NETWORKS AND TELECOMMUNICATIONS SERIES

Digital Communications 2

Digital Modulations

Mylène Pischella and Didier Le Ruyet

iSTE

WILEY

Digital Communications 2

Series Editor
Pierre-Noël Favennec

Digital Communications 2

Digital Modulations

Mylène Pischella
Didier Le Ruyet

ISTE

WILEY

First published 2015 in Great Britain and the United States by ISTE Ltd and John Wiley & Sons, Inc.

ISTE Ltd
27-37 St George’s Road
London SW19 4EU
UK

www.iste.co.uk

John Wiley & Sons, Inc.
111 River Street
Hoboken, NJ 07030
USA

www.wiley.com

Library of Congress Control Number: 2015946706

British Library Cataloguing-in-Publication Data
A CIP record for this book is available from the British Library
ISBN 978-1-84821-846-8

Contents

Preface

Humans have always used communication systems: in the past, native Americans used clouds of smoke, then Chappe invented his telegraph and Edison the telephone, which has deeply changed our lifestyle. Nowadays, smartphones enable us to make calls, watch videos and communicate on social networks. The future will see the emergence of the connected man and wider applications of smart objects. All current and future communication systems rely on a digital communication chain that consists of a source and a destination separated by a transmission channel, which may be a portion of a cable, an optical fiber, a wireless mobile or satellite channel. Whichever the channel, the processing blocks implemented in the communication chain have the same basis. This book aims at detailing them, across two volumes:

– the first volume deals with source coding and channel coding. After a presentation of the fundamental results of information theory, the different lossless and lossly source coding techniques are studied. Then, error-correcting-codes (block codes, convolutional codes and concatenated codes) are theoretically detailed and their applications provided;

– the second volume concerns the blocks located after channel coding in the communication chain. It first presents baseband and sine waveform transmissions. Then, the different steps required at the receiver to perform detection, namely synchronization and channel estimation, are studied. Two variants of these blocks which are used in current and future systems, multicarrier modulations and coded modulations, are finally detailed.

This book arises from the long experience of its authors in both the business and academic sectors. The authors are in charge of several diploma and higher-education teaching modules at *Conservatoire national des arts et métiers* (CNAM) concerning digital communication, information theory and wireless mobile communications.

The different notions in this book are presented with an educational objective. The authors have tried to make the fundamental notions of digital communications as understandable and didactic as possible. Nevertheless, some more advanced techniques that are currently strong research topics but are not yet implemented are also developed.

Digital Communications may interest students in the fields of electronics, telecommunications, signal processing, etc., as well as engineering and corporate executives working in the same domains and wishing to update or complete their knowledge on the subject.

The authors would like to thank their colleagues from CNAM, and especially from the EASY department.

Mylène Pischella would like to thank her daughter Charlotte and husband Benjamin for their presence, affection and support.

Didier Le Ruyet would like to thank his parents and his wife Christine for their support, patience and encouragements during the writing of this book.

Mylène PISCHELLA
Didier LE RUYET
Paris, France
August 2015

List of Acronyms

ADSL: assymetric digital subscriber line

ASK: amplitude shift keying

BER: bit error rate

CD: compact disc

CFM: constellation's figure of merit

CMT: cosine modulated multitone

DFE: decision feedback equalizer

DWMT: discrete wavelet multitone

EFM: eight-to-fourteen modulation

EGF: extended Gaussian function

FBMC: filter bank multicarrier

FDDI: fiber distributed data interface

FFT: fast Fourier transform

FIR: finite impulse response

FMT: filtered multi-tone

FSK: frequency shift keying

GMSK: Gaussian minimum shift keying

GSM: global system for mobile communications

HDMI: high-definition multimedia interface

IFFT: inverse fast Fourier transform

IIR: infinite impulse response

IOTA: isotropic orthogonal transform algorithm

MAP: maximum a posteriori

MIMO: multiple input, multiple output

ML: maximum likelihood

MLSE: maximum likelihood sequence estimator

MLT-3: multilevel transmit 3

MMSE: minimum mean square error

MSK: minimum shift keying

NRZ: non-return to zero

NRZI: non-return to zero inverted

OFDM: orthogonal frequency division multiple access

OQAM: offset quadrature amplitude modulation

PAPR: peak to average power ratio

PSK: phase shift keying

QAM: quadrature amplitude modulation

QPSK: quadrature phase shift keying

RFID: radio frequency identification

RLL: run length limited

RZ: return to zero

SCH: synchronization channel

SMT: staggered multitone

SNR: signal-to-noise ratio

VDSL: very-high-bit-rate digital subscriber line

ZF: zero forcing

ZF-DFE: zero-forcing decision feedback equalizer

Notations

B: bandwidth

B_c: coherence bandwidth

$b(t)$: baseband additive white Gaussian noise

$c_k = a_k + jb_k$: baseband complex symbol, with a_k its real part and b_k its imaginary part

d_{min}: minimum distance

$E[x]$: expected value of a random variable x

E_b: average energy per bit

E_s: average energy per symbol

f_c: carrier frequency of narrowband passband signal

$g_c(t)$: impulse response of the transmit baseband channel

$g(t)$: transmit filter

$g_r(t)$: matched filter at the receiver

$\gamma_{xx}(f)$: power spectrum density of a random process x

$h(t)$: impulse response of the narrowband passband channel

$H(z)$: linear digital equalization filter

N: noise power,or number of samples in an OFDM or FBMC symbol

N_{CP}: number of samples in OFDM cyclic prefix

$n(t)$: noise filtered by the receive filter

N_0: one-sided noise power spectrum density

M: alphabet size of the modulation's constellation

P: signal power

P_e: error probability per symbol

p: probability of a random variable

$p(t) = g(t) * g_c(t) * g_r(t)$: complete equivalent channel

Ψ: phase of the passband signal

$R_{ss}(t)$: autocorrelation function of random process s

$r(t)$: output signal of narrowband passband channel

$s(t)$: input signal of narrowband passband channel

T: symbol period

T_b: transmission duration of a bit

T_m: maximum delay of the channel

T_N: transmission duration of an OFDM or FBMC symbol

$u(t)$: step function

$x(t)$: transmitted signal on the baseband channel

$\mathbf{y}$: received signal at channel's output after matched filter and sampling

$y(t)$: received baseband signal after matched filter

Introduction

The fundamental objective of a communication system is to reproduce at a destination point, either exactly or approximately, a message selected at another point. This was theorized by Claude Shannon in 1948 [SHA 48].

The communication chain is composed of a source (also called transmitter) and a destination (also called receiver). They are separated by a transmission channel, which may, for instance, be a wired cable if we consider assymetric digital subscriber line (ADSL) transmission, optical fiber, a wireless mobile channel between a base station and a mobile terminal or between a satellite and its receiver, a hard drive and so forth. The latter example indicates that when we refer to *point*, we may consider either location or time. The main issue faced by communication systems is that the channel is subject to additional noise, and may also introduce some distortions on the transmitted signal. Consequently, advanced techniques must be implemented in order to decrease the impact of noise and distortions on the performances as much as possible, so that the receiver signal may be as similar as possible to the transmitted signal.

The performance of a transmission system is evaluated by either computing or measuring the error probability per received information bit at the receiver, also called the bit error rate. The other major characteristics of a communication system are its complexity, its bandwidth, its consumed and transmitted power and the useful data rate that it can transmit. The bandwidth of many communication systems is limited; it is thus highly important to maximize the spectral efficiency, which is defined as the ratio between the binary data rate and the bandwidth. Nevertheless, this should be done without increasing the bit error rate.

This book consists of two volumes. It aims at detailing all steps of the communication chain, represented in Figure I.1. Source and channel coding are deeply studied in the first volume: *Digital Communications 1: Source and Channel Coding*. Even though both volumes can be read independently, we will sometimes refer to some of the notions developed in the first volume. The present volume focuses on the following blocks: modulation, synchronization and equalization. Once source and channel coding have been performed, data have first been compressed, and then encoded by adding some well-chosen redundancy that protects the transmitted bits from the channel's disruptions. In this second volume, we are first located at the modulator's input. The modulator takes the binary data at the output of the channel encoder in order to prepare them for transmission on the channel.

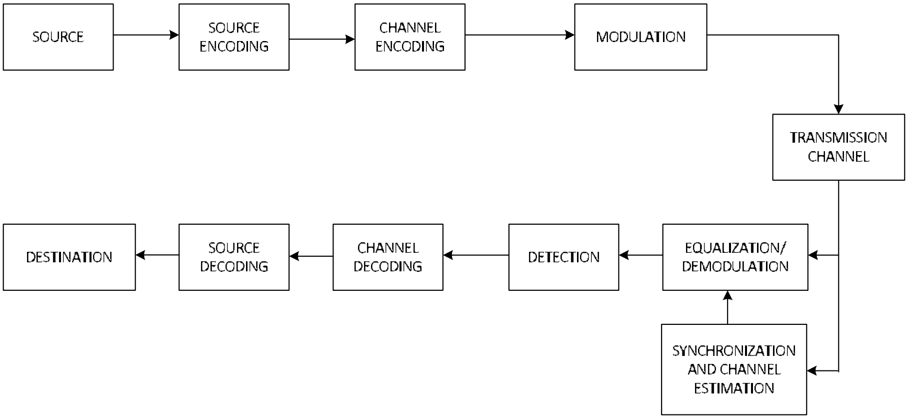

Figure I.1. *Block diagram of the communication chain*

The coded bits are then associated with symbols that will be transmitted during a given symbol period. They are shaped by a transmit filter before being sent on the channel. At the stage, transmission techniques vary, depending on whether baseband transmission (when the signal is carrier by the null frequency) or sine waveform transmission takes place. In the latter case, the frequency carrier is far larger than the bandwidth. In the former case, symbols are carried by the specific value of a line code; whereas in the latter case, they are carried by either the signal's amplitude, phase or carrier frequency value. Baseband transmissions are detailed in Chapter 2. Digital modulations on sine waveforms are studied in Chapter 3. Furthermore, advanced digital modulation techniques, called coded modulations, are presented in Chapter 6. They associate modulations and channel encoding in order to increase the spectral efficiency. Chapter 2 introduces some fundamental notions such as optimum detection on the additive white Gaussian noise channel, and Chapter 3 explains the connection between baseband transmissions and sine waveform transmissions. Modulations' performances are determined in terms of bit error rate depending on the signal-to-noise ratio and on the bandwidth.

The demodulator aims at extracting samples while maximizing a given criterion (signal-to-noise ratio, bit error rate, etc.). The detector's objective is to determine the most likely transmitted symbol. If channel decoding takes hard inputs, the detector directly provides the estimated binary sequence. If channel decoding takes soft inputs, then the detector provides soft information on bits, which generally are logarithms of likelihood ratio.

In order to have an efficient detection, the input samples should be as close as possible to the transmitted ones. Yet, the channel may generate distortions on the transmitted signal, as well as delays which may result in a loss of the symbol period, or frequency shifts. Demodulation and detection must consequently be preceded by a synchronization at the receiver with the transmitted signal in order to correct these shifts. Thus, if the channel has introduced distortions that lead to intersymbol interference, an equalization step must be added in order to estimate the transmitter symbols. These two steps are presented in Chapter 4. We can notice that synchronization may be blind, which means that it is performed without any channel knowledge, in which case its performances will be far lower than if synchronization is based on a training sequence. Similarly, equalization requires channel estimation, which must be implemented prior to these two steps.

In order not to use equalization at the receiver, most recent communication systems split the original wideband channel into several subchannels, where each subchannel only produces a constant signal attenuation but no distortion. This step modifies the modulation block and is thus located before transmission on the channel. Multi-carrier modulations are detailed in Chapter 5.

Finally, we can notice that some useful mathematics and digital signal processing background are provided in Chapter1.

The summary of the contributions of this book shows that we have tried to be as exhaustive as possible, while still studying a large spectrum of the digital communications domain. Some topics have not been considered due to lack of space. For instance, the wireless mobile channel's characteristics or the multiple-access techniques on this channel have not been mentioned, as well as multiple antenna *multiple input, multiple output* (MIMO) techniques. Moreover, we have only focused on the most modern techniques, only considering older techniques with an educational objective, when the latter allow us to better understand more complex modern techniques.

Most of the chapters detail fundamental notions of digital communications and are thus necessary to comprehend the whole communication chain. They provide several degrees of understanding by giving an introduction to these techniques, while still giving some more advanced details. In addition, they are illustrated by examples of implementations in current communications systems. Some exercises are proposed at the end of each chapter, so that the readers may take over the presented notions.

We can nevertheless notice that Chapter 6 proposes an opening to advanced modulation and coding techniques. It may be read as a complement in order to strengthen knowledge.

1

Background

1.1. Introduction

This chapter provides some background necessary for the remainder of this book. The common operations and functions are presented first. The common transforms required for calculations are detailed in section 1.3. Then, some background on discrete and continuous probabilities is provided in section 1.4. Finally, some elements of digital signal processing are recalled in section 1.5.

1.2. Common operations and functions

1.2.1. *Convolution*

The convolution between two signals $s(t)$ and $h(t)$ is defined as:

$$r(t) = (s * h)(t) = \int_{-\infty}^{\infty} s(u)h(t-u)du \qquad [1.1]$$

We will write it as $(s * h)(t) = s(t) * h(t)$ with a notational abuse.

The convolution is linear and invarious to time. Consequently, for any continuous signals $s_1(t)$ and $s_2(t)$, any complex values α_1, α_2 and any time delay t_1, t_2, we can write:

$$\begin{aligned}&(\alpha_1 s_1(t-t_1) + \alpha_2 s_2(t-t_2)) * h(t)\\ &\qquad = \alpha_1 (s_1 * h)(t-t_1) + \alpha_2 (s_2 * h)(t-t_2)\end{aligned} \qquad [1.2]$$

Digital Communications 2: Digital Modulations, First Edition. Mylène Pischella and Didier Le Ruyet.

1.2.2. *Scalar product*

The scalar product between two continuous signals $s(t)$ and $r(t)$ is defined as:

$$\langle s, r \rangle = \int_{-\infty}^{\infty} s(t) r^*(t) dt \qquad [1.3]$$

For any continuous signals $s_1(t)$, $s_2(t)$, $r_1(t)$ and $r_2(t)$ and any complex values α_1, α_2, the following linearity properties hold:

$$\begin{aligned} \langle \alpha_1 s_1 + \alpha_2 s_2, r \rangle &= \alpha_1 \langle s_1, r \rangle + \alpha_2 \langle s_2, r \rangle \\ \langle s, \alpha_1 r_1 + \alpha_2 r_2 \rangle &= \alpha_1^* \langle s, r_1 \rangle + \alpha_2^* \langle s, r_2 \rangle \end{aligned} \qquad [1.4]$$

For vectors of size $m \times 1$, $\mathbf{s} = [s_0, s_1, ..., s_{m-1}]^T$ and $\mathbf{r} = [r_0, r_1, ..., r_{m-1}]^T$, the scalar product is similarly defined:

$$\langle \mathbf{s}, \mathbf{r} \rangle = \sum_{k=0}^{m-1} s_k r_k^* = \mathbf{r}^H \mathbf{s} \qquad [1.5]$$

1.2.3. *Dirac function, Dirac impulse and Kronecker's symbol*

The continuous Dirac function, denoted as $t \mapsto \delta(t)$, is defined in the following way: for any finite-energy signal $s(t)$ and any real value τ,

$$\int_{-\infty}^{\infty} s(t) \delta(t - \tau) dt = s(t - \tau) \qquad [1.6]$$

From [1.6] and [1.1], the convolution of any function $s(t)$ by a Dirac function delayed by τ is equal to the function $s(t)$ delayed by τ:

$$s(t) * \delta(t - \tau) = s(t - \tau) \qquad [1.7]$$

The Dirac impulse is defined for discrete signals as follows:

$$\delta_n = \begin{cases} 1 \text{ if } n = 0 \\ 0 \text{ if } n \neq 0 \end{cases} \qquad [1.8]$$

The Kronecker's symbol is a function of two variables which is equal to 1 if both variables are equal, and to 0 elsewhere.

$$\delta_{n,n_0} = \begin{cases} 1 \text{ if } n = n_0 \\ 0 \text{ if } n \ \neq n_0 \end{cases} \qquad [1.9]$$

As a result, the Dirac impulse is equal to the Kronecker symbol when $n_0 = 0$.

1.2.4. *Step function*

The step function is a continuous function which is equal to 0 when the input value is negative, and equal to 1 when the input value is positive:

$$u(t) = \begin{cases} 0 \text{ if } t \leq 0 \\ 1 \text{ if } t > 0 \end{cases} \qquad [1.10]$$

1.2.5. *Rectangular function*

The rectangular function is a continuous function which is equal to 0 outside of a time interval T that starts at $t = 0$:

$$\Pi_T(t) = \begin{cases} 0 \text{ if } t \notin [0, T[\\ 1 \text{ if } t \in [0, T[\end{cases} \qquad [1.11]$$

1.3. Common transforms

1.3.1. *Fourier transform*

1.3.1.1. *Fourier transform of a continuous signal*

The Fourier transform is a particularly important tool of the field of digital communications. It allows us to study a signal no longer in the time domain, but in the frequency domain. The spectral properties of a signal are more relevant to characterize it than its time properties. For instance, due to its spectral properties, a signal can be determined as baseband or passband, its bandwidth can be characterized and so on.

The Fourier transform of a continuous signal $s(t)$ is:

$$TF[s](f) = \int_{-\infty}^{\infty} s(t)e^{-j2\pi ft}dt \tag{1.12}$$

It is often denoted by $S(f) = TF[s](f)$.

The inverse Fourier transform allows us to recover the time signal $s(t)$ from its Fourier transform $S(f)$:

$$TF^{-1}[S](t) = \int_{-\infty}^{\infty} S(f)e^{j2\pi ft}df \tag{1.13}$$

The Fourier transform is linear. In digital communications, the following property is of particular interest: the Fourier transform of a convolutional product is equal to the product of both Fourier transforms. The opposite property also stands:

$$\begin{aligned} &TF[s * h](f) = TF[s] \times TF[h] \\ &\Leftrightarrow TF[s \times h](f) = TF[s] * TF[h] \end{aligned} \tag{1.14}$$

We list here the Fourier transform of some functions that are useful in digital communications and signal processing:

– the Fourier transform of the Dirac function is a constant: $s(t) = \delta(t) \Leftrightarrow S(f) = 1$;

– the Fourier transform of the rectangular function between $-T/2$ and $T/2$ is a cardinal sine function: $s(t) = \Pi_T(t - T/2) \Leftrightarrow S(f) = T\frac{sin(\pi fT)}{\pi fT}$;

– the Fourier transform of a cosine is a sum of two Dirac functions: $s(t) = \cos(2\pi f_0 t) \Leftrightarrow S(f) = \frac{1}{2}\left(\delta(f - f_0) + \delta(f + f_0)\right)$;

– the Fourier transform of a sine is a difference between two Dirac functions: $s(t) = \sin(2\pi f_0 t) \Leftrightarrow S(f) = \frac{1}{2}\left(\delta(f - f_0) - \delta(f + f_0)\right)$.

1.3.1.2. *Discrete Fourier transform*

Le $\mathbf{x}$ be a discrete signal composed of N samples, $\mathbf{x} = [x_0, x_1, ..., x_{N-1}]$, with sampling frequency F_e and sampling period $T_e = 1/F_e$. $x_n = x(nT_e)$ is the sample corresponding to time nT.

Its Fourier transform is defined as:

$$TF[x](f) = X(f) = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x_n e^{-j2\pi f n T_e} \qquad [1.15]$$

The discrete Fourier transform is generally determined for some discrete frequency values:

$$X\left(\frac{kF_e}{N}\right) = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x_n e^{-j2\pi \frac{nk}{N}} \qquad [1.16]$$

The N samples of the Fourier transform are denoted as $\mathbf{X} = [X_0, X_1, ..., X_{N-1}]$ with $X_k = X\left(\frac{kF_e}{N}\right)$.

The inverse discrete Fourier transform is similarly defined:

$$TF^{-1}[X](t) = x(nT_e) = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} X\left(\frac{kF_e}{N}\right) e^{j2\pi \frac{nk}{N}} \qquad [1.17]$$

Energy is maintained between the time sample's vector $\mathbf{x}$ and the frequency sample vector $\mathbf{X}$. This is proven by Parseval's equality:

$$\begin{aligned}
\frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} |x(nT_e)|^2 &= \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x(nT_e) x^*(nT_e) \\
&= \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x(nT_e) \sum_{k=0}^{N-1} X^*\left(\frac{kF_e}{N}\right) e^{-j2\pi \frac{nk}{N}} \\
&= \sum_{k=0}^{N-1} X^*\left(\frac{kF_e}{N}\right) \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x(nT_e) e^{-j2\pi \frac{nk}{N}} \\
&= \sum_{k=0}^{N-1} X^*\left(\frac{kF_e}{N}\right) X\left(\frac{kF_e}{N}\right) \\
&= \sum_{k=0}^{N-1} \left| X\left(\frac{kF_e}{N}\right) \right|^2 \qquad [1.18]
\end{aligned}$$

The discrete Fourier transform and its inverse can be implemented with low complexity by using the fast Fourier transform (FFT). This recursive algorithm was established by Cooley and Tuckey in 1965. For N samples, the FFT requires $N\log_2(N)$ operations, whereas a direct application of equation [1.17] would require N^2 operations.

1.3.2. *The z transform*

The z transform is used in the fields of signal processing and digital communications to model filtering operations, and especially time delays.

It is applied on discrete sampled signals $\mathbf{x} = [x_0, x_1, ..., x_{N-1}]$, and is defined as follows:

$$TZ[x](z) = X(z) = \sum_{n=0}^{N-1} x_n z^{-n} \qquad [1.19]$$

We can see that the z transform is equal, up to a proportionality factor, to the discrete Fourier transform [1.19] when $z = e^{j2\pi f T_e}$.

1.4. Probability background

Probability theory is a mathematical domain that describes and models random processes. In this section, we present a summary of this theory. We recommend for further reading [PAP 02] and [DUR 10].

Let X be an experiment or an observation that can be repeated under similar circumstances several times. At each repetition, the result of this observation is an event denoted by x , which can take several possible outcomes. The set of these values is denoted by $\mathcal{A}_X$.

The result $X = x$ of this observation is not known before it takes place. X is consequently called a random variable. It is modeled by the frequency of appearance of all its outcomes.

Two classes of random variables can be distinguished:

– discrete random variables, when the set of outcomes is discrete;

– continuous random variables, when their distribution functions are continuous.

1.4.1. *Discrete random variables*

A discrete random variable X takes its values in a discrete set, called its alphabet $\mathcal{A}_X$. This alphabet may be infinite (for instance, if $\mathcal{A}_X = \mathbb{N}$) or finite with a size n if $\mathcal{A}_X = \{x_1, x_2, \ldots, x_n\}$. Each outcome is associated with a probability of occurrence $P_X = \{p_1, p_2, \ldots, p_n\}$:

$$Pr(X = x_i) = p_i \quad p_i \geq 0 \quad \text{and} \quad \sum_{x_i \in \mathcal{A}_X} p_i = 1 \tag{1.20}$$

For discrete random variables, the probability density $f_X(x)$ is defined by:

$$f_X(x) = \sum_{x_i \in \mathcal{A}} \delta(x - x_i) p_i \tag{1.21}$$

where $\delta(u)$ is the Dirac function.

1.4.1.1. *Joint probability*

Let X and Y be two discrete random variables with respective sets of possible outcomes $\mathcal{A}_X = \{x_1, x_2, \ldots, x_n\}$ and $\mathcal{A}_Y = \{y_1, y_2, \ldots, y_m\}$.

$Pr(X = x_i, Y = y_j)$ is called the joint probability of the events $X = x_i$ and $Y = y_j$. Of course, the following property is verified:

$$\sum_{x_i \in \mathcal{A}_x} \sum_{y_j \in \mathcal{A}_y} Pr(X = x_i, Y = y_j) = 1 \tag{1.22}$$

1.4.1.2. *Marginal probability*

The probability $Pr(X = x_i)$ can be computed from the set of joint probabilities $Pr(X = x_i, Y = y_j)$:

$$Pr(X = x_i) = \sum_{y_j \in \mathcal{A}_y} Pr(X = x_i, Y = y_j) \tag{1.23}$$

1.4.1.3. *Conditional probability*

$$Pr(X = x_i|Y = y_j) = \frac{Pr(X = x_i, Y = y_j)}{Pr(Y = y_j)} \qquad [1.24]$$

Similarly, we can write:

$$Pr(Y = y_j|X = x_i) = \frac{Pr(X = x_i, Y = y_j)}{Pr(X = x_i)} \qquad [1.25]$$

As a result, the following relation stands:

$$\begin{aligned} Pr(Y = y_j, X = x_i) &= Pr(X = x_i|Y = y_j)Pr(Y = y_j) \\ &= Pr(Y = y_j|X = x_i)Pr(X = x_i) \end{aligned} \qquad [1.26]$$

which can be further developed to:

$$\begin{aligned} Pr(Y = y_j|X = x_i) &= \frac{Pr(X = x_i|Y = y_j)Pr(Y = y_j)}{Pr(X = x_i)} \\ &= \frac{Pr(X = x_i|Y = y_j)Pr(Y = y_j)}{\sum_{y_k \in \mathcal{A}_y} Pr(X = x_i, Y = y_k)} \\ &= \frac{Pr(X = x_i|Y = y_j)Pr(Y = y_j)}{\sum_{y_k \in \mathcal{A}_y} Pr(X = x_i|Y = y_k)Pr(Y = y_k)} \end{aligned} \qquad [1.27]$$

Equation [1.27] is called the Bayes law. From this equation, we can see that $Pr(X = x_i|Y = y_j)$ is the *a posteriori* probability, whereas $Pr(Y = y_i)$ is the *a priori* probability.

1.4.1.4. *Independence*

If two discrete random variables X and Y are independent, then:

$$Pr(X, Y) = Pr(X)P(Y) \qquad [1.28]$$

and

$$Pr(X|Y) = Pr(X) \qquad [1.29]$$

1.4.2. *Continuous random variables*

The random variable X is continuous if its cumulative distribution function $F_X(x)$ is continuous. $F_X(x)$ is related to the probability density in the following way:

$$f_X(x) = \frac{dF_X(x)}{dx} \quad \Leftrightarrow \quad F_X(x) = \int_{-\infty}^{x} f_X(u)du \qquad [1.30]$$

The random variable mean is defined as:

$$m_X = E\big[X\big] = \int_{-\infty}^{\infty} x f_X(x)dx \qquad [1.31]$$

Its N^{th} moment is equal to:

$$E\big[X^N\big] = \int_{-\infty}^{\infty} x^N f_X(x)dx \qquad [1.32]$$

1.4.3. *Jensen's inequality*

Let us first recall that a function $f(x)$ is convex if, for any x, y and $0 < \lambda < 1$, the following inequality stands:

$$\lambda f(x) + (1-\lambda)f(y) \geq f(\lambda x + (1-\lambda)y) \qquad [1.33]$$

Let f be a convex function, $[x_1, \ldots, x_n]$ a real n-tuple belonging to the definition set of f and $[p_1, \ldots, p_n]$ a real positive n-tuple such that $\sum_{i=1}^{n} p_i = 1$. Thus:

$$f\left(\sum_{i=1}^{n} p_i x_i\right) \leq \sum_{i=1}^{n} p_i f\left(x_i\right) \qquad [1.34]$$

Jensen's inequality is obtained by interpreting the p_i terms as probabilities: if $f(x)$ is convex for any real discrete random variable X, then:

$$f\left(E[X]\right) \leq E\left[f(X)\right] \qquad [1.35]$$

1.4.4. *Random signals*

The signals used in digital communications depend on time t.

Signal $x(t)$ is deterministic if the function $t \mapsto x(t)$ is perfectly known. If, on the contrary, the values taken by $x(t)$ are unknown, the signal follows a random process. At time t, the random variable is denoted by $X(t)$, and an outcome of this random variable is denoted as $x(t)$. The set of all signal values $x(t)$, for any t in the definition domain, is a given outcome of the random process X.

A random process is defined by its probability density and statistical moments. The probability density is equal to:

$$f_X(x,t) = \lim_{\Delta x \to 0} \frac{Pr\left(x \leq X(t) \leq x + \Delta x\right)}{\Delta x} \quad [1.36]$$

The random process is stationary if its probability density is independent of the time: $f_X(x,t) = f_X(x)\ \forall t$. As a result, all of its statistical properties are independent of t. Its probability density can thus be obtained from equation [1.30] in the following way:

$$f_X(x) = \lim_{\Delta X \to 0} \frac{F_{X+\Delta X}(x) - F_X(x)}{\Delta X} \quad [1.37]$$

$m_x(t)$, the mean of the random variable $x(t)$ from the random process X, is defined as:

$$m_x(t) = E\big[x(t)\big] \quad [1.38]$$

The autocorrelation function $R_{xx}(\tau)$ of a random variable is:

$$R_{xx}(t_1,t_2) = E\big[x(t_1)x^*(t_2)\big] \quad [1.39]$$

The random process X is second-order stationary or wide-sense stationary if, for any random signal $x(t)$:

– its mean $m_x(t)$ is independent of t;

– its autocorrelation function verifies $R_{xx}(t_1,t_2) = R_{xx}(t_1+t, t_2+t)\ \forall t$.

Thus, it can simply be denoted as:

$$R_{xx}(\tau) = E\big[x(t)x^*(t-\tau)\big] \qquad [1.40]$$

In this case, the power spectrum density $\gamma_{xx}(f)$ is obtained by applying the Fourier transform on the autocorrelation function:

$$\begin{aligned}\gamma_{xx}(f) &= TF\left[R_{xx}\right](f)\\ &= \int_{-\infty}^{+\infty} R_{xx}(\tau)e^{-j2\pi f\tau}d\tau\end{aligned} \qquad [1.41]$$

Reciprocally, the autocorrelation function $R_{xx}(\tau)$ is determined from the power spectrum density as follows:

$$\begin{aligned}R_{xx}(\tau) &= TF^{-1}\left[\gamma_{xx}\right](\tau)\\ &= \int_{-\infty}^{+\infty} \gamma_{xx}(f)e^{+j2\pi f\tau}df\end{aligned} \qquad [1.42]$$

Generally, the mean and autocorrelation function of a stationary random process are estimated from a set of outcomes of the signal $X(t)$. When the mean over time tends to the random process's mean, the random process is ergodic. Only one time set of outcomes of the random process X is required to evaluate its mean and autocorrelation function. Most random processes that are considered in digital communications are second-order stationary and ergodic.

For discrete signals (for instance, signals that have been sampled from a continuous random signal $x(t)$ at frequency $\frac{1}{T_e}$) $x_n = x(nT_e)$, the autocorrelation function $R_{xx}(\tau)$ is only defined at discrete times $\tau = nT_e$, and the power spectrum density becomes:

$$\begin{aligned}\gamma_{xx}(f) &= TF\left[R_{xx}\right](f)\\ &= \sum_{n=-\infty}^{+\infty} R_{xx}(nT_e)e^{-j2\pi fnT_e}\end{aligned} \qquad [1.43]$$

The power spectrum density can be estimated with the periodogram. When N samples are available, it is equal to:

$$\widetilde{\gamma}_{xx}(f) = \frac{1}{NT_e}\left|\sum_{n=0}^{N-1} x(nT_e)e^{-j2\pi nT_e f}\right|^2 \qquad [1.44]$$

1.4.4.1. *Power*

The power of $x(t)$ is defined as:

$$\begin{aligned} P &= \int_{-\infty}^{+\infty} \gamma_{xx}(f)df \\ &= R_{xx}(0) \\ &= E\left[\, |x(t)|^2 \,\right] \end{aligned} \qquad [1.45]$$

For discrete signals, it is equal to:

$$P = E\left[\, |x_n|^2 \,\right] \qquad [1.46]$$

1.4.4.2. *Energy*

The energy of a random signal $x(t)$ with finite energy is:

$$\mathcal{E} = \int_{-\infty}^{+\infty} |x(t)|^2 \, dt \qquad [1.47]$$

For discrete signals, it is equal to:

$$\mathcal{E} = \sum_{n=-\infty}^{+\infty} |x_n|^2 \qquad [1.48]$$

1.4.4.3. *Characteristic function*

The characteristic function of a random variable is:

$$\Phi_x(u) = E\left[e^{iuX}\right] = \int_{-\infty}^{\infty} e^{jux} f_X(x)dx \qquad [1.49]$$

1.4.4.4. *Cyclostationary signals*

Cyclostationary signals are not stationary. As a result, their autocorrelation function depends on both variables t_1 and t_2. However, the signals are wide-sense cyclostationary with period T_c if:

$$R_{xx}(t_1, t_2) = R_{xx}(t_1 + T_c, t_2 + T_c) \quad \forall(t_1, t_2) \qquad [1.50]$$

1.5. Background on digital signal processing

This section presents some digital signal processing background that is useful for digital communications. For more information on this subject, the readers can refer to [OPP 96].

1.5.1. *Sampling*

1.5.1.1. *Sampling theorem*

An analog signal $x(t)$ is expressed by a set of continuous values of t. Sampling at period T_e consists of expressing the signal x only at time multiples in T_e. The resulting sequence of samples is a digital signal, whose samples are denoted as $x_n = x(nT_e)$, for $n \in \mathbb{Z}$.

Sampling is equivalent to multiplying the analog time-domain signal $x(nT_e) = x(t)w_{T_e}(t)$ by a Dirac comb function, composed of Dirac functions at period nT_e with $n \in \mathbb{Z}$, defined by:

$$w_{T_e}(t) = \sum_{-\infty}^{\infty} \delta(t - nT_e) = \begin{cases} 1 \text{ if } t = nT_e, n \in \mathbb{Z} \\ 0 \text{ elsewhere} \end{cases} \qquad [1.51]$$

The Fourier transform of a Dirac comb with time period T_e is another Dirac comb, with frequency period $F_e = 1/T_e$:

$$TF[w_{T_e}](f) = F_e w_{F_e}(f) \qquad [1.52]$$

Since the Fourier transform of the product $x(t)w_{T_e}(t)$ is the convolution of the Fourier transforms, we get:

$$\begin{aligned} TF[x(t)w_{T_e}(t)](f) &= X(f) * w_{F_e}(f) \\ &= F_e \sum_{-\infty}^{\infty} X(f - nF_e) \end{aligned} \qquad [1.53]$$

where $X(f)$ is the Fourier of $x(t)$.

As a result, the spectrum of the time-domain signal sampled at period T_e is equal to the spectrum of the frequency-domain signal $X(t)$ with replications at every frequency multiple of F_e.

Let us first assume that spectrum $X(f)$ is passband and band-limited, with maximum frequency f_{max}. Equation [1.53] indicates that if $F_e/2 < f_{\text{max}}$, the replications of $X(f)$ will overlap.

If, on the contrary $F_e/2 \geq f_{\text{max}}$, then the pasted versions are separated in the frequency domain. In order to recover the analog signal from the digital signal, it is necessary to apply on the digital signal spectrum [1.53] a lowpass filter with values different from zero in the interval $[-F_e/2, F_e/2]$. This allows us to extract $F_eX(f)$ only, and consequently to recover the spectrum of the original analog signal. $x(t)$ is finally obtained lossily by applying an inverse Fourier transform to $X(f)$.

This result is called Shannon's sampling theorem: a signal that does not possess any spectral component higher than f_{max} can be fully determined by the sequence of its values regularly spaced by $T_e = 1/F_e$, if and only if $F_e/2 \geq f_{\text{max}}$.

1.5.2. *Discrete, linear and time-invariant systems*

When signals are transmitted on a channel, they may be degraded by it. Passing through a channel can mathematically be written as a filtering operation, which may be continuous in the time-domain if the transmitted signal is continuous. In this case, the receiver will digitize the filtered signal. The filtering operation may also be considered in the digital domain. The channel can then be characterized by its impulse response, which corresponds to its response to a Dirac impulse at sampling times nT_e. In this section, we present a summary of the most common digital filters.

A digital filter is a discrete, linear and time-invariant system.

A system is discrete if to any input discrete sequence $\mathbf{x} = [x_n]_{\{-\infty \leq n \leq \infty\}}$ corresponds an output discrete sequence $\mathbf{y}$:

$$S[\mathbf{x}] = \mathbf{y} \quad [1.54]$$

A system is linear if for any input couple of sequences $\mathbf{x}_1$ and $\mathbf{x}_2$ such that $S[\mathbf{x}_1] = \mathbf{y}_1$ and $S[\mathbf{x}_2] = \mathbf{y}_2$, and for any real or complex scalar a:

$$S[\mathbf{x}_1 + a\mathbf{x}_2] = \mathbf{y}_1 + a\mathbf{y}_2 \quad [1.55]$$

A system is time-invariant if to any input sequence shifted by m samples $\tilde{\mathbf{x}}$ such that $\tilde{x}_n = x_{n-m}\ \forall n \in \mathbb{Z}$ corresponds the output sequence shifted by m samples $\tilde{\mathbf{y}}$, such that $\tilde{y}_n = y_{n-m}\ \forall n \in \mathbb{Z}$:

$$S[\tilde{\mathbf{x}}] = \tilde{\mathbf{y}}\ \forall m \in \mathbb{Z} \quad [1.56]$$

A digital filtering operation possesses these three characteristics. It is identified by its impulse response $[h_n]_{n=-\infty}^{\infty}$:

$$\begin{aligned} S[\mathbf{x}]_n = y_n &= \sum_{m=-\infty}^{\infty} x_m h_{n-m}\ \forall n \in \mathbb{Z} \\ &= \sum_{m=-\infty}^{\infty} x_{n-m} h_m\ \forall n \in \mathbb{Z} \end{aligned} \quad [1.57]$$

Equation [1.57] shows that filtering is a convolution in the digital domain:

$$(\mathbf{x} * \mathbf{h})_n = \sum_{m=-\infty}^{\infty} x_{n-m} h_m\ \forall n \in \mathbb{Z} \quad [1.58]$$

The filter coefficients $[h_m]_{-\infty \leq m \infty}$ are generally causal: $h_m = 0\ \forall\ m < 0$. The filter's impulse response is equal to the sequence $[h_m]$. It is obtained by selecting a Dirac function on sample m as the system's input, since in that case:

$$(\delta_m * \mathbf{h})_n = \sum_{m'=-\infty}^{\infty} \delta_{m',m} h_{n-m'} \; \forall n \in \mathbb{Z}$$
$$= \begin{cases} h_m \text{ if } m = n \\ 0 \text{ elsewhere} \end{cases} \qquad [1.59]$$

For any integer m, the system's impulse response coefficient h_m is obtained by providing as input a Dirac δ_m.

A discrete, linear and time-invariant system is stable if to any bounded input sequence corresponds a bounded output sequence. The necessary and sufficient stability condition is:

$$\sum_{n=-\infty}^{\infty} |h_n| < +\infty \qquad [1.60]$$

The frequency response of a filter with impulse response $[h_n]_{n=-\infty}^{\infty}$, denoted as $H(f)$, is the Fourier transform of its impulse response:

$$H(f) = \sum_{n=-\infty}^{\infty} h_n e^{-j2pifnT_e} \qquad [1.61]$$

Its transfer function is defined as the z transform of its impulse response:

$$H(z) = \sum_{n=-\infty}^{\infty} h_n z^{-n} \qquad [1.62]$$

Since filtering is a convolution in time-domain (see [1.58]), by applying the Fourier transform, it becomes a product in frequency domain:

$$y_n = h_n * x_n \Leftrightarrow Y(f) = H(f)X(f) \qquad [1.63]$$

As the z transform is equivalent to the discrete Fourier transform, the filtering operation can also be written as $Y(z) = H(z)X(z)$.

Consequently, the transfer function of the filter is equal to the ratio of the z transform of the output samples and the z transform of the input samples:

$$H(z) = \frac{Y(z)}{X(z)} \tag{1.64}$$

1.5.3. *Finite impulse response filters*

A filter has a finite impulse response (FIR) if its transfer function is a polynomial. It can, therefore, be expressed as:

$$H(z) = \sum_{i=0}^{N} a_i z^{-i} \tag{1.65}$$

The output samples y_n corresponding to the input samples x_n only depend on the previous x samples:

$$y_n = \sum_{i=0}^{N} a_i x_{n-i} \tag{1.66}$$

Finite impulse response filters are always stable.

1.5.4. *Infinite impulse response filters*

A filter has an infinite impulse response (IIR) if its transfer function is an algebraic fraction whose denominator is a polynomial:

$$H(z) = \frac{\sum_{i=0}^{N} a_i z^{-i}}{1 - \sum_{j=1}^{M} b_j z^{-j}} \tag{1.67}$$

The current output sample not only depends on the previous input samples, but also on the previous output samples:

$$y_n = \sum_{i=0}^{N} a_i x_{n-i} + \sum_{j=1}^{M} b_j y_{n-j} \tag{1.68}$$

Any value z_i such that $H(z_i) = 0$ is called a zero of the filter. Any p_j such that $H(p_j) = \infty$ is a pole of the filter. The transfer function can be written as a function of the zeros and poles of H:

$$H(z) = \frac{a_0 \prod_{i=0}^{N}(z - z_i)}{\prod_{j=1}^{M}(z - p_j)} \qquad [1.69]$$

In order to study the stability of IIR filters, the poles must be determined. A filter is stable if and only if all its poles are strictly inside the unit circle: $|P_j| < 1 \, \forall j \in \{1, ..., M\}$.

Moreover, representing the filter by its poles and zeros allows us to determine if the filter is lowpass, passband, etc. by studying the relative locations of the poles and zeros. We can finally notice that a real filter necessarily possesses either real or complex conjugate roots.

2

Baseband Transmissions

2.1. Introduction

Two types of digital communications exist: baseband transmissions and transmissions on a sine waveform, also called transmissions on a carrier frequency. In baseband transmissions, the symbols to be transmitted in the channel do not have their spectrum relocated around a carrier frequency. For transmissions on sine waveforms, the symbols to be transmitted are carried by a frequency $f_c > 0$. In order to separate both transmission types, the baseband modulator is called a line coder. Although both types of communications share their main principles, we will first study baseband transmissions in this chapter, and then study transmissions on a carrier frequency in Chapter 3.

This chapter first presents the most common line codes, explaining for each of them which signals are chosen to represent a bit or a group of bits, and the transition constraints between these signals, if relevant. Section 2.5.2 then details the transmission channel, and especially the additive white Gaussian noise channel, which constitutes the reference case. Section 2.4 deals with the processes at the receiver to obtain an optimum reception on the additive white Gaussian noise channel. Demodulation can be performed with a correlator of the matched filter, which is shown to be equivalent. Then the optimum detector, based on maximum *a priori* likelihood criterion, is detailed. Its performances for a reference line code are evaluated. Finally, the last section concerns limited-bandwidth transmissions. We herein introduce the notion of intersymbol interference; we then determine the Nyquist criterion, which allows us to remove all intersymbol interferences. Finally, the

Digital Communications 2: Digital Modulations, First Edition. Mylène Pischella and Didier Le Ruyet.

adapted filter's optimality for maximizing the signal-to-noise ratio (SNR) is proven, when the Nyquist criterion is verified.

2.2. Line codes

Line codes transmit baseband real signals on a spectral band which may be either limited or infinite. In this section, we detail the reference line codes, which are used in digital systems. Of course, this list is not exhaustive, and mostly aims at introducing the main problems (such as the spectral bandwidth, the possibility to recover the symbol period, etc.) and the solutions proposed to overcome these issues.

A different signal is transmitted during each symbol period T, or equivalently in the time interval $[kT, (k+1)T[$. It represents one bit or, in the last case studied in this section, a group of bits.

In the following, the transmitted bits $d_k \in \{0, 1\}$ have equal probability. They are associated with signals denoted as s_k. The line code is defined as the relation between d_k and s_k. The signals may be jointly dependent in the time domain, which means that s_k may depend on some prior s_{k-i}. In this case, a transition diagram explains the possible transitions between signals.

2.2.1. *Non-return to zero (NRZ) code*

The NRZ line code associates an amplitude $+A$ with each bit $d_k = 1$ and an amplitude $-A$ with each bit $d_k = 0$.

This code can be decomposed into symbols a_k multiplying a transmit filter $g(t)$:

$$x(t) = \sum_{k=-\infty}^{+\infty} a_k g(t - kT) \qquad [2.1]$$

where $g(t) = A\Pi_T(t)$ is a rectangular function with duration T, represented in Figure 2.1. The symbols are linked to the bits in the following way: $a_k = 1$ if $d_k = 1$, and $a_k = -1$ if $d_k = 0$.

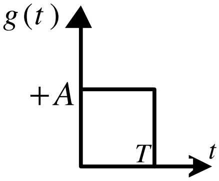

Figure 2.1. *Transmit filter $g(t)$ for NRZ line code*

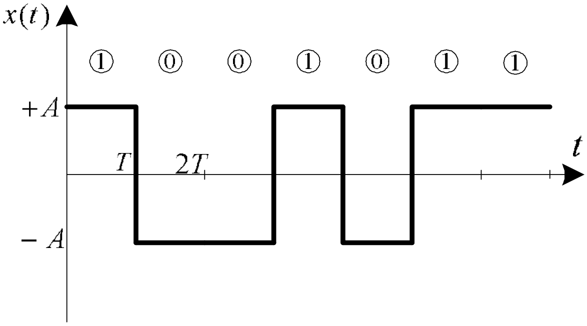

Figure 2.2. *Example of NRZ signal*

The power spectrum density $\gamma_{XX}(f)$ of a baseband signal $x(t)$ when the signal can be written as [2.1] is developed in Appendix A. It is shown that:

$$\gamma_{XX}(f) = \frac{1}{T}|G(f)|^2\gamma_{AA}(f) \qquad [2.2]$$

where $|G(f)|^2$ is the squared modulus of the transmit filter's frequency response and $\gamma_{AA}(f)$ is the Fourier transform of the autocorrelation function $R_{aa}(i) = E[a_{k+i}a_k^*]$:

$$\gamma_{AA}(f) = \sum_{i=-\infty}^{+\infty} R_{aa}(i)e^{-j2\pi fiT} \qquad [2.3]$$

As a result, the power spectrum density of the transmitted signal is determined based on the symbols' autocorrelation and on the frequency response of the waverform $g(t)$. Equation [2.2] is often called Bennett's formula.

Let us apply this formula for the NRZ line code. Since $g(t)$ is a rectangular function of width T and amplitude A, its frequency response $G(f)$ is:

$$G(f) = \int_{-\infty}^{+\infty} e^{-j2\pi ft} g(t)dt = \left[\frac{Ae^{-j2\pi ft}}{-j2\pi ft}\right]_{-T/2}^{+T/2} = \frac{AT \sin \pi fT}{\pi fT} \qquad [2.4]$$

and consequently:

$$|G(f)|^2 = A^2T^2 \left(\frac{\sin(\pi fT)}{\pi fT}\right)^2 \qquad [2.5]$$

In the NRZ case, symbols a_k can be equal to $+1$ and -1. Their mean is equal to zero and their variance σ^2 is equal to 1. From Appendix A, we get $\gamma_{AA}(f) = 1$.

Finally, the power spectrum density is equal to:

$$\gamma_{XX}(f) = A^2T \left(\frac{\sin(\pi fT)}{\pi fT}\right)^2 \qquad [2.6]$$

The power spectrum density of the NRZ line code, as well as those of some codes studied in the next sections, is represented in Figure 2.3.

In theory, the bandwidth of NRZ code is infinite. However, the power spectrum density's main lobe that corresponds to frequencies from 0 to $1/T$ contains 90% of the signal's power. The power spectral density is equal to 0 for all frequencies in multiples of $1/T$.

Since the power spectrum density does not have a peak at frequency $1/T$, the symbol period cannot be directly extracted from it. Thus, it is not possible to perform time-synchronization and correct time delays between the receiver and transmitter, if any. This is due to the fact that the transitions of the transmitted signals are the same as that of the binary sequence. As a result, whenever a long sequence of consecutive 0 or 1 takes place, the symbol clock is lost. In practice, to overcome this problem, a pilot sequence composed of bits that generate lots of transitions is periodically included in the signal.

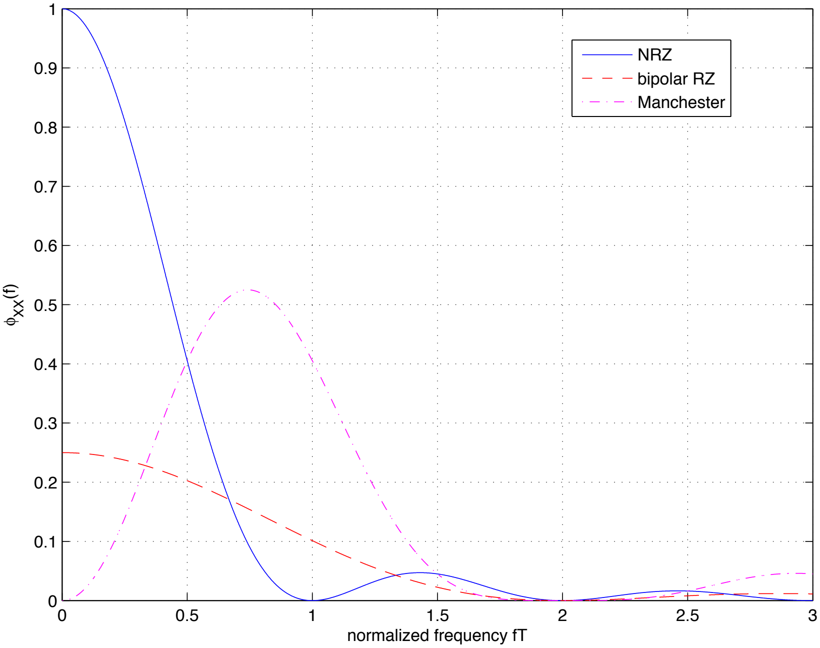

Figure 2.3. *Power spectrum density of NRZ, bipolar RZ and Manchester codes*

2.2.2. *Unipolar return-to-zero (RZ) code*

The binary codes studied in the next three sections (2.2.2, 2.2.3 and 2.2.4) force transitions to appear in the transmission of a signal associated with a bit. Both codes can be written with symbols and a transmit filter, according to equation [2.1].

The unipolar return-to-zero code, also called RZ 1/2 code, associates with each bit $d_k = 1$ an amplitude $+A$ during $T/2$, and then an amplitude 0 during $T/2$. With each bit $d_k = 0$ is associated an amplitude 0. An example of transmission with a unipolar RZ code is given in Figure 2.4.

Its transmit filter $g(t)$ is half a rectangular function:

$$g(t) = \begin{cases} A \text{ if } t \in [0, T/2[\\ 0 \text{ if } t \in [T/2, T[\end{cases}$$

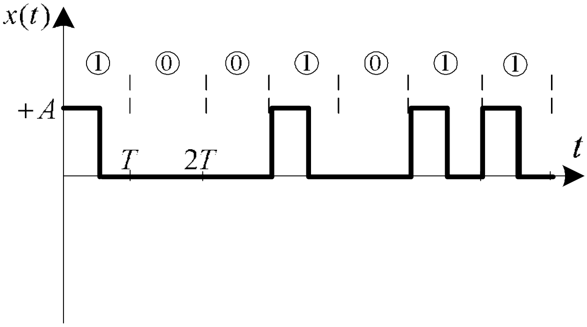

Figure 2.4. *Example of unipolar RZ signal*

And, we simply have $a_k = d_k$.

Since the transmitted symbols 0 and 1 have equal probability, the mean of the sequence μ is equal to 1/2 and its centered variance σ^2 is equal to 1/4. Consequently, the symbols' power spectrum density is:

$$\gamma_{AA}(f) = \frac{1}{4} + \frac{1}{4T} \sum_{i=-\infty}^{+\infty} \delta\Big(f - \frac{i}{T}\Big) \qquad [2.7]$$

and

$$G(f) = \frac{AT}{2} \frac{\sin(\pi f T/2)}{\pi f T/2} \qquad [2.8]$$

The power spectrum density of the unipolar RZ signal is given in special case 2 of Appendix A,

$$\begin{aligned}\gamma_{XX}(f) = {} & \frac{A^2T}{16}\left(\frac{\sin(\pi f T/2)}{\pi f T/2}\right)^2 \\ & + \frac{A^2}{16} \sum_{i=-\infty}^{+\infty} \delta\Big(f - \frac{i}{T}\Big)\left(\frac{\sin(\pi f T/2)}{\pi f T/2}\right)^2 \qquad [2.9]\end{aligned}$$

Since there is a peak at frequency $1/T$, the symbol period can be obtained by filtering.

2.2.3. *Bipolar return-to-zero (RZ) code*

This code associates with the bit $d_k = 1$ an amplitude $+A$ during time $T/2$ and an amplitude 0 during $T/2$. With the bit $d_k = 0$ is associated an amplitude $-A$ during $T/2$ and an amplitude 0 during $T/2$.

Contrary to the unipolar RZ code, transitions occur whether bit 0 or 1 is transmitted.

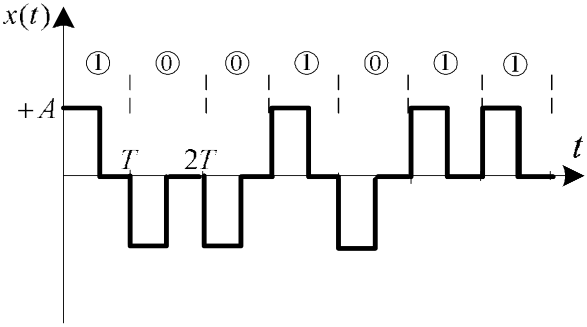

Figure 2.5. *Bipolar RZ signal*

As a result, the signal's transmit filter is half a rectangular function, similar to the unipolar RZ code. Symbols are linked to bits in the following way: $a_k = 1$ if $d_k = 1$ and $a_k = -1$ if $d_k = 0$.

Contrary to the unipolar RZ code, the mean of the sequence is equal to 0. Thus, the power spectrum density does not have any peak. As the variance is $\sigma^2 = 1$, the sequence's power spectrum density is:

$$\gamma_{XX}(f) = \frac{A^2T}{4}\left(\frac{\sin(\pi fT/2)}{\pi fT/2}\right)^2 \qquad [2.10]$$

The power spectrum density of the RZ bipolar code is represented in Figure 2.3.

2.2.4. *Manchester code*

Manchester code has been used on coaxial cable for 10Base2 Ethernet transmissions. It is currently still used for 10BASE T Ethernet, with RJ45 connectors. It associates with each bit $d_k = 1$ an amplitude $+A$ during $T/2$ and then an amplitude $-A$ during $T/2$. With the bit $d_k = 0$ is associated an amplitude $-A$ during $T/2$ and an amplitude $+A$ during $T/2$.

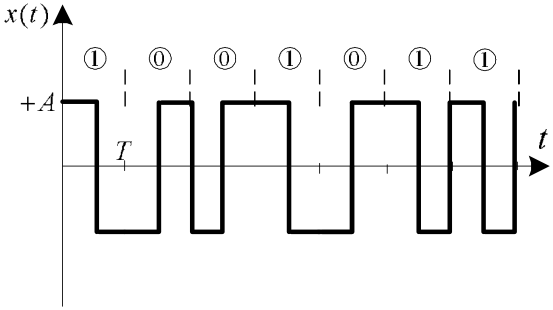

Figure 2.6. *Example of Manchester signal*

In equation [2.1], the transmit filter $g(t)$ is an inverted half rectangular function:

$$g(t) = \begin{cases} A \text{ if } t \in [0, T/2[\\ -A \text{ if } t \in [T/2, T[\end{cases}$$

and symbols are defined as follows: $a_k = 1$ if $d_k = 1$ and $a_k = -1$ if $d_k = 0$.

The power spectrum density of this code, represented in Figure 2.3, has the following expression:

$$\gamma_{XX}(f) = A^2 T \frac{(\sin(\pi f T/2))^4}{(\pi f T/2)^2} \qquad [2.11]$$

Due to the regular transitions, the time symbol can be easily extracted. However, the main lobe's bandwidth is twice as large as that of NRZ code.

2.2.5. *Alternate mark inversion code*

The alternate mark inversion (AMI) code successively associates with $d_k = 1$ an amplitude $+A$ (s_2 signal) and an amplitude $-A$ (s_3 signal). With the bit $d_k = 0$ is associated an amplitude 0 (s_1 signal). A change of sign is imposed between two successive amplitudes representing a bit equal to 1: consequently, if $d_{k-1} = 1$ and $d_k = 1$, then we pass from s_2 signal to s_3 signal between the $(k - 1)^{\text{th}}$ and k^{th} symbol period, or conversely. An example is given in Figure 2.8.

This code may be written with equation [2.1], with $a_k = 0, 1$ or -1, and $g(t) = A\Pi_T(t)$, or directly with the three s_k signals.

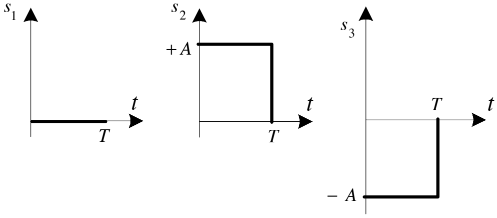

Figure 2.7. *Signals for the alternate mark inversion code*

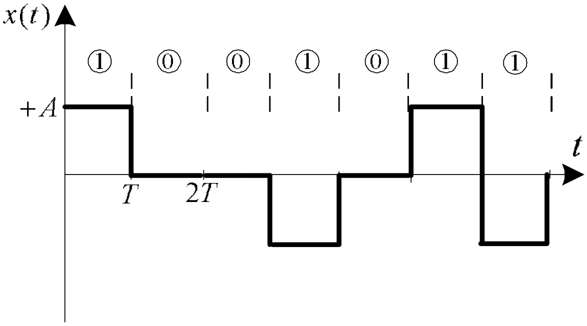

Figure 2.8. *Example of alternate mark inversion signal*

The AMI code can also be represented with a transition diagram that contains two different states, $\mathcal{S}_0$ and $\mathcal{S}_1$, as shown in Figure 2.9. The system is in state $\mathcal{S}_0$ if the last non-zero transmitted signal was s_3, and in state $\mathcal{S}_1$ if the last non-zero transmitted signal was s_2. The transition diagram represents the changes of states between times $(k - 1)T$ and kT. A dotted line arrow indicates that the input bit at time kT, d_k, is equal to 0; whereas a full line indicates that $d_k = 1$. Finally, above each transition, we indicate the value of the bit at time kT, d_k, and the value of the output signal at time kT.

For instance, if the system is in state $\mathcal{S}_1$ and $d_k = 1$, since a change of sign is mandatory and the last non-zero transmitted signal was s_2, that signal at time kT is equal to s_3. If, on the contrary, $d_k = 0$, the transmitted signal is s_1, and the system stays in state $\mathcal{S}_1$ since s_1 is equal to 0.

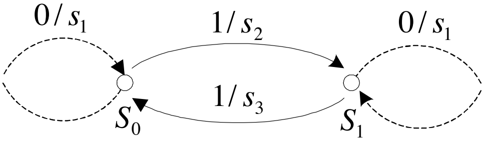

Figure 2.9. *Transition diagram of alternate mark inversion code*

The transition diagram allows us to determine the autocorrelation function of the symbols, $R_{aa}(i)$. Contrary to most codes studied up to now, it is no longer equal to a Dirac function in 0 since symbols a_k are no longer independent. The power spectrum density is obtained from Bennett's formula and $R_{aa}(i)$. It has the following expression, and is represented in Figure 2.10:

$$\gamma_{XX}(f) = A^2 T \sin^2(\pi f T) \left(\frac{\sin(\pi f T)}{\pi f T} \right)^2 \qquad [2.12]$$

The power spectrum density of this code does not contain any low frequencies, but it has a larger bandwidth than that of the NRZ code. It can be used when the transmission channel filters low frequencies. The AMI code was used in the first digital telephony systems on DS1/T1 lines. It has now been replaced by more efficient bipolar codes, such as B3ZS, HDB3, B6ZS and B8ZS.

2.2.6. *Miller code*

Miller code, also called delay modulation code, associates with each bit $d_k = 1$ either an amplitude $+A$ during $T/2$ and then an amplitude $-A$ during $T/2$ (s_2 signal), or an amplitude $-A$ during $T/2$ and then an amplitude $+A$ during $T/2$ (s_3 signal). With each bit $d_k = 0$, it associates an amplitude $-A$ (s_4 signal) or an amplitude $+A$ (s_1 signal) during the whole symbol period. The polarity of the signal associated with a bit $d_k = 1$ is chosen in order to obtain a continuity with the previous impulse. The polarity of a signal associated with a bit $d_k = 0$ is chosen in order to obtain a continuity with the previous impulse if it corresponded to a bit $d_k = 1$. An example of a signal encoded with Miller code is represented in Figure 2.12.

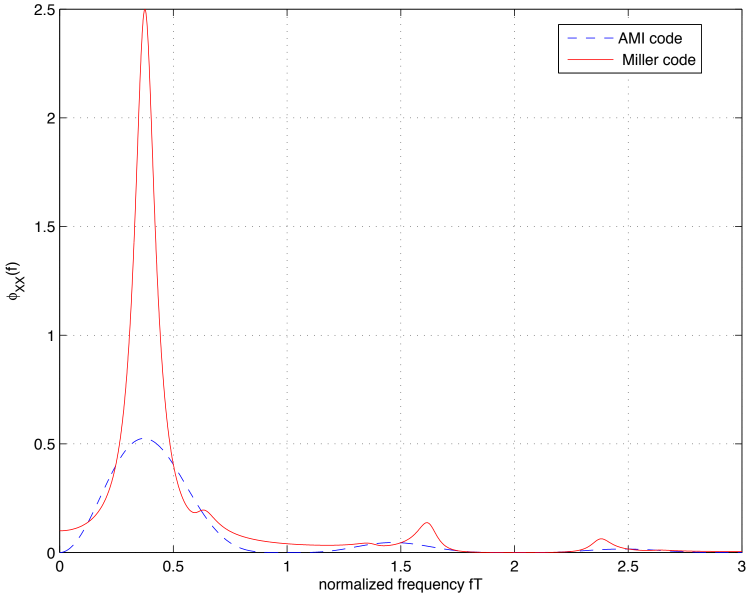

Figure 2.10. *Power spectrum density of AMI and Miller codes*

The transmitted signal can no longer be written with equation [2.1] because two different transmit filters coexist if we consider symbol period T. A more practical way of representing Miller code is to use a transition diagram with 4 different states, as shown in Figure 2.13 [PRO 08]. The states are described as follows:

– the system is in state $\mathcal{S}_1$ if the last signal amplitude is A and the last transmitted bit is $d_{k-1} = 0$;

– it is in state $\mathcal{S}_2$ if the last signal amplitude is $-A$ and the last transmitted bit is $d_{k-1} = 1$;

– it is in state $\mathcal{S}_3$ if the last signal amplitude is A and the last transmitted bit is $d_{k-1} = 1$;

– the system is in state $\mathcal{S}_4$ if the last signal amplitude is $-A$ and the last transmitted bit is $d_{k-1} = 0$.

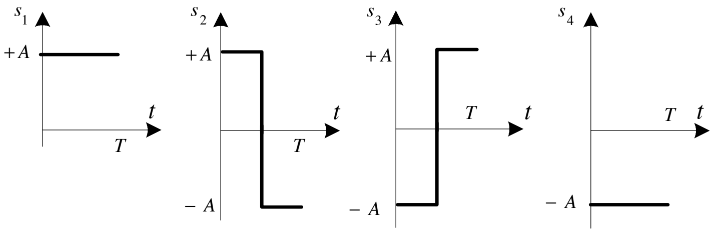

Figure 2.11. *Basic signals of Miller code*

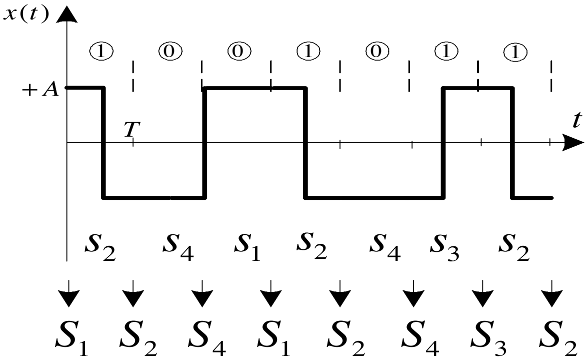

Figure 2.12. *Example of signal encoded with Miller code*

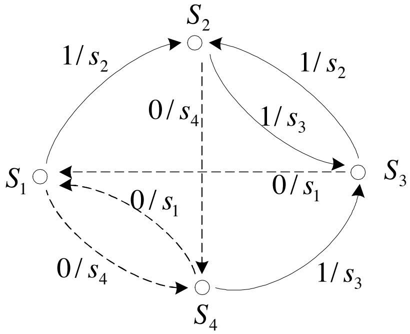

Figure 2.13. *Transition diagram of Miller code*

The power spectrum density of Miller code (see Figure 2.10) cannot be calculated with Bennett's formula. It can be shown that it is equal to [PRO 08]:

$$\begin{aligned}\gamma_{XX}(f) = {} & \frac{1}{2\psi^2(17+8\cos 8\psi)}\\ & (23 - 2\cos\psi - 22\cos 2\psi - 12\cos 3\psi + 5\cos 4\psi\\ & +12\cos 5\psi + 2\cos 6\psi - 8\cos 7\psi + 2\cos 8\psi)\end{aligned}$$

where $\psi = \pi f T$.

The power spectrum density is narrow around frequency $0.375/T$ and does not contain any low frequencies. These two properties are interesting when the transmission channel filters low frequencies. This justifies why Miller code is used for writing on hard drives. Miller code is also used in radio frequency identification (RFID) standard EPC UHF Gen 2 RFID.

2.2.7. *Non-return to zero inverted (NRZI)*

As mentioned in section 2.2.1, the NRZ code sends the same signal during a long time if the data to be transmitted contain long sequences of 0 or 1. To overcome this problem, other codes have been proposed, such as NRZI and MLT-3 codes.

NRZI code presents some similarities with NRZ code since only two amplitudes, A and $-A$, are used to transmit the binary data during symbol period T. However, the value of the transmitted bit is not indicated by the amplitude, but by the transitions: if there is a change of state (A to $-A$ or $-A$ to A), then the transmitted bit is equal to 1; whereas if there is no change of state, the transmitted bit is equal to 0. An example is given in Figure 2.14.

Due to NRZI code, the problem of the lack of transition when a long sequence of 1 is transmitted is avoided. However, this issue is still present if we consider a long sequence of 0.

The NRZI code is used in fiber distributed data interface (FDDI) to connect local networks with optical fiber, with a data rate up to 100 Mbits/s.

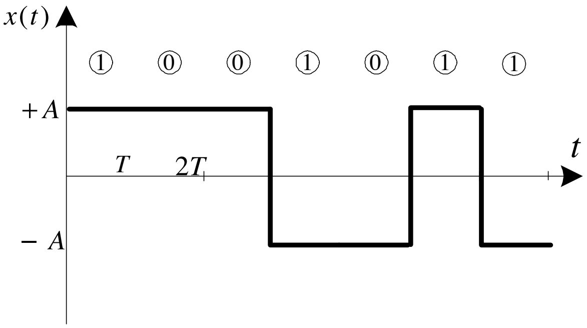

Figure 2.14. *Example of NRZI signal*

2.2.8. *Multi level transmit 3 (MLT-3) code*

This code uses three amplitudes: A, 0 and $-A$. Similarly to NRZI code, transitions between amplitudes indicate the values of the transmitted bits. Consequently, if there is no change of state between symbol periods $(k-1)T$ and kT, the transmitted bit is $d_k = 0$. If, on the contrary, a change of state happens, the transmitted bit is $d_k = 1$. The state's changes are cyclical and ordered as follow: $-A$, 0, A and 0. For instance, if $d_k = 1$ and the last transmitted signal was $-A$, then the amplitude of the signal transmitted in symbol period kT is 0. If the next bit is $d_{k+1} = 1$, the signal transmitted in symbol period $(k+1)T$ will be A, and so on.

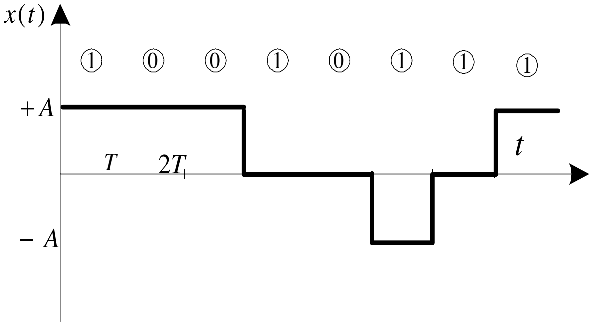

Figure 2.15. *Example of MLT-3 signal*

This cyclic process implies that a minimum of 4 symbol periods are necessary to return to the initial state. As a result, the main lobe's bandwidth of the power spectrum density is equal to $1/4$ of the bandwidth. This line code thus provides a signal which is well localized in the frequency domain.

2.2.9. *RLL(d,k) codes*

Run length limited (RLL) codes are used to avoid long sequences of 1 or 0, with no increase in the bandwidth. The parameters of these codes are denoted as d and k.

Two levels of coding are applied on the binary data: first, each group of input bits is associated with a codeword of bits that are determined according to a given table. The size of the input bits' sequence is not necessarily set, but if variable-length sequences are chosen, then decoding of codewords must be feasible in sequence without any ambiguity. Then, the sequence of codewords is encoded with an NRZI or MLT-3 line code at the symbol period.

The codewords are defined so that each bit 1 is separated by at least d bits 0 and by at most k bits 0 when considering sequences of bits, that is when aggregating successive codewords. The maximum number of successive 0 allows us to avoid long durations with no transition in the transmitted signal, which may lead to a loss of the symbol period T. The minimum number of bits 0 allows us to space out the same signal's transmissions.

We may notice that the previously-described Miller code can be seen as an RLL$(3,1)$ code.

We give an example of RLL(0,2) code called 4B/5B. Its encoding table is provided in 2.1.

The 4B/5B code is used in FDDI local networks over optical fiber with NRZI line code, and on twisted pairs with MLT-3 coding. It is also used on 100BaseTX Fast Ethernet, with data rates up to 100 Mbits/s.

Another example of RLL code with variable-length is the RLL(2,7) code with ratio $1/2$, given in Table 2.2. Both input and output codes are instantaneous.

Input bits	Encoded bits	Input bits	Encoded bits
0000	11110	1000	10010
0001	01001	1001	10011
0010	10100	1010	10110
0011	10101	1100	11010
0100	01010	1100	11010
0101	01011	1101	11011
0110	01110	1110	11100
0111	01111	1111	11101

Table 2.1. *4B/5B code*

Input bits	Encoded bits
10	0100
11	1000
000	000100
010	100100
011	001000
0010	00100100
0011	00001000

Table 2.2. *RLL(2,7) code*

RLL codes are well adapted to hard drive writing and high-data rate transmissions. For instance, the 8B/10B code was invented by IBM in the 1970s. It is now also used for optical fiber transmissions with 1000BASE-SX, 1000BASE-LX and 1000BASE-ZX Gigabit Ethernet, and in the USB 3.0 standard. Similarly, the 10GBASE-SR/LR/ER Ethernet 10 Gigabits standards on optical fiber are based on the 64B/66B code. Audio compact disc (CDs) are coded with the eight-to-fourteen modulation (EFM) code which is an RLL(2,10) code.

We may also note that more sophisticated line codes allowing us to reach higher data rates or lower bit error rates have been invented in recent years. For instance, Blu-ray disks use the 17PP code, which is an RLL $(1, 7)$ code with parity keeping. Likewise, high-definition multimedia interface (HDMI) links are based on a variant of the 8B/10B code.

For more details on RLL codes, the readers may refer to Immink, Siegel and Wolf's paper [IMM 98], and the references therein.

2.2.10. *M-ary NRZ code*

Up to now, we have only studied binary codes. The main applications of binary line codes are baseband transmissions on twisted pairs, coaxial cables and optical fibers. The bandwidth is not a limiting issue for these types of transmissions, which explains why binary codes are chosen. However, the spectral efficiency of line codes can be increased by transmitting several bits per symbol period. The M-ary NRZ code is an example of line code that does not modulate bits, but symbols composed of a several bits.

Similarly to the NRZ code, the transmitted signal can be represented by equation [2.1], with transmit filter $g(t)$ equal to a rectangular function of duration T and amplitude $+A$. However, symbols a_k may take more than two values. For M-ary NRZ codes with M equal to a power of 2, bits are grouped by blocks of $\log_2(M)$ bits. Each block is then encoded on a symbol a_k. The symbols may take M different values in alphabet $\mathcal{A} = \{\pm 1, \pm 3, \ldots, \pm(M-1)\}$. An example with $M = 4$ is given in Figure 2.16.

To transmit one bit, duration T_b is necessary. It can be deduced from the symbol period T by:

$$T = T_b \log_2(M) \qquad [2.13]$$

Since bits are independent and have the same probability, the mean of the sequence of symbols is equal to 0 and its variance $E[(a_k)^2]$ is:

$$\begin{aligned} E[(a_k)^2] &= \frac{1}{M}\sum_{m=1}^{M}(2m-1-M)^2 \\ &= \frac{M^2-1}{3} \end{aligned} \qquad [2.14]$$

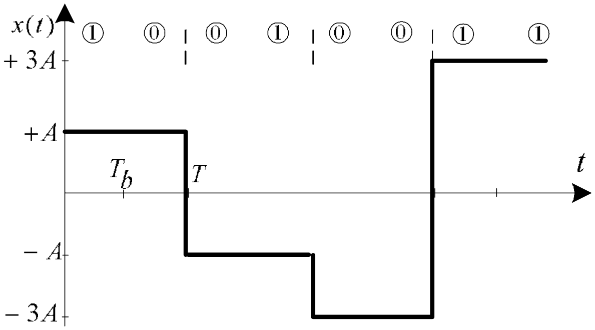

Figure 2.16. *Signal corresponding to an M-ary NRZ code with $M = 4$, $T = 2T_b$*

Finally, the power spectrum density of the M-ary NRZ code is obtained with Bennett's formula:

$$\gamma_{XX}(f) = \frac{A^2T}{3}(M^2-1)\left(\frac{\sin(\pi fT)}{\pi fT}\right)^2 \qquad [2.15]$$

The power spectrum density of the M-ary NRZ code with $M = 4$ is compared to that of NRZ code in Figure 2.17.

2.3. Additive white Gaussian noise channel

The additive white Gaussian noise channel is a transmission channel that only adds a white Gaussian noise to the transmitted signal. The frequency response of the channel, $G_c(f)$, is consequently equal to 1 in the whole bandwidth. Additive white Gaussian noise channel is represented in Figure 2.18. It allows us to model channels where the predominant noise is thermal.

If the receive filter is not taken into account (for instance, if the channel has an infinite bandwidth), the input $x(t)$ and the output $y(t)$ of the additive white Gaussian noise channel are related in the following way:

$$y(t) = x(t) + b(t) \qquad [2.16]$$

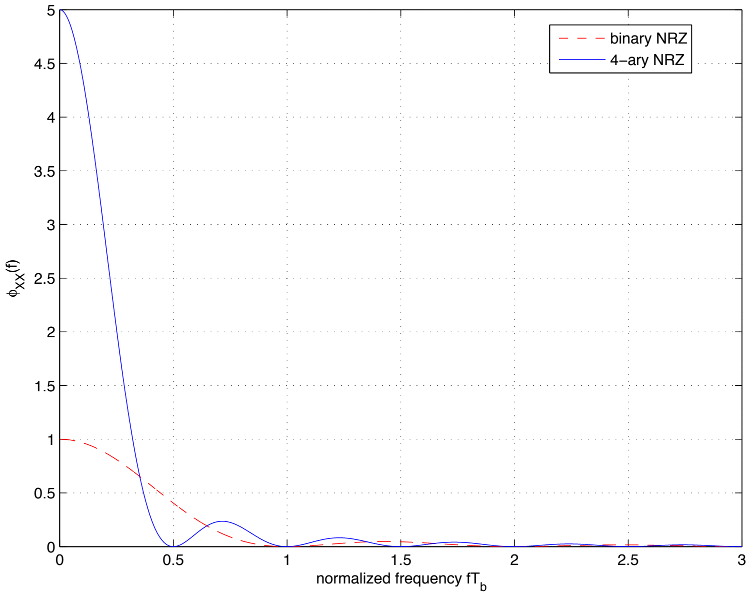

Figure 2.17. *Power spectrum density of binary NRZ and 4-ary NRZ codes*

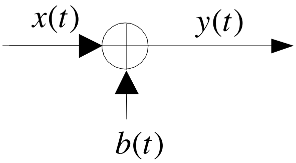

Figure 2.18. *Transmit and receive signals with the additive white Gaussian noise channel*

The white noise $b(t)$ is a wide-sense stationary random process whose power spectrum density is independent of the frequency and equal to $N_0/2$:

$$\gamma_{bb}(f) = \frac{N_0}{2} \text{ W/Hz} \qquad [2.17]$$

It has the following properties:

– it is centered: $m_b = E\left[b(t)\right] = 0$;

– its samples are independent: as a result, its autocorrelation function is equal to 0 at all points, except at 0;

– its probability density is a Gaussian.

The white noise power, $R_{bb}(0)$, is equal to the inverse Fourier transform of $\gamma_{bb}(f)$ at 0. As its power spectrum density is constant whatever the frequency, $R_{bb}(0)$ is infinite. Consequently, the white noise is not physically feasible with infinite bandwidth.

An approximate white noise is thus studied in a finite bandwidth B. It has the following characteristics: it is a random process with the same properties as that of the white noise, but its power spectrum density is equal to $N_0/2$ in bandwidth $[-B, B]$, and to 0 elsewhere. Its power is equal to BN_0. Its characteristics are studied in detail in section 3.2.3.5. N_0 is in general called the noise's unilateral power spectrum density. N_0 is equal to kT, where $k = 1.38 \times 10^{-23}$ Joules/Kelvin is Boltzmann's constant and T is the ambient temperature, expressed in Kelvin. Its value is -174 dBmW/Hz when the temperature is 300 $K = 27\ C$.

2.4. Optimum reception on the additive white Gaussian noise channel

2.4.1. *Introduction*

This section studies the block diagram of the optimum receiver. The receiver identifies the most likely transmitted symbols after they have passed through the channels. It is composed of two steps: demodulation and then detection. In this chapter, we restrict ourselves to detections that generate a symbol-by-symbol estimation. The estimation of a sequence of symbols will be studied in section 4.3.5.

We assume that the receiver is coherent: it has acquired the frequency, phase and symbol period of the transmitter, and is perfectly synchronized with it. The synchronization methods are presented in Chapter 4.

Before considering the receiver, we first detail the modulator's block diagram, since the demodulator applies its inverse steps.

2.4.2. *Modulator's block diagram*

The channel encoder provides groups of n bits. Thus, there are $M = 2^n$ different possible messages $c \in \{c_1, c_2, \ldots, c_M\}$.

The modulator aims at associating with each message $c = c_i$ a signal $x_i(t)$, defined on the finite interval $0 \leq t \leq T$ and chosen among a set of $M = 2^n$ signals. The energy of each real signal $x_i(t)$, between 0 and T, is denoted as E_{si}. It is equal to:

$$E_{si} = \int_0^T x_i(t)^2 dt \qquad [2.18]$$

The detailed block diagram of the modulator is represented in Figure 2.19. The amplitudes of signals $x_1(t)$ and $x_2(t)$ have been chosen so that the energy per symbol is equal to E_s, independently of the transmitted signal.

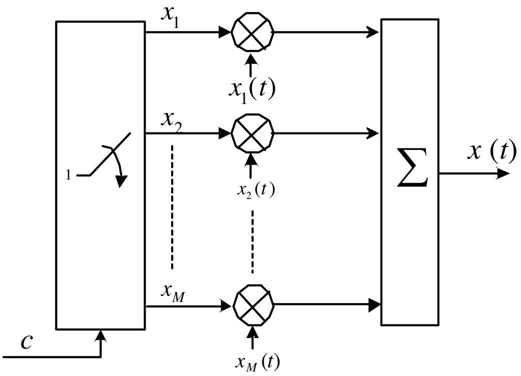

Figure 2.19. *Detailed diagram of the modulator*

In order to transmit the signal $x_i(t)$ associated with message $c = c_i$, we need $x_i = 1$ and $x_j = 0$ for all $j \neq i$.

As an example, for the binary NRZ line code composed of $M = 2$ elementary messages, the signals are:

$$x(t) = \begin{cases} x_1(t) & \text{if} \quad c = 0 \quad (x_1 = 1, x_2 = 0) \\ x_2(t) & \text{if} \quad c = 1 \quad (x_1 = 0, x_2 = 1) \end{cases}$$

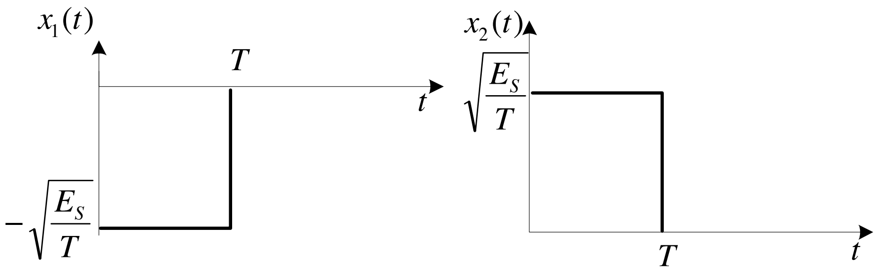

Figure 2.20. *Set of signals $x_1(t)$ and $x_2(t)$ for the binary NRZ line code*

They are represented in Figure 2.20.

In another example, we consider $M = 4$ messages modulated with the following four biorthogonal signals (see Figure 2.21):

$$x(t) = \begin{cases} x_1(t) & \text{if} \quad c = 00 \quad (x_1 = 1, x_2 = 0, x_3 = 0, x_4 = 0) \\ x_2(t) & \text{if} \quad c = 01 \quad (x_1 = 0, x_2 = 1, x_3 = 0, x_4 = 0) \\ x_3(t) & \text{if} \quad c = 10 \quad (x_1 = 0, x_2 = 0, x_3 = 1, x_4 = 0) \\ x_4(t) & \text{if} \quad c = 11 \quad (x_1 = 0, x_2 = 0, x_3 = 0, x_4 = 1) \end{cases}$$

The modulator's block diagram can be simplified by representing the M possible signals $x_i(t)$ by linear combinations of N functions from an orthonormal basis $f_j(t)$ with unitary energy and $N \leq M$. Generally, the $f_j(t)$ functions are obtained by applying the Gram–Schmidt orthogonalization technique.

The $x_i(t)$ signals are then expressed as follows:

$$x_i(t) = \sum_{j=1}^{N} x_{ij} f_j(t) \qquad [2.19]$$

and inversely,

$$x_{ij} = \int_0^T x_i(t) f_j(t) dt \qquad [2.20]$$

where we assumed that the baseline functions $f_1(t), f_2(t), \ldots, f_N(t)$ are orthonormal, which means that they verify constraint [2.21]:

$$\int_0^T f_j(t) f_{j'}(t) dt = \delta_{j'j} = \begin{cases} 1 & \text{if} \quad j = j' \\ 0 & \text{if} \quad j \neq j' \end{cases} \qquad [2.21]$$

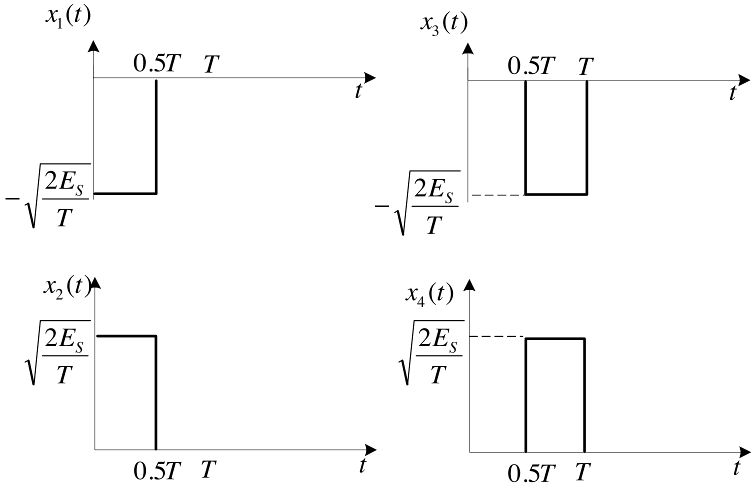

Figure 2.21. *Set of signals $x_i(t)$ $i = 1, 2, 3$ and 4 for $M = 4$ biorthogonal signals*

Since the baseline functions have a unitary energy, the energy of each signal $x_i(t)$ between 0 and T is equal to E_{si}:

$$E_{si} = \int_0^T x_i(t)^2 dt = \sum_{j=1}^{N} {x_{ij}}^2 \qquad [2.22]$$

If all symbols have equal probability, the average energy E_s is equal to:

$$E_s = \frac{1}{M} E_{si} \qquad [2.23]$$

Using these notations, we can obtain Figure 2.22.

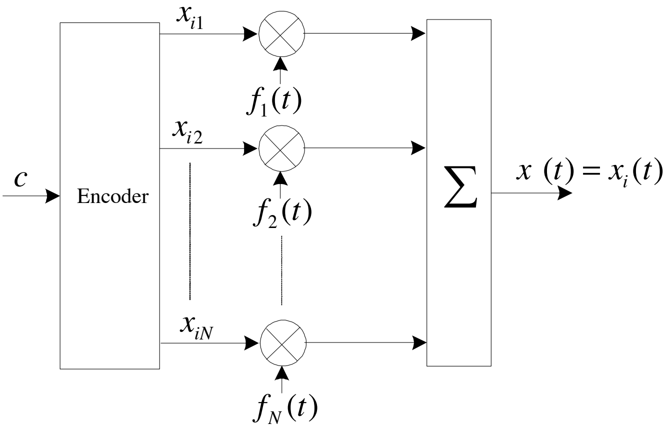

Figure 2.22. *Detailed modulator's diagram*

For NRZ line codes with $M = 2$ signals, $N = 1$ since only one function $f_1(t)$, represented in Figure 2.23, is required to generate both signals $x_1(t)$ and $x_2(t)$.

$$x(t) = \begin{cases} x_1(t) = -\sqrt{E_s} f_1(t) & \text{if} \quad c = 0 \quad (x_{11} = -\sqrt{E_s}) \\ x_2(t) = +\sqrt{E_s} f_1(t) & \text{if} \quad c = 1 \quad (x_{21} = +\sqrt{E_s}) \end{cases}$$

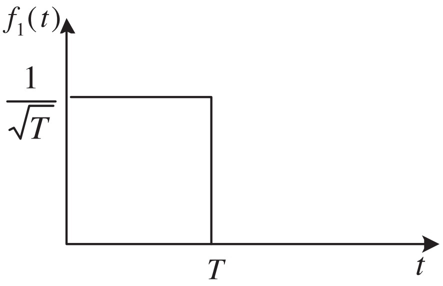

Figure 2.23. $f_1(t)$ *signal for binary NRZ line code*

The modulator's block diagram is given in Figure 2.24.

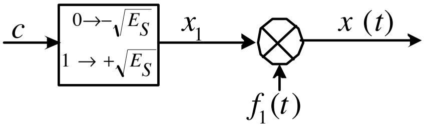

Figure 2.24. *Block diagram of the binary NRZ modulator*

In this example, the baseline function $f_1(t)$ corresponds to the impulse response $g(t)$ of the transmit filter. The diagram of the corresponding modulator is represented in Figure 2.25.

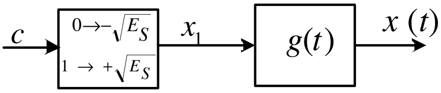

Figure 2.25. *Diagram of the NRZ modulator*

On the contrary, for biorthogonal signals with $M = 4$, two functions $f_1(t)$ and $f_2(t)$ (represented in Figure 2.26) are necessary to generate the 4 signals in the following way:

$$x(t) = \begin{cases} x_1(t) = -\sqrt{E_s} f_1(t) & \text{if} \quad c = 00 \quad (x_{11} = -\sqrt{E_s}, x_{12} = 0) \\ x_2(t) = +\sqrt{E_s} f_1(t) & \text{if} \quad c = 01 \quad (x_{21} = +\sqrt{E_s}, x_{22} = 0) \\ x_3(t) = -\sqrt{E_s} f_2(t) & \text{if} \quad c = 10 \quad (x_{31} = 0, x_{32} = -\sqrt{E_s}) \\ x_4(t) = +\sqrt{E_s} f_2(t) & \text{if} \quad c = 11 \quad (x_{41} = 0, x_{42} = +\sqrt{E_s}) \end{cases}$$

In that case, $N = 2$.

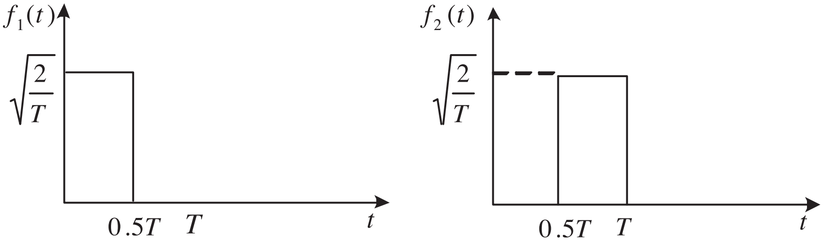

Figure 2.26. *Set of signals $f_1(t)$ and $f_2(t)$ for biorthogonal signals*

2.4.3. *Optimum receiver for the additive white Gaussian noise channel*

We assume that one signal $x_i(t)$ among the M possible signals has been transmitted during time T on a white additive Gaussian noise channel.

At the receiver input, the signal is:

$$q(t) = x_i(t) + b(t) \qquad 0 \leq t \leq T \tag{2.24}$$

where $b(t)$ is a white Gaussian noise of power spectrum density $N_0/2$.

In this chapter, we suppose that the receiver determines each symbol independently of the others, at each time clock T. The optimum reception consists of two phases: first, optimum demodulation, which processes the received signal so as to maximize the SNR, and then, detection, which estimates the most likely transmitted symbols based on this postprocessed signal.

The optimum demodulator can be achieved with two different block diagrams: the correlator and the matched filter. We will demonstrate that both block diagrams are equivalent.

2.4.3.1. *Correlator*

The correlator projects the received signal on the N baseline functions $f_j(t)$. Its block diagram is represented in Figure 2.27. On each of the N branches, three steps are applied in parallel: first, the received signal is multiplied by the baseline function $f_j(t),\ j \in \{1, ..., N\}$. Then, the resulting signal is integrated over the symbol time T, and finally, it is sampled at symbol period T.

For $j \in \{1, ..., N\}$, y_j is defined as the output of the j^{th} sampler at time T. Since the baseline functions $f_j(t)$ have a unitary energy between 0 and T, y_j can be expressed as follows:

$$\begin{aligned} y_j &= y_j(T) \\ &= \int_0^T q(t) f_j(t) dt \end{aligned}$$

$$= \int_0^T x_i(t) f_j(t) dt + \int_0^T b(t) f_j(t) dt$$

$$= \int_0^T \sum_{j'=1}^{N} x_{ij'} f_{j'}(t) f_j(t) dt + \int_0^T b(t) f_j(t) dt$$

$$= \sum_{j'=1}^{N} x_{ij'} \int_0^T f_{j'}(t) f_j(t) dt + \int_0^T b(t) f_j(t) dt$$

$$= x_{ij} + n_j \qquad [2.25]$$

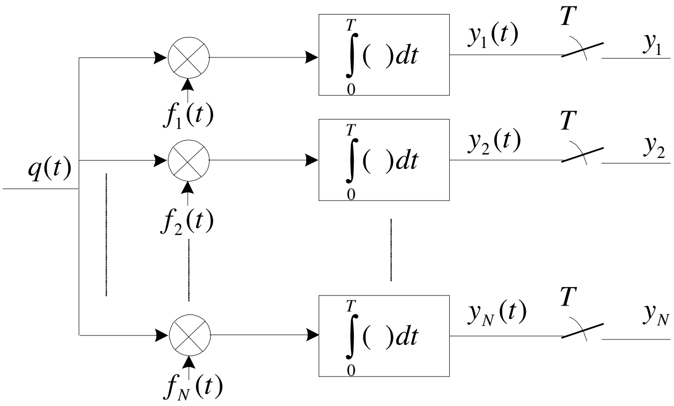

Figure 2.27. *Block diagram of the correlator*

The received signal $q(t)$ is now represented by a vector with N components y_j. n_j terms are random variables relative to the additive noise of the transmit channel. These random variables follow a Gaussian distribution, since the linear filtering of a Gaussian noise does not modify its Gaussian nature. Their mean is equal to:

$$E\left[n_j\right] = \int_0^T b(t) f_j(t) dt = 0 \qquad [2.26]$$

and their variance is:

$$\begin{aligned}\sigma^2 &= E\left[n_j^2\right] \\ &= E\left[\int_0^T \int_0^T b(t)b(u)f(t)f(u)dtdu\right] \\ &= \frac{N_0}{2}\int_0^T \int_0^T \delta(t-u)f(t)f(u)dtdu \\ &= \frac{N_0}{2}\int_0^T f(t)^2 dt \\ &= \frac{N_0}{2}\end{aligned} \qquad [2.27]$$

Consequently, the noise samples are centered with variance $\sigma^2 = \frac{N_0}{2}$. It can also be shown that these samples are uncorrelated and independent. Thus, filtering and sampling have not modified the fundamental characteristics of the noise.

The samples $y_1, \ldots, y_N$ are an exhaustive summary of signal $q(t)$. The complete transmission chain is summarized by N equations [2.25].

2.4.3.2. *Matched filter*

A set of filters can be used at the receiver instead of correlators to obtain the y_j samples. The block diagram of the demodulator is then given in Figure 2.28.

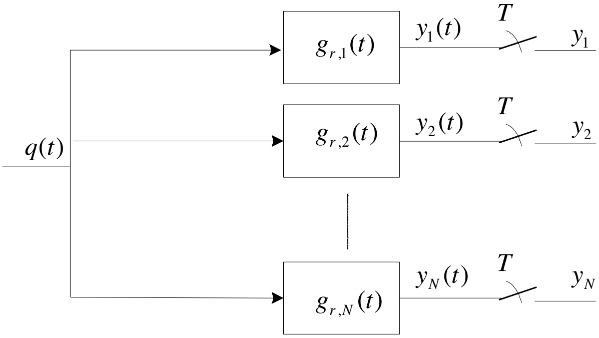

Figure 2.28. *Block diagram of the demodulator with matched filters*

By definition, the receive filter matched to the j^{th} baseline function has the following impulse response:

$$g_{r,j}(t) = f_j(T-t) \qquad \text{with} \quad 0 \leq t \leq T \tag{2.28}$$

Its frequency response is consequently equal to:

$$G_{r,j}(f) = F_j^*(f)e^{-j2\pi fT} \tag{2.29}$$

The output signal after the matched filter is the convolution of $q(t)$ and $g_{r,j}(t)$:

$$\begin{aligned} y_j(t) &= q(t) * g_{r,j}(t) \\ &= \int_{-\infty}^{+\infty} q(\tau)g_{r,j}(t-\tau)d\tau \\ &= \int_0^t q(\tau)g_{r,j}(t-\tau)d\tau \\ &= \int_0^t q(\tau)f_j(T-t+\tau)d\tau \end{aligned}$$

At time T, signal $y_j(T) = y_j$ is equal to:

$$y_j(T) = \int_0^T q(\tau)f_j(\tau)d\tau \tag{2.30}$$

Thus, equations [2.25] and [2.30] show that the output of filters $g_{r,j}(t)$ at time $t = T$ is similar to that of the block diagram with the correlator.

It is important to note that the SNR at the output of the matched filter does not depend on the signal's shape, but on its energy.

2.4.3.3. *Optimality of the matched filter for the additive white Gaussian noise channel*

In this section, we will demonstrate that the matched filter [2.28] leads to a maximization of the SNR at the receiver. Let $h(t)$ be any receive filter applied on branch j. The received signal $y_j(T)$ (equation [2.30]) contains a term of useful signal, denoted as $y_{j,u}(T)$, and a term of filtered noise, $n(T)$. Since the transmitted signal is $x_{ij}f_j(t)$, the received signal at any time t is equal to:

$$y_j(t) = x_{ij}f_j(t) * h(t) + b(t) * h(t) \qquad [2.31]$$

In the frequency domain, the useful signal can be written as $Y_{j,u}(f) = x_{ij}F_j(f)H(f)$. After applying the inverse Fourier transform, it becomes:

$$y_{j,u}(t) = x_{ij}\int_{-\infty}^{+\infty} F_j(f)H(f)e^{j2\pi ft}df \qquad [2.32]$$

When $t = T$, the squared modulus of the previous equation is:

$$\begin{aligned}|y_{j,u}(T)|^2 &= x_{ij}\left|\int_{-\infty}^{+\infty} F_j(f)H(f)e^{j2\pi fT}df\right|^2 \\ &\leqslant x_{ij}\int_{-\infty}^{+\infty} |F_j(f)e^{j2\pi fT}|^2 df \times \int_{-\infty}^{+\infty} |H(f)|^2 df \qquad [2.33]\end{aligned}$$

where [2.33] is obtained by using Cauchy–Schwarz inequality.

The variance of the noise at the receiver output is:

$$\sigma^2 = \frac{N_0}{2}\int_{-\infty}^{+\infty} |H(f)|^2\, df \qquad [2.34]$$

The SNR is defined as the ratio between the signal's energy and the noise variance. Assuming that x_{ij} is equal to 1 or -1, the SNR is $\frac{|y_{j,u}(T)|^2}{\sigma^2}$. It is expressed by using equations [2.33] and [2.34]:

$$\begin{aligned}\frac{|y_{j,u}(T)|^2}{\sigma^2} &= \frac{2}{N_0}\frac{\left|\int_{-\infty}^{+\infty} F_j(f)H(f)e^{j2\pi fT}df\right|^2}{\int_{-\infty}^{+\infty} |H(f)|^2 df} \\ &\leqslant \frac{2}{N_0}\int_{-\infty}^{+\infty} \left|F_j(f)e^{j2\pi fT}\right|^2 df \qquad [2.35]\end{aligned}$$

The SNR is maximized when Cauchy–Schwarz equality is verified:

$$\left|\int_{-\infty}^{+\infty} F_j(f)H(f)e^{j2\pi fT}df\right|^2 = \int_{-\infty}^{+\infty} |F_j(f)e^{j2\pi fT}|^2 df \times \int_{-\infty}^{+\infty} |H(f)|^2 df \qquad [2.36]$$

This constraint is equivalent to the following one:

$$H(f) = \beta.F_j^*(f)e^{-j2\pi fT} \qquad [2.37]$$

where β is any real value. In the time domain, the receive filter is consequently obtained from the transmit filter as follows:

$$h(t) = \beta f_j^*(T-t) \qquad [2.38]$$

If we choose $\beta = 1$, since the transmit filter $f_j(t)$ is real, we finally get:

$$h(t) = f_j(T-t) = g_{r,j}(t) \qquad [2.39]$$

where $g_{r,j}(t)$ is defined in equation [2.28].

Consequently, $g_{r,j}(t)$ is the receive filter that maximizes the SNR, assuming that the transmit filter is known.

2.4.3.4. *Optimum detector*

The optimum detector aims at determining the symbol that is the most likely to have been transmitted.

$y_j = x_{ij} + n_j$ is the output of the j^{th} corrrelator or matched filter. The detector wants to determine, on each element j of the orthonormal base, the value of x_{ij}. The receiver can then reconstruct the transmitted signal, $x_i(t) = \sum_{j=1}^{N} x_{ij}f_j(t)$.

We focus on the j^{th} branch. The receiver knows that the transmitted signal belongs to a set of M real signals, denoted as $\mathcal{S} = \{x_{1j}, x_{2j}, ..., x_{Mj}\}$.

The objective of the detection is to determine the most likely x_{ij}, knowing that y_j has been received. This rule is called maximum *a posteriori* (MAP).

$$x_{i^*j} = \text{arg} \max_{i \in \{1,...M\}} Pr(x_{ij}|y_j) \qquad [2.40]$$

Another statistical estimator is called maximum likelihood (ML). It is defined as follows:

$$x_{i^*j} = \text{arg} \max_{i \in \{1,...M\}} p(y_j|x_{ij}) \qquad [2.41]$$

Bayes' formula implies that:

$$Pr(x_{ij}|y_j) = \frac{p(y_j|x_{ij})Pr(x_{ij})}{p(y_j)} \qquad [2.42]$$

The probability of y_j is expressed as follows:

$$p(y_j) = \sum_{i=1}^{M} p(y_j|x_{ij})Pr(x_{ij}) \qquad [2.43]$$

If all M signals have equal probability, $Pr(x_{ij}) = 1/M$, $\forall i \in \{1, ...M\}$ and equation [2.43] becomes:

$$p(y_j) = \frac{1}{M} \sum_{i=1}^{M} p(y_j|x_{ij}) \qquad [2.44]$$

By replacing $p(y_j)$ in equation [2.42], we get the expression of the *a posteriori* probability:

$$Pr(x_{ij}|y_j) = \frac{p(y_j|x_{ij})}{\sum_{i=1}^{M} p(y_j|x_{ij})} \qquad [2.45]$$

Since $y_j - x_{ij} = n_j$ only depends on the noise and does not depend on the transmitted signal x_{ij}, $\sum_{i=1}^{M} p(y_j|x_{ij})$ in equation [2.45] is independent of x_{ij}. Thus, maximizing the *a posteriori* probability is equivalent to maximizing the likelihood:

$$x_{i^*j} = \arg \max_{i\in\{1,\ldots M\}} Pr(x_{ij}|y_j)$$
$$\Leftrightarrow x_{i^*j} = \arg \max_{i\in\{1,\ldots M\}} p(y_j|x_{ij}) \qquad [2.46]$$

Detecting symbols with equal probability is similar to applying the ML criterion.

On the additive white Gaussian noise channel, the conditional probability $p(y_j|x_{ij})$ only depends on the distribution of noise n, whose probability density is:

$$p(n_j) = \frac{1}{\sqrt{2\pi\sigma_n^2}} \exp\left(-\frac{n_j^2}{2\sigma^2}\right) \qquad [2.47]$$

Thus:

$$p(y_j|x_{ij}) = p(n_j) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{\left(y_j - x_{ij}\right)^2}{2\sigma^2}} \qquad [2.48]$$

By taking the logarithm of equation [2.49], we obtain the expression of the log-likelihood:

$$\ln\left(p(y_j|x_{ij})\right) = -\frac{1}{2}\ln(\pi\sigma^2) - \frac{1}{2\sigma^2}\left(y_j - x_{ij}\right)^2 \qquad [2.49]$$

The signal that maximizes [2.49] on all signals x_{ij} is equal to the signal that minimizes the Euclidean distance between y_j and x_{ij}:

$$x_{i^*j} = \arg \max_{i\in\{1,\ldots M\}} \ln\left(p(y_j|x_{ij})\right)$$
$$\Leftrightarrow x_{i^*j} = \arg \min_{i\in\{1,\ldots M\}} \left(y_j - x_{ij}\right)^2 \qquad [2.50]$$

Moreover, since $\left(y_j - x_{ij}\right)^2 = y_j^2 + x_{ij}^2 - 2y_jx_{ij}$, if all signals x_{ij} have the same energy, then the ML criterion is equivalent to maximizing y_jx_{ij}.

In this section, we have only considered the real case. If the transmitted signals are complex, as for transmissions on a sine waveform that are pulled back over baseband (see Chapter 3), then the ML criterion is applied both on the real and imaginary parts of the signal, so as to reconstruct it.

2.4.4. *Evaluation of the bit error rate for the binary NRZ signal on the additive white Gaussian noise channel*

2.4.4.1. *Assumptions*

In order to evaluate the performances of digital transmission chains, it is important to determine the symbol error probability as a function of the SNR. In this section, we consider the binary NRZ line code.

The NRZ demodulator by correlator is reduced to one branch, and represented in Figure 2.29.

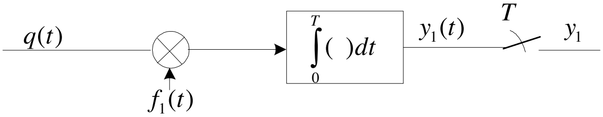

Figure 2.29. *Block diagram of the NRZ demodulator*

The output of the sampler at time T, y_1, can be written as y and becomes:

$$y = x_i + n$$

and the output noise n is Gaussian, centered with variance $\sigma^2 = \frac{N_0}{2}$.

Figure 2.30 provides an example of the signals at the outputs of the modulator, channel and correlator, respectively.

The SNR after sampling is equal to:

$$\text{SNR} = \frac{E[x_i^2]}{\sigma^2} = \frac{\sum_i Pr(X = x_i)x_i^2}{\sigma^2} = \frac{x_i^2}{\sigma^2} \qquad [2.51]$$

Let E_b be the average energy per bit. Since a symbol corresponds to one bit, $E_b = E_s = x_i^2$. Moreover, as $\sigma^2 = \frac{N_0}{2}$, the SNR can be expressed as a function of the signal's binary energy and the power spectrum density of the noise:

$$\mathrm{SNR} = \frac{2E_b}{N_0} \qquad [2.52]$$

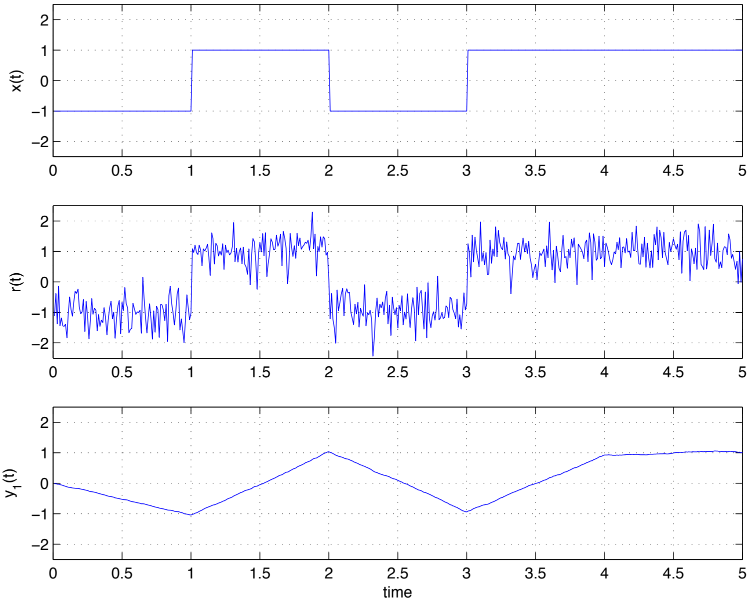

Figure 2.30. *Signals at the outputs of the modulator, channel and correlator*

We now focus on the performances of the NRZ line code on the additive white Gaussian noise channel. The transmitted symbol, denoted as x_i, is equal to $x_1 = -\sqrt{E_b}$ or $x_2 = \sqrt{E_b}$. The estimated symbol is denoted by $\hat{x}$.

For the transmission of each bit, the conditional probability densities are equal to the probability densities of a Gaussian random variable centered at x_1 and x_2, respectively:

$$p(y|(x_1)) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(y-x_1)^2/(2\sigma^2)} \qquad [2.53]$$

and

$$p(y|(x_2)) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(y-x_2)^2/(2\sigma^2)} \qquad [2.54]$$

At the receiver, a detection threshold S is set. The following detection rule is applied: if $y < S$, then the estimated symbol $\hat{x}$ is equal to x_1. Else, it is equal to x_2. Let us now evaluate the optimum threshold value S that allows us to minimize the symbol error probability.

The symbol error probability is equal to the weighted sum of the probability to detect $\hat{x} = x_1$, knowing that x_2 was transmitted, and of the probability to detect $\hat{x} = x_2$, knowing that x_1 was transmitted:

$$\begin{aligned} P_e &= \sum_{i=1}^{2} Pr(x_i) \times p(\hat{x} \neq x_i) \\ &= Pr(x_2) \int_{-\infty}^{S} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y-x_2)^2}{2\sigma^2}} dy \\ &\quad + Pr(x_1) \int_{S}^{\infty} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y-x_1)^2}{2\sigma^2}} dy \end{aligned} \qquad [2.55]$$

where $Pr(x_i)$ is the probability to transmit x_i.

The optimum threshold is obtained when the error is minimized, which means that $\frac{\partial P_e}{\partial S} = 0$. This is equivalent to:

$$\begin{aligned} & Pr(x_2) e^{-\frac{(S-x_2)^2}{2\sigma^2}} = Pr(x_1) e^{-\frac{(S-x_1)^2}{2\sigma^2}} \\ & \Leftrightarrow 2\sigma^2 \ln\left(\frac{Pr(x_2)}{Pr(x_1)}\right) = (S-x_2)^2 - (S-x_1)^2 \end{aligned} \qquad [2.56]$$

where, since $x_1 = -\sqrt{E_b}$ and $x_2 = -x_1$, equality [2.56] becomes:

$$\begin{aligned} & 2\sigma^2 \ln\left(\frac{Pr(\sqrt{E_b})}{Pr(-\sqrt{E_b})}\right) = -4\sqrt{E_b} S \\ & \Leftrightarrow S = \frac{\sigma^2}{2\sqrt{E_b}} \ln\left(\frac{Pr(-\sqrt{E_b})}{Pr(\sqrt{E_b})}\right) \end{aligned} \qquad [2.57]$$

If both symbols have equal probability, the optimum threshold is $S = 0$. Thus, the error probability P_e is equal to:

$$\begin{aligned} P_e &= \frac{1}{2}\int_{-\infty}^{0} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y-\sqrt{E_b})^2}{2\sigma^2}} dy + \frac{1}{2}\int_{0}^{\infty} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y+\sqrt{E_b})^2}{2\sigma^2}} dy \\ &= \frac{1}{2\sqrt{\pi}} \int_{-\infty}^{\frac{-\sqrt{E_b}}{\sqrt{2}\sigma}} e^{-x^2} dx + \frac{1}{2\sqrt{\pi}} \int_{\frac{\sqrt{E_b}}{\sqrt{2}\sigma}}^{\infty} e^{-x^2} dx \end{aligned} \qquad [2.58]$$

where we used two changes of variable: $x = \frac{y-\sqrt{E_b}}{\sqrt{2}\sigma}$ in the first integral, and $x = \frac{y+\sqrt{E_b}}{\sqrt{2}\sigma}$ in the second integral. After applying the change of variable $u = -x$ in the first integral, we obtain:

$$\begin{aligned} P_e &= \frac{1}{2\sqrt{\pi}} \int_{\frac{\sqrt{E_b}}{\sqrt{2}\sigma}}^{\infty} e^{-u^2} du + \frac{1}{2\sqrt{\pi}} \int_{\frac{\sqrt{E_b}}{\sqrt{2}\sigma}}^{\infty} e^{-x^2} dx \\ &= \frac{1}{\sqrt{\pi}} \int_{\sqrt{\frac{E_b}{2\sigma^2}}}^{\infty} e^{-x^2} dx \end{aligned} \qquad [2.59]$$

The symbol error probability is equal to the bit error rate, $P_e = \text{BER}$, since the transmitted symbols are bits. In the next section, we will express the bit error rate (BER) with common error functions.

2.4.4.2. *Error functions*

Two error functions can be found in the literature:

– the area underlain by the canonical normal curve from a to $+\infty$:

$$\mathrm{Q}(a) = \frac{1}{\sqrt{2\pi}} \int_{a}^{\infty} \exp\left(-\frac{z^2}{2}\right) dz \qquad [2.60]$$

– erf and erfc functions:

$$\mathrm{erf}(a) = \frac{2}{\sqrt{\pi}} \int_{-\infty}^{a} \exp(-z^2) dz \qquad [2.61]$$

$$\mathrm{erfc}(a) = 1 - \mathrm{erf}(a) = \frac{2}{\sqrt{\pi}} \int_{a}^{+\infty} \exp(-z^2) dz \qquad [2.62]$$

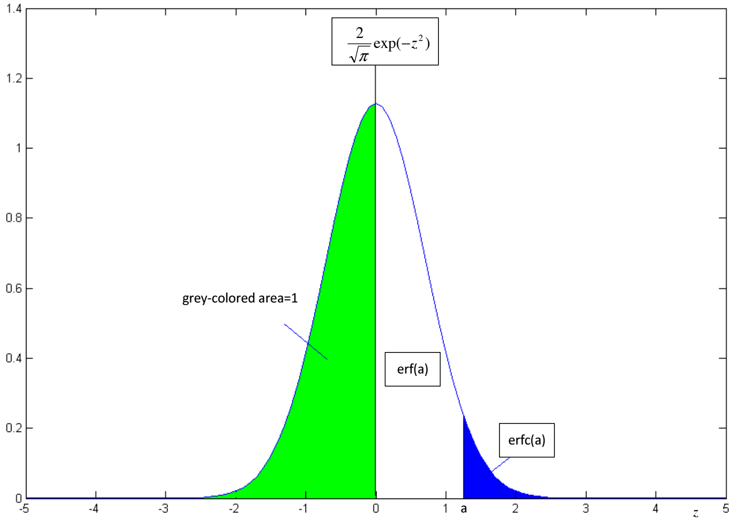

Figure 2.31. *erf and erfc functions*

The erfc function can be approximated by a sum of two exponential functions as follows [LOS 09]:

$$\mathrm{erfc}(a) \approx 0,416\exp(-1,942a^2) + 0,294\exp(-1,05x^2) \qquad [2.63]$$

The Q function is deduced from erfc function by:

$$\mathrm{Q}(a) = \frac{1}{2}\mathrm{erfc}\left(\frac{a}{\sqrt{2}}\right) \qquad [2.64]$$

and the erfc function is deduced from the Q function by:

$$\mathrm{erfc}(a) = 2\mathrm{Q}(\sqrt{2}a) \qquad [2.65]$$

As a result, the Q function can also be approximated by two exponential functions:

$$\mathrm{Q}(a) \approx 0,208\exp(-0,971a^2) + 0,147\exp(-0,525x^2) \qquad [2.66]$$

a	erfc(a)	a	erfc(a)	a	erfc(a)
0.0	1.0000e+000	1.7	1.6210e-002	3.4	1.5220e-006
0.1	8.8754e-001	1.8	1.0909e-002	3.5	7.4310e-007
0.2	7.7730e-001	1.9	7.2096e-003	3.6	3.5586e-007
0.3	6.7137e-001	2.0	4.6777e-003	3.7	1.6715e-007
0.4	5.7161e-001	2.1	2.9795e-003	3.8	7.7004e-008
0.5	4.7950e-001	2.2	1.8628e-003	3.9	3.4792e-008
0.6	3.9614e-001	2.3	1.1432e-003	4.0	1.5417e-008
0.7	3.2220e-001	2.4	6.8851e-004	4.1	6.7000e-009
0.8	2.5790e-001	2.5	4.0695e-004	4.2	2.8555e-009
0.9	2.0309e-001	2.6	2.3603e-004	4.3	1.1935e-009
1.0	1.5730e-001	2.7	1.3433e-004	4.4	4.8917e-010
1.1	1.1979e-001	2.8	7.5013e-005	4.5	1.9662e-010
1.2	8.9686e-002	2.9	4.1098e-005	4.6	7.7496e-011
1.3	6.5992e-002	3.0	2.2090e-005	4.7	2.9953e-011
1.4	4.7715e-002	3.1	1.1649e-005	4.8	1.1352e-011
1.5	3.3895e-002	3.2	6.0258e-006	4.9	4.2189e-012
1.6	2.3652e-002	3.3	3.0577e-006	5.0	1.5375e-012

Table 2.3. *erfc function table*

Using equation [2.59] and $\sigma^2 = \frac{N_0}{2}$, the bit error rate is expressed as a function of the SNR as follows:

$$\text{BER} = \frac{1}{2}\text{erfc}\left(\sqrt{\frac{E_b}{N_0}}\right) \qquad [2.67]$$

It can also be expressed with function Q:

$$\text{BER} = \text{Q}\left(\sqrt{\frac{2E_b}{N_0}}\right) \qquad [2.68]$$

The BER depending on E_b/N_0 is represented in Figure 2.32.

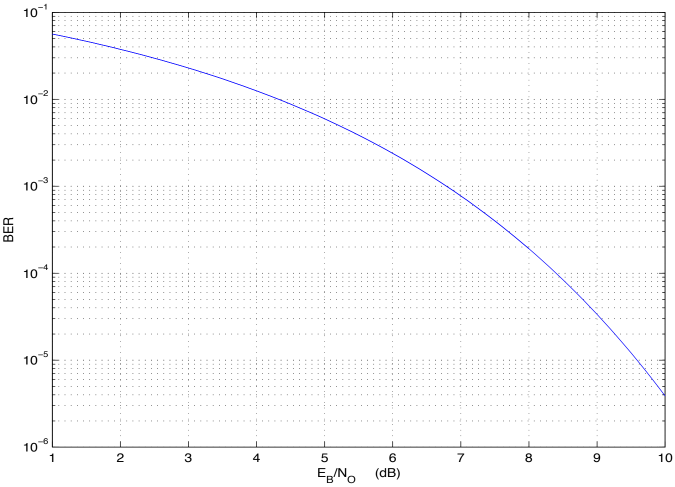

Figure 2.32. *BER as a function of E_b/N_0 for NRZ binary line code on the additive white Gaussian noise channel*

2.4.4.3. *Pairwise error probability: scalar case*

Let x_i and x_j be two symbols with Euclidean distance $d(x_i, x_j) = |x_i - x_j|^2$. On the additive white Gaussian noise, the probability $Pr(x_i \rightarrow x_j)$ that y is closer to x_j than it is to the transmitted symbol x_i is given by:

$$Pr(x_i \rightarrow x_j) = \frac{1}{2}\text{erfc}\left(\frac{d(x_i, x_j)}{2\sqrt{N_0}}\right) \qquad [2.69]$$

$Pr(x_i \rightarrow x_j)$ is called the pairwise error probability.

In the NRZ line code case, the Euclidean distance is equal to $2\sqrt{E_b}$. Thus, we recognize the expression of the bit error rate from equation [2.67] when both symbols have equal probability.

2.4.4.4. *Estimation of the BER with the Monte-Carlo method*

The BER can be statistically estimated with the Monte-Carlo method. This may be necessary when an analytical expression of the theoretical BER cannot exactly be obtained, unlike in the binary additive white Gaussian noise channel case.

The Monte-Carlo method allows estimating the mean of a random variable from K independent outcomes of this variable. It converges in $o(1/\sqrt{K})$. For instance, if we wish to numerically check formula [2.67], we must randomly draw K outcomes of the transmitted signal, belonging to $\{-\sqrt{E_b}, \sqrt{E_b}\}$ and K outcomes of a white Gaussian noise, centered and of variance σ^2. The k^{th} received signal is then:

$$y_k = x_k + n_k \tag{2.70}$$

Detection by threshold implies that the k^{th} estimated signal is then equal to:

$$\widehat{x}_k = \text{sign}(y_k)\sqrt{E_b} \tag{2.71}$$

An error has occurred if $\widehat{x}_k$ is different from x_k:

$$e_k = \begin{cases} 1 \text{ if } \widehat{x}_k \neq x_k \\ 0 \text{ if } \widehat{x}_k = x_k \end{cases}$$

e_k is the outcome of a random variable Er which modeled the bit error rate. The Monte-Carlo method leads to an estimation of the BER in the following way:

$$BER = E[Er] \approx \frac{1}{K}\sum_{k=1}^{K} e_k \tag{2.72}$$

The estimate's reliability increases with the number of outcomes. Since e_k are independent and identically distributed, applying the limit central theorem to [2.72] leads to an error estimate of $\epsilon_K = TEB - \frac{1}{K}\sum_{k=1}^{K} e_k$ which tends to a centered Gaussian random variable when K tends to infinity.

Let $\sigma_e^2 = E[Er^2] - (E[Er])^2$ be the variance of the random variable Er. Then, the standard deviation of the error estimate ϵ_K is equal to $\frac{\sigma_e}{\sqrt{K}}$.

2.5. Nyquist criterion

2.5.1. *Introduction*

Up to now, we have considered digital communications on channels with infinite bandwidths. Nevertheless, in many applications (such as phone modems, microwave terrestrial links, satellite communications, wireless mobile communications, etc.), the bandwidth is limited, and the objective is to transmit with the highest possible data rate in the given bandwidth.

The transmit filter's frequency response defines the spectral characteristics of the transmission. Since the signal has a limited bandwidth, the transmit filter cannot be a rectangular function of duration T. Indeed, the frequency response of such a transmit filter is $G(f) = \frac{\sin(\pi t/T)}{\pi t/T}$, whose spectral band is infinite. As a result, it is necessary to determine finite-band transmit filters that also allow us to remove intersymbol interference at the receiver. This section aims at determining these transmit filters.

Nevertheless, we can also note that a bandwidth can be defined for signals shaped with an infinite-band transmit filter, such as the rectangular function. Among the different definitions, we can distinguish the bandwidth limited to a ratio of the power. It corresponds to the frequency interval that contains the ratio $(1 - \epsilon)$ of the signal's total power. If $g(t)$ is a rectangular function of duration T, the bandwidth limited to 99% of the power corresponds to 10.2 times the width of the main sidelobe, whereas the bandwidth limited to 90% of the power only corresponds to 0.85 times the width of the main sidelobe.

As a result, some of the proposed exercises may consider, to simplify, that $g(t)$ is a rectangular function. Thus, the bandwidth is defined, by an abuse of language, with one of these definitions.

If the transmit bandwidth is limited, the impulse response of the transmit filter $g(t)$ is of infinite duration. Thus, the main problem consists of finding that transmit filter so that the transmitted samples can be fully recovered at the receiver, at each symbol period T.

If there is no intersymbol interference, the receiver can estimate each symbol independently of the others, since each symbol is simply corrupted by noise.

2.5.2. *Transmission channel*

In the remainder of this chapter, we assume that the signal transmitted on the baseband channel is given by equation [2.1], where $[a_k]_{k=-\infty}^{\infty}$ is a set of real symbols, and $g(t)$ is the transmit filter. This signal is then modified by the transmission channel, which is in general modeled by a linear filter with impulse response $g_c(t)$. In addition, the channel adds a white Gaussian noise $b(t)$.

The received signal $q(t)$ is equal to:

$$\begin{aligned} q(t) &= x(t) * g_c(t) + b(t) \\ &= \int_{-\infty}^{+\infty} g_c(\tau)x(t-\tau)d\tau + b(t) \end{aligned}$$

$q(t)$ is then filtered by a receive filter with impulse response $g_r(t)$. The signal after filtering, $y(t)$, is then written as:

$$\begin{aligned} y(t) &= x(t) * g_c(t) * g_r(t) + n(t) \\ &= \sum_{k=-\infty}^{+\infty} a_k g(t-kT) * g_c(t) * g_r(t) + n(t) \\ &= \sum_{k=-\infty}^{+\infty} a_k p(t-kT) + n(t) \end{aligned} \qquad [2.73]$$

where $n(t) = b(t) * g_r(t)$ is the noise after filtering.

Consequently, the equivalent filter $p(t) = g(t) * g_c(t) * g_r(t)$ is the impulse response of the convolution of the transmit filter, transmission channel and receive filter. Its frequency response is:

$$P(f) = G(f) \times G_c(f) \times G_r(f) \qquad [2.74]$$

$y(t)$ is then sampled at symbol period T. At each time kT, it is equal to:

$$\begin{aligned} y(kT) &= \sum_{i=-\infty}^{+\infty} a_i p(kT - iT) + n(kT) \\ &= a_k p(0) + \sum_{i \neq k}^{+\infty} a_i p((k-i)T) + n(kT) \end{aligned} \quad [2.75]$$

This expression contains three elements:

– the first element is proportional to the k^{th} transmitted symbol;

– the second element is the contribution of all other transmitted symbols on sample $y(kT)$. It is called intersymbol interference;

– the third element is the noise contribution.

Both intersymbol interference and filtered noise decrease the performances of the transmission system, since they prevent the receiver from correctly estimating the symbol transmitted at time kT. As a result, both transmit and receive filters must be chosen so as to minimize them.

2.5.3. *Eye diagram*

The eye diagram allows us to visualize intersymbol interferences. It consists of superimposing all sections of duration T of signal $y(t)$. Two examples are presented in Figures 2.33 and 2.34 for a binary signal ($a_k = \pm 1$) where the total filter is a raised cosine.

Figures 2.33 and 2.34 graphically illustrate the impact of intersymbol interference and noise on the eye diagram. We can notice that even in the absence of noise, the chosen filter tends to close the eye diagram, which may make the decisions on the sampling time and on the symbol's amplitude more complicated.

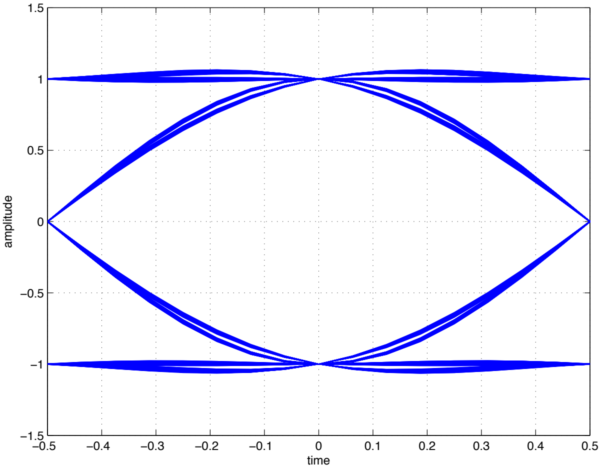

Figure 2.33. *Eye diagram of a binary signal with raised cosine filter when* $\alpha = 1$

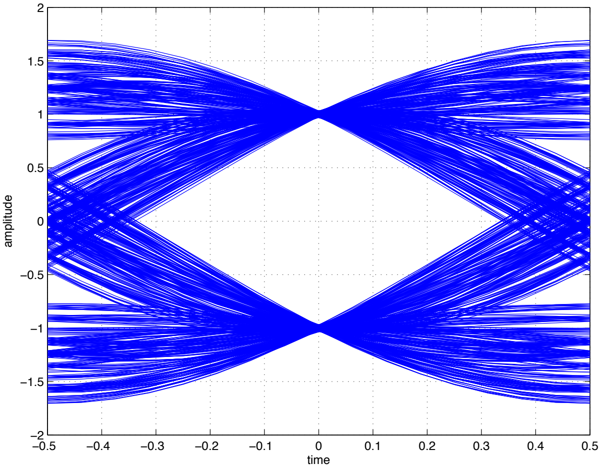

Figure 2.34. *Eye diagram of a binary signal with raised cosine filter when* $\alpha = 0,35$

2.5.4. *Nyquist criterion*

In order to guarantee the absence of intersymbol interference, according to equation [2.75], the complete filter $p(t)$ must verify the following conditions:

$$p(kT) = \begin{cases} 1 & \text{if} \quad k = 0 \\ 0 & \text{if} \quad k \neq 0 \end{cases} \qquad [2.76]$$

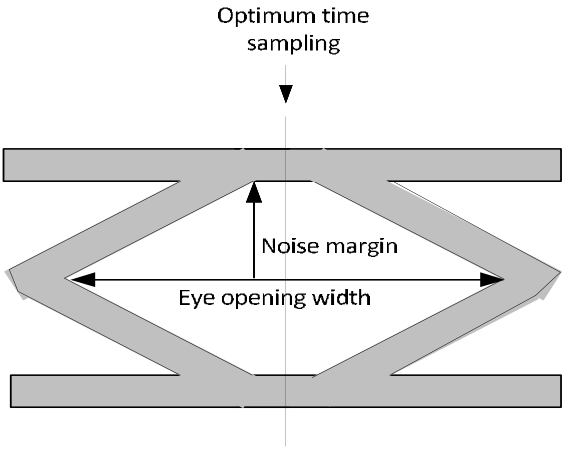

Figure 2.35. *Eye diagram*

In the frequency domain, Nyquist criterion becomes [PRO 08]:

$$\sum_{i=-\infty}^{+\infty} P\left(f+\frac{i}{T}\right)=T \qquad [2.77]$$

The channel transmission's bandwidth is denoted by B. It is defined as follows: $G_c(f)=0$ when $f>B$.

Whether criterion [2.77] is feasible depends on the bandwidth B and the symbol period T. Its feasibility is represented in Figure 2.36. There are three cases:

– if $B<\frac{1}{2T}$, then there is no filter $P(f)$ that verifies the Nyquist criterion;

– if $B=\frac{1}{2T}$, a unique solution exists: $P(f)$ must be the frequency response of a perfect lowpass filter with cutoff frequency B:

$$P(f)=\begin{cases} T & \text{if} \quad |f|<B=\frac{1}{2T} \\ 0 & \text{elsewhere} \end{cases} \qquad [2.78]$$

Its impulse response is:

$$p(t)=\frac{\sin(\pi t/T)}{\pi t/T} \qquad [2.79]$$

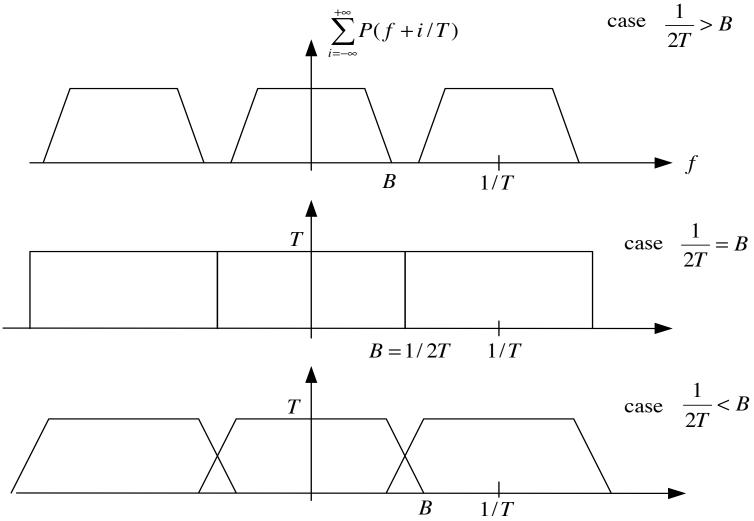

Figure 2.36. *Frequency responses* $\sum P(f+i/T)$ *for* $1/2T > B$, $1/2T = B$ *and* $1/2T < B$

It is represented in Figure 2.37. We must outline that a perfect lowpass filter with tight cutoff at frequency $f = 1/(2T)$ cannot physically be implemented. Moreover, the impulse response of this filter has a low decrease, in $1/t$, and is of infinite duration. Thus, low time synchronization errors would lead to high levels of intersymbol interference, if this filter was used;

– if $B > \frac{1}{2T}$, then several filters $P(f))$ verify the Nyquist criterion. All classes of filters whose frequency response is symmetrical with respect to the point of coordinates $(0.5T, 0.5T)$ satisfy constraint [2.77]. Among these filters, the set of raised cosine filters is of particular interest, since it is smooth, which implies that its implementation is easier. In addition, raised cosine filters can be defined in order to verify the channel's spectrum constraints.

The frequency response of a raised cosine filter is:

$$P(f) = \begin{cases} T & \text{if} \quad 0 \leq |f| \leq \frac{1-\alpha}{2T} \\ T\cos^2\left(\frac{\pi}{4\alpha}(2fT-(1-\alpha))\right) & \text{if} \quad \frac{1-\alpha}{2T} < |f| < \frac{1+\alpha}{2T} \\ 0 & \text{if} \quad |f| \geq \frac{1+\alpha}{2T} \end{cases} \quad [2.80]$$

where α is the roll-off factor, lying between 0 and 1. The perfect lowpass filter with minimum bandwidth is obtained when $\alpha = 0$; nevertheless, the eye diagram is then the most closed. On the contrary, the bandwidth is maximum when α is equal to 1, but then the eye diagram is the most open. In practice, the roll-off factor is generally chosen between 0.22 and 0.35.

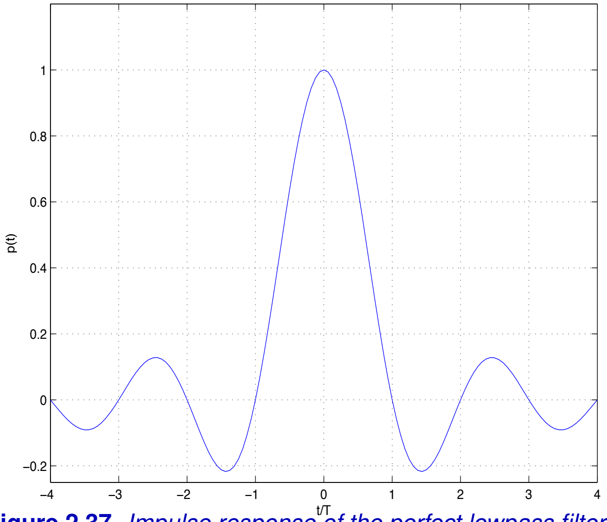

Figure 2.37. *Impulse response of the perfect lowpass filter*

The frequency response of a raised cosine filter is represented in Figure 2.38 for $\alpha = 1$ and $\alpha = 0.35$.

Its impulse response $p(t)$ is the following:

$$p(t) = \frac{\sin(\pi t/T)}{\pi t/T} \frac{\cos(\alpha \pi t/T)}{1 - 4\alpha^2 t^2/T^2} \qquad [2.81]$$

It is represented for several values of α in Figure 2.39.

2.5.5. *Transmit and receive filters with matched filter*

In section 2.4.3.3, we showed that the optimum receive filter is the matched filter. If we consider an additive white Gaussian channel (such $G_c(f) = 1$ in the considered bandwidth), the matched filter is:

$$G_r(f) = G^*(f)e^{-j2\pi fT} \qquad [2.82]$$

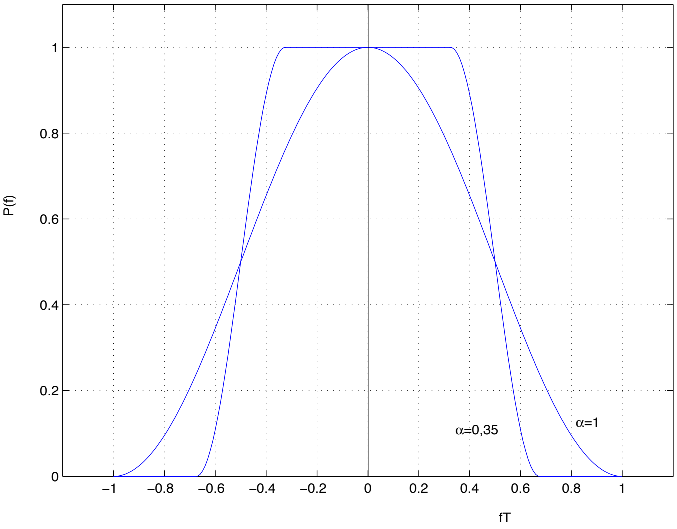

Figure 2.38. *Frequency response of the raised cosine filter*

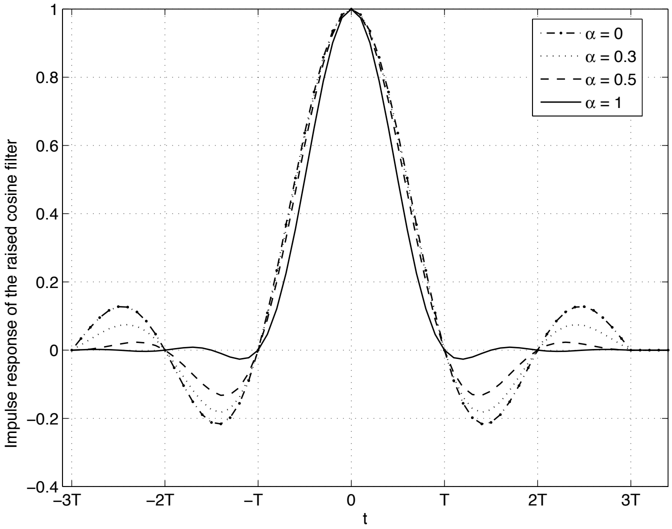

Figure 2.39. *Impulse response of the raised cosine filter*

The complete filter, including both transmission and reception, is consequently equal to:

$$P(f) = G(f)G^*(f)e^{j2\pi fT} = |G(f)|^2 e^{-j2\pi fT} \quad [2.83]$$

Thus, the receive and transmit filters can be decomposed from $P(f)$ as follows:

$$G(f) = \sqrt{P(f)}e^{-j\psi(f)}$$

$$G_r(f) = \sqrt{P(f)}e^{j(\psi(f)-2\pi fT)} \qquad [2.84]$$

where $\psi(f)$ is a real phase.

If $P(f)$ is a raised cosine filter (and we do not take into account the phase delay in [2.83]), then, in practice, the raised cosine filter is separated into two identical filters, called root raised cosine filters, whose frequency response is:

$$G(f) = G_r(f) = \begin{cases} \sqrt{T} & \text{if} \quad 0 \leq |f| \leq \frac{1-\alpha}{2T} \\ \sqrt{T}\cos\left(\frac{\pi}{4\alpha}(2fT-(1-\alpha))\right) & \text{if} \quad \frac{1-\alpha}{2T} < |f| < \frac{1+\alpha}{2T} \\ 0 & \text{if} \quad |f| \geq \frac{1+\alpha}{2T} \end{cases} \qquad [2.85]$$

2.6. Conclusion

This chapter has studied baseband transmissions on a channel with infinite spectrum, and then on a channel with finite bandwidth. The main line codes used with infinite spectrum channel have been detailed. The main advantages and drawbacks of each line code have been presented, both in terms of the main lobe width of their power spectrum density and the ability to recover the symbol period.

The additive white Gaussian noise channel and the general block diagram of the transmission modulator have then been introduced. For this channel, we proved that the optimal receiver's block diagram can be either based on the correlator, or on the matched filter, and that both techniques are equivalent. The performances of the binary NRZ line code were then evaluated on the additive white Gaussian noise channel with optimum reception.

Finally, the last section of the chapter concerned the finite bandwidth channel. We proved that in order to cancel intersymbol interference at the receiver, the equivalent channel must verify Nyquist criterion. The expression of the transmit filter when the equivalent channel is a raised cosine filter and the matched filter is used has been obtained.

Most notions studied in this chapter will serve as basis to those presented in Chapter 3 on modulations on a sine waveform. In addition, in this chapter, we assumed that the receiver is perfectly synchronized with the receiver; the synchronization techniques will be studied in Chapter 4. Finally, in the same chapter, we will see how to treat data when the Nyquist criterion is no longer verified.

2.7. Exercises

2.7.1. *Exercise 1: power spectrum density of several line codes*

1) Determine the power spectrum of the elementary impulse represented in Figure 2.40.

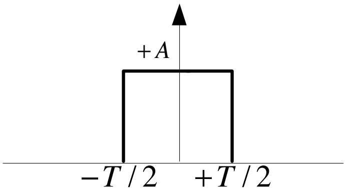

Figure 2.40. *Exercise 5.1, elementary impulse*

2) For each following line code, determine the autocorrelation function of the associated symbols' sequence, and then compute its power spectrum density. Draw the power spectrum density of the line code when using the elementary impulse from Figure 2.40. We consider:

a) the NRZ code;

b) the RZ unipolar code;

c) the AMI code.

2.7.2. *Exercise 2: Manchester code*

Manchester code associates with each bit 1 a level $+A$ during $T/2$ and then a level $-A$ during $T/2$. With each bit 0 is associated a level $-A$ during $T/2$ and then a level $+A$ during $T/2$.

1) Graphically represent the coded signal $x(t)$ for the following input sequence a_k: 1001011.

2) Express Bennett's formula, which allows us to compute the power spectrum density of $x(t)$.

3) Let d_k be the sequence obtained from $x(t)$ by dividing its period by 2 compared to a_k. Compute the autocorrelation function $\Phi_{dd}(i)$ of sequence d$_k$ for $i = 0$, 1 and -1. To achieve this, two cases must be considered, depending on whether the sequence starts at time 0 or $T/2$.

4) Show that $\Phi_{dd}(f) = 2\sin^2 \frac{\pi f T}{2}$.

5) Deduce the power spectrum density of $x(t)$.

2.7.3. *Exercise 3: study of a magnetic recording system*

This exercise deals with a magnetic recording system such as a hard drive. The transmission chain model is represented in Figure 2.41.

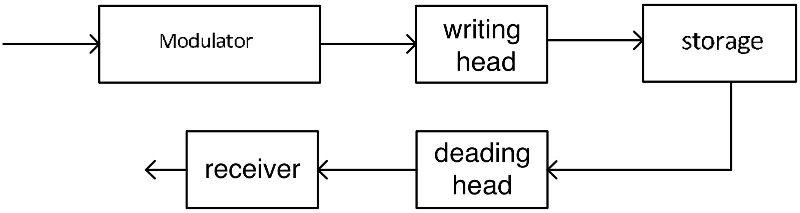

Figure 2.41. *Exercise 5.3, transmission chain*

The signal at the output of the reading head relative to a positive step (which means that a transition from -1 to 1 takes place) may be modeled as follows:

$$s(t) = \frac{1}{1 + (2t/T_h)^2}$$

For a negative step, the signal is $-s(t)$.

1) Draw $s(t)$ when $Th = 1$.

We consider a positive step followed by a negative step. We assume that time $T_b = \frac{T_h}{2}$ takes place between two steps.

2) Draw the response to these two consecutive steps. Is there any intersymbol interference?

To remove this interference, a constrained code is added. It has the following parameters:

– d_0: minimum number of 0 between two 1;

– k_0: maximum number of 0 between two 1, whatever the output signals succession is.

This constrained code is denoted as (d_0,k_0).

Let the following list be the set of codewords from a constrained code used on some hard drives from IBM:

Input sequence	Output sequence
10	1000
11	0100
011	000100
010	001000
000	100100
0011	00100100
0010	00001000

3) Determine the code rate and its parameters d_0 and k_0.

We now consider the constrained code defined by the following encoding table:

Input bit b_k	Output bits (u_k v_k)
0	x 0
1	0 1

where $x = 0$ if b_{k-1}=1 and $x = 1$ if b_{k-1}=0.

4) Express the output bits, u_k and v_k, as functions of b_k and b_{k-1}.

5) Determine the state diagram of this constrained code.

6) Determine the rate of that code and its parameters d_0 and k_0.

We consider the input sequence represented in Figure 2.42.

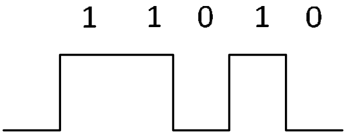

Figure 2.42. *Exercise 5.3, input sequence*

7) Determine the output sequence of the constrained code. We will assume that the initial state is zero.

We associate this code with an NRZI line code. We recall that the NRZI line code is defined by the mathematical equation $p_k = d_k \oplus p_{k-1}$, where $\{d_k\}$ is the sequence to be encoded.

8) Represent the state diagram of NRZI line code.

9) Represent the hardware block diagram of NRZI line code.

10) Determine the diagram state of the set constrained code + NRZI line code. The number of states is equal to the product of the number of states of both codes.

11) Determine the output sequence of the NRZI line code from the previously computed sequence. Identify this code. Draw the shape of the signal at the output of the reading hard and conclude.

2.7.4. *Exercise 4: line code and erasure*

Figure 2.43 represents the equivalent model of the transmission chain studied in this exercise.

The noise samples are independent and follow a Gaussian distribution, centered with variance $\sigma^2 = \frac{No}{2}$.

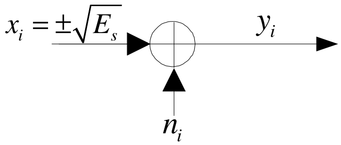

Figure 2.43. *Exercise 5.4, equivalent model*

The following decision is taken on y_i:

$$\begin{cases} \text{if } y_i \geq 0 \text{ then } \hat{c} = 1 \\ \text{if } y_i < 0 \text{ then } \hat{c} = 0 \end{cases}$$

1) Express the bit error rate as a function of the $\frac{E_s}{N_0}$ ratio.

2) Perform a numerical application when E_s=1 and $\sigma^2 = 1$.

The decision process is modified as follows: if the received sample lies between $-d$ and $+d$, an erasure has taken place, and the symbol must be retransmitted.

We assume that the source transmits $x_i = -\sqrt{E_s}$. We set: P=$Pr(y_i \geq +d|x_i = -\sqrt{E_s})$ and P_A=$Pr(-d < y_i < +d|x_i = -\sqrt{E_s})$.

Retransmissions take place as much as needed. Express P and P_A as functions of d, $\sqrt{E_s}$ and $\sqrt{N_0}$.

3) Express the error probability P_e as a function of P and P_A, using the following mathematical sequence:

$$1 + x + x^2 + ... = \frac{1}{1-x}$$

4) Express the average number of transmitted samples $E(n)$ as a function of P_A.

5) Express P_e as a function of P and $E(n)$.

6) Perform a numerical application when d=0.5.

2.7.5. *Exercise 5: 4 levels NRZ modulation*

We consider a baseband bimodal distribution ($x_i = \pm\sqrt{E_s}$) on the additive white Gaussian noise channel with unilateral power spectrum density N_0.

1) Provide the equivalent scheme after matched filter and sampling.

We recall that the pairwise error probability $P_{e,\text{pair}}$ has the following expression: $P_{e,\text{pair}} = \frac{1}{2}\text{erfc}(\frac{d_{eucl}}{2\sqrt{N_0}})$ where d_{eucl} is the Euclidean distance between two points.

2) Determine the symbol error probability and the bit error rate as functions of the E_b/N_0 ratio. Make a numerical application when $(E_b/N_0)_{dB}$ =11dB.

We now consider a baseband NRZ transmission with 4 levels ($x_i = -3A, -A, +A, +3A$).

3) Express A as a function of $\sqrt{E_s}$ and$\sqrt{E_b}$.

4) Determine the symbol error rate.

5) Compute the symbol error rate and bit rate, assuming that an erroneous symbol is equivalent to an erroneous bit (Gray coding). Make a numerical application when E_b/N_0 =11dB.

6) We add an error correcting code of rate R=1/2 and coding gain 4dB. Is the NRZ-4 transmission with that code more efficient than the NRZ binary line code without an error correcting code ?

2.7.6. *Exercise 6: Gaussian transmit filter*

We consider a communication system using a Gaussian transmit filter as follows:

$$x(t) = \exp(-\pi a^2 t^2)$$

Its Fourier transform is equal to:

$$X(f) = \frac{1}{a}\exp(-\frac{\pi f^2}{a^2})$$

1) In order to limit the intersymbol interference threshold, determine a to obtain $x(T) = 0.01$, where T is the symbol period.

2) We define the signal's bandwidth B_W as follows: $X(B_W)/X(0) = 0.01$. Determine the value of B_W. Compare this bandwidth with that of a raised cosine filter with roll-off factor equal to 1.

2.7.7. *Exercise 7: Nyquist criterion*

We consider a transmission system using a binary modulation ($a_k = \pm 1$). The transmitted signal can be written as:

$$x(t) = \sum_k a_k g(t - kT)$$

T is the binary symbol period, a_k are the binary symbols, which are mutually uncorrelated. They can take values $\pm$ 1 with the same probability. $g(t)$ is the impulse response of the transmit filter and $G(f)$ is its Fourier transform.

We consider an ideal channel with frequency response $G_c(f) = 1$. The equivalent receiver block diagram is represented in Figure 2.44.

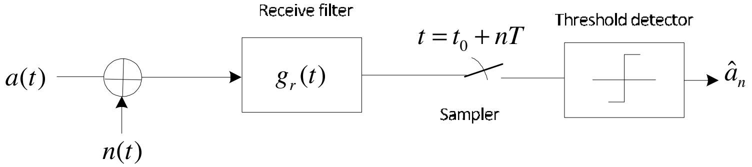

Figure 2.44. *Exercise 5.7, equivalent model*

The receive noise $n(t)$ is additive, white, Gaussian and centered. $G_r(f)$ is the Fourier of the impulse response $g_r(t)$ of the receive filter.

We assume that $G(f)$ and $G_r(f)$ are equal and have the same function $H(f)$, equal to:

$$H(f) = \begin{cases} \sqrt{T}\cos(\pi f T/2) & \text{if} \quad |f| < \frac{1}{T} \\ 0 & \text{elsewhere} \end{cases}$$

1) Determine the complete frequency response of the transmission chain in the absence of noise. Draw this frequency response and check if it verifies Nyquist criterion.

2) Find the expression of signal $y(t)$ at the output of the receive filter. Check that there is no intersymbol interference.

2.7.8. *Exercise 8: raised cosine filter*

Determine the theoretical and practical frequencies for which a raised cosine filter leads to an attenuation of 30 dB. Symbols are transmitted with a data rate D_{symb}=1 Msymb/s. Evaluate it numerically when the roll-off factor is $\alpha = 0.3$ or $\alpha = 0.5$.

3

Digital Modulations on Sine Waveforms

3.1. Introduction

In Chapter 2, baseband digital modulations were studied. In this case, the line codes can transmit bits or symbols representing groups of bits, as for M-ary non return to zero (NRZ) transmissions with $M = 2^n$ and $n > 1$. These transmissions are well adapted to wired cable media where the bandwidth constraint is not strong. In other systems, and in particular in wireless mobile communication systems, large bandwidth cannot be used for several reasons. First, the radio spectrum available for the systems is limited. Moreover, it is important to transmit in spectrum bands where propagation has the required characteristics. Finally, interferences between competing systems using the same bandwidth generate major problems, since propagation may lead to interferences between far-away transmitter-receiver pairs, contrary to wired transmissions that are, in principle, isolated from external interferences.

Consequently, digital transmissions on sine waveforms with high spectral efficiencies must be defined in limited bandwidth, around potentially high carrier frequencies. For instance, in civilian cellular wireless communication systems, the carrier frequencies currently used lie between 800 MHz and 2.6 GHz and their bandwidth lies between 200 kHz and 5 MHz. They are called narrowband and bandpass signals.

These signals can be transposed to equivalent baseband signals. In section 3.2, we will demonstrate that this property always holds. Yet, the transposition to the equivalent baseband signal generally transforms the

Digital Communications 2: Digital Modulations, First Edition. Mylène Pischella and Didier Le Ruyet.

original real signal into a complex equivalent lowpass signal, including both in-phase and quadrature components of the original signal. It is called the complex envelope of the complex baseband signal. The influence of filtering is studied, as well as the statistical properties of the complex envelope of a second-order random process. The specific case of the additive white Gaussian noise is detailed. These basic concepts are then necessary to study the linear bandpass digital modulations that are detailed in section 3.3.

After having defined their main characteristics, regular amplitude, phase and quadrature digital modulations (amplitude shift keying (ASK), phase shift keying (PSK) and quadrature amplitude modulation (QAM)) are presented. They are called amplitude, phase and quadrature shift-keying. Then, for the reference additive white Gaussian noise case, we introduce the threshold-based detection technique. The performances of each modulation in terms of symbol error rate and bit error rate (BER) as a function of the E_b/N_0 ratio and the signal-to-noise ratio are evaluated.

Finally, section 3.4 deals with frequency shift-keying (FSK) modulations that are nonlinear, and can consequently not be equivalently studied by their complex envelope. The constraint imposed to have continuous-phase FSK and the orthogonality constraint required to be able to demodulate the signal at the receiver are detailed. The chapter is concluded by a comparison of all these modulation techniques.

3.2. Passband transmission and equivalent baseband chain

3.2.1. *Narrowband signal*

3.2.1.1. *Introduction*

In Chapter 2, the considered signals were all baseband, which means that they were centered around frequency $f = 0$. Yet, digital signals carrying information on wireless channels are transmitted on a carrier frequency $f_c > 0$, and are then filtered by the channel, which only keeps the energy centered around f_c. Whatever the location of this carrier frequency in the spectrum, signals are transmitted in a limited bandwidth and centered around f_c. If the bandwidth is very low with respect to the carrier frequency, the signals are called narrowband.

Any real narrowband signal can be converted into an equivalent complex baseband signal. In this section, we explain the connections between a narrowband signal and its baseband equivalent signal. Then, we study the impact of the filtering of a narrowband signal by a passband channel centered around the same carrier frequency. The statistical characteristics of narrowband random variables and their complex envelope are then explained, with the white noise as an application example. Finally, synchronous detection is presented. It uses the equivalent baseband model to evaluate the transmitted data on a noisy passband channel.

NOTE.– We refer to the equivalent passband chain of a baseband chain by making an analogy with the complete channel filter, which is lowpass for the baseband signal, and passband for the narrowband signal.

3.2.1.2. *Representation of a narrowband signal*

Let $s(t)$ be a narrowband signal, centered on f_c and of bandwidth B. By definition, B is the bandwidth in the positive frequencies where the signal is different from zero. As a result, the bandwidth of a real signal is half its total frequency band.

The frequency response $S(f)$ is zero outside of bandwidth B: $f \notin \left[f_c - \frac{B}{2}; f_c + \frac{B}{2}\right] \cup \left[-f_c - \frac{B}{2}; -f_c + \frac{B}{2}\right]$. It is symmetrical, with values around f_c and $-f_c$, since the narrowband signal is real. An example of frequency response is given in Figure 3.1.

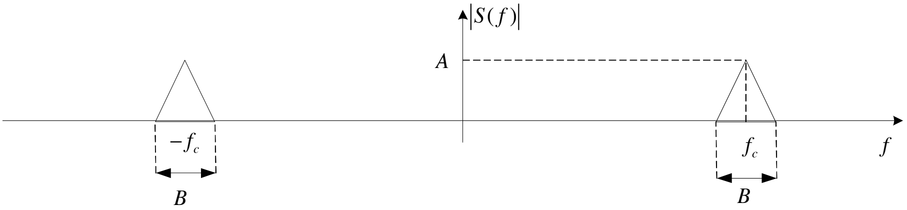

Figure 3.1. *Narrowband signal*

The frequency response can be separated into its values in positive frequencies, on the one hand, and its values in negative frequencies, on the other hand:

$$S(f) = \frac{1}{2}\left(S_+(f) + S_-(f)\right) \qquad [3.1]$$

where

$$S_+(f) = S(f).2U(f) \qquad [3.2]$$
$$S_-(f) = S(f).2U(-f)$$

where $U(f)$ is the step function, defined as follows: $U(f) = 1$ if $f > 0$ and $U(f) = 0$ elsewhere. Applying the inverse Fourier transform to $S_+(f)$ gives:

$$s_+(t) = s(t) + j\hat{s}(t) \qquad [3.3]$$

where $\hat{s}(t)$ is the Hilbert transform of signal $s(t)$. The Hilbert transform of a signal is the result of the filtering of this signal by a Hilbert filter. The Hilbert filter is linear with impulse response $h(t) = \frac{1}{\pi t}$, and frequency response $H(f) = -j$ if $f > 0$, $H(0) = 0$ and $H(f) = j$ if $f < 0$.

$s_+(t)$ is the analytic signal associated with $s(t)$.

Thus, while the original narrowband signal $s(t)$ is real, its associated analytic signal $s_+(t)$ is complex. This is due to the fact that its frequency response is not symmetrical.

The operation that transforms a real narrowband signal $s(t)$ into its analytic signal $s_+(t)$ is linear and time-invariant. Moreover, if $s(t)$ is a random Gaussian variable, then $s_+(t)$ is also a random Gaussian variable.

3.2.1.3. *Equivalent baseband signal*

The equivalent baseband signal's spectrum, denoted as $S_l(f)$, is obtained by translating $S_+(f)$ of f_c in order to center the signal on $f = 0$:

$$S_l(f) = S_+(f + f_c) \qquad [3.4]$$

The different steps required to pass from the original narrowband signal's spectrum to the equivalent baseband signal's spectrum are represented in Figure 3.2.

In the time domain, the equivalent baseband signal $s_l(t)$ is:

$$\begin{aligned} s_l(t) &= s_+(t)e^{-j2\pi f_c t} \\ &= [s(t) + j\hat{s}(t)]\, e^{-j2\pi f_c t} \qquad [3.5] \end{aligned}$$

$s_l(t)$ is also called the complex envelope of the real signal $s(t)$. It is defined as:

$$s_l(t) = s_c(t) + js_s(t) \quad [3.6]$$

where $s_s(t)$ and $s_c(t)$ are real functions. The original signal is obtained from the equivalent baseband signal as follows:

$$s(t) = \Re\left[[s_c(t) + js_s(t)]\, e^{j2\pi f_c t}\right] = \Re\left[s_l(t)e^{j2\pi f_c t}\right] \quad [3.7]$$

where $\Re[a]$ is the real component of a.

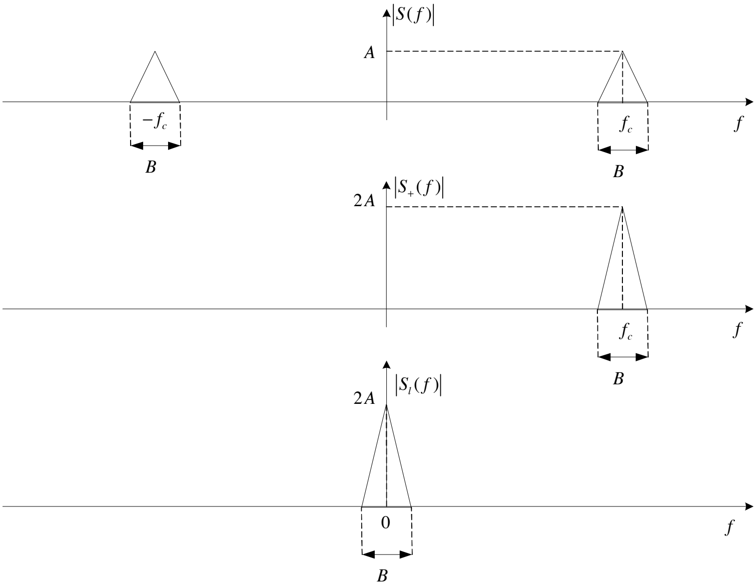

Figure 3.2. *Narrowband and equivalent baseband signals spectrum*

The narrowband signal can also be expressed as a function of its components $s_c(t)$ and $s_s(t)$:

$$s(t) = s_c(t)\cos(2\pi f_c t) - s_s(t)\sin(2\pi f_c t) \quad [3.8]$$

$s_c(t)$ is called the in-phase component, and $s_s(t)$ the quadrature component, since $s_c(t)$ and $s_s(t)$ respectively modulate carriers $\cos(2\pi f_c t)$

and $\sin(2\pi f_c t)$. Equation [3.8] shows that any narrowband signal can be written as a the sum of two baseband signals: an in-phase signal and a quadrature signal.

Besides equations [3.8] and [3.7], the signal can also be represented as a function of its envelope and phase.

$$s(t) = e(t)\cos\left(2\pi f_c t + \Psi(t)\right) \qquad [3.9]$$

where the envelope of $s(t)$ is defined as:

$$e(t) = \sqrt{s_c^2(t) + s_s^2(t)} \qquad [3.10]$$

It is real, since it is equal to the modulus of the complex envelope $s_l(t)$: $e(t) = |s_l(t)|$. The phase of $s(t)$ is equal to:

$$\Psi(t) = \tan^{-1}\left(\frac{s_s(t)}{s_c(t)}\right) \qquad [3.11]$$

3.2.1.4. *Spectrum*

The real narrowband signal's spectrum $S(f)$ is expressed depending on the equivalent baseband signal's spectrum $S_l(f)$ as follows:

$$\begin{aligned} S(f) &= \frac{1}{2}\left(S_+(f) + S_-(f)\right) \\ &= \frac{1}{2}\left(S_+(f) + S_+^*(-f)\right) \\ &= \frac{1}{2}\left[S_l(f - f_c) + S_l^*(-f - f_c)\right] \qquad [3.12] \end{aligned}$$

We used $S_-(f) = S_+^*(-f)$ to pass from the first to the second line. This is due to the fact that $s(t)$ is real.

3.2.2. *Filtering of a narrowband signal in a passband channel*

3.2.2.1. *Representation of a passband channel*

We consider a passband channel, modeled by a linear filter of impulse response $h(t)$ and frequency response $H(f)$. Since $h(t)$ is real, $H^*(-f) = H(f)$.

The frequency response of its equivalent baseband channel, H_l, is defined as follows:

$$\begin{cases} H_l(f - f_c) = H(f) \text{ if } f > 0 \\ H_l(f - f_c) = 0 \text{ elsewhere} \end{cases} \qquad [3.13]$$

The frequency response of the original channel is:

$$H(f) = H_l(f - f_c) + H_l^*(-f - f_c) \qquad [3.14]$$

Consequently, the impulse response of the passband channel can be expressed as a function of the impulse response of the equivalent passband channel:

$$h(t) = 2\Re\left[h_l(t)e^{j2\pi f_c t}\right] \qquad [3.15]$$

We can note that compared to the narrowband signal, coefficient 2 is added in equation [3.15]. This is due to the definition of the frequency response of the equivalent baseband channel.

3.2.2.2. *Filtering*

Let $s(t)$ be a real narrowband signal, whose equivalent baseband signal is $s_l(t)$. This signal is the input of a passband channel of impulse response $h(t)$. The impulse response of the equivalent baseband channel is denoted by $h_l(t)$. Both narrowband signal and passband channels have the same carrier f_c and the same bandwidth B.

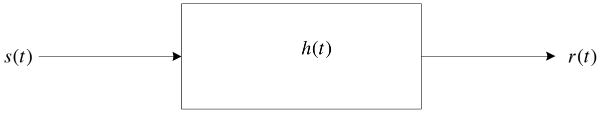

Figure 3.3. *Filtering of a narrowband signal by a passband signal*

Then, the output signal, $r(t)$, is narrowband and centered around f_c. It can be written as a function of its complex envelope:

$$r(t) = \Re\left[r_l(t)e^{j2\pi f_c t}\right] \qquad [3.16]$$

We now determine the relationship between the output equivalent baseband signal, $r_l(t)$, the input equivalent baseband signal $s_l(t)$ and the impulse response of the equivalent baseband channel $h_l(t)$. For this purpose, we consider the frequency domain:

$$\begin{aligned} R(f) &= S(f)H(f) \\ &= \frac{1}{2}\left[S_l(f-f_c) + S_l^*(-f-f_c)\right]\left[H_l(f-f_c) + H_l^*(-f-f_c)\right] \\ &= \frac{1}{2}\left[S_l(f-f_c)H_l(f-f_c) + S_l^*(-f-f_c)H_l^*(-f-f_c)\right] \\ &= \frac{1}{2}\left[R_l(f-f_c) + R_l^*(-f-f_c)\right] \end{aligned} \quad [3.17]$$

where the following notation has been introduced: $R_l(f) = S_l(f)H_l(f)$.

Equations [3.16] and [3.17] imply the following relation:

$$r(t) = \Re\left[(s_l * h_l)(t)e^{j2\pi f_c t}\right] \quad [3.18]$$

and finally: $r_l(t) = (s_l * h_l)(t)$.

Consequently, the impulse response of a narrowband passband channel can be directly obtained by considering the equivalent baseband functions. When a signal passes through a passband channel, its associated complex envelope is filtered by the equivalent baseband channel.

3.2.3. *Complex order of a second-order stationary random process*

Let us first recall that filtering by $h(t)$ with frequency response $H(f)$ changes a random variable $s(t)$ into a filtered random variable $r(t)$ of power spectrum density:

$$\gamma_{rr}(f) = \frac{1}{T}\gamma_{ss}(f)\left|H(f)\right|^2 \quad [3.19]$$

3.2.3.1. *Main characteristics of the signal and its complex envelope*

We assume that $s(t)$ is a centered, second-order stationary random variable with complex envelope $s_l(t)$. Equations [3.7] and [3.8] show that the complex envelope, the analytic signal as well as the in-phase and quadrature components are also centered and second-order stationary random variables.

3.2.3.2. *Power spectrum density*

Since the analytic signal $s_+(t)$ is the result of the filtering of a narrowband signal $s(t)$ by the filter of frequency response $2U(f)$, its power spectrum density can be expressed as, using equation [3.19]:

$$\gamma_{s_+s_+}(f) = \gamma_{ss}(f)\left|2U(f)\right|^2 = 4\gamma_{ss}(f)U(f) \qquad [3.20]$$

because the step function $U(f)$ is such that $\left|U(f)\right|^2 = U(f)$.

The autocorrelation of the baseband signal can be written as a function of that of the analytic signal, by applying its definition [3.5]:

$$\begin{aligned} R_{s_ls_l}(\tau) &= E\left[s_l(t)s_l^*(t-\tau)\right] \\ &= E\left[s_+(t)e^{-j2\pi f_ct}{s_+}^*(t-\tau)e^{j2\pi f_c(t-\tau)}\right] \\ &= E\left[s_+(t){s_+}^*(t-\tau)\right]e^{-j2\pi f_c\tau} \\ &= R_{s_+s_+}(\tau)e^{-j2\pi f_c\tau} \qquad [3.21] \end{aligned}$$

Consequently, the power spectrum density of the baseband signal is equal to:

$$\begin{aligned} \gamma_{s_ls_l}(f) &= \int_{-\infty}^{\infty} r_{s_ls_l}(\tau)e^{-j2\pi f\tau}d\tau \\ &= \int_{-\infty}^{\infty} r_{s_+s_+}(\tau)e^{-j2\pi(f+f_c)\tau}d\tau \\ &= \gamma_{s_+s_+}(f+f_c) \qquad [3.22] \end{aligned}$$

Equations [3.20] and [3.22] show that the power spectrum density of the complex baseband signal is equal to the power spectrum density of the real narrowband signal, limited to its values in the positive frequencies spectrum, and then shifted around frequency 0, and whose amplitude has been multiplied by 4.

Inversely, the power spectrum density of the real narrowband signal can be expressed as a function of that of its complex envelope as follows:

$$\gamma_{ss}(f) = \frac{1}{4}\left(\gamma_{s_l s_l}(f - f_c) + \gamma_{s_l s_l}(-f - f_c)\right) \qquad [3.23]$$

3.2.3.3. *Power*

The power of the complex envelope and the analytic signal is obtained as a function of that of the real narrowband signal, using equations [3.20] and [3.21]:

$$\begin{aligned} R_{s_l s_l}(0) &= 2R_{ss}(0) \\ R_{s_+ s_+}(0) &= 2R_{ss}(0) \end{aligned} \qquad [3.24]$$

Thus, passing to the complex envelope doubles the power, compared to that of the original narrowband signal.

3.2.3.4. *In-phase and quadrature components*

Two important properties of autocorrelation and intercorrelation of the in-phase component $s_c(t)$ and quadrature component $s_l(t)$ are obtained by computing the autocorrelation of the narrowband random process $s(t)$. It is equal to:

$$\begin{aligned} R_{ss}(\tau) = {} & E\left[(s_c(t)\cos(2\pi f_c t) - s_s(t)\sin(2\pi f_c t))\right. \\ & \left.(s_c(t-\tau)\cos(2\pi f_c(t-\tau)) - s_s(t-\tau)\sin(2\pi f_c(t-\tau)))\right] \\ = {} & \frac{1}{2}\left\{R_{s_c s_c}(\tau) - R_{s_s s_s}(\tau)\right\}\cos(2\pi f_c(2t-\tau)) \\ & -\frac{1}{2}\left\{E\left[s_c(t)s_s(t-\tau)\right] + E\left[s_s(t)s_c(t-\tau)\right]\right\}\sin(2\pi f_c(2t-\tau)) \\ & +\frac{1}{2}\left\{R_{s_c s_c}(\tau) + R_{s_s s_s}(\tau)\right\}\cos(2\pi f_c\tau) \\ & +\frac{1}{2}\left\{E\left[s_c(t)s_s(t-\tau)\right] - E\left[s_s(t)s_c(t-\tau)\right]\right\}\sin(2\pi f_c\tau) \end{aligned} \qquad [3.25]$$

Since $s(t)$ is a second-order stationary random process, its autocorrelation is independent of t. As a result, the following relations are deduced from

[3.28]:

$$R_{s_cs_c}(\tau) = R_{s_ss_s}(\tau) \qquad [3.26]$$

and, by setting $R_{s_cs_s}(\tau) = E\left[s_c(t)s_s(t-\tau)\right]$:

$$R_{s_cs_s}(\tau) = -R_{s_ss_c}(\tau) \qquad [3.27]$$

Finally, the autocorrelation of $s(t)$ can be written as:

$$R_{ss}(\tau) = R_{s_cs_c}(\tau)\cos(2\pi f_c\tau) - R_{s_ss_c}(\tau)\sin(2\pi f_c\tau) \qquad [3.28]$$

The autocorrelation of the equivalent baseband signal is expressed as a function of s_c and s_s:

$$\begin{aligned} R_{s_ls_l}(\tau) &= E\left[(s_c(t)+js_s(t))\left(s_c(t-\tau)-js_s(t\tau)\right)\right] \\ &= R_{s_cs_c}(\tau)+R_{s_ss_s}(\tau)+j\left\{E\left[s_s(t)s_c(t-\tau)\right]\right. \\ &\quad \left.-E\left[s_c(t)s_s(t-\tau)\right]\right\} \\ &= 2R_{s_cs_c}(\tau)+2jR_{s_ss_c}(\tau) \end{aligned} \qquad [3.29]$$

As powers $R_{s_ls_l}(0)$ and $R_{s_cs_c}(0)$ are real, equation [3.29] indicates that $R_{s_ss_c}(0) = 0$. Thus, the in-phase random variable $s_c(t)$ and the quadrature random variable $s_s(t)$ are independent for any value of t. Moreover, the power of both in-phase signal and quadrature signal is equal to half the power of the baseband signal:

$$\begin{aligned} R_{s_cs_c}(0) &= \frac{1}{2}R_{s_ls_l}(0) \\ R_{s_ss_s}(0) &= \frac{1}{2}R_{s_ls_l}(0) \end{aligned} \qquad [3.30]$$

Consequently, equation [3.24] shows that the power of both in-phase and quadrature signals is equal to the power of the original narrowband signal, $s(t)$.

In the particular case where the power spectrum density of $s(t)$ is symmetrical around the carrier frequency f_c, the power spectrum density of $s_l(t)$ is symmetrical around $f = 0$. Since the inverse Fourier transform of an even function is a real function, the autocorrelation function $r_{s_ls_l}(\tau)$ is real.

Equation [3.29] shows that in this case, $r_{s_s s_c}(\tau)$ is equal to 0, and this is valid for any τ. Thus, the in-phase random variable s_c and quadrature random variable s_s are uncorrelated. Moreover, in this case, the power spectrum density of the baseband signal is equal to the double of that of the in-phase and quadrature components:

$$\gamma_{s_l s_l}(f) = 2\gamma_{s_c s_c}(f)$$
$$\gamma_{s_l s_l}(f) = 2\gamma_{s_s s_s}(f) \qquad [3.31]$$

3.2.3.5. *Example: white noise*

The white noise is a random variable whose power spectrum density is constant on the whole spectrum. It does not have any equivalent baseband representation, since it is a wideband signal. In order to express its equivalent baseband characteristics, the white noise must be limited to a given bandwidth, B. In practice, the corresponding bandwidth is that of the signal over which the white noise will be added, if we consider the additive white Gaussian noise channel. Filtering the noise by a passband filter introduces some distortions on it; however, these distortions can be neglected.

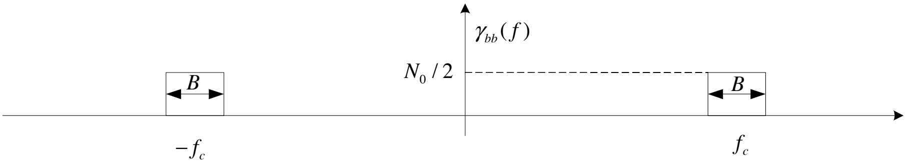

Figure 3.4. *Power spectrum density of passband white noise*

The power spectrum density of noise $b(t)$ is:

$$\gamma_{bb}(f) = \frac{N_0}{2} \qquad [3.32]$$

if $f \notin \left[f_c - \frac{B}{2}; f_c + \frac{B}{2}\right] \cup \left[-f_c - \frac{B}{2}; -f_c + \frac{B}{2}\right]$, and $\gamma_{bb}(f) = 0$ elsewhere. Its total power is consequently:

$$R_{bb}(0) = BN_0 \qquad [3.33]$$

The equivalent baseband noise is $b_l(t)$. Its power spectrum density is obtained by the following formula:

$$\gamma_{b_l b_l}(f) = 4\gamma_{bb}^{+}(f + f_c) \qquad [3.34]$$

$$= \begin{cases} 2N_0 \text{ si } & |f| \leq \frac{1}{2}B \\ 0 \text{ si } & |f| > \frac{1}{2}B \end{cases}$$

The autocorrelation function of the noise's complex envelope is deduced by applying the inverse Fourier transform:

$$R_{b_l b_l}(\tau) = 2BN_0 \frac{\sin(\pi B\tau)}{\pi B\tau} \qquad [3.35]$$

The power of the equivalent baseband complex noise is:

$$R_{b_l b_l}(0) = 2BN_0 \qquad [3.36]$$

When bandwidth B tends to infinity, the autocorrelation function tends to a Dirac function:

$$R_{b_l b_l}(\tau) \rightarrow 2N_0\delta(\tau) \qquad [3.37]$$

Moreover, since $\gamma_{b_l b_l}$ is symmetrical, the random processes of the in-phase and quadrature components of the noise are uncorrelated, $\gamma_{b_c b_s}(f) = 0$. From equation [3.31], they have the same power spectrum density, which is equal to:

$$\gamma_{b_c b_c}(f) = P_{b_s b_s}(f) = \frac{1}{2}\gamma_{b_l b_l}(f) = N_0 \text{ when } |f| \leq \frac{1}{2}B \qquad [3.38]$$

The white Gaussian noise consequently has an equivalent baseband representation which contains an in-phase component $b_c(t)$ and a quadrature component $b_s(t)$ which are both Gaussian, centered, with variance N_0 and bandwidth $B/2$ in the positive spectrum. They are independent and both components have the same power as the original signal:

$$R_{b_c b_c}(0) = R_{b_s b_s}(0) = BN_0 \qquad [3.39]$$

3.2.4. *Synchronous detection*

3.2.4.1. *Principle*

This section details how the receiver processes the signal in order to recover the transmitted signal, in case the useful information is completely contained in the complex envelope of the signal. This corresponds to linear digital modulations on sine waveforms, which will be studied in the next section. The receiver aims at obtaining both in-phase and quadrature components of the signal, in order to reconstitute the complex envelope $s_l(t) = s_c(t) + js_s(t)$.

We assume that the receiver is perfectly synchronized with the transmitter. Thus, there is no phase shift between the receiver and transmitter: we can then denote this case as synchronous detection. We also assume, to simplify, that the channel is ideal and that the additive white Gaussian noise term can be neglected. The receiver applies two parallel processes on the received signal (see Figure 3.5):

– multiplication of $s(t)$ by $2\cos(2\pi f_c t)$ followed by lowpass filtering;

– multiplication of $s(t)$ by $-2\sin(2\pi f_c t)$ followed by lowpass filtering.

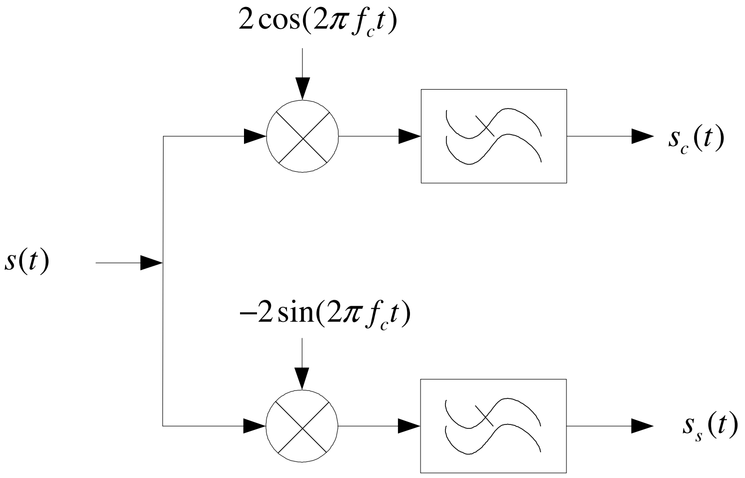

Figure 3.5. *Synchronous detection*

On the first branch, after having multiplied by $2\cos(2\pi f_c t)$, the signal becomes:

$$2s(t)\cos(2\pi f_c t) = s_c(t) + (s_c(t)\cos(4\pi f_c t) - s_s(t)\sin(4\pi f_c t)) \qquad [3.40]$$

Lowpass filtering removes the components over frequency carrier $2f_c$. Consequently, after filtering, the output of the first branch is equal to the in-phase signal $s_c(t)$.

Similarly, on the second branch, after multiplying by $-2\sin(2\pi f_c t)$, the signal becomes:

$$-2s(t)\sin(2\pi f_c t) = s_s(t) - (s_c(t)\cos(4\pi f_c t) + s_s(t)\sin(4\pi f_c t)) \qquad [3.41]$$

Lowpass filtering allows us to obtain $s_s(t)$.

If synchronous detection cannot be performed because the receiver has a phase shift compared to the transmitted signal, the detector multiplies the received signal by $2\cos(2\pi f_c t + \Psi)$, where Ψ is the phase shift. Then, after lowpass filtering, the signal obtained on the in-phase branch is:

$$s_c(t)\cos(\Psi) + s_s(t)\sin(\Psi) \qquad [3.42]$$

Thus, the in-phase signal at the receiver contains a part of the quadrature component, and vice versa. This generates cross-talk between both branches. Consequently, it is not possible to separate the in-phase and quadrature signals when perfect synchronization is not available.

The various techniques allowing the receiver to acquire phase synchronization with the transmitter (even if the channel modifies the transmitter's phase) will be studied in Chapter 4. In the remainder of this chapter, we will assume that synchronous detection can be performed, and also that the receiver's clock is perfectly synchronized with that of the transmitter.

3.2.4.2. *Application to the additive white Gaussian noise channel*

We consider a signal $s(t)$ on the additive white Gaussian noise described by Figure 3.4. After synchronous detection, the receiver signal is equal to

$r_c(t) = s_c(t) + b_c(t)$ on the in-phase branch, and $r_s(t) = s_s(t) + b_s(t)$ on the quadrature branch.

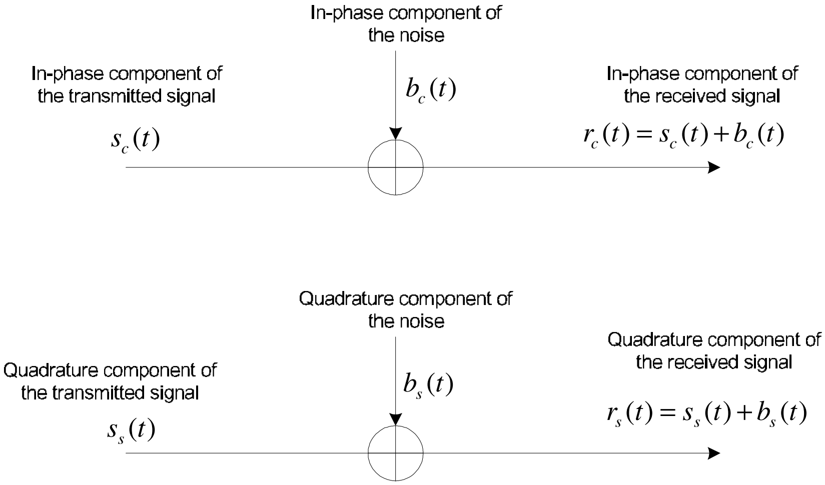
In-phase component of the noise
$b_c(t)$
In-phase component of the transmitted signal
$s_c(t)$
In-phase component of the received signal
$r_c(t) = s_c(t) + b_c(t)$
Quadrature component of the noise
$b_s(t)$
Quadrature component of the transmitted signal
$s_s(t)$
Quadrature component of the received signal
$r_s(t) = s_s(t) + b_s(t)$

Figure 3.6. *Passband additive white Gaussian noise channel*

The in-phase and quadrature components are independent and have the same power, which is equal to that of the original narrowband signal. Consequently, the in-phase and quadrature components can be processed separately and independently, as if they were two signals filtered by an additive white Gaussian noise channel (see Figure 3.6). Since the power of the in-phase and quadrature components of the complex envelope is the same as the power of the original signal, the signal-to-noise ratio on each component is equal to the signal-to-noise ratio of the narrowband signal. Thus, any real narrowband signal can be equivalently studied as two baseband signals on the additive white Gaussian noise channel.

3.3. Linear digital modulations on sine waveforms

3.3.1. *Main characteristics of linear digital modulations*

3.3.1.1. *Introduction*

The line codes that were introduced in Chapter 2 are used to modulate symbols for baseband transmissions. They consist of mapping symbols onto real analog waveforms. Moreover, the transmitted signals then generally have an infinite bandwidth. In this chapter, we are interested in transmissions of symbols in narrowband channels, around a carrier frequency f_c.

This signal, transmitted with symbol-period T, is denoted as $s(t)$. In the time interval $t \in [kT, (k+1)T[$ with k an integer, the digital signal carries the information corresponding to one symbol. Then, the signal can be decomposed in the following way:

$$s(t) = \sum_{k=-\infty}^{+\infty} s_k(t - kT) \qquad [3.43]$$

The elementary signal carrying the k^{th} symbol $s_k(t)$ is written as:

$$s_k(t) = |c_k| \cos(2\pi f_{c,k} t + \varphi_k) g(t - kT)$$

with: $|c_k|$ the signal's amplitude, $f_{c,k}$ its carrier frequency, φ_k its phase and g its transmit filter (also called waveform or shaping filter).

One of the following three parameters can carry the information on which symbol is transmitted: its amplitude, frequency or phase. Modulating the signal consists of varying one or several of these parameters. The information on the symbol transmitted in the symbol-period is then carried by the value of the modified parameter(s). The considered modulations contain M symbols. They are called M-order modulations. A modulation state is associated with a set of n bits, called symbol. Thus, the number of bits per symbol is $n = \log_2(M)$.

Two types of digital modulations over sine waveforms exist: linear and nonlinear modulations.

When the frequency is constant, $f_{c,k} = f_c$ whatever k, it does not contain any information on the symbols. Consequently, the information on the symbols may be contained in the amplitude of the signal, its phase or a combination of both. The equivalent baseband signal of $s(t)$ is:

$$s_l(t) = \sum_{k=-\infty}^{+\infty} s_{k,l}(t) = \sum_{k=-\infty}^{+\infty} |c_k|\, e^{j\varphi_k} g(t - kT) \qquad [3.44]$$

A modulation is linear if the information is equivalently contained in its complex envelope.

On the contrary, nonlinear modulations depend on the carrier frequency. Section 3.4 will describe in detail these modulations, called FSK.

We are first interested in linear modulations. They can be of three types: modulations that modify the amplitude of the signal (called ASK), modulations that modify its phase (PSK) and those that modify both amplitude and phase (QAM).

In this section, after introducing some useful notations and definitions, we detail how regular linear modulations are generated, and we provide an analysis of their performances.

3.3.1.2. *Notations*

The complex envelope of the signal is expressed as follows:

$$x(t) = \sum_{k=-\infty}^{+\infty} c_k g(t-kT) = \sum_{k=-\infty}^{+\infty} a_k g(t-kT) + j \sum_{k=-\infty}^{+\infty} b_k g(t-kT) \tag{3.45}$$

$g(t-kT)$ is the transmit filter. When it is used with its matched filter at reception, the serial concatenation of both filters must verify the Nyquist criterion in the considered bandwidth B, which is finite. In most of the cases, filter g is consequently chosen as a root raised cosine filter. $c_k = a_k + jb_k$ is a complex number, a_k and b_k are reals.

Once it is set on its carrier frequency, the signal becomes:

$$\begin{aligned} s(t) &= \Re\left[\sum_{k=-\infty}^{+\infty} c_k g(t-kT) e^{j(2\pi f_c t)}\right] \\ &= \sum_{k=-\infty}^{+\infty} a_k g(t-kT)\cos(2\pi f_c t) - \sum_{k=-\infty}^{+\infty} b_k g(t-kT)\sin(2\pi f_c t) \end{aligned} \tag{3.46}$$

$x(t)$ is the equivalent baseband signal of $s(t)$: $x(t) = s_l(t)$.

The following notations are used for the in-phase and quadrature components: thus, $\sum_{k=-\infty}^{+\infty} a_k g(t - kT)$ modulates the in-phase carrier $\cos(2\pi f_c t)$ in amplitude, and $\sum_{k=-\infty}^{+\infty} b_k g(t - kT)$ modulates the quadrature carrier $\sin(2\pi f_c t)$ in amplitude.

By setting $s(t) = \sum_{k=-\infty}^{+\infty} s_k(t)$, the elementary modulated signal for symbol k becomes:

$$s_k(t) = a_k g(t - kT) \cos(2\pi f_c t) - b_k g(t - kT) \sin(2\pi f_c t) \qquad [3.47]$$

It is often denoted as $s_k(t) = s_{k,I}(t) + s_{k,Q}(t)$, where $= s_{k,I}(t)$ is the in-phase signal (denoted I for in-band) and $s_{k,Q}(t)$ is the quadrature signal (denoted by Q).

It is represented in Figure 3.7.

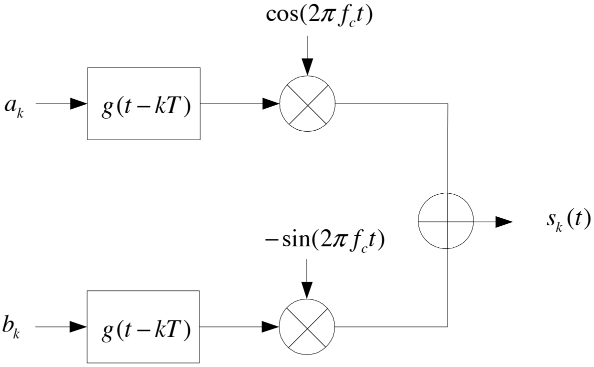

Figure 3.7. *Modulator: in-phase and quadrature components*

The performances of the modulations are studied by passing on the equivalent baseband signal and studying the transmission of symbols on the additive white Gaussian noise channel. Thus, we denote by E_s the average energy per symbol of the complex envelope of the signals after filtering and sampling, $c_k = a_k + jb_k$. The white Gaussian noise added in baseband on these symbols has variance $\sigma^2 = N_0/2$ per dimension. The white noise is always complex. The transmitted symbols are either real or complex, depending on the chosen modulation.

3.3.1.3. *Constellation*

The constellation is the representation in the complex plane of all points $c_k = a_k + jb_k$ associated with the modulated symbols.

This representation allows us to determine the distance between all symbols, the adjacency relations between symbols and so forth. An example is given in Figure 3.8.

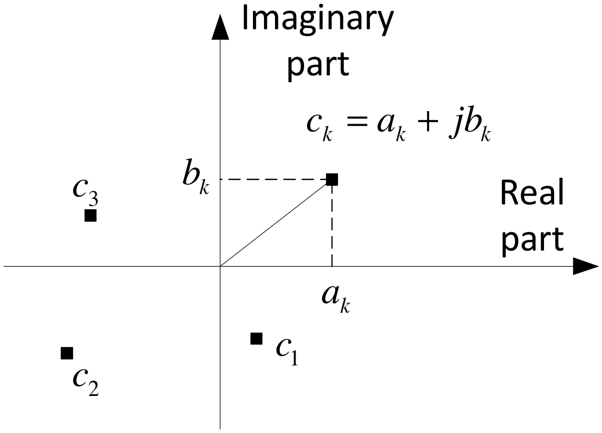

Figure 3.8. *Example of constellation*

The most important characteristic of a constellation is its minimum distance, d_{min}. It is equal to the minimum Euclidean distance between any two points of the constellation.

$$d_{\text{min}} = \min_{k \neq j} d_{kj} \text{ with } {d_{kj}}^2 = |c_k - c_j|^2 \qquad [3.48]$$

3.3.2. *Parameters of an M-symbols modulation*

An M-symbols linear modulation with symbol-period T is characterized by several parameters.

The energy per symbol c_k is equal to $E_{s,k} = |c_k|^2$. It is expressed in Joules. By averaging over all symbols, we obtain the constellation's energy, in Joules per symbol:

$$E_s = \frac{1}{M}\sum_{k=1}^{M}|c_k|^2 \qquad [3.49]$$

The constellation's energy can also be expressed in Joules per bit, by dividing by the number of bits per symbol:

$$E_b = \frac{E_s}{\log_2(M)} \qquad [3.50]$$

The symbol rate is equal to the number of transmitted symbols per second. It is expressed in Bauds.

$$D_s = \frac{1}{T} \qquad [3.51]$$

The binary transmission period is deduced from the symbol-period:

$$T_b = \frac{T}{\log_2(M)} \qquad [3.52]$$

Consequently, the rate in bits per second is equal to:

$$D_b = \frac{1}{T_b} = \log_2(M)D_s \qquad [3.53]$$

The power of the constellation is, by definition, equal to the average energy required to transmit a symbol, divided by the symbol-period:

$$P_s = \frac{E_s}{T} = E_sD_s \qquad [3.54]$$

It can also be written as a function of the binary energy and rate:

$$P_s = \frac{E_b}{T_b} = E_bD_b \qquad [3.55]$$

The power is expressed in Joules/second or Watts.

3.3.2.1. *Pairwise error probability*

In this section, we provide a reminder on the error symbol probability P_e depending on the signal-to-noise ratio $\frac{E_b}{N_0}$ for the additive white Gaussian noise channel, with a binary modulation. The transmitted symbols are real, with amplitude A or $-A$, which, respectively, represent bits 1 and 0, and they have equal probability of being transmitted. The equivalent baseband received signal sampled at the symbol-period T is:

$$y_k = a_k + n(kT) \qquad [3.56]$$

where $n(kT)$ is the equivalent baseband noise, filtered by the received filter and sampled. We assume that the receive filter did not modify the statistical properties of the noise.

It has been shown in section 2.4.4 that detection by a threshold is optimum if the chosen threshold is $S = 0$: if $y_k < 0$, then $\hat{x}_k = A$, else, $\hat{x}_k = -A$.

Thus, the BER, which is equal to the symbol error probability since each symbol represents one bit, is equal to:

$$P_e = \frac{1}{2}\text{erfc}\left(\sqrt{\frac{E_b}{N_0}}\right) \qquad [3.57]$$

The error probability can also be written as a function of the distance between the constellation's symbols. In the case of a binary modulation, it is equal to $d_{\text{min}} = 2\sqrt{E_b} = 2A$. Thus:

$$P_e = \frac{1}{2}\text{erfc}\left(\frac{d_{\text{min}}}{2\sqrt{N_0}}\right) \qquad [3.58]$$

This error probability is called the pairwise error probability. It will be used in the whole chapter to compute the performances of non-binary modulations.

3.3.3. *Amplitude shift keying*

3.3.3.1. *Main principle*

ASK modulations are linear modulations that only modify the amplitude of the signal. They are real modulations, transforming a real signal into another real signal. Consequently, modulation is only performed on the in-phase component of the signal $\cos(2\pi f_c t)$:

$$s(t) = \sum_{k=-\infty}^{+\infty} a_k g(t - kT) \cos(2\pi f_c t) \qquad [3.59]$$

and $b_k = 0$ for any k. Thus, the equivalent baseband signal is real. The $M = 2^n$ symbols set is the following:

$$\mathcal{C} = \{-(M-1), -(M-3), ..., -1, 1, ..., (M-3), (M-1)\}A \qquad [3.60]$$

During time interval $[kT, (k+1)T[$, the transmitted symbol a_k takes a value in $\mathcal{C}$ which corresponds to a different group of bits.

The symbols set is always centered around 0, so as to minimize the energy of the constellation. Two examples of ASK modulations of orders 4 and 8 are given in Figures 3.9 and 3.10.

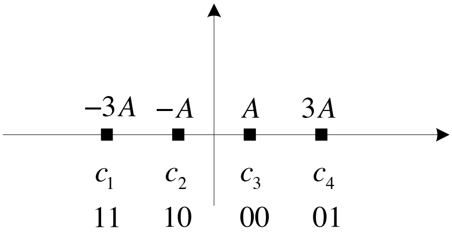

Figure 3.9. *4-ASK modulation*

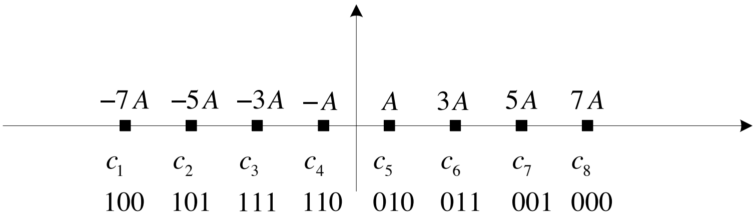

Figure 3.10. *8-ASK modulation*

The code associated with the modulation characterizes how groups of bits are mapped onto the symbols constellation. The most usual encoder for linear modulations is the Gray encoder. It consists of associating two symbols with adjacent amplitudes with two groups of n bits that only vary of one bit. An example of Gray encoding for the 8-ASK modulation (corresponding to 3 bits per symbol) is given in Table 3.1.

Message	Corresponding symbol
100	$-7A$
101	$-5A$
111	$-3A$
110	$-A$
010	A
011	$3A$
001	$5A$
000	$7A$

Table 3.1. *Gray encoding for 8-ASK modulations*

3.3.3.2. *Demodulation*

We consider an M-ASK modulation with a transmit filter that, together with its matched receive filter, $g_r(t)$, verifies Nyquist criterion. Moreover, we assume that the equivalent filter verifies $(g * g_r)(0) = f(0) = 1$. The channel adds a white Gaussian noise $b(t)$ in the signal's bandwidth, around f_c. Demodulation consists of recovering the transmitted symbols at the receiver. Its different steps are represented in Figure 3.11.

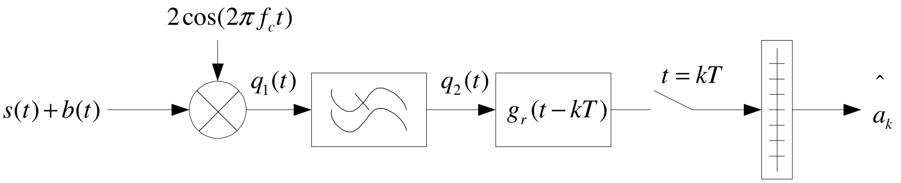

Figure 3.11. *Demodulation of ASK modulations*

The receiver first multiplies the received signal by a global oscillator, $2\cos(2\pi f_c t)$. The receiver is assumed synchronized both in frequency and phase with the transmitter. Thus, the signal after this multiplication, $q_1(t)$, is equal to:

$$\begin{aligned} q_1(t) &= \sum_{k=-\infty}^{+\infty} a_k g(t-kT)\cos(2\pi f_c t).2\cos(2\pi f_c t) + 2b(t)\cos(2\pi f_c t) \\ &= \sum_{k=-\infty}^{+\infty} a_k g(t-kT)\left(1+\cos(4\pi f_c t)\right) + 2b(t)\cos(2\pi f_c t) \qquad [3.61] \end{aligned}$$

A lowpass filter is then applied on signal $q_1(t)$. It removes the elements varying at frequency $2f_c$. The signal after lowpass filtering is denoted as $q_2(t)$:

$$q_2(t) = \sum_{k=-\infty}^{+\infty} a_k g(t-kT) + \widetilde{b}(t) \qquad [3.62]$$

The matched receive filter, $g_r(t)$, is then applied on $q_2(t)$:

$$q_3(t) = \sum_{k=-\infty}^{+\infty} a_k g(t-kT) * g_r(t-kT) + n(t) \qquad [3.63]$$

where $n(t)$ is the noise after lowpass filtering and application of the matched filter. The receiver then recovers the symbol-period T. It samples $q_3(t)$ at period T. Since the equivalent filter $g(t-kT) * g_r(t-kT)$ verifies the Nyquist criterion, the symbol sampled at time kT is equal to:

$$y_k = a_k f(0) + n(kT) = a_k + n(kT) \qquad [3.64]$$

The sampled noise is white and Gaussian since, as we have already seen, passing from the narrowband noise to its equivalent baseband noise does not change these two characteristics. Moreover, we assume that the receive filter does not have any influence on the whiteness of the noise. Both these assumptions are kept in the remainder of the chapter.

Finally, in order to determine the value of a_k, a threshold detector is used. It contains $(M-1)$ thresholds, all located at equivalent distance between two symbols of the constellation $\mathcal{C} = \{-(M-1), -(M-3), ..., -1, 1, ...,$

$(M-3),(M-1)\}A$. Consequently, the threshold values are:

$$\mathcal{S}=\{-(M-2),-(M-4),...,0,...,(M-4),(M-2)\}A \qquad [3.65]$$

For instance, if the symbol at the input of the threshold detector r_k is in interval $]-(M-4)A,-(M-6)A]$, then the transmitted symbol is estimated as the central value of this interval: $\hat{a}_k=-(M-5)A$. If $r_k \leq -(M-2)A$, the estimated symbol is $\hat{a}_k=-(M-1)A$.

3.3.3.3. *Performances of an M-ASK modulation*

We consider an M-ASK modulation. To simplify, we assume that filtering neither changed the samples' amplitude nor the noise power. The transmitted symbols are $a_m=(2m-1-M)A$ for $m=1,2,...,M$. The minimum distance between two symbols is $d_{\text{min}}=2A$.

The average energy per symbol[1] is:

$$\begin{aligned}E_s &= \frac{1}{M}\sum_{m=1}^{M}\left((2m-1-M)^2A^2\right)\\ &= \frac{A^2}{M}\left(\frac{1}{3}M(M^2-1)\right)=\frac{1}{3}(M^2-1)A^2 \qquad [3.66]\end{aligned}$$

After demodulation, the threshold detector compares $y_k=a_k+n(kT)$ with $M-1$ thresholds located at equal distance from two adjacent symbols of the constellation. As a result, the error probability on one symbol is equal to the probability that the noise exceeds half of the distance between two adjacent symbols of the constellation.

Let us consider an internal symbol of the constellation, for instance c_5 with amplitude A in the 8-ASK modulation represented in Figure 3.12. The estimated symbol $\hat{a_k}$ is equal to c_5 if the received signal, r_k, is located in interval $]0,2A]$. Thus, an error takes place on the estimated symbol if r_k is not

[1] Please note that this formula is the same as the one obtained for an M-ary NRZ signal in section 2.2.10. Indeed, M-ary NRZ line codes are equivalent to M-ASK modulations, once they are set back in baseband. Consequently, any computation performed in this section is also valid for M-ary NRZ codes.

in this interval. The error probability, knowing that c_5 has been transmitted, is consequently equal to:

$$\begin{aligned} Pr\left(\hat{a_k} \neq c_5 | a_k = c_5\right) &= \int_{-\infty}^{0} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y-A)^2}{2\sigma^2}} dy + \int_{2A}^{\infty} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y-A)^2}{2\sigma^2}} dy \\ &= -\frac{1}{\sqrt{\pi}} \int_{-\infty}^{-\frac{A}{\sqrt{2}\sigma}} e^{-x^2} dx + \frac{1}{\sqrt{\pi}} \int_{\frac{A}{\sqrt{2}\sigma}}^{\infty} e^{-x^2} dx \\ &= \frac{1}{\sqrt{\pi}} \int_{\frac{A}{\sqrt{2}\sigma}}^{\infty} e^{-x^2} dx + \frac{1}{\sqrt{\pi}} \int_{\frac{A}{\sqrt{2}\sigma}}^{\infty} e^{-x^2} dx \\ &= \text{erfc}\left(\frac{A}{\sqrt{2}\sigma}\right) \end{aligned} \quad [3.67]$$

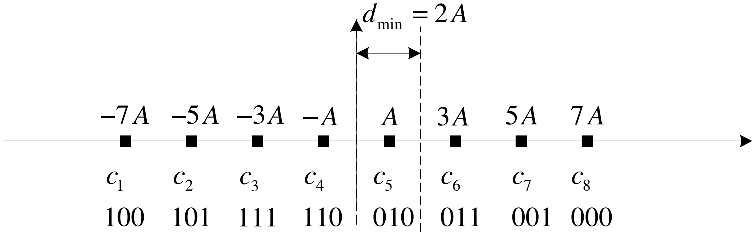

Figure 3.12. *Minimum distance in an ASK modulation*

Thus, the error probability knowing that c_5 has been transmitted is equal to twice the pairwise error probability. This result applies to the $(M-2)$ symbols that are inside the constellation.

The error probability for the symbols that are outside the constellation is lower since they only have one adjacent symbol. Let us consider symbol c_8, with amplitude $7A$. The estimated symbol $\hat{a_k}$ is equal to c_8 if the received signal, r_k, is higher than $6A$. The error probability knowing that c_8 has been transmitted is consequently equal to:

$$\begin{aligned} Pr\left(\hat{a_k} \neq c_8 | a_k = c_8\right) &= \int_{-\infty}^{6A} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y-7A)^2}{2\sigma^2}} dy \\ &= \frac{1}{\sqrt{\pi}} \int_{-\infty}^{\frac{A}{\sqrt{2}\sigma}} e^{-x^2} dx \\ &= \frac{1}{2}\text{erfc}\left(\frac{A}{\sqrt{2}\sigma}\right) \end{aligned} \qquad [3.68]$$

For the symbols that are outside of the constellation, the error probability is equal to the pairwise error probability.

Finally, the symbol error probability, averaged over all M symbols, is equal to:

$$\begin{aligned} P_e &= \frac{1}{M}\left((M-2) + \frac{1}{2} + \frac{1}{2}\right) \text{erfc}\left(\frac{A}{\sqrt{2}\sigma}\right) \\ &= \frac{(M-1)}{M}\text{erfc}\left(\frac{A}{\sqrt{2}\sigma}\right) \end{aligned} \qquad [3.69]$$

It may also be written depending on the energy per symbol E_s, using equation [3.66]:

$$P_e = \frac{(M-1)}{M}\text{erfc}\left(\sqrt{\frac{3E_s}{(M^2-1)N_0}}\right) \qquad [3.70]$$

An illustration of the performances of ASK modulations is given in Figure 3.13.

It is more relevant to express the symbol error probability as a function of E_b/N_0 ratio, with the constellation's energy per bit. This indeed allows us to compare the performances of modulations of various orders. The energy per symbol is related to the energy per bit by $E_s = \log_2(M)E_b$. Thus, the symbol error probability is determined depending on E_b/N_0 by the following equation:

$$P_e = \frac{(M-1)}{M}\text{erfc}\left(\sqrt{\frac{3\log_2(M)}{(M^2-1)}\frac{E_b}{N_0}}\right) \qquad [3.71]$$

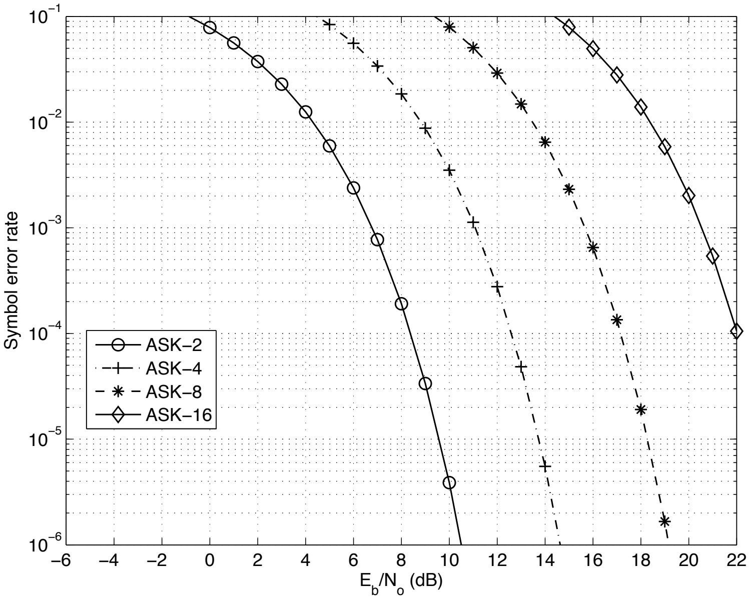

Figure 3.13. *Performances of ASK modulations*

It may also be useful to consider the error probability per bit, also called BER. In general, the BER cannot be directly deduced from the symbol error probability. Nevertheless, if Gray encoding is used, the following equation holds at high $\frac{E_b}{N_0}$:

$$BER \approx \frac{P_e}{\log_2(M)} \qquad [3.72]$$

This equation comes from the fact that with Gray encoding, two adjacent symbols only possess one different bit. At high $\frac{E_b}{N_0}$, most errors between two symbols take place between adjacent symbols and then correspond to an error on one bit. The BER for several ASK modulations, computed with formula [3.72], is represented in Figure 3.14.

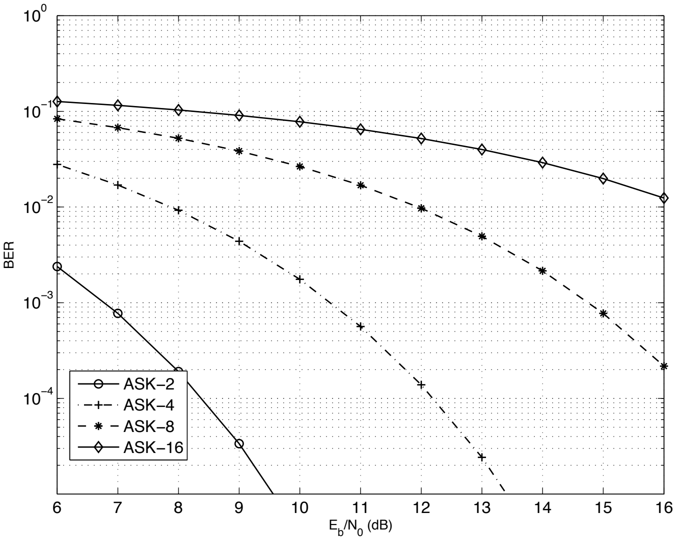

Figure 3.14. *Bit error rate of ASK modulations*

3.3.4. *Phase shift keying*

3.3.4.1. *Main principle*

PSK modulations only modify the signal's phase.

The modulated signal is:

$$s(t) = \Re\left[\sum_{k=-\infty}^{+\infty} c_k g(t-kT)e^{j(2\pi f_c t)}\right] \qquad [3.73]$$

The complex symbols c_k are distributed over a circle of radius A:

$$c_k = a_k + jb_k = Ae^{j\varphi_k} \qquad [3.74]$$

Thus, the in-phase and quadrature components are equal to:

$$\begin{cases} a_k = A\cos(\varphi_k) \\ b_k = A\sin(\varphi_k) \end{cases} \qquad [3.75]$$

The signal transmitted in interval $[kT,(k+1)T[$ is:

$$\begin{aligned} s(t) &= \Re\left[g(t-kT)Ae^{j(2\pi f_c t+\varphi_k)}\right] \\ &= Ag(t-kT)\cos(2\pi f_c t+\varphi_k) \\ &= Ag(t-kT)\cos(\varphi_k)\cos(2\pi f_c t) - Ag(t-kT)\sin(\varphi_k)\sin(2\pi f_c t) \end{aligned} \quad [3.76]$$

Equation [3.76] indicates that the in-phase carrier $\cos(2\pi f_c t)$ is amplitude-modulated by $Ag(t-kT)\cos(\varphi_k)$ and that the quadrature carrier $\sin(2\pi f_c t)$ is amplitude-modulated by $Ag(t-kT)\sin(\varphi_k)$.

All symbols are equally distributed over a circle of radius A. Consequently, each symbol is represented by a phase φ_k which belongs to the following set:

$$\Psi = \left\{0, \frac{2\pi}{M}, ..., \frac{2\pi(M-1)}{M}\right\} \quad [3.77]$$

An example of PSK modulation with four symbols is given in Figure 3.15. In this simple case, the symbols are: $\{A, jA, -A, -jA\}$. 2-PSK modulation is equal to 2-ASK, since they are both binary transmissions. 8-PSK is represented in Figure 3.16.

NOTE.– We can notice that in most practical systems, it is chosen not to transmit symbols with phase $\varphi_k = 0$. In order to obtain this, a rotation of angle π/M must be applied on the whole constellation. Of course, the demodulation threshold must adequately be modified.

3.3.4.2. *Demodulation*

PSK demodulation is performed with a synchronous detector (see Figure 3.17). The synchronous detector aims at separating in-phase and quadrature components. It is required for PSK modulations of order higher than 2, since the modulated signal is then complex. The synchronous detector estimates $A\cos(\varphi_k)$ on the in-phase branch and $A\sin(\varphi_k)$ on the quadrature branch. It then deduces the estimated phase, $\hat{\varphi}_k$. It is equal to the closest phase among the set of possibles phases: $\Psi = \left\{0, \frac{2\pi}{M}, ..., \frac{2\pi(M-1)}{M}\right\}$. Thus, the estimated symbol is equal to the closest symbol of the constellation. The threshold detector then uses thresholds that are located at equal distance between two phases of the constellations' symbols:

$$S = \left\{ \frac{\pi}{M}, \frac{3\pi}{M}, ..., \frac{(2M-1)\pi}{M} \right\} \qquad [3.78]$$

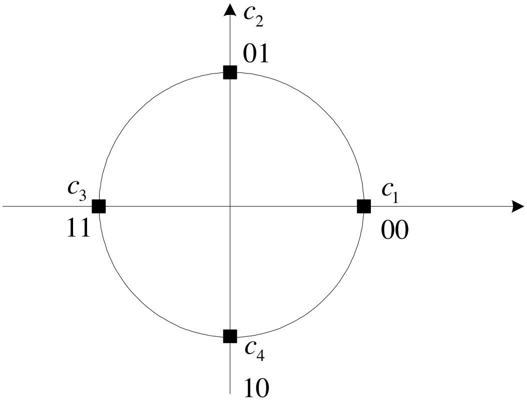

Figure 3.15. *4-PSK modulation*

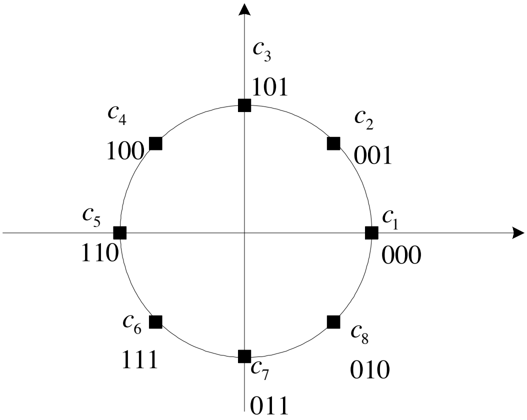

Figure 3.16. *8-PSK modulation*

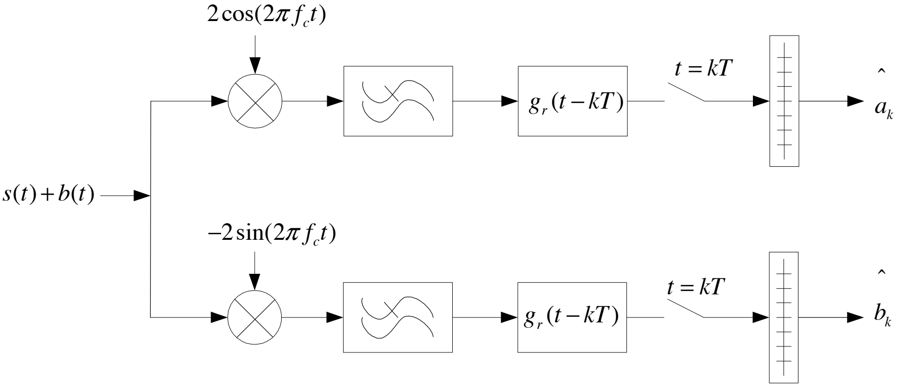

Figure 3.17. *Synchronous detection for PSK modulations*

3.3.4.3. *Performances of M-PSK modulations*

The transmitted symbols can be represented as vectors as follows, for $m = 1, 2, ..., M$:

$$c_m = \left[A\cos\left(\frac{2\pi}{M}(m-1)\right), A\sin\left(\frac{2\pi}{M}(m-1)\right)\right]$$

As shown in Figure 3.18, the minimum distance between two symbols is $d_{\min} = 2A\sin\left(\frac{\pi}{M}\right)$.

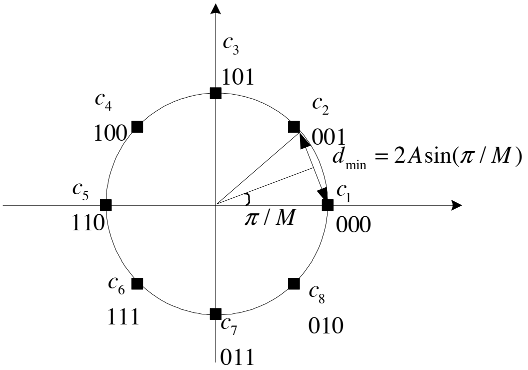

Figure 3.18. *Minimum distance of an 8-PSK modulation*

The average energy per symbol is equal to:

$$E_s = \frac{1}{M}\sum_{m=1}^{M} A^2 = A^2 \qquad [3.79]$$

since all symbols have the same modulus.

After demodulation, the threshold detector compares $y_k = c_k + n(kT)$ with $M-1$ phase-dependent thresholds.

The pairwise error probability is computed depending on the minimum distance:

$$\begin{aligned} P_{e,\text{pair}} &= \frac{1}{2}\text{erfc}\left(\frac{d_{\min}}{2\sqrt{N_0}}\right) \\ &= \frac{1}{2}\text{erfc}\left(\sqrt{\frac{\left(2A\sin\left(\frac{\pi}{M}\right)\right)^2}{4N_0}}\right) \\ &= \frac{1}{2}\text{erfc}\left(\sqrt{\frac{E_s\left(\sin\left(\frac{\pi}{M}\right)\right)^2}{N_0}}\right) \\ &= \frac{1}{2}\text{erfc}\left(\sqrt{\frac{E_b}{N_0}\log_2(M)\left(\sin\left(\frac{\pi}{M}\right)\right)^2}\right) \end{aligned} \qquad [3.80]$$

Contrary to the M-ASK modulations, the symbol error probability cannot be directly deduced from the pairwise error probability. Nevertheless, an upper bound to the symbol error probability is obtained by considering the fact that each symbol of an M-PSK modulation possesses two adjacent symbols located at the same distance $d_{\min}$. The error probability is consequently less than or equal to twice the pairwise error probability. Moreover, it can be shown that this upper bound is close to the real value at high $\frac{E_b}{N_0}$:

$$P_e \approx \text{erfc}\left(\sqrt{\frac{E_b}{N_0}\log_2(M)\left(\sin\left(\frac{\pi}{M}\right)\right)^2}\right) \qquad [3.81]$$

In order to illustrate the differences between this value and the effective symbol error probability, we consider an 8-PSK modulation, represented in

Figure 3.19. We assume that symbols are denoted by c_1 to c_8, starting with the symbol corresponding to 000 bits, and moving counterclockwise. Then, if c_1 was transmitted, a detection error takes place if the angle of the received signal y_k does not belong to interval $]-\frac{\pi}{8},\frac{\pi}{8}]$. The pairwise error probability between c_1 and c_2 corresponds to the error probability, knowing that c_1 was transmitted, if the angle of the received signal is located between $[\frac{\pi}{8}]$ and $[\frac{9\pi}{8}]$. Similarly, the pairwise error probability between c_1 and c_8 corresponds to the error probability, knowing that c_1 was transmitted, if the angle of the received signal is located between $[-\frac{\pi}{8}]$ and $[\frac{7\pi}{8}]$. As a result, the sum of both pairwise error probabilities implies that the error probability for the received signals with angles located in interval $]\frac{7\pi}{8},\frac{9\pi}{8}]$ is counted twice.

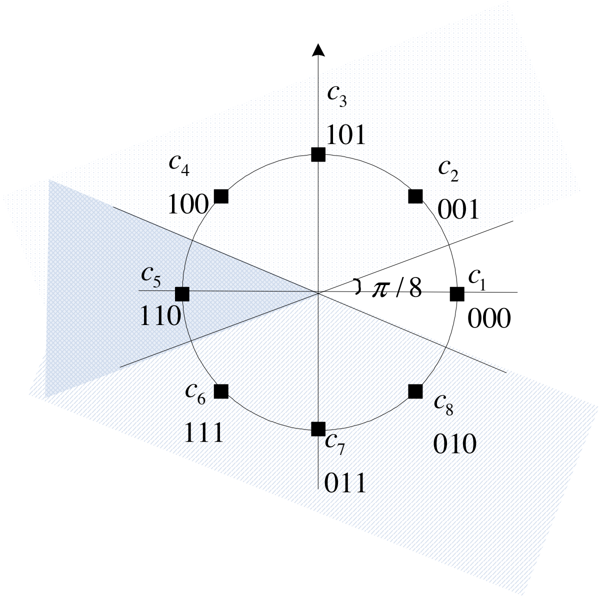

Figure 3.19. *Approximation of the symbol error probability by pairwise error probability for 8-PSK modulation*

Nevertheless, the actual symbol error probability is always higher than the error probability when only one of the two adjacent symbols is taken into account. Finally, it is bounded by:

$$\frac{1}{2}\mathrm{erfc}\left(\sqrt{\frac{E_b}{N_0}\log_2(M)\left(\sin\left(\frac{\pi}{M}\right)\right)^2}\right) \leq P_e$$

$$P_e \leq \mathrm{erfc}\left(\sqrt{\frac{E_b}{N_0}\log_2(M)\left(\sin\left(\frac{\pi}{M}\right)\right)^2}\right) \qquad [3.82]$$

Yet, the upper bound is closer to the effective symbol error probability than the lower bound, which explains why the upper bound is used in practice.

When Gray encoding is used, the BER can be approximated at high $\frac{E_b}{N_0}$ by formula [3.72].

The performances of PSK modulations (using equation [3.81]) are illustrated in Figure 3.20.

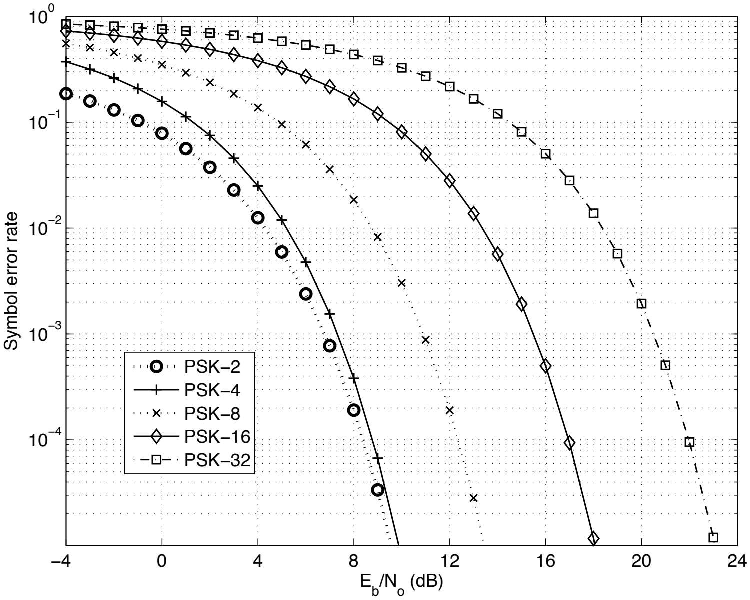

Figure 3.20. *Performances of PSK modulations*

3.3.5. *Quadrature amplitude modulations*

QAM modulations modify both amplitude and phase of the signal.

The constellation's points are distributed in the complex plane with variable amplitudes.

The modulated signal is still written as:

$$s(t) = \sum_{k=-\infty}^{+\infty} a_k g(t - kT) \cos(2\pi f_c t) - \sum_{k=-\infty}^{+\infty} b_k g(t - kT) \sin(2\pi f_c t) \quad [3.83]$$

but a_k and b_k may take values in two different sets, $\mathcal{A} = \{A_1, A_2, ..., A_i\}$ and $\mathcal{B} = \{B_1, B_2, ..., B_l\}$.

A QAM modulation is regular if both sets are equal and if the number of states $M = \left(\sqrt{M}\right)^2$ is a perfect square, with M a power of 2. Then:

$$\mathcal{A} = \mathcal{B} = \{A_1, A_2, ..., A_{\sqrt{M}}\}$$

We will use this assumption in the remainder of this section. In practice, regular modulations are used in the majority of cases. The usual orders of the constellations are 4, 16 or 64.

Two examples of 4-QAM and 16-QAM are given in Figures 3.21 and 3.22. 4-QAM modulation is equivalent to a 4-PSK modulation with the same energy, after phase rotation of $\pi/4$. They are most commonly called quadrature phase shift keying (QPSK).

Gray encoding of QAM can be simply obtained with two $\sqrt{M}$-ASK. Gray encoding is applied on both ASK, on the real and imaginary parts, respectively. It thus concerns $1/2 \log_2(M)$ bits. Then, the code of each M-QAM symbol is obtained by combining both $\sqrt{M}$-ASK codes: the first $1/2 \log_2(M)$ bits are those of the imaginary part's code, and the $1/2 \log_2(M)$ last bits are those of the real part's code.

An example of modulation where M is not a perfect square is given for $M = 32$ in Figure 3.23. The constellation's points do not belong to the set

$\mathcal{A}\times\mathcal{B}$, but to a subset of it, since some combinations of real and imaginary parts are not allowed. 32-QAM modulation is called a cross-constellation. Other constellations of higher order, such as for instance 128-QAM, follow the same scheme.

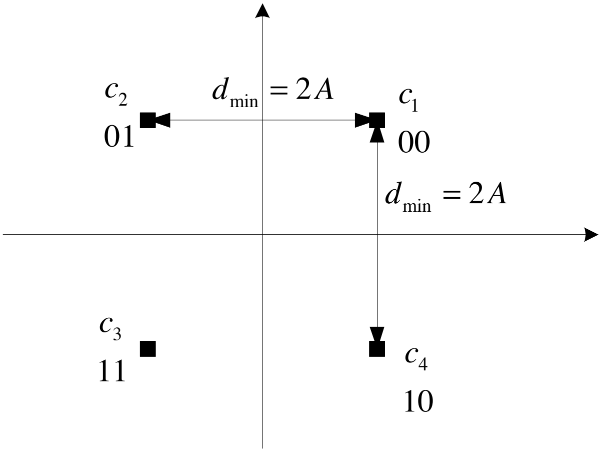

Figure 3.21. *QPSK modulation*

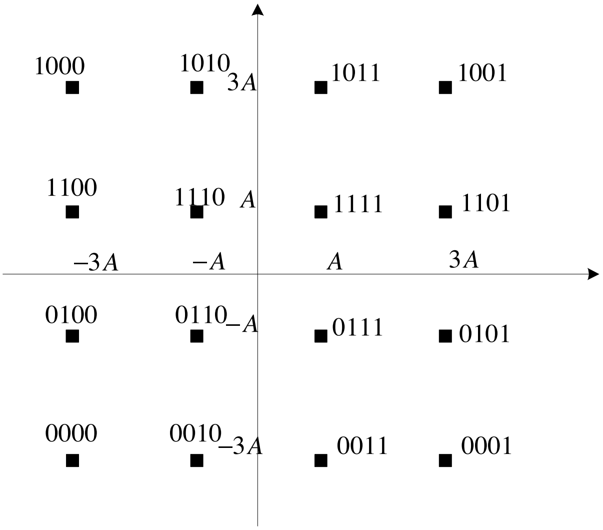

Figure 3.22. *16-QAM modulation*

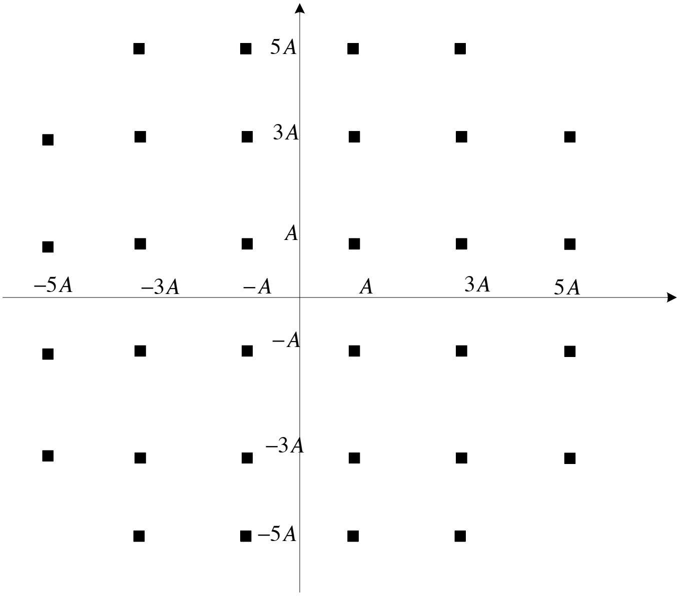

Figure 3.23. *32-QAM modulation*

3.3.5.1. *Demodulation*

As for PSK, demodulation is performed with a synchronous detector, since both in-phase and quadrature components of the signal must be considered in order to determine the transmitted complex symbol. The same principle is used (see Figure 3.17). On each branch, a threshold detector estimates the values of a_k and b_k. The complex symbol $\widehat{c}_k = \widehat{a}_k + j\widehat{b}_k$ is then deduced from these estimates.

3.3.5.2. *Performances of M-QAM*

We consider a regular QAM where M is a perfect square. a_k and b_k both take their value in the same set, $\mathcal{A} = \{A_1, A_2, ..., A_{\sqrt{M}}\}$ with $A_m = (2m-1-\sqrt{M})A$ for $m = 1, 2, ..., \sqrt{M}$. The minimum distance between two symbols is then: $d_{\text{min}} = 2A$.

A regular M-QAM is equivalent to two $\sqrt{M}$-ASK in quadrature. Consequently, the energy per symbol of the M-QAM is the double of the energy per symbol of an equivalent $\sqrt{M}$-ASK:

$$E_s = \frac{2}{3}(M-1)A^2 \qquad [3.84]$$

The detailed proof is the following:

$$\begin{aligned} E_s &= \frac{1}{M}\sum_{k=1}^{M}|c_k|^2 \\ &= \frac{1}{M}\left(\sum_{l=1}^{\sqrt{M}}\sum_{j=1}^{\sqrt{M}}\left(a_l^2+b_j^2\right)\right) \\ &= \frac{1}{M}\left(\sqrt{M}\sum_{l=1}^{\sqrt{M}}a_l^2+\sqrt{M}\sum_{l=1}^{\sqrt{M}}b_j^2\right) \\ &= \frac{2}{\sqrt{M}}\sum_{l=1}^{\sqrt{M}}a_l^2 = 2E_{\text{ASK}-\sqrt{M}} \end{aligned} \qquad [3.85]$$

Assuming synchronous detection, the demodulator perfectly separates the in-phase and quadrature components. Consequently, the symbol error probability P_e of the M-QAM can analytically be obtained from the symbol error probability of two $\sqrt{M}$-ASK.

No error takes place on the transmitted M-QAM symbol if there is no error on the estimates of both in-phase and quadrature component values. Thus, the probability to make a correct decision on M-QAM is:

$$P_{c,M-\text{QAM}} = \left(1-P_{e,\sqrt{M}-\text{ASK}}\right)^2 \qquad [3.86]$$

We recall that the symbol error probability of a $\sqrt{M}$-ASK is:

$$P_{e,\sqrt{M}-\text{ASK}} = \frac{(\sqrt{M}-1)}{\sqrt{M}}\text{erfc}\left(\sqrt{\frac{3}{(M-1)}\frac{E_s}{N_0}}\right) \qquad [3.87]$$

Moreover, for each $\sqrt{M}$-ASK constellation, the energy per symbol is: $E_s = \log_2(\sqrt{M})E_b = \frac{1}{2}\log_2(M)E_b$. The previous error probability can then be written as:

$$P_{e,\sqrt{M}-\mathrm{ASK}} = \frac{(\sqrt{M}-1)}{\sqrt{M}}\mathrm{erfc}\left(\sqrt{\frac{3\log_2(M)}{2(M-1)}\frac{E_b}{N_0}}\right) \qquad [3.88]$$

Equations [3.86] and [3.88] lead to expressing the symbol error probability of the M-QAM as follows:

$$\begin{aligned} P_{e,M-\mathrm{QAM}} &= 1 - P_{c,M-\mathrm{QAM}} \\ &= 1 - \left(1 - P_{e,\sqrt{M}-\mathrm{ASK}}\right)^2 \\ &= 2 \times P_{e,\sqrt{M}-\mathrm{ASK}} - \left(P_{e,\sqrt{M}-\mathrm{ASK}}\right)^2 \\ &= \frac{2\left(\sqrt{M}-1\right)}{\sqrt{M}}\mathrm{erfc}\left(\sqrt{\frac{3\log_2(M)}{2(M-1)}\frac{E_b}{N_0}}\right) \\ &\times \left[1 - \frac{1}{2}\frac{\left(\sqrt{M}-1\right)}{\sqrt{M}}\mathrm{erfc}\left(\sqrt{\frac{3\log_2(M)}{2(M-1)}\frac{E_b}{N_0}}\right)\right] \qquad [3.89] \end{aligned}$$

At high E_b/N_0, the error probability is low enough for the approximation:

$$\left(P_{e,\sqrt{M}-\mathrm{ASK}}\right)^2 \approx 0$$

to hold. Consequently, the symbol error probability of M-QAM can be approximated by formula:

$$P_{e,\mathrm{QAM}-M} \approx 2\frac{\left(\sqrt{M}-1\right)}{\sqrt{M}}\mathrm{erfc}\left(\sqrt{\frac{3\log_2(M)}{2(M-1)}\frac{E_b}{N_0}}\right) \qquad [3.90]$$

Approximation $\left(P_{e,\sqrt{M}-\mathrm{ASK}}\right)^2 \approx 0$ implies that some areas are counted twice in the error probability, similarly to the example provided for the 8-PSK modulation in section 3.3.4.3. At high E_b/N_0, their contribution in the error probability can be neglected.

The performances of QAM modulations are illustrated in Figure 3.24. Once more, the BER can be estimated at high E_b/N_0 by formula [3.72] when Gray encoding is used.

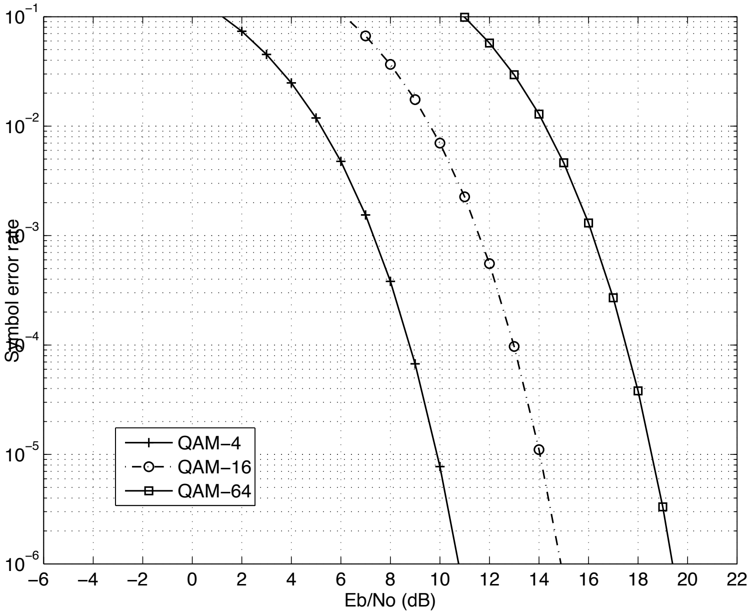

Figure 3.24. *Performances of QAM modulations*

Equation [3.89] only holds for regular QAM. There is no general expression of the symbol error probability for QAM where M is not a perfect square. Approximate formulas can, however, be obtained by using pairwise error probabilities. For instance, the 32-QAM cross-modulation of Figure 3.23 has a minimum distance $d_{\min} = 2A$, and its energy per symbol is $E_s = 20A^2$. 16 central points of the constellation all have 4 neighbors at minimum distance; 8 semi-external points (of coordinates $(5A, 3A)$, for example) have 3 neighbors at minimum distance; and 8 external points have 2 neighbors at $d_{\min}$. As a result, if we only take into account errors between points located at $d_{\min}$, the error probability is equal to:

$$\begin{aligned} P_e &= \frac{1}{32} \times \left(16 \times 4 \times P_{e,\text{pair}} + 8 \times 3 \times P_{e,\text{pair}} + 8 \times 2 \times P_{e,\text{pair}}\right) \\ &= \frac{13}{8} \times \text{erfc}\left(\sqrt{\frac{2A^2}{N_0}}\right) \\ &= \frac{13}{8} \times \text{erfc}\left(\sqrt{\frac{E_b}{4N_0}}\right) \\ &= 1.625 \times \text{erfc}\left(\sqrt{0.25\frac{E_b}{N_0}}\right) \end{aligned} \qquad [3.91]$$

Yet, if we improperly applied equation [3.89] to 32-QAM modulation, we would obtain:

$$P_e = 1.6464 \times \text{erfc}\left(\sqrt{0.2419\frac{E_b}{N_0}}\right) \qquad [3.92]$$

Thus, equation [3.89] is quite close to that obtained with the detailed analysis. It may then be used to approximately evaluate the performances of cross-modulations at high E_b/N_0, provided that they are constructed in a similar way as 32-QAM cross-constellation. This is particularly true for high values of M.

3.3.6. *Link between $\frac{E_b}{N_0}$ and signal-to-noise ratio depending on the power values*

The signal-to-noise ratio is expressed depending on the power values in most systems. In order to compute the symbol error probability of all modulations, it is necessary to express the signal-to-noise ratio depending on $\frac{E_b}{N_0}$.

The useful signal's power is: $P = \frac{E_s}{T} = \frac{E_b}{T_b}$.

The noise power in bandwidth B is $N = 2B \times (N_0/2) = B \times N_0$, because power in both positive and negative frequencies must be taken into account (see section 5.3). Consequently, the signal-to-noise ratio is equal to:

$$\frac{P}{N} = \frac{1}{T_b B}\frac{E_b}{N_0} = \frac{D_b}{B}\frac{E_b}{N_0} = \eta\frac{E_b}{N_0} \qquad [3.93]$$

where $\eta = \frac{D_b}{B}$ is the ratio between the binary rate and the bandwidth required to carry it. It is called spectral efficiency.

If the frequency response of the equivalent filter of the whole channel is rectangular, the bandwidth is minimum. Considering a linearly modulated signal of bandwidth B over frequency carrier f_c, the bandwidth is:

$$B = \frac{1}{T} = \frac{1}{\log_2(M)T_b} = \frac{D_b}{\log_2(M)} \qquad [3.94]$$

The spectral efficiency is then maximum and is equal to: $\eta = \log_2(M)$.

Then, for modulations of order M with maximum spectral efficiency, the signal-to-noise ratio is obtained from the $\frac{E_b}{N_0}$ as follows:

$$\frac{P}{N} = \log_2(M)\frac{E_b}{N_0} = \frac{E_s}{N_0} \qquad [3.95]$$

3.3.7. *Power spectrum density of regular modulations*

The power spectrum density of the equivalent baseband signal $s_l(t)$ has the same expression for all regular linear modulations (ASK, PSK and QAM). According to equation [3.19], it is equal to the product of the power spectrum density of the symbols by the squared modulus of the frequency response of the transmit filter, $G(f)$, divided by the symbol-period:

$$\gamma_{s_l s_l}(f) = \frac{1}{T}\gamma_{cc}(f)|G(f)|^2 \qquad [3.96]$$

The transmitted symbols c_k belong to a second-order stationary random sequence, with mean $m_c = 0$ and autocorrelation function:

$$R_{cc}(k) = E\left[c_{n+k}c_n^*\right] \qquad [3.97]$$

Since symbols are uncorrelated, $R_{cc}(k) = 0$ if $k \neq 0$. The power spectrum density of the symbols is: $\gamma_{cc}(f) = R_{cc}(0)$ because the Fourier transform of a Dirac function is a constant function.

The power spectrum density of the shaped signal is then:

$$\gamma_{s_l s_l}(f) = \frac{1}{T} R_{cc}(0) |G(f)|^2 \qquad [3.98]$$

Since the signal has a finite bandwidth, $G(f)$ (together with its matched receive filter) must verify the Nyquist criterion to guarantee that there is no intersymbol interference at the receiver. Thus, if it is chosen as a root raised cosine filter, the power spectrum density of the baseband signal is equivalent to a raised cosine function.

We now consider the narrowband passband signal. Its power spectrum density is deduced from the equivalent baseband signal by equation [3.23]. It is equal to:

$$\gamma_{s_l s_l}(f) \frac{1}{4T} R_{cc} \left(|G(f - f_c)|^2 + |G(-f - f_c)|^2 \right) \qquad [3.99]$$

Consequently, it is composed of two raised cosine functions centered on frequencies f_c and $-f_c$.

3.3.8. *Conclusion*

Performances of linear modulations can jointly be compared from Figures 3.13, 3.20 and 3.24. PSK modulations are more efficient than ASK modulations. The main advantage of ASK modulations is their easy implementation, which is due to the fact that they are real modulations. However, they are rarely used in practice due to their poor performances.

In all cases, QAMs are more efficient than PSK and ASK. They are thus used in many systems, as for instance assymetric digital subscriber line (ADSL), digital terrestrial television, cellular systems from their 3$^{\text{rd}}$ generation and so forth. PSK modulations are nevertheless used for satellite communications, where the fact that their complex envelope has a constant modulus is interesting, since it avoids nonlinearities at power amplifiers.

Finally, we may note that when the constellation's order M increases, the binary rate increases for a fixed bandwidth, or equivalently, the bandwidth decreases for a fixed binary rate. However, the required E_b/N_0 ratio to reach a given BER increases with M. As the power budget is limited in transmission systems, the constellation's size must be chosen as the highest possible, for a given power. In wireless mobile systems, this is called modulation-level adaptation: if for instance, QPSK, 16-QAM and 64-QAM are available, the chosen modulation depends on the useful power level at the receiver. Thus, the further away a mobile is from its base station, the lower its useful received power becomes (because of the attenuation which increases with the distance), the lower the constellation size of the modulation is. Broadly speaking, a mobile terminal located next to the base station uses 64-QAM, a mobile terminal located in the middle of the cell uses 16-QAM, and a mobile terminal at the border of the cell uses QPSK. Since the bandwidth is fixed (and assuming that the channel coding remains unchanged), this implies that a user at the center of the cell obtains 3 times higher data rate than a user at the border of the cell.

3.4. Frequency shift keying

3.4.1. *Definitions*

FSK modulations are nonlinear modulations, as the transmitted information is contained in the instantaneous frequency. The signal's amplitude is constant, but its carrier frequency varies in time. To each value of the instantaneous frequency corresponds a modulated symbol. This type of modulation is nonlinear since, contrary to amplitude, phase and quadrature modulations, the equivalent baseband signal of the original narrowband signal does not allow the receiver to recover the transmitted symbols.

The modulated signal is written as follows:

$$\begin{aligned} s(t) &= \sum_{k=-\infty}^{+\infty} Re\left[Ae^{j(2\pi f_0 t)}e^{j\Psi(t)}.\right]\Pi_T(t-kT) \\ &= \sum_{k=-\infty}^{+\infty} A\cos(2\pi f_0 t + \Psi(t))\Pi_T(t-kT) \end{aligned} \qquad [3.100]$$

where f_0 is the central frequency, $\Psi(t)$ is the phase shift and $\Pi_T(t)$ is the rectangular function, which is equal to 1 between 0 and T and to 0 elsewhere.

The instantaneous frequency $f_c(t)$ is the derivative of phase $f_0 t + \frac{1}{2\pi}\Psi(t)$ with respect to time:

$$f_c(t) = f_0 + \frac{1}{2\pi}\frac{\partial\Psi(t)}{\partial t} \qquad [3.101]$$

$\frac{1}{2\pi}\frac{\partial\Psi(t)}{\partial t}$ is called the frequency deviation. It represents the variation of the instantaneous frequency compared to the central frequency. The frequency deviation's value indicates which modulated symbol is carried by the signal in the considered symbol-period.

The minimum distance between two values of the instantaneous frequency [3.101], called frequency excursion, is denoted byΔf.

We consider an M-order FSK modulation. The instantaneous frequency is determined as a function of the symbol to be transmitted, based on the symbols of a regular ASK of the same order (with $A = 1$): $a_k \in \{-(M-1), ..., -3, -1, 1, ..., (M-1)\}$. The frequency deviation is written as a function of a_k as follows:

$$\frac{1}{2\pi}\frac{\partial\Psi(t)}{\partial t} = \frac{\Delta f}{2}\sum_{k=-\infty}^{+\infty} a_k h(t-kT) \qquad [3.102]$$

where $h(t)$ is the frequency impulse. We first assume that it is equal to a rectangular function: $h(t) = \Pi_T(t)$. Consequently, the frequency deviation linearly depends on the symbols to transmit. Considering the k^{th} symbol transmitted in interval $t \in [kT, (k+1)T[$, the frequency deviation is:

$$\frac{1}{2\pi}\frac{\partial\Psi(t)}{\partial t} = \frac{\Delta f}{2}a_k \qquad [3.103]$$

The phase impulse is obtained by integrating this function over $t \in [kT, (k+1)T[$:

$$\Psi(t) = \pi\Delta f a_k(t-kT) + \theta_k \qquad [3.104]$$

where $\theta_k = \Psi(kT)$ is a constant value.

Equation [3.104] shows that the phase linearly varies in interval $[kT, (k+1)T[$. This variation is equal to $\pi \Delta f a_k T$.

The constant's value, θ_k, determines the FSK type, which may be either discontinuous or continuous. These two types of modulations are studied in the next sections.

3.4.2. *Discontinuous-phase FSK*

In the most simple case when $\theta_k = \pi \Delta f a_k k T$, the instantaneous frequency is equal to:

$$f_c(t) = f_0 + \frac{\Delta f}{2} a_k \qquad [3.105]$$

The modulated signal is then written as follows:

$$s(t) = \sum_{k=-\infty}^{+\infty} A \cos\left(2\pi(f_0 + \frac{\Delta f}{2} a_k)t\right) \Pi_T(t - kT) \qquad [3.106]$$

As equation [3.106] shows, the phase at transition times kT is abruptly modified. As a result, the modulation is called discontinuous-phase FSK.

An example of discontinuous-phase FSK with two states is given in Figure 3.25. In this case, the central frequency and frequency excursion are equal to $f_0 = 5$ KHz and $\Delta f = 7.4$ KHz, respectively. Symbols $a_k = \{-1, 1\}$ are used to transmit bits $\{0, 1\}$ and amplitude A is equal to 1. The instantaneous frequency is thus equal to either $f_c(t) = f_1 = f_0 + \frac{\Delta f}{2}$ or $f_{-1} = f_0 - \frac{\Delta f}{2}$. By its definition, the central frequency is $f_0 = \frac{f_1 + f_{-1}}{2}$, and the frequency excursion is $\Delta f = |f_1 - f_{-1}|$.

The phase discontinuities when the binary sequence 01101 is transmitted are illustrated in Figure 3.26.

Discontinuous-phase FSK is implemented with a switch that passes from one oscillator to another depending on the symbol to transmit, at each symbol-period. Its main drawback is that phase discontinuities widen the spectrum, since the more irregular a signal is, the wider its side lobes become.

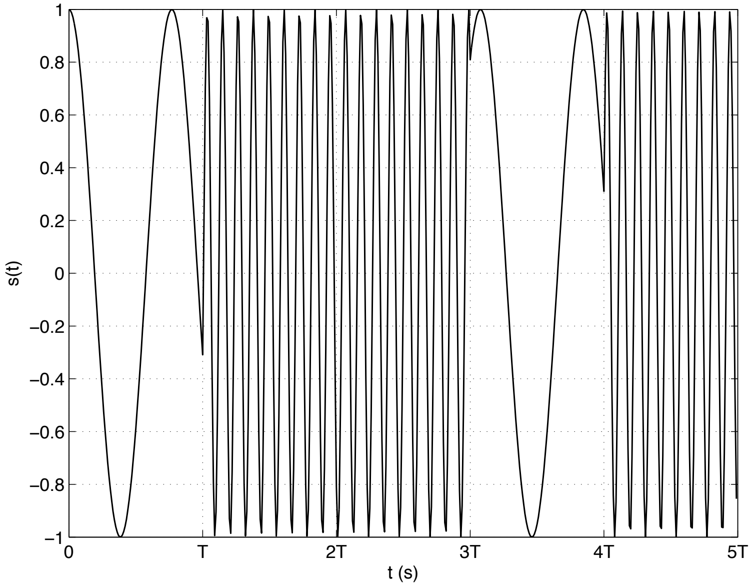

Figure 3.25. *Discontinuous-phase 2-FSK:* $s(t)$

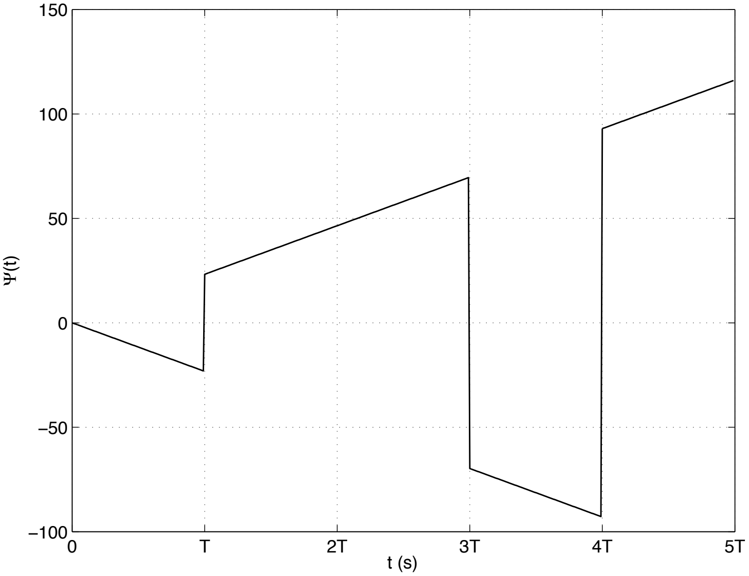

Figure 3.26. *Discontinuous-phase 2-FSK:* $\Psi(t)$

3.4.3. *Continuous-phase FSK*

Continuous-phase FSK must be used in order to decrease the bandwidth. Phase continuity is obtained by varying parameter θ_k in equation [3.104] from one symbol-period to the other.

θ_k must be determined for each symbol-period so that phase continuity is ensured between the end of transmission of a_{k-1} and the beginning of transmission of a_k. This constraint is equivalent to the following equation:

$$\theta_k = \theta_{k-1} + \pi \Delta f T a_{k-1} \qquad [3.107]$$

Thus, constant θ_k in symbol-period k depends on the value of this constant in the previous symbol-period, and on the value of the previously transmitted symbol a_{k-1}. This leads to a continuity of the phase impulse $\Psi(t)$ whenever its slope changes.

An example of binary continuous-phase FSK is given in Figure 3.27. The parameters are the same as in the previous section. We can verify in Figure 3.28 that the phase is continuous.

In order to implement continuous-phase FSK modulations, only one oscillator must be used, and its frequency must be modulated at the symbol-period.

3.4.4. *Demodulation*

In this section, we detail how the receiver must process the signal transmitted with FSK, denoted as $s(t)$, and that has passed through an additive white Gaussian noise channel with noise $b(t)$. At the receiver, the passband signal $r(t) = s(t) + b(t)$ is compared with all possible signals of the FSK. Let $\Psi_m(t)$ be the phase of the m^{th} FSK symbol. The receiver computes the following scalar product:

$$(r|s_m) = \int_{kT}^{(k+1)T} r(t) \times A\cos(2\pi f_0 t + \Psi_m(t))\Pi_T(t-kT)dt \qquad [3.108]$$

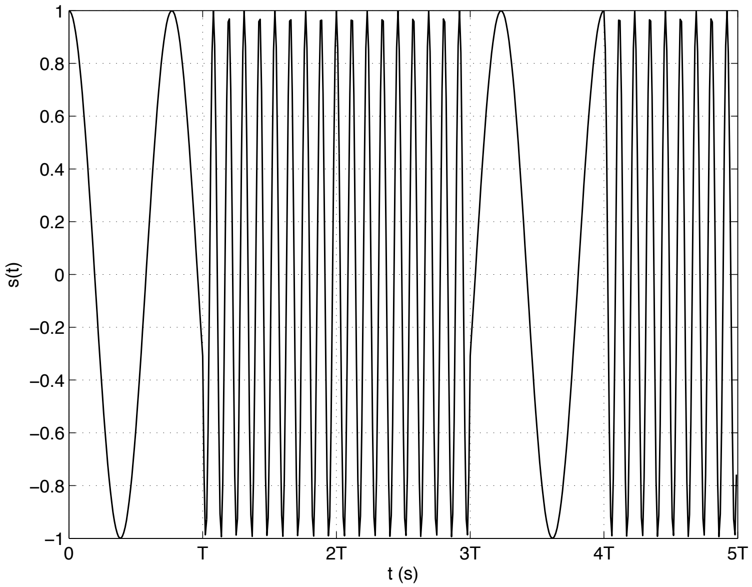

Figure 3.27. *Continuous-phase 2-FSK:* $s(t)$

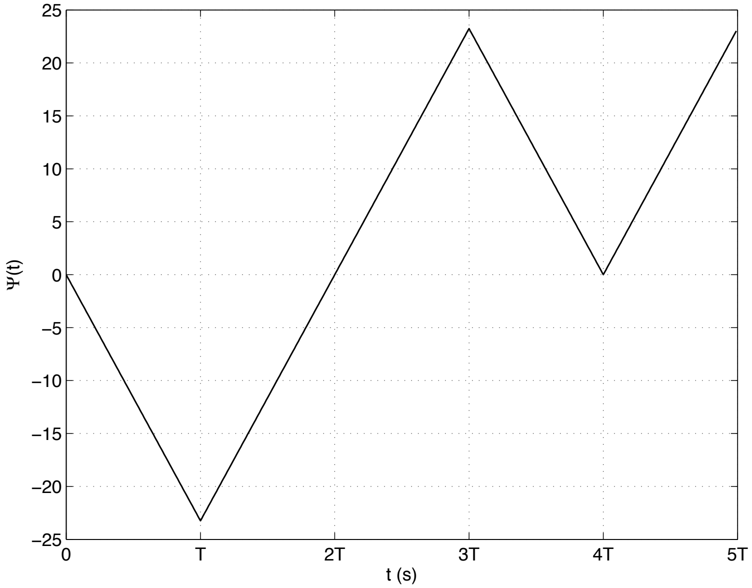

Figure 3.28. *Continuous-phase 2-FSK:* $\Psi(t)$

The possible instantaneous frequencies' values are chosen so that all couples of transmitted noiseless signals are jointly orthogonal:

$$\begin{aligned}(s_n|s_m) &= \int_{kT}^{(k+1)T} A^2 \cos(2\pi f_0 t + \Psi_n(t)) \\ &\quad \times \cos(2\pi f_0 t + \Psi_m(t))\,(\Pi_T(t-kT))^2\,dt \\ &= \begin{cases} 0 \text{ if } n \neq m \\ 1 \text{ if } n = m \end{cases} \end{aligned} \qquad [3.109]$$

As a result, if the noise power is low, the receiver estimates that the transmitted symbol is a_m when the scalar product $(r|s_m)$ is maximum.

In order to verify the orthogonality constraint [3.109], the frequency excursion Δf must take some particular values. They can easily be computed if we consider a binary FSK. The solution can then directly be extended to higher order modulations.

With 2- FSK, the signal transmitted in symbol-period T is either $s_1(t) = A\cos(2\pi f_1 t)$, or $s_2(t) = A\cos(2\pi f_2 t)$. $s_1(t)$ corresponds to symbol $a_k = -1$ and $s_2(t)$ corresponds to symbol $a_k = 1$.

The scalar product of $s_1(t)$ with itself, which is equal by definition to the energy $E_{s,1}$ of the transmitted signal, is:

$$\begin{aligned}(s_1|s_1) &= \frac{1}{2}\int_0^T A^2 \cos^2(2\pi f_1 t)\,(\Pi_T(t-kT))^2\,dt \\ &= \frac{A^2}{2}\int_0^T (1+\cos(4\pi f_1 t))\,dt \\ &= \frac{A^2}{2}\left(T + \frac{1}{4\pi f_1}\sin(4\pi f_1 T)\right) \\ &\approx \frac{A^2 T}{2} \end{aligned} \qquad [3.110]$$

since $\frac{sin(4\pi f_1 T)}{4\pi f_1 T}$ is equal to 0 when $f_1 T$ is high with respect to 1.

To fulfill constraint [3.109], the scalar product of two different signals must be equal to 0:

$$\begin{aligned}(s_1|s_2) &= \int_0^T A^2 \cos(2\pi f_1 t)\cos(2\pi f_2 t)\left(\Pi_T(t-kT)\right)^2 dt\\ &= \frac{A^2}{2}\int_0^T \left(\cos(2\pi(f_2-f_1)t)+\cos(2\pi(f_1+f_2)t)\right) dt\\ &= \frac{A^2}{2}\left(\frac{1}{2\pi(f_2-f_1)}\sin(2\pi(f_2-f_1)T)\right.\\ &\quad\left.+\frac{1}{2\pi(f_1+f_2)}\sin(2\pi(f_1+f_2)T)\right)\\ &= \frac{A^2T}{2}\times\left(\frac{1}{2\pi(f_2-f_1)T}\sin(2\pi(f_2-f_1)T)\right.\\ &\quad\left.+\frac{1}{2\pi(f_1+f_2)T}\sin(2\pi(f_1+f_2)T)\right)\end{aligned} \qquad [3.111]$$

We can estimate that $\frac{sin(2\pi(f_1+f_2)T)}{2\pi(f_1+f_2)T} \approx 0$ since$(f_1+f_2)T >> 1$. Thus, in order to have a null scalar product, the following constraint must hold: $\sin(2\pi(f_2-f_1)T)=0$. This is equivalent to:

$$2\pi(f_2-f_1)T = k\pi \qquad [3.112]$$

where k is any integer value. Finally, the frequency excursion must verify:

$$(f_2-f_1) = \frac{k}{2T} \qquad [3.113]$$

The minimum frequency excursion is obtained $k=1$. It is equal to:

$$\Delta f = (f_2-f_1) = \frac{1}{2T} \qquad [3.114]$$

In order to minimize the required bandwidth of 2- FSK modulations, it is preferable to choose this value of the frequency excursion. The modulation index is defined as the product $m = T\Delta f$. It is then equal to $m = 1/2$.

Minimum shift keying (MSK) is a continuous-phase FSK modulation whose modulation index is $1/2$. It can be shown that the power spectrum density of the equivalent baseband signal $s_l(t) = Re\left[Ae^{j\Psi(t)}\right]$ of MSK is [PRO 08]:

$$\gamma_{s_l s_l}(f) = \frac{16A^2T}{\pi^2}\left(\frac{\cos(2\pi fT)}{1-16(fT)^2}\right)^2 \qquad [3.115]$$

Each transmitted symbol has the same energy E_s, equal to $A^2T/2$. This energy is located around frequency f_1 when signal s_1 is transmitted, and around f_2 when signal s_2 is transmitted. Thus, symbols can be represented as vectors in an orthonormal basis as follows: $\mathbf{s}_1 = \left(\sqrt{E_s}, 0\right)$ and $\mathbf{s}_2 = \left(0, \sqrt{E_s}\right)$. The distance between these symbols consequently is: $d = \sqrt{2E_s}$.

Orthogonal FSK with higher number of symbols M can similarly be represented in vector space. The i^{th} symbol is represented in an orthonormal basis of dimension M by $\mathbf{s}_i = \left(0, 0, ..., \sqrt{E_s}, ..., 0\right)$, and its energy is located around carrier frequency f_i. All symbols are then at equal distance $d_{\text{min}} = \sqrt{2E_s}$, which is the minimum distance of an M- FSK modulation.

3.4.5. *GMSK modulation*

Up to now, we assumed that the frequency impulse $h(t)$ in the frequency deviation was a rectangular function over T (see equation [3.102]). This leads to discrete modifications of the instantaneous frequency between two symbol-periods. The power spectrum density of FSK modulations is hard to determine analytically. However, its bandwidth depends on the phase impulse $\Psi(t)$, and consequently of the frequency impulse $h(t)$. The shorter impulse $h(t)$ is in the time domain, the larger the bandwidth of the power spectrum density becomes. When $h(t)$ is a rectangular function in the symbol-period, the signal's bandwidth has its largest bandwidth.

Gaussian minimum shift keying (GMSK) is a binary continuous-phase FSK modulation with $\Delta f = \frac{1}{2T}$, in which the frequency impulse $h(t)$ is widened in time, in order to reduce the signal's bandwidth. It is defined as follows:

$$h(t) = g_a(t) * \Pi_T(t) \qquad [3.116]$$

where $\Pi_T(t)$ is the rectangular function over T, and $g_a(t)$ is a Gaussian function:

$$g_a(t) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-t^2}{2\sigma^2}} \qquad [3.117]$$

As a result, the phase variation during symbol-period $[kT, (k+1)T[$ is no longer linear. Variance σ^2 is determined depending on parameter $B = \frac{\ln(2)}{4\pi^2\sigma^2}$ which is related to the 3 dB-decrease of the frequency response h:

$$\frac{|H(B)|^2}{|H(0)|^2} = \frac{1}{2} \qquad [3.118]$$

The impulse response $h(t)$ is computed as follows:

$$\begin{aligned} h(t) &= g_a(t) * \Pi_T(t) \\ &= \int_{-\infty}^{\infty} h(u) * \Pi_T(t-u) du \\ &= \int_{t-T}^{t} \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-u^2}{2\sigma^2}} du \\ &= \int_{-\infty}^{t} \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-u^2}{2\sigma^2}} du - \int_{-\infty}^{t-T} \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-u^2}{2\sigma^2}} du \\ &= \int_{-\frac{t}{2\sigma^2}}^{\infty} \frac{1}{\sqrt{\pi}} e^{-x^2} dx - \int_{-\frac{(t-T)}{2\sigma^2}}^{\infty} \frac{1}{\sqrt{\pi}} e^{-x^2} dx \\ &= \frac{1}{2}\left[\text{erfc}\left(-\frac{t}{2\sigma^2}\right) - \text{erfc}\left(-\frac{(t-T)}{2\sigma^2}\right)\right] \end{aligned} \qquad [3.119]$$

$h(t)$ can finally be written as a function of B:

$$h(t) = \frac{1}{2}\left[\text{erfc}\left(-\frac{2\pi B}{\sqrt{2\ln(2)}}t\right) - \text{erfc}\left(-\frac{2\pi B}{\sqrt{2\ln(2)}}(t-T)\right)\right] \qquad [3.120]$$

Equation [3.120] indicates that $h(t)$ has an infinite duration. Nevertheless, in practice, it is truncated in order to obtain a finite duration. It is represented in Figure 3.29, where the transmission period has been centered around 0 and time is normalized by T. We can see that the lower BT becomes, the slower the impulse response decrease is. As a result, the frequency response $H(f)$ has a lower bandwidth.

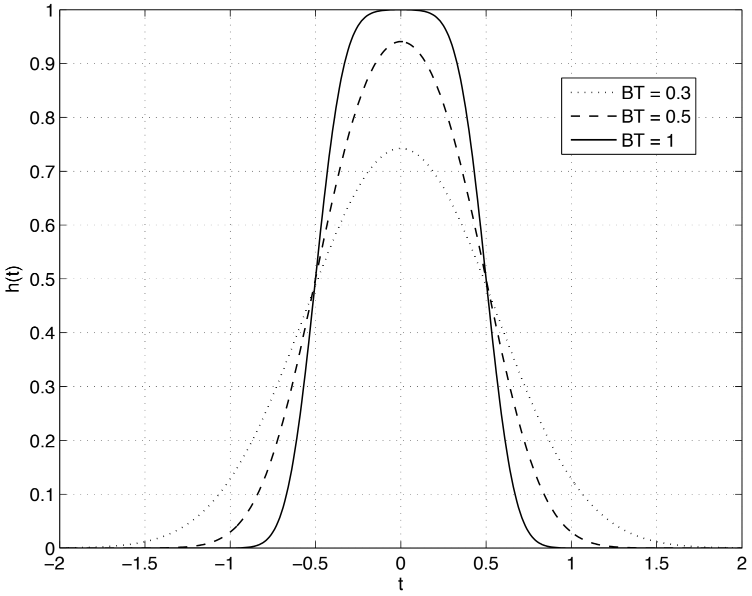

Figure 3.29. *Impulse response of* $h(t)$

GMSK is well adapted to narrowband systems. It has been chosen for the second generation of cellular mobile systems global system for mobile communications (GSM), with $BT = 0.3$.

3.4.6. *Performances*

FSK performances depend on several parameters: the constellation's size, phase continuity and frequency excursion. It is far more complicated to provide a BER formula as a function of the E_b/N_0 ratio than for linear modulations. In order to achieve good performances, the transmitted signals (corresponding to various instantaneous frequencies) must be orthogonal. This implies that the higher the order of the constellation becomes, the larger the bandwidth becomes.

An upper bound on the symbol error probability of orthogonal FSK modulations with M symbols can be evaluated. All symbols are at the same distance $d_{\text{min}} = \sqrt{2E_s}$. Each symbol has $M-1$ neighbors with whom a detection error may take place on the additive white Gaussian noise channel. Thus, the error probability is less than or equal to $M-1$ times the pairwise error probability, which is equal to:

$$\begin{aligned} P_{e,\text{pair}} &= \frac{1}{2}\text{erfc}\left(\frac{d_{\text{min}}}{2\sqrt{N_0}}\right) \\ &= \frac{1}{2}\text{erfc}\left(\sqrt{\frac{E_s}{2N_0}}\right) \\ &= \frac{1}{2}\text{erfc}\left(\sqrt{\frac{\log_2(M)E_b}{2N_0}}\right) \end{aligned} \qquad [3.121]$$

Consequently:

$$P_e \leq \frac{M-1}{2}\text{erfc}\left(\sqrt{\frac{\log_2(M)}{2}\frac{E_b}{N_0}}\right) \qquad [3.122]$$

The BER is computed from the symbol error probability by using the following argument: let us consider the l^{th} bit, set for instance to 1, in the sequence of $\log_2(M)$ bits corresponding to one symbol. Among all symbols, $M/2$ symbols have a different value for the l^{th} bit. In our case, it is consequently equal to 0. An error may occur at the receiver when demodulating $M/2$ symbols among the $M-1$ symbols that are different from the one we studied. To conclude, the BER is equal to:

$$\text{BER} = \frac{M}{2(M-1)}P_e \qquad [3.123]$$

It is represented in Figure 3.30 using the upper bound [3.122].

The BER performances depending on the E_b/N_0 ratio improve when the constellation's size increases. Thus, the behavior of FSK is opposite that of linear modulations concerning the following two points:

– spectral efficiency: with linear modulations, when M increases, the bandwidth B required to reach a given data rate D_b decreases. As a result, the spectral efficiency increases. With FSK, when M increases, the required bandwidth also increases, in order to fulfill the orthogonality constraint between instantaneous frequencies. If $M = 2^n$, the required bandwidth is $B = (M-1)\Delta f$ and the binary rate is $D_b = n/T$. The spectral efficiency is:

$$\eta(n) = \frac{D_b}{B} = \frac{2n}{M-1} = \frac{2n}{2^n-1} \qquad [3.124]$$

where we assumed that $\Delta f = 1/(2T)$. The ratio between spectral efficiencies $\eta(n+1)/\eta(n)$ is always lower than 1, and decreases when n increases. Thus, the spectral efficiency of FSK modulations decreases when M increases. We can note that we considered here the most favorable case, when the frequency excursion is at its minimum;

– performances: with linear modulation, when M increases, the required E_b/N_0 to reach a given BER increases. On the contrary, with FSK, when M increases, the required E_b/N_0 to reach a given BER decreases (see Figure 3.30).

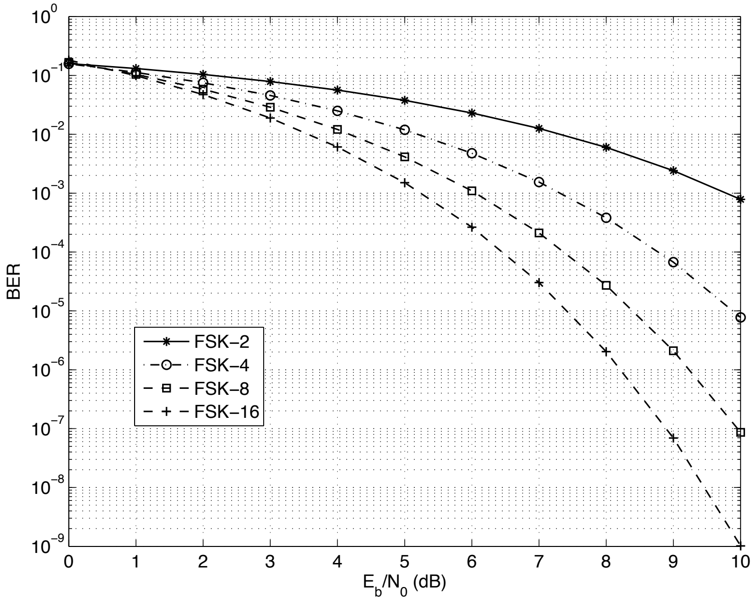

Figure 3.30. *Approximated BER of FSK*

3.5. Conclusion

This chapter has detailed modulations on sine waves and their performances. It first studied the equivalent baseband model. It was proven that we can equivalently consider a narrowband passband signal filtered by a channel with the same characteristics, or consider the signal's complex envelope, filtered by the equivalent baseband channel of the original channel. This result allows us to bring back the study of any narrowband passband signal to that of its complex envelope. In the case of linear modulations, the complex envelope carries the symbols. Thus, the performances of linear modulations are independent of the carrier frequency of the transmitted signal.

The main linear modulations have been detailed in terms of constellation, energy and minimum distance. The error probability depending on E_b/N_0 ratio or on the signal-to-noise ratio was then deduced for ASK, PSK and QAM. The latter are the most efficient in terms of symbol error probability and BER, as they allow us to transmit a high number of symbols with low energy and low minimum distance. We can note that we only considered regular modulations; an example of irregular modulation is proposed in exercise 3.6.2.

Finally, FSK modulations were introduced in the last section of this chapter. For these modulations, it is important to achieve phase continuity and to use the lowest possible bandwidth.

All results obtained in this chapter assume that the channel adds a white Gaussian noise and that there is no time or frequency shift. Synchronization and equalization techniques, which allow us to estimate symbols even if the channel introduces distortions, will be studied in Chapter 4.

3.6. Exercises

3.6.1. *Exercise 1: constellations of 8-QAM*

We consider two 8-QAM modulations represented in Figure 3.31. The minimum distance between two adjacent points is $2A$.

1) Determine the coordinates of the constellation's points on both cases.

2) Evaluate the energy of each constellation, and deduce which constellation is the most efficient.

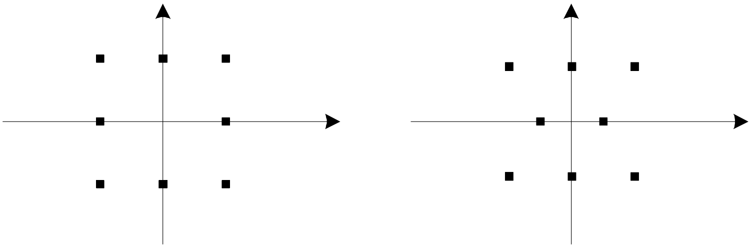

Figure 3.31. *8-QAM modulations*

3.6.2. *Exercise 2: irregular ASK modulation*

We consider an irregular ASK with eight symbols, whose constellation is given in Figure 3.32.

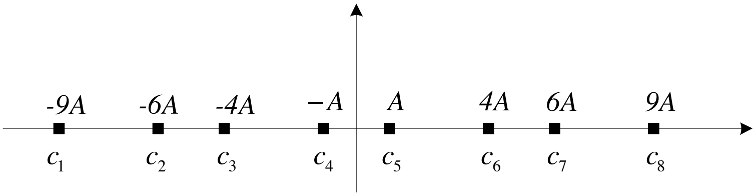

Figure 3.32. *Irregular constellation*

1) Compute the minimum distance.

2) Compute the energy per symbol E_s, and the energy per bit E_b.

3) The signal is subject to a white additive Gaussian noise of variance $\sigma^2 = N_0/2$. Compute the error probability knowing that symbol c_k has been transmitted, for $k = 1, 2, ..., 8$, depending on A and N_0. We assume that errors are likely to happen only between two adjacent symbols, and that the detection thresholds at the receiver are set in the middle of each pair of adjacent neighbors.

4) Deduce the average symbol error probability depending on E_b/N_0.

5) Compare to the average symbol error probability of a regular 8-ASK with optimum detection thresholds. Perform a numerical application when

$E_b/N_0 = 10$ dB, using the following approximate formula: $\frac{1}{2}\text{erfc}(\sqrt{\frac{4.03}{2}}) = 0.0223$, $\frac{1}{2}\text{erfc}(\sqrt{\frac{1.791}{2}}) = 0.0904$ and $\frac{1}{2}\text{erfc}(\sqrt{\frac{2.857}{2}}) = 0.0455$.

3.6.3. *Exercise 3: comparison of two PSK*

We consider a coaxial cable with attenuation 13 dB/100 m. The receiver is located at 1 km from the transmitter.

The signal is subject to an additive white Gaussian noise of power spectrum density $N_0/2$, with $N_0 = -204$ dB/Hz.

It wants to receive a signal with data rate $D = 10$ Mbits/s and BER = 10^{-2}. The equivalent filter of the whole channel is assumed rectangular in the frequency domain.

Performances are evaluated with two modulations: 4-PSK and 16-PSK.

1) Compute the required bandwidths B_4 and B_{16} for each modulation.

2) Compute the noise power in both bandwidths B_4 and B_{16}.

3) Assuming we used Gray encoding, compute the symbol error probability with both modulations.

4) Using the performance curves of PSK, determine the $\frac{E_b}{N_0}$ ratio corresponding to each modulation.

5) Compute the signal-to-noise ratio, $\frac{P}{N}$, for each modulation.

6) Deduce the received power in both cases.

7) Deduce the required transmit power in both cases.

3.6.4. *Exercise 4: comparison of QAM and PSK modulations*

In this exercise, 16-QAM and 16-PSK modulations are compared. The minimum distance in 16-QAM constellation is $2B$. Points of the 16-PSK constellation are located on a circle of radius A.

1) Graphically represent both constellations.

2) Evaluate the number of bits per symbol.

3) Propose Gray encoding for each constellation.

4) Determine the energy per symbol and the energy per bit for both constellations.

5) When A is fixed, express B depending on A so that the energy per symbol of 16-QAM is equal to the energy per symbol of 16-PSK.

6) Compare the minimum distance of both constellations in that case. Provide a numerical application. What can we deduce on the performances?

7) Give the expression of the symbol error probability and BER for both modulations, depending on E_b/N_0.

8) Perform numerical application when $E_b/N_0 = 10$ dB and $E_b/N_0 = 20$ dB, using the following approximate formula:

$$Q(x) = 0.208\exp(-0.971x^2) + 0.147\exp(-0.525x^2)$$

where $Q(a) = \frac{1}{2}\text{erfc}\left(\frac{a}{\sqrt{2}}\right)$.

9) Conclude on the performances of both modulations.

3.6.5. *Exercise 5: comparison of 8-PSK and 8-QAM modulations*

We consider two constellations with eight symbols: 8-PSK and 8-QAM (see Figure 3.33). The minimum distance, denoted as d, is the same for both constellations.

1) Compute the radius r of 8-PSK depending on d.

2) In the considered 8-QAM, we assume that the distance between points C2 and C3 is equal to the distance between points C1 and C2. Compute radii a and b depending on d. For this, use the following steps:

a) Compute the distance between C2 and C3 and deduce a depending on d.

b) Compute the distance between C1 and C2. Compute b so that this distance is equal to the distance between C2 and C3. Deduce b depending on d.

3) Compute the average energy of both constellations, E_{s-8PSK} and E_{s-8QAM} and compare them.

4) Give the expression of the symbol error probability of 8-PSK depending on E_b/N_0.

5) Compute the expression of the symbol error probability of 8-QAM, using pairwise error probability between adjacent points of the constellation.

6) Compute the expression of the symbol error probability with both constellations if $E_b/N_0 = 8$ dB. For this, use the following approximate formula:

$$Q(x) = 0.208\exp(-0.971x^2) + 0.147\exp(-0.525x^2)$$

$$Q(a) = \frac{1}{2}\text{erfc}\left(\frac{a}{\sqrt{2}}\right)$$

7) Give the link between BER and symbol error probability if Gray encoding is used. Determine, if possible, a Gray encoding for both constellations.

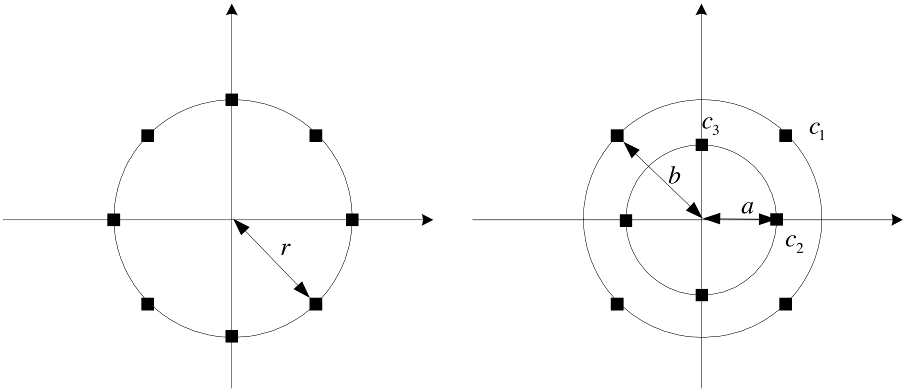

Figure 3.33. *8-PSK and 8-QAM modulations*

3.6.6. *Exercise 6: comparison of 2-FSK and 2-ASK modulations*

We consider a 2-FSK. During symbol-period T, the transmitted signal is either $s_1(t) = \cos(2\pi f_1 t)$, or $s_2(t) = \cos(2\pi f_2 t)$. We assume that $f_1T >> 1$ and $f_2T >> 1$, and that $\frac{sin(x)}{x} \approx 0$ if x is large enough.

1) Compute the mean square distance distance between both signals: $d^2 = \int_0^T (s_1(t) - s_2(t))^2 \, dt$.

2) Study this distance's variations depending on $m = T\,|f_2 - f_1|$.

3) Deduce the performance loss in terms of distance compared to 2-ASK. Perform the numerical application when $m = 0.5$ and when the mean square distance of 2-FSK is maximum, and conclude.

3.6.7. *Exercise 7: comparison of 16-QAM and 16-FSK*

We aim at transmitting the binary sequence 0101110011100010, where the most significant bit is on the left. The carrier frequency is denoted by f_c.

The complete filter is assumed rectangular in frequency domain (roll-off $= 0$).

1) We use a 16-QAM modulation with minimum distance $2A$. Represent the constellation and provide Gray encoding.

2) How many symbol-periods are required to transmit the binary sequence?

3) Express the transmitted signal $s(t)$ put on sine waveform in each symbol-period.

4) If E_b/N_0 is equal to 12 dB, what is the value of the symbol error probability?

5) Was it relevant to use 16-QAM? Would another linear modulation have been more efficient ?

6) A 16-FSK is now used. The central frequency is $f_0 = 2$ GHz and the difference of instantaneous frequency between two adjacent symbols is $\Delta f = 200$ kHz. Give the values of the 16 instantaneous frequencies.

7) $\Delta f = 1/(2T)$ is chosen, where T is the symbol-period. Compute the symbol rate. Deduce the binary rate of 16-FSK.

8) Compute the total bandwidth B used with 16-FSK.

9) Compute the symbol rate if the same bandwidth B is used with a 16-QAM. Deduce the binary rate and conclude.

4

Synchronization and Equalization

4.1. Introduction

In Chapter 3, we studied the performances of digital modulations in the ideal additive white Gaussian channel case with perfect synchronization between transmitter and receiver. Yet, these two assumptions are unrealistic in most transmission systems, which may introduce distortions, phase and frequency shifts, as well as delays.

The receiver must then correct these issues in order to perform signal detection. This chapter aims at detailing the operations performed at the receiver in order to apply detection as optimally as possible, in spite of the inter-symbol interference introduced by distortion and asynchronicity. The chapter is divided into separate sections: synchronization and equalization.

Section 4.2 presents the different techniques implemented at the receiver so that it can acquire synchronicity parameters and estimate the channel. These techniques are either data-aided, which means that the transmitter sends some training sequence already known by the receiver, of blind. This chapter studies statistical methods for the estimation of the frequency shift and time delay. Cramer–Rao's bound is then introduced. It measures the relevance of each estimator. The specific case of synchronization in multi-carrier orthogonal frequency division multiple access (OFDM) systems is not detailed in this chapter, but in section 5.3.6.

Once synchronization parameters have been acquired by the receiver, equalization is performed before detection, or jointly with detection,

Digital Communications 2: Digital Modulations, First Edition. Mylène Pischella and Didier Le Ruyet.

depending on the receiver's complexity. Three main classes of equalizers are detailed in section 4.3: linear equalizations, decision-feedback equalizers and maximum likelihood equalizer.

NOTE.– In the whole chapter, both signal and channel are baseband. The obtained results can directly be extended to passband signals.

4.2. Synchronization

Detection at the receiver is performed in three steps (see section 3.2.4): first, the frequency carrier must be removed to work on the baseband signal, then the signal must be sampled at symbol-period, equalized if necessary, and finally symbols must be detected either with a threshold detector, or with Viterbi's algorithm.

Yet, all these steps require some knowledge at the receiver on the transmitted signal, on: the carrier frequency, phase, symbol-period and, for digital-framed transmission systems, the beginning of each frame.

In order to pass from the narrowband passband signal to the equivalent baseband signal, the carrier frequency f_c must be known. Any frequency shift δf at the receiver leads to rotations of the constellation that may vary in time. This frequency shift may be due to poor frequency calibration at the receiver, or due to the Doppler effect in the case of wireless mobile channels (with mobile transmitter or receiver) and non-geostationary satellites.

Poor calibration of phase Ψ at the receiver also leads to rotations of the constellation that are yet less problematic since they are constant in time. Nevertheless, as seen in section 3.2.4, phase asynchronicity leads to cross-talk between in-phase and quadrature components of the signal when phase shift keying (PSK) and quadrature amplitude modulation (QAM) modulations are used in particular.

In digital systems, time synchronization is performed in two steps. First, the beginning of each frame must be determined. Then, the symbol-period must be obtained. Knowledge of the symbol-period is critical for sampling. Let us assume that the channel is verified. In this case, there is no inter-symbol interference if samples are taken every symbol-period T. These sampling moments correspond to the moments where the eye diagram is the most open (see Figure 2.35 in section 2.5.3). If sampling does not take place every kT, the Nyquist criterion no longer holds and inter-symbol interference occurs. This degrades the signal-to-noise plus-interference ratio (SNR) and, consequently, the bit error rate.

In the previous chapters, we assumed perfect synchronization at the receiver: there was no frequency, phase or time shift at the receiver. We now present several synchronization techniques implemented to correct frequency shifts and recover the symbol-period. Phase shift correction is not considered, since this shift can be included in the channel's impulse response and can thus be corrected once the channel has been estimated. To simplify, we will assume that symbol-period recovery is performed independently of frequency shift correction. Nevertheless, joint techniques exist [KOB 71, MEY 80]. In the specific case of OFDM systems, frame period recovery will be studied in Chapter 5, as well as frequency shift correction.

Finally, channel estimation is performed after frequency synchronization, if the channel introduces distortions. This step is necessary to perform equalization afterward.

In this book, we decided to only consider synchronization on sampled data, after they passed in the analog to digital converter. Analog synchronization techniques have been widely studied in the literature, and are almost no longer used in modern systems. For more information on these techniques, which are mainly based on phase-locked loops, the readers may refer to [MEY 98].

In the following, the transmitted signal is:

$$s(t) = \Re\left[s_l(t)e^{j2\pi f_c t}\right] = \Re\left[\sum_{k=-\infty}^{+\infty} c_k g(t-kT)e^{j2\pi f_c t}\right] \qquad [4.1]$$

where $s_l(t) = \sum_{k=-\infty}^{+\infty} c_k g(t-kT)$ is the equivalent passband signal. We generally assume that the complex baseband signals belong to a QAM modulation: $c_k = a_k + jb_k$.

$g(t)$ is the transmit filter, defined with its matched filter $g_r(t)$ so that the equivalent complete channel verifies the Nyquist criterion. The impulse response of the channel is $g_c(t)$. The equivalent channel is denoted as

$p(t) = g(t) * g_c(t) * g_r(t)$. An additive white Gaussian noise $\tilde{b}(t)$ is added to the channel. It is equal to the noise filtered by $g_r(t)$.

At the receiver input, the signal is:

$$r(t) = \Re\left[\sum_{k=-\infty}^{+\infty} c_k p(t - kT - \tau) e^{j2\pi(f_c+\delta f)t+\delta\Psi}\right] + \tilde{b}(t) \qquad [4.2]$$

We will note δf the frequency shift, $\delta\Psi$ the phase shift at the receiver and τ the time delay. If there is some Doppler effect, the frequency shift is due to the joint influence of oscillator's shift and Doppler effect, which depends on the relative speed between transmitter and receiver.

The equivalent baseband signal of [4.2] is:

$$y(t) = \sum_{k=-\infty}^{+\infty} c_k p(t - kT - \tau) e^{j2\pi\delta f t+\delta\Psi} + n(t) \qquad [4.3]$$

where $n(t) = \tilde{b}_l(t)$ is the complex envelope of the noise.

To simplify, in the following, we do not take into account $\delta\Psi$ which is assumed to be included in the channel's contribution p.

4.2.1. *Frequency shift correction*

We first assume that the symbol-period is known and that the time delay τ has been compensated for. The sampled impulse response of the channel $\mathbf{p}$ is assumed causal and of finite length L: $\mathbf{p} = [p_0, p_1, ..., p_{L-1}]^T$.

Once sampled at period T [4.3], the received signal is equal to:

$$\begin{aligned} y_k &= \sum_{m=0}^{L-1} p_m c_{k-m} e^{j2\pi\delta f kT} + n_k \\ &= e^{j\delta\omega k} \sum_{m=0}^{L-1} p_m c_{k-m} + n_k \end{aligned} \qquad [4.4]$$

where $y_k = y(kT)$, $b_k = b(kT)$.

We aim at estimating the carrier frequency shift $\delta\omega = 2\pi\delta f T$, and then estimating the impulse response of the channel. The receiver performs this estimation based on N received samples: $\mathbf{y} = [y_0, y_1, ..., y_{N-1}]^T$. Two cases can be distinguished: either a training sequence composed of a deterministic sequence of known symbols c_k is used, or synchronization and estimation are performed without any training sequence. They are then called blind synchronization and estimation.

4.2.1.1. *Data-aided correction of the frequency shift*

We first consider the most simple case where a training sequence is known by the receiver.

If the impulse response of the channel only has one coefficient, $\mathbf{p} = [p_0]$, using training sequence $c_k = 1,\ \forall k \in \{0, ..., N-1\}$, the received signal is

$$y_k = e^{j\delta\omega k} p_0 + n_k \tag{4.5}$$

$\mathbf{y}$ is consequently a sinusoid corrupted by noise $\mathbf{n} = [n_0, n_1, ..., n_{N-1}]^T$.

Its frequency can be obtained by estimating the power spectrum density of $\mathbf{y}$, which is approximated by the periodiogram (see equation [1.44]) of sequence $\mathbf{y} = [y_0, y_1, ..., y_{N-1}]^T$:

$$\widetilde{\gamma}_{yy}(\omega) = \frac{1}{NT}\left|\sum_{k=0}^{N-1} y_k e^{-j\omega k}\right|^2 \tag{4.6}$$

Assuming that the signal-to-noise ratio is large enough, the periodogram has a peak value in ω. Detection is consequently performed by identifying the pulse that maximizes the periodogram's modulus:

$$\widehat{\delta\omega} = \arg\max_{\omega}\left|\frac{1}{NT}\sum_{k=0}^{N-1} y_k e^{-j\omega k}\right|^2 \tag{4.7}$$

We can deduce $\widehat{\delta f} = \widehat{\delta\omega}/(2\pi T)$, and then the channel's coefficient p_0, by averaging over all samples once the frequency shift has been removed: $p_0 \approx \sum_{k=0}^{N-1} y_k e^{-j\widehat{\delta\omega}k}/N$.

We now generalize this method to any channel $\mathbf{p}$. The objective is to estimate $\delta\omega$ and $\mathbf{p}$ from the N received samples [4.4].

Equation [4.4] becomes:

$$y_k = e^{j\delta\omega k}\left[c_k, c_{k-1}, ..., c_{k-L+1}\right]\mathbf{p} + n_k \tag{4.8}$$

This can be equivalently written with matrix notations:

$$\mathbf{y} = \mathbf{\Gamma}(\delta\omega)\mathbf{C}\mathbf{p} + \mathbf{n} \tag{4.9}$$

where $\mathbf{\Gamma}(\omega) = \text{diag}\left[1, e^{j\delta\omega}, e^{j2\delta\omega}, ..., e^{j(N-1)\delta\omega}\right]$ is a diagonal matrix of size $N \times N$, and $\mathbf{C}$ is the $N \times L$ matrix defined as follows: $(\mathbf{C})_{k,l} = c_{k-l}$ for $k \in \{0, 1, ..., N-1\}$ and $l \in \{0, 1, ..., L-1\}$.

The distribution of $\mathbf{y}$ depends on the parameters ω and $\mathbf{p}$ that we aim at estimating. $\mathbf{C}$ contains the training sequence and is, therefore, known by the receiver. Since ω and $\mathbf{p}$ are deterministic, the distribution of $\mathbf{y}$ follows that of $\mathbf{b}$ and is thus Gaussian, with mean $\mathbf{\Gamma}(\delta\omega)\mathbf{C}\mathbf{p}$.

The maximum likelihood estimator determines the values of $(\omega, \mathbf{p})$ that maximize probability Pr $(\mathbf{y}|\omega, \mathbf{p})$:

$$\left(\widehat{\delta\omega}, \widehat{\mathbf{p}}\right) = \arg \max_{(\omega, \mathbf{p})} \Pr\left(\mathbf{y}|\omega, \mathbf{p}\right) \tag{4.10}$$

As the noise is Gaussian, the maximum likelihood is obtained when the distance between $\mathbf{y}$ and $\mathbf{\Gamma}(\omega)\mathbf{C}\mathbf{p}$ is minimum (see the demonstration in section 2.4.3.4).

$$\begin{aligned}\left(\widehat{\delta\omega}, \widehat{\mathbf{p}}\right) &= \arg \min_{(\omega, \mathbf{p})} \left\|\mathbf{y} - \mathbf{\Gamma}(\omega)\mathbf{C}\mathbf{p}\right\|^2 \\ &= \arg \min_{(\omega, \mathbf{p})} \left\|\mathbf{\Gamma}(\omega)^{-1}\mathbf{y} - \mathbf{C}\mathbf{p}\right\|^2 \end{aligned} \tag{4.11}$$

To simplify, we first estimate $\delta\omega$. The maximum likelihood estimator determines $\delta\omega$ such that $\mathbf{\Gamma}(\delta\omega)^{-1}\mathbf{y}$ is the closest to the vector space generated by the columns of matrix $\mathbf{C}$. It is denoted as $\text{Im}(\mathbf{C}) = \left\{\mathbf{C}\mathbf{x}, \mathbf{x} \in \mathbb{C}^L\right\}$.

The projection matrix on space $\mathrm{Im}(\mathbf{C})$ is denoted as $\mathbf{P}_C = \mathbf{C}\left(\mathbf{C}^H\mathbf{C}\right)^{-1}\mathbf{C}^H$. The maximum likelihood estimator of ω is equal to:

$$\widehat{\delta\omega} = \text{arg}\ \min_{\omega}\ \left\|\left(\mathbf{I} - \mathbf{P}_C\right)\mathbf{\Gamma}(\omega)^{-1}\mathbf{y}\right\|^2 \qquad [4.12]$$

where $\left\|\left(\mathbf{I} - \mathbf{P}_C\right)\mathbf{\Gamma}(\omega)^{-1}\mathbf{y}\right\|^2 = \left\|\mathbf{\Gamma}(\omega)^{-1}\mathbf{y}\right\|^2 - \left\|\mathbf{P}_C\mathbf{\Gamma}(\omega)^{-1}\mathbf{y}\right\|^2$ by Pythagoras's theorem.

Moreover, $\left\|\mathbf{\Gamma}(\omega)^{-1}\mathbf{y}\right\|^2 = \left\|\mathbf{y}\right\|^2$ and is independent of ω. Finally, by definition of $\mathbf{\Gamma}(\omega)$, we have $\mathbf{\Gamma}(\omega)^{-1} = \mathbf{\Gamma}(-\omega)$. As a result, the maximum likelihood criterion [4.12] is finally equivalent to:

$$\widehat{\delta\omega} = \text{arg}\ \max_{\omega}\ \left\|\mathbf{P}_C\mathbf{\Gamma}(-\omega)\mathbf{y}\right\|^2 \qquad [4.13]$$

Equation [4.13] is written as:

$$\widehat{\delta\omega} = \text{arg}\ \max_{\omega}\ \left|\sum_{k=-N+1}^{N-1} z_k^* e^{-j\omega k}\right|^2 \qquad [4.14]$$

where $z_k,\ k \in \{-N+1, ..., N-1\}$ is defined as a function of y_l and the projection matrix on space $\mathrm{Im}(\mathbf{C})$:

$$z_k = \sum_{l=\max\{0,k\}}^{\min\{N-1+k, N-1\}} \left(\mathbf{P}_C\right)_{l,l-k} y_l^* y_{l-k} \qquad [4.15]$$

We can note that equation [4.14] has the same shape as that obtained when the channel was limited to one coefficient [4.7]. $\widehat{\delta\omega}$ is similarly determined: it is equal to the pulse that maximizes the periodogram of $\mathbf{z}^*$.

We then estimate carrier frequency $\widehat{\delta f} = \widehat{\delta\omega}/(2\pi T)$. The channel $\mathbf{p}$ is evaluated by assuming that the noise can be neglected in equation [4.9]. Since in this case $\mathbf{C}\widehat{\mathbf{p}} = \mathbf{\Gamma}(-\widehat{\delta\omega})\mathbf{y}$, $\mathbf{p}$ is obtained by applying the pseudoinverse of $\mathbf{C}$, $\left(\mathbf{C}^H\mathbf{C}\right)^{-1}\mathbf{C}^H$:

$$\widehat{\mathbf{p}} = \left(\mathbf{C}^H\mathbf{C}\right)^{-1}\mathbf{C}^H\mathbf{\Gamma}(-\widehat{\delta\omega})\mathbf{y} \qquad [4.16]$$

Implementing this method requires us to evaluate all z_k, and then to compute their periodogram by applying the discrete inverse Fourier transform. The complexity of this method is consequently low. Nevertheless, its performances strongly depend on the number of considered samples N. Moreover, the periodogram's peak is often determined with low accuracy. It may then be necessary to refine the estimation of $\widehat{\delta\omega}$ by adding a second step of likelihood maximization with this peak as the initial value. Gradient descent techniques [BOY 04] may, for instance, be used.

The channel's influence is then evaluated. The relevance of the frequency shift estimator is quantified by computing the relative distance between the estimated and actual shifts, set in percentage. Table 4.1 summarizes this relative distance with only one channel coefficient $p_0 = 1$ and with a channel of length 3, $[p_0, p_1, p_2] = [1, 0.5, 0.125]$ for several values of E_b/N_0 arising from 1 to 20 dB. $N = 100$ samples are considered. The training sequence is binary. One conclusion of the simulation results is that the relative distance barely depends on E_b/N_0. Besides, it is very low with both channels.

	Averaged over all E_b/N_0	Minimum distance	Maximum distance
$p_0 = 1$	0.26%	0.14%	0.55%
$[p_0, p_1, p_2]$	0.37%	0.17%	0.91%

Table 4.1. *Relative distance between estimated and actual frequency shifts*

4.2.1.2. *Blind frequency shift correction*

Synchronization is blind if it is performed without any training sequence. This may happen when the training sequence's length is too high compared to the available monitoring time, for instance if the transmitter sends low-duration frames, or if the channel time variations are too high compared to the training sequence duration. The latter case corresponds to transmissions subject to the Doppler effect, if the relative speed between transmitter and receiver is large enough.

Blind synchronization can, however, also be chosen. Training sequences indeed do not transmit any useful data; their presence consequently decreases the spectral efficiency when considering the useful rate only. Nevertheless, if blind synchronization's performances are poor, the theoretical spectral efficiency gain may be compensated for by the rate loss due to the inter-symbol interference that the frequency shift generates.

Carrier estimation may be performed statistically based on the received samples, assuming that the statistical properties of the transmitted symbols are known.

Thus, we suppose that the transmitted sequence of symbols c_k is independent and identically distributed (i.i.d.), with zero mean $E[c_k] = 0$ and that they are non-circular at order P. Non-circularity implies that $E[c_k^P] \neq 0$.

These assumptions are consistent with linear digital modulation, which are always i.i.d. with zero mean. In addition, order-M amplitude shift keying (ASK) modulations are non-circular at order 2, since:

$$E[(c_{k,\text{ASK}})^2] = \sum_{m=1}^{M}(2m-1-M)^2A^2 = \frac{1}{3}M(M^2-1)A^2 \neq 0 \tag{4.17}$$

Similarly, regular QAM modulations are non-circular at order 4. M- PSK modulations are non-circular at order M:

$$E[(c_{k,\text{PSK}})^M] = \sum_{m=0}^{M-1}\left(Ae^{j2\pi\frac{m}{M}}\right)^M = \sum_{m=0}^{M-1}A^M e^{j2\pi m} = MA^M \neq 0 \tag{4.18}$$

The received samples are:

$$y_k = e^{j\delta\omega k}\sum_{m=0}^{M-1}p_m c_{k-m} + n_k \tag{4.19}$$

Let us denote as P the non-circularity order of symbols c_k. Then, y_k raised to the power of P is equal to:

$$y_k^P = e^{jP\omega k}\alpha + \widetilde{n}_k \tag{4.20}$$

where α neither depends on ω nor on the noise, and $\widetilde{n}_k$ is an additive non-Gaussian noise term.

The following frequency shift estimator is based on the non-circularity of the symbols [VIT 83]:

$$\widehat{\delta\omega} = \frac{1}{P}\arg \max_{\omega} \left| \sum_{k=0}^{N-1} y_k^P e^{-j\omega k} \right|^2 \qquad [4.21]$$

It is no longer optimum according to maximum likelihood criterion, as it was when a training sequence was used, because of the non-Gaussianity of the noise. Moreover, the estimated value of $\delta\omega$ necessarily lies within $[-\pi/P, \pi/P]$, with an ambiguity of π/P on the frequency shift.

Blind frequency shift estimator's performances are evaluated for a quadrature phase shift keying (QPSK) modulation, by computing the relative distance between the estimated and actual shifts, set in percentage. As in the previous case with the training sequence, two channels are considered: a simple channel $p_0 = 1$ and a channel of length 3, $[p_0, p_1, p_2] = [1, 0.5, 0.125]$. $N = 100$ unknown QPSK samples are randomly drawn. The possible frequency shifts are limited to interval $\delta\omega \in [-\pi/4, \pi/4]$. The relative distance for E_b/N_0 arising from 10 to 20 dB is in average equal to 0.5% when $p_0 = 1$, with a maximum of 1.2%. It is, however, far higher when the channel has 3 coefficients, with quite poor results on some sequences of samples. Comparing these results with those of Table 4.1 when a training sequence is used, we can deduce that blind synchronization performs poorly. In the studied case, it is consequently better to introduce a training sequence, even if this decreases the useful data's spectral efficiency. Other more complex blind estimation techniques, leading to better performances, can nevertheless be used [WAN 02, WAN 03, CAR 06].

4.2.2. *Time synchronization*

4.2.2.1. *Determination of the frame starting time*

In digital systems based on frame transmission, the first step of time synchronization consists of determining the beginning of each frame. Then, the time delay generated by the channel is corrected at the receiver.

Determination of the frame starting time is performed due to training sequences. For instance, in the global system for mobile communications (GSM) cellular system, a training sequence of 64 bits is sent on the synchronization channel (SCH) every 10 frames. Another training sequence is included in each data burst (sent every $577\mu s$) to refine synchronization and estimate the channel.

The training sequence contains N symbols $\mathbf{c}$ known at the receiver, which are assumed uncorrelated for high enough N:

$$\frac{1}{N}\sum_{m=0}^{N-1} c_{m+k}c_m^* \approx \begin{cases} 1 \text{ if } k = 0 \\ 0 \text{ if } k \neq 0 \end{cases} \qquad [4.22]$$

The receiver computes the correlation between the training sequence and the received data. It possesses a peak at the beginning of the training sequence. Knowing the frame's structure, the receiver can deduce the beginning of the frame. The training sequence is indeed not necessarily located at the beginning of the frame.

Other more efficient techniques are detailed for the specific case of OFDM in sections 5.3.6.1 and 5.3.6.2.

4.2.2.2. *Correction of time delay*

Time delay between transmitter and receiver can be estimated by implementing a rhythm error detector. This only provides an error signal; it is then necessary to add a feedback loop in order to deduce the time delay. If transmission is based on frames, feedback loops are not easy to implement since the number of studied symbols per frame is not enough for the feedback loop to converge. Consequently, direct techniques are more efficient.

Such a technique has been determined by Oerder and Meyr in [OER 88]. It provides an estimator with good performances that does not require any training sequence.

The time delay τ is assumed in $[-0.5T, 0.5T[$. We denote it as δT. We suppose that there is no frequency shift. The received signal, therefore, is:

$$y(t) = \sum_{k=-\infty}^{+\infty} c_k p(t - kT - \delta T) + n(t) \qquad [4.23]$$

Correction of the time delay is based on oversampling of the received signal at q times the symbol frequency, $F_e = q/T$. q must be higher than 2 in order to fulfill Shannon's sampling theorem. In practice, oversampling of factor $q = 4$ leads to an easier implementation. After oversampling, the signal is denoted as y_q. Its sample at time mT/q, with $m \in \mathbb{R}$, is equal to:

$$y_q\left(\frac{mT}{q}\right) = \sum_{k=-\infty}^{+\infty} c_k p\left(\frac{mT}{q} - kT - \delta T\right) + n\left(\frac{mT}{q}\right) \qquad [4.24]$$

Sequence $\left|y_q\left(\frac{mT}{q}\right)\right|^2$ contains samples oversampled by factor q and squared. It possesses a spectral component at frequency $1/T$. This component is extracted for each block of qL_r samples, where $L_r \in \mathbb{R}$, by computing the Fourier coefficients of $\left|y_q\left(\frac{mT}{q}\right)\right|^2$ at symbol rate:

$$Z_n = \sum_{m=nqL_r}^{(n+1)qL_r-1} \left|y_q\left(\frac{mT}{q}\right)\right|^2 e^{-j2\pi m/q} \qquad [4.25]$$

It has been shown in [OER 88][1] that time delay is obtained with the following estimator:

$$\widehat{\delta\delta T} = -\frac{1}{2\pi}\,\text{angle}\left(Z_n\right) \qquad [4.26]$$

where the function angle is defined as follows for complex number $c = |c|e^{j\Phi}$: angle$(c) = \Phi$.

The [4.26] estimator is unbiased if the complete channel filter verifies a specific constraint. In all cases, it provides δT with a given shift α:

$$E\left[\widehat{\delta T}\right] = \delta T + \alpha \qquad [4.27]$$

Moreover, the obtained value lies within $[-0.5T, 0.5T[$. Between two blocks of qL_r samples, δT estimates may vary strongly, especially at low signal-to-noise ratio. Some post-treatment on the noise may remove this issue. Finally, the successive estimations are smoothed due to the use of

[1] We do not detail the proof due to lack of space.

modulo 1 function, as indicated below:

$$\widetilde{\widetilde{\delta T}}_n = \widetilde{\widetilde{\delta T}}_{n-1} + \left[\widetilde{\widetilde{\delta T}}_n - \widetilde{\widetilde{\delta T}}_{n-1}\right]_{\text{mod } 1} \qquad [4.28]$$

4.2.3. *Channel estimate with training sequence*

We now assume that all synchronization steps have been performed, but that the channel has not been estimated yet. The channel is linear with length L and impulse response $[p_0, p_1, ..., p_{L-1}]^T$ after sampling at period T. In order to perform equalization, its coefficients must be estimated by the receiver.

The training sequence $[c_k]_{0 \leq k \leq N-1}$ is assumed uncorrelated (see equation [4.22]). The receiver then computes the correlation between $[y_0, y_1, ..., y_{N-1}]^T$ and $[c_0, c_1, ..., c_{N-1}]^T$:

$$\begin{aligned} z_k &= \sum_{m=0}^{N-1} y_{k+m} c_m^* \\ &= \sum_{m=0}^{N-1} \sum_{j=0}^{L-1} p_j c_{k+m-j} c_m^* + \sum_{m=0}^{N-1} n_{k+m} c_m^* \\ &= \sum_{j=0}^{L-1} p_j \left(\sum_{m=0}^{N-1} c_{k+m-j} c_m^* \right) + \chi_k \approx N p_k \end{aligned} \qquad [4.29]$$

since $\sum_{m=0}^{N-1} c_{k+m-j} c_m^* = N$ if $k = j$ and is equal to zero if $m \neq j$. We have assumed that the noise term χ_k can be neglected compared to $N p_k$.

Consequently, z_k provides an approximation of the channel's impulse response p_k.

Another technique consists of iteratively searching for the channel's impulse response. To achieve this, an initial vector of approximate channel coefficients is selected: $\widehat{\mathbf{p}}_0$. Then, vector $\widehat{\mathbf{y}}_0$ corresponding to samples at channel output, if the channel is equal to $\widehat{\mathbf{p}}_0$, is computed using the training sequence and neglecting the noise. Consequently, $\widehat{y}_{0,k} = \sum_{j=0}^{L-1} \widehat{p}_{0,j} c_{k-j}$.

This vector is then compared to the symbols $\mathbf{y}$ that have actually been received. The square error between the received symbol y_k and $\widehat{y}_{0,k}$ is equal to:

$$\begin{aligned} J_0 &= E\left[|y_k - \widehat{y}_{0,k}|^2\right] \\ &= (\mathbf{p} - \widehat{\mathbf{p}}_{\mathbf{0}})^\dagger\, E\left[c_k^* c_k^{'}\right] (\mathbf{p} - \widehat{\mathbf{p}}_{\mathbf{0}}) + \sigma^2 \\ &= |\mathbf{p} - \widehat{\mathbf{p}}_{\mathbf{0}}|^2 + \sigma^2 \end{aligned} \qquad [4.30]$$

since training symbols are uncorrelated, $E\left[c_k^* c_k^{'}\right]$ is equal to the identity matrix. σ^2 is the noise variance.

The estimate of vector $\mathbf{p}$ is iteratively obtained by applying an optimization algorithm whose objective is to minimize the square error $J_l = E\left[|y_k - \widehat{y}_{l,k}|^2\right]$ based on the knowledge of the training sequence and the received symbols. At iteration l, a new value of J_l and a new vector $\widehat{\mathbf{p}}_l$ are obtained from values J_{l-1} and $\widehat{\mathbf{p}}_{l-1}$ of the previous iteration. J_l should tend toward its minimum value, σ^2, when l tends to infinity. Convex optimization techniques such as gradient and stochastic gradient methods may be used [BEN 99].

4.2.4. *Cramer–Rao's bound*

The performances of the digital estimators used for synchronization can be quantified. Cramer–Rao's bound and its modified bound are statistical tools that allow us to evaluate if an estimator is of minimum variance.

4.2.4.1. *Fisher's information*

Let $\mathbf{y} = [y_0, y_1, ..., y_{N-1}]$ be a vector of random variables that depend on a scalar parameter θ. is defined as the information brought by vector $\mathbf{y}$ on parameter θ. It is denoted as $I_N(\theta)$ and is computed as follows:

$$I_N(\theta) = E\left[\left(\frac{\partial ln(p(\mathbf{y}|\theta))}{\partial\theta}\right)^2\right] \qquad [4.31]$$

where $p(\mathbf{y}|\theta)$ is the product of the conditional probability densities of variables $[y_n]_{0\leq n\leq N-1}$ with respect to parameter θ:

$$p(\mathbf{y}|\theta) = \prod_{n=0}^{N-1} p(y_n|\theta) \qquad [4.32]$$

If θ is not a scalar but a vector $\theta = [\theta_0, \theta_1, ..., \theta_{K-1}]$, becomes a matrix $\mathbf{J}(\theta)$ of size $K \times K$, defined as follows:

$$(\mathbf{J}(\theta))_{i,j} = E\left[\frac{\partial ln(p(\mathbf{y}|\theta))}{\partial\theta_i} \times \frac{\partial ln(p(\mathbf{y}|\theta))}{\partial\theta_j}\right] \qquad [4.33]$$

An unbiased estimator $\widehat{\theta_i}$ of θ_i is such that the estimator's expected value is equal to θ_i:

$$E\left[\widehat{\theta_i}(\mathbf{y})\right] = \theta_i \qquad [4.34]$$

It can be shown [VAN 68, SAP 90] that the variance of any unbiased estimator is lower bounded:

$$\text{var}\left[\theta_i - \widehat{\theta_i}(\mathbf{y})\right] \geq \left(\mathbf{J}^{-1}\right)_{i,i} \qquad [4.35]$$

Consequently, the variance of any estimator of θ_i necessarily is higher than the element of indices (i, i) of the inverse of Fisher's information matrix. $\left(\mathbf{J}^{-1}\right)_{i,i}$ is called Cramer–Rao's bound for the estimate of parameter θ_i. An estimator that reaches Cramer–Rao's bound is of minimum variance. It is called consistent. We should note that the bound cannot always be reached.

4.2.4.2. *Cramer–Rao's bound*

If only one parameter θ must be estimated, Cramer–Rao's bound [4.35] simplifies to:

$$\text{CRB}(\theta) = \frac{1}{E\left[\left(\frac{\partial ln(p(\mathbf{y}|\theta))}{\partial\theta}\right)^2\right]} \qquad [4.36]$$

From the definition of the baseband signal [4.3], assuming the initial phase Ψ is known and the noise can be neglected, the received signal is:

$$y(t) = \sum_{k=-\infty}^{+\infty} c_k p(t - kT - \tau) e^{j2\pi\delta f t + \delta\Psi} \qquad [4.37]$$

Cramer–Rao's bound contains too many unknown parameters (τ, δf and $\delta\Psi$) to be computable in general. Another bound, called the modified Cramer–Rao bound and proposed by d'Andrea, Mengali and Regiannini in [D'AN 94], is easier to evaluate. A random vector $\mathbf{u}$ is introduced. It contains all parameters of $y(t)$ other than the one we wish to estimate (denoted as θ) and its probability density, $p(\mathbf{u})$, is known. By including the conditional probability density $p(\mathbf{y}|\mathbf{u}, \theta)$, we get: $p(\mathbf{y}|\theta) = \int p(\mathbf{y}|\mathbf{u}, \theta)p(\mathbf{u})d\mathbf{u}$. The modified Cramer–Rao bound is then defined as:

$$\text{MCRB}(\theta) = \frac{1}{E\left[\left(\frac{\partial ln(p(\mathbf{y}|\mathbf{u},\theta))}{\partial\theta}\right)^2\right]} \qquad [4.38]$$

It is computed, respectively, for τ, δf and $\delta\Psi$, by setting one of these parameters to a given value and introducing the other parameters in the random vector, denoted as $\mathbf{u}$. For the frequency shift δf estimate with N samples and with a given E_s/N_0 ratio, the modified Cramer-Rao bound is obtained by considering vector $\mathbf{u} = (\delta\Psi, \tau, \mathbf{c})$. It is equal to [D'AN 94]:

$$\text{MCRB}(\delta f) = \frac{3T}{2\pi^2(NT)^3} \frac{1}{\left(\frac{E_s}{N_0}\right)} \qquad [4.39]$$

Similarly, for the phase shift:

$$\text{MCRB}(\delta\Psi) = \frac{1}{2N} \frac{1}{\left(\frac{E_s}{N_0}\right)} \qquad [4.40]$$

And finally, for the:

$$\text{MCRB}(\tau) = \frac{1}{8N\pi\xi} \frac{T^2}{\left(\frac{E_s}{N_0}\right)} \qquad [4.41]$$

where ξ depends on the complete channel filter:

$$\xi = \frac{\int_{-\infty}^{\infty} T^2 f_c^2 \left|P(f)\right|^2}{\int_{-\infty}^{\infty} \left|P(f)\right|^2} \qquad [4.42]$$

4.3. Equalization

In Chapter 2, we proved that in a band-limited channel with additive white Gaussian noise, inter-symbol interference can totally be suppressed if the complete filter verifies Nyquist criterion. The additive white Gaussian noise channel model studied in Chapter 2 is an ideal channel. Actual systems are, however, not ideal, due to the following reasons:

– the channel generates distortions. This case is detailed in the next section;

– synchronization is not perfect. The receiver may have wrongly estimated the sampling times, the frequency carrier or the phase.

In this section, we assume that synchronicity holds. The considered signal is the equivalent baseband signal. The transmit symbols are linearly modulated by ASK, PSK or QAM, and denoted as c_k. The transmission/reception chain is modeled in Figure 4.1. It contains a transmit filter $g(t)$, and then the channel with impulse response $g_c(t)$. It is subject to an additive white Gaussian noise of variance σ^2 per dimension. At the receiver, a receive filter $g_r(t)$ is applied. It is chosen as the matched filter to $g(t) * g_c(t)$. Distortions generated by $g_c(t)$ prevent the equivalent channel to fulfill Nyquist criterion, and lead to inter-symbol interferences.

Thus, if sampling is performed at the symbol-period and threshold detection is directly used at the output of the channel, the receiver's symbol estimates are highly likely to be erroneous, because they are based on interfered and noisy samples. In order to recover the transmitted symbols, an equalizer is added in the reception chain after sampling. It may be either a digital linear filter with transfer function $H_{eq}(z)$, as shown in Figure 4.1, or a nonlinear process.

The equalizer has one of the following objectives: remove inter-symbol interference while not taking into account the noise contribution; minimize the square error between transmitted and estimated symbols; jointly equalize a sequence of symbols instead of each symbol independently, with a maximizing likelihood criterion. Several techniques allow us to fulfill these objectives. Linear equalization, which uses linear filters, equalizes each symbol independently and is easy to implement, but not very efficient. Nonlinear equalizers may use a feedback loop or Viterbi's algorithm.

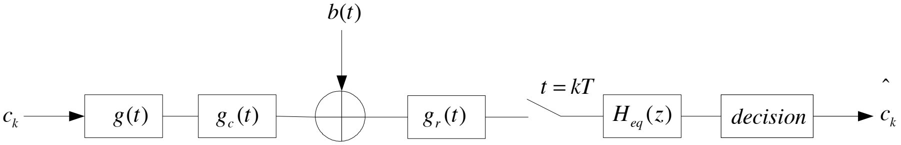

Figure 4.1. *Transmission/reception chain with equalization*

4.3.1. *Channel generating distortions*

Let $x(t)$ be the baseband transmit signal, composed of a sequence of modulated and shaped symbols: $x(t) = \sum_{k=-\infty}^{+\infty} c_k g(t - kT)$. The channel filters the transmit signal, resulting in signal $\sum_{k=-\infty}^{+\infty} c_k g(t - kT) * g_c(t)$.

The Fourier transform of the channel, $G_c(f)$, characterizes its frequency response. Thus, if $G_c(f) = 0$ when $f > B$, then the channel is band-limited. $G_c(f) = |G_c(f)| \, e^{j\psi(t)}$ may be separated in its modulus (that represents the amplitude response of the channel) and its phase (that represents the phase response of the channel). The channel generates amplitude and phase distortions on the signal if its modulus and phase are not constant in the bandwidth.

For example, the channels of wired assymetric digital subscriber line (ADSL) transmissions are wideband and generate distortions because attenuation depends on the frequency.

The wireless mobile channel used in cellular networks also generates distortions, due to multi-path transmissions of the original signal. Several paths may be created because of reflections on obstacles such as buildings, hills, etc. If a wireless mobile channel generates L paths, each one with attenuation α_n and delay τ_n, the channel's impulse response is: $g_c(t) = \sum_{n=0}^{L-1} \alpha_n \delta(t - \tau_n)$. If the maximum delay exceeds the symbol-period T, then the symbols transmitted at times kT and $(k+1)T$ interfere at the receiver.

4.3.2. *Discrete representation of a channel with inter-symbol interference and preprocessing*

The transmit equivalent filter and complete filter are, respectively, denoted as $h(t) = g(t) * g_c(t)$ and $p(t) = g(t) * g_c(t) * g_r(t)$. The receive filter is matched to the transmit equivalent filter: $g_r(t) = h^*(-t)$. The complete filter is then equal to: $p(t) = h(t) * h^*(-t)$. Its impulse response is $[p_0, p_1, ..., p_{L-1}]^T$. The modulated complex symbol transmitted at time kT is c_k. The received signal is:

$$y(t) = \sum_{k=-\infty}^{+\infty} c_k p(t - kT) + n(t) \qquad [4.43]$$

where $n(t) = b(t) * g_r(t)$ is the noise passed through the receive filter. After sampling at symbol-period T, the signal becomes:

$$\begin{aligned} y_k &= \sum_{m=0}^{L-1} p_m c_{k-m} + n_k \\ &= c_k p_0 + \sum_{m=1}^{L-1} p_m c_{k-m} + n_k \end{aligned} \qquad [4.44]$$

where $y_k = y(kT)$ is the sample at time kT.

When the Nyquist criterion is not fulfilled, equalization is applied at the receiver in order to estimate the transmit symbols, c_k, in spite of the inter-symbol interference term, $\sum_{m=1}^{L-1} p_m c_{k-m}$. This section aims at detailing the most usual equalization techniques and their performances. The following assumptions are made in the whole section: (1) synchronization at the receiver is perfect; (2) equalization is discrete, at symbol-period T; (3) the channel is assumed perfectly known at the receiver; and (4) a whitening filter is applied at the receiver. For more details on equalization techniques that do not rely on these four assumptions, the readers may refer to [BEN 99, BOT 11].

4.3.2.1. *Discrete matched whitening filter*

The discrete channel of equation [4.44] is characterized by its impulse response, $\mathbf{p} = [p_k]_{k=-\infty}^{\infty}$, which is defined as follows:

$$p_k = (h * g_r)(kT) = \int h(t)h^*(t - kT)dt \qquad [4.45]$$

We can easily show that $p_{-k} = p_k^*$:

$$\begin{aligned} p_{-k} &= \int h(t)h^*(t - kT)dt \\ &= \int h(u - kT)h^*(u)du \\ &= \left(\int h(u)h^*(u - kT)du\right)^* = p_k^* \end{aligned} \qquad [4.46]$$

The impulse response consequently has an odd number of coefficients that are symmetrical with respect to 0 if all coefficients are real, and antisymmetrical if they are complex. Let us assume that all coefficients are equal to zero beyond $k = L$. Then, the impulse response has non-zero coefficients between $-L + 1$ and $L - 1$ [2].

After having been filtered by the matched filter $g_r(t)$, the noise is no longer white.

The resulting noise samples $[n_k]$ belong to a centered and Gaussian random process, since it is equal to the white Gaussian noise b_k that has been linearly filtered, but it is colored. Its autocorrelation is equal to:

$$R_{nn}(k) = 2\sigma^2 p_k = N_0 p_k \qquad [4.47]$$

where σ^2 is the variance of noise $b(t)$. We can note that if the transmitted symbols, impulse response and noise are all real, then the autocorrelation is equal to $R_{nn}(k) = \sigma^2 p_k$ if p_k is different from zero.

[2] This implies that the filter $\mathbf{p}$ is not causal. In practice, it can be turned into a causal filter by including a delay of $L - 1$ samples on the decision step at the receiver.

Colored noise complexifies all receiver processes as well as performance evaluation. In particular, maximum likelihood equalization requires that the processed discrete data are all subject to white Gaussian noise. A solution to obtain such an equivalent channel consists of whitening noise after sampling.

In order to achieve this, let us note $P(z) = \sum_{k=-L+1}^{L-1} p_k z^{-k}$ the z transform of the complete filter. Since $p_{-k} = p_k^*$, $P(z) = P^*(1/z^*)$. $P(z)$ is a polynomial with $2L-1$ roots, such that if ρ is a root, then $1/\rho^*$ is also a root of $P(z)$. Thus, we can build a polynomial $F(z)$ only containing the $L-1$ roots ρ with modulus less than 1, and a polynomial $F^*(1/z^*)$ only containing the $L-1$ roots $1/\rho^*$ with modulus higher than 1. We assume that no root is located on the unit circle. The equivalent filter can then be decomposed into two filters:

$$P(z) = \lambda^2 F(z) F^*(1/z^*) \qquad [4.48]$$

where λ^2 is a constant that allows us to normalize filter $F(z)$: $\sum_{k=0}^{L-1} |f_k|^2 = 1$. Filter $1/\left(\lambda^2 F^*(1/z^*)\right)$ is finally chosen to whiten noise at the receiver. This filter is anticausal and stable. At its output, the noise is white.

After having applied $1/\left(\lambda^2 F^*(1/z^*)\right)$, the received signal becomes:

$$\begin{aligned} v_k &= \sum_{n=0}^{L-1} f_n c_{k-n} + u_k \\ &= f_0 c_k + \sum_{n=1}^{L-1} f_n c_{k-n} + u_k \end{aligned} \qquad [4.49]$$

where $\mathbf{f} = [f_k]_{0 \leq k \leq L-1}$ is the impulse response of $F(z)$, and u_k is the whitened noise. Moreover, we can note that since all the roots of $F(z)$ are inside the unit circle, it has a minimum phase. In addition, the whole system is now causal.

The complete channel with whitening filter and its equivalent model are represented in Figure 4.2.

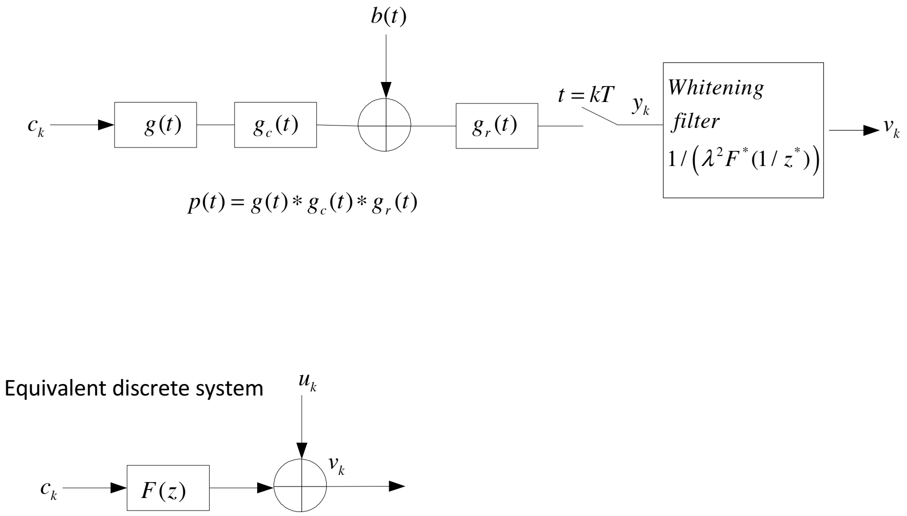
$b(t)$
c_k
$g(t)$
$g_c(t)$
$g_r(t)$
$t = kT$
y_k
Whitening filter
$1/\left(\lambda^2 F^*(1/z^*)\right)$
v_k
$p(t) = g(t) * g_c(t) * g_r(t)$
Equivalent discrete system
u_k
c_k
$F(z)$
v_k

Figure 4.2. *Whitening filter and equivalent transmit/receive chain*

4.3.3. *Linear equalization*

This section presents the easiest equalization technique, called linear equalization. Equalization is linear if it allows us to estimate each symbol independently of the others. It consists of including a digital filter at the end of the transmit/receive chain, after the sampling step. The linear filter has a finite impulse response (FIR) of length N. Its transfer function is:

$$H_{eq}(z) = \sum_{k=0}^{N} w_k z^{-k} \qquad [4.50]$$

In general, length N is imposed by complexity and delay constraints. Equalization consists of determining coefficients w_k for a known frequency response of the channel, and then updating these coefficients if the channel varies. The error probability cannot be minimum if linear equalizers are used. Two criteria exist: zero-forcing and minimum mean square error criteria. They lead to mitigation of inter-symbol interference, which may also avoid amplifying the noise to a large extent.

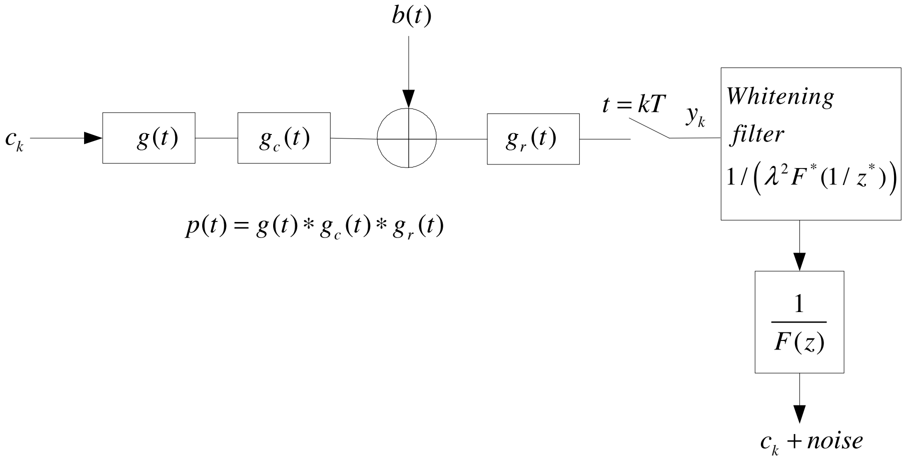
$b(t)$
c_k
$g(t)$
$g_c(t)$
$g_r(t)$
$t = kT$
y_k
Whitening filter $1/\left(\lambda^2 F^*(1/z^*)\right)$
$\frac{1}{F(z)}$
$c_k + noise$
$p(t) = g(t) * g_c(t) * g_r(t)$

Figure 4.3. *Zero-forcing equalization*

4.3.3.1. *Zero-forcing equalization*

Using the z transform, the samples entering the equalizer can be represented as follows:

$$C(z)F(z) + U(z) \qquad [4.51]$$

Since the transmitted symbols are $c(t) = \sum_{k=-\infty}^{+\infty} c_k \delta(t - kT)$, their z transform is $C(z) = \sum_{k=-\infty}^{+\infty} c_k z^{-k}$. Similarly, $U(Z)$ is the z transform of the noise samples, and $F(z)$ is the transfer function of the complete filter after noise has been whitened.

Zero-forcing equalization simply inverts the complete filter in order to remove any inter-symbol interference. The transfer function of the equalizer consequently is:

$$\widehat{H}_{\mathrm{ZF}}(z) = \frac{1}{F(z)} \qquad [4.52]$$

The signal at the output of the equalizer is thus: $C(z) + \widehat{H}_{\mathrm{ZF}}(z)U(z)$. At time kT, the output signal is: $c_k + \tilde{u}_k$, where $\tilde{u}_k$ is the noise filtered by the equalizer. Consequently, the denoised sample at time kT is equal to the transmitted symbol, c_k.

The total equalizer can be seen as the aggregation of the whitening filter and $\widehat{H}_{\mathrm{ZF}}(z)$. It is thus equal to:

$$H_{\mathrm{ZF}}(z) = \frac{1}{\lambda^2 F^*(1/z^*)F(z)} = \frac{1}{P(z)} \tag{4.53}$$

The zero-forcing filter can consequently be directly written as the inverse of the complete filter, which is applied on samples y_k (see equation [4.44]).

The zero-forcing filter minimizes the distortion generated by the channel by removing inter-symbol interference. Yet, since the channel's filter $P(z)$ is an FIR filter, $H_{\mathrm{ZF}}(z)$ is an infinite impulse response (IIR) filter that cannot be implemented in practice. Moreover, it is unstable since some of its poles may be outside of the unit circle. Consequently, an FIR approximation of this filter must be determined.

For this purpose, we must obtain the FIR filter $\widehat{H}_{eq}(z) = \sum_{k=0}^{N} w_k z^{-k}$ that minimizes the following distance:

$$\left| \frac{1}{F(z)} - \sum_{k=0}^{N} w_k z^{-k} \right|^2 \tag{4.54}$$

At this filter's output, some inter-symbol interferences will not have been removed because of this approximation. An example of FIR zero-forcing equalizer is provided at the end of this section.

Moreover, another major issue of zero-forcing equalizers is that they may highly increase the noise power spectrum density in some frequency bands. Since the signal-to-noise ratio then becomes very low, the error probability after threshold detection may become important.

The power spectrum density of the colored noise entering the complete equalizer $H_{eq}(z)$ is, according to equation [4.47]:

$$\gamma_{nn}(f) = N_0 P(f) \tag{4.55}$$

At the equalizer's output, the power spectrum density of noise $\tilde{U}(z) = H_{\text{ZF}}(z)N(z)$ is:

$$\begin{aligned}\gamma_{\tilde{u}\tilde{u}}(f) &= N_0 P(f)\left|H_{\text{ZF}}(f)\right|^2 \\ &= \frac{N_0}{P(f)}\end{aligned} \qquad [4.56]$$

In equation [4.56], the Fourier transform[3] has been substituted for the z transform, by setting $z = e^{j2\pi fT}$. As a result, if channel $P(f)$ is highly attenuated on one of the frequencies contained in its bandwidth, the noise power spectrum density will be highly increased, and the signal-to-noise ratio strongly decreased.

This issue is illustrated in Figure 4.4. A normalized noise has been filtered by zero-forcing equalization with the complete channel filter after noise whitening $F(z) = \frac{1}{1+\rho^2}\left(1 - 2\rho\cos(2\pi f_0/F_s)z^{-1} + \rho^2 z^{-2}\right)$, where $\rho = 0.98$. This filter strongly attenuates frequency f_0, equal to 1000 Hz in this example, since the sampling frequency is set to 4000 Hz. The noise at the equalizer's output is increased around 1000 Hz and its symmetrical frequency with respect to $F_s/2$, 3000 Hz. The signal-to-noise ratio at these frequencies is extremely poor, and detection on the corresponding symbols becomes almost unfeasible.

The signal-to-noise ratio at the output of the zero-forcing equalizer with infinite length is:

$$\text{SNR}_{\text{ZF},\infty} = \frac{E_s}{N_0}\frac{1}{T\int_{-0.5/T}^{0.5/T}\frac{1}{P(f)}df} \qquad [4.57]$$

where E_s is the energy of symbols c_k. In the absence of inter-symbol interference with a perfect channel ($P(f) = 1$), the signal-to-noise ratio is equal to E_s/N_0.

For FIR zero-forcing equalizers, the contribution of residual inter-symbol interference must be included in the noise.

[3] Please note that this equation holds whether the noise has been whitened or not.

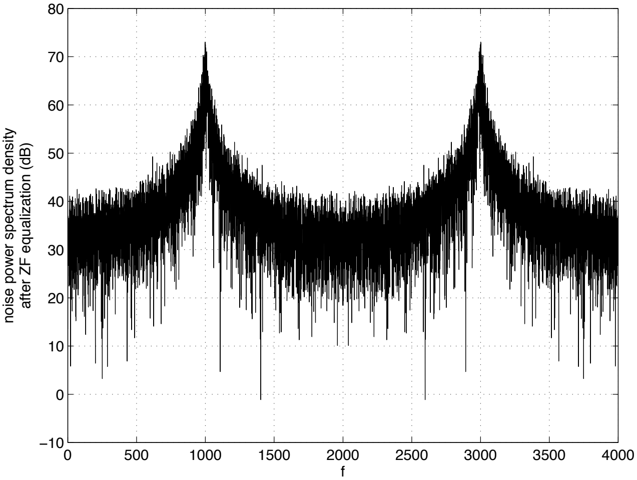

Figure 4.4. *Noise power spectrum density after zero-forcing equalization*

We now detail an example of zero-forcing equalization. Let us consider a transmit/receiver channel whose complete filter has the following transfer function: $F(z) = f_0 + f_1 z^{-1} = 1 + 0.5z^{-1}$. Binary symbols $c_k \in \{-1; 1\}$ are transmitted, thus $E_s = 1$. The signal is subject to an additive white Gaussian noise, with power spectrum density $N_0 = 0.1$. We assume that it has been whitened at the output of the matched filter.

We first determine the infinite-length zero-forcing equalizer. By definition, it is equal to[4] $\widehat{H}_{ZF}(z) = 1/F(z)$. Thus, it is equal to:

$$\widehat{H}_{ZF}(z) = \frac{1}{1 + 0.5z^{-1}} \qquad [4.58]$$

This is an IIR filter that cannot be implemented in practice, since it is equivalent to an FIR filter with infinite number of coefficients.

Consequently, it is necessary to compute the FIR zero-forcing equalizer with a finite number of coefficient. We choose length $N = 2$ in this example. The z transform of this equalizer is denoted as $\widehat{H}_{ZF}(z) = w_0 + w_1 z^{-1} +$

[4] In this example, we do not take into account the whitening filter's contribution. To simplify, filter $F(z)$ has not been normalized.

w_2z^{-2}. Since

$$\frac{1}{1+0.5z^{-1}} = \sum_{k=0}^{+\infty} \left(-0.5z^{-1}\right)^k \quad [4.59]$$

The length-2 filter that minimizes $\left|\frac{1}{F(z)} - \sum_{k=0}^{N} w_k z^{-k}\right|^2$ is obtained by taking the first three terms of [4.59]:

$$\widehat{H}_{ZF}(z) = 1 - 0.5z^{-1} + 0.25z^{-2} \quad [4.60]$$

We can note that another way of obtaining this filter is to aim at achieving:

$$F(z) \times \widehat{H}_{ZF}(z) = 1 \quad [4.61]$$

Equation [4.61] becomes, after having been developed:

$$\begin{aligned} &\left(1+0.5z^{-1}\right)\left(w_0 + w_1z^{-1} + w_2z^{-2}\right) = 1 \\ &\Leftrightarrow w_0 + (0.5w_0 + w_1)z^{-1} + (0.5w_1 + w_2)z^{-2} + 0.5w_2z^{-3} = 1 \end{aligned} \quad [4.62]$$

The z^{-3} term must be ignored for the system to be solvable. The previous system becomes:

$$\begin{cases} w_0 = 1 \\ w_1 = -0.5 \\ w_2 = 0.25 \end{cases}$$

We finally obtain the same equation [4.61] for the equalizer.

The total transfer function, $T(z) = F(z) \times \widehat{H}_{ZF}(z)$, is equal to :

$$T(z) = 1 + 0.125z^{-3} \quad [4.63]$$

In equation [4.63], the z^{-3} term corresponds to residual inter-symbol interference. It is due to the approximation of the filter by an FIR finite-length filter. Thus, each sample at the output of the channel is interfered by the sample that was sent 3 symbol-periods before.

In order to quantify the performances of the equalizer, we must compute the signal-to-noise ratio at the receiver. For this, we must express the powers of the noise, inter-symbol interference and the useful signals at the equalizer's output.

Let us start with the noise. After equalization, it has been filtered by the equalizer, and is consequently equal to: $\tilde{u}(t) = u(t) * \widehat{h}_{ZF}(t)$, where $u(t)$ is the additive white Gaussian noise prior to equalization. After sampling, the filtering equation becomes:

$$\begin{aligned}\tilde{u}_k &= \sum_{i=0}^{2} \widehat{h}_{ZF,i} u_{k-i} \\ &= u_k - 0.5u_{k-1} + 0.25u_{k-2} \qquad [4.64]\end{aligned}$$

By definition, the noise power is $E_{\tilde{u}} = E\left[|\tilde{u}_k|^2\right]$. It is expressed depending on the additive white Gaussian noise's power:

$$P_{\tilde{u}} = E\left[|u_k - 0.5u_{k-1} + 0.25u_{k-2}|^2\right] \qquad [4.65]$$

Since $\mathbf{u}$ is a white noise, the autocorrelation between any two different samples is equal to zero: $E\left[u_k u_{k-i}^*\right] = 0$ if $i \neq 0$. The noise power is consequently equal to:

$$\begin{aligned}P_{\tilde{u}} &= E\left[|u_k|^2\right] + 0.25E\left[|u_{k-1}|^2\right] + 0.0625E\left[|u_{k-2}|^2\right] \\ &= N_0 + 0.25N_0 + 0.0625N_0 = 0.13125 \qquad [4.66]\end{aligned}$$

Inter-symbol interference corresponds to the z^{-3} term. Its power is equal to:

$$P_{ISI} = E\left[|0.125c_{k-3}|^2\right] = 0.0156E\left[|c_{k-3}|^2\right] = 0.0156 \qquad [4.67]$$

Similarly, the useful signal's power at the output of the equalizer is:

$$P_s = E\left[|c_k|^2\right] = 1 \qquad [4.68]$$

Finally, the signal-to-noise ratio is:

$$SNR = \frac{P_s}{P_{ISI} + P_{\tilde{u}}} = \frac{1}{0.0156 + 0.13125} = 6.8097 \qquad [4.69]$$

It can be compared with the signal-to-noise ratio without equalization, which is equal to:

$$SNR = \frac{P_s}{P_{ISI} + P_u} = \frac{1}{0.5^2 + N_0} = 2.8571 \qquad [4.70]$$

Since in this case, inter-symbol interference comes from the z^{-1} term.

Consequently, zero-forcing equalization has increased the signal-to-noise ratio.

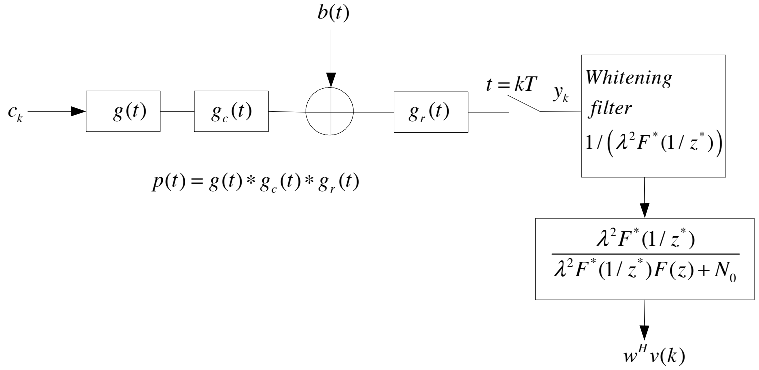
$b(t)$
c_k
$g(t)$
$g_c(t)$
$g_r(t)$
$t = kT$
y_k
Whitening filter
$1/\left(\lambda^2 F^*(1/z^*)\right)$
$p(t) = g(t) * g_c(t) * g_r(t)$
$\frac{\lambda^2 F^*(1/z^*)}{\lambda^2 F^*(1/z^*)F(z) + N_0}$
$w^H v(k)$

Figure 4.5. *Minimum mean square error equalizer*

4.3.3.2. *Minimum mean square error equalizer*

The second type of linear equalizers, called minimum mean-square error (MMSE), aims at minimizing the mean square error between any transmitted symbol c_k and its estimate at the output of the equalizer $\hat{c}_k$.

We consider the samples v_k at the output of the chain including the whitening filter, given by equation [4.49]. They enter into the filter $\hat{H}_{\text{MMSE}}(z) = \sum_{i=0}^{N} w_i z^{-i}$. The output samples are then estimated. A delay D

may be included. The estimated samples are:

$$\hat{c}_{k-D} = \sum_{i=0}^{N} w_i v_{k-i} = \mathbf{w}^H \mathbf{v}(k) \qquad [4.71]$$

where H stands for the Hermitian operator, which transposes the vector and conjugates all its elements. The complex conjugate vector of the filter's impulse response[5] is denoted as $\mathbf{w}^H = [w_0^*, w_1^*, ..., w_N^*]$. $\mathbf{v}(k) = [v_k, v_{k-1}, ..., v_{k-N}]^T$ is the vector of received samples. The mean square error is:

$$\begin{aligned} J &= E\left[\left|\mathbf{w}^H\mathbf{v}(k) - c_{k-D}\right|^2\right] \\ &= \mathbf{w}^H E\left[\mathbf{v}(k)\mathbf{v}(k)^H\right]\mathbf{w} - \mathbf{w}^H E\left[\mathbf{v}(k)c_{k-D}^*\right] \\ &\quad - E\left[c_{k-D}\mathbf{v}(k)^H\right]\mathbf{w} + E\left[\left|c_{k-D}\right|^2\right] \\ &= \mathbf{w}^H\mathbf{R}\mathbf{w} - \mathbf{r}^H\mathbf{w} - \mathbf{w}^H\mathbf{r} + E\left[\left|c_{k-D}\right|^2\right] \end{aligned} \qquad [4.72]$$

$\mathbf{R} = E\left[\mathbf{v}(k)\mathbf{v}(k)^H\right]$ is the autocorrelation matrix of the samples at the equalizer's input. $\mathbf{r} = E\left[\mathbf{v}(k)c_{k-D}^*\right]$ is the cross-correlation vector between samples at the input of the equalizer and the transmitted symbol c_{k-D}.

Mean square error is minimized when vector $\mathbf{w}$ sets the derivative of J with respect to $\mathbf{w}^* = [w_0, w_1, ..., w_N]^T$ to zero:

$$\frac{\partial J}{\partial \mathbf{w}^*} = -\mathbf{r} + \mathbf{R}\mathbf{w} \qquad [4.73]$$

where the following properties of differential calculus have been used [FIS 02]:

$$\begin{aligned} &\frac{\partial \mathbf{r}^H\mathbf{w}}{\partial \mathbf{w}^*} = 0 \\ &\frac{\partial \mathbf{w}^H\mathbf{r}}{\partial \mathbf{w}^*} = r \\ &\frac{\partial \mathbf{w}^H\mathbf{R}\mathbf{w}}{\partial \mathbf{w}^*} = \mathbf{R}\mathbf{w} \end{aligned} \qquad [4.74]$$

[5] Conjugates are chosen here in order to simplify the analytical derivation.

$\frac{\partial J}{\partial \mathbf{w}^*} = 0$ implies that:

$$\mathbf{w} = \mathbf{R}^{-1}\mathbf{r} \tag{4.75}$$

The minimum mean square error is then equal to:

$$\begin{aligned} J_{\text{min}} &= E\left[|c_{k-D}|^2\right] - \mathbf{r}^H\mathbf{R}\mathbf{r} \\ &= E_s - \mathbf{r}^H\mathbf{R}\mathbf{r} \end{aligned} \tag{4.76}$$

Moreover, it can be shown that if the equalizer has an infinite length [BEN 99, PRO 08], then the z transform of $\hat{H}_{\text{MMSE}}(z)$ is:

$$\hat{H}_{\text{MMSE}}(z) = \frac{\lambda^2 F^*(1/z^*)}{\lambda^2 F(z)F^*(1/z^*) + N_0/E_s} \tag{4.77}$$

The complete equalizer that applies to samples y_k (see equation [4.44]) is the product of the whitening filter, $1/\left(\lambda^2 F^*(1/z^*)\right)$, and $\hat{H}_{\text{MMSE}}(z)$. It is consequently equal to:

$$H_{\text{MMSE}}(z) = \frac{1}{\lambda^2 F(z)F^*(1/z^*) + N_0/E_s} = \frac{1}{P(z) + N_0/E_s} \tag{4.78}$$

The MMSE equalizer has a transfer function quite similar to that of the zero-forcing equalizer (see equation [4.52]), but it takes the noise into account. They become equal if there is no noise. The MMSE equalizer provides a trade-off between minimizing distortions and increasing the noise.

When the filter has an infinite length, the minimum mean square error J_{min} is expressed as a function of the channel as follows, after having set $z = e^{j2\pi ft}$ [BEN 99]:

$$J_{\text{min}} = T\int_{-0.5/T}^{0.5/T} \frac{N_0}{P(f) + N_0/E_s} df \tag{4.79}$$

The signal-to-noise ratio is deduced from the minimum mean square error:

$$\text{SNR} = \frac{E_s - J_{\text{min}}}{J_{\text{min}}} \tag{4.80}$$

We can note that this equation holds even if the equalizer is of finite length.

If there is no inter-symbol interference and if the channel is perfect ($P(f) = 1$), then the minimum mean square error is equal to $N_0/(1 + N_0/E_s)$ and the signal-to-noise ratio is E_s/N_0, and is thus the same as that achieved with zero-forcing in the same conditions. In the general case, however, MMSE equalizers lead to higher performances than zero-forcing equalizers.

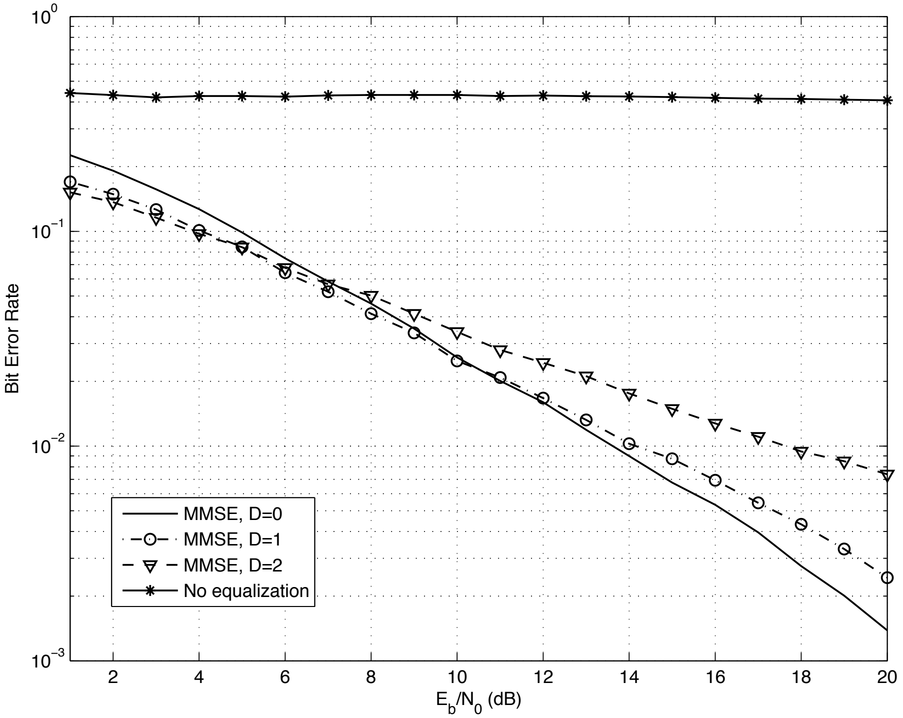

Figure 4.6. *Performances of MMSE equalizer when* $F(z) = 1 + 0.9z^{-1} + 0.7z^{-2}$ *for several values of* D

The influence of delay D is illustrated when the channel has transfer function $F(z) = 1 + 0.9z^{-1} + 0.7z^{-2}$ with an additive white Gaussian noise. It is represented in Figure 4.6. The MMSE equalizer has length $N = 3$. Figure 4.6 shows that better performances are achieved without any delay at high E_b/N_0.

We now detail an example of MMSE equalization. The same channel as in zero-forcing equalization is considered: $F(z) = f_0 + f_1 z^{-1} = 1 + 0.5z^{-1}$. Binary symbols $c_k \in \{-1; 1\}$ are transmitted, with $E_s = 1$. An additive white Gaussian noise u_k with power spectrum density $N_0 = 0.1$ is added on the signal. We assume that it is still white at the output of the receive filter.

We determine the MMSE equalizer with $N = 2$ and without any delay ($D = 0$).

The signal at the equalizer's input is:

$$v_k = c_k + 0.5c_{k-1} + u_k \tag{4.81}$$

The autocorrelation matrix of v_k is, by definition:

$$\mathbf{R} = \begin{pmatrix} R_{vv}(0) & R_{vv}(1) & R_{vv}(2) \\ R_{vv}(1) & R_{vv}(0) & R_{vv}(1) \\ R_{vv}(2) & R_{vv}(1) & R_{vv}(0) \end{pmatrix} \tag{4.82}$$

We must first evaluate the 3 autocorrelation coefficients. Since the input symbols c_k are uncorrelated and noise is white, autocorrelation in 0 is equal to:

$$\begin{aligned} R_{vv}(0) &= E\left[v_k {v_k}^*\right] \\ &= E\left[|c_k|^2\right] + 0.5^2 E\left[|c_{k-1}|^2\right] + E\left[|n_k|^2\right] \\ &= 1 + 0.5^2 + \sigma^2 = 1.35 \end{aligned} \tag{4.83}$$

Similarly,

$$\begin{aligned} R_{vv}(1) &= E\left[v_k {v_{k-1}}^*\right] \\ &= 0.5E\left[|c_{k-1}|^2\right] = 0.5 \end{aligned} \tag{4.84}$$

and:

$$R_{vv}(2) = E\left[v_k {v_{k-2}}^*\right] = 0 \tag{4.85}$$

Finally, the autocorrelation matrix is:

$$\mathbf{R} = \begin{pmatrix} 1.35 & 0.5 & 0 \\ 0.5 & 1.35 & 0.5 \\ 0 & 0.5 & 1.35 \end{pmatrix} \qquad [4.86]$$

This matrix is real symmetrical, which is consistent with theory.

The cross-correlation vector for delay $D = 0$ is equal to:

$$\mathbf{r} = E\left[\begin{pmatrix} v_k \times c_k \\ v_{k-1} \times c_k \\ v_{k-2} \times c_k \end{pmatrix}\right] = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \qquad [4.87]$$

Finally, after applying the MMSE equalizer's equation [4.75], its coefficients are:

$$\mathbf{w} = \mathbf{R}^{-1}\mathbf{r} = \begin{pmatrix} 0.8808 \\ -0.3781 \\ 0.14 \end{pmatrix} \qquad [4.88]$$

The transfer function of the equalizer thus is:

$$H_{MMSE}(z) = 0.8808 - 0.3781z^{-1} + 0.14z^{-2} \qquad [4.89]$$

The total transfer function, $T(z) = F(z) \times H_{MMSE}(z)$, is equal to:

$$T(z) = 0.8808 + 0.0623z^{-1} - 0.0491z^{-2} + 0.07z^{-3} \qquad [4.90]$$

where the z^{-1}, z^{-2} and z^{-3} terms correspond to residual inter-symbol interference. Consequently, each sample at the output of the channel is interfered by the 3 samples sent before it.

In order to compute the signal-to-noise ratio at the receiver, the noise power at the output of the equalizer is first evaluated. The noise after equalization is denoted as $\tilde{u}(t)$. Its samples are expressed as functions of those of the white noise before the equalizer, $u(t)$, as follows:

$$\begin{aligned}\tilde{u}_k &= \sum_{i=0}^{2} h_{MMSE,i} u_{k-i} \\ &= 0.8808u_k - 0.3781u_{k-1} + 0.14u_{k-2}\end{aligned} \qquad [4.91]$$

Since **u** is white, the noise power $P_{\widehat{u}} = E\left[|\widehat{u}_k|^2\right]$ is equal to:

$$\begin{aligned}P_{\tilde{u}} &= E\left[|\tilde{u}_k|^2\right] \\ &= E\left[|0.8808u_k - 0.3781u_{k-1} + 0.14u_{k-2}|^2\right] \\ &= 0.8808^2 E\left[|u_k|^2\right] + 0.3781^2 E\left[|u_{k-1}|^2\right] \\ &\quad + 0.14^2 E\left[|u_{k-2}|^2\right] = 0.09384\end{aligned} \qquad [4.92]$$

Inter-symbol interference corresponds to the z^{-1}, z^{-2} and z^{-3} terms. Inter-symbol interference's power is computed using the fact that the input symbols c_k have unitary power and are uncorrelated.

$$\begin{aligned}P_{ISI} &= E\left[|0.0623c_{k-1} - 0.0491c_{k-2} + 0.07c_{k-3}|^2\right] \\ &= 0.0623^2 + 0.0491^2 + 0.07^2 = 0.0112\end{aligned} \qquad [4.93]$$

Finally, the useful signal's power is:

$$P_s = E\left[|0.8808c_k|^2\right] = 0.8808^2 = 0.7758 \qquad [4.94]$$

The signal-to-noise ratio taking into account inter-symbol interference is finally expressed as follows:

$$SNR = \frac{P_s}{P_{ISI} + P_{\tilde{u}}} = \frac{0.7758}{0.0112 + 0.09384} = 7.3858 \qquad [4.95]$$

Equation [4.95] shows that the signal-to-noise ratio is higher than that achieved with zero-forcing criterion (see equation [4.69]) and of course than when no equalization is performed (see equation [4.70]), which is consistent with theory.

The signal-to-noise ratio can also be computed with formula [4.80]. The same result is obtained.

We now study the influence of the delay on the MMSE equalizer's performances. A delay $D = 1$ is added and the MMSE filter with this delay is determined.

The autocorrelation matrix M is unchanged. On the contrary, the cross-correlation vector with delay $D = 1$ is now equal to:

$$\mathbf{r} = E\left[\begin{pmatrix} v_k \times c_{k-1} \\ v_{k-1} \times c_{k-1} \\ v_{k-2} \times c_{k-1} \end{pmatrix}\right] = \begin{pmatrix} 0.5 \\ 1 \\ 0 \end{pmatrix} \qquad [4.96]$$

Then, the MMSE equalizer's equation is applied:

$$\mathbf{w} = \mathbf{R}^{-1}\mathbf{r} = \begin{pmatrix} 0.0623 \\ 0.8318 \\ -0.3081 \end{pmatrix} \qquad [4.97]$$

Finally, the MMSE equalizer with one sample's delay has the following transfer function:

$$H_{MMSE}(z) = 0.0623 + 0.8318z^{-1} - 0.3081z^{-2} \qquad [4.98]$$

The total transfer function, $T(z) = P(z) \times H_{MMSE}(z)$, is equal to:

$$T(z) = 0.0623 + 0.8621z^{-1} + 0.1078z^{-2} - 0.154z^{-3} \qquad [4.99]$$

The useful signal corresponds to the z^{-1} term; the remaining terms are residual inter-symbol interference.

By using the same method as when $D = 0$, the noise power is equal to:

$$P_{\tilde{u}} = E\left[|\widehat{u}|^2\right] = 0.0782 \qquad [4.100]$$

Inter-symbol interference corresponds to all terms except the z^{-1} term. Its power is:

$$P_{ISI} = E\left[\left|0.0623c_k + 0.1078c_{k-2} - 0.154c_{k-3}\right|^2\right]$$
$$= 0.0623^2 + 0.1078^2 + 0.154^2 = 0.0392 \qquad [4.101]$$

The useful signal's power is equal to the power of the symbols received at time $k-1$:

$$P_s = E\left[\left|0.8621c_{k-1}\right|^2\right] = 0.8621^2 = 0.7432 \qquad [4.102]$$

Finally, the signal-to-noise ratio is:

$$SNR = \frac{P_s}{P_{ISI} + P_{\widehat{u}}} = \frac{0.7432}{0.0782 + 0.0392} = 6.3306 \qquad [4.103]$$

Thus, setting a one-sample delay has decreased the signal-to-noise ratio after equalization. In this particular case, it is better not to include any delay.

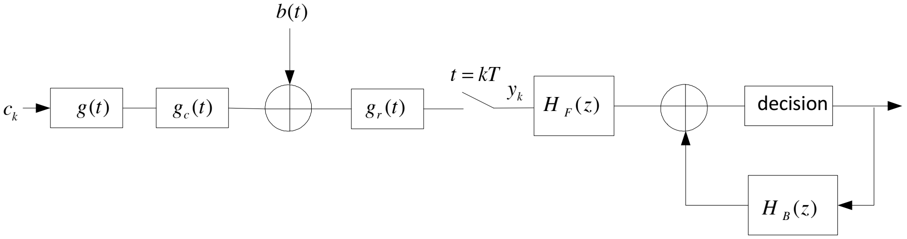

Figure 4.7. *Decision-feedback equalization*

4.3.4. *Decision-feedback equalization*

Zero-forcing and MMSE equalizers lead to poor performance results when inter-symbol interference is high or when the channel is highly frequency-selective. An alternative, which remains suboptimum but provides better performances, is to use a decision-feedback equalizer (DFE). These equalizers use symbols on which a decision has already been taken in order to estimate the next symbols. They are based on nonlinear processing with a feedback loop. The complete block diagram is represented in Figure 4.7. It is composed of two linear filters:

– a feedforward filter $H_F(z)$;

– a feedback filter $H_B(z)$.

The feedforward and feedback filters are, respectively, denoted as $H_F(z) = \sum_{i=-N_1}^{0} h_{F,i} z^{-i}$ and $H_B(z) = \sum_{j=1}^{N_2} h_{B,j} z^{-j}$. $H_F(z)$ is anticausal and stable [6] and $H_B(z)$ is an FIR causal filter. We denote by $\widetilde{c}_k$ the symbols after feedforward filtering and $\hat{c}_k$ the final symbols after equalization (see Figure 4.7). The DFE block diagram is applied on symbols y_k.

The samples at the output of the equalizer are equal to:

$$\hat{c}_k = \sum_{i=-N_1}^{0} h_{F,i} y_{k-i} + \sum_{j=1}^{N_2} h_{B,j} \widetilde{c}_{k-j} \qquad [4.104]$$

Feedforward and feedback filters are selected according to zero-forcing or MMSE criterion.

Zero-forcing decision feedback equalizer (ZF-DFE) aims at minimizing inter-symbol interference. It has been demonstrated in [PRI 72] that this equalizer is obtained by using the infinite-length whitening filter as feedforward filter: $H_F(z) = 1/\left(\lambda^2 F^*(1/z^*)\right)$. The finite-length feedback filter is then equal to: $H_B(z) = 1 - F(z)$. We have assumed here that filter $F(z)$ was normalized (which means that the sum of the squared modulus of its coefficients is equal to 1).

The objective of MMSE-DFE is to minimize the mean square error $E\left[|c_k - \hat{c}_k|^2\right]$. As for ZF-DFE, it is deduced from filter $F(z)$. Assuming that the previously-estimated symbols $\widetilde{c}_{k-i}$ are correct, the feedforward filter's coefficients of the MMSE-DFE are defined by the following set of equations [PRI 72, SAL 73]:

$$\sum_{j=-N_1}^{0} \Psi_{lj} h_{F,j} = f_{-l}^* \quad \forall l \in \{-N_1, ..., -1\} \qquad [4.105]$$

[6] In order to obtain a causal system, decision on samples will eventually be delayed by N_1 samples.

where Ψ_{lj} is defined as:

$$\Psi_{lj} = \sum_{m=0}^{-l} f_m^* f_{m+l-j} + N_0 \delta_{lj} \quad \forall (l,j) \in \{-N_1, ..., -1\}^2 \qquad [4.106]$$

with δ_{lj} the Kronecker's symbol, equal to 1 if $l = j$ and to 0 elsewhere.

Finally, the feedback filter's coefficients are deduced from those of the feedforward filter:

$$h_{B,k} = - \sum_{j=-N_1}^{0} h_{F,j} f_{k-j} \quad \forall k \in \{1, .., N_2\} \qquad [4.107]$$

MMSE-DFE completely removes inter-symbol interference on the detected symbols if N_2 is larger than the channel's impulse response length L, and if all previous decisions were correct.

If the length of the feedforward filter N_1 is infinite, the minimum mean square error J_{min} of MMSE-DFE is expressed as follows, by setting $z = e^{j2\pi ft}$ [SAL 73]:

$$J_{\text{min}} = \exp\left\{ T \int_{-0.5/T}^{0.5/T} \log\left(\frac{N_0}{P(f) + N_0/E_s} df \right) \right\} \qquad [4.108]$$

The signal-to-noise ratio depending on J_{min} is obtained with equation [4.80].

If there is no inter-symbol interference and $P(f) = 1$, the minimum mean square error is equal to $J_{\text{min}} = N_0/(1 + N_0/E_s)$ and the signal-to-noise ratio is E_s/N_0. MMSE-DFE is in this case equivalent to direct an MMSE equalizer with no feedback loop.

DFEs provide better performances than linear equalizers at high-enough signal-to-noise ratio. At low signal-to-noise ratio, some of the previous decisions are erroneous, which leads to errors in the next decisions. This error propagation phenomenon limits the performances of DFE, and consequently their practical use.

4.3.5. *Maximum likelihood sequence estimator*

4.3.5.1. *Principle*

A maximum likelihood equalizer, called a maximum likelihood sequence estimator (MLSE), directly estimates the sequences of transmitted symbols from received sequences of interfered and noisy symbols. Consequently, this type of equalization does not increase the noise power.

Equalization is no longer performed per symbol, but for a sequence of symbols. Viterbi's algorithm is used to maximize the probability that the estimated sequence is equal to the transmitted sequence.

At the input of MLSE, a whitening filter $1/\left(\lambda^2 F^*(1/z^*)\right)$ is used to whiten the noise. The equivalent channel before equalization consequently is $F(z)$. Let us assume that the channel's impulse response generates L inter-symbol interference terms. Its z transform is then: $F(z) = \sum_{n=0}^{L-1} f_n z^{-n}$. The transmit symbols, denoted as c_k, belong to an M-symbols modulation whose elements are located in set $\mathcal{C}$.

Let us assume that N interfered and noisy symbols $\mathbf{v} = [v_0, v_1, ..., v_{N-1}]$ are received. The noise samples, $\mathbf{u} = [u_0, u_1, ..., u_{N-1}]$, are i.i.d., white due to the whitening filter, and of variance $\sigma^2 = N_0/2$.

The probability to receive $\mathbf{v} = [v_0, v_1, ..., v_{N-1}]$ knowing that $\mathbf{c}$ was transmitted depends on the probability density of noise $u_k = v_k - \sum_{m=0}^{L-1} f_m c_{k-m}$ on each element k of the sequence. Since the noise samples are independent, it is equal to:

$$p(\mathbf{v}|\mathbf{c}) = \prod_{k=0}^{N-1} \frac{1}{\sqrt{\pi N_0}} \exp\left\{ -\frac{\left| v_k - \sum_{m=0}^{L-1} f_m c_{k-m} \right|^2}{N_0} \right\}$$

$$= \left(\frac{1}{\sqrt{\pi N_0}} \right)^N \exp\left\{ -\frac{1}{N_0} \sum_{k=0}^{N-1} \left| v_k - \sum_{m=0}^{L-1} f_m c_{k-m} \right|^2 \right\} \qquad [4.109]$$

The optimum estimated transmitted sequence according to the maximum likelihood criterion is consequently defined as follows:

$$\widehat{\mathbf{c}} = \arg \max_{\mathbf{c}\in\mathcal{C}^N} \left(\frac{1}{\sqrt{\pi N_0}}\right)^N \exp\left\{-\frac{1}{N_0}\sum_{k=0}^{N-1}\left|v_k - \sum_{m=0}^{L-1} f_m c_{k-m}\right|^2\right\} \qquad [4.110]$$

To simplify the previous objective, it is equivalent and preferable to study the log-likelihood:

$$\begin{aligned}\widehat{\mathbf{c}} &= \arg \max_{\mathbf{c}\in\mathcal{C}^N} \left(-\sum_{k=0}^{N-1}\left|v_k - \sum_{m=0}^{L-1} f_m c_{k-m}\right|^2\right)\\ &= \arg \min_{\mathbf{c}\in\mathcal{C}^N} \sum_{k=0}^{N-1}\left|v_k - \sum_{m=0}^{L-1} f_m c_{k-m}\right|^2 \qquad [4.111]\end{aligned}$$

We can note that for any given sequence $\mathbf{c}$, $d_k = \sum_{m=0}^{L-1} f_m c_{k-m}$ depends on $[c_k, c_{k-1}, ..., c_{k-L+1}]$. The received signal at time $k+1$, d_{k+1} depends on $[c_{k+1}, c_k, c_{k-1} ..., c_{k-L+2}]$. Consequently, two consecutive received signals jointly share L symbols.

Since c_{k+1} belongs to an M symbols modulation, there are M possible paths to pass from d_k to d_{k+1}. In order to determine c_{k+1}, a trellis with M input paths and M output paths can be built. This trellis will contain M^L states since L symbols that can take M values are stored in shift registers. The transmit symbols sequence can then be obtained by applying Viterbi's algorithm on this trellis. Looking for the most likely sequence is equivalent to determining the most likely path in the trellis, knowing the received sequence symbols and the channel's impulse response.

Thus, Viterbi's algorithm must be applied on the received symbols using the squared Euclidean distance for branches metrics, $\left|v_k - \sum_{m=0}^{L-1} f_m c_{k-m}\right|^2$. To simplify the computation, the absolute value metric is often used: $\left|v_k - \sum_{m=0}^{L-1} f_m c_{k-m}\right|$ instead of the squared Euclidean distance. Details of Viterbi's algorithm were presented in the first volume of this book [LER 15]. They are applied to equalization in the next section with an example.

4.3.5.2. *Example*

We now detail an example of maximum likelihood sequence equalization. The considered channel has the following complete filter: $F(z) = f_0 + f_1 z^{-1} + f_2 z^{-2} = 1 + 0.5z^{-1} + 0.25z^{-2}$. At the channel's input, binary symbols $c_k \in \{-1; 1\}$ are transmitted. The noise at the filter's output is white and Gaussian.

The channel is represented with shift registers in Figure 4.8.

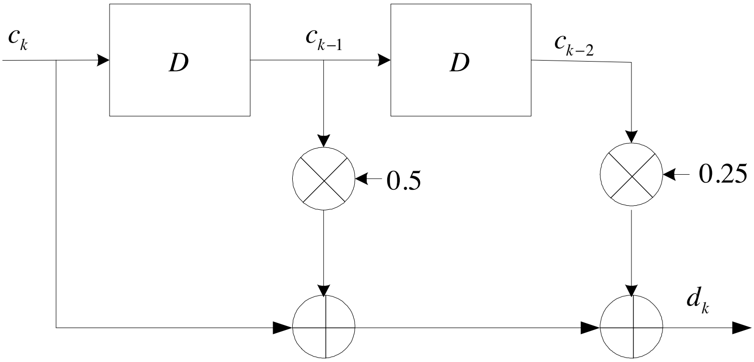

Figure 4.8. *Channel with shift registers*

Input symbols are related to output symbols, denoted as d_k, by the following equation:

$$d_k = \sum_{n=0}^{2} f_n c_{k-n} = c_k + 0.5c_{k-1} + 0.25c_{k-2} \qquad [4.112]$$

The first step consists of detailing the internal states of the trellis. There are $M = 2$ symbols and the number of shift registers is $L = 2$. Consequently, the number of internal states of the trellis is $M^L = 2^2 = 4$. The four internal states are denoted as a, b, c and d.

Table 4.2, describing internal states, contains the values stored in each shift register.

Internal state	c_{k-1}	c_{k-2}
a	-1	-1
b	-1	1
c	1	-1
d	1	1

Table 4.2. *Table describing internal states*

In a second step, the transitions between states depending on the input bit c_k must be determined.

For instance, let us assume that the system is in state a and that $c_k = 1$. At the next clock, $c_{k-1} = 1$ and $c_{k-2} = -1$. The next state is then c. Other transitions are similarly deduced.

The different states are then connected in the trellis according to the following rule:

– by a full line if the input bit is $c_k = 1$;

– by a dotted line if the input bit is $c_k = -1$.

To complete the trellis, branch labels must be determined. They are equal to the output value for each branch of the elementary trellis. The output value is the value of d_k when a transition takes place from one state to another. For instance, if we consider transition from a to a: state a indicates that $c_{k-1} = -1$ and $c_{k-2} = -1$. Transition to a means that $c_k = -1$. The output value of the noiseless filter thus is: $d_k = -1 - 0.5 - 0.25 = -1.75$.

The elementary trellis with branch labels is given in Figure 4.9.

Viterbi's algorithm is now used to decode a received interfered and noisy sequence. Let us assume that the received sequence with 4 symbols is:

$$\mathbf{y} = [0.25, 0.25, -0.5, 0.5]$$

where $v_k = d_k + u_k$ and $\mathbf{u}$ is an additive white Gaussian noise. We must find the sequence of the most likely 4 symbols c_k. The branch metric used in Viterbi's algorithm is the absolute value: $|v_k - d_k|$.

The initial state is assumed to be state a = $\{-1, -1\}$.

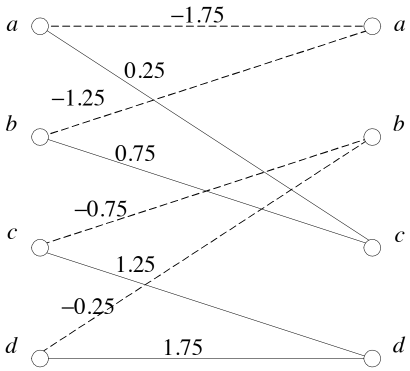

Figure 4.9. *Complete trellis*

Viterbi's algorithm starts with the computation of the branch metric $|v_k - d_k|$ for all branches. For instance, for the first symbol:

– a to a branch: $|v_k - d_k| = |0.25 + 1.75| = 2$;

– a to c branch: $|v_k - d_k| = |0.25 - 0.25| = 0$.

The cumulated metric in a is 2, and the cumulated metric in c is 0 (see Figure 4.10).

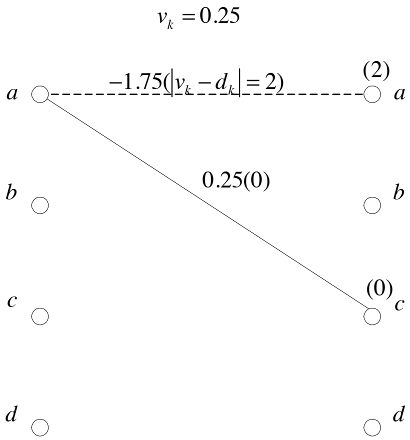

Figure 4.10. *Viterbi's algorithm for MLSE, first step*

When two paths lead to the same state, only the path with the lowest cumulated metric is kept. This is used in the following steps.

Finally, the path with the lowest cumulated metric ends in state c, with metric 1.5 (see Figure 4.11).

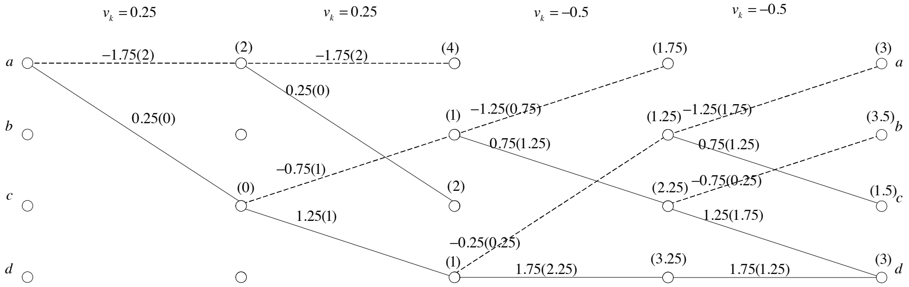

Figure 4.11. *Viterbi's algorithm for MLSE, at the end of the sequence*

In order to recover the most likely symbols c_k, we must go back up along the best path, using the type of transition that is represented by the nature of the line connecting two states. In this example, as Figure 4.12 shows, the most likely sequence is $\widehat{\mathbf{c}} = [1, 1, -1, 1]$.

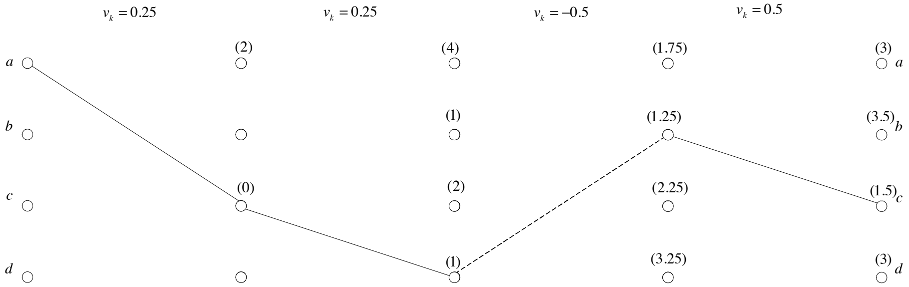

Figure 4.12. *Viterbi's algorithm, most likely sequence*

Performances of zero-forcing, MMSE equalizers and MLSE are compared in Figure 4.13 on a channel with transfer function $F(z) = 1+0.5z^{-1}+0.25z^{-2}$ with an additive white Gaussian noise. MMSE equalizer is chosen without any delay ($D = 0$) and both linear equalizers have length $N = 3$. The performance gain between MMSE and zero forcing (ZF) is very low, but almost 2 dB are gained when using MLSE compared to MMSE when the bit error rate is equal to 10^{-4}. These results of course depend on the studied channel.

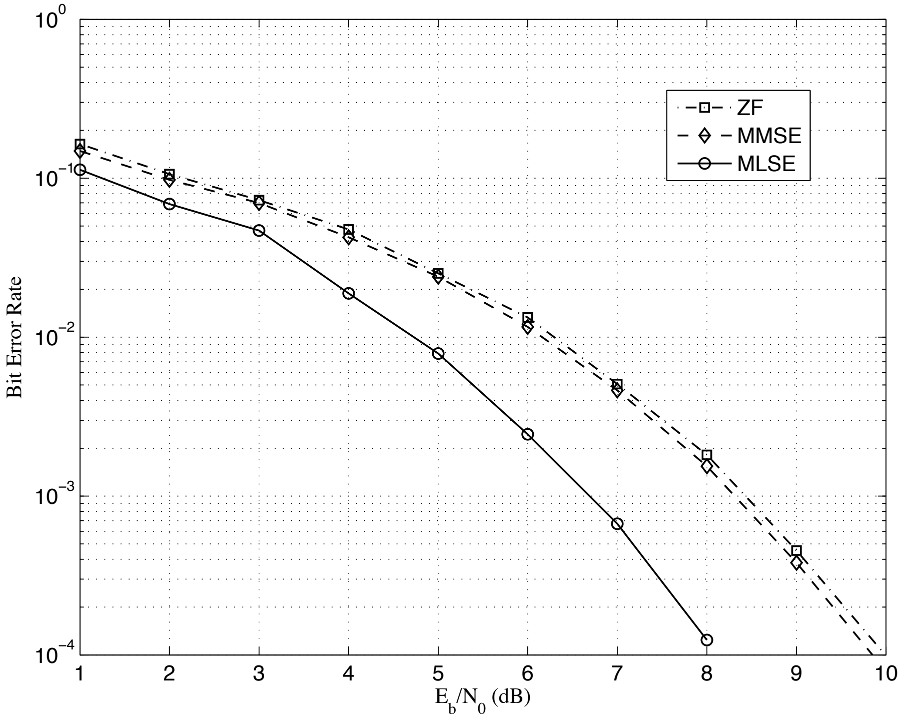

Figure 4.13. *Performances of ZF, MMSE and MLSE equalizers for channel* $F(z) = 1 + 0.5z^{-1} + 0.25z^{-2}$

4.4. Conclusion

This chapter first presented several techniques to correct the frequency shifts and time delays introduced by the channel at the receiver's level. The various synchronization techniques have been studied on the equivalent baseband channel without any specific assumption on the channel. Synchronization may be blind: it then corrects shifts with the sole knowledge of the channel's statistical properties. It may also use training sequences, known by the receiver. It has been shown for frequency shift correction that synchronization with a training sequence provides better results than blind synchronization, even if this implies a loss in the useful signal's spectral efficiency.

Once synchronization has been acquired, if some inter-symbol interference has been generated by the channel that does not fulfill Nyquist criterion, the receiver must equalize the channel in order to estimate data as accurately as possible. Different types of equalizers have been studied, from the simplest to the most complex: linear equalizers, which are simple but lead to poor performances, DFEs, which are based on linear equalizers but can improve their performances due to a feedback loop and finally MLSE, which aims at maximizing the likelihood of a sequence of symbols. Since inter-symbol interference is a critical issue in systems, we should note that in practice MLSE criterion is preferred if the number of states of the trellis is reasonable, for instance for a binary modulation. Finally, modern techniques of equalizations based on a turbo principle exist: for more information on turbo-equalization, the readers can refer to the introductive paper of [KOE 04], and to [TÜC 11] for a recent tutorial.

4.5. Exercises

4.5.1. *Exercise 1: estimation of a constant signal from noisy observations*

A receiver aims at estimating a constant signal A transmitted on the additive white Gaussian noise with N samples. The received signals are $y_k = A + n_k$, with $k \in \{0, ..., N-1\}$ and $[n_k]_{k=0}^{N-1}$ are noise samples, which are centered, Gaussian, independent and have the same variance σ^2.

We propose to estimate A with the following estimator:

$$A = \frac{1}{N}\sum_{k=1}^{N} y_k \qquad [4.113]$$

1) Prove that this estimator is unbiased.

2) Prove that this estimator is consistent.

NOTE.– This exercise is extracted from [TER 13].

4.5.2. *Exercise 2: frequency shift correction*

In a radio transmission similar to GSM, a logical channel is available to perform frequency synchronization. It is composed of a block of 64 QPSK symbols known by the receiver. We will assume that the training sequence is composed of symbols that are all equal to $1 + j$. The signal-to-noise ratio at the receiver is $E_s/N_0 = 20$ dB for these symbols.

The data rate is equal to 10 ksymbols/s. At the receiver, a frequency shift must be corrected.

1) What is the influence of the frequency shift on the received symbols?

2) Propose an algorithm to estimate the frequency shift δf.

3) We assume that an unbiased estimator of the frequency shift is available. Compute Cramer–Rao's bound for this estimator.

4) This bound on the minimum variance of the estimator is too high. Which solution would be the most efficient to decrease it, between: (1) doubling the number of symbols of the training sequence or (2) doubling the training sequence's power?

NOTE.– This exercise is extracted from [TER 13].

4.5.3. *Exercise 3: zero-forcing equalization*

We consider a transmit channel whose total transfer function, including the noise whitening filter, is $F(z) = 1 + 0.8z^{-1} + 0.6z^{-2}$. The white Gaussian noise has power spectrum density $N_0 = 0.2$. The data $c_k \in \{-1; 1\}$ applied at the channel's input are uncorrelated. The symbol-period is denoted as T.

1) Represent the transmit/receive chain. What is the transfer function of the infinite-length ZF equalizer?

2) Compute the transfer function of ZF equalizer $\widehat{H}_{\text{ZF}}$ with finite length $N = 3$.

3) Give the expression of the total transfer function $T(z) = F(z) \times \widehat{H}_{\text{ZF}}(z)$.

4) Compute the noise power at the output of the equalizer.

5) Compute the inter-symbol interference power at the output of the equalizer.

6) Deduce the signal-to-noise ratio at the output of the equalizer.

7) Compare with the signal-to-noise ratio without equalization.

4.5.4. *Exercise 4: MMSE equalization*

We consider a transmit channel whose total transfer function, including the noise whitening filter, is $F(z) = 1 + 0.8z^{-1} + 0.6z^{-2}$. The white Gaussian noise has power spectrum density $N_0 = 0.2$. The data $c_k \in \{-1; 1\}$ applied at the channel's input are uncorrelated. The symbol-period is denoted as T. An MMSE equalizer of length $N = 3$ is used.

1) Recall how MMSE filter coefficients are obtained.

2) Compute the autocorrelation matrix of size 4×4 of the signal at equalizer's input.

3) The inverse of the autocorrelation matrix is equal to:

$$\mathbf{R}^{-1} = \begin{pmatrix} 0.7167 & -0.4458 & -0.0103 & 0.1276 \\ -0.4458 & 0.9713 & -0.4375 & -0.0103 \\ -0.0103 & -0.4375 & 0.9713 & -0.4458 \\ 0.1276 & -0.0103 & -0.4458 & -0.4458 \end{pmatrix} \qquad [4.114]$$

Compute the transfer function of the equalizer $\widehat{H}_{\text{MMSE}}(z)$ that minimizes the mean square error with no delay ($D = 0$).

4) Give the expression of the total transfer function $T(z) = F(z) \times \widehat{H}_{\text{MMSE}}(z)$.

5) Compute the noise power at the output of the equalizer.

6) Compute the inter-symbol interference power at the output of the equalizer.

7) Deduce the signal-to-noise ratio at the output of the equalizer.

8) Compare with the signal-to-noise ratio of the ZF equalizer and without equalization.

4.5.5. *Exercise 5: MMSE-DFE equalization*

We consider a transmit channel whose complete filter (assumed such that the noise at the output of the channel is white) is $F(z) = \frac{1}{1.25}\left(1 + 0.5z^{-1}\right)$. The data applied at the channel's input $c_k \in \{-1; 1\}$ are uncorrelated. The symbol-period is denoted as T.

Compute the equations of the feedforward and feedback filters of the MMSE-DFE equalizer, taking filter lengths $N_1 = 1$ and $N_2 = 1$ for the feedforward and feedback filters, respectively.

4.5.6. *Exercise 6: MLSE equalization with one shift register*

We consider a transmit channel whose complete transfer function is $F(z) = 1 + 0.8z^{-1}$.

The data applied at the channel's input are $c_k \in \{-5; 5\}$. We wish to decode a received sequence with MLSE criterion. We assume that a whitening filter has been used at the equalizer's input, and that it is included in $F(z)$ transfer function.

1) Represent the channel model with shift registers corresponding to the transfer function.

2) Express the output bit d_k without noise depending on the input bits.

3) Represent the elementary trellis associated with the channel.

4) The receiver's input v_k is the sum of d_k and a Gaussian noise u_k: $v_k = d_k + u_k$. The following sequence is received: $\mathbf{v} = [5, 2, -8, -2, 3, 14, -1]$ We assume that the initial state is $c_0 = -5$. Find the most likely transmitted sequence by using Viterbi's algorithm on the trellis.

To simplify the branches metrics, use the absolute value metric: $|v_k - d_k|$.

4.5.7. *Exercise 7: MLSE equalization with two shift registers*

We consider the transmit channel represented by the following figure:

where $f_0 = 0.407$, $f_1 = 0.815$ and $f_2 = 0.407$.

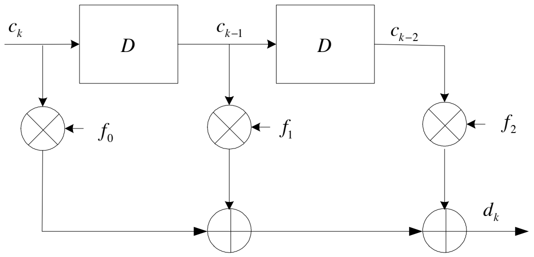

Figure 4.14. *Exercise 7*

The data applied at the channel's input are $c_k \in \{-1; 1\}$. We wish to decode a received sequence with MLSE criterion.

1) Determine the transfer function of the channel.

2) Express the output bit d_k without noise depending on the input bits.

3) Determine the number of internal states of the trellis.

4) Represent the elementary trellis associated with the channel.

5) The receiver's input v_k is the sum of d_k and a Gaussian noise u_k: $v_k = d_k + u_k$. The following sequence is received: $\mathbf{v} = [0.7, 0, 0.5]$ We assume that the initial state is $\{-1, -1\}$. Find the most likely transmitted sequence by using Viterbi's algorithm on the trellis, using the absolute value metric.

5

Multi-carrier Modulations

5.1. Introduction

As already mentioned in Chapter 4, the channel may generate distortions on the received symbols. The channel is frequency-selective if its frequency response is uncorrelated on the considered bandwidth [PRO 08, GOL 05]. The coherence band B_c is defined as the frequency band on which two frequency components of the channel can be assumed uncorrelated.

The frequency-selectivity of a channel is equivalently characterized by one of the following definitions: either the maximum delay of the channel's impulse response T_m is large compared to the symbol period T, or the channel's coherence band is small compared to its bandwidth B. If the channel is frequency-selective, it generates inter-symbol interference at the receiver. On the contrary, we assume that a channel is non-frequency-selective with an almost flat frequency response in its bandwidth if $B \leq 0.1 B_c$.

The maximum delay of the channel's impulse response and the channel's coherence band are related as follows:

$$B_c \approx \frac{1}{T_m}$$

An example of frequency-selective channel is given in Figure 5.1. It represents the modulus of the frequency response of channel g_c with impulse

Digital Communications 2: Digital Modulations, First Edition. Mylène Pischella and Didier Le Ruyet.

response:

$$\{g_c[0] = 1, g_c[1] = 0, g_c[2] = 0, g_c[3] = 0.7, g_c[4]$$
$$= 0, g_c[5] = 0.5, g_c[6] = 0.3\}$$

in baseband and with bandwidth $B = 1$ MHz. This channel is frequency-selective, since its maximum delay $T_m = 6 \times T_s = 6\ \mu s$. Consequently, the signal received on some of the frequencies may be strongly attenuated.

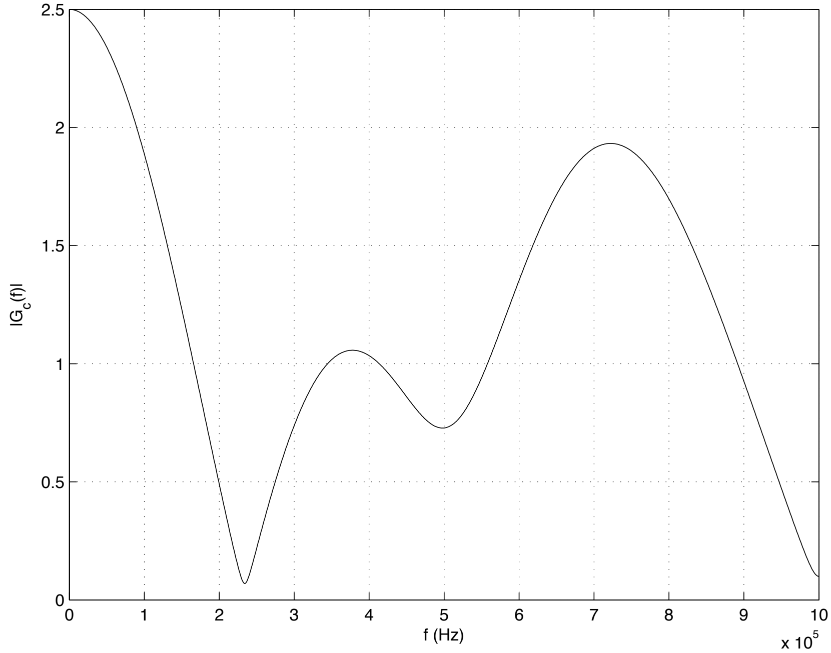

Figure 5.1. *Frequency-selective channel G_c*

A channel is non-time-selective if its frequency response remains the same during the whole symbol period. If the channel is frequency-selective, but non-time-selective, the receiver must add an equalization step in order to detect the transmitted symbols in spite of inter-symbol interference. In order to avoid channel equalization, the channel may be separated into several sub-bands, each with a bandwidth far lower than the coherence band. In this case, there is no inter-symbol interference in each subchannel. This principle is the rationale for multi-carrier modulations.

In the example of Figure 5.1, the total bandwidth has been split into 64 sub-bands with the same bandwidth in order to obtain non-frequency-selective sub-bands, as illustrated in Figure 5.2. We can notice that in most multi-carrier modulations, and especially in the ones considered in this book, all sub-bands are always chosen with the same bandwidth.

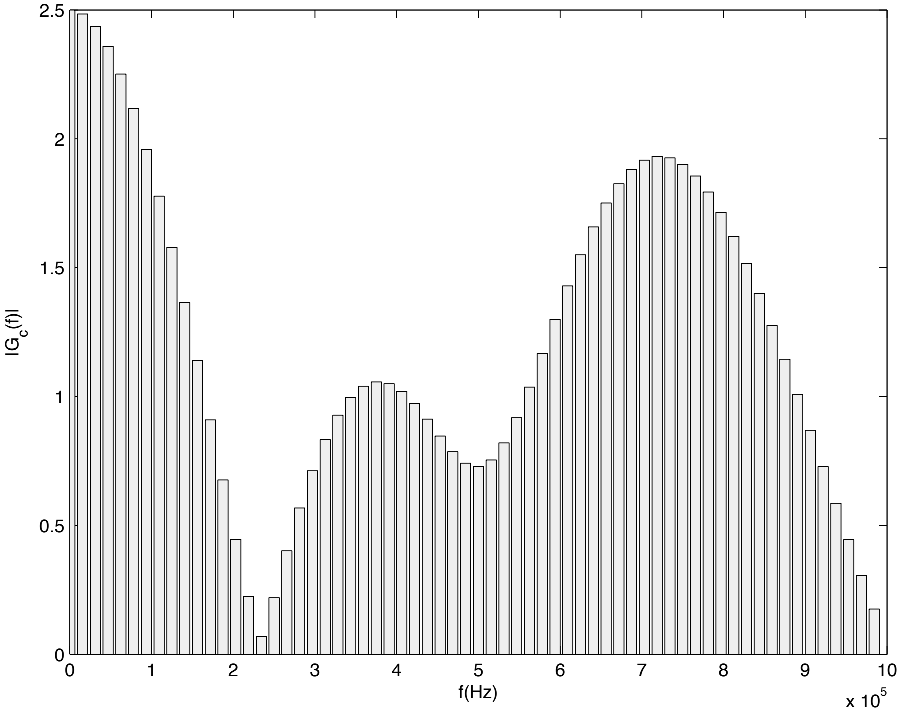

Figure 5.2. *Channel G_c split into 64 non-frequency-selective sub-bands*

Several types of multi-carrier modulations exist. In this chapter, we will first generally study in section 5.2 multi-carrier modulations based on orthogonal carriers, with and without carrier overlapping. Then, in section 5.3, we will be interested in the multi-carrier modulation which is the most common nowadays, and which has been chosen in many digital systems: orthogonal frequency division multiplex (OFDM). Finally, a particular case of filter bank-based multi-carrier modulations called filter bank multi-carrier (FBMC) is presented in section 5.4. Both advantages and drawbacks of each technique are enlightened in their respective sections.

5.2. General principles of multi-carrier modulation

5.2.1. *Parallel transmission on subcarriers*

We consider a channel with bandwidth B. If the channel is frequency-selective ($B >> B_c$), equalization at the receiver may become complex or achieve low performances. In order to avoid this, it is better to split the channel into a set of sub-bands that are each non-frequency-selective. This is obtained by separating the channel into N sub-bands of bandwidth B_N, with B_N very low with respect to the coherence band B_c. Sub-bands are generally called subcarriers, referring to their central carrier frequency. We note $B_N = B/N$.

Transmission is then performed as follows: N quadrature amplitude modulation (QAM) symbols to be transmitted are serially grouped, and then set in parallel. The n^{th} symbol, with $n = 0, ..., N-1$, is modulated by its specific carrier frequency, f_n, and then shaped by its transmit filter $g(t)$.

Transmission of N symbols in their respective sub-bands requires $T_N = 1/B_N = NT$ duration.

The flow of data to transmit in time is separated into N parallel signals each containing a subset of symbols:

$$X_n(t) = \sum_{k=-\infty}^{+\infty} c_{n,k}\, g(t - kT_N)\ \forall n \in \{0, 1, ..., N-1\} \qquad [5.1]$$

where $c_{n,k}$ is the QAM symbol sent on carrier frequency f_n in time interval $[kT_N, (k+1)T_N[$.

At the transmitter, the complex signals modulated on each subcarrier are added to generate the total transmit signal:

$$x(t) = \sum_{n=0}^{N-1} X_n(t) e^{j2\pi f_n t} \qquad [5.2]$$

Then, in order to obtain the real passband signal transmitted on the channel, only the real part of $x(t)$ must be kept (see equation [3.7]).

At the receiver, the signal transmitted around frequency f_n must be recovered. For this purpose, a general solution consists of filtering in parallel with N passband filters, each centered around frequency f_n. Assuming that the channel only adds a white Gaussian noise, signal $X_n(t)$ is then obtained at the output of the n^{th} filter, with a noise term. Synchronous detection is then applied on the in-phase and quadrature branches, and successive symbols $c_{n,k}$ are demodulated with a threshold detector. Finally, the data are serially combined in order to reconstitute the original signal.

This type of transmit/receive chain is called FBMC modulations. Another implementation solution called OFDM allows us not to use any filter bank neither at the transmitter nor at the receiver, which leads to lower implementation costs.

We can note that if the transmit symbols $c_{n,k}$ are real (which corresponds to amplitude shift keying (ASK) modulations), the transmitted signal simply becomes:

$$x(t) = \sum_{n=0}^{N-1} X_n(t)\cos(2\pi f_n t) \qquad [5.3]$$

5.2.2. *Non-overlapping multi-carrier modulations: FMT*

Non-overlapping multi-carrier modulations are called filtered multitone (FMT) modulations. They allow us to transmit several QAM symbols in parallel on different sub-bands. If the channel does not introduce any distortion and the receive filter is equal to the matched filter to $g(t)$, $g_r(t) = g^*(-t)$, then the complete filter $p(t) = g(t) * g_r(t)$ must verify the Nyquist criterion in order not to have any inter-symbol interference per sub-band. Thus, waveform $G(f)$ is chosen as a root raised cosine filter with roll-off factor α.

Moreover, in order to avoid interferences between symbols of two different sub-bands, the waveforms must not overlap in frequency domain. Thus, their frequency carrier must be separated from at least the waveform's spectral occupancy, B_N. In order to minimize the total bandwidth, we choose: $f_n = f_c + n(B_N)$, $n = 0, ..., N-1$, where f_c is the reference carrier frequency.

Since $B_N = (1 + \alpha)/T_N$ for a root raised cosine filter, spectral efficiency is maximized when $\alpha = 0$. Yet, in practice, this filter with perfect cutoff frequency is not feasible; thus, a guard band must be included in order to deal with transition bands between each sub-bands. The bandwidth actually occupied by the useful signal is equal to $1/T_N$, but carrier frequencies are separated from $B_N = 1/T_N + \epsilon$. This leads to a decrease in the useful spectral efficiency.

5.2.3. *Overlapping multi-carrier modulations*

Higher spectral efficiencies can be reached if some partial overlapping of carriers is allowed. The waveform $G(f)$ must be designed so that most of its energy is located in the main lobe of width $B_N = 1/T_N$; yet, contrary to the FMT case, we allow that some energy may be present in the secondary lobes that are outside of B_N. These modulations are called overlapping multi-carrier modulations. In order not to have any interferences between adjacent carriers, it is then necessary to determine orthogonality conditions on the chosen waveforms.

These conditions have been defined in 1960 by Chang [CHA 66] for real ASK symbol transmissions. Saltzberg [SAL 67] later demonstrated that Chang's method could be adapted to QAM complex symbols transmissions.

OFDM is a multi-carrier modulation technique that uses rectangular transmit filters in the time domain. OFDM allows us not to use filter banks and can be completely implemented with discrete Fourier transforms.

A class of overlapping filter banks is called discrete wavelet multitone (DWMT) [SAN 95]. It is also sometimes called cosine modulated multitone (CMT) [FAR 03]. In this case, real ASK symbols are transmitted in parallel on the sub-bands using a root raised cosine filter with $\alpha = 0$, with some overlapping due to the sidelobes. As real symbols are transmitted, $2N$ subcarriers must be used to transmit N complex symbols, whose real and imaginary parts have been separated before transmission.

A second class of overlapping filter banks is based on quadrature amplitude modulations and is called OFDM/OQAM [SIO 02], FBMC/OQAM or sometimes staggered multitone (SMT). Real ASK symbols are multiplexed by applying a phase shift of $\frac{\pi}{2}$ between adjacent symbols in both the time and frequency axis.

5.2.4. *Chapter's structure*

This chapter aims at studying in detail two main types of multi-carrier modulations: OFDM and FBMC/OQAM.

OFDM is the multi-carrier technique used in most modern telecommunication systems since the end of the 1990s. For instance, we may cite WiFi (IEEE 802.11), fourth-generation cellular systems (3GPP LTE [DAH 11], WiMAX IEEE 802.16 [AND 07]), digital terrestrial television DVB-T, wired assymetric digital subscriber line (ADSL) and very-high-bit-rate digital subscriber line (VDSL) transmissions.

FBMC/OQAM is a filter-bank-based multi-carrier modulation which is more complex to implement than OFDM, but which solves some of OFDM's limitations, especially its strong sensitivity to frequency asynchronicity. We will present it as an example of FBMC. We can note that FBMCs are among the favored candidates for post-OFDM systems.

REMARK 5.1.– In the whole chapter, we assume that the channel is non-frequency-selective in each sub-band B_N.

5.3. OFDM

OFDM is a multi-carrier modulation technique with channel overlapping. The transmit filter is a rectangular function on $[0, T_N]$, denoted by $\Pi_{T_N}(t)$. The set of functions:

$$\left\{\Pi_{T_N}(t)e^{j2\pi(f_c+n/T_N)t},\ n = 0, ..., N-1\right\} \qquad [5.4]$$

is orthogonal, as:

$$\int_{-\infty}^{+\infty} \Pi_{T_N}(t-kT_N)\Pi_{T_N}(t-k_0T_N)e^{j2\pi(n-n_0)/T_N t}dt = \delta_{k,k_0}\delta_{n,n_0} \qquad [5.5]$$

where δ is Kronecker's symbol, which is equal to $\delta_{n,n_0} = 1$ if $n = n_0$ and $\delta_{n,n_0} = 0$ elsewhere.

These orthogonal functions are spectrally distant of $B_N = 1/T_N$. Yet, since the Fourier transform of the rectangular function is the cardinal sine function, the frequency response of these functions has an infinite bandwidth. They consequently do not have disjoint frequency bands (see Figure 5.3). Nevertheless, their orthogonality property allows us to recover data at the receiver without interference. This principle is used in OFDM.

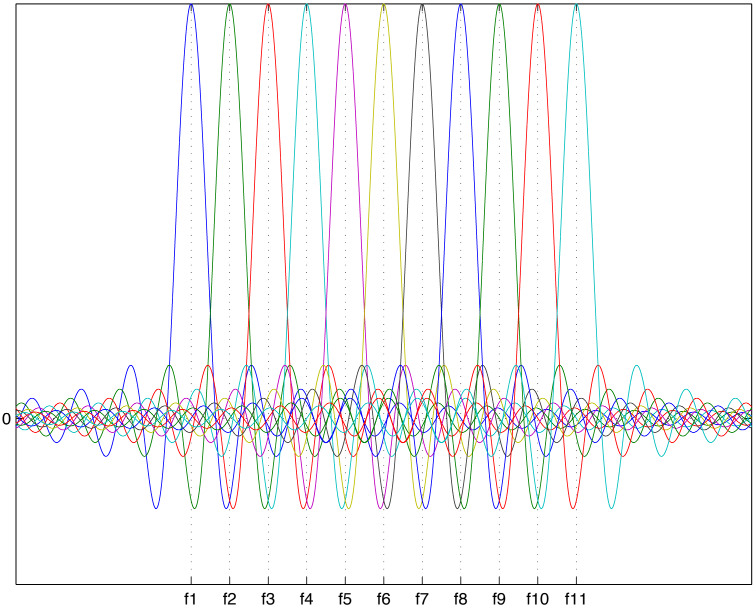

Figure 5.3. *Spectrum with frequency overlap: OFDM*

The transmitted signal consequently is:

$$\begin{aligned} x(t) &= \sum_{n=0}^{N-1} X_n(t) e^{j2\pi(f_c+n/T_N)t} \\ &= \sum_{n=0}^{N-1} \sum_{k=-\infty}^{+\infty} c_{n,k}\, \Pi_{T_N}(t-kT_N) e^{j2\pi(f_c+n/T_N)t} \end{aligned} \quad [5.6]$$

The main drawback of OFDM is that due to frequency overlapping, orthogonality may be lost if there is any frequency shift between transmitter and receiver. Even a small shift may lead to high-performance degradation, because of the induced inter-symbol interference.

OFDM is obtained by using:

– a discrete inverse Fourier transform at transmission;

– a discrete Fourier transform at reception.

First, this section presents the transmitter and receiver's block diagrams when the channel does not generate any distortion. Second, the cyclic prefix, which is required to avoid inter-block interference for the other types of channels, is detailed. Third, power allocation in OFDM is presented. It aims at maximizing the transmit data rate. OFDM's limitations are studied in sections 5.3.4 and 5.3.5: the peak to average power ratio (PAPR), and then its sensitivity to asynchronous transmissions. Finally, specific synchronization techniques for OFDM are summarized in the last section 5.3.6.

5.3.1. *Transmission and reception in OFDM*

We consider transmission of symbols in time interval $t \in [kT_N, (k+1)T_N[$ using the transmit filter $\Pi_{T_N}(t)$. According to the general equation of multi-carrier modulations [5.1], the n^{th} signal in this time interval is:

$$X_n(t) = c_{n,k} \tag{5.7}$$

Since it is constant, in the following, we denote it as $X_n(t) = X^k[n]$. It is the QAM symbol sent on the n^{th} frequency carrier in the k^{th} time interval.

The vector of N QAM symbols to transmit in $t \in [kT_N, (k+1)T_N[$ is written as follows:

$$\mathbf{X}^k = \left[X^k[0], X^k[1], .., X^k[N-1]\right] \tag{5.8}$$

The orthogonal modulation basis is given by [5.4]. Each parallel symbol is transmitted in a different subcarrier $f_n = \frac{n}{T_N} + f_c$. At the transmitter, the signal is at first complex and in baseband ($f_c = 0$). Consequently, the n^{th} frequency carrier is equal to $\frac{n}{T_N}$.

The transmit signal is the sequence of symbols from [5.2], sampled at symbol period $T = \frac{T_N}{N}$. Their amplitude is divided by $1/\sqrt{N}$ to simplify implementation, thus allowing us to directly use a discrete inverse fast Fourier transform (IFFT).

Thus, for $0 \leq i \leq N-1$:

$$\begin{aligned} x^k(iT_N/N) &= \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} X^k[n] e^{j2\pi \frac{n}{T_N} i \frac{T_N}{N}} \\ &= \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} X^k[n] e^{j2\pi \frac{ni}{N}} \\ &= x^k[i] \end{aligned} \qquad [5.9]$$

As equation [5.9] shows, time symbols $\mathbf{x}^k = \left[x^k[0], ..., x^k[N-1]\right]$ are obtained by applying a discrete IFFT on frequency symbols $\mathbf{X}^k = \left[X^k[0], X^k[1], .., X^k[N-1]\right]$.

If the channel does not generate any distortion, symbols $\left[x^k[0], ..., x^k[N-1]\right]$ are then serialized and passed in a digital to analog converter. Thus, we obtain the complex envelope of the continuous-time OFDM signal. This signal is transmitted on the channel after having been transferred on the central frequency carrier f_c and turned into a passband signal.

OFDM transmission chain is represented in Figure 5.4.

If there is no channel distortion, symbols are recovered at the receiver after demodulation around f_c by applying a discrete Fourier transform.

5.3.2. *Cyclic prefix principle*

We now assume that the channel's impulse response $g_c(t)$ sampled at symbol period T has length $L-1$, where $L-1 = T_m/T$, with T_m the maximum delay of the channel. The impulse response $g_c(t)$ can consequently be written depending on its samples: $[g_c[0], g_c[1], ..., g_c[L-1]]$.

If the transmitter only uses an IFFT, the last components of the OFDM time block transmitted in interval $t \in [(k-1)T_N, kT_N[$ interfere with the first components of the OFDM time block transmitted in interval $t \in [kT_N, (k+1)T_N[$.

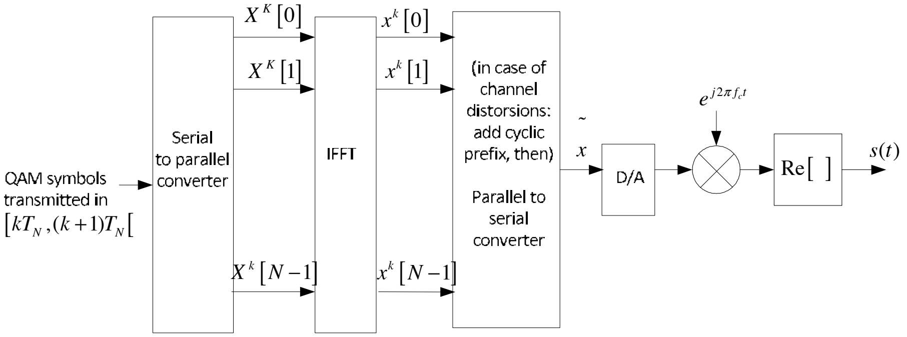

Figure 5.4. *OFDM transmission chain*

We denote by $\mathbf{x}^k$ the data block transmitted in $t \in [kT_N, (k+1)T_N[$:

$$\mathbf{x}^k = \Big[x^k[0], x^k[1], .., x^k[N-1]\Big] \qquad [5.10]$$

In order to avoid inter-block interference, a cyclic prefix of length N_{CP} must be introduced after IFFT.

5.3.2.1. *Transmitted symbols*

Adding the prefix cyclic consists of pasting the last N_{CP} components

$$\Big[x^k[N-N_{\text{CP}}], x^k[N-N_{\text{CP}}+1], ..., x^k[N-1]\Big] \qquad [5.11]$$

of the block transmitted in $t \in [kT_N, (k+1)T_N[$ at the beginning of that block (see Figure 5.5). The N_{CP} last components are called the "cyclic prefix", because they are included at the beginning of the block (therefore, the "prefix" denotation) and because they contain information that is also present at the end of the block (therefore, the "cyclic" denotation).

In order to include the cyclic prefix, distance N_{CP} must first be added between two consecutive blocks.

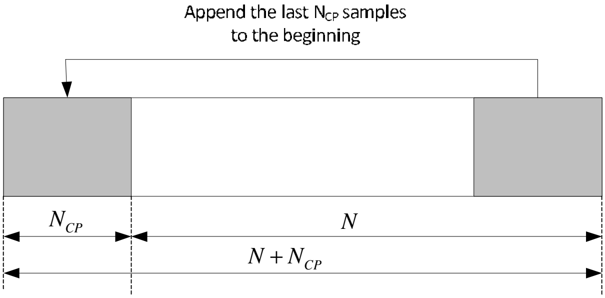

Figure 5.5. *Cyclic prefix*

The transmitted sequence is denoted by $\widetilde{x}^k$. Thus, it contains the $N + N_{\text{CP}}$ following data:

$$\widetilde{\mathbf{x}}^k = \Big[x^k[N - N_{\text{CP}}], x^k[N - N_{\text{CP}} + 1], ...,$$
$$x^k[N-1], x^k[0], x^k[1], ..., x^k[N-1]\Big] \qquad [5.12]$$

The time block given by equation [5.12] is called an OFDM symbol. It contains $N_{\text{OFDM}} = N + N_{\text{CP}}$ samples, as illustrated in Figure 5.5. Its duration is consequently $T_{\text{OFDM}} = T \times (N + N_{\text{CP}}) = T_N + T \times N_{\text{CP}}$.

We may note that adding a cyclic prefix decreases the transmission's spectral efficiency. Indeed, the ratio of useful signal in the transmitted signal is T_N/T_{OFDM}. Yet, the symbol rate is: N/T_{OFDM} symbols/s. If all symbols belong to an M-QAM constellation, the spectral efficiency is:

$$\begin{aligned}\eta &= \frac{N}{T_{\text{OFDM}}B}\log_2(M) \\ &= \frac{1}{1+\frac{N_{\text{CP}}}{N}}\log_2(M)\end{aligned} \qquad [5.13]$$

For instance, if we consider a quadrature phase shift keying (QPSK) modulation with $N = 128$ subcarriers and a prefix cyclic containing $N_{\text{CP}} = 8$ samples, spectral efficiency is equal to $\eta = 1.88$. If the bandwidth is $B = 10$ MHz, the useful data rate is $D = \eta \times B = 18.82$ Mbits/s, even though the data rate transmitted on the channel is equal 20 Mbits/s. As a result, the cyclic prefix must be accurately dimensioned, so that the spectral efficiency does not decrease too much. The minimum cyclic prefix's length is determined in the next section.

5.3.2.2. *Transmission on delay-generating channels*

When it is transmitted on the channel, the data vector $\widetilde{\mathbf{x}}^k$ [5.12] is linearly filtered by the channel. The received data are consequently equal to the convolution of $\widetilde{\mathbf{x}}^k$ and the channel's impulse response. If we do not take the noise into account, the received data after sampling are, for $-N_{\text{CP}} \leq n \leq N-1$:

$$\begin{aligned} q[n] &= \sum_{p \in \mathcal{Z}} g_c[p] \widetilde{x}^k[n-p] \\ &= \sum_{p=0}^{L-1} g_c[p] \widetilde{x}^k[n-p] \end{aligned} \qquad [5.14]$$

We now detail the values of some particular samples of $\mathbf{q}$. Sample $-N_{\text{CP}}$ is equal to:

$$\begin{aligned} q[-N_{\text{CP}}] &= g_c[0]\widetilde{x}^k[-N_{\text{CP}}] + g_c[1]\widetilde{x}^k[-N_{\text{CP}}-1] + ... \\ &\quad + g_c[L-1]\widetilde{x}^k[-N_{\text{CP}} - (L-1)] \\ &= g_c[0]x^k[N - N_{\text{CP}}] + g_c[1]x^{k-1}[N-1] + ... \\ &\quad + g_c[L-1]x^{k-1}[N-(L-1)] \end{aligned} \qquad [5.15]$$

It contains L samples of the data block transmitted in $[(k-1)T_N, kT_N[$, and only one sample of the useful data block transmitted in $[kT_N, (k+1)T_N[$. Sample 0 is equal to:

$$\begin{aligned} q[0] &= g_c[0]\widetilde{x}^k[0] + g_c[1]\widetilde{x}^k[-1] + ... + g_c[L-1]\widetilde{x}^k[-(L-1)] \\ &= g_c[0]x^k[0] + g_c[1]x^k[N-1] + ... + g_c[L-1]x^k[N-(L-1)] \end{aligned} \qquad [5.16]$$

It does not contain any sample from the interfering data block transmitted in $t \in [(k-1)T_N, kT_N[$. Similarly, all samples $q[n]$ with $n \geq 0$ no longer contain any sample from the data block transmitted in the previous OFDM symbol period. They are free of any inter-block interference.

In order to completely remove inter-block interference, the cyclic prefix must "absorb" all interferences. It is consequently necessary that $N_{\mathrm{CP}} \geq L-1$.

This implies that the cyclic prefix must possess at least as many samples as the channel's impulse response length. The cyclic prefix duration is then $T_{\mathrm{CP}} = T \times N_{\mathrm{CP}}$. Since $L-1 = T_m/T$, the cyclic prefix duration must be at least equal to the maximum delay of the channel.

5.3.2.3. *Received symbols*

The received block of symbols can be expressed with matrices as follows:

$$\begin{pmatrix} q[-N_{\mathrm{CP}}] \\ q[-N_{\mathrm{CP}}+1] \\ \dots \\ q[0] \\ \dots \\ q[N-1] \end{pmatrix}$$

$$= \begin{pmatrix} g_c[0] & 0 & \dots & \dots & 0 \\ g_c[1] & g_c[0] & 0 & \dots & 0 \\ \dots & & & & \\ g_c[L-1] & \dots & \dots & \dots & 0 \\ 0 & g_c[L-1] & \dots & \dots & 0 \\ \dots & & & & \\ 0 & \dots & g_c[L-1] & \dots & g_c[0] \end{pmatrix} \begin{pmatrix} x^k[N-N_{\mathrm{CP}}] \\ \dots \\ x^k[N-1] \\ x^k[0] \\ \dots \\ x^k[N-1] \end{pmatrix}$$

$$+ \begin{pmatrix} 0 \dots 0 & g_c[L-1] & \dots & g_c[1] \\ 0 \dots 0 & 0 & \dots & g_c[2] \\ \dots & & & \\ 0 \dots 0 & 0 & \dots & g_c[L-1] \\ 0 \dots 0 & 0 & \dots & 0 \\ 0 \dots 0 & 0 & \dots & 0 \\ 0 \dots 0 & 0 & \dots & 0 \end{pmatrix} \begin{pmatrix} x^{k-1}[N-N_{\mathrm{CP}}] \\ \dots \\ x^{k-1}[N-1] \\ x^{k-1}[0] \\ \dots \\ x^{k-1}[N-1] \end{pmatrix} \qquad [5.17]$$

In equation [5.17], the first part only depends on the symbols transmitted in interval $[kT_N, (k+1)T_N[$, whereas the second part corresponds to interferences between the blocks transmitted in $[kT_N, (k+1)T_N[$ and $[(k-1)T_N, kT_N[$. Both matrices are squared of size $(N + N_{\text{CP}})\ (N + N_{\text{CP}})$.

At the receiver, the first N_{CP} samples that contain inter-block interference are removed. The received samples can then be written as:

$$\begin{pmatrix} q[0] \\ \ldots \\ q[N-1] \end{pmatrix} = \begin{pmatrix} 0 \ldots 0 & g_c[L-1] & \ldots & g_c[0] & 0 & \ldots & 0 \\ 0 \ldots 0 & 0 & g_c[L-1] & \ldots & g_c[0] & \ldots & 0 \\ 0 \ldots 0 & \ldots & & & & & \\ 0 \ldots 0 & 0 & \ldots & \ldots & g_c[L-1] & \ldots & g_c[0] \end{pmatrix} \times \begin{pmatrix} x^k[N-N_{\text{CP}}] \\ \ldots \\ x^k[N-1] \\ x^k[0] \\ \ldots \\ x^k[N-1] \end{pmatrix} \quad [5.18]$$

The matrix relating the received symbols to $\widetilde{\mathbf{x}}^k$ is now of size $N \times (N + N_{\text{CP}})$. It contains a submatrix of zeros of size $N \times (N_{\text{CP}} - L)$, and a submatrix of size $N \times (N + L)$ that contains the channel's impulse response coefficients.

Using the redundancy in $\widetilde{\mathbf{x}}^k$ due to the cyclic prefix, equation [5.18] can be rewritten as follows by introducing a circulant matrix, denoted as $\mathbf{C}$:

$$\begin{pmatrix} q[0] \\ \ldots \\ q[N-1] \end{pmatrix}$$

$$= \begin{pmatrix} g_c[0] & 0 & ... & g_c[L-1] & ... & g_c[1] \\ g_c[1] & g_c[0] & ... & 0 & g_c[L-1] & ... \\ ... & & & & & \\ g_c[L-1] & ... & ... & ... & ... & g_c[L-1] \\ 0 & g_c[L-1] & ... & ... & ... & 0 \\ 0 & 0... & ... & g_c[L-1] & ... & g_c[0] \end{pmatrix}$$

$$\times \begin{pmatrix} x^k[0] \\ ... \\ x^k[N-1] \end{pmatrix}$$

$$\Leftrightarrow \begin{pmatrix} q[0] \\ ... \\ q[N-1] \end{pmatrix} = \mathbf{C} \begin{pmatrix} x^k[0] \\ ... \\ x^k[N-1] \end{pmatrix} \qquad [5.19]$$

Now, symbols $x^k[n]$ are obtained from $X^k[i]$ by applying an IFFT.

Let $\mathbf{F} = \left(\frac{1}{\sqrt{N}} e^{-j2\pi ni/N}\right)_{(i,n)}$ be the direct Fourier transform matrix (FFT). IFFT is simply obtained by applying $\mathbf{F}^H$, where ^{H} is the Hermitian operator. Thus:

$$\begin{pmatrix} x^k[0] \\ ... \\ x^k[N-1] \end{pmatrix} = \mathbf{F}^H \begin{pmatrix} X^k[0] \\ ... \\ X^k[N-1] \end{pmatrix} \qquad [5.20]$$

Since matrix $\mathbf{C}$ is circulant, it can be diagonalized in a Fourier eigenvector basis.

$$\mathbf{C} = \mathbf{F}^H \begin{pmatrix} G_c[0] & 0 & ... & 0 \\ 0 & G_c[1] & ... & 0 \\ ... & & & \\ 0 & 0 & ... & G_c[N-1] \end{pmatrix} \mathbf{F} \qquad [5.21]$$

where $\mathbf{G} = [G_c[0], G_c[1], ..., G_c[N-1]]$ are the coefficients obtained by discrete Fourier transform on the channel's impulse response $[g_c[0], h[1], ..., g_c[L-1]]$ at frequencies $f = n/T_N$.

Thus, if we apply the discrete Fourier transform on vector $[q[0], .., q[N-1]]^T$, the received symbols can be written as follows:

$$\begin{pmatrix} Y^k[0] \\ ... \\ Y^k[N-1] \end{pmatrix} = \mathbf{F} \begin{pmatrix} q[0] \\ ... \\ q[N-1] \end{pmatrix}$$

$$= \begin{pmatrix} G_c[0] & 0 & ... & 0 \\ 0 & G_c[1] & ... & 0 \\ ... & & & \\ 0 & 0 & ... & G_c[N-1] \end{pmatrix} \begin{pmatrix} X^k[0] \\ ... \\ X^k[N-1] \end{pmatrix} \qquad [5.22]$$

The received signal can consequently be separated per subcarrier n:

$$Y^k[n] = G_c[n] X^k[n] \qquad [5.23]$$

where $G_c[n] = G_c(f_n)$ is the channel attenuation in subcarrier n, assumed constant since the sub-band's bandwidth is far lower than the coherence band.

The values of symbols $X^k[n]$ are eventually obtained by dividing the received symbols by $G_c(f_n)$. Equalization is thus very simple; it is called frequency equalization.

To conclude, OFDM transmission on a wideband frequency-selective channel can be equivalently seen as the parallel transmission of N data flows on N narrowband non-frequency-selective sub-bands.

The OFDM receiver's block diagram is represented in Figure 5.6.

5.3.3. *Optimum power allocation in OFDM*

Equation [5.23] shows that OFDM transmission can be separated per orthogonal subcarrier. On each subcarrier, a complex additive white Gaussian noise term modeling baseband thermal noise must be added. The channel per subcarrier is thus an additive white Gaussian noise channel with complex attenuation $G_c[n]$:

$$Y^k[n] = G_c[n]X^k[n] + B^k[n] \qquad [5.24]$$

where $B^k[n]$ is the noise per subcarrier, whose power in sub-band is equal to $N_0 B_N$, where N_0 is the one-sided noise power spectrum density. To simplify, in the remainder of this section, we remove block exponent k.

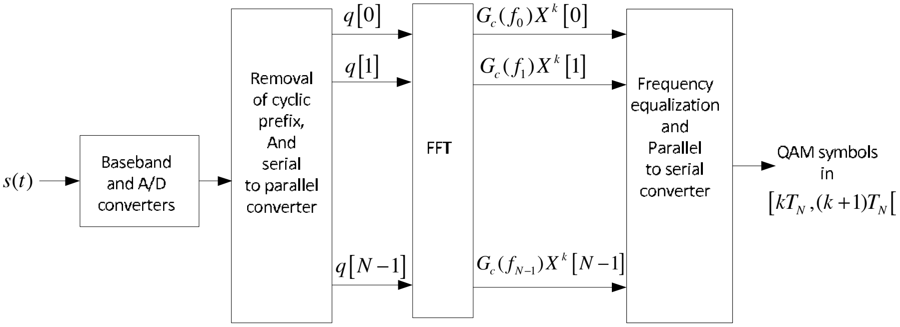
$s(t)$
Baseband and A/D converters
Removal of cyclic prefix, And serial to parallel converter
$q[0]$
$q[1]$
$q[N-1]$
FFT
$G_c(f_0)X^k[0]$
$G_c(f_1)X^k[1]$
$G_c(f_{N-1})X^k[N-1]$
Frequency equalization and Parallel to serial converter
QAM symbols in
$[kT_N,(k+1)T_N[$

Figure 5.6. *OFDM receiver with a channel generating distortions and delays*

At the receiver, demodulation of each symbol is performed as follows: first, frequency equalization is applied. The received signal becomes $X^k[n] + B[n]/G_c[n]$. The noise sample remains complex, but its amplitude is modified: if the channel gain $|G_c[n]|^2$ is higher than 1, the noise's amplitude decreases; else it increases. Assuming synchronous detection, a threshold detector is then applied on both real and imaginary parts of the signal, in order to estimate $\widehat{X}^k[n]$. The quality of this estimation depends on the signal-to-noise ratio, which itself depends on the channel's gain.

By applying Shannon's capacity formula for the additive white Gaussian noise channel, we obtain the maximum data rate that can be achieved on subcarrier n:

$$C[n] = B_N \log_2\left(1 + \frac{|G_c[n]|^2 P_X[n]}{N_0 B_N}\right) \qquad [5.25]$$

where $P_X[n]$ is defined as the power of symbol $X[n]$.

The data rate summed over all subcarrier finally is:

$$C_{\text{tot}} = B_N \sum_{n=1}^{N} \log_2\left(1 + \frac{|G_c[n]|^2 P_X[n]}{N_0 B_N}\right) \qquad [5.26]$$

The power can be adapted per subcarrier in order to maximize C_{tot}. Let P_{max} be the total power budget at the transmitter. If the transmitter has no knowledge of the channel, which means that it does not know the channel gains $|G_c[n]|^2$, fair power allocation is optimum: $P_X[n] = P_{\text{max}}/N$.

On the contrary, if the transmitter has full knowledge of the channel, including the noise power, then power allocation must provide a solution to the following optimization problem[1], with $\mathbf{P_X} = [P_X[0], P_X[1], \ldots P_X[N-1]]$:

$$\begin{aligned}
&\mathbf{P_X^*} = \text{arg} \max_{\mathbf{P_X}} \; B_N \sum_{n=1}^{N} \log_2\left(1 + \frac{|G_c[n]|^2 P_X[n]}{N_0 B_N}\right) \\
&\text{s.t.} \sum_{n=1}^{N} P_X[n] \leq P_{\text{max}} \\
&\text{s.t. } P_X[n] \geq 0 \; \forall n \in \{0, 1, .., N-1\}
\end{aligned} \qquad [5.27]$$

Since the objective function increases with $\mathbf{P_X}$, its maximum is achieved when $\sum_{n=1}^{N} P_X[n] = P_{\text{max}}$. Consequently, problem [5.27] is a convex optimization problem with equality constraint that possesses a unique solution.

The problem's Lagrangian is:

$$\begin{aligned}
L(\mathbf{P_X}, \lambda, \nu) = {} & B_N \sum_{n=1}^{N} \log_2\left(1 + \frac{|G_c[n]|^2 P_X[n]}{N_0 B_N}\right) \\
& + \lambda\left(P_{\text{max}} - \sum_{n=1}^{N} P_X[n]\right) + \sum_{n=1}^{N} \nu_n P_X[n]
\end{aligned} \qquad [5.28]$$

where λ and ν are the Lagrange multipliers.

[1] Where s.t. in [5.27] means "subject to".

The optimum solution is the unique point that verifies all Karush–Kuhn–Tucker conditions [BOY 04], defined as follows:

$$\begin{aligned}
&\nabla L(\mathbf{P}_{\mathbf{X}}^{\text{opt}}, \lambda, \nu) = \mathbf{0} \\
&\lambda \left(P_{\max} - \sum_{n=1}^{N} P_X^{\text{opt}}[n] \right) = 0 \\
&\nu_n P_X^{\text{opt}}[n] = 0 \quad \forall n \in \{0, 1, .., N-1\} \\
&\lambda \geq 0 \\
&\nu_n \geq 0 \quad \forall n \in \{0, 1, .., N-1\}
\end{aligned} \qquad [5.29]$$

The Lagrangian's derivative [5.28] with respect to $P_X^{\text{opt}}[n]$ is equal to:

$$\frac{\gamma_n}{1 + \gamma_n P_X^{\text{opt}}[n]} - \lambda' + \nu_n' = 0 \qquad [5.30]$$

where $\gamma_n = |G_c[n]|^2 / (N_0 B_N)$, and where constant $B_N / \log(2)$ has been included in Lagrange multipliers to simplify notations: $\lambda' = \lambda \times \log(2)/B_N$ and $\nu' = \nu \times \log(2)/B_N$.

Constraint $\nu_n' \geq 0$ and equation [5.31] imply that:

$$\lambda' \geq \frac{\gamma_n}{1 + \gamma_n P_X^{\text{opt}}[n]} = 0 \qquad [5.31]$$

Moreover, constraint $\nu_n' P_X^{\text{opt}}[n] = 0$ is equivalent to:

$$P_X^{\text{opt}}[n] \left(\lambda' - \frac{\gamma_n}{1 + \gamma_n P_X^{\text{opt}}[n]} \right) = 0 \qquad [5.32]$$

Yet, if equation [5.31] holds with strict inequality, constraint [5.32] only holds if $P_X^{\text{opt}}[n] = 0$. In the opposite case, there is an equality in [5.31] and $P_X^{\text{opt}}[n]$ may be different from zero. Consequently, the optimum solution of optimization problem [5.27] is:

$$P_X^{\text{opt}}[n] = \left[\frac{1}{\lambda'} - \frac{1}{\gamma_n} \right]^+ = 0 \quad \forall n \in \{0, 1, .., N-1\} \qquad [5.33]$$

where:

$$[x]^+ = \begin{cases} 0 \text{ si } x \leq 0 \\ x \text{ si } x \geq 0 \end{cases} \quad [5.34]$$

The Lagrange multiplier λ' is determined in order to fulfill the sum power constraint: $\sum_{n=1}^{N} P_X^{\text{opt}}[n] = P_{\text{max}}$.

Solution [5.33] is called water-filling or water-pouring. It may be understood as follows: the power budget is distributed over subcarriers, seen as channels with inverse gains $1/\gamma_n$, until sum P_{max} is reached. When channel gains are too low, no power is provided. As a result, only subcarriers which have high channel gains get some power.

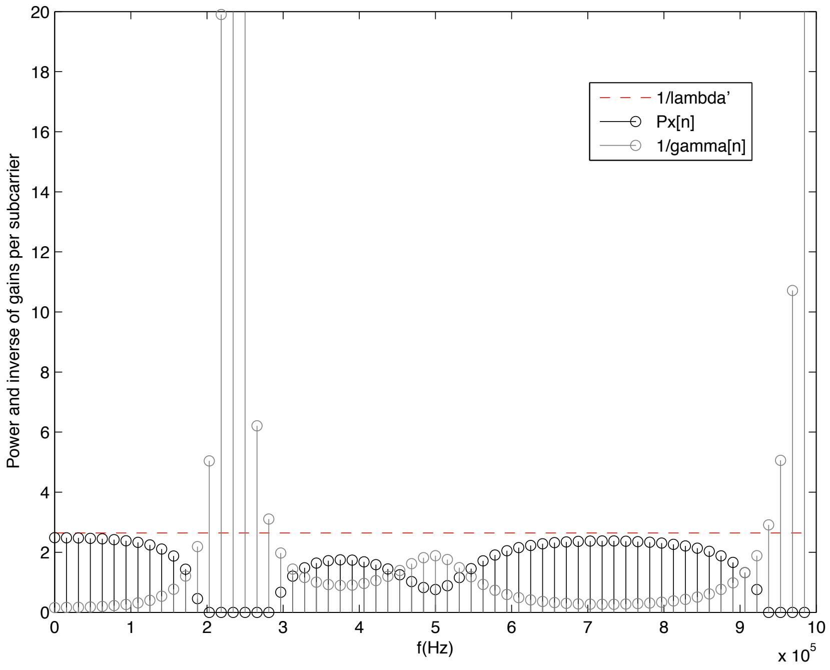

Figure 5.7. *Power per subcarrier with the channel of Figure 5.2, first case*

This classical technique is also used for multi-antenna transmissions (multiple input multiple output (MIMO))[2]. It is illustrated in Figure 5.7 for

[2] We do not detail MIMO techniques in this book.

the channel g_c whose impulse response once the channel has been separated into 64 sub-bands is as represented in Figure 5.2. To simplify, we assume that the noise power is compensated for by path losses [GOL 05]. The maximum power[3] is equal to 100 W. The power per subcarrier obtained with water-filling [5.33] is drawn in black, and the $1/\lambda'$ level is drawn in gray. We can see that subcarriers with low gains get no power because their gain's inverse exceeds level $1/\lambda'$. When the maximum power is lower (20 W), the water level $1/\lambda'$ decreases, and power is distributed on less subcarriers, as shown in Figure 5.8.

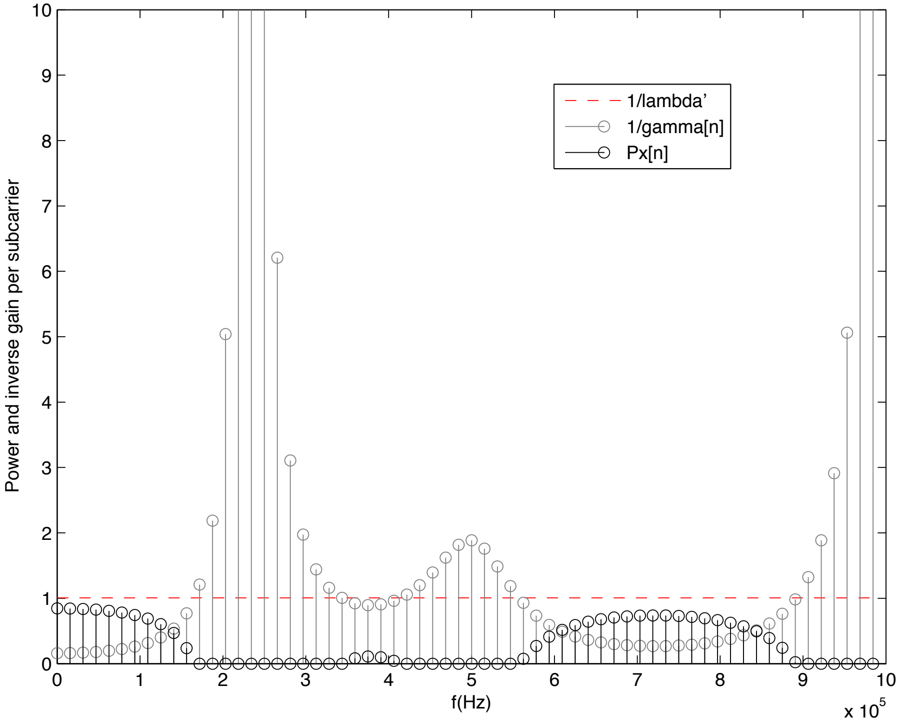

Figure 5.8. *Power per subcarrier with the channel of Figure 5.2, second case*

[3] We can of course notice that this value is unrealistic, since path losses are not explicitly considered in this example.

Water-filling's proof is based on Shannon's capacity, which represents an upper bound on the data rate. In practice, the data rate per subcarrier is not continuous but must belong to a set of discrete values, and depend on the chosen modulation. Consequently, water-filling must be performed on the modulation's choice per subcarrier. This is called bit loading [CAM 98, PAP 07]. An approximate solution of bit loading consists of rounding down equation [5.33] to discrete values of the number of bits per symbol of the available QAM constellations. The discrete optimum solution is based on the same principle as water-filling: the higher the gain is, the higher the constellation size chosen in the subcarrier becomes.

5.3.4. *PAPR*

Several drawbacks restrict the usage of OFDM. Among these, we will detail the most important: PAPR and sensitivity to frequency asynchronicity.

In this section, we are first interested in PAPR and the various techniques proposed to decrease it.

The PAPR of a vector or time symbols $\mathbf{x} = [x[0], ..., x[N-1]]$ is equal to the ratio between the maximum power over all transmitted time symbols and the average power of the symbols:

$$\text{PAPR} = \frac{\max_{0\leq n\leq N-1} |x[n]|^2}{\frac{1}{N}\sum_{k=0}^{N-1} |x[k]|^2} \qquad [5.35]$$

We assumed in equation [5.35] that the OFDM symbol does not contain any cyclic prefix, but an extension to the cyclic prefix case is straightforward.

Equation [5.9] shows that each time symbol $x[n]$ is the sum of N weighted frequency symbols, which are independent and identically distributed. If N is large, using the central limit theorem, the real and imaginary parts of $x[n]$ both tend toward a Gaussian distribution, as shown [4] in Figure 5.9 for $N = 64$ and $N = 1024$. As a result, the modulus of $x[n]$ tends toward a Rayleigh distribution and its phase toward a uniform distribution. Yet, since Rayleigh distribution has an infinite width, $|x[n]|$ is not bounded. The peak power

[4] Only the real part is drawn in this figure. The imaginary part's distribution is similar.

$\max_n |x[n]|^2$ always exceeds a given threshold with a non-zero probability, and the PAPR cannot be controlled.

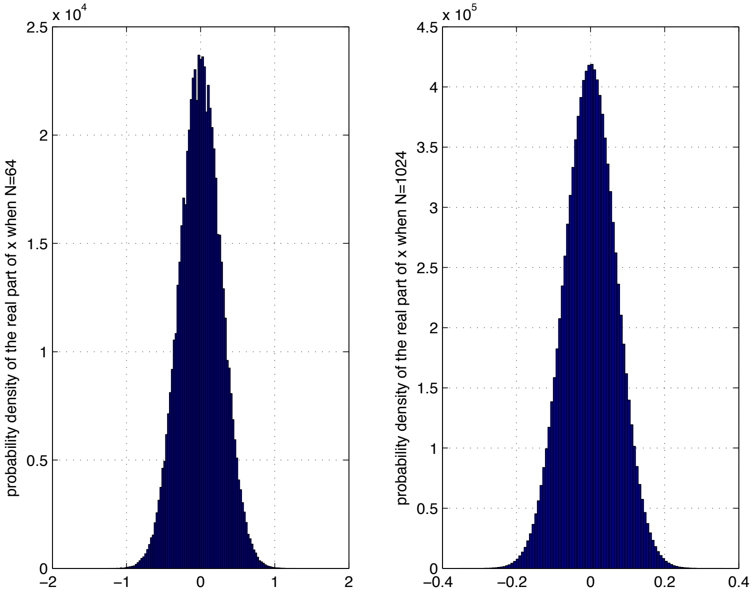

Figure 5.9. *Distribution of the real part of time symbols when $N = 64$ and $N = 1024$*

Yet, when time signal variations are too large, power amplifiers must handle some peak powers. Classical amplifiers are not able to follow these variations, and generate nonlinearities on the transmitted signal.

PAPR increases with the number of subcarriers for low values of N. When N is high enough for the central limit theorem to hold, since the distribution of $x[n]$ tends to a Gaussian for both real and imaginary parts, the PAPR tends to the PAPR of Gaussian complex symbols, and has an asymptote. Consequently, decreasing the number of subcarriers to decrease the PAPR is not efficient. In any case, this solution is not really practical, since most current systems are wideband, and a large number of subcarriers are required in order to obtain non-frequency-selective sub-bands.

Several techniques exist in order to restrict PAPR. Among them, we can distinguish clipping, coding techniques and rotation techniques (selective mapping and partial transmit sequences).

Clipping the modulus of the transmitted symbols is the simplest technique [O'NE 95]. Before transmitting each symbol, its modulus is limited, so that the value of:

$$\left(|x[n]|^2\right) / \left(1/N \sum_{k=0}^{N-1} |x[k]|^2\right)$$

is always lower than a threshold value. The phase of the clipped symbols remains unchanged. Figure 5.10 represents the distribution of the real part of time symbols $x[n]$ before and after clipping, when $N = 1024$. In this example, the threshold value is set to 10. We can note that the average and maximum powers have been decreased. PAPR has been reduced by approximately factor 5, and influences around 10% of all symbols.

Because of clipping, inter-symbol interference and interference outside of the transmit band may occur. It is then necessary to add some filtering in order to reduce out-of-band interferences. Obviously, the bit error rate increases due to inter-symbol interference [LI 97].

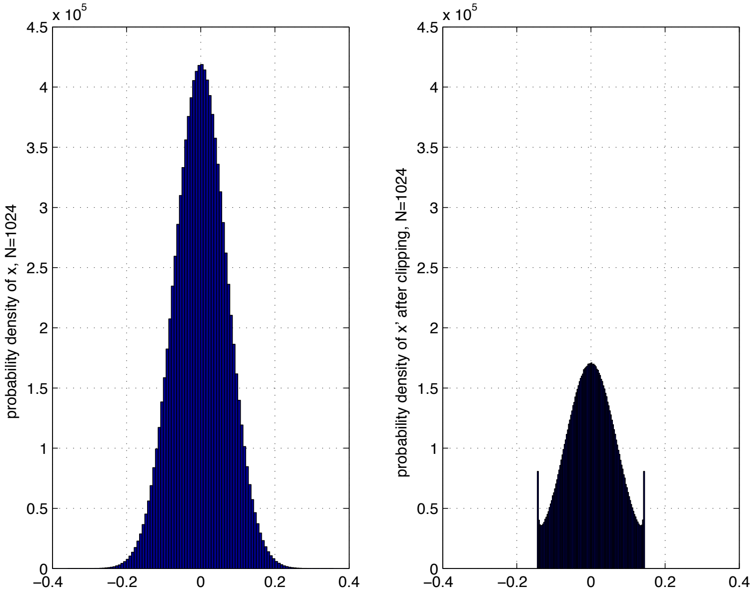

Figure 5.10. *Distribution of the real part of time symbols when $N = 1024$, before and after clipping*

Another more efficient technique is based on the addition of an error-correcting code before transmission. It may be a block code [JON 94], and in particular a Reed–Muller code [POP 91]. The objective is to avoid sequences of N frequency symbols with the same phase, which leads to a power peak on time symbols. For this purpose, symbols are coded, and then the PAPR of the resulting codewords (after the IFFT) is evaluated. Only a subset of codewords generating low PAPR is finally used.

Probabilistic techniques have also been recently proposed. The first technique, called selective mapping [BAU 96], sets phase rotations on several versions of the same signal, and then evaluates the resulting PAPR. The transmitted signal is chosen as the one with the lowest PAPR, after multiplication by the best phase rotation vector. This technique provides good results in terms of PAPR reduction. Yet, it requires that the rotation vector chosen by the transmitter should be sent to the receiver. It also requires implementing several IFFTs at the transmitter, which increases the complexity of the system.

The other probabilistic technique, called partial transmit sequence [MUL 97], is based on the same principle, but sets phase rotations no longer on the whole signal, but on a subset of subcarriers. The other subcarriers are set to zero before the IFFT. The same phase rotation is applied on all block symbols. The objective is then to determine the phase rotation vector that minimizes PAPR. Partial transmit sequences imply, such as selective mapping, to use several IFFTs at transmission, and to inform the receiver on the chosen rotation vector. Its complexity is higher, but its performances in terms of PAPR reduction are better.

5.3.5. *Sensitivity to asynchronicity*

5.3.5.1. *Frequency asynchronicity*

Another drawback of OFDM is that it is highly sensitive to asynchronicity, and especially frequency asynchronicity. Subcarrier orthogonality may indeed be lost at the receiver due to asynchronicity.

Frequency asynchronicity may be due to asynchronous frequency oscillators, or due to the Doppler effect.

Subcarriers are no longer orthogonal. Since the frequency response of the signal transmitted in each subcarrier is an infinite-width cardinal sine, even a low frequency shift generates interferences between symbols received on subcarrier f_n after FFT and symbols of adjacent subcarriers. The level of received interference depends on the frequency shift. Thus, if it exceeds $B_N/10$, recovering the transmitted symbols is almost impossible.

On the contrary, if the frequency shift is low and inter-subcarrier interference is of the same order as the noise power, interference may be seen as an additional noise.

5.3.5.2. *Time asynchronicity*

Time asynchronicity is less an issue for OFDM than frequency asynchronicity. In most systems, the cyclic prefix is overdimensioned in order to cover up for the worst case, which rarely occurs. Consequently, if the sum of the time delay due to asynchronicity and of the maximum delay of the channel T_m is less than the cyclic prefix duration, the time delay is absorbed by the cyclic prefix.

We may nevertheless notice that time delay exceeding the cyclic prefix duration strongly degrades performances. It has thus been shown in [MED 09] that the interference normalized power exceeds 10^{-3} on 8 adjacent subcarriers, at both left and right of subcarrier f_n, when the cyclic prefix duration is $N_{\text{CP}} = 1/8^{\text{th}}$ of T and the time delay is uniformly distributed in interval $[N_{\text{CP}}/2, T + 3N_{\text{CP}}/2]$.

5.3.6. ***OFDM synchronization techniques***

In Chapter 4, we have detailed synchronization techniques that may be used whatever the transmission system. Nevertheless, the specificities of the transmission may be taken advantage of in order to simplify synchronization or improve its performances. In OFDM, specific techniques have been proposed for time synchronization and frequency shift correction. Thus, Schmidl and Cox [SCH 97] or Shi and Serpedin [SHI 04] have proposed efficient estimators to determine the beginning of OFDM symbols and the symbol period. Moose estimator [MOO 94] allows us to evaluate the frequency shift. All three techniques are based on training sequences. Time synchronization may also be performed blindly due to the structure of the cyclic prefix [VAN 97]. In the following, we detail these estimators.

The OFDM symbol duration is $N_T = N + N_{\text{CP}}$. In the following, to avoid confusion, an OFDM symbol is equivalently called an OFDM block. The k^{th} transmitted OFDM block is:

$$x^k[i] = \begin{cases} \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} X^k[n] e^{j2\pi \frac{ni}{N}} & \forall i \in \{0, ..., N-1\} \\ x^k[i+N] & \forall i \in \{-N_{\text{CP}}, ..., -1\} \end{cases} \qquad [5.36]$$

g_c is the channel impulse response of length $L - 1$. The received baseband signal after sampling at period $1/T$ is:

$$y[i] = \sum_{k=-\infty}^{+\infty} \sum_{l=0}^{L-1} g_c[l] x^k[i - l - kN_T] + b[i] \qquad [5.37]$$

where b is an additive white Gaussian noise.

If at the receiver, a time synchronization error represented as a multiple θ of the sampling period occurs, the received signal is shifted by θ samples:

$$y[i] = \sum_{k=-\infty}^{+\infty} \sum_{l=0}^{L-1} g_c[l] x^k[i - l - \theta - kN_T] + b[i] \qquad [5.38]$$

Frequency shift between transmitter and receiver is represented by a phase shift $e^{j2\pi\delta f nT}$ at sampling times nT. We note $\epsilon = \delta f NT$ the frequency shift normalized by $1/NT$. By adding the synchronization error, the received signal becomes:

$$y[i] = e^{j2\pi\epsilon i/N} \sum_{k=-\infty}^{+\infty} \sum_{l=0}^{L-1} g_c[l] x^k[i - l - \theta - kN_T] + b[i] \qquad [5.39]$$

The objective of synchronization techniques is to determine the estimates of θ and ϵ, denoted by $\widehat{\theta}$ and $\widehat{\epsilon}$. Frequency shift correction is then performed by multiplying $y[i]$ by $e^{-j2\pi\widehat{\epsilon}i/N}$. Time delay correction allows us to determine on which set of samples the Fourier transform must be applied at the receiver, so that the Fourier transform starts at the beginning of the OFDM block.

5.3.6.1. *Time synchronization with preambles*

OFDM frame synchronization at the receiver can be performed in a first training phase, which is based on the usage of blocks of training symbols located at the beginning of the frame and called preambles.

Synchronization with the help of preambles may be quite coarse; it is then preferable to follow it by an adaptive phase in order to counter-balance parameter variations.

In OFDM, the receiver must estimate the beginning of the frame in order to know where to start the discrete Fourier transform, and to estimate the symbol period. Since frequency carrier variations are not yet known, the proposed techniques for time synchronization must be robust to frequency shifts.

The preamble must contain time symbols with a specific repetition pattern. Thus, a time estimator that determines the correlation peak between repetitive patterns is robust.

The solution proposed by Schmidl and Cox [SCH 97] consists of transmitting a block of N preamble symbols, such that the $N/2$ first symbols are equal to the $N/2$ last symbols. We thus assume that a frame is composed of several OFDM blocks, where the first block is the preamble, which has the following pattern: $[B\ B]$, with B a sub-block of symbols of length $N/2$. The preamble is generated in frequency domain before the IFFT, by choosing for even frequency symbols $X^k[2p]$, $p \in \{0, ..., N/2-1\}$ a pseudo-random sequence, and setting all odd frequency symbols $X^k[2p+1]$, $p \in \{0, ..., N/2-1\}$ to zero.

Thus, if the cyclic prefix is longer than the maximum delay of the channel, the received signal corresponding to the $N/2$ last symbols is equal to the received signal corresponding to the $N/2$ first symbols that has just been frequency-shifted because of ϵ. Let $y_{\mathrm{u}}[i]$ be the useful signal at the output of the channel. Then the $N/2$ first received symbols are:

$$y[i] = y_{\mathrm{u}}[i]e^{j2\pi\epsilon i/N} + b[i] \quad \forall i \in \theta \leq N \leq \theta + N/2 - 1 \qquad [5.40]$$

and the $N/2$ last received symbols are:

$$y[i+N/2] = y_{\mathrm{u}}[i]e^{j2\pi\epsilon i/N}e^{j2\pi\epsilon} + b[i+N/2]$$
$$\forall i \in \theta \leq N \leq \theta + N/2 - 1 \qquad [5.41]$$

The normalized cross-correlation between both sequences is expressed as follows:

$$\Gamma_{\text{SC}}(\widetilde{\theta}) = \frac{\sum_{q=\widetilde{\theta}}^{\widetilde{\theta}+N/2-1} y[q+N/2]y^*[q]}{\sum_{q=\widetilde{\theta}}^{\widetilde{\theta}+N/2-1} |y[q+N/2]|^2} \qquad [5.42]$$

The cross-correlation reaches a maximum when it concerns vectors $y[i]$ corresponding to the $N/2$ first symbols and the $N/2$ last symbols of the preamble. Any delay compared to the frame's beginning implies that the considered symbols for the first $y[i]$ or the last $y[i]$ do not completely belong to the preamble: therefore, they are not fully correlated. We can note that if the time delay θ is less than $N/2$, some part of the preamble is taken into account in equation [5.42], and cross-correlation is thus not equal to zero.

The frame's beginning is estimated as the value of $\widehat{\theta}$ that maximizes cross-correlation [5.42]:

$$\widehat{\theta} = \text{arg max}_{\widetilde{\theta}} \left|\Gamma_{\text{SC}}(\widetilde{\theta})\right| \qquad [5.43]$$

This estimator is implemented as follows: the receiver constantly computes the value of [5.42]. As soon as cross-correlation exceeds a given threshold, the receiver estimates that the frame starts. Thus, the precise frame's beginning is evaluated by [5.43].

This technique is relatively simple but may provide rough results because of the frequency shift ϵ and the threshold decision, which may be inaccurate. Some improvements have been proposed. They take advantage of the preamble's structure. Shi and Serpedin's technique [SHI 04] is based on a preamble composed of 4 blocks of time symbols: $[B\;B\;\; -B\;B]$, where sub-block B has length $N/4$. The metric that should be maximized at the receiver depends on cross-correlations of these blocks and is not detailed here due to lack of space. It provides a more accurate estimate of the frame's beginning. This principle can be generalized, as shown in [MIN 03]. Using preambles with more adequately chosen sub-blocks allows us to even more refine synchronization.

5.3.6.2. *Blind time synchronization with cyclic prefix*

The cyclic prefix (see section 5.3.2) may be used to perform time synchronization of OFDM frames at the receiver. This technique was initially proposed by Van de Beek *et al.* [VAN 97]. In this case, preambles are not required, and synchronization may be performed independently per OFDM symbol.

The receiver computes the cross-correlation between the sequence of N_{CP} samples $y[i]$ starting from index p, and the sequence of N_{CP} samples $y[i]$ starting from index $p+N$:

$$\Gamma_{\mathrm{PC}}(p) = \sum_{q=0}^{N_{\mathrm{CP}}-1} y[p+q]y^*[p+q+N] \qquad [5.44]$$

Since the N_{CP} first samples of the OFDM symbol (that is the cyclic prefix) are equal to the N_{CP} last samples of the OFDM symbol, of course subject to some additional noise, estimator [5.44] has a periodic peak, of period $N+N_{\mathrm{CP}}$. Each peak corresponds to the beginning of the OFDM frame (see Figure 5.11, where cyclic prefix is shaded).

$$\widehat{\theta} = \text{arg max}_p \ |\Gamma_{\mathrm{PC}}(p)| \qquad [5.45]$$

In order to reduce the noise's influence, it is better to perform this operation on several successive OFDM symbols and to average the obtained values of $\widehat{\theta}$.

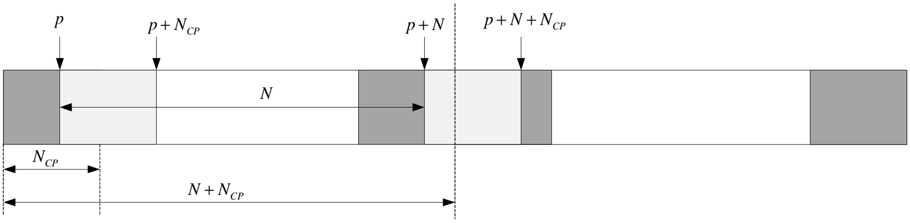

Figure 5.11. *General principle of cyclic prefix time synchronization*

5.3.6.3. *Frequency shift correction with training sequence*

Frequency synchronization is performed after time synchronization has been acquired. It is obtained due to training sequences contained in each frame's preamble. The principle is similar to that of frame synchronization: the training sequence is composed of repeating symbols, so that the signal obtained at the receiver is the same on several sub-blocks of the preamble, except for the frequency shift.

Frequency synchronization may be implemented both in time and frequency domains. In the latter case, the receiver applies its estimate in symbols at the FFT output. This is of course feasible because frame synchronization has already been performed, and consequently the receiver knows on which symbols sequence to apply the FFT.

Moose [MOO 94] proposed an estimator based on two sub-blocks that will be identical after the Fourier transform if there is no frequency shift. Let $Y_i[n]$ be the component on the n^{th} subcarrier of the Fourier transform of the i^{th} sub-block, for $i = \{1, 2\}$, and $Y_{\text{u}}[n]$ the Fourier transform of the first sub-block. For $n \in \{0, ..., N-1\}$, the frequency symbols are:

$$\begin{aligned} Y_1[n] &= Y_{\text{u}}[n] + B_1[n] \\ Y_2[n] &= Y_{\text{u}}[n] e^{j2\pi\epsilon N_T/N} + B_2[n] \end{aligned} \qquad [5.46]$$

ϵ can be estimated by computing the cross-correlation between Y_1 and Y_2 on all components of the FFT:

$$\widehat{\epsilon} = \frac{1}{2\pi N_T/N} \text{ angle} \left(\sum_{n=0}^{N-1} Y_2[n] Y_1^*[n] \right) \qquad [5.47]$$

Estimator [5.47] provides poor results if the frequency shift is high. Indeed, the angle function gives a value in interval $[-\pi, \pi[$, which implies that $|\widehat{\epsilon}| \leq N/(2N_T) \leq 1/2$ and consequently that the estimated absolute frequency shift is located between $-1/(2NT)$ and $1/(2NT)$, which is half of a subcarrier's bandwidth. Thus, the frequency shift estimated by [5.47] will necessarily lie within half of a subcarrier's bandwidth.

In order to overcome this limitation, several techniques have been proposed, among which Schmidl and Cox's method [SCH 97] and also Morelli and Mengali's method [MOR 99]. A important criterion is that the preamble's length must increase for the frequency shift's estimation to be performed on a larger variation area. For more details on these estimators, the readers may refer to [MOR 07].

5.4. FBMC/OQAM

As already studied in section 5.3.5.1, OFDM is highly sensitive to frequency shifts because of the sidelobes of the cardinal sine which has been chosen as transmit waveform. Other multi-carrier modulation techniques that are less sensitive to frequency shifts have been defined. They are based on the use of narrowband sub-bands generated by a filter bank that produces waveforms that are well localized in the frequency domain.

In this section, we detail FBMC modulations FBMC/OQAM. On top of their lower sensitivity to frequency shifts, they present another advantage compared to OFDM: they provide higher spectral efficiencies than OFDM on useful data because they do not require the addition of any cyclic prefix.

This section first presents continuous-time FBMC/OQAM transmission. Then, the equivalent discrete-time notations are introduced. Efficient implementation techniques are mentioned. Finally, the prototype filter's choice and the different solutions proposed up to now in the literature are detailed.

5.4.1. *Principles of continuous-time FBMC/OQAM*

Transmission of an FBMC/OQAM signal is represented in Figure 5.12. Symbols are set on their frequency carrier with width $B_N = 1/T_N$ between subcarriers, similarly to OFDM.

The transmitted symbols do not belong to a classical QAM modulation, but are offset quadrature amplitude modulation (OQAM) symbols [SIO 02, BÖL 03]. The real and imaginary parts of each symbol are time staggered by half an OFDM symbol period, $T_N/2$. Moreover, a time delay of $T_N/2$ is included on QAM symbols' imaginary part on even subcarriers, whereas a time delay of $T_N/2$ is included on QAM symbols' real part on odd subcarriers (see Figure 5.13).

The complex symbol to transmit on subcarrier f_n in time interval $[kT_N, (k+1)T_N[$ is denoted as:

$$c_{n,k} = a_{n,k} + jb_{n,k} \qquad [5.48]$$

where $a_{n,k}$ is the real part of the symbol and $b_{n,k}$ is its imaginary part. In the following, to simplify, we assume that $f_n = n/T_N$, considering baseband transmission ($f_c = 0$).

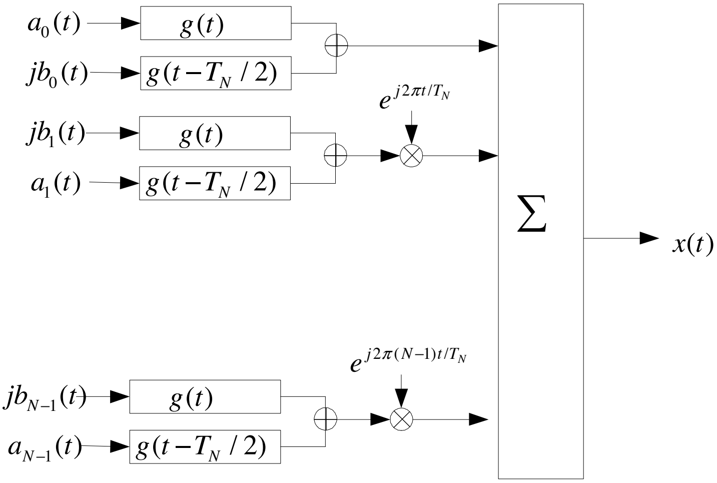

Figure 5.12. *Block diagram of continuous-time FBMC/OQAM transmitter*

The transmitted signal is then:

$$\begin{aligned} x(t) = \sum_{n=0}^{N/2-1} \sum_{k=-\infty}^{+\infty} &(a_{2n,k}g(t-kT_N) \\ &+jb_{2n,k}g\left(t-kT_N-\frac{T_N}{2}\right)\bigg) e^{j\frac{2\pi}{T_N}(2n)t} \\ &+\left(a_{2n+1,k}g\left(t-kT_N-\frac{T_N}{2}\right)\right. \\ &+jb_{2n+1,k}g(t-kT_N)) e^{j\frac{2\pi}{T_N}(2n+1)t} \end{aligned} \quad [5.49]$$

where $g(t)$ is the impulse response of the prototype filter, which will be defined in section 5.4.3. This filter must be real symmetrical and fulfill the following orthogonality constraint:

$$\int_{-\infty}^{+\infty} g\left(t-k_0\frac{T_N}{2}\right) g\left(t-k\frac{T_N}{2}\right)\cos\left(\frac{2\pi}{T_N}(n-n_0)t+\varphi_{n,k}-\varphi_{n_0,k_0}\right)dt$$
$$=\delta_{n_0,n}\delta_{k_0,k} \qquad [5.50]$$

where $\varphi_{n,k}$ is defined as follows :

$$\varphi_{n,k}=\frac{\pi}{2}(n+k)-\pi nk \qquad [5.51]$$

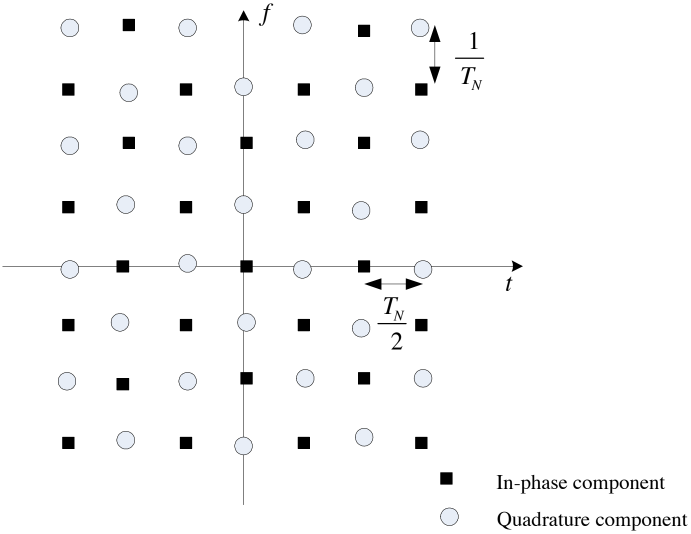

Figure 5.13. *Time and frequency representation of FBMC-OQAM*

We now assume that the number of subcarriers N is even. To simplify $x(t)$'s writing, we introduce real elements $d_{n,k}$, defined as follows:

$$d_{2n,2k}=a_{2n,k}$$
$$d_{2n,2k+1}=b_{2n,k}$$
$$d_{2n+1,2k}=b_{2n+1,k}$$
$$d_{2n+1,2k+1}=a_{2n+1,k} \qquad [5.52]$$

Equation [5.49] thus becomes:

$$x(t) = \sum_{n=0}^{N-1} \sum_{k=-\infty}^{+\infty} d_{n,k} g\left(t - k\frac{T_N}{2}\right) e^{j\frac{2\pi}{T_N}nt} e^{j\varphi_{n,k}}$$
$$= \sum_{n=0}^{N-1} \sum_{k=-\infty}^{+\infty} d_{n,k} \gamma_{n,k}(t) \qquad [5.53]$$

with:

$$\gamma_{n,k}(t) = g(t - k\frac{T_N}{2}) e^{j\frac{2\pi}{T_N}nt} e^{j\varphi_{n,k}} \qquad [5.54]$$

If transmission is ideal, an estimate of d_{n_0,k_0} denoted by $\hat{d}_{n_0,k_0}$ is obtained by taking the real part of the signal at the output of the matched filter. It is equal to:

$$\hat{d}_{n_0,k_0} = \Re\left[\int_{-\infty}^{+\infty} \gamma^*_{n_0,k_0}(t) x(t) dt\right] \qquad [5.55]$$

By replacing [5.53] and [5.54] in equation [5.55], we can show that this estimate is accurate: $\hat{d}_{n_0,k_0} = d_{n_0,k_0}$.

We now assume that the channel generates distortions. Its impulse response $g_c(t)$ has length $L_c - 1$. The l^{th} coefficient is denoted as $g_{c,l}$. The associated delay is $\frac{k_l}{N} T_N$. The channel's impulse response is given by equation [5.56].

$$g_c(t) = \sum_{l=0}^{L_c-1} g_{c,l} \delta\left(t - \frac{k_l}{N} T_N\right) \qquad [5.56]$$

As previously in the perfect channel case, additive noise is neglected. This received signal is thus equal to the result of the filtering of $x(t)$ by the channel,

$y(t) = g_c(t) * x(t)$:

$$y(t) = \sum_{l=0}^{L_c-1} g_{c,l}\delta\left(t - \frac{k_l}{N}T_N\right) * \sum_{n=0}^{N-1}\sum_{k=-\infty}^{+\infty} d_{n,k}\gamma_{n,k}(t)$$

$$= \sum_{n=0}^{N-1}\sum_{k=-\infty}^{+\infty} d_{n,k}e^{j\varphi_{n,k}} \sum_{l=0}^{L_c-1} g_{c,l}g\left(t - k\frac{T_N}{2} - \frac{k_l}{N}T_N\right) e^{j\frac{2\pi}{T_N}n(t-\frac{k_l}{N}T_N)}$$

[5.57]

Similarly to OFDM, we assume that the channel is non-frequency-selective in each sub-band of width $B_N = 1/T_N$. Thus, its frequency response is constant, and can be denoted as:

$$G_c[n] = \sum_{l=0}^{L_c-1} g_{c,l}e^{-j\frac{2\pi}{N}nk_l} \qquad [5.58]$$

In this case, equation [5.57] becomes:

$$y(t) = \sum_{n=0}^{N-1}\sum_{k=-\infty}^{+\infty} G_c[n]d_{n,k}e^{j\varphi_{n,k}}g\left(t - k\frac{T_N}{2}\right) e^{j\frac{2\pi}{T_N}nt} \qquad [5.59]$$

The receiver demodulates the signal transmitted on frequency carrier n in time interval $[kT_N, (k+1)T_N[$ by filtering by the matched filter:

$$Z_{n_0,k_0} = \int_{-\infty}^{+\infty} \gamma^*_{n_0,k_0}(t)y(t)dt$$

$$= \sum_{n=0}^{N-1}\sum_{k=-\infty}^{+\infty} G_c[n]d_{n,k}e^{j(\varphi_{n,k}-\varphi_{n_0,k_0})}$$

$$\times \int_{-\infty}^{+\infty} g\left(t - k\frac{T_N}{2}\right) g\left(t - k_0\frac{T_N}{2}\right) e^{j\frac{2\pi}{T_N}(n-n_0)t}dt \qquad [5.60]$$

If the prototype filter is well localized both in the time and frequency domains, then the double sum in [5.60] can be reduced to a sum on a limited subset, denoted as Ω_{n_0,k_0}.

In addition, if we assume that the channel is at least constant on subset Ω_{n_0,k_0}, then the demodulated signal becomes:

$$\begin{aligned}Z_{n_0,k_0} = G_c[n_0] &\sum_{(n,k)\in\Omega_{n_0,k_0}} d_{n,k} e^{j(\varphi_{n,k}-\varphi_{n_0,k_0})} \\ &\times \int_{-\infty}^{+\infty} g\left(t-(k)\frac{T_N}{2}\right) g\left(t-k_0\frac{T_N}{2}\right) e^{j\frac{2\pi}{T_N}(n-n_0)t} dt \end{aligned} \qquad [5.61]$$

Frequency equalization is then applied on Z_{n_0,k_0} by dividing it by $G_c[n_0]$. Thus, only the real part is kept, since OQAM transmission has been performed. The detected symbol finally is:

$$\begin{aligned}\hat{d}_{n_0,k_0} &= \Re\left[\frac{Z_{n_0,k_0}}{G_c[n_0]}\right] \\ &= \sum_{(n,k)\in\Omega_{n_0,k_0}} d_{n,k} \\ &\int_{-\infty}^{+\infty} g\left(t-(k)\frac{T_N}{2}\right) g\left(t-k_0\frac{T_N}{2}\right) \\ &\quad \times \cos\left(\frac{2\pi}{T_N}(n-n_0)t+\varphi_{n,k}-\varphi_{n_0,k_0}\right) dt \\ &= \sum_{(n,k)\in\Omega_{n_0,k_0}} d_{n,k}\delta_{n_0,n}\delta_{k_0,k} \\ &= d_{n_0,k_0} \end{aligned} \qquad [5.62]$$

where we used orthogonality constraint [5.50] to pass from the second line to the third line. Consequently, in the absence of noise and even if the channel generates distortions, the receiver can accurately recover the transmitted symbols. The initial complex symbols $c_{n,k}$ are eventually obtained by inverting equations [5.52].

5.4.2. *Discrete-time notations for FBMC/OQAM*

In the previous section, we considered continuous-time signals. Yet, similarly to OFDM, discrete-time signals can be obtained by sampling at the symbol period, $T = T_N/N$.

The discrete prototype filter must be causal in the time domain. Its samples are denoted by $g[i]$. In order to restrict its length to L_g, $g(t)$ is truncated in time interval $[-(L_g/2)T, +(L_g/2)T[$. It is then delayed by $((L_g-1)/2)T$ seconds to fulfill the causality constraint. As a result, the i^{th} sample is:

$$g[i] = g\left(\left(i - \frac{L_g-1}{2}\right)T\right) \qquad [5.63]$$

According to equation [5.53], discrete-signal time $x[i]$ is written as follows [SIO 02]:

$$\begin{aligned} x[i] &= x\left(\left(i - \frac{L_g-1}{2}\right)T\right) \\ &= \sum_{n=0}^{N-1}\sum_{k\in\mathbf{Z}} d_{n,k}g\left[i - k\frac{N}{2}\right] e^{j\frac{2\pi}{N}n(i-\frac{D}{2})}e^{j\varphi_{n,k}} \end{aligned} \qquad [5.64]$$

where $D = L_g - 1$

Equation [5.64] can be written in a simpler way:

$$x[i] = \sum_{n=0}^{N-1}\sum_{k\in\mathbf{Z}} d_{n,k}g_{n,k}[i], \qquad [5.65]$$

where $g_{n,k}[i]$ is a version of $g[i]$ that has been shifted both in time and frequency.

The symbol in the n_0^{the} subcarrier and in time interval k_0 is obtained by computing the scalar product of $x[i]$ and $g_{n_0,k_0}[i]$. If the channel is perfect, we get:

$$Z_{n_0,k_0} = \langle x, g_{n_0,k_0} \rangle = \sum_{i=-\infty}^{+\infty} x[i] g^*_{n_0,k_0}[i]$$
$$= \sum_{i=-\infty}^{+\infty} \sum_{n=0}^{N-1} \sum_{k \in \mathbf{Z}} d_{n,k} g_{n,k}[i] g^*_{n_0,k_0}[i]. \qquad [5.66]$$

The orthogonality constraint [5.50] implies that:

$$\Re\left[\sum_{i=-\infty}^{+\infty} g_{n_0,k_0}[i] g^*_{n,k}[i]\right] = \delta_{n,n_0}\delta_{k,k_0} \qquad [5.67]$$

If the channel has frequency response G_c and we still neglect the noise, demodulation becomes:

$$Z_{n_0,k_0} = G_c[n_0] d_{n_0,k_0} + \underbrace{\sum_{(n,k)\neq(n_0,k_0)} G_c[n] d_{n,k} \sum_{i=-\infty}^{+\infty} g_{n,k}[i] g^*_{n_0,k_0}[i]}_{I_{n_0,k_0}} \qquad [5.68]$$

where I_{n_0,k_0} is an intrinsic interference term.

We assume, as we previously did, that most of the impulse response's energy is localized in a limited subset around the studied symbol (n_0, k_0). Then the demodulated signal is:

$$Z_{n_0,k_0} \approx G_c[n_0]\left(d_{n_0,k_0} + \underbrace{\sum_{(n,k)\in\Omega_{n_0,k_0}} d_{n,k} \Gamma_{n-n_0,k-k_0}}_{\hat{I}_{n_0,k_0}} \right) \qquad [5.69]$$

where the impulse response of transmultiplexer $\Gamma_{n-n_0,k-k_0}$ is equal to:

$$\Gamma_{n-n_0,k-k_0} = \sum_{i=-\infty}^{\infty} g_{n,k}[i] g^*_{n_0,k_0}[i] \qquad [5.70]$$

Since all symbols $d_{n,k}$ are real and the orthogonality constraint [5.67] holds, the intrinsic interference term $\hat{I}_{n_0,k_0}$ is completely imaginary. Thus, and similarly to the continuous-time case, after frequency equalization and OQAM decision, the receiver can recover the transmitted symbol: $\hat{d}_{n_0,k_0} = d_{n_0,k_0}$.

Polyphase implementation of FBMC/OQAM filter banks is detailed in Appendix B.

5.4.3. *Prototype filter*

In FBMC/OQAM, the prototype filter must be chosen with particular care. This indeed provides adaptation opportunities compared to OFDM: prototype filters may, for instance, be built in order to verify given objectives, such as time and frequency locations, regularity and so forth [SIO 00].

Nevertheless, the prototype filter must still fulfill the orthogonality constraint expressed in continuous-time by equation [5.50], and which can also be expressed in discrete time by some constraints on the polyphase components of the prototype filter, as explained in [SIO 02].

This section reviews some examples of prototype functions. Time and frequency equations are all normalized by ratio t/T_N and $f/B_N = fT_N$, respectively on all figures.

5.4.3.1. *Extended Gaussian function (EGF)*

This class of prototype filter arises from an orthogonalization algorithm in two steps which is applied on the Gaussian function [LEF 95, ALA 96].

EGF is defined in time domain by:

$$g_{\alpha,\nu_0,\tau_0}(t) = \frac{1}{2}\sum_{k=0}^{\infty} q_{k,\alpha,\nu_0}\left[f_\alpha\left(t+\frac{k}{\nu_0}\right)+f_\alpha\left(t-\frac{k}{\nu_0}\right)\right]\sum_{l=0}^{\infty} q_{l,1/\alpha,\tau_0}\cos\left(2\pi l\frac{t}{\tau_0}\right) \qquad [5.71]$$

where $f_\alpha(t) = (2\alpha)^{1/4}\exp^{-\pi\alpha t^2}$ and α is the roll-off factor. The detailed computation of real coefficients q_{k,α,ν_0} is provided in [SIO 00].

EGFs have initially been determined in [LEF 95] using isotropic orthogonal transform algorithm (IOTA). The IOTA prototype filter is obtained by setting $\nu_0 = \tau_0 = 1/\sqrt{2}$ and $\alpha = 1$.

One particularity of the IOTA prototype filter is that it is identical to its Fourier transform [LEF 95]. It is consequently similarly localized both in frequency and time domains.

5.4.3.2. *PHYDYAS prototype filter*

The prototype filter proposed by Bellanger in [BEL 01, D5. 09] is based on a frequency sampling technique.

The analytical formulas to compute this filter's coefficients provide some flexibility on the choice of many parameters: number of subcarriers N, overlapping factor K and roll-off factor α. The latter is generally chosen as $\alpha = 1$, which implies that the transition band of a sub-band stops at the center of the adjacent sub-band. Consequently, only directly adjacent subcarriers may interact with each other and induce non-negligible interferences.

The filter design starts with choosing $L_G = KN$ values for $G(k/L_G)\ \forall k \in \{0, ..., L_G - 1\}$ in frequency domain. When $K = 4$, they are set to:

$$\begin{aligned}
&G_0 = 1, G_1 = 0.971960\\
&G_2 = 1/\sqrt{2}\\
&G_3 = \sqrt{1 - G_1^2}\\
&G_k = 0 \quad \forall k \in \{4, ..., L - 1\}
\end{aligned} \qquad [5.72]$$

Then the prototype filter's coefficients are obtained by applying the IFFT:

$$g(t) = \begin{cases} \frac{1}{\sqrt{A}} \left[1 + 2 \sum_{k=1}^{K-1} (-1)^k G_k \cos\left(\frac{2\pi}{KT_N} kt\right)\right] & \text{if } t \in [0, KT_N] \\ 0 & \text{elsewhere} \end{cases} \qquad [5.73]$$

where A is a normalization factor:

$$A = \int_0^{KT_N} \left[1 + 2 \sum_{k=1}^{K-1} (-1)^k G_k \cos\left(\frac{2\pi}{KT_N} kt\right)\right]^2 dt = KT_N \left[1 + 2 \sum_{k=1}^{K-1} G_k^2\right] \qquad [5.74]$$

The frequency and time responses of PHYDYAS prototype filter are represented in Figure 5.14.

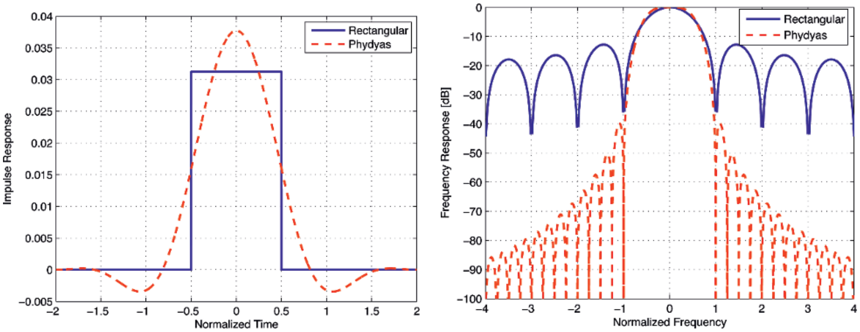

Figure 5.14. *Impulse response and frequency response of PHYDYAS prototype filter*

The transmultiplexer's response for PHYDYAS prototype filter is given by Tables 5.1 and 5.2.

	n_0-4	n_0-3	n_0-2	n_0-1	n_0
k_0-1	$0,0054$	$0,0429j$	$-0,1250$	$-0,2058j$	$0,2393$
k_0	0	$-0,0668$	$0,0002$	$0,5644$	1
k_0+1	$0,0054$	$-0,0429j$	$-0,1250$	$0,2058j$	$0,2393$

Table 5.1. *Transmultiplexer's response for PHYDYAS prototype filter, part 1*

	n_0+1	n_0+2	n_0+3	n_0+4
k_0-1	$0,2058j$	$-0,1250$	$-0,0429j$	$0,0054$
k_0	$0,5644$	$0,0002$	$-0,0668$	0
k_0+1	$-0,2058j$	$-0,1250$	$0,0429j$	$0,0054$

Table 5.2. *Transmultiplexer's response for PHYDYAS prototype filter, part 2*

5.5. Conclusion

This chapter presented the basis of multi-carrier modulations. These modulations aim at transmitting data on a frequency-selective channel without creating any inter-symbol interference at the receiver, by splitting the wideband channel into a set of narrowband subchannels, which are all non-frequency-selective.

The general principles of multi-carrier modulations were first introduced. They can be non-overlapping in frequency domain, in which case they inherently do not generate any interference between subcarriers. Yet, overlapping may be allowed at transmission, if the transmission basis is orthonormal: in this case, interferences are removed at the receiver. This technique decreases the required bandwidth and thus increases spectral efficiency. It is used both in OFDM and FBMC.

OFDM was then detailed. After describing the transmit and receive techniques with cyclic prefix, we considered optimum power allocation and the main drawbacks of OFDM, namely PAPR and its sensitivity to asynchronous transmissions. Then, we presented some synchronization algorithms that are specific to OFDM and that complete the algorithms from Chapter 4.

Finally, OQAM FBMC modulations were introduced. Although FBMC/OQAM is more complex to implement than OFDM, its main advantage is that it is far less sensitive to asynchronicity because the prototype filter is well localized in frequency. FBMC and other filter bank modulations are currently an important field of research because they are potential candidates for future digital transmission systems.

5.6. Exercises

5.6.1. *Exercise 1*

A user is moving in a high-speed train. It wants to use its fourth-generation mobile phone based on OFDM, on central frequency carrier 2.6 GHz. Its total bandwidth is $B = 20$ MHz. OFDM is used with $N = 2048$ points in Fourier transform. The maximum delay of the channel is $T_m = 1 \mu s$.

1) Compute the bandwidth of an OFDM sub-band.

2) Indicate if the channel is frequency-selective in an OFDM sub-band.

3) We would like to estimate the channel's gain in each OFDM sub-band. Compute the maximum speed of the train that allows us to have a channel which is non-time-selective, considering the bandwidth of an OFDM sub-band.

4) Conclude: can the fourth generation be used in high-speed trains? Would the same conclusion stand if the central frequency was 800 MHz?

5) Evaluate the cyclic prefix duration both in time and samples number.

6) Evaluate the OFDM symbol size in samples number.

7) Compute the overhead, which is equal to the ratio of cyclic prefix in an OFDM symbol.

8) Compute the binary data rate if the chosen modulation is 64-QAM.

5.6.2. *Exercise 2*

OFDM is used in bandwidth $B = 5$ MHz with a 512 point Fourier transform. The maximum delay of the channel is $T_m = 10\mu s$. The cyclic prefix duration is $T_{\text{CP}} = 17\mu s$.

1) Is the channel per sub-band frequency-selective?

2) Evaluate the OFDM symbol duration.

3) Compute the overhead, which is equal to the ratio of cyclic prefix in an OFDM symbol.

4) The chosen modulation is 16-QAM and an error-correction code of rate $2/3$ is used. Compute the binary data rate.

6

Coded Modulations

This chapter presents coding techniques well adapted to high signal-to-noise ratio regimes. We showed in Chapter 3 that when the signal-to-noise ratio is high, large-order quadrature amplitude modulation (QAM) constellations allow us to achieve high spectral efficiencies. Squared QAM constellations, when $M = 2^m$, are regular in the two-dimensional space $\mathbb{Z}^2$. They are based on the repetition of a reference pattern. This principle can be extended to constellations in higher dimensions than 2. A constellation may be defined as a repetitive infinite grid of points, restricted to a limited area. These repetitive grids are mathematically based on lattices. For instance, squared QAM all follow the same grid, but they are differentiated by their boundary, which gets larger when the size of the constellation increases.

If error-correcting codes are used separately from modulations, they lead to an increase in the required bandwidth, due to the introduced redundancy. Coded modulations combine channel encoding with modulations in order to maintain the same bandwidth, while improving performances. Redundancy is introduced by the use of a larger size constellation than what would be necessary without coding. For instance, let us assume that we want to transmit 2 bits per symbol. Without channel encoding, the symbol period is denoted by T and the bandwidth by B, using quadrature phase shift keying (QPSK) modulation. With channel encoding of rate $2/3$ and the same modulation, the symbol period becomes $2T/3$, and the bandwidth increases to $3B/2$. If on the contrary, the same channel encoding of rate $2/3$ is used with an 8-PSK modulation with 3 bits per symbol, the symbol period and the bandwidth remain T and B, respectively. However, 8-PSK modulation

Digital Communications 2: Digital Modulations, First Edition. Mylène Pischella and Didier Le Ruyet.

requires a higher E_b/N_0, and thus a transmit power increase, to achieve the same performances in terms of bit error rate than QPSK. This loss may be compensated for by considering not only one 8-PSK symbol, but also a sequence of symbols, and compelling transitions between symbols so that the achieved minimum distance increases.

Consequently, the successive constellation points are jointly connected, and only some sequences of symbols are authorized. When a codeword is composed of a symbol's sequence of finite size, the coded modulation is a block-coded modulation. In this case, joint coding and modulation create a bounded constellation based on a lattice of finite dimension with good characteristics in terms of minimum distance and density. On the contrary, if the transmitted symbols belong to infinite-length sequences, the coded modulation is a trellis-coded modulation. Transition constraints from one trellis state to another allow us to increase the minimum distance. We can directly make an analogy between these two types of coded modulations and block codes and convolutional codes: block-coded modulations are built on finite-dimensional block codes, whereas trellis-coded modulations are built on infinite-dimensional convolutional codes.

In this chapter, we first study the mathematical properties of lattices and their main characteristics, by giving some examples of the most useful lattices. Then, section 6.2 details block-coded modulations, and more specifically block-coded modulations that generate binary lattices. Section 6.3 deals with trellis-coded modulations.

NOTE.– The main notions introduced in this chapter, as well as the used notations, mainly arise from Forney [FOR 84, FOR 88a, FOR 88b] and Ungerboeck [UNG 82], and from the reference book by Conway and Sloane [CON 99].

6.1. Lattices

6.1.1. *Definitions*

In the first volume [LER 15] we saw that in order to approach the additive white Gaussian noise channel capacity, it is necessary to build a dense packing of spheres within a sphere of higher dimension. In practice, the packing should be well structured.

A lattice is a discrete set of vectors of dimension N that has a group structure. In order to generalize the reasoning to any dimension, in the remainder of the chapter we no longer refer to spheres but to points with given volumes.

6.1.1.1. *General definition of lattices*

A lattice Λ is a discrete subgroup of rank N of $\mathbb{R}^N$. It includes point $\mathbf{0}$, and for any point $\boldsymbol{\lambda}$ and Λ, $-\boldsymbol{\lambda}$ belongs to Λ. Thus, all points of lattice Λ uniformly occupy space $\mathbb{R}^N$.

The set of integers on $\mathbb{R}^N$, $\mathbb{Z}^N$, is a discrete subgroup of $\mathbb{R}^N$. An integer lattice is a sublattice of $\mathbb{Z}^N$.

Since lattice Λ is a group, it itself contains subgroups and can be partitioned in them. Moreover, the geometrical properties of Λ are the same as those of the group it belongs to: thus, since $\Lambda \subset \mathbb{R}^N$, the distance between any two points of Λ is equal to the Euclidean distance, and volumes are defined similarly as in $\mathbb{R}^N$.

6.1.1.2. *Influence of mathematical operations on lattices*

If r is a scalar in $\mathbb{R}$, the set $r\Lambda$ is a lattice that contains all multiples $r\boldsymbol{\lambda}$ of vectors $\boldsymbol{\lambda}$ and Λ. Scaling operation consequently generates another lattice. This property may be generalized to any orthogonal scaling operation T in the N-dimensional space. Thus, $T\Lambda$ is a lattice containing the set of transformations of vectors $\boldsymbol{\lambda}$ of Λ, denoted by $T\boldsymbol{\lambda}$.

An example of such an operation is rotation $\mathbf{R}$. It is defined in $\mathbb{R}^2$ by the following matrix:

$$\mathbf{R} = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \qquad [6.1]$$

$\mathbf{R}\mathbb{Z}^2$ is the lattice obtained from $\mathbb{Z}^2$ by applying a $\pi/4$ rotation and multiplying by $\sqrt{2}$. Since all points of $\mathbf{R}\mathbb{Z}^2$ belong to $\mathbb{Z}^2$, $\mathbf{R}\mathbb{Z}^2$ is a sublattice of $\mathbb{Z}^2$. It is represented by the set of black points in Figure 6.1.

The Cartesian product of dimension M of a lattice Λ is another lattice, denoted by Λ^M, whose elements are MN-dimensional vectors, $(\boldsymbol{\lambda}_1, ..., \boldsymbol{\lambda}_M)$, where each $\boldsymbol{\lambda}_j$ belongs to Λ. For instance, $\mathbb{Z}^N$ is the Cartesian product of dimension N of $\mathbb{Z}$.

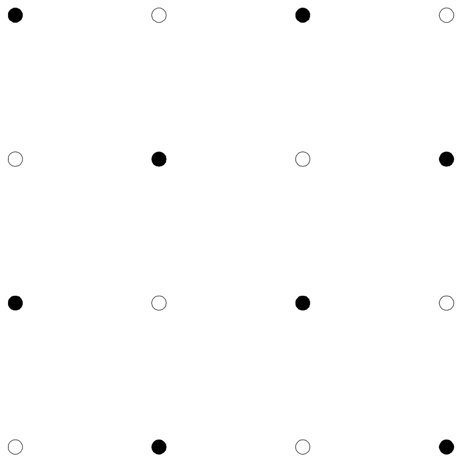

Figure 6.1. *$\mathbb{Z}^2$ lattice and its partition $\mathbf{R}\mathbb{Z}^2$*

6.1.1.3. *Generator matrix*

Any lattice may be characterized by a set of vectors $\mathbf{g}_1, \mathbf{g}_2, \dots, \mathbf{g}_N$ that generate the columns of $N \times N$-dimensional matrix $\mathbf{G}$:

$$\Lambda = \{\mathbf{aG} = \sum_{j=1}^{N} a_j \mathbf{g}_j | \mathbf{a} \in \mathbb{Z}^N\} \qquad [6.2]$$

$\mathbf{G}$ is the generator matrix of Λ.

The region $P = \{\mathbf{aG} = \sum_{j=1}^{N} a_j \mathbf{g}_j | a_j \in \{0, 1\} \ \forall j \in \{1, ..., N\}\}$ is the lattice's Voronoï region.

The fundamental volume $vol(\Lambda)$, also denoted by $\det(\Lambda)$, is the volume of the Voronoï region of Λ. We get:

$$vol(\Lambda) = |\det(\mathbf{G})| \qquad [6.3]$$

since $|\det(\mathbf{G})|$ is the modulus of the vector product of all N vectors of the basis generating Λ, $\mathbf{g}_1 \wedge \mathbf{g}_2 \wedge ... \wedge \mathbf{g}_N$.

For example, the hexagonal lattice $A_2 = \{a(1,0) + b(\frac{1}{2}, \frac{\sqrt{3}}{2}) | (a,b) \in \mathbb{Z}^2\}$, represented in Figure 6.2, is the densest lattice in $\mathbb{R}^2$. Its generator matrix is equal to:

$$\mathbf{G} = \begin{pmatrix} 1 & 0 \\ \frac{1}{2} & \frac{\sqrt{3}}{2} \end{pmatrix} \qquad [6.4]$$

Its fundamental volume is $vol(A_2) = |\det(\mathbf{G})| = \frac{\sqrt{3}}{2}$.

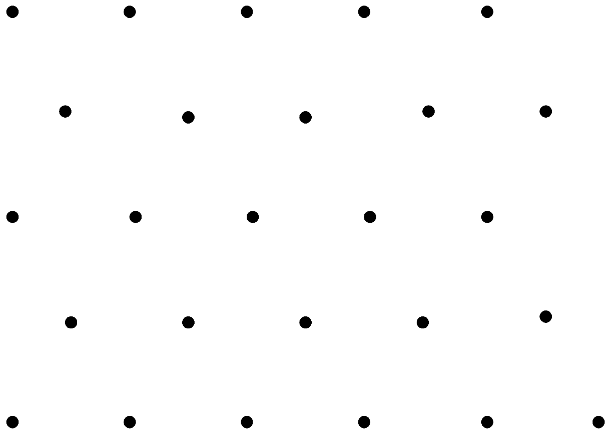

Figure 6.2. *Hexagonal lattice A_2*

6.1.1.4. *Voronoï region*

For any point $\boldsymbol{\lambda} \in \Lambda$, the Voronoï region of $\boldsymbol{\lambda}$ is defined as follows:

$$V(\boldsymbol{\lambda}) = \{\mathbf{x} \in \mathbb{R}^N | \,\forall\, \mathbf{y} \in \Lambda, \; ||\mathbf{x} - \boldsymbol{\lambda}|| \le ||\mathbf{x} - \mathbf{y}||\} \qquad [6.5]$$

where $||\mathbf{a}||$ is the Euclidean distance of $\mathbf{a}$ in $\mathbb{R}^N$. Since the lattice is an additive group, all Voronoï regions of the lattice have the same volume. Thus, it is by definition chosen as the volume at $\mathbf{0}$. It is equal to the fundamental volume of the lattice: $vol(V(\mathbf{0})) = vol(\Lambda)$.

An example of a Voronoï region is given for lattice A_2 in Figure 6.3.

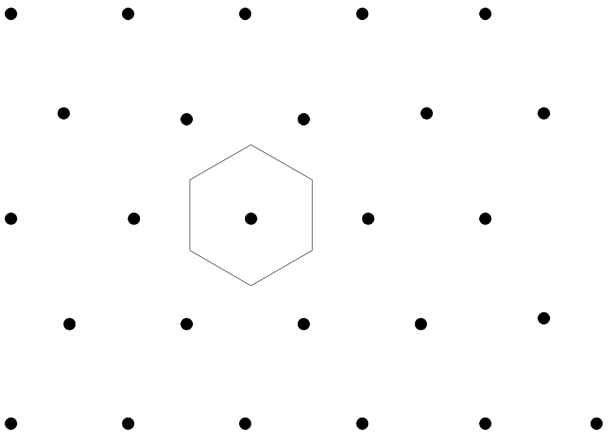

Figure 6.3. *Voronoï region of lattice A_2*

6.1.1.5. *Minimum distance and kissing number*

The minimum distance d_{min} of lattice Λ is equal to the minimum distance between any two different points of Λ.

The number of points located at minimum distance from any lattice point is called the kissing number and is denoted by $K_{min}(\Lambda)$.

It is, for instance, equal to 6 for the hexagonal lattice A_2, since any point has 6 neighbors at minimum distance $d_{\text{min}} = 1$.

6.1.1.6. *Packing and overlapping radiuses*

The lattice's packing radius ρ is the radius of the largest inscribed sphere in the Voronoï region.

The lattice's overlapping radius R is the radius of the largest circumscribed sphere in the Voronoï region.

The minimum distance d_{min} is equal to the distance between two points of two adjacent spheres of the lattice. Consequently, $d_{\text{min}} = 2\rho$.

6.1.1.7. *Lattice density and fundamental coding gain*

The density Δ of lattice Λ is equal to the ratio between the volume of the sphere of radius ρ and the fundamental volume:

$$\Delta = \frac{V_N \rho^N}{vol(\Lambda)} \qquad [6.6]$$

where V_N is the volume of a unity-radius sphere. It is equal to:

$$\begin{aligned} V_N &= \frac{\pi^{N/2}}{\Gamma(N/2+1)} \\ &= \begin{cases} \frac{\pi^{N/2}}{(N/2)!} \text{ if N is even} \\ \frac{2^N \pi^{(N-1)/2}((N-1)/2)!}{(N)!} \text{ if N is odd} \end{cases} \end{aligned} \qquad [6.7]$$

The lattice's density is hard to evaluate for high-dimensional lattices.

The fundamental coding gain, also called Hermite's parameter, allows us to estimate the lattice's density but is easier to compute. It is defined as follows:

$$\gamma_c(\Lambda) = \frac{{d_{\min}}^2}{(vol(\Lambda))^{2/N}} \qquad [6.8]$$

Density and coding gain are related by the following formula:

$$\Delta = \frac{V^N}{2^N} \gamma_c(\Lambda)^{N/2} \qquad [6.9]$$

Lattices must be as dense as possible. For instance, the fundamental coding gain of $\mathbb{Z}^N$ is 1, whereas that of lattice A_2 is $\gamma_c(A_2) = 2/\sqrt{3} \approx 1.1547$. The number of lattice points per volume unit is $1/vol(\Lambda)$. Consequently, if we consider several lattices with the same minimum distance, the lattice with the highest fundamental coding gain is the densest: it contains more points per volume unit. This lattice will thus achieve a higher spectral efficiency.

6.1.2. *Group properties of a lattice*

Many properties arise from the group structure of lattices. In this section, we only list the most useful properties for coded modulations.

From lattice Λ in $\mathbb{R}^N$, we may define a coset translated of $\mathbf{c}$, where $\mathbf{c}$ is a point in $\mathbb{R}^N$. The coset denoted by $\Lambda + \mathbf{c}$ contains all elements $\boldsymbol{\lambda} + \mathbf{c}$, where $\boldsymbol{\lambda}$ belongs to Λ. It is consequently equal to lattice Λ translated of vector $\mathbf{c}$.

Moreover, two N-dimensional elements are equivalent modulo Λ if their difference is in Λ. As a result, coset $\Lambda + \mathbf{c}$ is the set of points equivalent to $\mathbf{c}$ modulo Λ. $\mathbf{c}$ is called the coset leader of this coset.

As previously mentioned, lattice Λ contains one or several sublattices. A sublattice, denoted by Λ', is a subset of Λ and is a lattice itself. A sublattice Λ' leads to partitioning the initial lattice into equivalence classes modulo Λ'. This partition is denoted by Λ/Λ'. It contains $|\Lambda/\Lambda'|$ equivalence classes. The number of equivalence classes is called the partition's order. Each equivalence class is a coset of Λ'.

For example, we saw in section 6.1.1.2 that $\mathbf{R}\mathbb{Z}^2$ is a sublattice of $\mathbb{Z}^2$ that contains one every two points of $\mathbb{Z}^2$. The partition's order $\mathbb{Z}^2/\mathbf{R}\mathbb{Z}^2$ is thus equal to 2. The first coset is $\mathbf{R}\mathbb{Z}^2$, whose coset leader is point of coordinates $(0, 0)$. The second coset is $\mathbf{R}\mathbb{Z}^2 + (1, 0)$, represented by white points in Figure 6.1, whose coset leader is $(1, 0)$. The set $\mathbb{Z}^2$ is the union of both equivalence classes.

Each coset is identified by its coset leader. The set of $|\Lambda/\Lambda'|$ coset leaders is denoted by $[\Lambda/\Lambda']$. Any point in Λ can uniquely be written as :

$$\boldsymbol{\lambda} = \boldsymbol{\lambda}' + \mathbf{c} \qquad [6.10]$$

where $\boldsymbol{\lambda}' \in \Lambda'$ and $\mathbf{c}$ is the coset leader of the class in which $\boldsymbol{\lambda}$ belongs. Decomposition into cosets can be written as follows:

$$\Lambda = \Lambda' + \left[\Lambda/\Lambda'\right] \qquad [6.11]$$

Continuing on $\mathbb{Z}^2$ example, we can write that this lattice is composed of the union of $\mathbf{R}\mathbb{Z}^2 + (0, 0)$ and $\mathbf{R}\mathbb{Z}^2 + (1, 0)$.

Since only one point among $|\Lambda/\Lambda'|$ points of Λ belongs to Λ', the fundamental volume of a sublattice is deduced from that of its originating lattice as follows:

$$vol(\Lambda') = |\Lambda/\Lambda'| vol(\Lambda) \qquad [6.12]$$

Thus, the volume of $\mathbf{R}\mathbb{Z}^2$ is twice that of $\mathbb{Z}^2$, since it contains twice fewer points. This property will be useful later to evaluate the density of binary lattices.

Partitioning may be extended to a sublattice chain. Let us take as an example the one-dimensional set of integers multiples of 2^n, denoted as $2^n\mathbb{Z}$, with $n \in \mathbb{Z}$. Sublattice $2^n\mathbb{Z}$ contains all integers equal to 0 modulo 2^n. It can be partitioned into two sublattices:

– the set of integers equal to 0 modulo 2^{n+1}, denoted as $2^{n+1}\mathbb{Z}$;

– the set of integers equal to 2^n modulo 2^{n+1}, denoted as $2^{n+1}\mathbb{Z} + 2^n$.

Since this property holds true for any $n \in \mathbb{Z}$, the partition chain $\mathbb{Z}/2\mathbb{Z}/4\mathbb{Z}/...$ is infinite.

Integers m are represented by a binary partition chain by simply writing them as:

$$m = a_0 + 2a_1 + 4a_2 + ... \qquad [6.13]$$

where each element a_n is equal to either 0 or 1. $2^n a_n$ indicates in which coset m belongs in partition $2^n\mathbb{Z}/2^{n+1}\mathbb{Z}$. Thus:

– if $a_n = 0$, m belongs to $2^{n+1}\mathbb{Z}$, whose coset leader is 0;

– if $a_n = 1$, m belongs to $2^{n+1}\mathbb{Z} + 2^n$, whose coset leader is 2^n.

Binary decomposition of order 3 is detailed in Figure 6.4. The label associated with each integer is equal to $\{a_0 a_1 a_2\}$. We will see later that this labeling follows Ungerboeck's method.

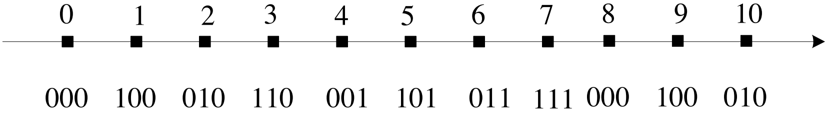

Figure 6.4. *Binary decomposition of order* 3

More generally, a partition chain is a sequence of lattices where each lattice is included in the previous lattice. For instance, the $\Lambda/\Lambda'/\Lambda''$ chain is written as follows:

$$\Lambda = \Lambda'' + \left[\Lambda'/\Lambda''\right] + \left[\Lambda/\Lambda'\right] \qquad [6.14]$$

6.1.3. *Lattice classification*

6.1.3.1. *Complex lattice*

An N-dimensional complex lattice Λ is a discrete group of $\mathbb{C}^N$. Any N-dimensional complex lattice is isomorphic to a $2N$-dimensional real lattice.

Thus, the complex lattice $\mathbb{G}$ is isomorphic to $\mathbb{Z}^2$: for any couple of reals (a, b), the point of coordinates (a, b) in $\mathbb{Z}^2$ is equivalent to point $a + jb$ in $\mathbb{G}$. $\mathbb{G}$ is called the Gaussian integers' set. Its invertible elements are ± 1 and $\pm j$.

A Gaussian integer is prime if it can only be divided by itself and by the invertible elements. The prime Gaussian integer of the lowest modulus is $\Phi = 1 + j$, whose squared modulus is equal to 2.

$\Phi\mathbb{G}$ is a sublattice of $\mathbb{G}$ of order $|\Phi|^2 = 2$, which is equivalent to the real lattice $\mathbf{R}\mathbb{Z}^2$. Partition $[\mathbb{G}/\Phi\mathbb{G}]$ contains two coset leaders: 0 (which represents $\Phi\mathbb{G}$) and 1 (which represents $\Phi\mathbb{G} + 1$). Figure 6.1 illustrates these two sublattices, using the analogy with $\mathbf{R}\mathbb{Z}^2$.

This example can be generalized: for any natural number u, $\Phi^u\mathbb{G}$ is a sublattice $\mathbb{G}$ of order $|\Phi|^{2u} = 2^u$ which is equivalent to the real lattice $\mathbf{R}^u\mathbb{Z}^2$. Yet, $\mathbf{R}^u\mathbb{Z}^2$ is itself equal to $2^{u/2}\mathbb{Z}^2$ if u is even and to $2^{(u-1)/2}\mathbf{R}\mathbb{Z}^2$ if u is odd.

$\Phi^u\mathbb{G}$ is the set of integers whose squared modulus is multiple of 2^u. Its minimum distance is ${d_{\text{min}}}^2 = 2^u$.

The partition chain $\mathbb{G}/\Phi\mathbb{G}/\Phi^2\mathbb{G}/\Phi^3\mathbb{G}/...$ is infinite. Each partition $\Phi^n\mathbb{G}/\Phi^{n+1}\mathbb{G}$ has two equivalence classes. This partition chain is equivalent to the real partition chain based on $\mathbf{R}\mathbb{Z}^2$, as indicated below:

$$\begin{aligned} &\mathbb{G}/\Phi\mathbb{G}/\Phi^2\mathbb{G}/\Phi^3\mathbb{G}/... \\ \Leftrightarrow\ &\mathbb{Z}^2/\mathbf{R}\mathbb{Z}^2/2\mathbb{Z}^2/2\mathbf{R}\mathbb{Z}^2/4\mathbb{Z}^2/... \end{aligned} \qquad [6.15]$$

6.1.3.2. *Binary lattices*

An N-dimensional real lattice is binary if it is a subgroup of $\mathbb{Z}^N$ that contains at least one sublattice $2^m\mathbb{Z}^N$, where m is an integer. The lattice's binary depth is equal to the lowest m fulfilling this characteristic. Consequently, Λ is included in the partition chain $\mathbb{Z}^N/\Lambda/2^m\mathbb{Z}^N$.

The binary depth of the most usual lattices is equal to 1 or 2. A lattice with binary depth equal to one is called a mod-2 lattice, whereas a lattice with binary depth equal to two is called a mod-4 lattice.

The redundancy of a binary lattice, $r(\Lambda)$, is defined as follows:

$$|\mathbb{Z}^N/\Lambda| = 2^{r(\Lambda)} \qquad [6.16]$$

From equation [6.12], $vol(\Lambda) = 2^{r(\Lambda)}vol(\mathbb{Z}^N) = 2^{r(\Lambda)}$. As a result, the fundamental coding gain [6.8] of a binary lattice is equal to:

$$\gamma_c(\Lambda) = 2^{-2/Nr(\Lambda)}{d_{\text{min}}}^2 = 2^{-\rho(\Lambda)}{d_{\text{min}}}^2 \qquad [6.17]$$

where $\rho(\Lambda)$ is the normalized redundancy (per two dimensions), $\rho(\Lambda) = 2/Nr(\Lambda)$.

Equation [6.17] indicates that the coding gain brought by the minimum distance's increase when the lattice's depth gets higher is somewhat counterbalanced by the $2^{-\rho(\Lambda)}$ term. This factor depends on the increase in fundamental volume of the constellation: the constellation points are more distant, which implies that the required energy to transmit them is higher.

We finally consider the labeling of binary lattices according to Ungerboeck's rule. Let Λ be a binary lattice and Λ' its sublattice. The partition order is $|\Lambda/\Lambda'| = 2^K$, with K an integer. The rule that associates a set of K binary integers $\mathbf{a} = \{a_1, a_2, ...a_K\}$ with a coset of Λ' is called labeling. It should not be ambiguous. The label is denoted as $c(\mathbf{a})$.

By definition, we have $c(\mathbf{0}) = \mathbf{0}$, which implies that the $\mathbf{0}$ label corresponds to Λ'.

We can show that there exists a set of lattices $\Lambda_0 = \Lambda, \Lambda_1, ..., \Lambda_K = \Lambda'$ that can be broken up into a partition chain $\Lambda_0/\Lambda_1/.../\Lambda_K$, such that each partition Λ_k/Λ_{k+1} has two cosets, for any $k \in \{0, ..., K-1\}$. Thus, Λ can be decomposed as follows:

$$\Lambda = \Lambda' + \left\{ \sum_{k=0}^{K-1} a_k \mathbf{g}_k \right\} \qquad [6.18]$$

with $a_k \in \{0, 1\}$, and the set $\{a_k \mathbf{g}_k\}$ represents the coset leaders of $[\Lambda_k/\Lambda_{k+1}]$, for any $k \in \{0, ..., K-1\}$.

For instance, for the infinite binary partition chain $\mathbb{Z}/2\mathbb{Z}/4\mathbb{Z}/...$, Ungerboeck's labeling is simply equal to the binary decomposition of integers [6.13], which has been illustrated in Figure 6.4.

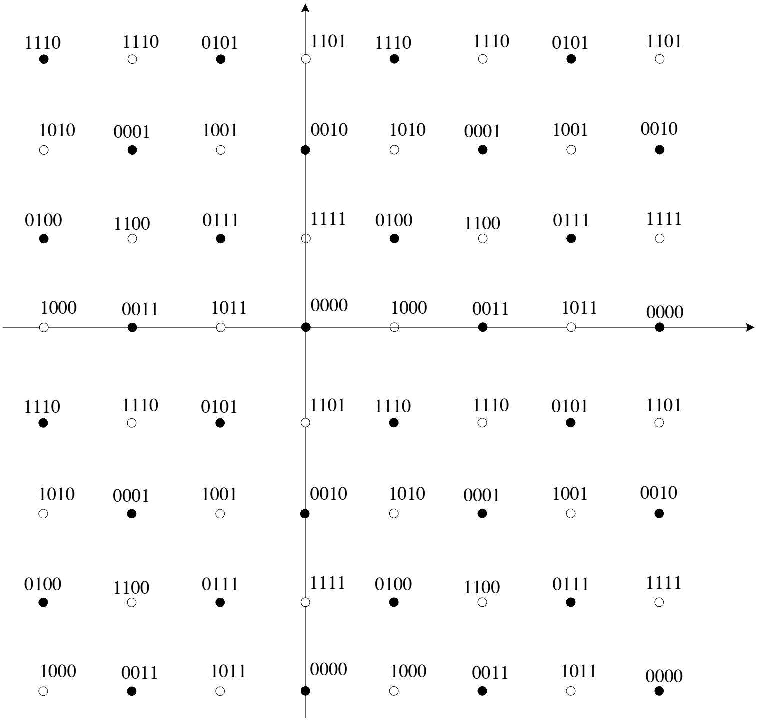

Figure 6.5. *Order-4 Ungerboeck's labeling of* $\mathbb{Z}^2$

Similarly, the order-4 labeling of $\mathbb{Z}^2$ is represented in Figure 6.5 when using the following decomposition:

$$\begin{aligned}\mathbb{Z}^2 = 4\mathbb{Z}^2 &+ \left[2\mathbf{R}\mathbb{Z}^2/4\mathbb{Z}^2\right] + \left[2\mathbb{Z}^2/2\mathbf{R}\mathbb{Z}^2\right]\\ &+ \left[\mathbf{R}\mathbb{Z}^2/2\mathbb{Z}^2\right] + \left[\mathbb{Z}^2/\mathbf{R}\mathbb{Z}^2\right]\end{aligned} \qquad [6.19]$$

An N-dimensional complex lattice is a binary lattice if there exists an integer μ such that $\Phi^\mu \mathbb{G}^N \subset \Lambda \subset \mathbb{G}^N$. The lattice's Φ-depth is equal to the lowest μ fulfilling this constraint. Consequently, Λ is included in the partition chain $\mathbb{G}^N/\Lambda/2^\mu\mathbb{G}^N$.

As the complex lattice $2^\mu\mathbb{G}^N$ is equivalent to the real lattice $2^{\mu/2}\mathbb{Z}^{2N}$ if μ is even and to $2^{(\mu-1)/2}\mathbb{Z}^{2N}$ if μ is odd, the complex lattice's Φ-depth is deduced from the binary depth of its equivalent real lattice. In the following, the considered depth is Φ-depth. Moreover, the other characteristics of any complex binary lattice are deduced from that of its isomorphic real binary lattice.

6.1.4. *Lattice performances on the additive white Gaussian noise channel*

We consider a transmitter that sends points from lattice Λ on the additive white Gaussian noise channel. The additive noise has dimension N, where each component is centered and of variance $\sigma^2 = N_0/2$. The encoder is limited to a finite constellation centered around $\mathbf{0}$, denoted as C, which is a subset of Λ. Since the constellation's volume is finite, its energy is finite too. The set of points in C is denoted as $\{\boldsymbol{\lambda}_k\}_{1\leq k\leq |C|}$, where $|C|$ is the cardinal of C. Since the fundamental volume only contains one point of Λ, the constellation has the following number of points:

$$|C| \approx \frac{vol(C)}{vol(\Lambda)} \qquad [6.20]$$

6.1.4.1. *General expression of the error probability*

We assume that point $\boldsymbol{\lambda}_k$ has been transmitted and that the noisy vector $\mathbf{y}$ has been received. The receiver applies the maximum likelihood criterion and thus considers that the most likely transmitted symbol is the closest point to

$\mathbf{y}$. Consequently, an error occurs if $\mathbf{y}$ does not belong to the Voronoï region of $\boldsymbol{\lambda}_k$, denoted as $V(\boldsymbol{\lambda}_k)$. The probability of detecting $\boldsymbol{\lambda}_k$ without any error, knowing that it has been transmitted, is equal to:

$$P_c = \frac{1}{\left(\sigma\sqrt{2\pi}\right)^N} \int_{V(\boldsymbol{\lambda}_k)} e^{-\|\mathbf{y}-\boldsymbol{\lambda}_k\|^2/(2\sigma^2)} d\mathbf{y} \qquad [6.21]$$

If we assume that all constellation points have the same probability of being transmitted, the constellation's error probability is:

$$P_e = 1 - \frac{1}{|C|} \sum_{k=1}^{|C|} \frac{1}{\left(\sigma\sqrt{2\pi}\right)^N} \int_{V(\boldsymbol{\lambda}_k)} e^{-\|\mathbf{y}-\boldsymbol{\lambda}_k\|^2/(2\sigma^2)} d\mathbf{y} \qquad [6.22]$$

Yet, all Voronoï regions are similar and equal to that centered around the origin, $V(\mathbf{0})$. This implies that the error probability is equal to:

$$P_e = 1 - \frac{1}{\left(\sigma\sqrt{2\pi}\right)^N} \int_{V(\mathbf{0})} e^{-\|\mathbf{y}\|^2/(2\sigma^2)} d\mathbf{y} \qquad [6.23]$$

This probability is very hard to compute because it requires an integration on $V(\mathbf{0})$. Nevertheless, at high signal-to-noise ratio, since any point has $K_{min}(\Lambda)$ nearest neighbors, the error probability can be approximated as follows [CON 99]:

$$\begin{aligned} P_e &\approx \frac{K_{min}(\Lambda)}{2} \text{erfc}\left(\frac{\rho}{\sigma\sqrt{2}}\right) \\ &= \frac{K_{min}(\Lambda)}{2} \text{erfc}\left(\frac{d_{\text{min}}}{2\sqrt{N_0}}\right) \end{aligned} \qquad [6.24]$$

6.1.4.2. *Coding and shaping gains*

When all symbols have equal probability, the average energy of constellation C is:

$$E_s = \frac{1}{|C|} \sum_{k=1}^{|C|} \|\boldsymbol{\lambda}_k\|^2 \qquad [6.25]$$

If the number of points of the constellation is high, the average energy is very close to the second moment of the Voronoï region:

$$E_s \approx \mathcal{E}(C) = \frac{1}{vol(C)} \int_C \|\mathbf{x}\|^2 \, d\mathbf{x} \qquad [6.26]$$

The average energy per two dimensions allows us to compare the energy of an N-dimensional lattice with that of two-dimensional constellations, as for instance QAM. It is equal to:

$$E_{s,2D} = \frac{2}{N|C|} \sum_{k=1}^{|C|} \|\boldsymbol{\lambda}_k\|^2 \approx \frac{2}{Nvol(C)} \int_C \|\mathbf{x}\|^2 \, d\mathbf{x} \qquad [6.27]$$

We now aim at expressing the error probability of an N-dimensional lattice depending on the signal-to-noise ratio per two dimensions, so as to compare it with the error probability of QAM modulations. The signal-to-noise ratio per two dimensions is $\frac{E_{s,2D}}{N_0}$, where N_0 is the noise power spectrum density when considering both dimensions. Inserting this expression in equation [6.24], we obtain:

$$P_e \approx \frac{K_{min}(\Lambda)}{2} \text{erfc}\left(\sqrt{\frac{d_{\text{min}}{}^2}{4E_{s,2D}} \frac{E_{s,2D}}{N_0}}\right) \qquad [6.28]$$

The constellation's figure of merit (CFM) is defined as follows:

$$\text{CFM} = \frac{d_{\text{min}}{}^2}{E_{s,2D}} \qquad [6.29]$$

The error probability can be expressed depending on the figure of merit:

$$P_e \approx \frac{K_{min}(\Lambda)}{2} \text{erfc}\left(\sqrt{\frac{\text{CFM}}{4} \frac{E_{s,2D}}{N_0}}\right) \qquad [6.30]$$

For instance, if we consider squared regular QAM modulations, according to equation [3.84], we have $d_{\text{min}}{}^2 = \frac{6E_s}{M-1}$. The figure of merit is then:

$$\text{CFM} = \frac{6}{M-1} \approx \frac{6}{2^n} \qquad [6.31]$$

where the approximation is performed at high M, with n the number of bits per symbol.

This result is valid for QAM and may be generalized to the elementary lattice $\mathbb{Z}^N + \left(\frac{1}{2}, \frac{1}{2}, ... \frac{1}{2}\right)$ bounded by the N-dimensional hypercube centered on $\mathbf{0}$. This constellation is still denoted as C. Its figure of merit is $\text{CFM}_0 = 6/2^\beta$, with $\beta = 2/N \log_2(|C|)$. This figure of merit is chosen as a reference to compute the signal-to-noise ratio gain in the other lattices. According to equation [6.20], we have:

$$2^\beta = |C|^{2/N} \approx \left[\frac{vol(C)}{vol(\Lambda)}\right]^{2/N} \qquad [6.32]$$

Thus, the N-dimensional elementary figure of merit is:

$$\text{CFM}_0 \approx 6\left[\frac{vol(\Lambda)}{vol(C)}\right]^{2/N} \qquad [6.33]$$

The figure of merit of any lattice can be expressed depending on CFM_0:

$$\begin{aligned} \text{CFM} &= \frac{{d_{\text{min}}}^2}{E_{s,2D}} \\ &= 6\left[\frac{vol(\Lambda)}{vol(C)}\right]^{2/N} \times \frac{{d_{\text{min}}}^2}{(vol(\Lambda))^{2/N}} \times \frac{(vol(C))^{2/N}}{6E_{s,2D}} \\ &\approx \text{CFM}_0 \times \gamma_c(\Lambda) \times \gamma_s(C) \end{aligned} \qquad [6.34]$$

Thus, the figure of merit CFM_0 is increased by the fundamental coding gain $\gamma_c(\Lambda)$ [6.8], which only depends on lattice Λ's structure, and the shaping gain, defined by:

$$\gamma_s(C) = \frac{(vol(C))^{2/N}}{6E_{s,2D}} \qquad [6.35]$$

The shaping gain depends on the boundary of the constellation. It is equal to 1 (or equivalently to 0 dB) in the simplest case when this region is a hypercube centered on the origin. The more spherical the boundary and the higher the constellation's dimension gets, the highest the shaping gain gets. Its asymptotic value is $\gamma_{s,\text{max}} = \pi e/6$, equal to 1.53 dB. The shaping gain of N-dimensional spherical lattices is represented in Figure 6.6 which comes from the paper of Forney and Ungerboeck [FOR 98].

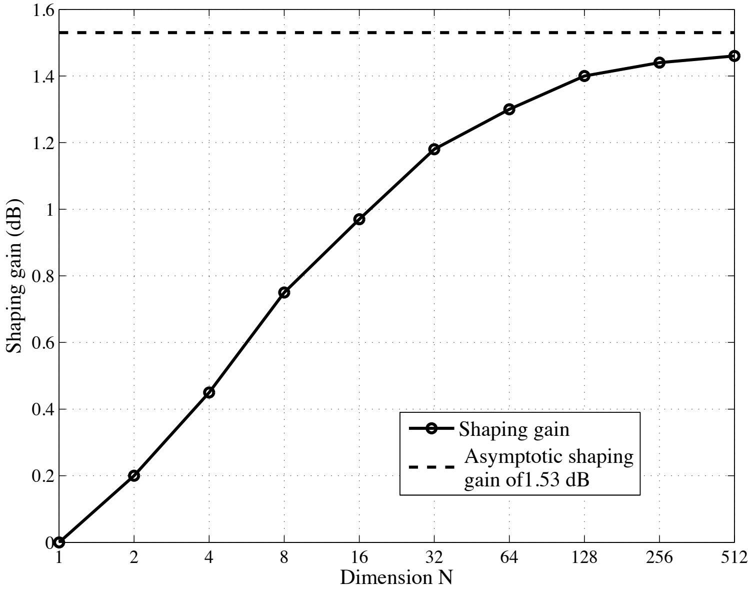

Figure 6.6. *Shaping gain of N-dimensional spherical lattices*

6.2. Block-coded modulations

In the remainder of this chapter, we only consider binary lattices. Forney [FOR 84] has proven that finite-dimensional binary lattices can be generated by the following three steps:

– binary data are encoded by a conventional linear block code;

– the obtained coded bits are used to select a sublattice of the original lattice;

– additional uncoded bits are used to select one point in the chosen sublattice.

Block-coded modulations are thus binary lattices. In this section, we first detail the main constructions that generate these lattices. Then, some implementation examples with 64-QAM modulations are detailed.

6.2.1. *Main algebraic constructions of lattices*

6.2.1.1. *Construction A*

An N-dimensional real lattice Λ is a mod-2 lattice obtained with construction A if and only if it is congruent modulo 2 to the codeword $\mathbf{c}$ of a binary linear block code $\mathcal{C}(N, K)$ with rate K/N. The lattice is then defined as follows:

$$\Lambda = 2\mathbb{Z}^N + \mathcal{C} \qquad [6.36]$$

Λ is the union of 2^K cosets of $2\mathbb{Z}^N$, where each coset leader is a codeword of $\mathcal{C}$.

The lattice's redundancy is then equal to $r(\Lambda) = N - K$. Its minimum squared distance is the minimum between 4 and the minimum Hamming distance of code $\mathcal{C}$:

$$d_{\text{min}}{}^2 = \min\{4, d_{\text{min}}(\mathcal{C})\} \qquad [6.37]$$

where the 4 value is due to the fact that points $(2, 0, ...0)$ in $2\mathbb{Z}^N$ all belong to the lattice.

Λ's generator matrix is deduced from that of $\mathcal{C}$, denoted as $\mathbf{G}_{\mathcal{C}}$. If the latter is systematic:

$$\mathbf{G}_{\mathcal{C}} = \begin{pmatrix} \mathbf{I}\ \mathbf{P} \end{pmatrix} \qquad [6.38]$$

then the generator matrix of Λ is:

$$\mathbf{G}_{\Lambda} = \begin{pmatrix} 2\mathbf{I} & \mathbf{0} \\ \mathbf{P} & \mathbf{I} \end{pmatrix} \qquad [6.39]$$

Some examples of lattices obtained with construction A are checkerboard lattices D_N, which are built from parity-check codes $(N, N-1, 2)$.

Schäfli's lattice $D_4 = 2\mathbb{Z}^4 + (4, 3, 2)$ is an interesting example of a checkerboard lattice. It corresponds to the set of vectors of $\mathbb{Z}^4$ that have an even number of odd components. $\mathbb{Z}^4$ may be partitioned into two sublattices:

D_4 and $D_4 + (1,0,0,0)$. Consequently, according to [6.12], $vol(D_4) = 2vol(\mathbb{Z}^4) = 2$. Moreover, from equation [6.37], the D_4 lattice has a minimum squared distance equal to 2. We can deduce from [6.8] that the lattice's coding gain is equal to $\gamma_c(D_4) = \sqrt{2}$.

As the minimum squared distance ${d_{\text{min}}}^2$ cannot exceed 4, construction A should not be used with binary block codes whose minimum Hamming distance is higher than 4. The Gosset code E_8 reaches this minimum distance because it is constructed from the extended Reed–Muller code $(8,4,4)$. It is written as follows:

$$E_8 = 2\mathbb{Z}^8 + (8,4,4) \qquad [6.40]$$

Its coding gain is equal to $\gamma_c(E_8) = 2$.

Construction A may also be applied on complex lattices. An N-dimensional complex lattice Λ is a mod-2 lattice if and only if it is equal to the set of N-dimensional Gaussian integers that are congruent modulo Φ^2 to codewords $\Phi\mathbf{c}_1 + \mathbf{c}_0$, where $\mathbf{c}_1$ belongs to a binary linear block code $\mathcal{C}_1(N,K)$, and $\mathbf{c}_0$ belongs to a subcode $\mathcal{C}_0(N, J-K)$ of $\mathcal{C}_1$.

The lattice's redundancy is $r(\Lambda) = 2N - J$. Its minimum squared distance is equal to:

$${d_{\text{min}}}^2 = \min\{4, 2d_{\text{min}}(\mathcal{C}_1), d_{\text{min}}(\mathcal{C}_0)\} \qquad [6.41]$$

6.2.1.2. *Construction B for decomposable lattices*

In order to increase the minimum squared distance of lattices, two block codes must be used. Construction B allows us to generate mod-4 lattices.

We consider two linear block codes $\mathcal{C}_0$ and $\mathcal{C}_1$ such that $\mathcal{C}_0$ is a subcode of $\mathcal{C}_1$. Thus, the following lattice:

$$\Lambda = 4\mathbb{Z}^N + 2\mathcal{C}_1 + \mathcal{C}_0 \qquad [6.42]$$

is a real decomposable lattice mod-4 obtained with construction B.

For instance, such a lattice is generated if all codewords of $\mathcal{C}_0(N,K,d)$ have their Hamming weight congruent to 4, and $\mathcal{C}_1$ is the parity-check code $(N, N-1, 2)$.

The minimum squared distance of Λ is equal to:

$$d_{\min}{}^{2} = \min\left\{16, 4d_{\min}(\mathcal{C}_1), d_{\min}(\mathcal{C}_0)\right\} \qquad [6.43]$$

Thus, the minimum squared distance of code RE_8 is 8. It is defined as follows:

$$RE_8 = 4\mathbb{Z}^8 + 2(8,7,2) + (8,1,8) \qquad [6.44]$$

The half-Leech lattice H_{24} of dimension 24 has the same minimum squared distance. It is generated by:

$$H_{24} = 4\mathbb{Z}^{24} + 2(24,23,2) + (24,12,8) \qquad [6.45]$$

where $(24,12,8)$ is the extended Golay code.

6.2.1.3. *Summary of the main lattices*

In this book, we have only presented a few examples of lattice constructions by restricting ourselves to the most simple cases.

Tables 6.1 and 6.2 come from Forney's paper [FOR 88a]. They summarize the most common binary lattices and how they are constructed, whether they are seen as real or complex lattices. The lattice's real dimension is indicated in its name. The depth value corresponds to the Φ-depth.

In this table, beyond repetition and parity-check codes, the following block codes can be identified:

– $(12,7,4)$ and $(16,11,4)$ are the extended Hamming codes;

– $(16,5,8)$ is an extended Bose Ray-Chaudhuri codes (BCH) code;

– $(24,12,8)$ is the extended Golay code;

– $(8,4,4)$, $(16,5,8)$, $(32,26,4)$ and $(32,16,8)$ are the Reed–Muller codes.

For more information on these codes, the readers may refer to the first volume of this book [LER 15].

Lattice	Depth	Real construction
$\mathbb{Z}^2$	0	$\mathbb{Z}^2$
$\mathbb{Z}^4$	0	$\mathbb{Z}^4$
D_4	1	$2\mathbb{Z}^4 + (4, 3, 2)$
$\mathbb{Z}^8$	0	$\mathbb{Z}^8$
D_8	1	$2\mathbb{Z}^8 + (8, 7, 2)$
E_8	2	$2\mathbb{Z}^8 + (8, 4, 4)$
$\mathbb{Z}^{16}$	0	$\mathbb{Z}^{16}$
D_{16}	1	$2\mathbb{Z}^{16} + (16, 15, 2)$
H_{16}	2	$2\mathbb{Z}^{16} + (16, 11, 4)$
Λ_{16}	3	$4\mathbb{Z}^{16} + 2(16, 15, 2) + (16, 5, 8)$
$\mathbb{Z}^{24}$	0	$\mathbb{Z}^{24}$
H_{24}	3	$4\mathbb{Z}^{24} + 2(24, 23, 2) + (24, 12, 8)$
Λ_{24}	4	$4\mathbb{Z}^{24} + 2(24, 18, 4) + (24, 6, 16)'$
$\mathbb{Z}^{32}$	0	$\mathbb{Z}^{32}$
D_{32}	1	$2\mathbb{Z}^{32} + (32, 31, 2)$
X_{32}	2	$2\mathbb{Z}^{32} + (32, 26, 4)$
H_{32}	3	$4\mathbb{Z}^{32} + 2(32, 31, 2) + (32, 16, 8)$

Table 6.1. *Most common lattices, real construction*

6.2.2. *Construction of block-coded modulations*

We now detail how block-coded modulations can be constructed from a 64-QAM modulation. The complex construction defined in Table 6.2 is used.

A 64-QAM modulation provides a complex symbol of $\mathbb{G}$. Thus, to represent a complex N-dimensional lattice, N successive QAM symbols are required. The sequence of QAM symbols must verify specific constraints in order to belong to the lattice. Indeed, the lattice does not contain all points in the set of 2^{6N} points that N successive 64-QAM may represent, if the symbols were allowed to take any possible value.

As already mentioned in the introduction to section 6.2, the following method is used:

– 64-QAM symbols are labeled according to Ungerboeck's rule, using a partition whose order is equal to the lattice's Φ-depth;

– the sublattice is selected due to bits coded by the block code (if the lattice was generated with construction A) or block codes (if the lattice was generated with construction B);

– then, a point in the sublattice is selected due to the remaining uncoded bits.

Lattice	Depth	Complex construction
$\mathbb{Z}^2$	0	$\mathbb{G}$
$\mathbb{Z}^4$	0	$\mathbb{G}^2$
D_4	1	$\Phi\mathbb{G}^2 + (2,1,2)$
$\mathbb{Z}^8$	0	$\mathbb{G}^4$
D_8	1	$\Phi\mathbb{G}^4 + (4,3,2)$
E_8	2	$\Phi^2\mathbb{G}^4 + \Phi(4,3,2) + (4,1,4)$
$\mathbb{Z}^{16}$	0	$\mathbb{G}^8$
D_{16}	1	$\Phi\mathbb{G}^8 + (8,7,2)$
H_{16}	2	$\Phi^2\mathbb{G}^8 + \Phi(8,7,2) + (8,4,4)$
Λ_{16}	3	$\Phi^3\mathbb{G}^8 + \Phi^2(8,7,2) + \Phi(8,4,4) + (8,1,8)$
$\mathbb{Z}^{24}$	0	$\mathbb{G}^{12}$
H_{24}	3	$\Phi^3\mathbb{G}^{12} + \Phi^2(12,11,2) + \Phi(12,7,4) + (12,5,8)'$
Λ_{24}	4	$\Phi^4\mathbb{G}^{12} + \Phi^3(12,11,2) + \Phi^2(12,7,4) + \Phi(12,5,8)' + (12,1,16)'$
$\mathbb{Z}^{32}$	0	$\mathbb{G}^{16}$
D_{32}	1	$\Phi\mathbb{G}^{16} + (16,15,2)$
X_{32}	2	$\Phi^2\mathbb{G}^{16} + \Phi(16,15,2) + (16,11,4)$
H_{32}	3	$\Phi^3\mathbb{G}^{16} + \Phi^2(16,15,2) + \Phi(16,11,4) + (16,1,16)$

Table 6.2. *Most common lattices, complex construction*

For example, for lattice $D_4 = \Phi\mathbb{G}^2 + (2,1,2)$, two successive QAM symbols are used to obtain point in D_4. The partition is of order $K = 1$ (see Figure 6.7): thus, QAM symbols are separated into two sublattices, labeled A and B, respectively. Referring to Ungerboeck's labeling, A and B are simply given by the value of a_0 in equation [6.18]. One bit is used to select the sublattice, with repetition code$(2,1,2)$. Due to the partition of the constellation, there are 32 different possible values among the A and B labeled values, respectively. These values are chosen for both bits generated by the repetition code, from $10 = 2 \times \log_2(32)$ uncoded bits. The sequence of two 64-QAM symbols thus allows us to build a point in lattice D_4.

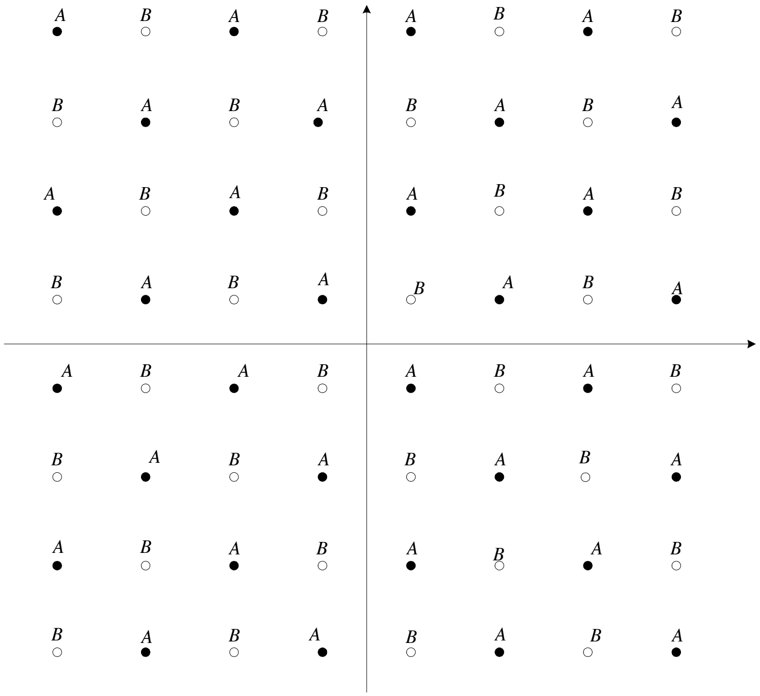

Figure 6.7. *Order-*1 *partition of* $\mathbb{Z}^2$

The number of points in this lattice is 2^{11}, whereas two independent QAM symbols would have allowed us to reach $2^{2\times6} = 2^{12}$ points. In the considered case, a factor 2 has been lost because of the partition into two sublattices. The D_4 code has a spectral efficiency of 5.5 bits/symbol since 11 bits are transmitted in 2 QAM symbols.

D_4 lattice's encoding is represented in Figure 6.8.

Similarly, lattice $E_8 = \Phi^2\mathbb{G}^4 + \Phi(4,3,2) + (4,1,4)$ is obtained in three steps, using 4 successive 64-QAM:

– 64-QAM symbols are labeled according to an order-2 Ungerboeck's labeling (see Figure 6.9);

– one bit selects the first sublattice, between A and B. It is kept in the four successive QAM. This bit is thus copied four times, with repetition code $(4, 1, 4)$;

– the parity-check code $(4, 3, 2)$ is used to select the sublattice corresponding to the second level of partition, between A_0 and A_1 if A was previously selected, and between B_0 and B_1 if B was previously selected. 4 bits are used, one for each QAM symbol. They are interdependent because they must fulfill the parity-check constraint;

– finally, $16 = 4 \times \log_2(6)$ uncoded bits are used to select, for each of the four successive QAM, the QAM symbol among the 16 locations repeating A_i and B_i in QAM constellation.

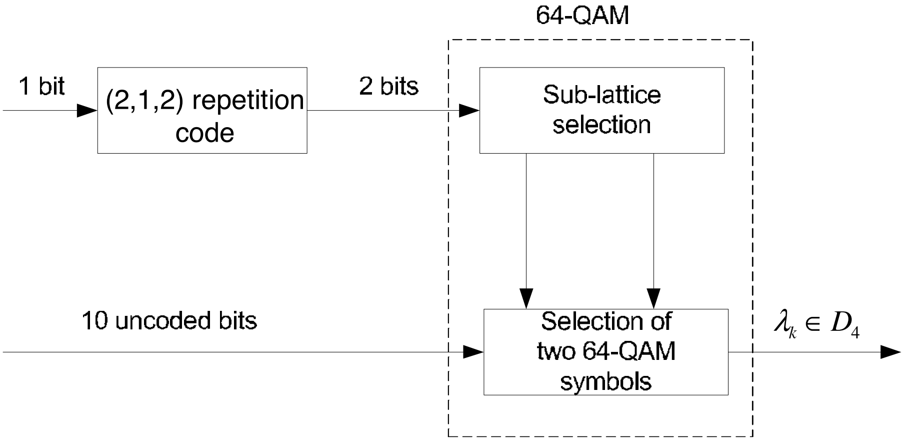

Figure 6.8. *Block diagram of D_4 lattice's encoding*

The block diagram of E_8 lattice's encoding is shown in Figure 6.10. Its spectral efficiency is equal to 5 bits per symbol, since 4 QAM symbols are necessary to transmit 20 information bits.

We now detail an example of encoding with E_8 lattice. The 20 information bits are: 01100100110010110010. The first bit, 0, selects partition A. It is repeated four times.

The next three bits, 110, are coded with the parity-check code. The associated codeword is 1100. They provide the following sequence to be transmitted on the 4 symbol periods: A_1, A_1, A_0, A_0.

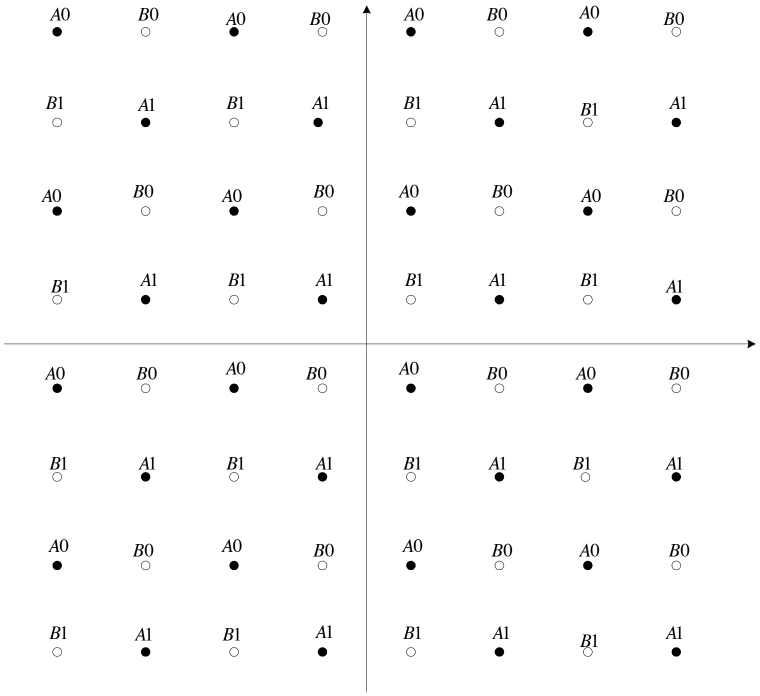

Figure 6.9. *Order-2 partition of $\mathbb{Z}^2$*

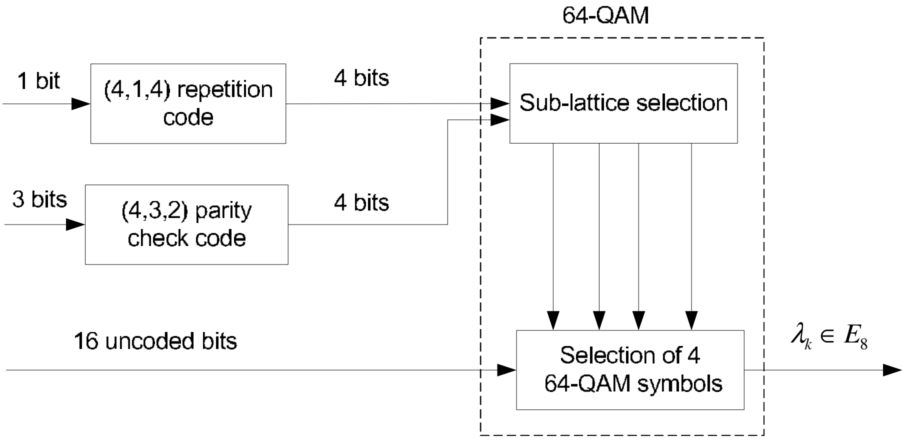

Figure 6.10. *Block diagram of E_8 lattice's encoding*

We finally need to determine, among the 16 locations repeating A_i and B_i in 64-QAM constellation, which one is selected in each symbol period. Figure 6.11 shows how the QAM symbols are labeled, from the left to the right and from the top to the bottom. The 16 uncoded bits allow us to select the 64-QAM symbols to transmit:

– the first four uncoded bits are 0100, which is equal to 4 in decimal notations. In the first symbol period, the chosen A_1 symbol is located at $(-5a, a)$, assuming that the minimum distance of 64-QAM constellation is $2a$;

– the next four uncoded bits are 1100, which is equal to 12 in decimal. In the second symbol period, the chosen A_1 symbol is located at $(-5a, -7a)$;

– the next four uncoded bits are 1011, which is equal to 11 in decimal. In the third symbol period, the chosen A_0 symbol is located at $(5a, -a)$;

– the last four uncoded bits are 0010, which is equal to 2 in decimal. In the fourth symbol period, the chosen A_0 symbol is located at $(5a, 3a)$.

Finally, Leech lattice Λ_{24}, represented in Figure 6.13, is obtained from 12 successive symbols of a 64-QAM modulation. The code's spectral efficiency is 4 bits/symbol: 12 QAM symbols carry 48 information bits. The partition, represented in Figure 6.12, is of order 4. It is chosen differently from Ungerboeck's labeling in Figure 6.5, but this does not have any influence on the minimum distance. The Leech lattice is obtained as follows:

– 64-QAM symbols are labeled according to 4-order labeling;

– one bit selects the first sublattice, between A and B. The repetition code $(12, 1, 12)$ is used to keep this bit during the 12 successive QAMs;

– 12 X_{ijk} symbols, with $X = A$ or B are selected as follows: 12 couples (i, j) are obtained by grouping two consecutive bits of the codeword generated by Golay code $(24, 12, 8)$. Then, 12 bits corresponding to index k are obtained with the parity-check code $(12, 11, 2)$, using an even parity rule if $X = A$ and an odd parity rule if $X = B$;

– finally, $24 = 12 \times \log_2(4)$ uncoded bits are used to select, for each of the 12 successive QAMs, the 64-QAM symbol among the 4 locations that repeat X_{ijk}.

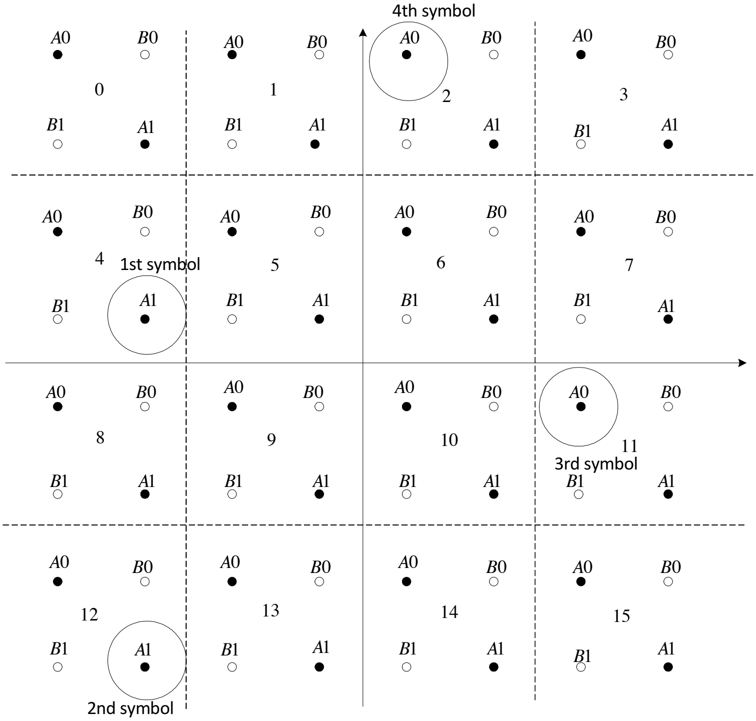

Figure 6.11. *Example of E_8 encoding*

6.2.2.1. *Block-coded modulations' decoding*

In order to achieve good performances in terms of bit error rate, the block-coded modulations implemented with QAM modulations are decoded with algorithms that apply the maximum-likelihood criterion with soft decisions. In this section, we describe the decoding algorithm of E_8, and then the algorithm implemented by Be'ery *et al.* [BE' 89, AMR 94] for the Leech lattice Λ_{24}. Decoding of other block-coded modulations can be extrapolated from these two examples.

Let us consider a sequence of four 64-QAM symbols that generate a point in E_8 lattice. Decoding is performed in three steps. First, the receiver identifies, for each QAM symbol, the point in subsets A_0, A_1, B_0 and B_1, respectively, which has the lowest distance to the received noisy signal, denoted as r_n, with $n \in \{1, .., 4\}$. There are 16 likely points in each subset. The squared minimum

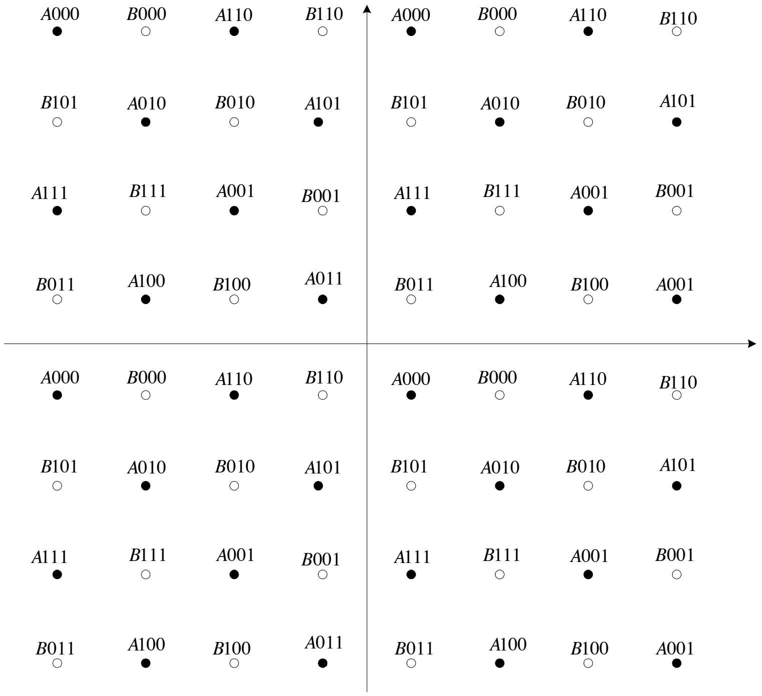
A000 B000 A110 B110 A000 B000 A110 B110
B101 A010 B010 A101 B101 A010 B010 A101
A111 B111 A001 B001 A111 B111 A001 B001
B011 A100 B100 A011 B011 A100 B100 A001
A000 B000 A110 B110 A000 B000 A110 B110
B101 A010 B010 A101 B101 A010 B010 A101
A111 B111 A001 B001 A111 B111 A001 B001
B011 A100 B100 A011 B011 A100 B100 A001

Figure 6.12. *Order-4 partition of $\mathbb{Z}^2$*

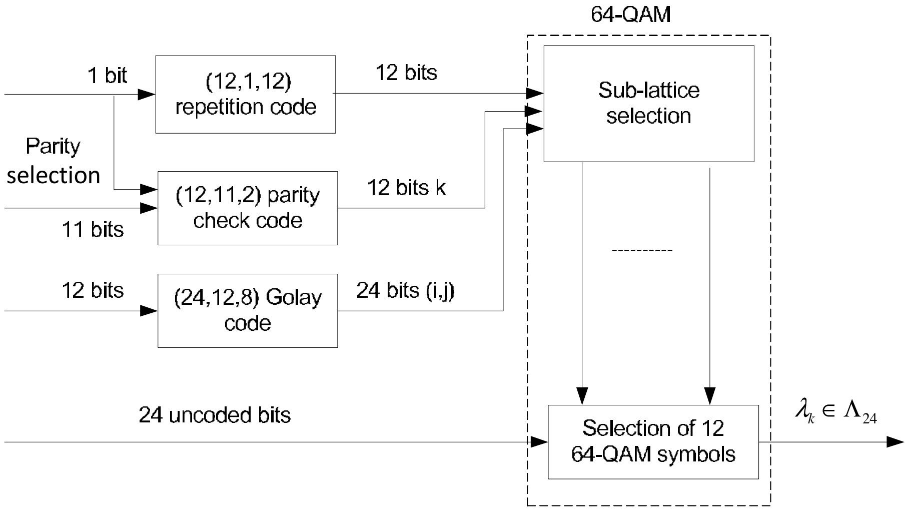

Figure 6.13. *Block diagram of Λ_{24} lattice's encoding*

Euclidean distance to each subset is stored:

$$d_{X_i}(n) = \min |r_n - X_i|^2 \ , X \in \{A, B\} \, , i \in \{0, 1\} \qquad [6.46]$$

The eight codewords generated by the parity-check code are denoted as $\mathbf{c}^m = [c_0^m, ..., c_3^m]$, with $m = \{1, .., 8\}$. For each sublattice A and B, the distance between the received symbol and each codeword is computed as:

$$M_X(\mathbf{c}^m) = \sum_{n=0}^{3} d_{X_{c_n^m}}(n) \ , X \in \{A, B\} \qquad [6.47]$$

The most likely codeword is the one that minimizes this distance:

$$m_X{}^* = \text{arg min}_{\ m} \ M_X(\mathbf{c}^m) \qquad [6.48]$$

Finally, bit X is identified by choosing the codeword that minimizes distances, taking into account the distance to sublattice A and the distance to sublattice B:

$$X^* = \text{arg min}_{\ X} \ M_X(\mathbf{c}^{m_X{}^*}) = \text{arg min}_{\ X} \ \left\{ M_A(\mathbf{c}^{m_A{}^*}), M_B(\mathbf{c}^{m_B{}^*}) \right\} \qquad [6.49]$$

When all $X_i = A_0, A_1, B_0$ or B_1 have been identified for the four symbols, the uncoded bits are recovered from the location of the closest points in each subset, which was obtained in the first step.

The performances of E_8 block-coded modulation are now estimated. Since 4 64-QAM symbols are necessary to send 20 bits, the spectral efficiency is $20/4 = 5$ bits per symbol. Thus, the performances of E_8 should be compared to those of a QAM modulation with the same number of bits per symbol, in the considered case a 32-QAM.

As previously seen, E_8 lattice's coding gain is $\gamma_c(E_8) = 2$, or equivalently 3 dB. Nevertheless, this lattice has a high kissing number: $K_{\text{min}} = 240$. Thus, the performance gain cannot reach 3 dB, since the kissing number must be taken into account in the error probability equation [6.30]. 32-QAM performances are approximately obtained with formula [3.91]. Figure 6.14 shows that the performance gain between E_8 and 32-QAM is equal to 2 dB when the bit error rate is equal to 10^{-8}. At low E_b/N_0, block-coded modulations are less efficient than uncoded modulations, because the kissing number has a strong influence on the bit error rate. On the contrary, at high E_b/N_0, the coding gain becomes larger than the kissing number, and the asymptotic gain can be approached.

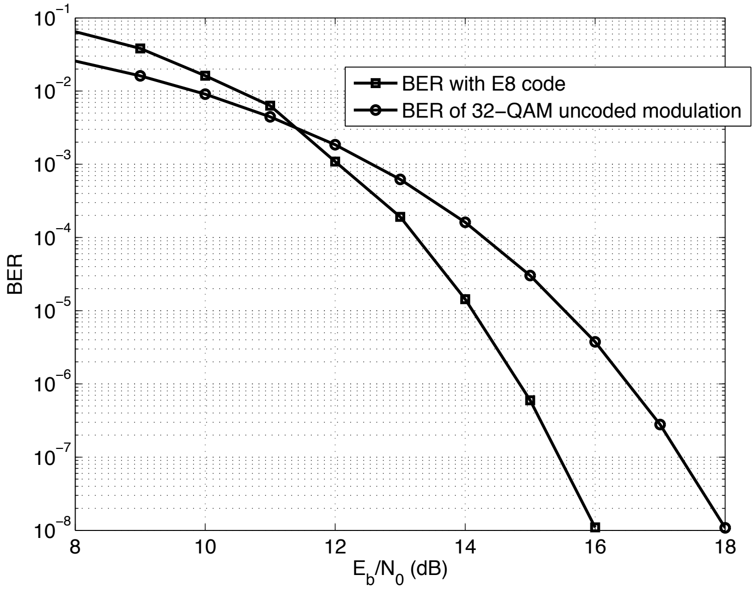

Figure 6.14. *Performances of E_8 block-coded modulation compared to uncoded 32-QAM*

The Leech lattice's decoding algorithm is of course more complex than that of E_8, since Leech lattices use both a parity-check code and the Golay code. The following method is used: the receiver first identifies, for each QAM symbol, the point in subset X_{ijk} which is the closest to the noisy received signal r_n, where $n \in \{1,..,12\}$. There are four likely points in each subset. The minimum squared Euclidean distance to each subset is denoted as follows:

$$d_{X_{ijk}}(n) = \min |r_n - X_{ijk}|^2 \ , X \in \{A, B\}, (i,j,k) \in \{0,1\}^3 \quad [6.50]$$

Then, the minimum distance between both parities k is selected, and the associated parity value $p_{X_{ij}}(n)$ is stored:

$$d_{X_{ij}}(n) = \min\left\{d_{X_{ij0}}(n), d_{X_{ij1}}(n)\right\} \qquad [6.51]$$

and

$$p_{X_{ij}}(n) = \begin{cases} 0 \text{ if } d_{X_{ij}}(n) = d_{X_{ij0}}(n) \\ 1 \text{ if } d_{X_{ij}}(n) = d_{X_{ij1}}(n) \end{cases} \qquad [6.52]$$

The difference between both minimum distances is kept in order to eventually correct errors at the end of the process. It is denoted as:

$$\delta_{X_{ij}}(n) = \left|d_{X_{ij0}}(n) - d_{X_{ij1}}(n)\right| \qquad [6.53]$$

Then, the distance between each Golay codeword and the received symbols is computed. Let us note $\mathbf{c}^m = [c_0^m, ..., c_{23}^m]$, with $m = \left\{1, .., 2^{12}\right\}$ the 2^{12} Golay codewords. The distance is equal to:

$$M_X(\mathbf{c}^m) = \sum_{n=0}^{11} d_{X_{c_{2n}^m c_{2n+1}^m}}(n) \qquad [6.54]$$

Parity must be fulfilled by the sequence of indices k in the 12 symbols corresponding to Golay codewords. It simply is:

$$P_X(\mathbf{c}^m) = \sum_{n=0}^{11} p_{X_{c_{2n}^m c_{2n+1}^m}}(n) \mod 2 \qquad [6.55]$$

If $X = A$, then parity must be even, which corresponds to $P_A(\mathbf{c}^m) = 0$. If on the contrary $X = B$, by construction, parity must be odd, thus $P_B(\mathbf{c}^m) = 1$.

If parity is not fulfilled (which means that either $P_A(\mathbf{c}^m) = 1$ or $P_B(\mathbf{c}^m) = 0$), then one of the couples of bits ij must be changed. This implies that the minimum distance is no longer taken in equation [6.51]. In order to minimize the induced total distance increase, the chosen couple of bits is the one that minimizes the difference between both minimum distances $\delta_{X_{c_{2n}^m c_{2n+1}^m}}(n)$. It is defined by:

$$\delta^{\min}(\mathbf{c}^m) = \min\left\{\delta_{X_{c_{2n}^m c_{2n+1}^m}}(n), n \in \{1, .., 12\}\right\} \qquad [6.56]$$

The new distance between the Golay codeword and the received symbols is then:

$$M_X(\mathbf{c}^m) = \sum_{n=0}^{11} d_{X_{c_{2n}^m c_{2n+1}^m}}(n) + \delta^{\min}(\mathbf{c}^m) \qquad [6.57]$$

The most likely codeword is the one that minimizes $M_X(\mathbf{c}^m)$.

$$m_X{}^* = \text{arg min}_{\ m}\ M_X(\mathbf{c}^m) \qquad [6.58]$$

Then, the value of X is identified by choosing the codeword that minimizes distances, when only the distance to sublattice A or sublattice B is taken into account:

$$X^* = \text{arg min}_{\ X}\ M_X(\mathbf{c}^{m_X{}^*}) = \text{arg min}_{\ X}\ \left\{M_A(\mathbf{c}^{m_A{}^*}), M_B(\mathbf{c}^{m_B{}^*})\right\} \qquad [6.59]$$

The most likely sequence of input bits is then deduced in a similar way as what was detailed for E_8 block-coded modulation.

6.3. Trellis-coded modulations

Trellis-coded modulations are based on the same principle as block-coded modulations, but they have an infinite length. The transmitted data are thus semi-infinite or infinite sequences. Forney's construction principle applies to trellis-coded modulations as well as it does for block-coded modulations. Nevertheless, trellis-coded modulations are not equivalent to lattices because of their infinite dimension. They are based on the partition of usual constellations (phase shift keying (PSK) or QAM) into subsets with the highest minimum distance. Transitions from one subset to the other are performed according to a convolutional code. The minimum distance gain must exceed the loss in terms of number of bits per symbol for the trellis-coded modulation to be more efficient than the standard uncoded modulations.

6.3.1. *Construction of trellis-coded modulations*

Regular QAM or PSK constellations are partitioned into subsets, so that the minimum distance between two subset symbols is maximized. Figure 6.15 shows an example of partition of an 8-PSK modulation at order 2. Four subsets are finally obtained. The minimum distance in each subset is $2\sqrt{E_s}$, whereas the minimum distance of the original constellation is equal to $2\sqrt{E_s}\sin\left(\frac{\pi}{8}\right) \approx 0.765E_s$.

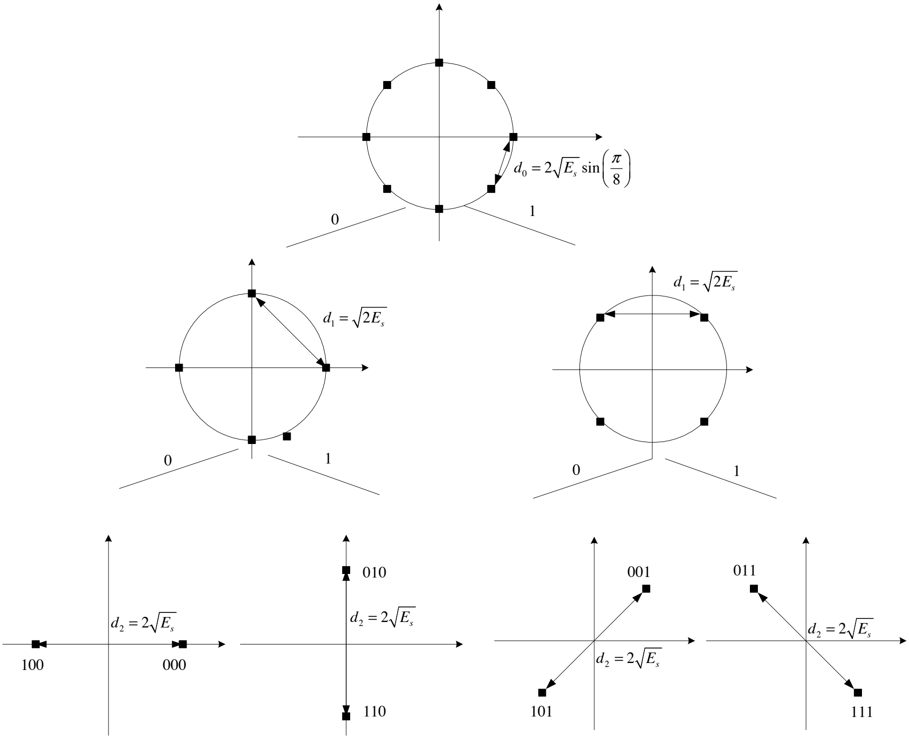

Figure 6.15. *Partition in four subset of an* 8*-PSK modulation*

A constellation's symbol is chosen for each block of m information bits. This block is separated into two parts of sizes k_1 and k_2. Thus:

– a convolutional code encodes the first k_1 bits, thus generating n coded bits;

– the obtained n coded bits are used to select a subset of the constellation;

– the k_2 uncoded bits are then used to select a constellation's point among the points of the chosen subset.

The convolutional code and the associated labels must follow some rules in order to maximize the minimum distance. These rules, as stated by Ungerboeck [UNG 82], are the following:

– all subsets must be fairly used;

– in the trellis, transitions starting from the same state or ending at the same state correspond to symbols that belong to subsets separated by the largest possible Euclidean distance;

– in the trellis, if parallel state transitions (starting from and ending to the same states) occur, then the corresponding symbols must be separated by the largest possible Euclidean distance. Parallel transitions happen when the number of uncoded bits k_2 is higher than zero.

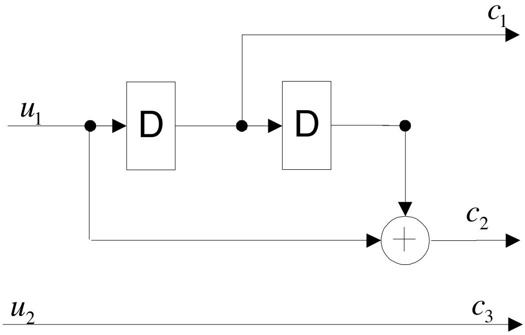

Figure 6.16. *Encoder for 8-PSK constellation*

For example, the 8-PSK partition previously described can be used according to Ungerboeck's rules with the encoder represented in Figure 6.16. The rate $1/2$ convolutional code with generator matrix $\mathbf{G}(D) = \left(D, 1 + D^2\right)$ determines transitions from one state to the other. There are four states, and each state corresponds to the values contained in the two shift registers. The coded bits are denoted as c_1 and c_2. They select a subset of the 8-PSK in its order-2 partition. A third uncoded bit, c_3, allows us to determine which symbol is chosen among the two PSK symbols in each subset. The constellation with labels associated with bits (c_3, c_2, c_1) is represented in Figure 6.17.

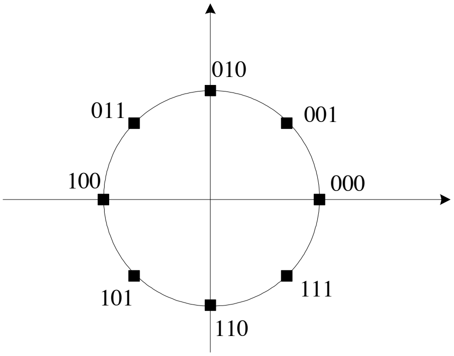

Figure 6.17. *8-PSK constellation symbols' labels depending on bits (c_3, c_2, c_1)*

Trellis 6.18 summarizes all state transitions and the values of the output bits. The branches labels are equal to bits (c_3, c_2, c_1). Since there is an uncoded bit c_3, two parallel transitions take place between any two states. Both symbols corresponding to parallel transitions belong to the same subset among the four final subset, and are consequently separated from the largest distance, $d_2 = 2\sqrt{E_s}$. Similarly, the trellis indicates that Ungerboeck's second rule is fulfilled: the distance between symbols starting from the same state or ending at the same state (aside from parallel transitions) is $d_1 = \sqrt{2E_s}$.

Trellis-coded modulation performances depend on its minimum Euclidean distance. The minimum Euclidean distance is equal to the minimum distance required to return to one symbol of the constellation, after a transition has led to a change in this symbol. For the 8-PSK modulation, two paths must be distinguished to evaluate it. First, parallel transitions allow us to stay in the same state, while changing that state's symbol. The minimum Euclidean distance between two points of the same subset is $d_2 = 2\sqrt{E_s}$. However, the minimum distance can also be evaluated when a change of state occurs. Let us assume that the initial state is 00. To return to that state, the following transitions must take place: $00 \rightarrow 10 \rightarrow 01 \rightarrow 00$. The output bits are then: $010, 001, 010$. The minimum Euclidean distance is then computed with respect to the reference output bits all equal to 000:

$$\begin{aligned}{d_{\text{min, trans}}}^2 &= d\left(000, 010\right)^2 + d\left(000, 001\right)^2 + d\left(000, 010\right)^2\\ &= {d_1}^2 + {d_0}^2 + {d_1}^2\\ &= 4.585E_s \end{aligned} \qquad [6.60]$$

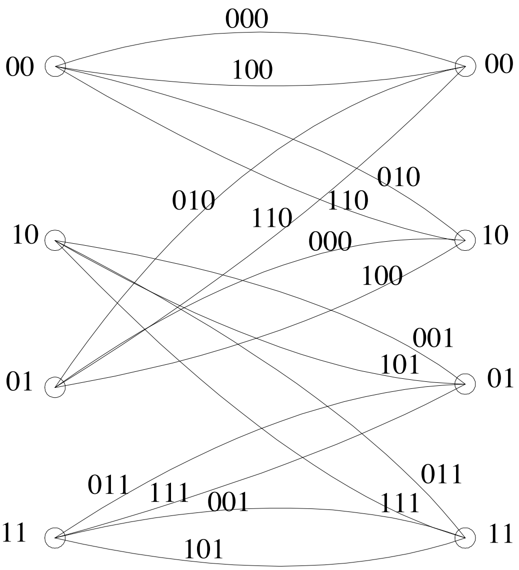

Figure 6.18. *Trellis associates with encoder $\mathbf{G}(D) = \left(D, 1 + D^2\right)$ and with 8-PSK*

We can check among the other paths that $d_{\text{min, trans}}$ is the minimum distance when a change of states occurs. Finally, the squared minimum distance is equal to:

$$ {d_{\text{min, free}}}^2 = \min\ \left\{d_2^2, {d_{\text{min, trans}}}^2\right\} = 4E_s \qquad [6.61]$$

This is called the trellis-coded modulations' free Euclidean distance.

It allows us to compare the studied trellis-coded modulation with an uncoded 4-PSK that has the same number of useful bits per symbol. In 4-PSK modulation, the minimum distance is $d_{\text{min,4-PSK}} = \sqrt{2E_s}$. Consequently, the coding gain is:

$$\gamma_c = \left(\frac{d_{\text{min, free}}}{d_{\text{min,4-PSK}}}\right)^2 = 2 \qquad [6.62]$$

Thus, the trellis-coded modulation provides a coding gain of 3 dB on performances. We can nevertheless notice that this coding gain does not take into account the number of sequences located at the minimum distance from a specific sequence. Similarly to block-coded modulations, the coding gain must be weighted by the number of nearest neighbors, denoted as $K_{min,free}$, when evaluating the bit error rate. At high signal-to-noise ratio, the bit error rate expression can consequently be approximated as follows:

$$P_e \approx \frac{K_{min,free}}{2} \text{erfc}\left(\frac{d_{\text{min, free}}}{2\sqrt{N_0}}\right) \qquad [6.63]$$

Similarly to block-coded modulations, the coding gain becomes dominant at high E_b/N_0. Trellis-coded modulations' asymptotic performances are thus far better than that of uncoded modulations. Yet, at low E_b/N_0, the effect of $K_{min,free}$ prevails, and it is then preferable not to use any encoding.

6.3.2. *Decoding of trellis-coded modulations*

Decoding is performed with Viterbi's algorithm, using as branch metric the Euclidean squared distance. Two successive steps are used. Let r_n be the n^{th} noisy received symbol.

The first step is called subset decoding. It consists of determining, among all the symbols of each subset, which symbol minimizes the Euclidean distance to r_n. This symbol is then the most likely symbol per subset.

Once this most likely symbol has been found, there are no more parallel transitions between states in the trellis. In a second step, Viterbi's algorithm (detailed in the first volume of this book) can be used on the resulting trellis. The metric is then the squared Euclidean distance between the received noisy symbol and the trellis' branches label.

6.4. Conclusion

Coded modulations allow us to increase the performances in high signal-to-noise ratio regime, due to an increase in the constellation's minimum distance. This chapter presented the main principles of these modulations. First, the underlying mathematical foundations of block-coded modulations were introduced, by the study of lattices. Then, we detailed the main constructions of block-coded modulations, and their corresponding decoding algorithms. Lattices have been studied in recent years in the field of network coding, which aims at increasing data rates on relaying networks, where the lattices' group structure is of particular interest. Among the various papers dealing with this topic, we may cite [NAZ 11] and [CHE 13].

In the second part of the chapter, we were interested in trellis-coded modulations. They provide as good performances as block-coded modulations, as convolutional codes are as efficient as block-codes. Trellis-coded modulations were implemented with an 8 state trellis in V.32 modem, and with 16 or 64 state trellis in V.34 modem [FOR 96].

Other more advanced techniques exist, which allow us to improve performances: for instance, turbo-coded modulations combine iterative coding techniques with modulations. To study this topic further, the readers may refer to [HEE 99, JOH 99].

Appendices

Appendix A

Evaluation of the Power Spectrum Density of a Baseband Signal

We consider a baseband signal which can be written as follows:

$$x(t) = \sum_{k=-\infty}^{\infty} a_k g(t - kT) \qquad [A.1]$$

where $[a_k]_{k=-\infty}^{\infty}$ is the discrete sequence of symbols to be transmitted, with mean $m_a = E[a_k]$ and autocorrelation function $R_{aa}(i)$, equal by definition to $E[a_{k+i}a_k^*]$.

The autocorrelation of $x(t)$ is given by:

$$R_{xx}(t + \tau, t) = E\left[x(t + \tau)x^*(t)\right] \qquad [A.2]$$

Notation * means complex conjugate (therefore, allowing $x(t)$ to be a complex signal).

The considered signal $x(t)$ is random but not stationary. It is second-order cyclostationary because its autocorrelation function is invariant to any translation from the time origin proportional to T.

Digital Communications 2: Digital Modulations, First Edition. Mylène Pischella and Didier Le Ruyet.

In order to compute the power spectrum density, we assume that the time origin is a random variable that follows a uniform law between 0 and T. Thus, instead of computing $R_{xx}(t+\tau, t)$, we can compute the average $\bar{R}_{xx}(\tau)$:

$$\bar{R}_{xx}(\tau) = \frac{1}{T}\int_{-T/2}^{+T/2} R_{xx}(t+\tau, t)dt \qquad [A.3]$$

By developing $R_{xx}(t+\tau, t)$, we get:

$$\begin{aligned}
\bar{R}_{xx}(\tau) &= \frac{1}{T}\int_{-T/2}^{+T/2} \sum_{k=-\infty}^{+\infty}\sum_{i=-\infty}^{+\infty} E[a_i a_k^*]g^*(t-kT)g(t+\tau-iT)dt \\
&= \frac{1}{T}\int_{-T/2}^{+T/2} \sum_{k=-\infty}^{+\infty}\sum_{i=-\infty}^{+\infty} R_{aa}(i-k)g^*(t-kT)g(t+\tau-iT)dt \\
&= \sum_{i'=-\infty}^{+\infty} R_{aa}(i') \sum_{k=-\infty}^{+\infty} \frac{1}{T}\int_{-T/2}^{+T/2} g^*(t-kT)g(t+\tau-i'T-kT)dt \\
&= \sum_{i'=-\infty}^{+\infty} R_{aa}(i') \sum_{k=-\infty}^{+\infty} \frac{1}{T}\int_{-T/2+kT}^{+T/2+kT} g^*(t)g(t+\tau-i'T)dt
\end{aligned} \qquad [A.4]$$

where in the next to last line, we have set $i' = i - k$.

The autocorrelation function of $g(t)$ is:

$$R_{gg}(\tau) = \int_{-\infty}^{+\infty} g^*(t)g(t+\tau)dt \qquad [A.5]$$

We finally obtain:

$$\bar{R}_{xx}(\tau) = \frac{1}{T}\sum_{i'=-\infty}^{+\infty} R_{aa}(i')R_{gg}(\tau - i'T) \qquad [A.6]$$

Thus $$\gamma_{AA}(f) = \sum_{i=-\infty}^{+\infty} R_{aa}(i)e^{-j2\pi f iT} \qquad [A.7]$$

and $$|G(f)|^2 = TF(R_{gg}(\tau)) \qquad \text{[A.8]}$$

Consequently:

$$\gamma_{XX}(f) = \frac{1}{T}|G(f)|^2\gamma_{AA}(f) \qquad \text{[A.9]}$$

This equation is called Bennett's formula.

A.1. Special case 1

We first consider the specific case when symbols a_k are independent, with zero mean $E[a_k] = 0$ and of variance $E[(a_k)^2] = \sigma^2$.

The autocorrelation function $R_{aa}(i)$ is equal to:

$$R_{aa}(i) = \begin{cases} \sigma^2 & \text{if} \quad i = 0 \\ 0 & \text{elsewhere} \end{cases} \qquad \text{[A.10]}$$

Then, $\gamma_{AA}(f) = \sigma^2$.

Consequently, we have:

$$\gamma_{XX}(f) = \frac{\sigma^2}{T}|G(f)|^2 \qquad \text{[A.11]}$$

A.2. Special case 2

We then consider the case where symbols a_k are independent, with non-zero mean μ and centered variance $E[(a_k)^2] = \sigma^2$.

The autocorrelation function $R_{aa}(i)$ is equal to:

$$R_{aa}(i) = \begin{cases} \mu^2 + \sigma^2 & \text{if} \quad i = 0 \\ \mu^2 & \text{elsewhere} \end{cases} \qquad \text{[A.12]}$$

Then:

$$\gamma_{AA}(f) = \sum_{i=-\infty}^{+\infty} R_{aa}(i) e^{-j2\pi f i T}$$
$$= \sigma^2 + \mu^2 \sum_{i=-\infty}^{+\infty} e^{-j2\pi f i T}$$
$$= \sigma^2 + \frac{\mu^2}{T} \sum_{i=-\infty}^{+\infty} \delta\Big(f - \frac{i}{T}\Big) \qquad [A.13]$$

Equality $U(f) = \sum_{i=-\infty}^{+\infty} e^{-j2\pi f i T} = \frac{1}{T}\sum_{i=-\infty}^{+\infty} \delta\Big(f - \frac{i}{T}\Big)$ is proven as follows: since summation occurs on infinite bounds, equation:

$$e^{-j2\pi f T} U(f) = U(f) \qquad [A.14]$$

holds. This implies that for any value of f, $U(f)$ fulfills:

$$\Big(1 - e^{-j2\pi f T}\Big) U(f) = 0 \qquad [A.15]$$

Thus, function $U(f)$ is equal to 0 for any value of f which is not a multiple in $1/T$. It is different from zero on the Dirac comb at frequency $F = 1/T$, with amplitude α_i per frequency i/T:

$$U(f) = \sum_{i=-\infty}^{+\infty} \alpha_i \delta\Big(f - \frac{i}{T}\Big) \qquad [A.16]$$

Yet, since the summation is on infinite bounds, $U(f)$ is invariant to translations of any multiple in $1/T$:

$$U\left(f + \frac{k}{T}\right) = U(f) \quad \forall k \in \mathbb{Z} \qquad [A.17]$$

Equation [A.17] implies that all α_i are equal. Moreover, it can be shown that they are equal to $1/T$.

The power spectrum density is finally written as:

$$\gamma_{XX}(f) = \frac{\sigma^2}{T}|G(f)|^2 + \frac{\mu^2}{T^2}\sum_{i=-\infty}^{+\infty}\left|G\left(\frac{i}{T}\right)\right|^2\delta\left(f - \frac{i}{T}\right) \qquad [A.18]$$

We can see that if the mean μ is different from zero, the power spectrum density is composed of two terms. The first term corresponds to the continuous spectrum and is directly related to the frequency response $G(f)$. The second term corresponds to spectral lines at all frequencies multiple of $1/T$.

We may state that in practical communication systems, it is preferable to have signals with zero means.

Appendix B

Polyphase Implementation of FBMC/OQAM

This section details the polyphase implementation of filter bank multicarrier (FBMC)/offset quadrature amplitude modulation (OQAM) multi-carrier modulations.

B.1. Polyphase representation

The transmit filter G is a finite impulse response filter containing $L_G = KN$ coefficients, whose z transform is:

$$G(z) = \sum_{l=0}^{L_G-1} g[l]z^{-l} \qquad [B.1]$$

K is the overlapping factor. $G(z)$ can be decomposed into N elementary filters. Thus, it generates a polyphase network:

$$\begin{aligned} G(z) &= \sum_{l=0}^{L_G-1} g[l]z^{-l} = \sum_{n=0}^{N-1} \sum_{k=0}^{K-1} g[kN+n]z^{-(kN+n)} \\ &= \sum_{n=0}^{N-1} \left[\sum_{k=0}^{K-1} g[kN+n]z^{-kN} \right] z^{-n} \\ &= \sum_{n=0}^{N-1} E_n(z^N)z^{-n} \end{aligned} \qquad [B.2]$$

Digital Communications 2: Digital Modulations, First Edition. Mylène Pischella and Didier Le Ruyet.

where $E_n(z^N) = \sum_{k=0}^{K-1} g[kN+n]z^{-kN}$ are the polyphase components of $G(z)$.

Let us now denote by $G_i(z)$ the filter deduced from $G(z)$ by a frequency shift of i/N:

$$G_i(z) = \sum_{l=0}^{L_G-1} g[l]e^{j\frac{2\pi}{N}il}z^{-l} \qquad \text{[B.3]}$$

Using polyphase representation, $G_i(z)$ is written as:

$$\begin{aligned} G_i(z) &= \sum_{n=0}^{N-1}\sum_{k=0}^{K-1} g[kN+n]e^{j\frac{2\pi}{N}i(kN+n)}z^{-(kN+n)} \\ &= \sum_{n=0}^{N-1} e^{j\frac{2\pi}{N}ni}E_n(z^N)z^{-n} \end{aligned} \qquad \text{[B.4]}$$

A uniform filter bank is obtained by shifting the response of $G(z)$ on the frequency axis. It is thus deduced from $G(z)$, which is, therefore, called the prototype filter of the filter bank.

If all filters are deduced from the prototype filter by frequency shift multiples in $1/N$, then the filter bank has the following expression:

$$\begin{bmatrix} G_0(z) \\ G_1(z) \\ \vdots \\ G_{N-1}(z) \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & w^{-1} & \cdots & w^{-(N-1)} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & w^{-(N-1)} & \ldots & w^{-(N-1)^2} \end{bmatrix}}_{IFFT} \begin{bmatrix} E_0(z^N) \\ z^{-1}E_1(z^N) \\ \vdots \\ z^{-(N-1)}E_N(z^N) \end{bmatrix} \qquad \text{[B.5]}$$

where $w = e^{-j\frac{2\pi}{N}}$, and the N-order inverse Fourier transform matrix is denoted by $IFFT$.

B.2. Transmission process: polyphase synthesis filter banks

The transmit signal is obtained by transmitting in parallel data $X^k[n]$ on the modulated filter banks. Multiplication by the N-order inverse Fourier transform matrix can be exchanged with the upsampling operation. Moreover, applying the first Noble identity [FAR 05], upsampling can also be exchanged with the polyphase components, which implies replacing $E_n(z^N)$ by $E_n(z)$.

Polyphase implementation of synthesis filter banks can consequently be simplified. This is represented in Figure A.1.

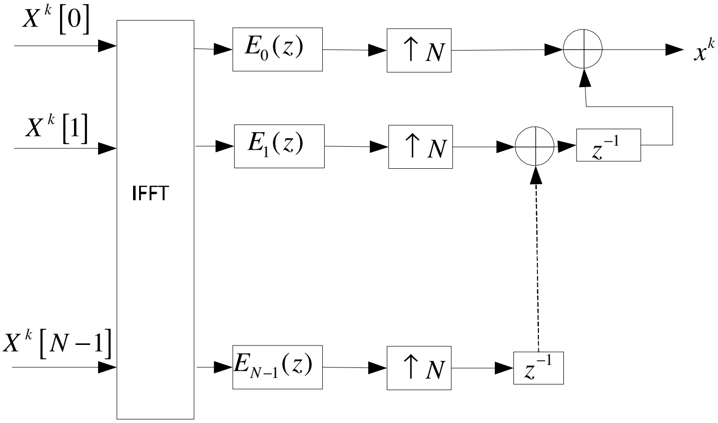

Figure A.1. *Simplified polyphase synthesis filter bank block diagram*

B.3. Receiver process: polyphase analysis filter banks

At the receiver, the transmit signal is obtained by decomposing the input signal in the frequency domain with the analysis filter bank.

The analysis filter bank has a dual structure to that of the synthesis filter bank. It is similarly obtained.

Let $G_{-i}(z)$ be the filter deduced from $G(z)$ after a frequency shift of $-i/N$:

$$G_{-i}(z) = \sum_{l=0}^{L_G-1} g[l]e^{-j\frac{2\pi}{N}il}z^{-l} \qquad [B.6]$$

Using polyphase representation, $G_{-i}(z)$ becomes:

$$\begin{aligned} G_{-i}(z) &= \sum_{n=0}^{N-1}\sum_{k=0}^{K-1} g[kN+n]e^{-j\frac{2\pi}{N}i(kN+n)}z^{-(KN+n)} \\ &= \sum_{n=0}^{N-1} e^{-j\frac{2\pi}{N}ni}E_n(z^N)z^{-n} \end{aligned} \qquad [B.7]$$

If we consider all filters deduced from $G(z)$ by negative frequency shifts of $1/N$, we get:

$$\begin{bmatrix} G_0(z) \\ G_{-1}(z) \\ \vdots \\ G_{-(N-1)}(z) \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & w & \cdots & w^{(N-1)} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & w^{(N-1)} & \cdots & w^{(N-1)^2} \end{bmatrix}}_{FFT} \begin{bmatrix} E_0(z^N) \\ z^{-1}E_1(z^N) \\ \vdots \\ z^{-(N-1)}E_N(z^N) \end{bmatrix} \qquad [B.8]$$

where FFT is the N-order Fourier transform matrix.

The input signal first passes in a chain that generates delays, and is then filtered by the polyphases components. The filtering operation is followed by a Fourier transform.

The Fourier transform may be exchanged with downsampling. Furthermore, the first Noble identity indicates that downsampling can be exchanged with the polyphase components. The analysis filter bank thus has a simplified structure, which is represented in Figure A.2.

B.4. FBMC/OQAM polyphase block diagram

The transmit filter bank's block diagram is represented in Figure A.3. It may be separated into two subsystems: the first subsystem uses prototype filter $g(t)$ and the second subsystem uses prototype filter $g(t - \frac{T_N}{2})$.

The delay of $N/2$ samples, which is equivalent to half the multi-carrier symbol-period $T_N/2$, can be shifted from $g(n - N/2)$ to outside the second subsystem. The transmitter's polyphase block diagram is then obtained by replacing each subsystem with its polyphase implementation (see Figure A.4).

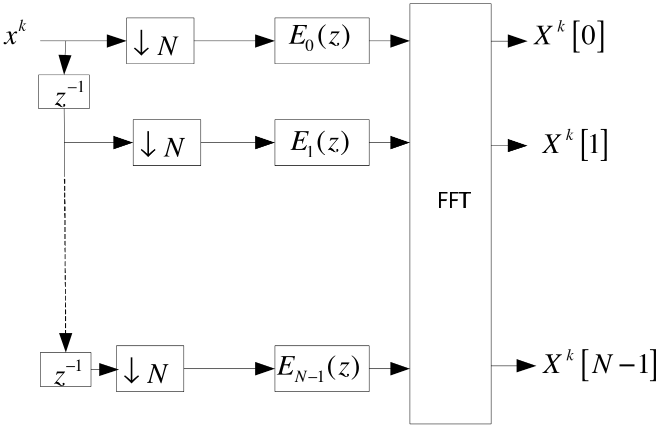

Figure A.2. *Simplified analysis filter bank block diagram*

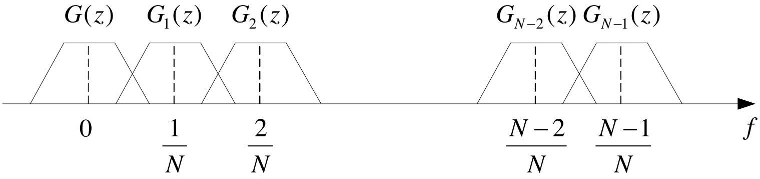

Figure A.3. *Frequency response of filter banks*

Similarly, the block diagram of the FBMC receiver can be separated into two subsystems using, respectively, the prototype filters $g(t)$ and $g(t + T_N/2)$. The delay of $-N/2$ samples can be shifted at the input of the second subsystem. Using the analysis filter bank from Figure A.2, the block diagram of the polyphase FBMC/OQAM receiver is obtained. This is represented in Figure A.5.

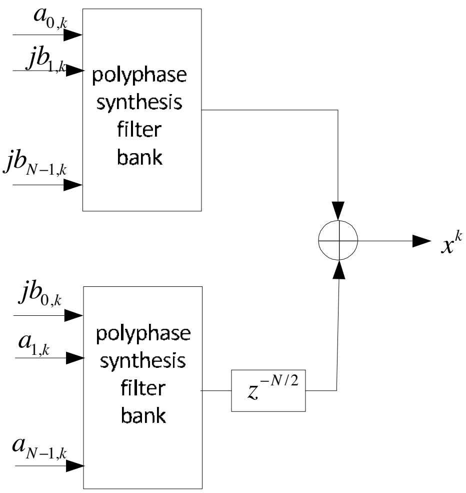

Figure A.4. *Block diagram of FBMC/OQAM polyphase transmitter*

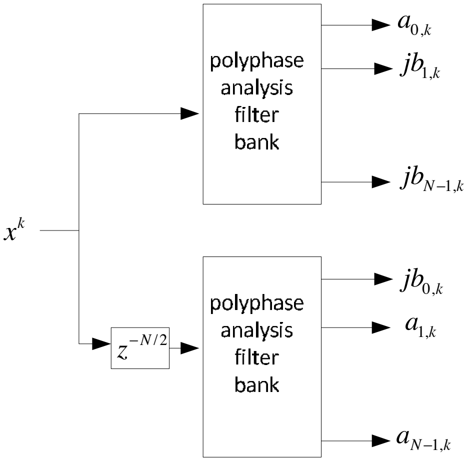

Figure A.5. *Block diagram of FBMC/OQAM polyphase receiver*

Bibliography

[ALA 96] ALARD M., Construction of a multicarrier signal, patent no. WO/1996/035278, November 1996.

[AMR 94] AMRANI O., BE'ERY Y., VARDY A. *et al.*, "The Leech lattice and the Golay code: bounded-distance decoding and multilevel constructions", *IEEE Transactions on Information Theory*, vol. 40, no. 4, pp. 1030–1043, 1994.

[AND 07] ANDREWS J.G., GHOSH A., MUHAMED R., *Fundamentals of WiMAX: understanding broadband wireless networking*, Prentice Hall Communications Engineering and Emerging Technologies Series, Boston, 2007.

[BÖL 03] BÖLCSKEI H., "Orthogonal frequency division multiplexing based on offset QAM", *Advances in Gabor Analysis*, Birkhäuser, Boston, pp. 321–352, 2003.

[BAU 96] BAUML R.W., FISHER R.F., HUBERT J.B., "Reducing the peak-to-average power ratio of multicarrier modulation by selected mapping", *IEE Electronic Letters*, vol. 32, no. 22, pp. 2056–2057, 1996.

[BE' 89] BE'ERY Y., SHAHAR B., SNYDERS J., "Fast decoding of the Leech lattice", *IEEE Journal on Selected Areas in Communications*, vol. 7, no. 6, pp. 959–967, 1989.

[BEL 01] BELLANGER M., "Specification and design of a prototype filter for filter bank based multicarrier transmission", *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, Salt Lake City, UT, USA, vol. 4, pp. 2417–2420, May 2001.

[BEN 99] BENEDETTO S., BIGLIERI E., *Principles of Digital Transmission: with Wireless Applications*, Kluwer Academic/ Plenum publishers, New York, 1999.

[BOT 11] BOTTOMLEY G.E., *Channel Equalization for Wireless Communications: From Concepts to Detailed Mathematics*, Wiley-IEEE Press, New York, 2011.

[BOY 04] BOYD S., VANDERBERGUE L., *Convex Optimization*, Cambridge University Press, Cambridge, 2004.

Digital Communications 2: Digital Modulations, First Edition. Mylène Pischella and Didier Le Ruyet.

[CAM 98] CAMPELLO J., "Optimal discrete bit loading for multicarrier modulation systems", *Proceedings of IEEE International Symposium on Information Theory*, Cambridge, MA, USA, August 1998.

[CAR 06] CARTWRIGHT K.V., KAMINSKY E.J., "A Simple Improvement to the viterbi and viterbi monomial-based phase estimators", *Proceedings of IEEE Global Communications Conference*, San Francisco, CA, USA, November 2006.

[CHA 66] CHANG R.W., "High-speed multichannel data transmission with bandlimited orthogonal signals", *Bell System Technical Journal*, vol. 15, no. 6, pp. 1775–1796, December 1966.

[CHE 13] CHEN F., SILVA D., KSCHISCHANG F.R., "An algebraic approach to physical-layer network coding", *IEEE Transactions on Information Theory*, vol. 59, no. 11, pp. 7576–7596, 2013.

[CON 99] CONWAY J., SLOANE N., *Sphere Packing, Lattices and Groups*, Springer, Third edition, New York, 1999.

[D5. 09] D5.1 D., Prototype Filter and Structure Optimization, Report, European project ICT-211887 PHYDYAS, January 2009.

[D'AN 94] D'ANDREA A.N., MENGALI U., REGGIANNINI R., "The modified Cramer-Rao bound and its application to synchronization problems", *IEEE Transactions on Communications*, vol. 42, pp. 1391–1399, 1994.

[DAH 11] DAHLMAN E., PARKVALL S., SKOLD J., *4G: LTE/LTE-Advanced for Mobile Broadband*, Academic Press, Amsterdam, 2011.

[DUR 10] DURRETT R., *Probability: Theory and Examples*, fourth edition, Cambridge series in statistical and probabilistic mathematics, Cambridge, 2010.

[FAR 03] FARHANG-BOROUJENY B., "Multicarrier modulation with blind detection capability using cosine modulated filter banks", *IEEE Transactions on Communications*, vol. 51, no. 12, pp. 2057–2070, December 2003.

[FAR 05] FARHANG-BOROUJENY B., *Signal Processing Techniques for Software Radios*, LuLu Publishing, Morrisville, 2005.

[FIS 02] FISCHER R.F.H., *Precoding and Signal Shaping for Digital Transmission*, IEEE Wiley-Interscience, New York, 2002.

[FOR 84] FORNEY JR G., GALLAGER R., LANG G., *et al.*, "Efficient modulation for band-limited channels", *IEEE Journal on Selected Areas in Communications*, vol. 2, pp. 632–647, 1984.

[FOR 88a] FORNEY JR G., "Coset codes – Part I: introduction and geometrical classification", *IEEE Transactions on Information Theory*, vol. 34, no. 5, pp. 1123–1151, 1988.

[FOR 88b] FORNEY JR G., "Coset codes – Part II: Binary lattices and related codes", *IEEE Transactions on Information Theory*, vol. 34, no. 5, pp. 1152–1187, 1988.

[FOR 96] FORNEY G.D., BROWN L., EYOBOGLU M.V. *et al.*, "The V.34 high-speed modem standard", *IEEE Communication Magazine*, vol. 34, no. 12, pp. 28–33, 1996.

[FOR 98] FORNEY G.D., UNGERBOECK G., "Modulation and coding for linear Gaussian channels", *IEEE Transactions on Information Theory*, vol. 44, no. 6, pp. 2384–2415, 1998.

[GOL 05] GOLDSMITH A., *Wireless Communications*, Cambridge University Press, New York, 2005.

[HEE 99] HEEGARD C., WICKER S., *Turbo coding*, Springer Science and Business Media, New York, 1999.

[IMM 98] IMMINK K., SIEGEL P., WOLF J., "Codes for digital recorders", *IEEE Transactions on Information Theory*, vol. 44, no. 6, pp. 2260–2299, 1998.

[JOH 99] JOHANNESSON R., ZIGANGIROV K., *Fundamentals of Convolutional Coding*, IEEE Press, New York, 1999.

[JON 94] JONES A.E., WILKINSON T.A., BARTON S.K., "Block coding scheme for reduction of peak to mean enveloppe power ratio of multicarrier transmission schemes", *IEE Electronic Letters*, vol. 30, pp. 2098–2099, 1994.

[KOB 71] KOBAYASHI H., "Simultaneous adaptative estimation and decision algorithm for carrier modulated data transmission systems", *IEEE Transactions on Communications*, vol. 19, pp. 268–280, 1971.

[KOE 04] KOETTER R., SINGER A., "Turbo equalization", *IEEE Signal Processing Magazine*, vol. 21, no. 1, pp. 67–80, 2004.

[LEF 95] LE FLOCH B., ALARD M., BERROU C., "Coded Orthogonal Frequency Division Multiplex [TV broadcasting]", *Proceedings of the IEEE*, vol. 83, no. 6, pp. 982–996, June 1995.

[LER 15] LE RUYET O., PISCHELLA M., *Digital Communications 1: Source and Channel Coding*, ISTE Ltd London and John Wiley & Sons, New York, 2015.

[LI 97] LI X., CIMINI JR L.J., "Effect of clipping and filtering on the performance of OFDM", *Proceedings of IEEE VTC*, Phoenix, USA, May 1997.

[LOS 09] LOSKOT P., BEAULIEU N.C., "A unified approach to computing error probabilities of diversity combining schemes over correlated fading channels", *IEEE Trans. on Communications*, vol. 57, no. 7, pp. 2031–2041, 2009.

[MED 09] MEDJAHD I.Y., TERRE M., LE RUYET D. *et al.*, "Inter-cell Interference Analysis for OFDM / FBMC Systems", *Proceedings of IEEE Workshop on Signal Processing Advances in Wireless Communications*, Perugia, Italy, June 2009.

[MEY 80] MEYERS M.H., FRANKS L.E., "Joint carrier and symbol timing recovery for PAM systems", *IEEE Transactions on Communications*, vol. 28, pp. 1121–1129, 1980.

[MEY 98] MEYR H., MOENECLAEY M., FECHTEL S.A., *Digital Communication Receivers – Synchronization, Channel Estimation and Signal Processing*, Wiley series in Telecommunications and Signal Processing, New York, 1998.

[MIN 03] MINN H., BHARGAVA V.K., LETAIEF K.B., "A robust timing and frequency synchronization for OFDM systems", *IEEE Transactions on Wireless Communications*, vol. 2, pp. 822–839, 2003.

[MOO 94] MOOSE P.H., "A technique for orthogonal frequency division multiplexing frequency offset correction", *IEEE Transactions on Communications*, vol. 42, pp. 2908–2914, 1994.

[MOR 99] MORELLI M., MENGALI U., "An improved frequency offset estimator for OFDM applications", *IEEE Communication Letters*, vol. 3, pp. 75–77, 1999.

[MOR 07] MORELLI M., KUO CC J., PUN M.-O., "Synchronization techniques for orthogonal frequency division multiple access (OFDMA): a tutorial review", *Proceedings of the IEEE*, vol. 95, no. 7, pp. 1394–1427, 2007.

[MUL 97] MULLER S.H., HUBER J.B., "OFDM with reduced peak-to-average power ratio by optimum combination of partial transmit sequences", *IEE Electronic Letters*, vol. 33, no. 5, pp. 36–69, 1997.

[NAZ 11] NAZER B., GASTPAR M., "Compute-and-forward: harnessing interference through structured codes", *IEEE Transactions on Information Theory*, vol. 57, no. 10, pp. 6463–6486, 2011.

[OER 88] OERDER M., MEYR H., "Digital filter and square timing recovery", *IEEE Transactions on Communications*, vol. 36, no. 5, pp. 3208–3214, 1988.

[O'NE 95] O'NEILL R., LOPES L.B., "Enveloppe variations and spectral splatter in clipped multicarrier signals", *Proceedings of IEEE International Symposium on Personal, Indoor Mobile Radio Communications*, Toronto, Canada, September 1995.

[OPP 96] OPPENHEIM A.V., WILLSKY A.S., HAMID S., *Signals and Systems*, 2nd edition, Prentice Hall, Boston, 1996.

[PAP 02] PAPOULIS A., PILLAI. S., *Probability, Random Variables and Stochastic Processes*, McGraw Hill Series in Electrical Engineering, New York, 2002.

[PAP 07] PAPANDREOU N., ANTONAKOPOULUS T., "Bit and power allocation in constrained multicarrier systems:the single-user case", *EURASIP Journal on Advanced in Signal Processing*, 2007.

[POP 91] POPOVIC B.M., "Synthesis of power efficient multitone signals with flat amplitude spectrum", *IEEE Transactions on Communications*, vol. 39, no. 7, pp. 1031–1033, 1991.

[PRI 72] PRICE R., "Nonlinearly feedback-equalized PAM vs. capacity", *Procedings of IEEE International Conference on Communications*, Philadelphia, PA, USA, June 1972.

[PRO 08] PROAKIS J.G., SALEHI M., *Digital Communications*, Fifth Edition, McGraw Hill Higher Education, 2008.

[SAL 67] SALTZBERG B., "Performance of an Efficient Parallel Data Transmission System", *IEEE Transactions on Communication Technology*, vol. 15, no. 6, pp. 805–811, December 1967.

[SAL 73] SALZ J., "Optimum mean-square decision-feedback equalization", *Bell System Technical Journal*, pp. 1341–1373, 1973.

[SAN 95] SANDBERG S., TZANNES M., "Overlapped discrete multitone modulation for high speed copper wire communications", *IEEE Jounrnal on Selected Areas in Communications*, vol. 13, no. 9, pp. 1571–1585, December 1995.

[SAP 90] SAPORTA G., *Probabilités, analyse de données et statistiques*, Editions Technip, Paris, 1990.

[SCH 97] SCHMIDL T.M., COX D.C., "Robust frequency and timing synchronization for OFDM", *IEEE Communication Letters*, vol. 45, pp. 1613–1621, 1997.

[SHA 48] SHANNON C.E., "A mathematical theory of communication", *Bell System Technical Journal*, vol. 27, pp. 623–659, 1948.

[SHI 04] SHI K., SERPEDIN E., "Coarse frame and carrier synchronization of OFDM systems: a new metric and comparison", *IEEE Transactions on Wireless Communications*, vol. 3, pp. 1271–1284, 2004.

[SIO 00] SIOHAN P., ROCHE C., "Cosine-modulated filterbanks based on extended Gaussian functions", *IEEE Transactions on Signal Processing*, vol. 48, no. 11, pp. 3052–3061, November 2000.

[SIO 02] SIOHAN P., SICLET C., LACAILLE N., "Analysis and design of OFDM/OQAM systems based on filter bank theory,", *IEEE Transactions on Signal Processing*, vol. 50, no. 5, pp. 1170–1183, May 2002.

[TÜC 11] TÜCHLER M., SINGER A., "Turbo equalization: an overview", *IEEE Transactions on Information Theory*, vol. 57, no. 2, pp. 920–952, 2011.

[TER 13] TERRE M., PISCHELLA M., VIVIER E., *Wireless Telecommunication Systems*, ISTE, London and John Wiley & Sons, New York, 2013.

[UNG 82] UNGERBOECK G., "Channel coding with multilevel/phase signals", *IEEE Transactions on Information Theory*, vol. 28, pp. 55–67, 1982.

[VAN 68] VAN TREES H.L., *Detection, Estimation and Modulation Theory*, John Wiley & Sons, New York, 1968.

[VAN 97] VAN DE BEEK J.J., SANDELL M., BORJESSON P. O., "ML estimation of timing and frequency offset in OFDM systems", *IEEE Transactions on Signal Processing*, vol. 45, pp. 1800–1805, 1997.

[VIT 83] VITERBI A.J., VITERBI A.M., "Non-linear estimation of PSK-modulated carrier phase with applications to burst digital transmission", *IEEE Transactions on Information Theory*, vol. 29, pp. 543–551, 1983.

[WAN 02] WANG Y., SERPEDIN E., "A class of blind phase recovery techniques for higher order QAM modulations: estimators and bounds", *IEEE Signal Processing Letters*, vol. 9, no. 10, pp. 301–304, 2002.

[WAN 03] WANG Y., SERPEDIN E., CIBLAT P., "Optimal blind nonlinear leastsquares carrier phase and frequency offset estimation for general QAM modulations", *IEEE Transactions on Wireless Communications*, vol. 2, pp. 1040–1054, 2003.

Index

Digital Communications 2: Digital Modulations, First Edition. Mylène Pischella and Didier Le Ruyet.

Summary of Volume 1:
Digital Communications 1

Digital Communications 2: Digital Modulations, First Edition. Mylène Pischella and Didier Le Ruyet.

Other titles from

ISTE

in

Networks and Telecommunications

2015

BENSLAMA Malek, KIAMOUCHE Wassila, BATATIA Hadj
Connections Management Strategies in Satellite Cellular Networks

BENSLAMA Malek, BATATIA Hadj, BOUCENNA Mohamed Lamine
Ad Hoc Networks Telecommunications and Game Theory

BERTHOU Pascal, BAUDOIN Cédric, GAYRAUD Thierry, GINESTE Matthieu
Satellite and Terrestrial Hybrid Networks

PUJOLLE Guy
Software Networks

2014

ANJUM Bushra, PERROS Harry
Bandwidth Allocation for Video under Quality of Service Constraints

BATTU Daniel
New Telecom Networks: Enterprises and Security

BEN MAHMOUD Mohamed Slim, GUERBER Christophe, LARRIEU Nicolas, PIROVANO Alain, RADZIK José
Aeronautical Air−Ground Data Link Communications

BITAM Salim, MELLOUK Abdelhamid
Bio-inspired Routing Protocols for Vehicular Ad-Hoc Networks

CAMPISTA Miguel Elias Mitre, RUBINSTEIN Marcelo Gonçalves
Advanced Routing Protocols for Wireless Networks

CHETTO Maryline
Real-time Systems Scheduling 1: Fundamentals
Real-time Systems Scheduling 2: Focuses

EXPOSITO Ernesto, DIOP Codé
Smart SOA Platforms in Cloud Computing Architectures

MELLOUK Abdelhamid, CUADRA-SANCHEZ Antonio
Quality of Experience Engineering for Customer Added Value Services

OTEAFY Sharief M.A., HASSANEIN Hossam S.
Dynamic Wireless Sensor Networks

PEREZ André
Network Security

PERRET Etienne
Radio Frequency Identification and Sensors: From RFID to Chipless RFID

REMY Jean-Gabriel, LETAMENDIA Charlotte
LTE Standards
LTE Services

TANWIR Savera, PERROS Harry
VBR Video Traffic Models

VAN METER Rodney
Quantum Networking

XIONG Kaiqi
Resource Optimization and Security for Cloud Services

2013

ASSING Dominique, CALÉ Stéphane
Mobile Access Safety: Beyond BYOD

BEN MAHMOUD Mohamed Slim, LARRIEU Nicolas, PIROVANO Alain
Risk Propagation Assessment for Network Security: Application to Airport Communication Network Design

BEYLOT André-Luc, LABIOD Houda
Vehicular Networks: Models and Algorithms

BRITO Gabriel M., VELLOSO Pedro Braconnot, MORAES Igor M.
Information-Centric Networks: A New Paradigm for the Internet

BERTIN Emmanuel, CRESPI Noël
Architecture and Governance for Communication Services

DEUFF Dominique, COSQUER Mathilde
User-Centered Agile Method

DUARTE Otto Carlos, PUJOLLE Guy
Virtual Networks: Pluralistic Approach for the Next Generation of Internet

FOWLER Scott A., MELLOUK Abdelhamid, YAMADA Naomi
LTE-Advanced DRX Mechanism for Power Saving

JOBERT Sébastien *et al.*
Synchronous Ethernet and IEEE 1588 in Telecoms: Next Generation Synchronization Networks

MELLOUK Abdelhamid, HOCEINI Said, TRAN Hai Anh
Quality-of-Experience for Multimedia: Application to Content Delivery Network Architecture

NAIT-SIDI-MOH Ahmed, BAKHOUYA Mohamed, GABER Jaafar,
WACK Maxime
Geopositioning and Mobility

PEREZ André
Voice over LTE: EPS and IMS Networks

2012

AL AGHA Khaldoun
Network Coding

BOUCHET Olivier
Wireless Optical Communications

DECREUSEFOND Laurent, MOYAL Pascal
Stochastic Modeling and Analysis of Telecoms Networks

DUFOUR Jean-Yves
Intelligent Video Surveillance Systems

EXPOSITO Ernesto
Advanced Transport Protocols: Designing the Next Generation

JUMIRA Oswald, ZEADALLY Sherali
Energy Efficiency in Wireless Networks

KRIEF Francine
Green Networking

PEREZ André
Mobile Networks Architecture

2011

BONALD Thomas, FEUILLET Mathieu
Network Performance Analysis

CARBOU Romain, DIAZ Michel, EXPOSITO Ernesto, ROMAN Rodrigo
Digital Home Networking

CHABANNE Hervé, URIEN Pascal, SUSINI Jean-Ferdinand
RFID and the Internet of Things

GARDUNO David, DIAZ Michel
Communicating Systems with UML 2: Modeling and Analysis of Network Protocols

LAHEURTE Jean-Marc
Compact Antennas for Wireless Communications and Terminals: Theory and Design

RÉMY Jean-Gabriel, LETAMENDIA Charlotte
Home Area Networks and IPTV

PALICOT Jacques
Radio Engineering: From Software Radio to Cognitive Radio

PEREZ André
IP, Ethernet and MPLS Networks: Resource and Fault Management

TOUTAIN Laurent, MINABURO Ana
Local Networks and the Internet: From Protocols to Interconnection

2010

CHAOUCHI Hakima
The Internet of Things

FRIKHA Mounir
Ad Hoc Networks: Routing, QoS and Optimization

KRIEF Francine
Communicating Embedded Systems / Network Applications

2009

CHAOUCHI Hakima, MAKNAVICIUS Maryline
Wireless and Mobile Network Security

VIVIER Emmanuelle
Radio Resources Management in WiMAX

2008

CHADUC Jean-Marc, POGOREL Gérard
The Radio Spectrum

GAÏTI Dominique
Autonomic Networks

LABIOD Houda
Wireless Ad Hoc and Sensor Networks

LECOY Pierre
Fiber-optic Communications

MELLOUK Abdelhamid
End-to-End Quality of Service Engineering in Next Generation Heterogeneous Networks

PAGANI Pascal *et al.*
Ultra-wideband Radio Propagation Channel

2007

BENSLIMANE Abderrahim
Multimedia Multicast on the Internet

PUJOLLE Guy
Management, Control and Evolution of IP Networks

SANCHEZ Javier, THIOUNE Mamadou
UMTS

VIVIER Guillaume
Reconfigurable Mobile Radio Systems

Humans have always used communication systems: first with smoke signals, and then with the telegraph and telephone. These systems and their technological advances have changed our lifestyle profoundly.

Nowadays, smartphones enable us to make calls, watch videos and use social networks, and with this comes the emergence of the connected man and the wider applications of smart objects. All current and future communication systems rely on a digital communication chain that consists of a source and a destination separated by a transmission channel, which may be a portion of a cable, an optical fiber, a wireless mobile or satellite channel. Whichever channel, the processing blocks implemented in the communication chain have the same basis.

This second volume of Digital Communications concerns the blocks located after channel coding in the communication chain. The authors present baseband and sine waveform transmissions and the different steps required at the receiver to perform detection, namely synchronization and channel estimation. Two variants of these blocks used in current and future systems, multi-carrier modulations and coded modulations, are also detailed.

Mylène Pischella is Associate Professor in Telecommunications at CNAM in Paris, France. Her research interests include resource allocation in multi-cellular networks, and cooperative and relay-assisted networks.

Didier Le Ruyet is Professor of Signal Processing for Communications at CNAM in Paris, France. His research interests include multi-user cooperative communications, detection algorithms for wireless communication, multicarrier modulation and cognitive radio.

iSTE
www.iste.co.uk

WILEY