \rfloor - \frac{n}{2}\right)}\right)^{p-j-1} + \left(1 + \frac{n}{2 \lfloor \frac{N-n}{2} \rfloor}\right)^{p-j-1} - 2 \right\} +$$

$$\left. \binom{2p-2}{p-1} (-1)^{p-1} \log \left(\left(1 + \frac{n}{2\left(\lfloor \frac{N+n}{2} \rfloor - \frac{n}{2}\right)}\right) \left(1 + \frac{n}{2 \lfloor \frac{N-n}{2} \rfloor}\right) \right) \right]$$

For $0 \leq n \leq N - 29$ the approximation $|\alpha_k| \leq C_p/k^5$ will give the best estimate. For $N - 29 < n \leq N$ we would get a better estimate if we also include some of the other approximations of $|\alpha_k|$ and single out those terms in the sum.

3.1 Wavelet toolbox

Now some words about the Matlab wavelet toolbox (version 1.2). If one wants the Meyer scaling function and wavelet in the toolbox two options are given. The first is calculated via DFT of closed expressions that the Fourier transform of scaling function and wavelet have. The other way is via a number of convolutions of finitely many filter coefficients. If one is interested in the Meyer wavelet packets only the second option is available. The main problem is not the way it is calculated, but the lack of sufficient accuracy for the filter coefficients. The packets calculated with these coefficients are not orthogonal. For example if we calculate $\langle \psi_{00}^2, \psi_{00}^6 \rangle$ using the functions given by the toolbox, we will get the answer 0.16, so the orthogonality is lost. As we have said before $\alpha_k = \alpha_{-k} = 1/2\varphi(k/2)$. Let dmey denote the filter coefficients in the toolbox and α_k are the coefficients numerically calculated to an accuracy of order 10^{-10}. Figure 3.5 shows the difference between α_k and dmey. The figure also shows a comparison with $1/2\varphi(k/2)$ calculated by the toolbox in the first way, i.e via IDFT.

Another, smaller, problem is when the wavelet packets are calculated using the second option in the toolbox. We get $\pm\psi^n(t_0 - t)$ instead of $\psi^n(t)$, plus for some packets and minus for some and $t_0 \approx 30$. If we calculate the first two packets using the first option (IDFT), we will get the correct packets $\psi^n(t)$ for, $n = 0, 1$.

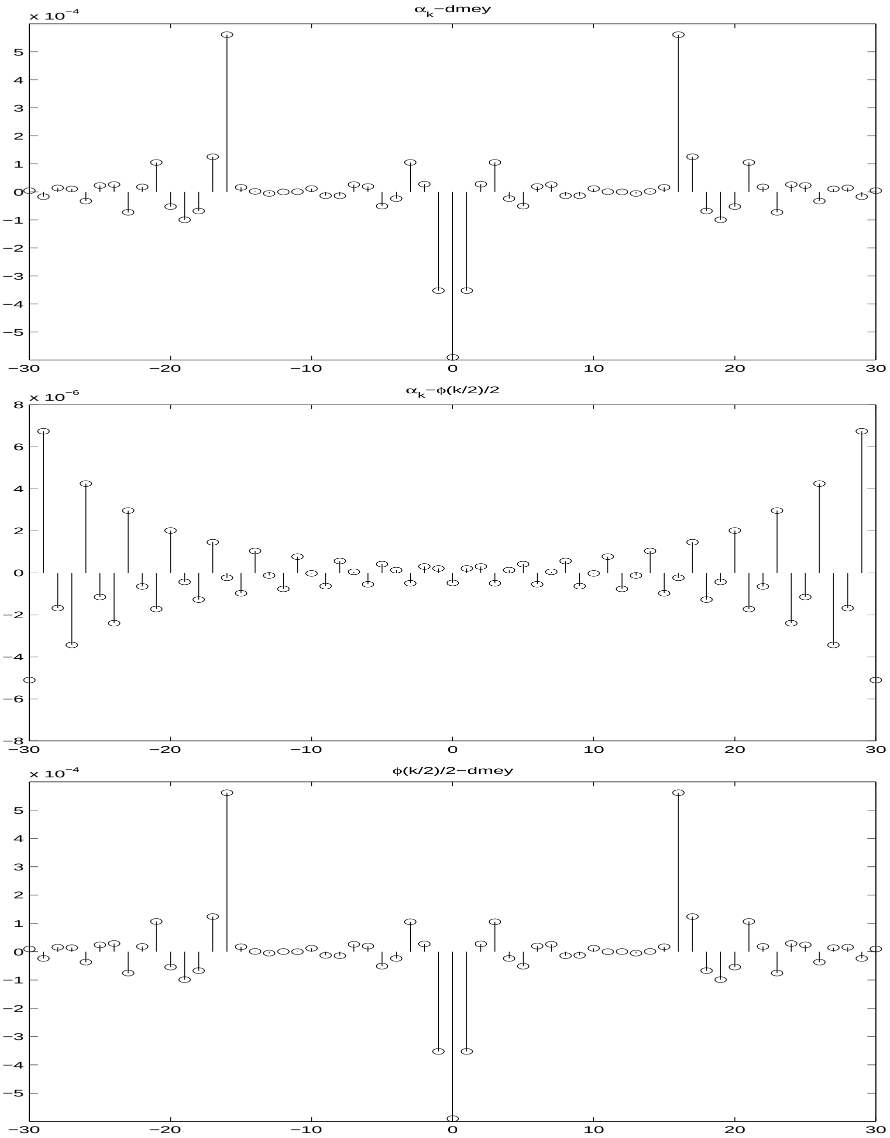

Figure 3.5: Errors in approximations of α_k

Bibliography

[AZ98] L. Ahlin and J. Zander. *Principles of Wireless Communication.* Studentlitteratur, 2 edition, 1998.

[ELS+96] O. Edfors, D. Landström, M. Sandell, F. Sjöberg, and JJ van de Beek. An introduction to orthogonal frequency-division multiplexing. Technical report, Luleå University of Technology, 1996.

[HW96] E. Hernandez and G. Weiss. *A First Course on Wavelets.* CRC Press, 1 edition, 1996.

[Law97] E. Lawrey. The suitability of ofdm as a modulation technique for wireless telecommunications, with a cdma comparison. Master's thesis, James Cook University, October 1997.

[LMR97] A. K. Louis, P. Maaß, and A. Rieder. *Wavelets Theory and Applications.* John Wiley & Sons, 1 edition, 1997.

[Mal99] S. Mallat. *A Wavelet Tour of Signal Processing.* Academic Press, 2 edition, 1999.

[MW95] R. N. McDonough and A. D. Wahlen. *Detection of Signals in Noise.* Academic Press, 2 edition, 1995.

[New98] H. M. Newlin. Developments in the use of wavelets in communication systems. In *MILCOM 98. Proceedings., IEEE*, volume 1, pages 343–349, October 1998.

[Pro95] John G. Proakis. *Digital Communications.* McGraw-Hill, 3 edition, 1995.

[Rum79] W. D. Rummler. A new selective fading model: Application to propagation data. *Bell System Technical Journal*, 58:1037–1071, May-June 1979.

[Wor96] G. W. Wornell. Emerging applications of multirate signal processing and wavelets in digital communications. *Proceedings of the IEEE*, 84:586–603, April 1996.