

UNIVERSITÀ DEGLI STUDI DI PARMA

Dottorato di Ricerca in Tecnologie dell'Informazione

XXI Ciclo

**ADVANCED MODULATION/DEMODULATION
SCHEMES FOR WIRELESS COMMUNICATIONS**

Coordinatore:

Chiar.mo Prof. Carlo Morandi

Tutor:

Chiar.mo Prof. Giulio Colavolpe

Dottorando: *Aldo Cero*

Gennaio 2009

*alla mia famiglia
e ai miei amici*

Contents

List of figures	viii
List of acronyms	ix
Foreword	1
1 Introduction	3
1.1 Background and Objectives	3
1.2 Continuous Phase Modulation Signals	7
1.3 General Frameworks	10
1.3.1 MAP Symbol Detection Strategy and BCJR Algorithm	10
1.3.2 Factor Graphs and Sum Product Algorithm	13
1.3.3 Iterative Joint Detection/Decoding Schemes	14
2 Capacity Evaluation for CPM signals	17
2.1 Introduction	17
2.2 Information Rate for Channel with Memory	19
2.2.1 Information Rate Definition	19
2.2.2 Arnold and Loeliger Method	20
2.3 IR of CPMs over AWGN	22
2.3.1 Optimal MAP Symbol Detection	22
2.4 IR of CPMs over Channel Affected by Phase Noise	26
2.4.1 Wiener Phase Noise Model	26

2.4.2	SATMODE Phase Noise Model	28
2.4.3	Detectors for IR Computation over Channels Affected by Phase Noise	38
2.4.4	Numerical results	45
2.5	Capacity Improvement: Shaper-Precoder Optimization	53
2.5.1	Problem Formulation	59
2.5.2	The Carson's Bandwidth for Correlated Input	62
2.5.3	Numerical Results	67
3	CPM Reduced-Complexity Soft-Output Detection	93
3.1	Introduction	94
3.2	Detector Model and Complexity Evaluation	96
3.3	Algorithms Based on MM Decomposition	98
3.3.1	MM Decomposition	98
3.3.2	MAP Symbol Detection Based on the MM Decomposition	100
3.4	Complexity-Reduction Based on the G Decomposition	106
3.4.1	The G Decomposition	106
3.4.2	MAP Symbol Detection Based on the G Decomposition	108
3.5	Complexity-Reduction Based on the MA Decomposition	109
3.5.1	The MA Decomposition	109
3.5.2	MAP Symbol Detection Based on the MA Decomposition	111
3.6	Reduced-Search Algorithms	113
3.6.1	Rationale	113
3.6.2	Optimization	114
3.6.3	Application to the Trellis of MM Detector	117
3.7	Numerical Results	118
3.7.1	Uncoded CPM Transmissions	119
3.7.2	Iterative Decoding of SCCPM Schemes	123
4	CPM Detection in the Presence of Phase Noise	135
4.1	Introduction	135
4.2	Channel Model and Ideal Coherent MAP Symbol Detection . .	137

4.3	Non-Bayesian Algorithms	139
4.3.1	Rationale	139
4.3.2	Proposed Algorithm	140
4.4	Bayesian Algorithms	145
4.4.1	Derivation of the Algorithms	145
4.4.2	Proposed Algorithm	150
4.4.3	Extension of the Algorithms	154
4.4.4	Bayesian Algorithms based on double-AR1 Phase Noise Model	155
4.5	Numerical results	162
4.5.1	Simulation environment	162
4.5.2	Channels affected by Wiener phase noise	164
4.5.3	Channels affected by the SATMODE phase noise	167
4.5.4	SATMODE phase noise effects at large spectral efficiency	172
5	Multicarrier schemes over doubly-selective channels	175
5.1	Introduction	176
5.2	System Model	178
5.2.1	Uniform Filter-Bank	183
5.2.2	Symbol-by-Symbol Detection	184
5.3	DFT-OFDM	186
5.3.1	DFT-DMT	188
5.3.2	DFT-PS	189
5.4	DCT-OFDM	190
5.4.1	DCT-MAN	192
5.4.2	DCT-PS	193
5.5	DTT-OFDM	193
5.6	ODTT-OFDM	195
5.6.1	Wilson Base Derivation	196
5.6.2	Modulator and Demodulator Derivation	197
5.6.3	ODTT-PS	200

5.7 Signal Spectrum and Bandwidth Computation	201
Conclusions	211
Bibliography	215
Acknowledgements	227

List of Figures

1.1	Block diagram of CPM modulator	10
2.1	Factor graph of the optimal MAP symbol detector in the n -th time interval.	24
2.2	SATMODE PSD of the continuous-time phase process $\theta(t)$. . .	29
2.3	SATMODE PSD mask and PSDs obtained from various double-AR1 approximations.	31
2.4	PSDs of slow and fast AR1 obtained with IA parameters. . . .	32
2.5	Time snapshot of the two phase noise AR1 components, generated according to the IA parameters.	33
2.6	Autocorrelation function of the continuous-time PN process $\theta(t)$ and of the symbol-time PN process ψ_n at 256 kBaud.	35
2.7	Factor graph of DP-BCJR in the n -th time-interval.	40
2.8	FG of Double-DP-BCJR detector in the n -th time-interval. . .	42
2.9	FG of I-DP-BCJR detector in the n -th time-interval.	44
2.10	Information rate for a binary modulation with 2-RC $h = 1/3$ and D-DP detector.	46
2.11	Information rate for a binary modulation with 2-RC $h = 1/3$ and DP and I-DP detectors.	48
2.12	Information rate for a binary modulation with 2-RC $h = 1/3$: comparison between D-DP and DP detectors.	50

2.13	Information rate for a quaternary modulation with 2-RC $h = 1/5$ and D-DP detector.	51
2.14	Information rate for a quaternary modulation with 2-RC $h = 1/5$ and DP and I-DP detectors.	52
2.15	Information rate for a quaternary modulation with 2-RC $h = 1/5$: comparison between D-DP and DP detectors.	54
2.16	Information rate for a quaternary modulation with 2-RC $h = 1/5$ and DP and I-DP detectors at 256 kBaud.	55
2.17	Information rate for a quaternary modulation with 2-RC $h = 1/5$ and DP and I-DP detectors at 2048 kBaud.	56
2.18	Interference due to adjacent channels in FDM systems with different bandwidth assumptions.	69
2.19	Interference when power and information rate of interferers are doubled respect to the CPM signal of interest.	70
2.20	Spectral efficiency for a binary 2-RC with $h = 1/3$	72
2.21	PSD of the CPM signal with i.u.d. inputs and with the optimized 1st- and 2nd-order Markov inputs described in Table 2.4.	74
2.22	Spectral Power Concentration for a binary 2-RC with $h = 1/3$	75
2.23	Autocorrelation function of the CPM transmitted signal for a binary 2-RC with $h = 1/3$	77
2.24	Spectral efficiency for a quaternary 3-RC with $h = 2/7$	78
2.25	PSD and SPC of a CPM signal with i.u.d. input and with the optimized 1st-order Markov input described in Table 2.6.	80
2.26	Autocorrelation function of the CPM transmitted signal for a quaternary 3-RC with $h = 2/7$	81
2.27	Spectral efficiency for an octal 2-RC with $h = 1/7$	82
2.28	PSD and SPC of a CPM signal with i.u.d. input and with the optimized 1st-order Markov input described in Table 2.8.	85
2.29	Autocorrelation function of the CPM transmitted signal for an octal 2-RC with $h = 1/7$	86

2.30	Spectral efficiency for a binary 2-RC with $h = 1/3$. Brute-force optimization w. r. t. the 99.5% bandwidth.	88
2.31	Spectral efficiency for a quaternary 2-REC with $h = 1/4$. Brute-force optimization w. r. t. the 99.5% bandwidth.	89
2.32	Spectral efficiency for a quaternary 3-RC with $h = 2/7$. Brute-force optimization w. r. t. the 99.5% bandwidth.	90
2.33	Information rate for a binary 2-RC with $h = 1/3$. Brute-force optimization w. r. t. the 99.5% bandwidth.	90
2.34	Information rate for a quaternary 2-REC with $h = 1/4$. Brute-force optimization w. r. t. the 99.5% bandwidth.	91
2.35	Information rate for a quaternary 3-RC with $h = 2/7$. Brute-force optimization w. r. t. the 99.5% bandwidth.	91
3.1	CPM detector block diagram.	97
3.2	Factor graph corresponding to eqn. (3.15), in the n -th time interval.	102
3.3	Signal pulses for a 3RC modulation with $h = 2/7$ and $M = 4$	104
3.4	Factor graph corresponding to MM based algorithm with some selected secondary pulses, in the n -th time interval.	105
3.5	2RC modulation with $h = 1/4$ and $M = 4$	121
3.6	2RC modulation with $h = 1/7$ and $M = 8$	121
3.7	3RC modulation with $h = 2/7$ and $M = 4$	122
3.8	3RC modulation with $h = 2/7$ and $M = 4$	122
3.9	Transmitter and receiver structure for the considered SCCPM schemes.	124
3.10	2RC modulation with $h = 1/4$ and $M = 4$. Comparison between symbol and bit interleaver.	126
3.11	2RC modulation with $h = 1/4$ and $M = 4$: FC algorithm with G-based FE and DT-NZ algorithm.	127
3.12	2RC modulation with $h = 1/4$ and $M = 4$: FC algorithm with MA-based algorithm.	128

3.13	3RC modulation with $h = 2/7$ and $M = 4$: performance of MM and DT-NZ algorithms.	131
3.14	3RC modulation with $h = 2/7$ and $M = 4$: performance of FC algorithm with G-based and MA-based FE.	132
4.1	Factor graph corresponding to eqn. (4.22).	148
4.2	MSK modulation, codeword of 2000 bits.	165
4.3	2RC modulation with $h = 2/7$ and $M = 4$, codeword of 2000 bits.	166
4.4	2RC modulation with $h = 1/7$ and $M = 8$, codeword of 1760 bits.	169
4.5	3RC modulation with $h = 2/7$ and $M = 4$, codeword of 1760 bits.	170
4.6	3RC modulation with $h = 2/7$ and $M = 4$, codeword of 3520 bits.	170
4.7	3RC modulation with $h = 2/7$ and $M = 4$, codeword of 1760 bits.	171
4.8	2RC modulation with $h = 1/5$ and $M = 4$ and code rate $r = 0.88$	172
5.1	Ideal continuous-time filter-bank system.	179
5.2	Practical oversampled discrete-time filter-bank system.	181
5.3	System model decomposed into three blocks: the modulator, the channel and the demodulator.	182
5.4	PSD of the transmitted signal $x(t)$ for DFT-OFDM.	204
5.5	PSD of the transmitted signal $x(t)$ for ODTT-OFDM.	208

List of acronyms

CPM continuous phase modulation

OFDM orthogonal frequency-division multiplexing

AWGN additive white Gaussian noise

PN phase noise

SCCPM serially-concatenated continuous phase modulation

MAP maximum a posteriori probability

APP a posteriori probability

SISO soft-input soft-output

BER bit error rate

ICI inter-carrier interference

CIR channel impulse response

DFT discrete Fourier transform

DCT discrete cosine transform

DST discrete sine transform

DTT discrete trigonometric transform

ASK	amplitude shift keying
CPE	continuous phase encoder
pmf	probability mass function
SISO	soft-input, soft-output
pdf	probability density function
BCJR	Bahl, Cocke, Jelinek, Raviv
FSM	finite-state machine
LLR	log-likelihood ratio
FG	factor graphs
SPA	sum product algorithm
QAM	quadrature-amplitude modulations
PSK	phase-shift keying
DVB-RCS	digital video broadcasting-return channel satellite
IR	information rate
PSD	power spectral density
AR1	1st order auto-regressive
SPC	spectral power concentration
CT	continuous-time
ST	symbol-time
ISI	inter-symbol interference
ACF	autocorrelation function

ML	maximum-likelihood
RC	raised-cosine
i.u.d.	independent and uniformly distributed
SIR	symmetric information rate
i.a.o.d.	independent and asymptotically optimally distributed
FM	frequency modulation
FDM	frequency division multiplexing
CC	convolutional code
REC	rectangular
MM	Mengali and Morelli
G	Green
PC	principal components
MA	Moqvist and Aulin
FC	full-complexity
FE	front end stage
DA	detection algorithm
DT	double-trellis
FT	forward-trellis
NZ	non-zero
MDT	modified double-trellis
SER	symbol error rate

PLL phase-locked loops

GA genie-aided

LTI linear time-invariant

DMT discrete multitone

PS pulseshape

IBI inter-block interference

MSE mean square error

CP cyclic prefix

Foreword

This thesis presents the results of the research activity I have carried out during the three-year period spent as a PhD. student at the Department of Information Engineering of the University of Parma. My work concerns the study of two different modulation schemes, particularly suitable for wireless communications systems, namely continuous phase modulations (CPM) and multicarrier modulation schemes. In detail, I face the major problems in the design of practical systems employing CPM signals (i.e., the large receiver complexity and the sensitiveness to carrier synchronization) and I cope with the problem of spectral efficiency evaluation and maximization. Special focus is devoted to typical satellite channels. Secondly, I address the issue stemming from the most critical drawback of standard orthogonal frequency-division multiplexing (OFDM), i.e., the increased sensitivity to the channel impulse response time variations. Hence, in order to perform reliable digital transmissions over doubly-selective channels, I resort to the derivation of some alternative multicarrier modulation formats, based on low-complexity discrete-time implementation schemes and inspired by the tight connection between multicarrier modulation and the Gabor communication theory.

Chapter 1

Introduction

1.1 Background and Objectives

Wireless communications is the fastest growing sector of the communication industry. In particular, such a growth concerns both wireless terrestrial communications (represented for example by cellular phones, wireless local networks, and wireless sensor networks) and satellite communications (which are probably the major component of the wireless communications infrastructure). However, many technical challenges remain in designing low cost, spectral/energy efficient wireless systems, in order to be competitive with respect to the corresponding wireline counterparts.

Continuous phase modulations (CPMs) and multicarrier schemes, are two modulation schemes which seem to be very suitable for wireless communications. In particular, CPM is a wide class of modulations characterized by continuous phase and constant envelope. Thanks to the constant envelope, CPMs do not require power amplifiers working in the linear region, but low cost amplifiers working in the saturation region can be employed. Moreover, since the phase is continuous and constrained to follow some well structured variations, this class of modulations shows very good bandwidth occupancy. Hence, CPM signals have received an enthusiastic interest by the international

research community in the past, but have so far been employed in a very limited number of applications only. Among these, a CPM signalling scheme, the GMSK (Gaussian Minimum Shift Keying) modulation has been successfully used by the GSM standard. More spectral/energy efficient CPM schemes have found no commercial applications in the past. The reason is that CPMs achieve good spectral/energy efficiency at the price of an increased complexity with respect to linear modulations. In fact, the main drawback of the phase continuity is the introduction of a memory in the modulator and demodulator stages. In particular, more efficient CPM schemes, imply a large complexity in the detection stage. The other major drawback of CPM signals, is the strong sensitivity to carrier synchronization.

In the following work, first of all we analyze the CPM signal from an information theoretic point of view. We employ the method proposed by Arnold and Loeliger to compute the information rate of CPMs over Additive White Gaussian Noise (AWGN) channel and over channels affected by phase noise (PN). In particular, we consider a Wiener PN process and also a more practical phase noise process, typical of some satellite real channels (SATMODE PN). Moreover, since despite the good spectral properties of CPMs, linear modulation offer a much better efficiency, especially at medium and high spectral efficiencies, we propose a spectral efficiency maximization method, where we modify the input probability distribution. We restrict our search to Markov inputs of a given order and we denote the maximum found spectral efficiency, as Markov capacity.

Secondly, in order to overcome one of the main CPM disadvantages, we face the problem of the design of reduced-complexity schemes for CPM signals. In particular, we consider serially-concatenated CPM (SCCPM) schemes, which admit a low-complexity receiver based on the serial concatenation of a CPM modulator with an outer error correcting code. These receivers are particularly interesting since they can achieve practically the same performance of an optimal receiver, in a low complexity way. The overall receiver complexity mainly depends on that of the CPM detector and hence, we focus on the derivation

of reduced-complexity CPM detectors. In particular, we resort to detection algorithms derived from the *maximum a posteriori probability* (MAP) symbol detection strategy, since it allows us to obtain *soft-decision* algorithms, necessary in a SCCPM scheme. We consider two alternative approaches for algorithm derivation: one based on some alternative representations of the CPM signal, and the other based on techniques for trellis complexity reduction. The combination of the two approaches is also investigated. We will show that, for all the considered CPM formats, at least one low-complexity receiver with optimal performance, exists.

Then, we address the other CPM main drawback (i.e., the sensitiveness to inaccurate carrier synchronization) by deriving reduced-complexity soft-input soft-output (SISO) detection algorithms suitable for iterative detection/decoding in the presence of PN. In particular, PN estimation is carried out jointly to the detection stage. In this case, we consider two approaches while deriving the algorithms: a non-Bayesian approach, which does not require any assumption on the statistical properties of the phase noise, since the PN is simply considered as a sequence of unknown parameters to be properly estimated, and the Bayesian approach, which consists of assuming a proper probabilistic model for the PN, and to exploit it for detection algorithm derivation. In particular, we propose Bayesian algorithms assuming both a Wiener PN model and a statistical model we have derived to describe SATMODE PN. We compare all the derived algorithms on channels affected by the two PN and we relate bit error rate (BER) results with information rate results previously obtained.

The other considered scenario is represented by multicarrier schemes, employed in digital transmissions over doubly-selective channels. We start by considering orthogonal frequency-division multiplexing (OFDM), which is an efficient modulation scheme belonging to the wide class of multicarrier modulations. OFDM is today well understood and largely implemented in terrestrial networks as a mean to provide good spectral and energy efficiency over frequency selective channels. Example are both wireline applications (as in the digital subscriber line (DSL) standards) as well as a wide range of wireless

applications, ranging from the digital audio and video broadcasting (DAB-T, DVB-T, DVB-SH, DVB-H) standards, to the local and metropolitan area networks (WLANs and WMANs). The channel characterizing these services is frequency selective, for which OFDM is particularly suitable, since OFDM decomposes linear time-invariant channels into a set of orthogonal, interference-free sub-channels. However, the main OFDM disadvantage is the strong sensitivity to the channel impulse response (CIR) time-variations: in the presence of a rapidly time-varying CIR, the orthogonality between the sub-carriers is destroyed and *inter-carrier interference* (ICI) appears. In such a case, we have two viable solutions:

- to derive receivers with memory, able to cope with the interference or complex equalization techniques;
- the design of modulation formats such as to reduce the interference (rather than to cope with it).

We resort to the second approach, by deriving multicarrier modulation schemes alternative to the OFDM, to reduce the sensitivity to time-variations, in order to employ these modulations on doubly-selective channels

In particular, starting from a general filter-bank system, we propose an oversampled discrete-time system model in order to get a practical implementation of various multicarrier modulation formats in realistic communication systems. We show that all multicarrier modulation formats can be derived from such a discrete-time model, with a general prototype filter (rather than with the classical rectangular filter) and a general time and frequency spacing between the coded symbols. In other words, we apply pulseshape techniques to all schemes, extending the techniques already proposed in the literature. Finally, inspired by the tight connection between multicarrier modulations and the Gabor communication theory, we consider the Wilson base, which is a clever way to design well-localized and orthogonal frames in the windowed Fourier framework, and we derive a novel practical multicarrier modulation scheme,

very promising on doubly selective channels. It is important to remark that for all modulation schemes we will derive, a fast implementation exists.

1.2 Continuous Phase Modulation Signals

The complex envelope of a CPM signal can be written as [1]

$$s(t; \mathbf{x}) = \sqrt{\frac{2E_S}{T}} \exp \left\{ j2\pi h \sum_{n=0}^{N-1} x_n q(t - nT) \right\} \quad (1.1)$$

where E_S is the energy per information symbol, T is the symbol interval, h is the *modulation index*, N is the number of information symbols, $\mathbf{x} = \{x_n\}_{n=0}^{N-1}$ is the information sequence, and $q(t)$ is the *phase response*, constrained to be such that

$$q(t) = \begin{cases} 0 & t \leq 0 \\ \frac{1}{2} & t \geq LT, \end{cases} \quad (1.2)$$

L being the *correlation length*. Several examples of commonly used phase responses are reported in [1].

We assume that the modulation index can be written as $h = r/p$ (where r and p are relatively prime integers), and that the information symbols belong to the M -ary alphabet $\{\pm 1, \pm 3, \dots, \pm(M-1)\}$, M being a power of two. In this case, it can be shown [2] that the CPM signal in the generic time interval $[nT, (n+1)T]$ is given by

$$s(t; \mathbf{x}) = \sqrt{\frac{2E_S}{T}} \exp \left\{ j2\pi h \sum_{l=0}^{L-1} x_{n-l} q(t - (n-l)T) \right\} \exp \left\{ j\pi h \sum_{l=0}^{n-L} x_l \right\} \quad (1.3)$$

and it is completely defined by the actual symbol x_n , the *correlative state*

$$\omega_n \triangleq (x_{n-1}, x_{n-2}, \dots, x_{n-L+1}) = \mathbf{x}_{n-L+1}^{n-1} \quad (1.4)$$

(whereas $\mathbf{x}_{n-L+1}^{n-1}$ is a vector collecting symbols from the time interval $n-L+1$ to $n-1$) and the *phase state* π_n

$$\pi_n \triangleq \left[\pi h \sum_{l=0}^{n-L} x_l \right]_{2\pi} \quad (1.5)$$

(where $[\cdot]_{2\pi}$ denotes the “modulo 2π ” operator) which can be recursively defined as

$$\pi_n = [\pi_{n-1} + \pi h x_{n-L}]_{2\pi} . \quad (1.6)$$

For the initialization of the recursion (1.6), we will always adopt the following conventions

$$\pi_0 = 0 \quad (1.7)$$

$$x_n = 0 \quad \forall n < 0 . \quad (1.8)$$

At any given time epoch n , the correlative state ω_n can assume M^{L-1} different values, while the phase state π_n can assume p different values [2]. In particular, if the numerator r of the modulation index h is even, the phase state can assume the following p values

$$\pi_n \in \left\{ 0, \pi \frac{r}{p}, \pi 2 \frac{r}{p}, \dots, \pi (p-1) \frac{r}{p} \right\} \quad (1.9)$$

while it can assume the following $2p$ possible values when r is odd:

$$\pi_n \in \left\{ 0, \pi \frac{r}{p}, \pi 2 \frac{r}{p}, \dots, \pi 2(p-1) \frac{r}{p} \right\} . \quad (1.10)$$

However, in the latter case, when the time interval n is odd π_n belongs to the alphabet $\mathcal{A}_o = \{2\pi h m, m = 0, 1, \dots, p-1\}$, while, when n is even, it belongs to the alphabet $\mathcal{A}_e = \{j2\pi h(m + 1/2), m = 0, 1, \dots, p-1\}$.

It is also possible to provide a time invariant CPM state representation by considering the following symbol mapping

$$x_n = 2\bar{x}_n - (M-1) \quad (1.11)$$

where $\bar{x}_n$ is an integer representation of the ASK symbols x_n , $\bar{x}_n \in \{0, 1, \dots, M-1\}$. We can also adopt the alternative integer representation ϕ_n for the phase state π_n

$$\phi_n \triangleq \left[\sum_{l=0}^{n-L} \bar{x}_l \right]_p \quad (1.12)$$

where $\phi_n \in \{0, 1, \dots, p-1\}$ independently of the time interval and the recursion (1.6) can be replaced by

$$\phi_n = [\phi_{n-1} + \bar{x}_n]_p. \quad (1.13)$$

The phase state dependence on the time index n appears in the relation between the two phase state definitions:

$$\begin{aligned} \pi_n &= \left[\pi h \sum_{l=0}^{n-L} x_l \right]_{2\pi} \\ &= \left[2\pi h \sum_{l=0}^{n-L} \bar{x}_l - \pi h (M-1)(n-L+1) \right]_{2\pi} \\ &= [2\pi h \phi_n - \pi h (M-1)(n-L+1)]_{2\pi} \end{aligned} \quad (1.14)$$

where we have substituted definitions (1.12) and (1.11) in (1.5).

We now reconsider to the expression (1.3) for the CPM signal in the generic time interval $[nT, (n+1)T]$; we note that it can be expressed as

$$s(t; \mathbf{x}) = \tilde{s}(t - nT; x_n, \sigma_n) \quad (1.15)$$

where

$$\tilde{s}(t; x_n, \sigma_n) \triangleq \bar{s}(t; x_n, \omega_n) e^{j\pi_n} \quad \forall t \in [0, T) \quad (1.16)$$

and

$$\bar{s}(t; x_n, \omega_n) \triangleq \sqrt{\frac{2E_S}{T}} \exp \left\{ j2\pi h \sum_{l=0}^L x_{n-l} q(t + lT) \right\} \quad (1.17)$$

and

$$\sigma_n \triangleq (\omega_n, \pi_n). \quad (1.18)$$

Hence, we can consider the CPM waveform (1.16) as the output of a finite-state machine with input x_n and state σ_n defined in (1.18), which can assume pM^{L-1} values. In other words, as stated in [2], the CPM modulator can be decomposed as a continuous-phase encoder (CPE) followed by a memoryless modulator, so that “encoding” operation can be studied independently of the

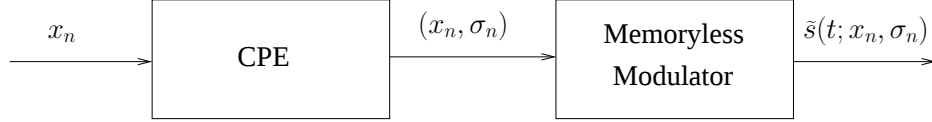


Figure 1.1: Block diagram of CPM modulator decomposed into a continuous-phase encoder and a memoryless modulator

modulation. Moreover, thanks to the assumed state definition, the CPE results time-invariant and linear modulo- p and hence it can be considered as a sort of convolutional encoder. Such a scheme is represented in Fig. 1.1: the symbols x_n are passed through a finite-state machine (the linear CPE), and then a time variant memoryless modulator generates the CPM waveforms (1.16) as a function of (x_n, σ_n) . Finally, such a decomposition is very useful to show how a CPM modulator is especially suitable to be concatenated with an outer convolutional encoder. For this reason, the scientific literature has reserved an increasing interest in SCCPM schemes described in [3–5], and based on low complexity iterative joint detection/decoding (see Section 1.3.3).

1.3 General Frameworks

In the following Section we describe some general frameworks we will employ in the remainder of this work.

1.3.1 MAP Symbol Detection Strategy and BCJR Algorithm

Given a sequence of transmitted symbols x_n collected into vector $\mathbf{x}$, where $\mathbf{x} = (x_0, x_1, \dots, x_{N-1})$ and a channel with memory, we denote by the vector $\mathbf{y}$ the sufficient statistics of the received signal $r(t)$, extracted by the receiver. In particular, the n -th element of the vector $\mathbf{y}$ can be a vector, denoted in the following by $\mathbf{y}_n$, since in general, at each time epoch n the number of sufficient statistics can be greater than one. Thus, the MAP symbol detection strategy

minimizing the average symbol error probability is

$$\hat{x}_n = \operatorname{argmax}_{x_n} P(x_n|\mathbf{y}) \quad (1.19)$$

while MAP sequence detection strategy is

$$\hat{\mathbf{x}} = \operatorname{argmax}_{\mathbf{x}} P(\mathbf{x}|\mathbf{y}) \quad (1.20)$$

where we denote by the capitol letter $P(\cdot)$ a probability mass function (pmf). In the following work, we always employ MAP symbol detection, since it provides soft-output decisions, while MAP sequence doesn't. In detail, MAP symbol detection strategy computes, as a by-product, the a posteriori probabilities $P(x_n|\mathbf{y})$ which can be considered as reliability estimates on the chosen symbols $\hat{x}_n$; these estimates allow us to derive *Soft-Input, Soft-Output* (SISO) detection (or decoding) algorithms, necessary in order to implement iterative joint detection/decoding schemes.

In particular, by employing the Bayes rule, we can express MAP symbol strategy in (1.19) as

$$\hat{x}_n = \operatorname{argmax}_{x_n} p(\mathbf{y}|x_n)P(x_n) \quad (1.21)$$

where $P(x_n)$ is the a priori probability of the symbol x_n and we denote by $p(\cdot)$ a probability density function (pdf). Thus, in order to satisfy the proposed maximization algorithm, we need to compute the pdf $p(\mathbf{y}|x_n)$. If we consider a channel with memory, which can be described as a finite-state machine (FSM), whose state is denoted by σ_n , we can solve the MAP symbol problem by the *Bahl, Cocke, Jelinek, Raviv* (BCJR) algorithm [6], based on a probabilistic derivation. In particular, $p(\mathbf{y}|x_n)$ expression is given by

$$p(\mathbf{y}|x_n) = \sum_{\sigma_n} \eta_{f,n}(\sigma_n) \eta_{b,n+1}(\sigma_{n+1}) F_n(x_n, \sigma_n) \quad (1.22)$$

where

- the function $F_n(x_n, \sigma_n)$ is

$$F_n(x_n, \sigma_n) \triangleq p(\mathbf{y}_n|x_n, \sigma_n) \quad (1.23)$$

where $\mathbf{y}_n$ is the vector collecting all the sufficient statistics at the time epoch n .

- $\eta_{f,n}(\sigma_n)$ is called *forward metric* and is defined as

$$\eta_{f,n}(\sigma_n) \triangleq p(\mathbf{y}_0^{n-1} | \sigma_n) P(x_n) \quad (1.24)$$

where we denote by $\mathbf{y}_{n_1}^{n_2}$ the vector collecting all statistics $\mathbf{y}_n$ from $n = n_1$ to $n = n_2$.

- $\eta_{b,n+1}(\sigma_{n+1})$ is called *backward metric* and has expression

$$\eta_{b,n+1}(\sigma_{n+1}) \triangleq p(\mathbf{y}_{n+1}^{N-1} | \sigma_{n+1}). \quad (1.25)$$

Forward and backward metrics can be recursively computed through the following forward and backward recursions

$$\eta_{f,n+1}(\sigma_{n+1}) = \sum_{x_n} \sum_{\sigma_n} \eta_{f,n}(\sigma_n) F_n(x_n, \sigma_n) P(x_n) \quad (1.26)$$

$$\eta_{b,n}(\sigma_n) = \sum_{x_n} \sum_{\sigma_{n+1}} \eta_{b,n+1}(\sigma_{n+1}) F_n(x_n, \sigma_n) P(x_n). \quad (1.27)$$

Hence the BCJR algorithm works in the following way:

- first, forward and backward metrics $\eta_{f,n}(\sigma_n)$ and $\eta_{b,n}(\sigma_n)$ are computed by means of (1.24) and (1.25), for each time epoch n and for each state value σ_n ;
- then, the pdf $p(\mathbf{y} | x_n)$ is derived by (1.22) exploiting $\eta_{f,n}(\sigma_n)$, $\eta_{b,n}(\sigma_n)$ and $F_n(x_n, \sigma_n)$ values;
- finally, the MAP strategy (1.21) can be implemented by computing the a posteriori probabilities $P(x_n | \mathbf{y})$.

We refer to [6] for more detail on the BCJR algorithm derivation.

Typically, when symbols $\{x_n\}$ are generated from an M -ary alphabet, we choose the set $\{\ell_{a,n}\}_a$ of $M-1$ logarithmic ratios of the a posteriori probabilities

$P(x_n|\mathbf{y})$, as the reliability estimates of the decision on the symbol x_n . $\ell_{a,n}$ is defined as

$$\ell_{a,n} = \log \frac{P(x_n = a|\mathbf{y})}{P(x_n = 0|\mathbf{y})} \quad (1.28)$$

where $a \in \{1, 2, \dots, M-1\}$. $\ell_{a,n}$ is denoted as *log-likelihood ratio* (LLR).

1.3.2 Factor Graphs and Sum Product Algorithm

The BCJR algorithm described in Section 1.3.1, solves the MAP symbol detection problem. It can be alternatively derived by means of the Factor Graphs (FG) and the Sum Product Algorithm (SPA), presented in [7]. These tools are particularly suited to find the way a complicated global function of many variables factors as a product of “local” functions, each of which depends on a subset of the variables. This factorization can be visualized with a FG, which is a bipartite graph that indicates which variable is argument of which local function. The SPA works on the FG and computes the marginal functions derived from the global function.

Let $\mathbf{x} = (x_1, x_2, \dots, x_n)$ be a collection of variables, where x_i takes on values on some (usually finite) domain $\mathcal{A}_i$, and let $f(\mathbf{x})$ a multivariate function. Suppose that $f(\mathbf{x})$ factors into a product of several local functions f_j , each having a subset $\mathbf{x}_j$ of $\mathbf{x}$, as argument:

$$f(\mathbf{x}) = \prod_{j \in J} f_j(\mathbf{x}_j) \quad (1.29)$$

where J is a discrete index set. A factor graph is a bipartite graph which has a variable node for each variable x_i , a factor node for each function f_j , and an edge connecting variable node x_i to function node f_j if and only if x_i is an argument of f_j . The SPA is defined by the computation rules at variable and at factor nodes and by a suitable node activation schedule. Denoting by $\mu_{x_i \rightarrow f_j}(x_i)$ a message sent from the variable node x_i to the factor node f_j , by $\mu_{f_j \rightarrow x_i}(x_i)$ a message in the opposite direction, and by $\mathcal{B}_i$ the set of functions f_j having x_i as argument, the message computations performed at variable

and factor nodes are, respectively [7]

$$\mu_{x_i \rightarrow f_j}(x_i) = \prod_{h \in \mathcal{B}_i \setminus f_j} \mu_{h \rightarrow x_i}(x_i) \quad (1.30)$$

$$\mu_{f_j \rightarrow x_i}(x_i) = \sum_{\sim \{x_i\}} \left[f_j(\{y \in \mathcal{B}_j\}) \prod_{y \in \mathcal{B}_j \setminus x_i} \mu_{y \rightarrow f_j}(y) \right] \quad (1.31)$$

where we indicate by $\sum_{\sim \{x_i\}}$ the *summary* operator, i.e., a sum over all variables excluding x_i .

Thus, we can factorize the pmf $P(\mathbf{x}|\mathbf{y})$ in order to find, through the SPA, the marginal a posteriori probabilities $P(x_i|\mathbf{y})$, required by the MAP symbol strategy (1.19). If the FG has cycles, the SPA is inherently iterative and convergence to the exact marginal pmf is not guaranteed. Nevertheless, for many relevant problems characterized by FGs with cycle, the SPA was found to provide very good results and therefore it represents a viable solution to the approximated marginalization of multivariate pmfs when exact calculation is not feasible because of complexity.

Finally, we define the *message-passing schedule* in the SPA as the specification of the order in which messages are updated. In general, especially for graphs with cycles, the so-called *flooding schedule* is adopted [8]: in each iteration, all variable nodes and subsequently all factor nodes, pass new messages to their neighbors.

1.3.3 Iterative Joint Detection/Decoding Schemes

When we consider a communication system characterized by an error correcting code together with a channel with memory, the cardinality of the overall system state can be very large, and thus, optimal MAP symbol detection strategy at the receiver becomes unfeasible. In these cases, we can resort to a suboptimal iterative joint detection/decoding scheme, which exhibits computational complexity absolutely lower than the complexity of the optimal scheme, but a performance which tends to the optimal one (as verified by numerical

results) [9]. In particular, here we describe the operations of a serially concatenated scheme, which is the scheme adopted in Chapters 3 and 4 for detection of CPM signal in the presence of an outer error correcting code.

In an iterative concatenated joint detection/decoding scheme, each component block (i.e., the detector and the decoder) works separately, by implementing the MAP symbol strategy optimal for the single block, assuming that no other memory sources are present in the system. They employ a detection (decoding) algorithm based on the MAP symbol rule, which provides reliability estimates on the algorithm decisions (Section 1.3.1), since we need soft-output decisions in order to concatenate the two blocks. In general, an iterative concatenated scheme is based on the following basic concept: each component block exploits the suggestion provided by the other component block, in order to derive decisions which becomes more reliable with the iteration. In detail, a serially concatenated scheme works as follow. First of all, the detector performs an instance of the detection algorithm, operating on the channel sufficient statistics $\mathbf{y}$. Then, the soft-decisions it produces on each symbol x_n , are forwarded to the decoder, which employs the detector a posteriori probabilities, as a priori probabilities on $\{x_n\}$ while performing decoding. Thus, it also produces soft-decisions on the x_n symbols, which are in turn passed to the detector. The detector exploits such reliability estimations as a priori probabilities on $\{x_n\}$ and it starts a new iteration of the serially concatenated scheme. The joint detection/decoding strategy continues for a fixed number of cycles; then, hard final decisions on the symbols $\{x_n\}$ are derived.

In order to accelerate the convergence of the iterative detection/decoding process, each component block must receive as input an information which is not self-produced. With this aim, in [10, 11] the concept of *extrinsic information* is introduced, which identifies the reliability information produced by a component block which does not depend on the information it received as input. In detail, if we denote by $\ell_{a,n}^{\text{out}}$ the LLRs defined in (1.28), produced by a block and representing the reliability measure of a MAP symbol algorithm on the decision on the symbol x_n , the extrinsic information $\ell_{a,n}^{\text{e,out}}$ generated by

such block, is given by

$$\ell_{a,n}^{\text{e,out}} = \ell_{a,n}^{\text{out}} - \ell_{a,n}^{\text{e,in}} . \quad (1.32)$$

The FG/SPA tool intrinsically propagates extrinsic information, as described by (1.30)-(1.31).

Chapter 2

Capacity Evaluation for CPM signals

2.1 Introduction

Continuous phase modulations (CPMs) form a class of constant envelope signaling formats that are efficient in power and bandwidth [1]. In order to compare CPMs against linear modulations, e.g., phase-shift keying (PSK) and quadrature-amplitude modulations (QAMs), as well as to optimize the various parameters which describe a CPM signal, namely phase response, alphabet size, and modulation index, information theoretic arguments could be used. In particular, the availability of expressions for the channel capacity, in terms of bits per channel use, and bandwidth occupancy of the signal, would allow to evaluate the amount of b/s carried per band unit (Hz), which is an important comparison criterion for system designer. Unfortunately, neither closed-form results for the capacity of a CPM signal nor simple expressions for its bandwidth exist.

In [12] and [13] an analytical study of the information transfer over additive white Gaussian noise (AWGN) channels for envelope constrained waveforms under a fixed bandwidth definition is described; however the results do not

seem be immediately applicable to the CPMs. First results on CPMs capacity are reported in [14], where they are presented to justify the choose of a certain outer code in a serially-concatenated CPM (SCCPM) scheme. The authors do not describe the method by which CPM capacity is computed and moreover, the results they carry out do not match with results we will obtain in the present work. In [15] multi-channel capacity results are applied to the decomposition [16] of CPM signal; unfortunately that method provides bounds too loose on CPM capacity to be considered of practical interest.

Finally in [17,18] and [19] CPM capacity is computed by using the method by Arnold and Loeliger in 2001 [20] and in 2006 [21]. By this technique, the information rate (IR) over a channel with memory can be computed by the forward metric of a BCJR algorithm (explained in Section 1.3.1) applied at the output of such a channel. This work was presented in the context of linear modulations over channels with memory, but can be transposed to all channels with memory. Hence, representing CPM signals as a suitable finite-state, finite-dimensional Markov process (as done in Section 1.2), it is possible to derive the information rate of the channel composed by the concatenation of a CPM modulator and an AWGN channel. Moreover, in [19] the information rate of a CPM signal is computed in the presence of a phase noise (PN) process, modeled as a discrete time Wiener random process, which is a commonly used model in telecommunication schemes.

The remainder of this chapter is organized as follows. First of all, we provide a more exhaustive description of the algorithm proposed by Arnold and Loeliger and we exploit it to derive the IR capacity of a CPM signal over an AWGN channel and over an AWGN channel affected by phase noise, modeled as a discrete time Wiener process. Then, we resort to a phase noise (PN) model of practical interest, i.e., the SATMODE phase noise model, which impairs satellite communications characterized by low-cost transmitter and receiver. SATMODE PN is a model usually employed to test the performance of DVB-RCS (*Digital Video Broadcasting-Return Channel Satellite*) systems. In particular, inspecting the phase noise power spectral density (PSD) [22], we see that the

discrete-time phase process can be well approximated by the sum of two 1st order auto-regressive (AR1) processes, and we derive CPM's IR in such a scenario. Finally, we propose a novel iterative method, similar to the generalized Blahut-Arimoto algorithm recently proposed by Kavčič [23], to evaluate the Markov capacity of a CPM signal over an AWGN channel. One of the novelty of our approach is that we maximize, with respect to the input distribution, the spectral efficiency in bps/Hz rather than the mutual information in bits per channel use. We address this problem by taking into account the bandwidth occupancy of the CPM signal by means of the (properly modified) Carson's rule bandwidth definition, and solving a linearly constrained non-linear optimization problem. The results show that Markov capacity obtained with the proposed input optimization algorithm strongly outperforms the capacity for independent and uniformly distributed inputs. In the numerical results, we also consider a proper bandwidth definition related to the spectral power concentration (SPC), although in this case the optimization can be performed by resorting to a brute-force approach only.

2.2 Information Rate for Channel with Memory

2.2.1 Information Rate Definition

Mutual information rate $I(X; Y)$ quantifies the amount of information that can be transmitted over a channel with input process X and output process Y and it is expressed in bits per channel use. In the following work we will focus on the case where both X and Y are discrete-time stationary sequences (in general not of the same length), denoted with $\mathbf{x}$ and $\mathbf{y}$, respectively. From information theory [24, 25], we know that for each channel $I(\mathbf{x}; \mathbf{y})$ can be expressed as

$$I(\mathbf{x}; \mathbf{y}) = h(\mathbf{x}) - h(\mathbf{x}|\mathbf{y}) \quad \left[\frac{\text{b/s}}{\text{Hz}} \right] \quad (2.1)$$

where $h(\mathbf{x})$ is the differential entropy rate of a source generating the random discrete-time stationary process $\mathbf{x}$

$$h(\mathbf{x}) \triangleq -\mathbb{E}[\log_2 p(\mathbf{x})] = \int_{-\infty}^{+\infty} p(\mathbf{x}) \log_2 \frac{1}{p(\mathbf{x})} d\mathbf{x} \quad (2.2)$$

and $h(\mathbf{x}|\mathbf{y})$ is the conditional differential entropy rate of the channel input process $\mathbf{x}$ given the channel output $\mathbf{y}$

$$h(\mathbf{x}|\mathbf{y}) \triangleq -\mathbb{E}[\log_2 p(\mathbf{x}|\mathbf{y})] = \int \int_{-\infty}^{+\infty} p(\mathbf{x}, \mathbf{y}) \log_2 \frac{1}{p(\mathbf{x}|\mathbf{y})} d\mathbf{x} d\mathbf{y} \quad (2.3)$$

which depends only on the channel characteristics (i.e., the transition probabilities $p(\mathbf{x}|\mathbf{y})$). It can be shown that (2.1) is equivalent to

$$I(\mathbf{x}; \mathbf{y}) = h(\mathbf{y}) - h(\mathbf{y}|\mathbf{x}) \quad \left[\frac{\text{b/s}}{\text{Hz}} \right]. \quad (2.4)$$

2.2.2 Arnold and Loeliger Method

In [20,21,26], it is described a method to compute the mutual information of a finite-state hidden Markov model employing the forward recursion of the well-known BCJR algorithm. Such a method can be extended to all channel models with an infinite number of states (for example an AWGN channel affected by PN) finding an auxiliary finite-state channel which well approximates the real channel; hence the algorithm allows to compute a lower limit of the actual information rate, which tends to such value when the number of states of the auxiliary channel grows towards infinity.

We now present the algorithm in [20]. Given a certain channel input sequence $\mathbf{x}_0^N \triangleq (x_0, x_1, \dots, x_N)$ and the corresponding output sequence $\mathbf{y}_0^N \triangleq (y_0, y_1, \dots, y_N)$, the computation of the differential entropy rate $h(\mathbf{y})$ and the conditional differential entropy rate $h(\mathbf{y}|\mathbf{x})$ in a simulation can be carried out thanks to the Shannon-McMillian-Breimann theorem [25,27] which ensures the convergence with probability one of

$$h(\mathbf{y}) = - \lim_{N \rightarrow +\infty} \frac{1}{N} \mathbb{E} [\log_2 p(\mathbf{y}_0^N)] \quad (2.5)$$

$$h(\mathbf{y}|\mathbf{x}) = - \lim_{N \rightarrow +\infty} \frac{1}{N} \mathbb{E} [\log_2 p(\mathbf{y}_0^N | \mathbf{x}_0^N)] \quad (2.6)$$

if $\mathbf{x}_0^N$ and $\mathbf{y}_0^N$ are stationary ergodic finite-state hidden Markov processes. By replacing (2.5) and (2.6) in (2.4), we get

$$I(\mathbf{x}; \mathbf{y}) = \lim_{N \rightarrow +\infty} \frac{1}{N} \mathbb{E} \left[\log_2 \frac{p(\mathbf{y}_0^N | \mathbf{x}_0^N)}{p(\mathbf{y}_0^N)} \right]. \quad (2.7)$$

Hence, from (2.7) it is clear that in order to compute the information rate it is sufficient to obtain the values of the probability density functions $p(\mathbf{y}_0^N)$ and $p(\mathbf{y}_0^N | \mathbf{x}_0^N)$. Such values can be effectively computed by the forward recursion of the BCJR algorithm, employed to implement a maximum a posteriori (MAP) symbol detection strategy. In particular, by defining σ_n the channel state, the forward message $\eta_{f,n}(\sigma_n)$ is obtained as

$$\begin{aligned} \eta_{f,n+1}(\sigma_{n+1}) &= p(y_n, y_{n-1}, \dots, y_1, \sigma_{n+1}) \\ &= \sum_{\sigma_n} p(\mathbf{y}_0^n, \sigma_{n+1}, \sigma_n) \\ &= \sum_{\sigma_n} p(y_n | \mathbf{y}_0^{n-1}, \sigma_{n+1}, \sigma_n) p(\mathbf{y}_0^{n-1}, \sigma_{n+1}, \sigma_n) \\ &= \sum_{\sigma_n} p(y_n | \mathbf{y}_0^{n-1}, \sigma_{n+1}, \sigma_n) p(\sigma_{n+1} | \sigma_n) \eta_{f,n}(\sigma_n) \end{aligned} \quad (2.8)$$

as shown in [28]. At the N -th time symbol interval, $p(\mathbf{y}_0^N)$ is obtained as the sum of the state metrics

$$p(\mathbf{y}_0^N) = \sum_{\sigma_{N+1}} \eta_{f,N+1}(\sigma_{N+1}). \quad (2.9)$$

Finally, for evaluating the mean value of $\log_2 p(\mathbf{y}_0^N)$, we need to repeat the simulation K times and, by denoting with ρ_n the $p(\mathbf{y}_0^N)$ measure at the n -th simulation, we find

$$\mathbb{E}[\log_2 p(\mathbf{y}_0^N)] = \lim_{K \rightarrow +\infty} \frac{1}{K} \sum_{k=1}^K \log_2(\rho_k). \quad (2.10)$$

2.3 IR of CPMs over AWGN

The method described in Section 2.2.2 can be applied to compute the CPM information rate for independent and uniformly distributed symbol input, over AWGN channel. In particular, in such a case, the complex envelope of the received signal can be written as:

$$r(t) = s(t; \mathbf{x}) + w(t) \quad (2.11)$$

where $s(t; \mathbf{x})$ is the CPM signal (1.1) and $w(t)$ is a complex-valued AWGN process with independent components, each with two-sided power spectral density N_0 . The value of N_0 is assumed perfectly known at the receiver¹ and we also assume independent and uniformly distributed symbols $\mathbf{x}$. Since the channel memory is concentrated in the CPM modulator which is described as a FSM model in Section 1.2, the expression for σ_n is given by (1.18) and the forward recursion we need to compute in the Arnold and Loeliger method, is the forward recursion $\eta_{f,n}(\sigma_n)$ (2.17) of the MAP symbol detection algorithm we will describe in Section 2.3.1.

2.3.1 Optimal MAP Symbol Detection

An overview of the optimal algorithm for MAP symbol detection, which turns out to be an instance of the well-known BCJR algorithm [6], is given in the following.

At each time epoch n and for each trial symbol x_n and trial state σ_n , let us define

$$\begin{aligned} y_n(x_n, \sigma_n) &\triangleq \int_{nT}^{(n+1)T} r(t) \tilde{s}^*(t - nT; x_n, \sigma_n) dt \\ &= e^{-j\pi n} \int_{nT}^{(n+1)T} r(t) \bar{s}^*(t - nT; x_n, \omega_n) dt \end{aligned} \quad (2.12)$$

¹The constant amplitude of a CPM signal makes the detection algorithm very robust to a possibly inaccurate estimate of the value of N_0 [1]. Hence, the hypothesis of a perfect knowledge of N_0 is not critical.

where $\tilde{s}(t; x_n, \sigma_n)$ is the CPM waveform in the interval $[nT, (n+1)T]$ defined in (1.16) and corresponding to the trial values of x_n and σ_n , while $\bar{s}^*(t; x_n, \omega_n)$ (defined in (1.17)) is the CPM waveform depending on just actual symbol x_n and correlative state ω_n . In practice, samples $\{y_n(x_n, \sigma_n)\}$ are obtained by sampling, at symbol rate, the output of a bank of filters matched to all waveforms which can occur in a symbol interval. However, note that matched filters are M^L instead of pM^L because the CPM phase state is constant over a symbol interval $[nT, (n+1)T]$; hence we need a matched filter for each of M^L combinations of actual symbol x_n and correlative state ω_n . The vector $\mathbf{y}$ collecting these samples gives a set of sufficient statistics for MAP symbol detection [1].

For each symbol interval n , the MAP symbol detection strategy provides the symbol x_n which satisfy the following condition (Section 1.3.1)

$$x_n = \underset{\hat{x}_n}{\operatorname{argmax}} P(\hat{x}_n | \mathbf{y}) . \quad (2.13)$$

The probabilities $P(\hat{x}_n | \mathbf{y})$ can be obtained through the well-known BCJR algorithm (see Section 1.3.1). This algorithm can be derived in a probabilistic way, as done in [6], but also thanks to FG and SPA, which allow us to marginalize the probability mass function (pmf) $P(\mathbf{x}, \boldsymbol{\sigma} | \mathbf{y})$ with respect to each variables x_n , where

$$\begin{aligned} \mathbf{x} &= (x_0, x_1, \dots, x_{N-1}) \\ \boldsymbol{\sigma} &= (\sigma_0, \sigma_1, \dots, \sigma_N) \end{aligned}$$

and $\mathbf{y}$ is the vector collecting all the sufficient statistics $y_n(x_n, \sigma_n)$ in (2.12), with n from 0 to N . In particular, we factorize $P(\mathbf{x}, \boldsymbol{\sigma} | \mathbf{y})$ as ²

$$\begin{aligned} P(\mathbf{x}, \boldsymbol{\sigma} | \mathbf{y}) &\propto p(\mathbf{y} | \mathbf{x}, \boldsymbol{\sigma}) P(\boldsymbol{\sigma} | \mathbf{x}) P(\mathbf{x}) \\ &= P(\sigma_0) \prod_{n=0}^{N-1} F_n(x_n, \sigma_n) T(x_n, \sigma_n, \sigma_{n+1}) P(x_n) \end{aligned} \quad (2.14)$$

²The proportionality symbol $\propto$ is used when two quantities differ for a positive multiplicative factor, irrelevant for the detection process.

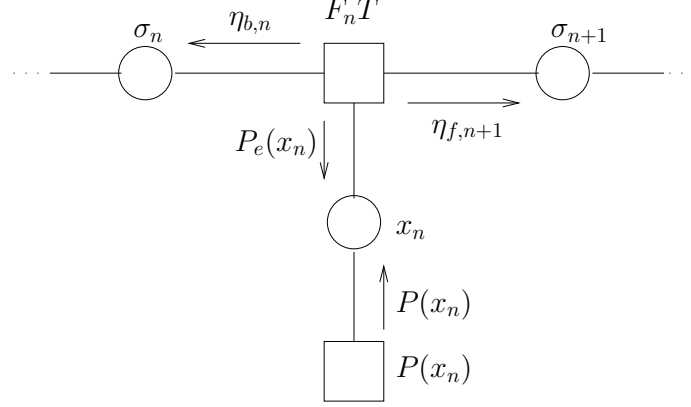


Figure 2.1: Factor graph of the optimal MAP symbol detector in the n -th time interval.

where $T(x_n, \sigma_n, \sigma_{n+1})$ is the trellis indicator function, equal to one if x_n , σ_n , and σ_{n+1} satisfy the trellis constraints and to zero otherwise, and $F_n(x_n, \sigma_n)$ is the branch metric function, defined as

$$F_n(x_n, \sigma_n) = \exp \left[\frac{1}{N_0} \text{Re}\{y_n(x_n, \sigma_n)\} \right] \quad (2.15)$$

and where we have assumed independent information symbols

$$P(\mathbf{x}) = \prod_{n=0}^{N-1} P(x_n). \quad (2.16)$$

In (2.14) we have introduced the following notation: $P(\cdot)$ indicates a pmf, while $p(\cdot)$ indicates a probability density function (pdf).

Hence, we can draw a factor graph whose section at the n -th time interval is represented in Fig 2.1, and by applying the SPA we derive the BCJR algorithm, characterized by the following forward and backward recursions

$$\eta_{f,n+1}(\sigma_{n+1}) = \sum_{\alpha_n} \sum_{\sigma_n} P(x_n) T(x_n, \sigma_n, \sigma_{n+1}) F_n(x_n, \sigma_n) \eta_{f,n}(\sigma_n) \quad (2.17)$$

$$\eta_{b,n}(\sigma_n) = \sum_{\alpha_n} \sum_{\sigma_{n+1}} P(x_n) T(x_n, \sigma_n, \sigma_{n+1}) F_n(x_n, \sigma_n) \eta_{b,n+1}(\sigma_{n+1}) \quad (2.18)$$

The state metrics $\eta_{f,n}(\sigma_n)$ and $\eta_{b,n}(\sigma_n)$ have the following probabilistic meanings

$$\eta_{f,n}(\sigma_n) \propto P(\sigma_n | \mathbf{y}_0^{n-1}) \quad (2.19)$$

$$\eta_{b,n}(\sigma_n) \propto p(\mathbf{y}_n^{N-1} | \sigma_n) \quad (2.20)$$

where $\mathbf{y}_{n_1}^{n_2}$ is the vector collecting all samples $\{y_n(\alpha_n, \sigma_n)\}$ from time epoch n_1 to time epoch n_2 . Hence, the metrics $\eta_{f,0}(\sigma_0)$ and $\eta_{b,N}(\sigma_N)$ are initialized according to the available information on the actual first state $\bar{\sigma}_0$ and last state $\bar{\sigma}_N$. When $\bar{\sigma}_0$ (respectively, $\bar{\sigma}_N$) is known at the receiver, $\eta_{f,0}(\bar{\sigma}_0)$ (respectively, $\eta_{b,N}(\bar{\sigma}_N)$) can be set to one while the other metrics $\eta_{f,0}$ (respectively, $\eta_{b,N}$) are initialized to zero. When $\bar{\sigma}_0$ (respectively, $\bar{\sigma}_N$) is unknown at the receiver, all metrics $\eta_{f,0}$ (respectively, $\eta_{b,N}$) are initialized to the same value—this value is irrelevant for the detection process, provided that it is positive. Finally, once performed both recursions, the extrinsic information $P_e(x_n)$ on x_n (defined as $P_e(x_n) = P(x_n | \mathbf{y}) / P(x_n)$) is computed as follows

$$P_e(x_n) \propto \sum_{\sigma_n} \sum_{\sigma_{n+1}} T(x_n, \sigma_n, \sigma_{n+1}) F_n(x_n, \sigma_n) \eta_{f,n}(\sigma_n) \eta_{b,n+1}(\sigma_{n+1}) \quad (2.21)$$

in the so-called completion stage.

The number of trellis states can be considered as a measure of the complexity of the BCJR algorithm. Hence, for a fixed value of the frame length N , the complexity results proportional to pM^{L-1} , making the implementation of the BCJR algorithm impractical when large values of p , M , or L are considered. On the other hand, large values of these parameters are of great interest since they often provide a better spectral/power efficiency [1]. The design of reduced-complexity detection algorithms for CPM formats characterized by a large number of states is addressed in Chapter 3.

2.4 IR of CPMs over Channel Affected by Phase Noise

We now consider the transmission of a CPM signal over a channel affected by phase noise, for example a typical satellite channel. We represent the phase noise as a random phase rotation before the addition of independent AWGN. This is the case of a lot of channels of practical interest (as the forward and return links in satellite communications). We assume that the difference between the phases of the transmitter and receiver oscillators is given by $[\Delta f t + \theta(t)]$ where Δf is the carrier frequency offset and will be considered null in the following work, while $\theta(t)$ is the PN, assumed a Gaussian zero-mean stationary random process. Hence, the complex envelope of the received signal can be modeled as

$$r(t) = s(t; \mathbf{x})e^{j\theta(t)} + w(t). \quad (2.22)$$

In the following we will consider two models for the random process $\theta(t)$: the Wiener PN, which is commonly assumed in literature and a more realistic model, i.e., the SATMODE phase noise, which impairs satellite communications those characterized by low-cost equipment and whose PSD mask is provided in the report [22]. For each model, we will provide a technique to evaluate the ultimate performance limits in order to quantify the information rate loss with respect to the CPM transmission over a coherent AWGN channel.

2.4.1 Wiener Phase Noise Model

We model the PN $\theta(t)$ as a time-continuous Wiener process [29, 30], which is a random process defined as having zero-mean normally distributed phase increments $[\theta(t_2) - \theta(t_1)]$ over any interval $[t_1, t_2]$ and independent phase increments over disjoint time intervals. We assume an incremental variance over a signaling interval equal to σ_Δ^2 and also assume that the channel phase $\theta(t)$ is slowly varying such that it can be considered constant over a symbol time

T . In other words we assume that the output of the matched filters is

$$\begin{aligned}
& \int_{nT}^{(n+1)T} r(t) \tilde{s}^*(t - nT; x_n, \sigma_n) e^{-j\theta(t)} dt \\
&= e^{-j\theta_n} e^{-j\pi_n} \int_{nT}^{(n+1)T} r(t) \bar{s}^*(t - nT; x_n, \omega_n) e^{j[\theta_n - \theta(t)]} dt \\
&\simeq e^{-j\theta_n} \int_{nT}^{(n+1)T} r(t) \bar{s}^*(t - nT; x_n, \omega_n) dt = y_n(x_n, \sigma_n) e^{-j\theta_n} \quad (2.23)
\end{aligned}$$

where $y_n(x_n, \sigma_n)$ defined in (2.12) is the output of a bank of matched filters for an AWGN channel, $\theta_n \triangleq \theta(nT)$ and we assume $\theta_n - \theta(t) \simeq 0$ in $[nT, (n+1)T]$, for the slowly varying assumption. From (2.23), only the samples of $\theta(t)$ at discrete-time nT are significant. These samples satisfy the discrete-time Wiener model:

$$\theta_{n+1} = \theta_n + \Delta_n \quad (2.24)$$

where $\{\Delta_n\}$ are real independent and identically distributed Gaussian random variables with zero mean and standard deviation σ_Δ , and θ_0 is assumed uniformly distributed in $[0, 2\pi)$. It is worth noting that, having assumed zero-mean increments Δ_n , we practically assumed a perfect carrier frequency synchronization (i.e., $\Delta f = 0$ as already stated in Section 2.4).

The advantage of having a discrete time PN model, applied before the additive thermal noise, is that we can define an overall system state, composed by the CPM modulator state and by the channel state, represented by the PN sample θ_n . However, since θ_n may assume with continuity all values between 0 and 2π , the cardinality of the overall system state is infinite. Hence, in order to apply the Arnold and Loeliger algorithm, at the receiver we evaluate the forward recursion $\eta_{f,n}(\sigma_n)$ of a BCJR algorithm matched to an auxiliary finite-state channel obtained by discretizing the phase θ_n . Such an algorithm is defined as the DP-BCJR (i.e., *discretized-phase* BCJR) and it is described in Section 2.4.3. In particular, computing the forward recursion of DP-BCJR, we evaluate the information rate of the auxiliary channel instead of the information rate of the real channel with continuous PN. In other words there is a mismatch

between the channel and the receiver and what we obtain is a lower limit on the real IR, representing the maximum achievable bits per channel use we can transmit over the channel, when the DP-BCJR detector is employed. However, increasing the number of quantization levels towards infinity, the auxiliary finite-state channel tends to the real one and hence, the IR value we compute tends to the IR of the real channel.

2.4.2 SATMODE Phase Noise Model

We now consider a more realistic PN model, i.e., the SATMODE phase noise which affects the satellite communication systems with low-cost equipment [22]. Its power spectral density is known and it is represented in Table. 2.1. Even if in principle that mask refers to the phasor $e^{j\theta(t)}$, it has been assumed that it approximately corresponds to the PSD of the phase process $\theta(t)$. In order to

Frequency [Hz]	PSD [dB/Hz]
10^1	-23
10^2	-48
10^3	-68
10^4	-68
10^5	-80
10^6	-100

Table 2.1: SATMODE PSD of the continuous-time phase process $\theta(t)$.

compute the IR for CPM modulation schemes over SATMODE channel and to derive some matched detection algorithms, we need to derive a finite-order statistical description of the PN. Since we want to deal with a discrete-time process, we sample the phase process with a sampling period $T_S = 10^{-7}$ in order to avoid aliasing; hence we obtain a discrete-time process with the same

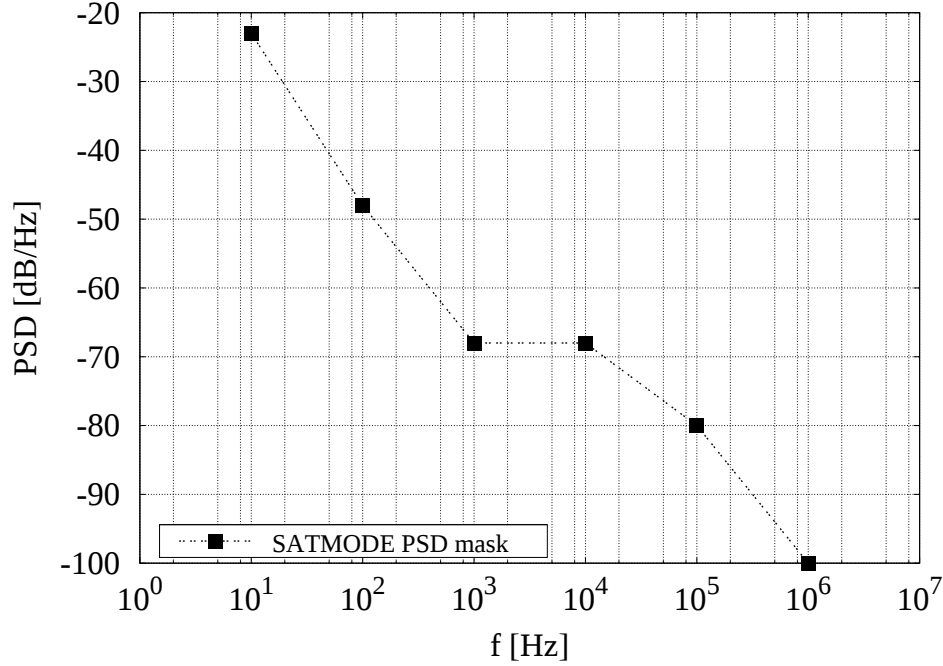


Figure 2.2: SATMODE PSD of the continuous-time phase process $\theta(t)$.

PSD mask of the continuous-time process.

By analyzing the SATMODE PSD, represented in Fig 2.2, we note that the mask can be seen as the sum of two different slopes and hence the PN process can be well approximated as the sum of two 1st order auto-regressive (AR1) Gaussian processes, instead of just one AR1. In detail, one AR1 describes the PN behaviour at low frequencies (slow process characterized by higher power) while the other one represents the PN at higher frequencies (fast process characterized by lower power).

In general a Gaussian AR1 process is defined as

$$u_k = a u_{k-1} + v_{a,k} \quad (2.25)$$

where $v_{a,k}$ are zero-mean independent and identically distributed Gaussian random variables, with variance σ_v^2 and where a is a real value such that

$|a| < 1$ (to ensure stability). The variance of u_k is then represented by

$$\sigma_a^2 \triangleq \mathbb{E} \{ |u_k|^2 \} = \frac{\sigma_v^2}{1 - a^2} \quad (2.26)$$

and its PSD is

$$S_u(f) = \frac{\sigma_a^2 (1 - a^2)}{1 + a^2 - 2a \cos(2\pi f T_S)}, \quad f \in \left[-\frac{1}{2T_S}, \frac{1}{2T_S} \right]. \quad (2.27)$$

Thus, in our case we have

$$\theta_k \triangleq u_{a,k} + u_{b,k} \quad (2.28)$$

where

$$\begin{aligned} u_{a,k} &= a u_{a,k-1} + v_{a,k} \\ u_{b,k} &= b u_{b,k-1} + v_{b,k}. \end{aligned} \quad (2.29)$$

Hence, our aim is to find the four parameters $a, b, \sigma_a^2, \sigma_b^2$ such that the target PSD of the discrete-time phase process (Fig 2.2) is well approximated by

$$\begin{aligned} S_\theta(f) &= \frac{\sigma_a^2 (1 - a^2)}{1 + a^2 - 2a \cos(2\pi f T_S)} + \frac{\sigma_b^2 (1 - b^2)}{1 + b^2 - 2b \cos(2\pi f T_S)}, \\ f &\in \left[-\frac{1}{2T_S}, \frac{1}{2T_S} \right]. \end{aligned} \quad (2.30)$$

In Fig 2.3 we report the SATMODE PSD mask along with the PSDs resulting from the double-AR1 approximations with three different set of parameters, reported in Table 2.2. The approximations denoted as UB (“upper bound”) and LB (“lower bound”) represent respectively an overestimated and an underestimated approximation of the target PSD, while the IA (“intermediate approximation”) is a PSD representation falling between the two extreme representations. From now on, the PN random process will be always generated according to the IA double-AR1 approximations. Moreover, since the PSD of the oversampled discrete-time phase process we generate by such parameters is equal to that of the continuous-time process up to $1/T_S$, we will denote such PN generation method as the continuous-time (CT) generation, in contrast with the symbol-time generation (ST) we will describe in the following.

	a	σ_a^2	b	σ_b^2
LB	$1 - 10^{-7}$	1.40	0.980	$8 \cdot 10^{-3}$
IA	$1 - 10^{-7}$	2.45	0.985	$16.2 \cdot 10^{-3}$
UB	$1 - 10^{-6}$	1.00	0.987	$2 \cdot 10^{-2}$

Table 2.2: Parameters of the oversampled continuous-time phase noise double-AR1 approximation.

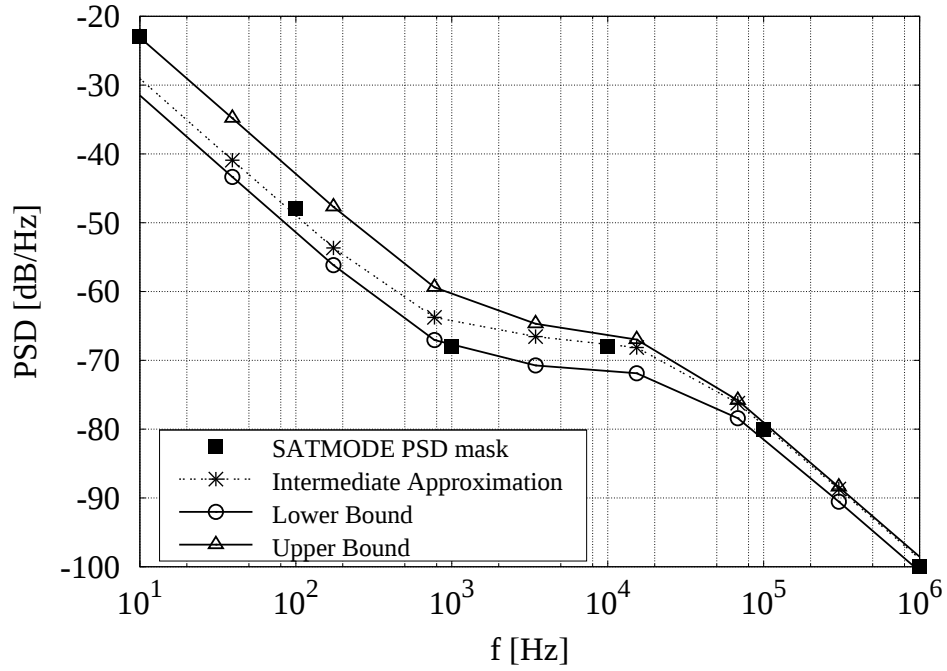


Figure 2.3: SATMODE PSD mask and PSDs obtained from various double-AR1 approximations.

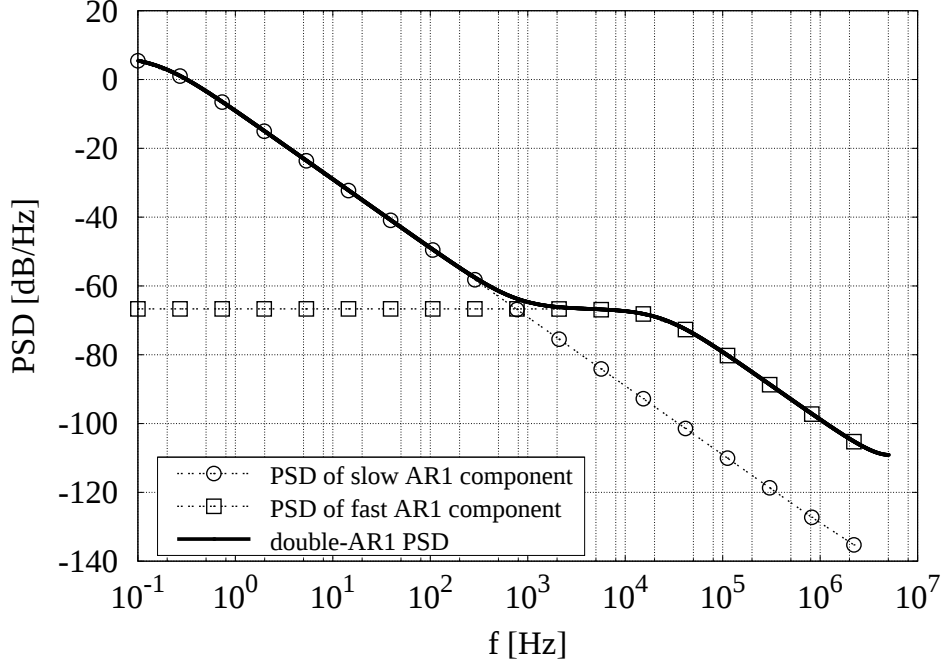


Figure 2.4: PSDs of slow and fast AR1 obtained with IA parameters.

In order to better understand the effects of the two PN components, we report two figures. In the first one, Fig 2.4, we see the PSD of each single AR1 components evaluated by replacing in (2.27) the IA parameters of Table 2.2: if one AR1 (denoted by a) determines the resulting phase noise PSD at lower frequencies, the other one (denoted by b) determines the phase noise PSD at higher frequencies. Fig 2.5 shows the time-domain snapshot of the two components of the continuous-time phase noise $\theta(t)$, i.e., the slow one and the fast one, generated following the double-AR1 approximations with IA parameters of Table 2.2. The behaviour of the two PN components is clear: the fast component is affected by very quick and low power fluctuations, while the slow component is characterized by high variance and slow fluctuations. Note that in practice it is very difficult to track the faster component, therefore it is

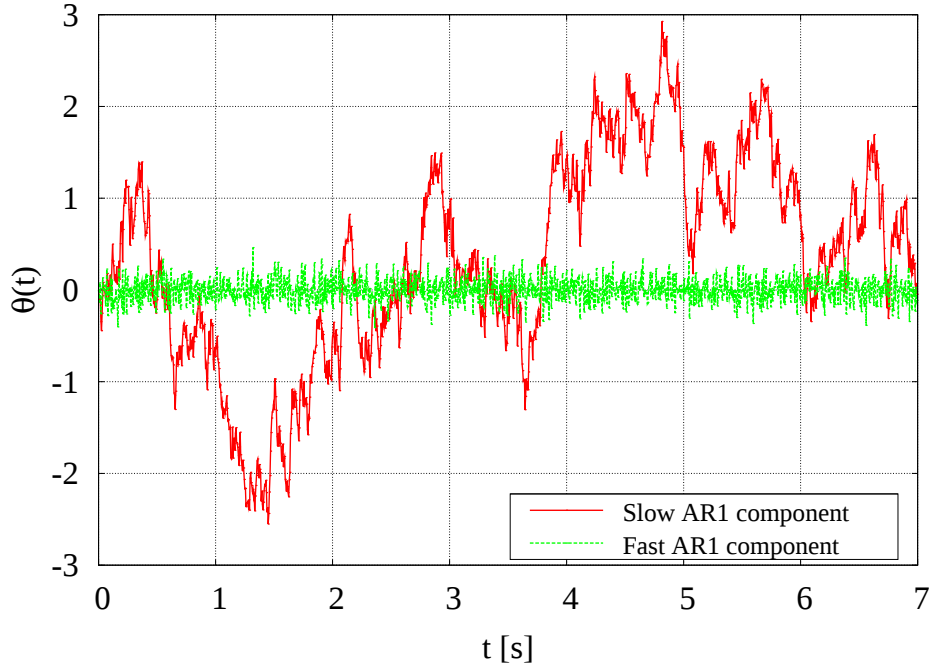


Figure 2.5: Time snapshot of the two phase noise AR1 components, generated according to the IA parameters.

the major reason of performance degradation. We discuss in the following, the effect of phase noise on digital communications. Given the CPM signal (1.1), the useful received signal is

$$r(t) = s(t; \mathbf{x})e^{j\theta(t)}.$$

We do not care about thermal noise since its statistics are not affected by PN. At the k -th symbol interval the output of the bank of matched filters is

$$e^{-j\pi_n} \int_{nT}^{(n+1)T} s(t; \mathbf{x}) \bar{s}^*(t - nT; x_n, \omega_n) e^{-j\theta(t)} dt \quad (2.31)$$

where the CPM waveform $\bar{s}^*(t - nT; x_n, \omega_n)$ and the phase state π_n are defined in (1.17) and (1.5) respectively. Due to the presence of the PN $\theta(t)$, the output

of the matched filter $\bar{s}^*(-t; x_n, \omega_n)$ is no more a sufficient statistic but it is a corrupted version of the sufficient statistic $y_n(x_n, \sigma_n)$ defined in (2.12). In other words, the phase noise causes a sort of *inter-symbol interference* (ISI) which, in principle, can have an extremely harmful effect when a receiver that neglects it is employed.

However, we verified that this is not the case in all scenarios of interest (as shown later). Hence, we neglect its effect and we also provide a different PN generation, denoted as symbol-time (ST) generation, by which just one sample is generated at each symbol interval, without taking into account the matched filtering distortion presents in the continuous-time (CT) generation. Then the matched filters output (2.31) can be approximated as

$$e^{-j\pi n} \int_{nT}^{(n+1)T} s(t; \mathbf{x}) \bar{s}^*(t - nT; x_n, \omega_n) e^{j\theta(t)} dt \simeq y_n(x_n, \sigma_n) e^{j\psi_n} \quad (2.32)$$

where

$$\psi_n = \frac{1}{T} \int_{nT}^{(n+1)T} \theta(t) dt. \quad (2.33)$$

Thus the PN process at symbol time is approximately a windowed and down-sampled version of the original PN process. In order to obtain a statistical characterization of ψ_n as done for the continuous-time process $\theta(t)$, we derive the autocorrelation function (ACF) of ψ_n

$$R_\psi(m) = \mathbb{E} \{ \psi_n \psi_{n-m} \}. \quad (2.34)$$

In Fig 2.6 the autocorrelation function (2.34) for a signal rate of 256 kBaud is carried out by numerical simulation: we have generated various $\theta(t)$ sequences employing IA parameters, derived the corresponding ψ_n sequences following (2.33) when the symbol rate is $R = 1/T = 256$ kBaud and then computed $R_\psi(m)$ by averaging over all simulated ψ_n sequences. We note from Fig 2.6 that the symbol-time phase noise process ψ_n can again be approximated as the sum of two independent AR1 processes, a slowly varying (i.e., highly correlated) component and a rapidly varying and low variance component. The set of parameters $a, b, \sigma_a^2, \sigma_b^2$ by which we derived the ACF in Fig 2.6

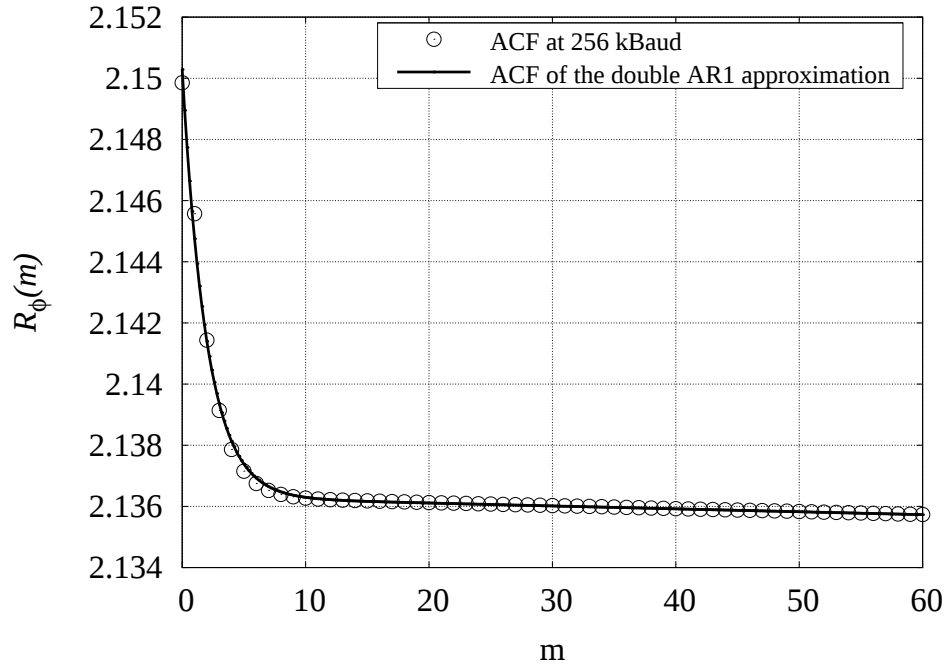


Figure 2.6: Autocorrelation function of the continuous-time PN process $\theta(t)$ and of the symbol-time PN process ψ_n at 256 kBaud.

are listed in Table 2.3, together with the same parameters obtained for a set of selected baudrates. As mentioned, the major reason of performance degr-

Baudrate [KBaud]	a	σ_a^2	b	σ_b^2
64	$1 - 1.76 \cdot 10^{-5}$	2.163	0.263	$8.51 \cdot 10^{-3}$
128	$1 - 8.9 \cdot 10^{-6}$	2.143	0.447	$1.150 \cdot 10^{-2}$
256	$1 - 4.45 \cdot 10^{-6}$	2.136	0.604	$1.399 \cdot 10^{-2}$
512	$1 - 2.15 \cdot 10^{-6}$	2.140	0.758	$1.558 \cdot 10^{-2}$
1024	$1 - 1.05 \cdot 10^{-6}$	2.126	0.871	$1.611 \cdot 10^{-2}$
2048	$1 - 4.7 \cdot 10^{-7}$	2.144	0.942	$1.609 \cdot 10^{-2}$

Table 2.3: Parameters of the symbol-time phase noise double-AR1 approximation, provided for different baudrate values.

dation is the faster component, which is very difficult to track. For example, from Fig. 2.6 it is clear that such a component at 256 KBaud has a standard deviation of about 4 degrees and the corresponding samples at symbol-time are almost independent.

Taking advantage of the double-AR1 PN description, it is interesting to investigate the IR of a channel affected by SATMODE phase noise, in order to quantify the mutual information rate loss due to such a PN. In particular we evaluate IR according to the Arnold and Loeliger method, for different channel representations and different detection algorithms (necessary for the forward metric computation). In particular, we consider two possible channel generation models:

- the real channel, which works in a practically continuous-time setting (i.e., sampling time 10^{-7} s), by adding an oversampled version of the PN process $\theta(t)$ generated according to the CT double-AR1 approximation, with IA set of parameters; in such a case an ISI effect on the sufficient

statistics may occurs;

- the auxiliary channel, which introduces a discrete-time PN process ψ_n constant over each symbol period and following the ST double-AR1 approximation.

The detection algorithms we propose are all based on MAP symbol detection, but in general they cannot be matched to the channel generation model. Moreover they are all derived following a Bayesian approach, i.e., we assume a proper probabilistic model for the phase noise $\{\theta_n\}$, and we exploit it for deriving algorithms for MAP symbol detection. In detail we consider:

- DP-BCJR, already anticipated in Section 2.4.1 and detailed in Section 2.4.3; in the algorithm derivation we consider a first order auto-regressive PN model or a Wiener PN model.
- Double-DP-BCJR, which is a generalization (provided in Section 2.4.3) of the DP-BCJR; in the algorithm derivation we consider two first order auto-regressive components for the PN model and not just one;
- Improved-DP-BCJR (Section 2.4.3), which assumes both the two AR1 components, but considers all the sample of one component (the faster one) independent to each others.

We note that when the real channel generated by the CT double-AR1 approximation is considered, there is no one detector that can be considered matched to the channel since all algorithms are based on a phase discretization technique (even if the number of levels is very large) and all algorithms consider just one PN sample for each symbol interval (and not a continuous-time generation). However it is interesting to evaluate the IR for the real channel with the Double-DP-BCJR in order to quantify the effect of the ISI distortion. On the other hand, when we take into account the ST channel generation, except for the phase discretization (which causes just a little mismatch between the detector and the channel model), the double-DP-BCJR is the optimum one

in the MAP sense. Moreover, it is interesting also to evaluate the IR by a DP-BCJR operating over a CT or ST channel generation, since we derive the maximum performance we can achieve, when employing a detection algorithm carried out by starting from a single first order PN model (Wiener or AR1), over a channel affected by SATMODE PN. Finally, the third type of detector is particularly suitable to compute the IR for double-AR1 PN generation at low baudrate values. Looking to the double-AR1 parameters in Table 2.3 at 64 kBaud, for example, we see that the fast components exhibits a b value so close to zero that in the detector derivation, we can consider its corresponding PN samples as almost independent. In such a way, in the detection derivation, we take into account both the two AR1 component but we also achieve a reduction in the detector complexity.

2.4.3 Detectors for IR Computation over Channels Affected by Phase Noise

All the following algorithms are derived by a Bayesian approach, and based on the trellis of optimal detection algorithm for CPM over AWGN channel, reported in Section 2.3.1.

DP-BCJR (DP)

We take into account the PN channel model (2.22) where the continuous-time process $\theta(t)$ is a Wiener process. Assuming the PN is slowly varying with respect to the symbol interval T (as done in Section 2.4.1), (2.22) in the n -th interval can be approximated as

$$r(t) \simeq s(t; \mathbf{x})e^{j\theta_n} + w(t), \quad t \in [nT, (n+1)T) \quad (2.35)$$

where θ_n represents a discrete-time Wiener process (2.24). In that case, the overall system state is composed by the CPM state σ_n and the phase value θ_n .

We project the receiver signal $r(t)$ onto the orthonormal basis by means of matched filters; hence for each trial symbol x_n , each trial CPM state σ_n and

trial phase value θ_n , we define the sample $r_n(x_n, \sigma_n, \theta_n)$ as

$$\begin{aligned} r_n(x_n, \sigma_n, \theta_n) &\triangleq \int_{nT}^{(n+1)T} r(t) \tilde{s}^*(t - nT; x_n, \sigma_n) e^{-j\theta_n} dt \\ &= y_n(x_n, \sigma_n) e^{-j\theta_n} \end{aligned} \quad (2.36)$$

where the vector $\mathbf{y}$ collecting $y_n(x_n, \sigma_n)$ (2.12) represents a set of sufficient statistic of the received signal.³ Finally we call $\mathbf{r}$ the vector collecting the sample $r_n(x_n, \sigma_n, \theta_n)$ for each trial symbol x_n , each trial CPM state σ_n , and phase value θ_n , and each discrete-time instant.

We now derive the MAP symbol detection through FG and SPA. To this purpose, we first factorize the joint distribution $p(\mathbf{x}, \boldsymbol{\sigma}, \boldsymbol{\theta} | \mathbf{r})$ as

$$\begin{aligned} p(\mathbf{x}, \boldsymbol{\sigma}, \boldsymbol{\theta} | \mathbf{r}) &\propto p(\mathbf{r} | \mathbf{x}, \boldsymbol{\sigma}, \boldsymbol{\theta}) P(\boldsymbol{\sigma} | \mathbf{x}) P(\mathbf{x}) p(\boldsymbol{\theta}) \\ &= P(\sigma_0) P(\theta_0) \prod_{n=0}^{N-1} F_n(x_n, \sigma_n, \theta_n) T(x_n, \sigma_n, \sigma_{n+1}) P(x_n) p(\theta_{n+1} | \theta_n) \end{aligned} \quad (2.37)$$

where $\boldsymbol{\theta}$ collects θ_n samples (n from 0 to N), $T(x_n, \sigma_n, \sigma_{n+1})$ is an indicator function, equal to one if x_n, σ_n and σ_{n+1} satisfy the CPM trellis constraints and to zero otherwise, and $F_n(x_n, \sigma_n, \theta_n)$ is the branch metric function, defined as

$$F_n(x_n, \sigma_n, \theta_n) = \exp \left[\frac{1}{N_0} \text{Re} \left\{ y_n(x_n, \sigma_n) e^{-j\theta_n} \right\} \right]. \quad (2.38)$$

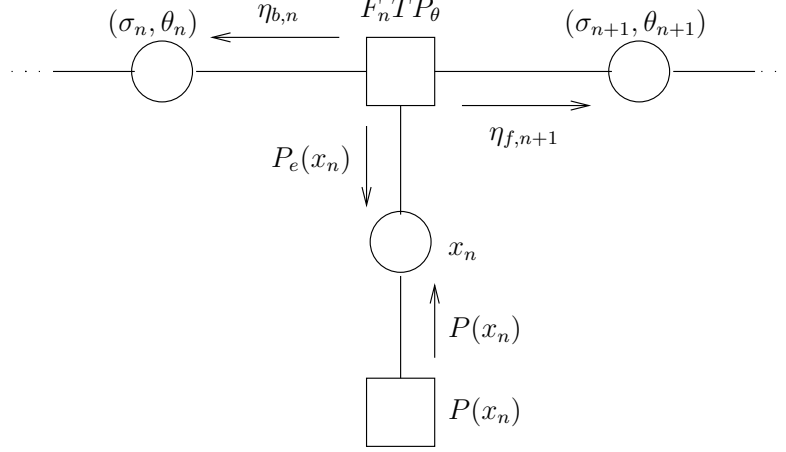
Finally, as stated in Section 2.4.1, $p(\theta_{n+1} | \theta_n)$ in (2.37) is a Gaussian pdf⁴

$$p(\theta_{n+1} | \theta_n) = \frac{1}{\sqrt{2\pi\sigma_\Delta^2}} \exp \left\{ -\frac{(\theta_{n+1} - \theta_n)^2}{2\sigma_\Delta^2} \right\}. \quad (2.39)$$

In order to have a finite state representation for such a channel, we discretize the values that samples θ_n may assume. Obviously, we are approximating the

³Note that also in this case, as well as in the coherent optimal detector, the vector $\mathbf{y}$ is obtained from a set of just M^L filters.

⁴Note that, since the channel phase is defined modulo 2π , the probability density function $p(\theta_{n+1} | \theta_n)$ can be approximated as Gaussian only if $\sigma_\Delta \ll 2\pi$.

Figure 2.7: Factor graph of DP-BCJR in the n -th time-interval.

real channel and, by increasing the number of quantization levels, better approximations can be achieved at the price of a greater channel state cardinality. In particular, we propose an uniform discretization, by which samples θ_n are considered belonging to the below alphabet

$$\theta_n \in \left\{ 0, \frac{2\pi}{D}, 2\frac{2\pi}{D}, \dots, (D-1)\frac{2\pi}{D} \right\} \quad (2.40)$$

where D is the number of quantization levels. Hence, the pdf $p(\theta_{n+1}|\theta_n)$ in (2.39) must be replaced by the pmf $P_\theta(\theta_{n+1}|\theta_n)$, which can be chosen for example as:

$$P_\theta(\theta_{n+1}|\theta_n) = \int_{\theta_{n+1}-\frac{\pi}{D}}^{\theta_{n+1}+\frac{\pi}{D}} \frac{1}{\sqrt{2\pi\sigma_\Delta^2}} \exp \left\{ -\frac{(x-\theta_n)^2}{2\sigma_\Delta^2} \right\} dx. \quad (2.41)$$

The FG corresponding to (2.37) has cycles. However, by clustering [7] the variables θ_n and σ_n , we obtain the FG in Fig. 2.7. Thus, by applying to it the SPA by using a *non-iterative* forward-backward schedule, we obtain the exact a posteriori probabilities $P(x_n|\mathbf{r})$ necessary to implement the MAP symbol detection strategy. With reference to the messages in Fig. 2.7, the resulting forward-backward algorithm is characterized by the following recursions and

completion:

$$\eta_{f,n+1}(\sigma_{n+1}, \theta_{n+1}) = \sum_{\sigma_n} \sum_{\theta_n} \sum_{x_n} P(x_n) F_n(x_n, \sigma_n, \theta_n) T(x_n, \sigma_n, \sigma_{n+1}) \quad (2.42)$$

$$\eta_{b,n+1}(\sigma_n, \theta_n) = \sum_{\sigma_{n+1}} \sum_{\theta_{n+1}} \sum_{x_n} P(x_n) F_n(x_n, \sigma_n, \theta_n) T(x_n, \sigma_n, \sigma_{n+1}) \quad (2.43)$$

$$P_e(x_n) \propto \sum_{\sigma_n} \sum_{\sigma_{n+1}} \sum_{\theta_n} \sum_{\theta_{n+1}} F_n(x_n, \sigma_n, \theta_n) T(x_n, \sigma_n, \sigma_{n+1}) P_\theta(\theta_{n+1}|\theta_n) \eta_{f,n}(\sigma_n, \theta_n) \eta_{b,n+1}(\sigma_{n+1}, \theta_{n+1}) \quad (2.44)$$

with the following initial conditions for the two recursions: $\eta_{f,0}(\sigma_0, \theta_0) = P(\sigma_0)/(2\pi)$ and $\eta_{b,N}(\sigma_N, \theta_N) = P(\sigma_N)/(2\pi)$. It is clear that a better approximation of the real channel, which is achieved by increasing the number of quantization levels D , determines also an increased detection complexity, since the cardinality of the trellis state (σ_n, θ_n) is proportional to $D p M^{L-1}$.

Finally, we can extend the DP-BCJR algorithm to all cases in which the discrete-time PN process is a Gaussian AR1 model (2.25). In such a case, it is sufficient to replace the pmf $P_\theta(\theta_{n+1}|\theta_n)$ in (2.41) by

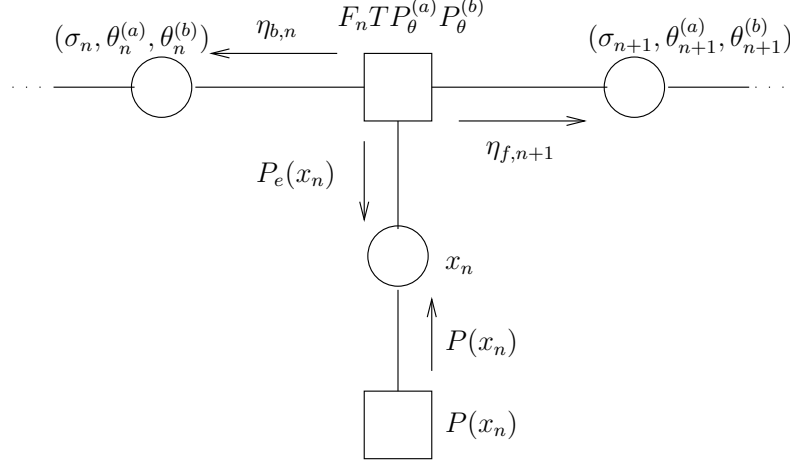
$$P_\theta(\theta_{n+1}|\theta_n) = \int_{\theta_{n+1}-\frac{\pi}{D}}^{\theta_{n+1}+\frac{\pi}{D}} \frac{1}{\sqrt{2\pi(1-a^2)\sigma_a^2}} \exp \left\{ -\frac{(x - a\theta_n)^2}{2(1-a^2)\sigma_a^2} \right\} dx. \quad (2.45)$$

Double-DP-BCJR (D-DP)

We now consider the double-AR1 phase noise model described in Section 2.4.2. Thus, the above DP-BCJR algorithm derivation can be generalized to the case of a channel whose PN model is given by the sum of two discrete-time processes $\theta_n^{(a)}$ and $\theta_n^{(b)}$ for which the following recursive definitions exist

$$\theta_{n+1}^{(a)} = a \theta_n^{(a)} + v_n^{(a)} \quad (2.46)$$

$$\theta_{n+1}^{(b)} = b \theta_n^{(b)} + v_n^{(b)} \quad (2.47)$$

Figure 2.8: FG of Double-DP-BCJR detector in the n -th time-interval.

where $v_n^{(a)}$ and $v_n^{(b)}$ are independent and identically distributed Gaussian zero-mean random variables of variance $(1-a^2)\sigma_a^2$ and $(1-b^2)\sigma_b^2$, respectively. Also in this case, the two phase values can be uniformly discretized in the domain and we denote by D_a and D_b the number of quantization levels for $\theta_n^{(a)}$ and $\theta_n^{(b)}$, respectively.

In order to apply the MAP symbol detection strategy, we factorize the global pmf $P(\mathbf{x}, \boldsymbol{\sigma}, \boldsymbol{\theta}^{(a)}, \boldsymbol{\theta}^{(b)} | \mathbf{r})$ in a way similar to (2.37) and we derive the FG in Fig. 2.8. The SPA allows us to compute the two recursions and completion; here we report just the expression for the forward recursion:

$$\begin{aligned} \eta_{f,n+1}(\sigma_{n+1}, \theta_{n+1}^{(a)}, \theta_{n+1}^{(b)}) &= \sum_{\sigma_n} \sum_{\theta_n^{(a)}} \sum_{\theta_n^{(b)}} \sum_{x_n} F_n(x_n, \sigma_n, \theta_n^{(a)}, \theta_n^{(b)}) T(x_n, \sigma_n, \sigma_{n+1}) \\ &\quad P(x_n) P_{\theta}^{(a)}(\theta_{n+1}^{(a)} | \theta_n^{(a)}) P_{\theta}^{(b)}(\theta_{n+1}^{(b)} | \theta_n^{(b)}) \eta_{f,n}(\sigma_n, \theta_n^{(a)}, \theta_n^{(b)}) \end{aligned} \quad (2.48)$$

where we have defined

$$F_n(x_n, \sigma_n, \theta_n^{(a)}, \theta_n^{(b)}) = \exp \left[\frac{1}{N_0} \text{Re} \left\{ y_n(x_n, \sigma_n) e^{-j(\theta_n^{(a)} + \theta_n^{(b)})} \right\} \right] \quad (2.49)$$

and $P_\theta^{(a)}(\theta_{n+1}^{(a)}|\theta_n^{(a)})$, $P_\theta^{(b)}(\theta_{n+1}^{(b)}|\theta_n^{(b)})$ are the pmfs describing the channel phase transition, defined as in (2.45).

In such a case, the trellis state is $(\sigma_{n+1}, \theta_{n+1}^{(a)}, \theta_{n+1}^{(b)})$ and its cardinality results $D_a D_b p M^L$.

Improved-DP-BCJR (I-DP)

Looking at the parameters describing the faster PN component in Table 2.3, we see that its b parameter is very close to zero, especially for low baudrate values. In other words, PN samples $\theta_n^{(b)}$ can be considered independent from each other and we assume the following model:

$$\theta_n^{(b)} = v_n^{(b)} \quad (2.50)$$

obtained replacing $b = 0$ in (2.46); hence

$$P(\theta_n^{(b)}) = \prod_{n=0}^{N-1} P_\theta^{(b)}(\theta_n^{(b)}). \quad (2.51)$$

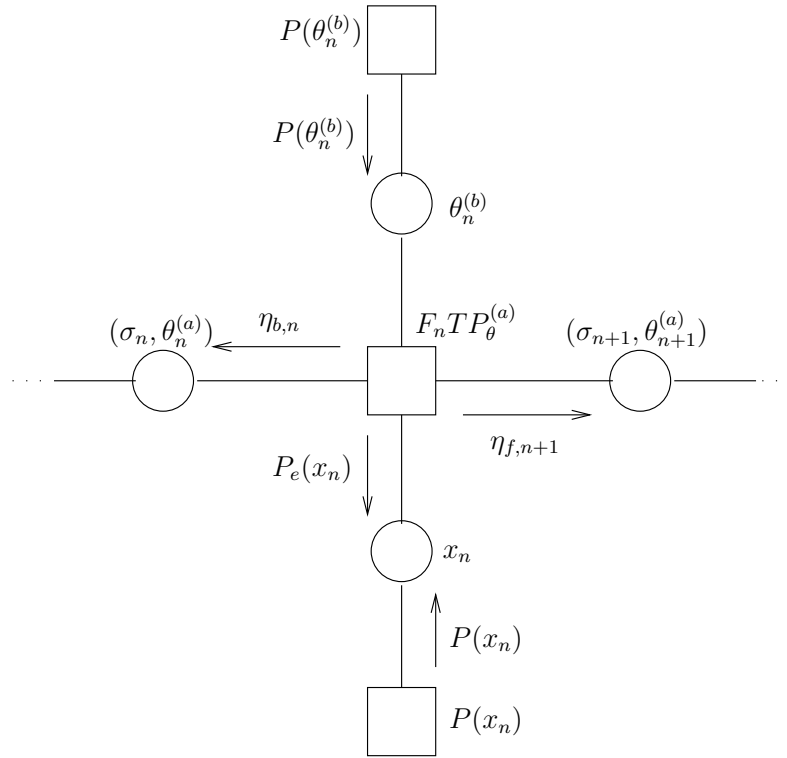
where $P_\theta^{(b)}(\theta_n^{(b)})$ expression is equivalent to (2.45) with $a = 0$. By factorizing the pmf $P(\mathbf{x}, \boldsymbol{\sigma}, \boldsymbol{\theta}^{(a)}, \boldsymbol{\theta}^{(b)}|\mathbf{r})$, we derive a new detection algorithm based on assumption (2.50), whose factor graph is provided in Fig 2.9. The forward recursion expression obtained through the SPA is

$$\begin{aligned} \eta_{f,n+1}(\sigma_{n+1}, \theta_{n+1}^{(a)}) &= \sum_{\sigma_n} \sum_{\theta_n^{(a)}} \sum_{x_n} P(x_n) F_n^{(a)}(x_n, \sigma_n, \theta_n^{(a)}) T(x_n, \sigma_n, \sigma_{n+1}) \\ &\quad P_\theta^{(a)}(\theta_{n+1}^{(a)}|\theta_n^{(a)}) \eta_{f,n}(\sigma_n, \theta_n^{(a)}). \end{aligned} \quad (2.52)$$

where the branch metric is

$$F_n^{(a)}(x_n, \sigma_n, \theta_n^{(a)}) \triangleq \sum_{\theta_n^{(b)}} F_n(x_n, \sigma_n, \theta_n^{(a)}, \theta_n^{(b)}) P_\theta^{(b)}(\theta_n^{(b)}) \quad (2.53)$$

and $F_n(x_n, \sigma_n, \theta_n^{(a)}, \theta_n^{(b)})$ is provided in (2.49). Hence, we can see that employing the assumption in (2.50), the double-DP-BCJR complexity is reduced since

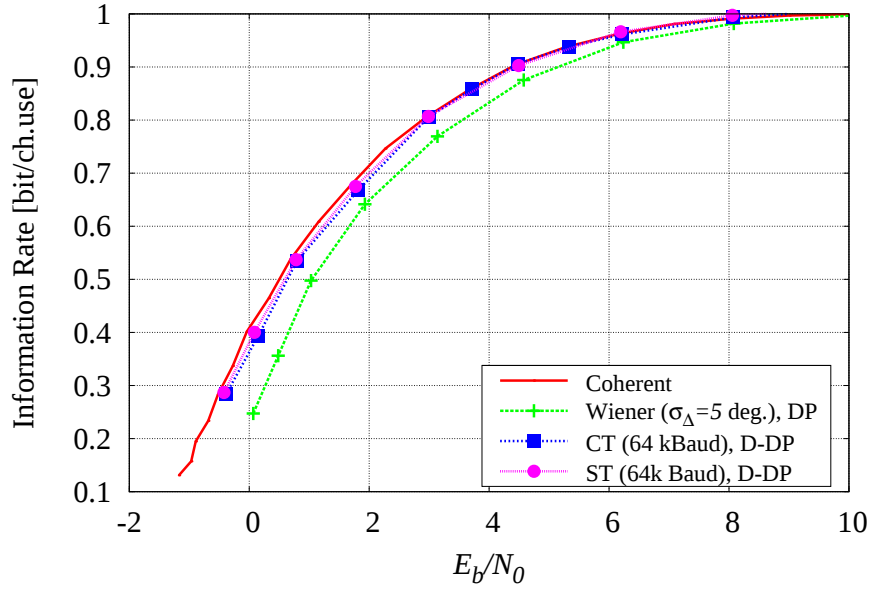
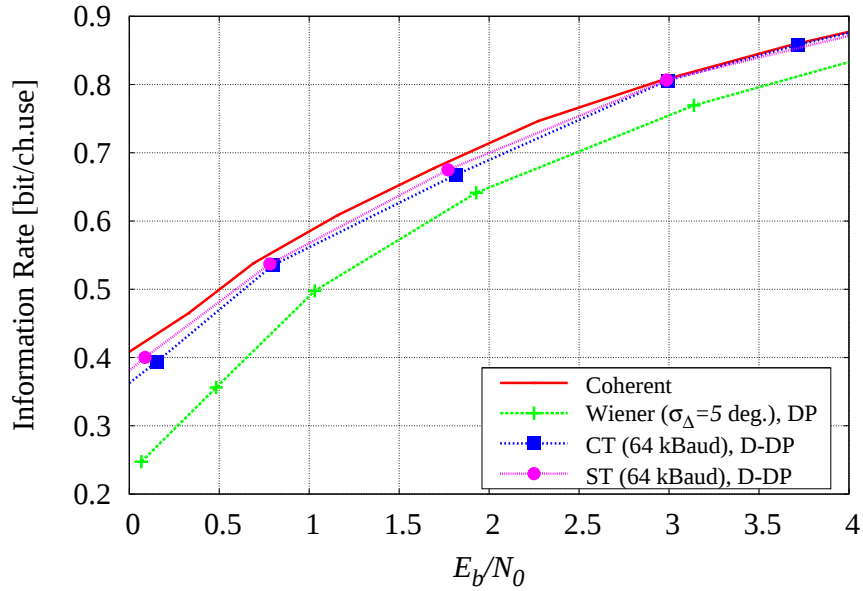
Figure 2.9: FG of I-DP-BCJR detector in the n -th time-interval.

here the trellis state is the same of the DP-BCJR, i.e. $(\sigma_n, \theta_n^{(a)})$. However, with respect to the DP-BCJR algorithm we take into account a second PN component, $\theta_n^{(b)}$, independent from each interval n , and so the branch metric computation is increased (see (2.53)).

2.4.4 Numerical results

In the following, we show some numerical results associated to the IR computation method described until now. We will consider IR of two CPM schemes over AWGN channel (curves denoted as *coherent*) as well as over channels affected by Wiener PN or by SATMODE PN. As already stated in Section 2.4.2, in the latter case we will consider two possible phase noise generation methods, i.e., the continuous-time (CT) generation and the symbol-time (ST) generation, and we will compute IR with three different types of detectors, the DP, the D-DP and the I-DP (derived in Section 2.4.3). In such a way we derive not the real IR, but just a lower bound on it, which represents the maximum information rate constrained to the employed detector. Such a technique is very useful in order to understand what is the maximum performance we can expect from various detectors.

We start by illustrating the information rate of a binary CPM format with $L = 2$, raised-cosine frequency pulse (2-RC) modulation scheme with $h = 1/3$. In all the proposed figures, we will report on the y-axis the information rate expressed in b/ch.use and on the x-axis E_b/N_0 , E_b being the received energy per information bit. In Fig. 2.10, we provide the IR curve for a Wiener channel with standard deviation $\sigma_\Delta = 5$ degrees and we use the DP-BCJR detector matched to the channel ($\sigma_\Delta = 5$) degrees also for the PN model employed in the detector derivation. We note that no perfect matching between the channel and the detector can be achieved, since the channel generates PN samples belonging to the continuous range $[0, 2\pi)$, while the detector assumes discretized PN values. Since a mismatch is always present for all couples channel-detector we can take into account, we introduce a parameter N_I which is a constant (lower than one) by which we multiply the thermal noise variance N_0 in all the branch

(a) E_b/N_0 from -2 to 10 dB(b) E_b/N_0 from 0 to 4 dBFigure 2.10: Information rate for a binary modulation with 2-RC $h = 1/3$ and D-DP detector.

metrics computation (see (2.38), (2.49), (2.53)). In such a way, we consider the mismatch by reducing the reliability of the detector decisions and, by properly optimizing the parameter N_I , we can increase the IR. The obtained IR is still achievable by the detector which considers a reduced thermal noise variance. In the detector associated to the Wiener channel in Fig 2.10, we choose $N_I = 10^{-2}$ and $D = 101$ quantization levels. The information rate loss with respect to the coherent channel (i.e., the IR loss due to the PN) can be of approximately 1 dB at high IR values. In Fig. 2.10 we also consider SATMODE PN model CT generated and ST generated, a symbol frequency of 64 kBaud and the D-DP detector. In both cases we discretize the two PN components in two different ways. For the slow component we uniformly discretized the interval $[-\pi, \pi)$ with $D_a = 157$ levels, while for the fast component we adopt a non uniform discretization, but thicker around the origin. For such a component just $D_b = 19$ levels are sufficient. Hence, looking at the two curves obtained when we consider CT and ST double-AR1 PN generation and D-DP detector, we can verify that the IR loss due to the PN is practically negligible (0.1 dB or less). We understand that the ISI effect, present in the CT generation case, is not relevant, since the curve corresponding to such a case is almost overlapped to the curve associated to the ST case. So in the following we will always consider just the ST double-AR1 model. In conclusion, when we employ a D-DP detector, we see that in principle we can achieve almost the same IR of the AWGN channel. However, the computational complexity (i.e., the number of trellis state) of the D-DP is too high and hence such a detector cannot be of practical interest.

Thus, in Fig. 2.11 we resort to some simplified detectors for the same scenario of Fig. 2.10. In detail, we generate the PN following the ST double-AR1 generation model but we employ a simple DP detector (with $D = 157$ quantization levels, $N_I = 10^{-1}$ and $a = 1 - 18 \cdot 10^{-6}$ and $\sigma_a^2 = 2.1625$). We have a reduction in the computational complexity of the detector but we pay the strong mismatch between the channel and the detector in terms of about than 0.5 dB of IR. Moreover, from Fig 2.12 we see that IR constrained to the DP

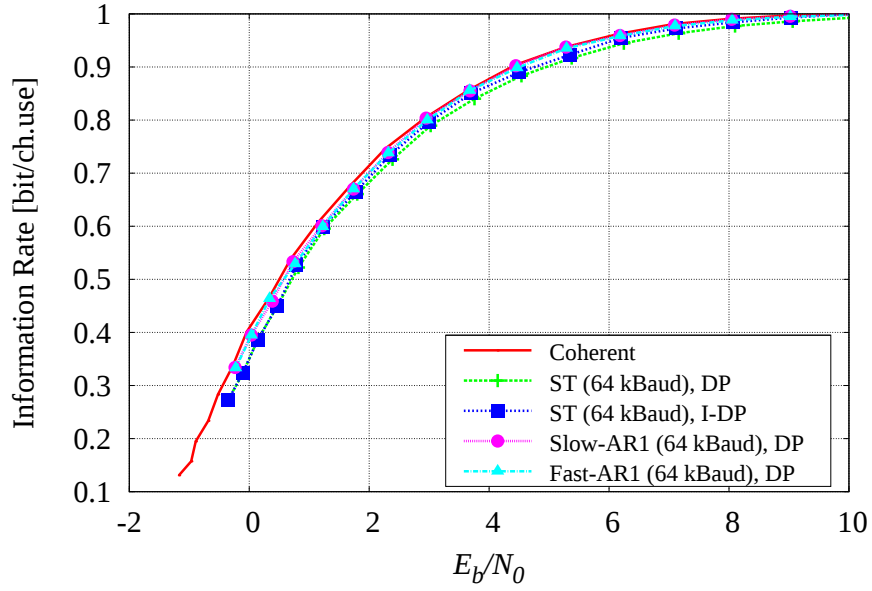
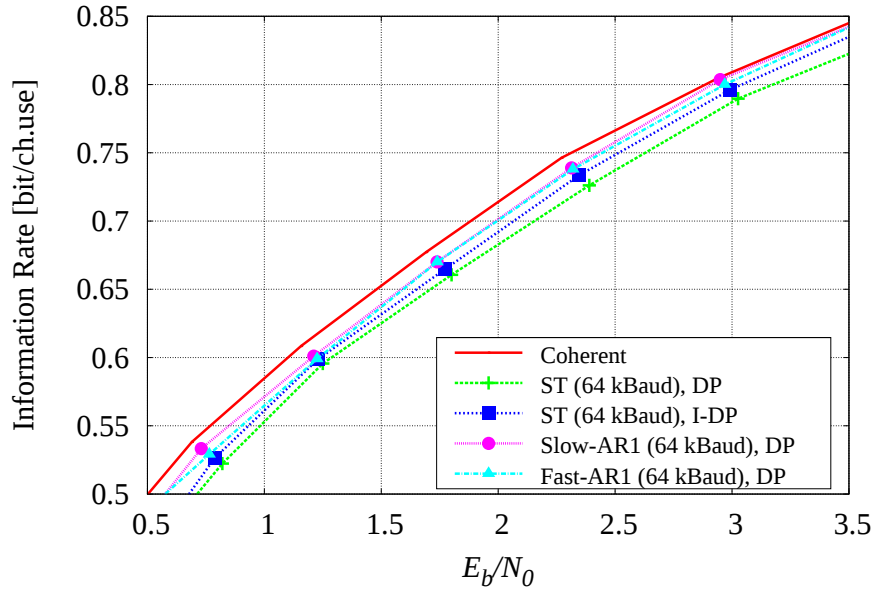
(a) E_b/N_0 from -2 to 10 dB(b) E_b/N_0 from 0.5 to 3.5 dB

Figure 2.11: Information rate for a binary modulation with 2-RC $h = 1/3$ and DP and I-DP detectors.

is just few tenths of dB far from IR constrained to the D-DP, so we conclude that a simplified detector based on a single first-order phase noise model (see for example the MM-DP algorithm in Section 4.4.2) can achieve approximately the same performance of a more complex algorithm base on D-DP. When we employ a I-DP, with $D = 157$ quantization levels for the slow component, we derive an IR slightly greater than the IR constrained to the DP. In other words, by simply increasing the branch metric computation of the DP (as done in the I-DP algorithm), we cannot recover the few tenths of dB which separate the IR curve constrained to the DP, from that one constrained to the D-DP. Finally, in Fig. 2.11 we also consider the case in which both the channel and the detector are based on a single-AR1 PN model, which is in turn matched to the slow or to the fast SATMODE PN components. We can see that the IR curves of both the slow and the fast components are practically overlapped to the IR curve of the coherent case. Hence, for the lowest spectral efficient CPM format we have considered, both components can be tracked by a detector based on a single-AR1 PN model matched to the channel; that result was expected, since also the performance of the D-DP detector in Fig. 2.10 is approximately overlapped to the coherent curve.

We now consider a more spectrally efficient CPM scheme: a quaternary 2-RC with $h = 1/5$ and with symbol frequency of 64 kBaud. In Fig. 2.13, we present, for such a quaternary scheme, the same curves reported in Fig. 2.10. Also in this case, the larger IR loss (from 1 to 2 dB) is exhibited by the Wiener channel, with $\sigma_\Delta = 5$ both in the channel and in the DP detector ($N_I = 10^{-2}$ and $D = 101$ quantization levels). The IR constrained to the D-DP detector (with $D_a = 101$ and $D_b = 19$) is less than one half dB (0.2 dB approximately) from the coherent IR, both for the channel with CT PN generation and for the channel with ST PN generation. Also in this case we can neglect the distortion effect due to the continuous-time PN generation (since the curve related to the CT case is overlapped to the ST curve), and with a D-DP detector we can obtain a performance quite close to the coherent one.

In Fig. 2.14, we take into account reduced-complexity detectors. First of

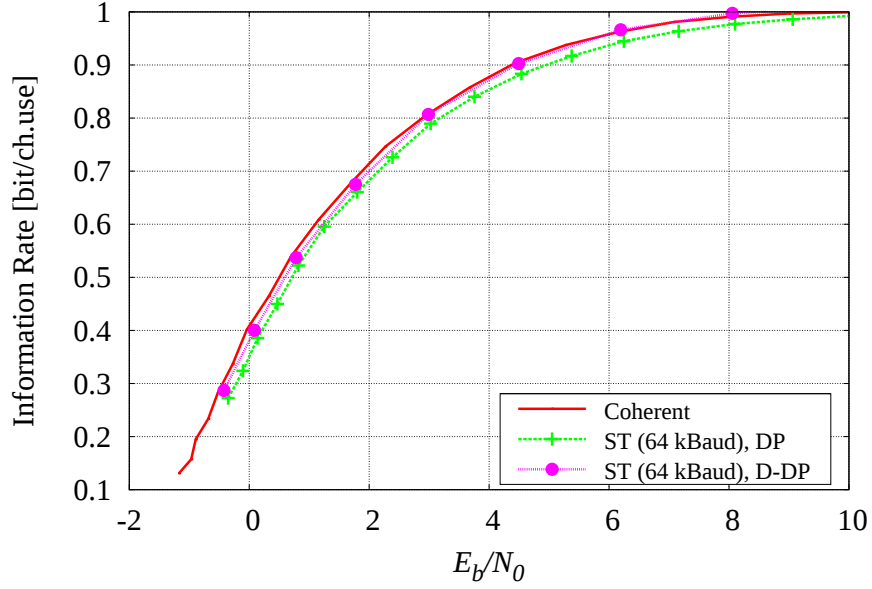
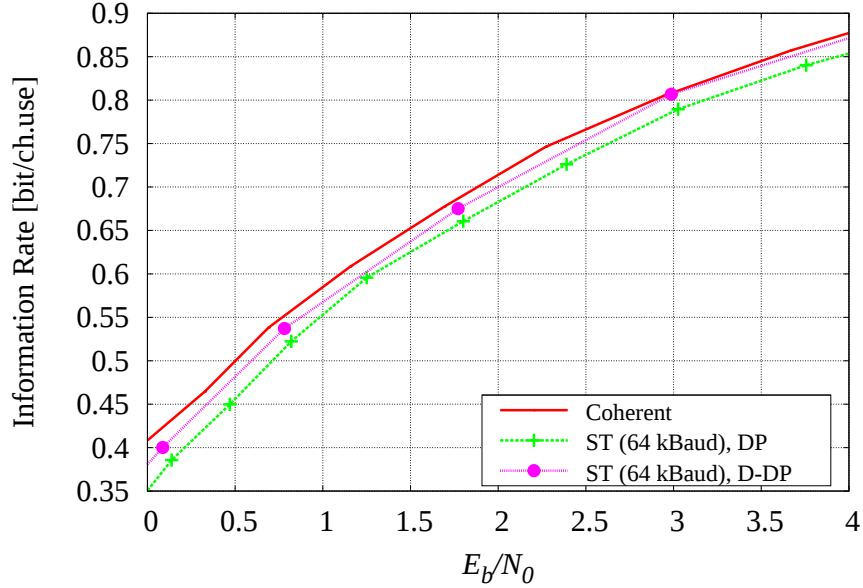
(a) E_b/N_0 from -2 to 10 dB(b) E_b/N_0 from 0.5 to 3.5 dB

Figure 2.12: Information rate for a binary modulation with 2-RC $h = 1/3$: comparison between D-DP and DP detectors.

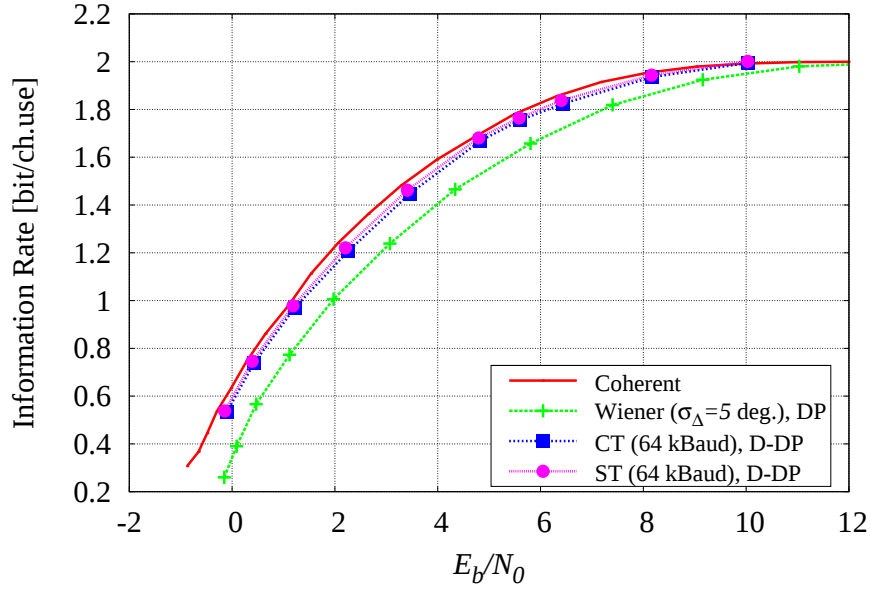
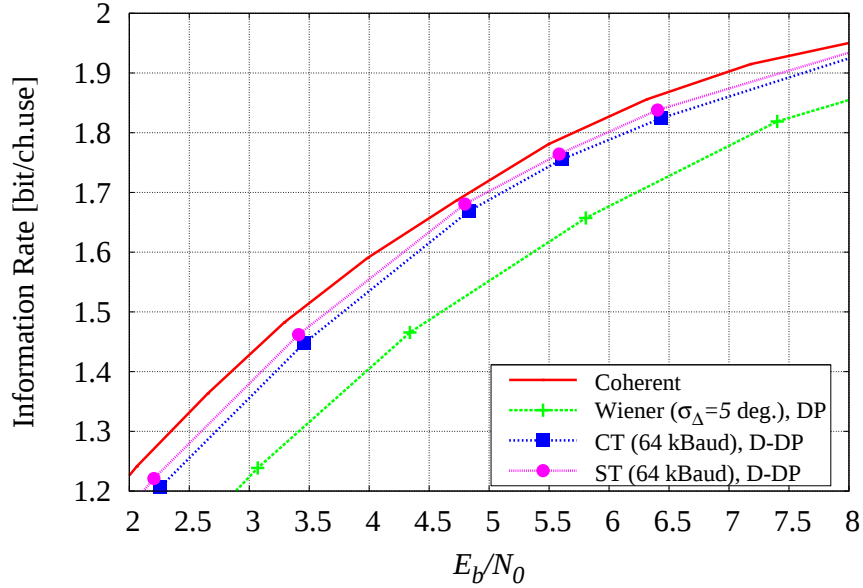
(a) E_b/N_0 from -2 to 12 dB(b) E_b/N_0 from 2 to 8 dB

Figure 2.13: Information rate for a quaternary modulation with 2-RC $h = 1/5$ and D-DP detector.

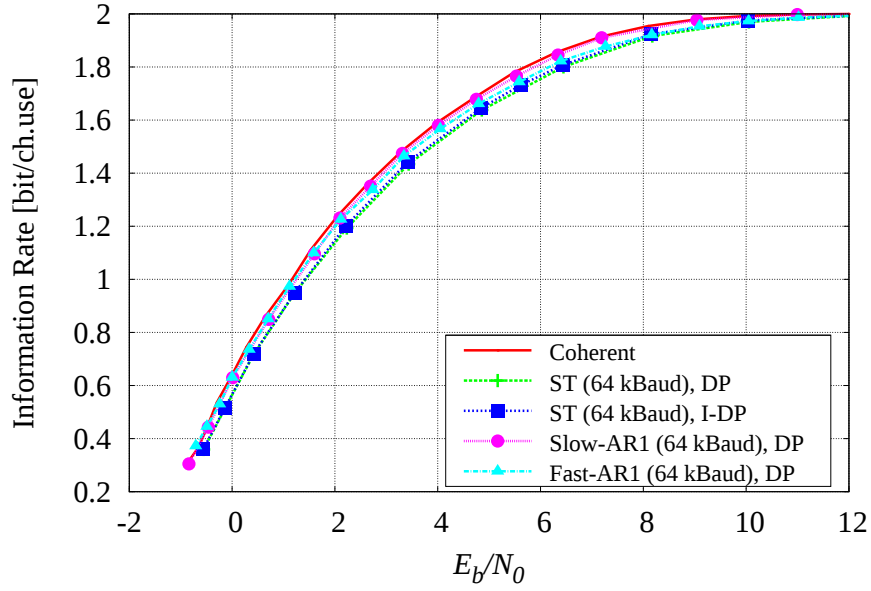
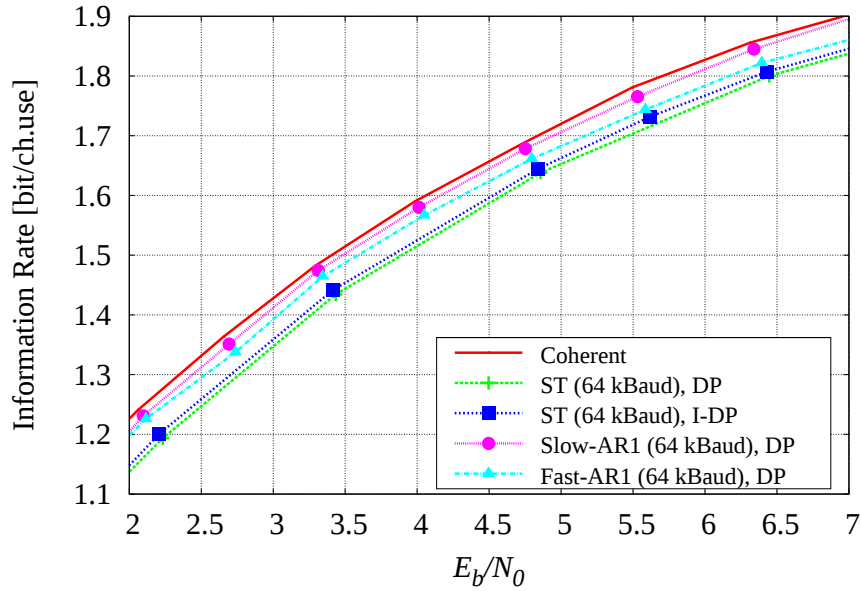
(a) E_b/N_0 from -2 to 12 dB(b) E_b/N_0 from 2 to 7 dB

Figure 2.14: Information rate for a quaternary modulation with 2-RC $h = 1/5$ and DP and I-DP detectors.

all, by comparing the IR curves obtained by the DP detector operating over a channel affected by just one PN component (with $D_a = 157$ for the slow component and $D_b = 19$ for the fast one), we derive that while the slow component is perfectly tracked by the DP detector (and thus its IR is the same of the coherent case), the curve related to the fast component is slightly separated from the coherent one. Hence we derive that fast AR1 PN process is harder to be tracked, and it is the only cause of the the 0.2 dB of IR loss of D-DP detector, remarked in Fig. 2.13.

If we employ a simple DP algorithm (with $D = 157$ quantization levels, $N_I = 10^{-1}$ and $a = 1 - 18 \cdot 10^{-6}$ and $\sigma_a^2 = 2.1625$) on the double-AR1 PN channel, the IR loss with respect to the coherent IR is approximately double of the IR constrained to the D-DP, as depicted in Fig. 2.15. Moreover, IR constrained to the DP degrades at large IR values, and the loss becomes almost 1 dB. Thus, passing from a complex D-DP detector to a simpler DP detector, we expect a BER performance degradation. We take into account the I-DP algorithm (with $D_a = 157$) and also in that case it does not improve the DP performance.

Finally in Fig. 2.16 and 2.17, we present the same curves of Fig. 2.14 for two different symbol rate frequencies, 256 and 2048 kBaud. What is relevant is the case at 256 kBaud, where we can see that the IR degradation related to DP detection of just the fast component, is greater than its counterpart at the 64 kBaud (of Fig. 2.14). This is an unexpected result, since in general the phase noise impairment is stronger at lower baudrates.

2.5 Capacity Improvement: Shaper-Precoder Optimization

In previous Sections, we derived algorithms to compute IR for CPM signals by assuming information symbols $\{x_n\}$ independent and with the same probability, so that the a priori probability $P(\mathbf{x}_0^{N-1})$ of the information sequence $\mathbf{x}_0^{N-1}$

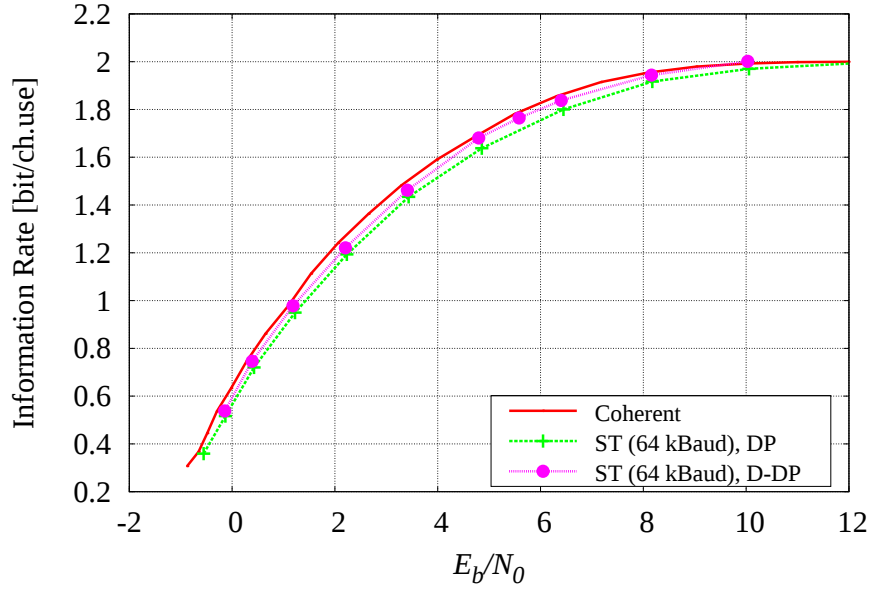
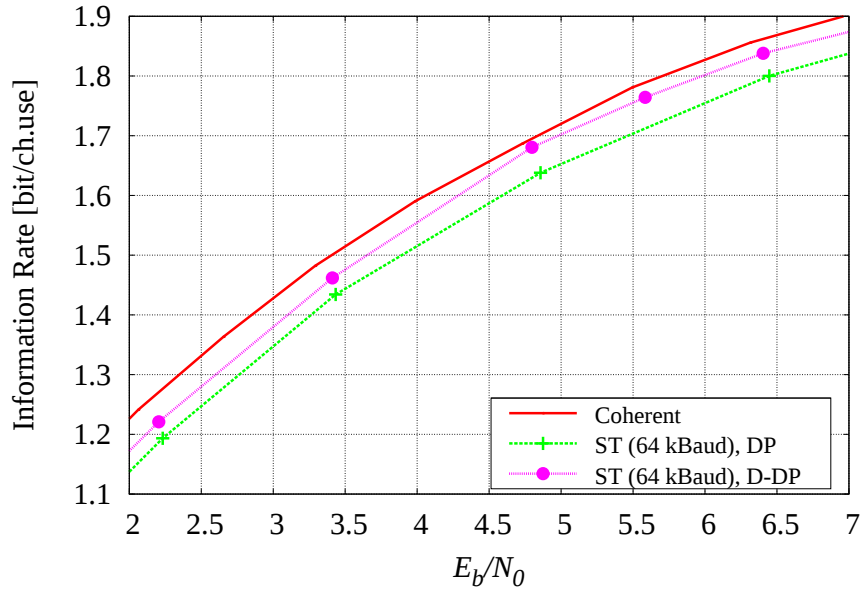
(a) E_b/N_0 from -2 to 12 dB(b) E_b/N_0 from 2 to 7 dB

Figure 2.15: Information rate for a quaternary modulation with 2-RC $h = 1/5$: comparison between D-DP and DP detectors.

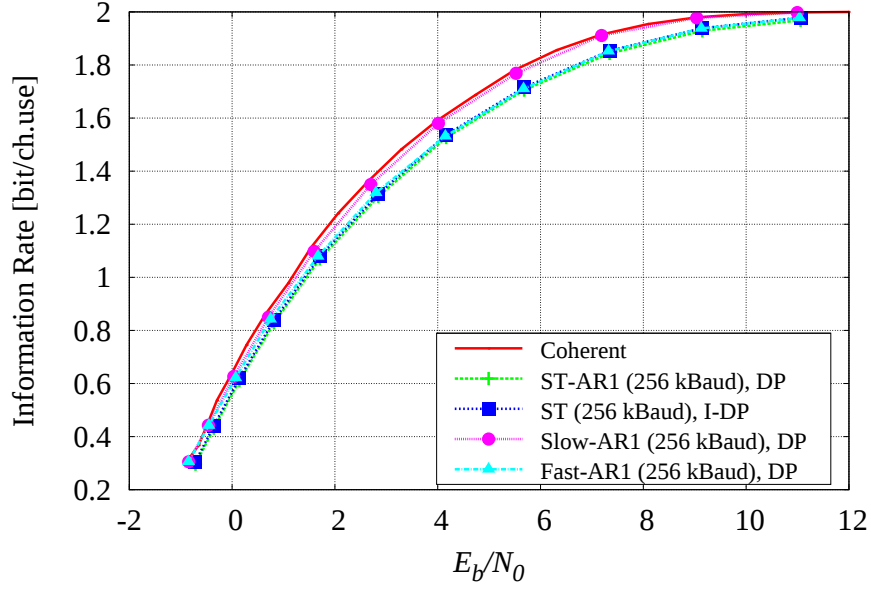
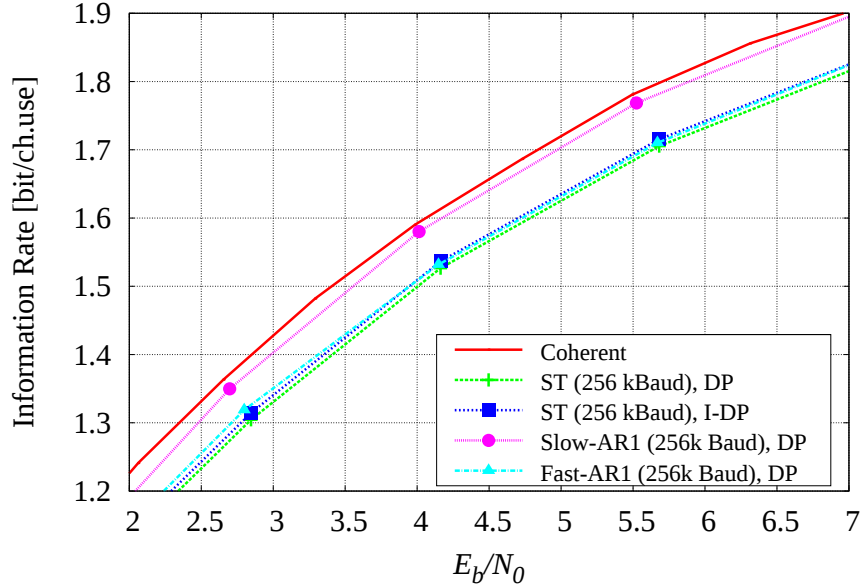
(a) E_b/N_0 from -2 to 12 dB(b) E_b/N_0 from 2 to 7 dB

Figure 2.16: Information rate for a quaternary modulation with 2-RC $h = 1/5$ and DP and I-DP detectors at 256 kBaud.

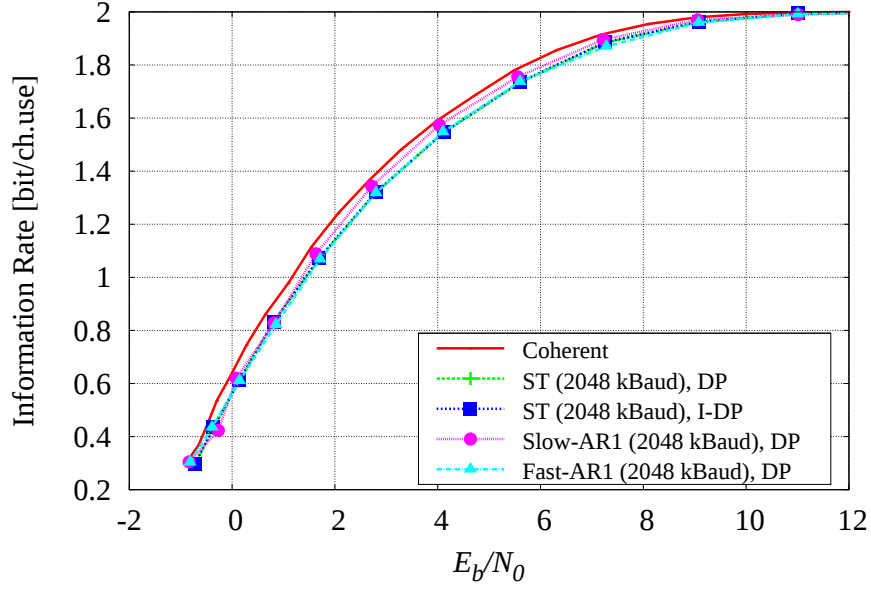
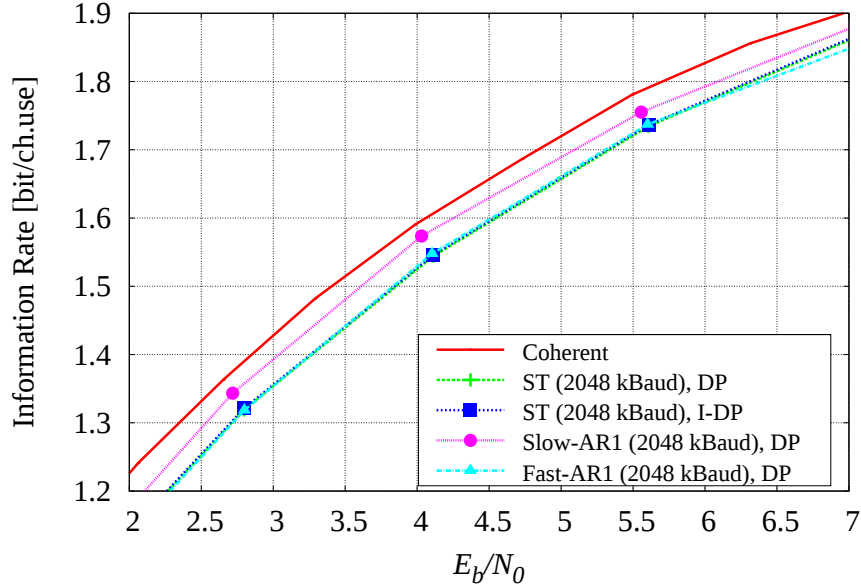
(a) E_b/N_0 from -2 to 12 dB(b) E_b/N_0 from 2 to 7 dB

Figure 2.17: Information rate for a quaternary modulation with 2-RC $h = 1/5$ and DP and I-DP detectors at 2048 kBaud.

can be factorized as

$$P(\mathbf{x}_0^{N-1}) = \prod_{k=0}^{N-1} P(x_k) = \frac{N}{M} \quad (2.54)$$

where M is the symbol alphabet cardinality. Hence, by dividing the IR results by the signal normalized bandwidth BW (normalized to the symbol rate $R \triangleq 1/T$), we obtain a measure of spectral efficiency η defined as

$$\eta \triangleq \frac{I(\mathbf{x}; \mathbf{y})}{BW} \left[\frac{b/s}{Hz} \right] \quad (2.55)$$

for independent and uniformly distributed (i.u.d.) symbols. The spectral efficiency value we have derived until now cannot be considered as the capacity for a CPM signal, since in order to evaluate the capacity, we need to maximize the spectral efficiency with respect to probability distribution of the input symbols. Hence, spectral efficiency results obtained until now, are the i.u.d. capacity $C_{i.u.d.}$, also known as symmetric information rate (SIR). In other words, $C_{i.u.d.}$ is defined as the spectral efficiency when the CPM modulator inputs are independent and uniformly distributed random variables. Of course, SIR is an (achievable) lower bound of the real capacity.

Furthermore, to be precise the bandwidth occupancy of any CPM signal is infinite, albeit in general its power spectral density rapidly goes to zero for large frequency values. Hence, some approximate definitions for the bandwidth of CPM signals were introduced in the literature. Traditionally, bandwidth is defined via the spectral power concentration. Under this approach, bandwidth is defined as the minimum number of Hertz carrying a given fraction of the total power (e.g., 99%). Another definition which was shown to possess good accuracy as well as mathematical tractability was the so-called Carson's rule bandwidth, originally developed for frequency modulations (FMs) [31] and extended to CPMs in [17].

In [17] the authors evaluated the SIR for CPM and plotted it in bps/Hz taking into account the bandwidth defined by the Carson's rule. By observing the results of [17], an interesting conclusion is that the high-SNR asymptotic SIR of CPMs approaches zero as the alphabet size goes to infinity, given that all

the other parameters describing the CPM format are kept constant. Though, at a first sight, this result could be misleading, in [18] it is shown that this is due to the sub-optimality of the uniformly distributed input when the alphabet size is large. Indeed, it is shown that, asymptotically, a much larger but still achievable lower bound on the capacity of CPMs can be obtained by considering a correctly shaped input. In particular, it is demonstrated that Gaussian shaping is asymptotically optimal. We will indicate the resulting lower bound as the independent and asymptotically optimally distributed (i.a.o.d.) capacity $C_{i.a.o.d.}$. In [18] it is also shown that, at least in the absence of thermal noise, $C_{i.a.o.d.}$ is strictly increasing with the alphabet size.

In this Section we want to overcome the difficulties in computing the spectral efficiency maximization required for capacity evaluation, for correlated input symbol. We consider here the Markov capacity [32], defined as the information rate obtained when the modulator input is a finite-state Markov sequence. In particular, we take into account input sequences whose state is defined through a shift register accumulating the last $Q \geq 1$ input symbols, Q being a design parameter. In [23], an efficient iterative method, denoted as “generalized Blahut-Arimoto algorithm”, which usually converges to the probability distribution of the input FSM transitions maximizing information rate, was proposed. We generalize this method to take into account the bandwidth also, and we apply it to CPMs. As a result, a new achievable lower bound on the CPM capacity, denoted as C_Q , being Q the above mentioned input memory parameter, will be obtained. Despite C_Q is still a lower bound of the true capacity, it always outperforms the previous definitions $C_{i.u.d.}$ and $C_{i.a.o.d.}$ and we can argue that it is arbitrarily close to the true capacity for large enough Q .

2.5.1 Problem Formulation

We recall the expression of the complex envelope of a CPM signal (1.1)

$$s(t; \mathbf{x}) = \sqrt{\frac{2E_S}{T}} \exp \left\{ j2\pi h \sum_{n=0}^{N-1} x_n q(t - nT) \right\}$$

and we refer to the Section 1.2 for explanation of the adopted notation.

Assuming that the signal (1.1) is transmitted over an AWGN channel, as described in Section 2.3.1, MAP symbol detection can be carried out on a trellis, whose state at time n is defined as $\sigma_n = (\omega_n, \pi_n)$, see (1.18). As mentioned a sufficient statistic for detection is the output of a bank of M^L matched filters (2.12). We collect the output samples of these filters in the above mentioned vector $\mathbf{y}$.

Depending on the considered capacity definition, transmitted symbols $\{x_n\}$ can be independent or correlated, with a given distribution. In our Markov capacity derivation, we will always assume that $\{x_n\}$ is a Markov sequence of memory Q , hence

$$P(x_n | x_0, \dots, x_{n-1}) = P(x_n | \mathbf{x}_{n-Q}^{n-1}). \quad (2.56)$$

Under the above mentioned assumption, a supertrellis as the one in [32] can be built including the source and the CPM modulator, whose state is $s_n = (\mathbf{x}_{n-G}^{n-1}, \phi_n)$, being $G \triangleq \max\{L-1, Q\}$.

In order to formulate the problem and the proposed optimization algorithm, we now introduce some definitions and assumptions:

- the input sequence $\{x_n\}$ and the state sequence $\mathbf{s} = \{s_n\}$ are stationary and ergodic processes;
- after having defined an integer representation for the pM^G supertrellis states, we denote $\mu_s(i) = P(s_n = i)$, $P_s(i, j) = P(s_{n+1} = j | s_n = i)$ and $Q_s(i, j) = P(s_n = i, s_{n+1} = j)$; i and j range in $0, \dots, pM^G - 1$ and we indicate with τ_s the set of compatible (i, j) ;⁵

⁵Clearly, $\mu_s(i)$ and $P_s(i, j)$ can be expressed as a function of the $Q_s(\cdot, \cdot)$ values.

- similarly, we denote by $\mu_x(n)$, $P_x(n, m)$ and $Q_x(n, m)$, respectively, the invariant probability, transition probability and joint probability of the input Markov chain state $\mathbf{x}_{n-Q}^{n-1}$. τ_x represents the set of compatible (n, m) ;
- we assume that the phase state ϕ_n is independent of the shift register part of the state s_n . Moreover, we assume that the phase state is uniformly distributed. Hence

$$P(\phi_n, \mathbf{x}_{n-G}^{n-1}) = P(\phi_n | \mathbf{x}_{n-G}^{n-1}) P(\mathbf{x}_{n-G}^{n-1}) = \frac{1}{p} P(\mathbf{x}_{n-G}^{n-1}).$$

From the above assumptions it turns out that the joint probability $Q_s(i, j)$ can be represented as a function of the input statistics only.

Since $\mathbf{x}$ is in one-to-one correspondence with the sequence of the supertrellis states, the mutual information between the input $\mathbf{x}$ and the output $\mathbf{y}$ can be equivalently expressed as the mutual information between the state sequence $\mathbf{s}$ and the output $\mathbf{y}$. Hence, the aim of this work is to optimize the input characteristics in order to maximize the overall spectral efficiency

$$C_Q = \max_{P_x(n, m)} \frac{I(\mathbf{s}; \mathbf{y})}{BW} \quad (2.57)$$

being $I(\mathbf{s}; \mathbf{y})$ the mutual information between sequences $\{s_n\}$ and $\{y_n\}$ and BW the normalized CPM bandwidth according to a given bandwidth definition. It is worth noting that the bandwidth also depends on the input distribution, hence it is inherently embedded in the maximization step⁶. Indeed, in this case we will show that it can be expressed as a function of the input process autocorrelation, which in turn depends on the joint pmf $Q_x(n, m)$.

Maximization in (2.57) turns out to be a hard task. A similar problem was solved in [23] for inter-symbol interference channels, but without taking into account the signal bandwidth modification due to input correlation. We pursued the same approach here, eventually finding the optimization algorithm 1.

Algorithm 1 Evaluation of Markov capacity C_Q

Initialization: pick arbitrary distributions $Q_x(n, m)$ and $\mu_x(n)$ respecting constraints (2.60).

repeat

Expectation step. Compute $\hat{T}_s(i, j)$ by means of the algorithm in [21], according to the input distribution Q_x .

Maximization step. Update Q_x and μ_x by solving the optimization problem of eq. (2.59) subject to the linear constraints (2.60).

until convergence

The algorithm requires the evaluation of the following quantities

$$T_s(i, j) = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{n=0}^{N-1} E \left[\log \frac{P(s_n = i, s_{n+1} = j | \mathbf{y})^{\frac{P(s_n = i, s_{n+1} = j | \mathbf{y})}{\mu_s(i) P_s(i, j)}}}{P(s_n = i | \mathbf{y})^{\frac{P(s_n = i | \mathbf{y})}{\mu_s(i)}}} \right] \quad (2.58)$$

where $E[\cdot]$ denotes expectation. An approximation of the above quantities can be evaluated by means of a Monte Carlo average and the algorithm in [21]. We will denote these approximated values as $\hat{T}_s(i, j)$.

The algorithm iteratively performs two steps: an expectation step, in which $\hat{T}_s(i, j)$ are evaluated according to the input distribution described by Q_x and a maximization step, in which the following constrained maximization is carried out

$$\arg \max_{Q_x(\cdot, \cdot), \mu_x(\cdot)} \frac{\sum_{(i, j) \in \tau_s} Q_s(i, j) \left(\hat{T}_s(i, j) - \log \frac{Q_s(i, j)}{\sum_k Q_s(i, k)} \right)}{BW(Q_x(\cdot, \cdot), \mu_x(\cdot))} \quad (2.59)$$

⁶ A bandwidth definition which ensures a mathematical tractability is that given by the Carson's rule described in the next Section and also considered in the numerical results.

subject to

$$\begin{aligned}
0 &\leq Q_x(n, m) \leq 1, \forall n, m \\
Q_x(n, m) &> 0 \quad \text{iff } (n, m) \in \tau_x \\
0 &\leq \mu_x(n) \leq 1, \forall n \\
\mu_x(k) &= \sum_n Q_x(n, k) = \sum_m Q_x(k, m), \forall k \\
\sum_k \mu_x(k) &= 1.
\end{aligned} \tag{2.60}$$

It is worth noting that, with respect to [23] in which the maximization step turns out to be an eigenvalue problem and hence can be carried out in closed-form, no closed-form solutions exist for the case at hand due to the presence of the denominator in (2.59). However, since the above constraints are linear, a linearly-constrained non-linear optimization algorithm, based on the projected gradient iterative method, is carried out to solve (2.59). Partial derivatives of the argument of (2.59) will be in part derived in the next Section, for the (properly modified) Carson's bandwidth. It is important to note that the derivatives of each $Q_s(i, j)$ with respect to each $Q_x(n, m)$ and $\mu_x(n)$ can be easily derived under the considered assumptions.

As a final remark, we would like to point out that we neither ensure the global convergence of the proposed iterative method, nor formulate theorems on its local convergence behavior. However, in all the simulations carried out the proposed algorithm was able to reach convergence in a few iterations.

2.5.2 The Carson's Bandwidth for Correlated Input

The one-sided Carson's bandwidth is defined for CPM signals⁷ as [17]

$$B = 2(\beta + \mathcal{K})f_m \tag{2.61}$$

⁷With respect to [17], a parameter $\mathcal{K}$ was introduced here, (similarly to what done in [31] for FM modulations, to better represent the CPM signal. In the original formula of [17] it was $\mathcal{K} = 1$.

where, denoting by $m(t)$ the signal

$$m(t) = \sum_n x_n u(t - nT)$$

f_m is the one-sided effective bandwidth of $m(t)$ and the parameter β is the so-called *effective FM modulation index* given by $\beta = h\sqrt{P_m}/f_m$. We can express the parameter f_m as [17]

$$f_m = \frac{P_m}{2 S_m(0)} \quad (2.62)$$

being P_m the power of $m(t)$ and $S_m(f)$ its PSD, given by

$$S_m(f) = \frac{S_x(f)}{T} |U(f)|^2 \quad (2.63)$$

whereas

$$P_m = \frac{1}{T^2} \sum_{n=-\infty}^{+\infty} R_x(n) p(nT) \quad (2.64)$$

with

$$p(t) = T u(t) \otimes u(-t)$$

with support in the interval $[-LT, LT]$. In (2.63), $U(f)$ is the Fourier transform of $u(t)$ and $S_x(f)$ denotes the PSD of the transmitted symbols

$$S_x(f) = \sum_{n=-\infty}^{+\infty} R_x(n) e^{-j2\pi n f T} \quad (2.65)$$

being $R_x(n) = E\{x_k x_{k-n}\}$ the autocorrelation function of $\{x_n\}$.

Using the above expressions, we have

$$f_m = \frac{2}{T} \frac{\sum_{n=-L}^L R_x(n) p(nT)}{\sum_{n=-\infty}^{+\infty} R_x(n)} \quad (2.66)$$

$$\beta = \frac{h}{2} \frac{\sum_{n=-\infty}^{+\infty} R_x(n)}{\sqrt{\sum_{n=-L}^L R_x(n) p(nT)}} \quad (2.67)$$

from which it follows that

$$BT = 2h \sqrt{\sum_{n=-L}^L R_x(n) p(nT)} + \mathcal{K} \frac{4 \sum_{n=-L}^L R_x(n) p(nT)}{\sum_{n=-\infty}^{+\infty} R_x(n)}. \quad (2.68)$$

The Carson's normalized bandwidth in (2.68) depends on the CPM parameters (phase pulse and modulation index) as well as on the input symbol autocorrelation. Since the input we are considering is an aperiodic irreducible Markov chain, it turns out that [33]

$$R_x(n) = \begin{cases} \mathbf{a}(n)^T \text{diag}(\boldsymbol{\mu}_x) \mathbf{a}(0) & 0 \leq n \leq Q-1 \\ \mathbf{a}(Q-1)^T \text{diag}(\boldsymbol{\mu}_x) \mathbf{P}_x^{n-Q+1} \mathbf{a}(0) & n \geq Q \end{cases} \quad (2.69)$$

where $\boldsymbol{\mu}_x$ and $\mathbf{P}_x$ are a vector and a matrix collecting the probability mass functions $\mu_x(n)$ and $P_x(n, m)$, respectively. Matrix $\text{diag}(\boldsymbol{\mu})$ is a diagonal matrix whose entries in the main diagonal are the element of $\boldsymbol{\mu}$. $\mathbf{a}(n)$ is a vector $[a_0(n), a_1(n), \dots, a_{M^Q-1}(n)]$ where we denoted with

$$a_\ell(n), \quad \ell = 0, \dots, M^Q - 1, \quad n = 0, \dots, Q - 1$$

the n -th most recent symbol of state ℓ , where ℓ encodes in a predefined way the M^Q available states.

In order to apply the optimization algorithm proposed in the previous Section, all partial derivatives $\frac{\partial BT}{\partial Q_x(\ell, m)}$ and $\frac{\partial BT}{\partial \mu_x(\ell)}$ have to be evaluated. From (2.68) and applying the composition rule for derivatives it turns easily out that

$$\frac{\partial BT}{\partial Q_x(\ell, m)} = \frac{h \sum_{n=-L}^L \frac{\partial R_x(n)}{\partial Q_x(\ell, m)} p(nT)}{\sqrt{\sum_{n=-L}^L R_x(n) p(nT)}} + \mathcal{K} \frac{4 \sum_{n=-L}^L \frac{\partial R_x(n)}{\partial Q_x(\ell, m)} p(nT)}{\sum_{n=-\infty}^{+\infty} R_x(n)} \quad (2.70)$$

$$- \mathcal{K} \frac{4 \sum_{n=-\infty}^{+\infty} \frac{\partial R_x(n)}{\partial Q_x(\ell, m)} \sum_{n=-L}^L R_x(n) p(nT)}{\left(\sum_{n=-\infty}^{+\infty} R_x(n) \right)^2}$$

$$\frac{\partial BT}{\partial \mu_x(\ell)} = \frac{h \sum_{n=-L}^L \frac{\partial R_x(n)}{\partial \mu_x(\ell)} p(nT)}{\sqrt{\sum_{n=-L}^L R_x(n) p(nT)}} + \mathcal{K} \frac{4 \sum_{n=-L}^L \frac{\partial R_x(n)}{\partial \mu_x(\ell)} p(nT)}{\sum_{n=-\infty}^{+\infty} R_x(n)} \quad (2.71)$$

$$- \mathcal{K} \frac{4 \sum_{n=-\infty}^{+\infty} \frac{\partial R_x(n)}{\partial \mu_x(\ell)} \sum_{n=-L}^L R_x(n) p(nT)}{\left(\sum_{n=-\infty}^{+\infty} R_x(n) \right)^2}.$$

Hence, the expressions of the partial derivatives of BT with respect to $\mu_x(\ell)$ and $Q_x(\ell, m)$ can be obtained from the partial derivatives of $R_x(n)$, which is the only term in (2.68) that depends on the statistical properties of the sequence $\mathbf{x}$. Therefore, we now derive the partial derivatives of $R_x(n)$. In order to highlight the fact that the autocorrelation function depends on the the probabilities $Q_x(\ell, m)$, $\mu_x(\ell)$ and $P_x(\ell, m) = \frac{Q_x(\ell, m)}{\mu_x(\ell)}$, we now denote it by

$R_x(n; \mathbf{P}_x(\mathbf{Q}_x, \boldsymbol{\mu}_x), \boldsymbol{\mu}_x)$, being $\mathbf{Q}_x, \mathbf{P}_x$ and $\boldsymbol{\mu}_x$ two matrices and a vector collecting the probabilities $Q_x(\ell, m), P_x(\ell, m)$ and $\mu_x(\ell)$ respectively. It turns out that

$$\begin{aligned} \frac{\partial R_x(n; \mathbf{P}_x(\mathbf{Q}_x, \boldsymbol{\mu}_x), \boldsymbol{\mu}_x)}{\partial \mu_x(\ell)} &= \sum_i \sum_j \frac{\partial R_x(n; \mathbf{P}_x(\mathbf{Q}_x, \boldsymbol{\mu}_x), \boldsymbol{\mu}_x)}{\partial P_x(i, j)} \frac{\partial P_x(i, j)}{\partial \mu_x(\ell)} \\ &\quad + \sum_i \frac{\partial R_x(n; \mathbf{P}_x(\mathbf{Q}_x, \boldsymbol{\mu}_x), \boldsymbol{\mu}_x)}{\partial \mu_x(i)} \frac{\partial \mu_x(i)}{\partial \mu_x(\ell)} \\ &= \sum_j \frac{\partial R_x(n; \mathbf{P}_x(\mathbf{Q}_x, \boldsymbol{\mu}_x), \boldsymbol{\mu}_x)}{\partial P_x(\ell, j)} \left(-\frac{Q_x(\ell, j)}{\mu_x(\ell)^2} \right) + \frac{\partial R_x(n; \mathbf{P}_x(\mathbf{Q}_x, \boldsymbol{\mu}_x), \boldsymbol{\mu}_x)}{\partial \mu_x(\ell)} \end{aligned} \quad (2.72)$$

$$\begin{aligned} \frac{\partial R_x(n; \mathbf{P}_x(\mathbf{Q}_x, \boldsymbol{\mu}_x), \boldsymbol{\mu}_x)}{\partial Q_x(\ell, m)} &= \sum_i \sum_j \frac{\partial R_x(n; \mathbf{P}_x(\mathbf{Q}_x, \boldsymbol{\mu}_x), \boldsymbol{\mu}_x)}{\partial P_x(i, j)} \frac{\partial P_x(i, j)}{\partial Q_x(\ell, m)} \\ &\quad + \sum_i \frac{\partial R_x(n; \mathbf{P}_x(\mathbf{Q}_x, \boldsymbol{\mu}_x), \boldsymbol{\mu}_x)}{\partial \mu_x(i)} \frac{\partial \mu_x(i)}{\partial Q_x(\ell, m)} \\ &= \frac{1}{\mu_x(\ell)} \frac{\partial R_x(n; \mathbf{P}_x(\mathbf{Q}_x, \boldsymbol{\mu}_x), \boldsymbol{\mu}_x)}{\partial P_x(\ell, m)} \end{aligned} \quad (2.73)$$

where

$$\frac{\partial R_x(n)}{\partial \mu_x(\ell)} = \begin{cases} [\mathbf{a}(n) \circ \mathbf{a}(0)]_\ell & 0 \leq n < Q \\ \left[\mathbf{a}(Q-1) \circ \left(\mathbf{P}_x^{n-Q+1} \cdot \mathbf{a}(0) \right) \right]_\ell & n \geq Q \end{cases} \quad (2.74)$$

being $\circ$ is the Hadamard product. Similarly, it follows that⁸

$$\frac{\partial R_x(n)}{\partial P_x(\ell, m)} = \begin{cases} 0 & 0 \leq n < Q \\ \left[\sum_{r=0}^{n-Q} (\mathbf{P}_x^r)^T \text{diag}(\boldsymbol{\mu}_x) \mathbf{a}(Q-1) \mathbf{a}(0)^T (\mathbf{P}_x^{n-Q-r})^T \right]_{\ell, m} & n \geq Q. \end{cases} \quad (2.76)$$

Remark. In the expression of the Carson's normalized bandwidth, in (2.68), an infinite summation on the elements of the autocorrelation function $R_x(n)$ appears. However, it can be proved [33, 35] that the autocorrelation is dominated by a geometrically decreasing sequence, hence there exist constants $c > 0$ and $0 < r < 1$ such that $|R_x(n)| < cr^n$. Therefore, the infinite summation is well approximated by truncating it to N' terms, for a suitably large N' .

2.5.3 Numerical Results

Bandwidth Considerations

First of all, we evaluate the effectiveness of the considered bandwidth definitions (those based on the spectral power concentration as well as those obtained by modifying the Carson's rule) to describe the communication system at hand. We consider a frequency division multiplexing (FDM) transmission of three independent CPM signals, with the same parameters but with independent data and timing offset. We assume that the receiver is interested in the central signal only and that the two interfering signals are placed at a distance from the central one given by the bandwidth definition. Neither the transmitter shaping filter nor the receiver front end are changed.

The evaluation of bit error rate of the main signal in this scenario is useful to assess the effectiveness of the assumed bandwidth definitions to describe the

⁸Equations (2.74) and (2.76) were obtained by means the following well-known result [34]

$$\frac{\partial}{\partial \mathbf{X}} \mathbf{a}^T \mathbf{X}^n \mathbf{b} = \sum_{r=0}^{n-1} (\mathbf{X}^r)^T \mathbf{a} \mathbf{b}^T (\mathbf{X}^{n-1-r})^T \quad (2.75)$$

where $\mathbf{a}$ and $\mathbf{b}$ are column vectors of a given size k and $\mathbf{X}$ is a non-singular square matrix of size k .

system at hand. In Fig. 2.18 two cases are considered, namely an octal CPM format with $L = 2$, raised-cosine frequency pulse (2-RC), $h = 1/7$, and a payload size of 880 information bits, and a quaternary 3-RC modulation with $h = 2/7$ and a payload size of 1760 in information bits. In both cases, a SCCPM system is considered, where the outer code is a $R = 1/2$ convolutional code (CC) with generators $(7, 5)_8$ and 10 iterations of the overall decoder are performed. For different bandwidth definitions, the resulting performance loss due to the interference from the adjacent channels is assessed. It can be observed that some definitions highly underestimate the effective bandwidth occupancy, thus the interference is large and so is the performance loss. Among the two considered bandwidth definitions and the various parameters, two effective bandwidth measures turn out to be the Carson bandwidth with parameter $\mathcal{K} = 1.5$ and the bandwidth definition based on spectral power concentration and a power ratio of at least 99.5%.

Finally, in Fig. 2.19 we consider a quaternary CPM scheme with 2-RC and $h = 2/7$ in the same scenario, but we impose different interference conditions. Indeed, we assess the BER performance when the power of the two adjacent CPM signals is doubled with respect to the central signal—they exhibit the same PSD but a doubled bandwidth. The same conclusions deriving from Fig. 2.18 still hold for this case: the Carson bandwidth with parameter $\mathcal{K} = 1.5$ and the 99.5% bandwidth are effective bandwidth measures, while the Carson bandwidth with $\mathcal{K} = 1$, 99% and 98% bandwidths underestimate the effective bandwidth occupancy.

Iterative Optimization of the Spectral Efficiency

In this Section, the outcomes of the proposed optimization algorithm in four different scenarios are shown in terms of spectral efficiency in bps/Hz (under the Carson's bandwidth definition with parameter $\mathcal{K} = 1.5$, where not explicitly stated otherwise) versus E_b/N_0 , E_b being the received signal energy per *information* bit and N_0 the one-sided noise power spectral density.

It is important to remark that the optimization is carried out for each

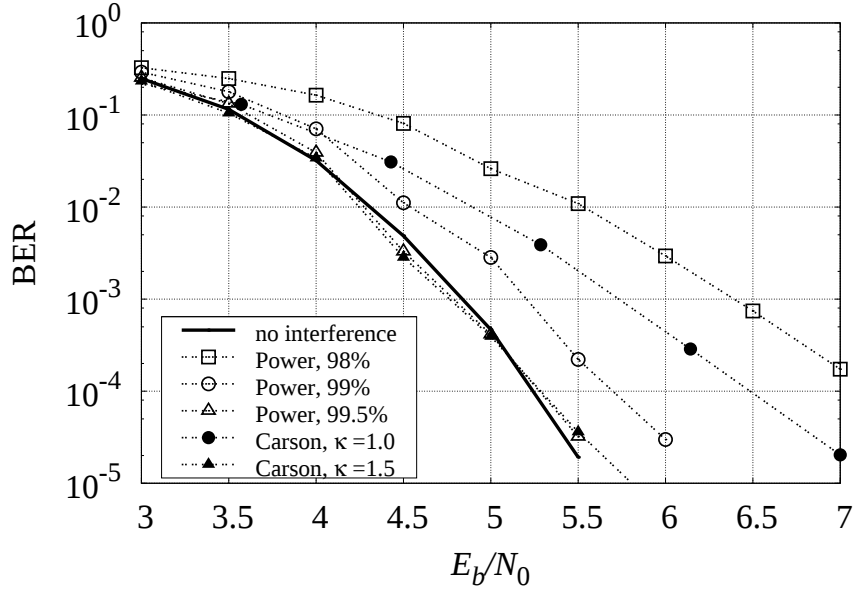
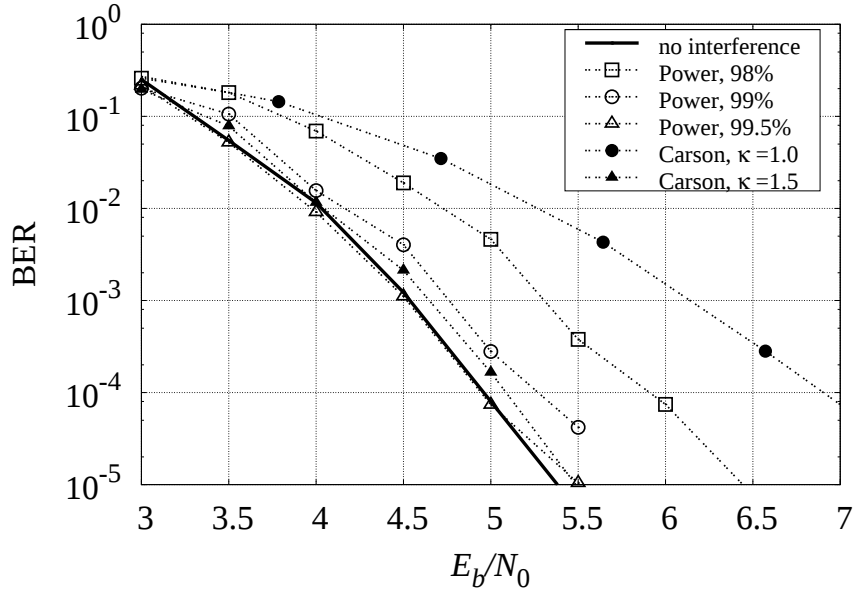
(a) Octal 2-RC, $h = 1/7$, payload size of 880 bits(b) Quaternary 3-RC, $h = 2/7$, payload size of 1760 bits

Figure 2.18: Interference due to adjacent channels in FDM systems with different bandwidth assumptions.

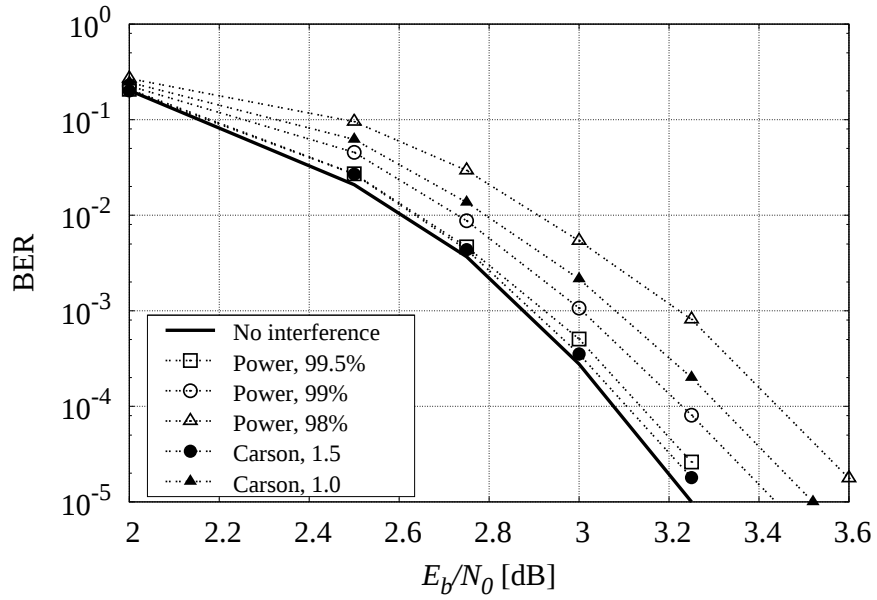


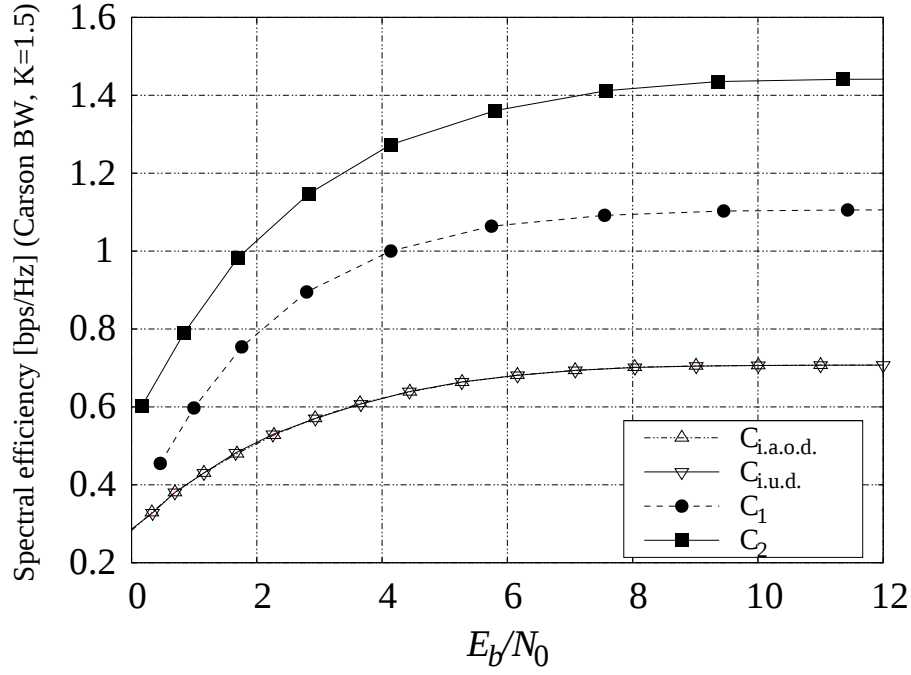
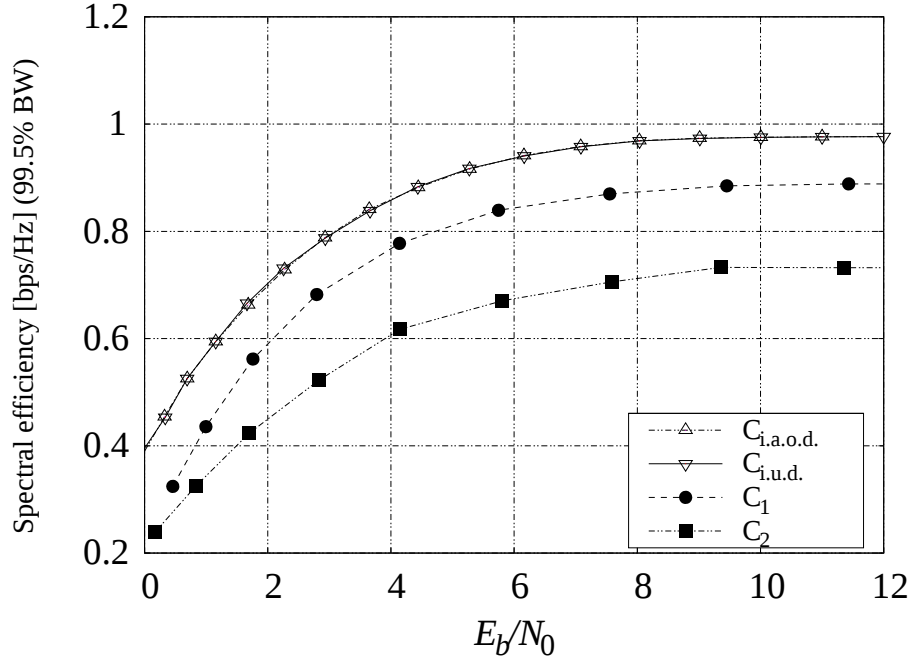
Figure 2.19: Interference due to adjacent channels in FDM systems with different bandwidth assumptions. Power and information rate of interferers are doubled respect to the CPM signal of interest.

E_S/N_0 and of course for a given E_S/N_0 Markov capacity C_Q is strictly larger than $C_{i.u.d.}$. On the contrary, for a fixed E_b/N_0 this property is not ensured and the optimized Markov capacity could be smaller than the symmetric information rate at the same E_b/N_0 . This stems from the fact that E_b/N_0 and E_S/N_0 are related by the information rate in bits per channel use (the numerator of (2.57)), which is a result of the optimization procedure.

The first considered CPM format is the binary 2-RC with a modulation index $h = 1/3$. The proposed input optimization method was carried out for 1st- and 2nd-order Markov input. In Fig. 2.20(a), beside the symmetric information rate $C_{i.u.d.}$, the capacity for independent and asymptotically optimally distributed input $C_{i.a.o.d.}$ following the technique in [18] is shown. It is important to note that for binary modulations $C_{i.a.o.d.} = C_{i.u.d.}$. Hence shaping only does not give benefits in this case. On the other hand, the same figure shows that Markov capacities C_1 and C_2 , obtained by means of the proposed optimization method, outperform $C_{i.u.d.}$, especially for medium to high E_b/N_0 values.

In Fig. 2.20(b) the same scenario is considered, but the spectral efficiency shown on the y-Axis was evaluated using the 99.5% bandwidth definition (although the optimization algorithm was carried out with the Carson's bandwidth as above). As it can be seen, optimization with respect to a given bandwidth definition does not usually give good results when a different definition is used in the spectral efficiency evaluation.

In order to give more insights on the results obtained by the proposed optimization algorithm, we analyze the optimal source transition probabilities for the above mentioned scenario at $E_b/N_0 = 4$ dB (corresponding to a spectral efficiency of 1 bps/Hz and 1.26 bps/Hz for $Q = 1$ and $Q = 2$ respectively). In Table 2.4 the optimized source transition probabilities are shown for both 1st- and 2nd-order Markov inputs. It can be observed that the optimal distributions, which are of course symmetric due to the symmetry of the original problem, tend to reduce the state transitions which are associated to an increase of the bandwidth.

(a) Carson's bandwidth ($K = 1.5$)

(b) 99.5% bandwidth

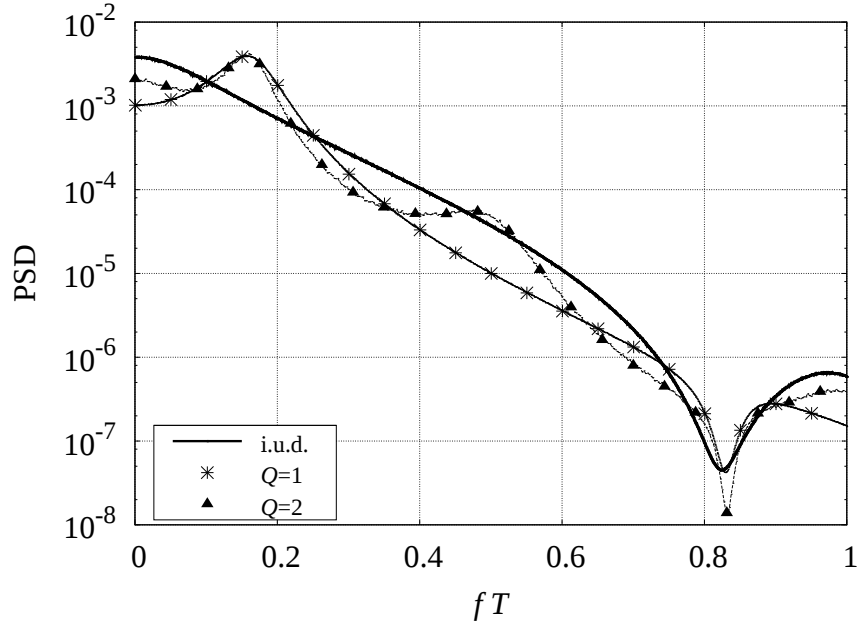
Figure 2.20: Spectral efficiency for a binary 2-RC with $h = 1/3$.

Table 2.4: Optimal Markov sources for a binary 2-RC with $h = 1/3$ at $E_b/N_0 = 4.0$ dB.

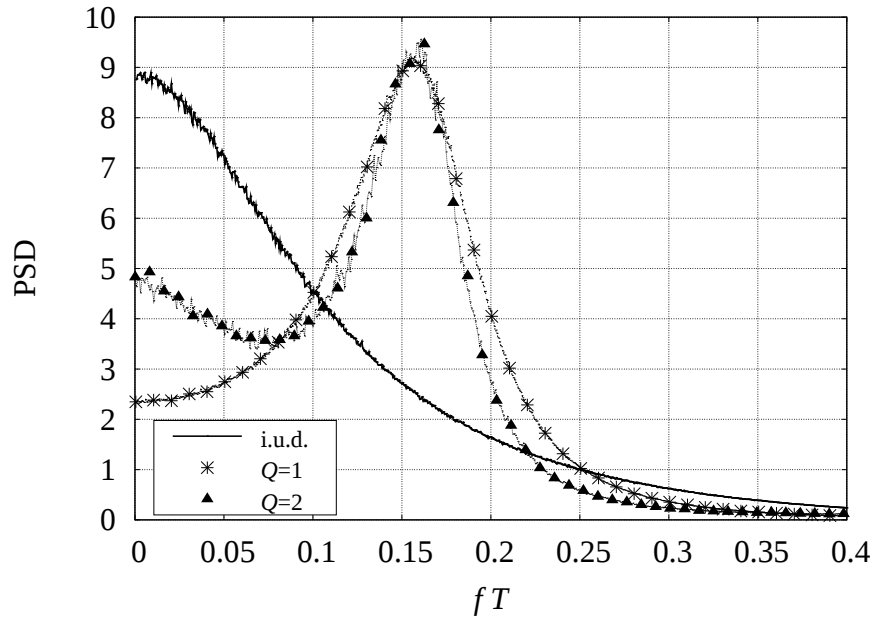
(a) 1st order			(b) 2nd order				
(x_{n-1})	0	1	(x_{n-2}, x_{n-1})	00	01	10	11
$x_n = 0$	.824	.176	$x_n = 0$	.855	.683	.317	.145
$x_n = 1$	.176	.824	$x_n = 1$	.145	.317	.683	.855

In Fig. 2.21, the PSD of the transmitted signal for the above mentioned 1st- and 2nd-order optimized sources is compared with the PSD of the transmitted signal with i.u.d. inputs. Due to the correlation of the input symbols $\{x_n\}$, in both cases the PSD tends to be concentrated around a peak at a normalized frequency of about $fT = 0.16$. Thanks to this power concentration property, the transmitted power allocated in the highest frequencies is much limited with respect to the i.u.d. case. In Fig. 2.22, the SPC for the same scenario is shown. It is instructive to see that, depending on the considered percentile of the spectral power bandwidth definition, different behaviors are observed. For example, for power percentile lower than 85%, the bandwidth in the i. u. d. case is lower than those of the correlated inputs, while within the range 85% and about 98% the correlated inputs lead to much lower bandwidths. Moreover, over 96% the 2nd-order input exhibits a bandwidth larger than that of the 1st-order case.

Table 2.5 shows the normalized Carson's rule bandwidth for both i.u.d. and correlated inputs. Moreover, for this scenario the normalized 90% and 99.5% bandwidths have been evaluated as well, through integration of the numerically evaluated PSDs, and this measure confirms the bandwidth reduction gained by the optimized 1st-order Markov input over the i.u.d. input. On the contrary, while the Carson's bandwidth of the 2nd-order correlated input is largely reduced with respect to the 1st-order case, its 99.5% bandwidth is larger. This is a further proof of the large differences between the considered bandwidth definitions, and of the fact that the optimization with respect to a bandwidth

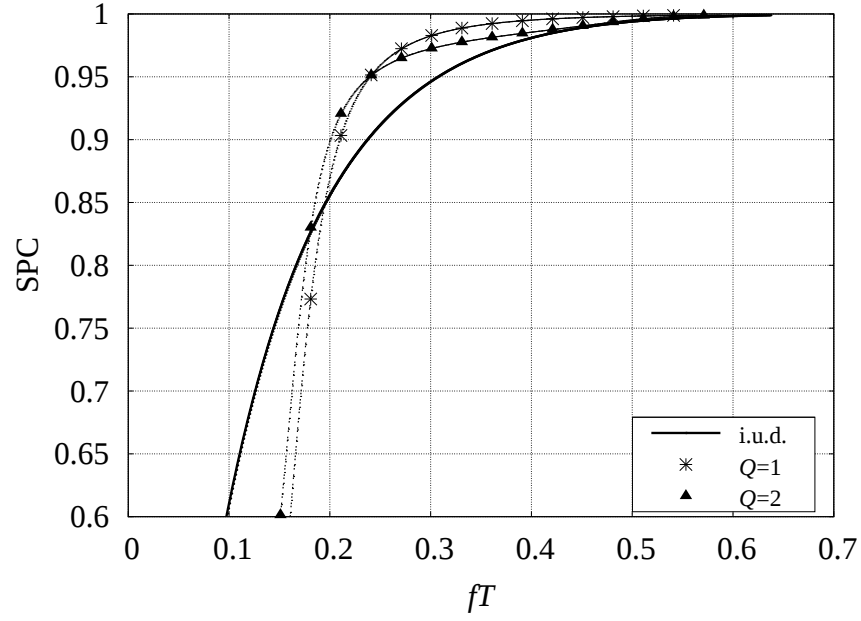


(a) Logarithmic scale



(b) Linear scale, zoomed

Figure 2.21: PSD of the CPM signal with i.u.d. inputs and with the optimized 1st- and 2nd-order Markov inputs described in Table 2.4.

Figure 2.22: Spectral Power Concentration for a binary 2-RC with $h = 1/3$.Table 2.5: Normalized bandwidth (under different definitions) for the i. u. d. case and the optimized correlated inputs for a binary 2-RC with $h = 1/3$.

	Carson ($\mathcal{K} = 1.5$)	90%	99.5%
i. u. d.	1.41	0.478	1.03
$Q = 1$	0.61	0.42	0.80
$Q = 2$	0.48	0.40	1.00

definition does not usually give good results using a different definition.

It is interesting to note that in all simulations we carried out, the largest impact of proposed optimization method is on the bandwidth rather than on the information rate, i.e., we observed that typically the optimized source leads to a possibly large reduction of the denominator of (2.57), rather than an increase of the numerator.

An interesting incidental result of the peakedness of the PSD after the input optimization is that it can help the carrier phase synchronization. Since it turns out that in CPM based communication systems phase synchronization could be one of the hardest tasks to be carried out at the receiver [36], this is a further benefit of employing a properly correlated input optimized with the proposed algorithm.

Finally, in Fig. 2.23 the autocorrelation function $R(\tau) = E\{s(t; \mathbf{x})s^*(t - \tau; \mathbf{x})\}$ for the CPM signals with i. u. d. and correlated inputs is considered. It is worth noting that for correlated inputs the autocorrelation function exhibits sidelobes, whereas the main lobe is tighter than in the i. u. d. case. This fact could have an impact also in the timing and frame synchronization parts of the receiver if performed following a non data-aided (NDA) technique. Obviously the autocorrelation function for correlated inputs is not significant if timing and frame synchronization techniques are data-aided (DA), since pilot symbols in the preamble are not precoded. In such a case, it is sufficient to properly design the symbols sequence in the preamble to avoid a bandwidth increase with respect to the payload part of the burst.

We now consider another quaternary modulation format, 3-RC with $h = 2/7$. In Fig. 2.24, the spectral efficiency curves are shown and considerations similar to the previous cases can be carried out. We analyze the optimal source transition probabilities for the above mentioned scenario at about $E_b/N_0 = 2.2$ dB (corresponding to a spectral efficiency of roughly 1.6 bps/Hz for $Q = 1$). From Table 2.6, which shows the optimized source transition probabilities for a 1st-order Markov input, it is clear that the state transitions associated to an increase of the bandwidth are not encouraged. In Fig. 2.25(a) it can be seen

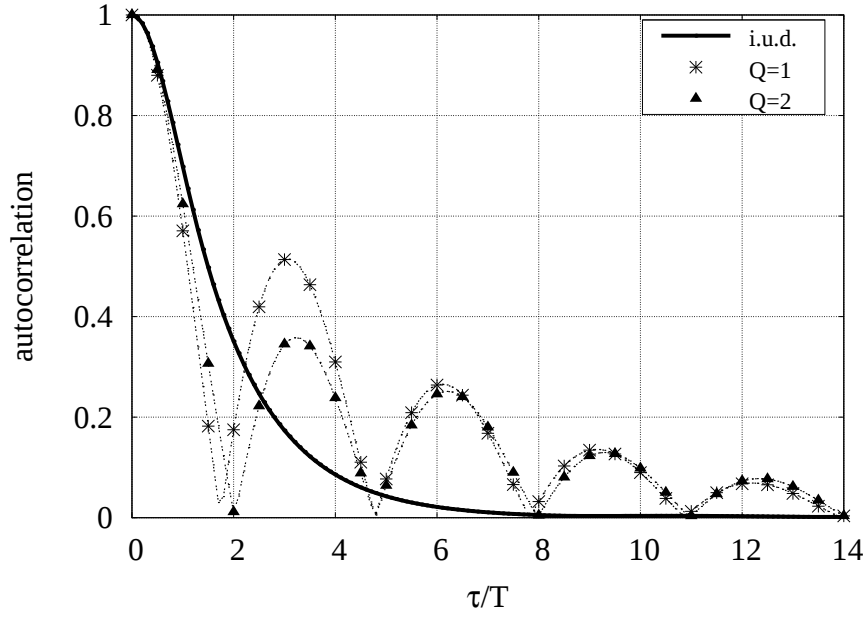
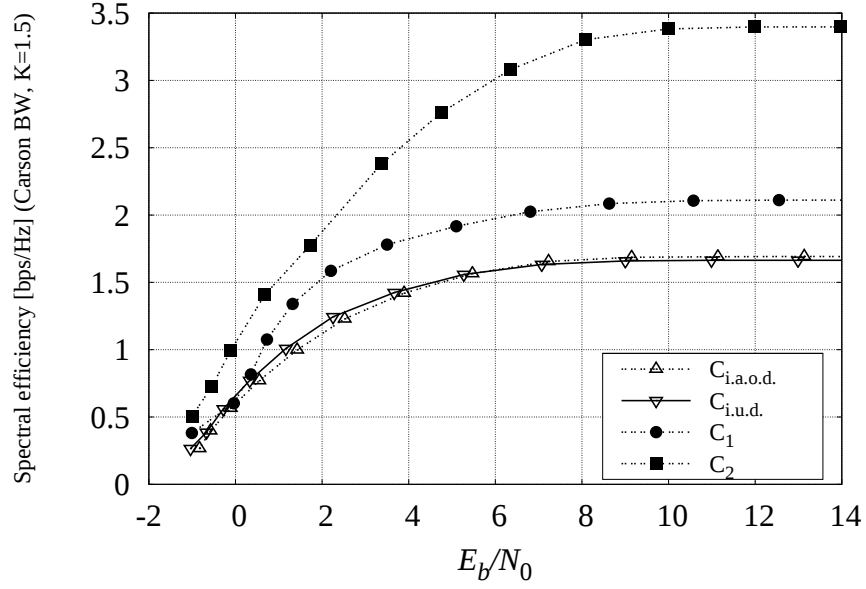
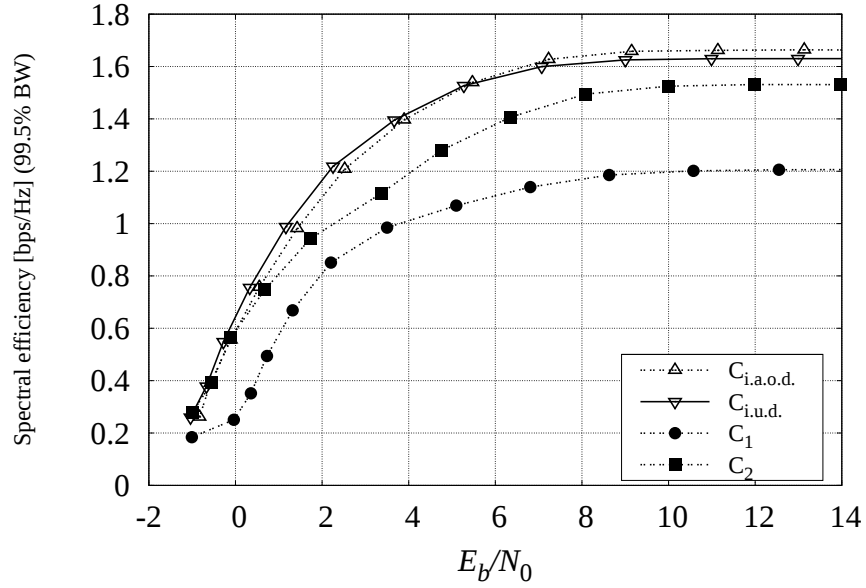


Figure 2.23: Autocorrelation function of the CPM transmitted signal for a binary 2-RC with $h = 1/3$.

Table 2.6: Optimal 1st-order Markov source for a quaternary 3-RC with $h = 2/7$ at $E_b/N_0 \simeq 2.2$ dB.

(a) Transition probabilities ($P_x(i, j)$)				
(x_{n-1})	0	1	2	3
$x_n = 0$	.364	.191	.003	.008
$x_n = 1$	.619	.680	.126	.009
$x_n = 2$	.009	.126	.680	.619
$x_n = 3$	.008	.003	.191	.363
(b) State probabilities ($\mu_x(i)$)				
(x_{n-1})	0	1	2	3
	.118	.382	.382	.118

(a) Carson's bandwidth ($\mathcal{K} = 1.5$)

(b) 99.5% bandwidth

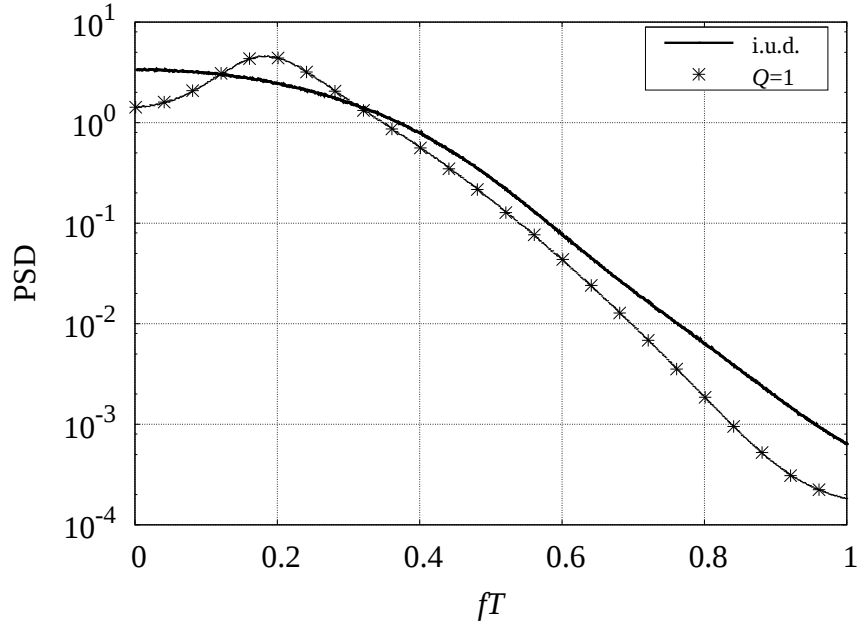
Figure 2.24: Spectral efficiency for a quaternary 3-RC with $h = 2/7$.

Table 2.7: Normalized bandwidth (under different definitions) for the i. u. d. case and the optimized correlated input for a quaternary 3-RC with $h = 2/7$.

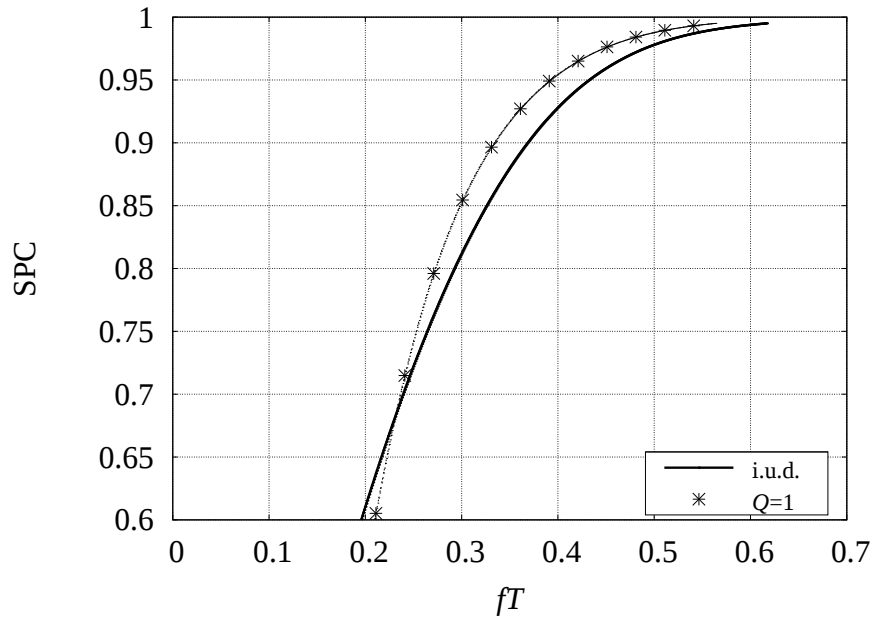
	Carson ($\mathcal{K} = 1.5$)	90%	99.5%
i. u. d.	1.20	0.74	1.24
$Q = 1$	0.60	0.67	1.13

that the correlation of the input symbols $\{x_n\}$ makes the power spectral density concentrated around a peak at a normalized frequency $fT = 0.19$. Hence, with a 1st-order Markov input optimized with respect to Carson's bandwidth, the spectral power allocated in the highest frequencies decreases with respect to the i.u.d. input as underlined by Fig. 2.25(b). However, we cannot achieve a significant reduction when a bandwidth definition based on the spectral power concentration is considered. Table 2.7 confirms this remark since while Carson's rule bandwidth significantly reduces, the 90% and 99.5% bandwidths remain unchanged when an optimized 1st-order Markov input is exploited instead of the i. u. d. input. To conclude with this modulation format, in Fig. 2.26 we show the autocorrelation functions. In this case also, for the optimized 1st-order Markov input we have some secondary lobes but also a slightly narrower principal lobe with respect to the i. u. d. input case.

Finally, in the following we take into account an octal modulation format 2-RC with $h = 1/7$. In this scenario the shaping technique proposed in [18] exhibits a limited gain at high E_b/N_0 values (see Fig. 2.27), even if the alphabet size is large. This is probably due to the fact that with a modulation index equal to $1/7$ the signal bandwidth is so reduced that, shaping the input, we cannot achieve a further significant bandwidth reduction. On the other hand, correlating the input by a 1st-order Markov chain, we obtain a large increase of the spectral efficiency. We analyze the optimal source transition probabilities for the above mentioned scenario at about $E_b/N_0 = -1$ dB (corresponding to a spectral efficiency of 0.4 bps/Hz for $Q = 1$). While in the former scenarios the



(a) Power Spectral Density



(b) Spectral Power Concentration

Figure 2.25: PSD and SPC of a CPM signal with i.u.d. input and with the optimized 1st-order Markov input described in Table 2.6.

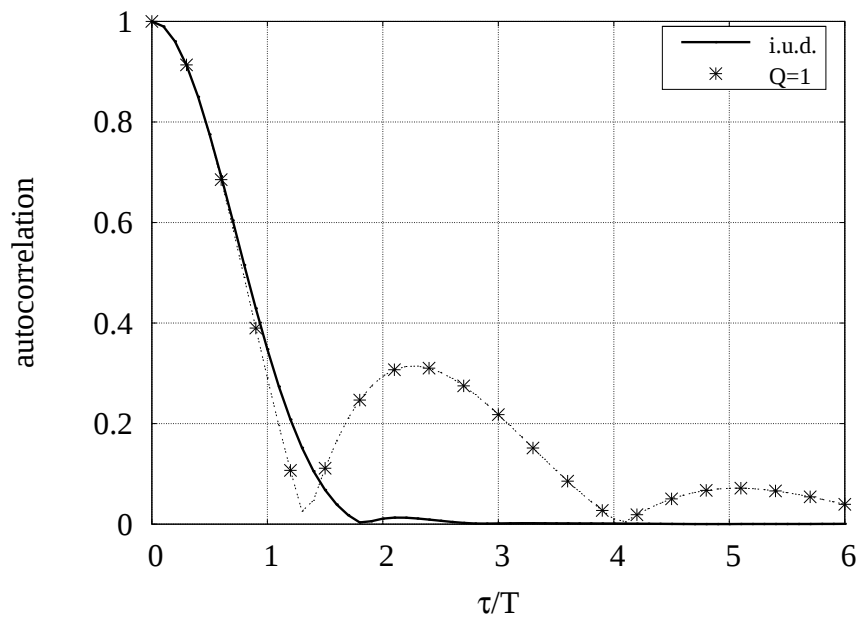
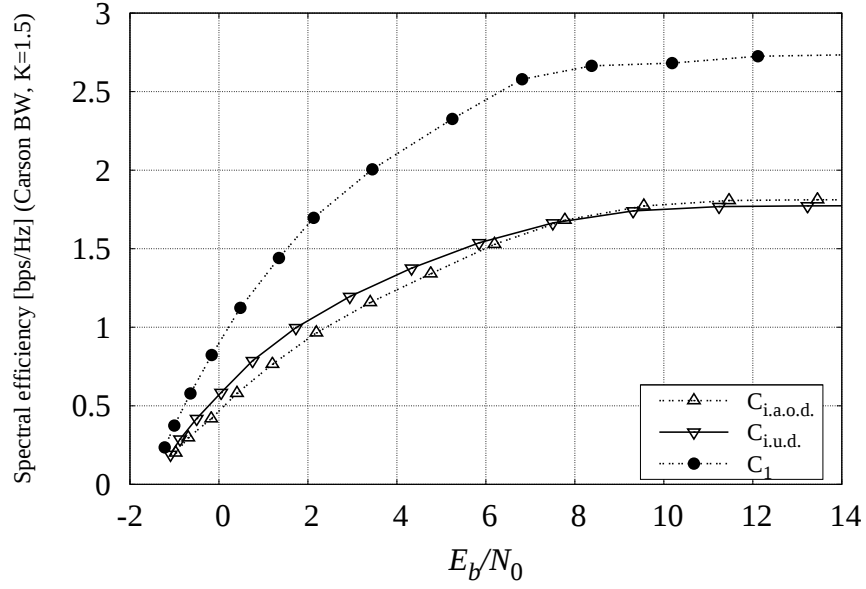
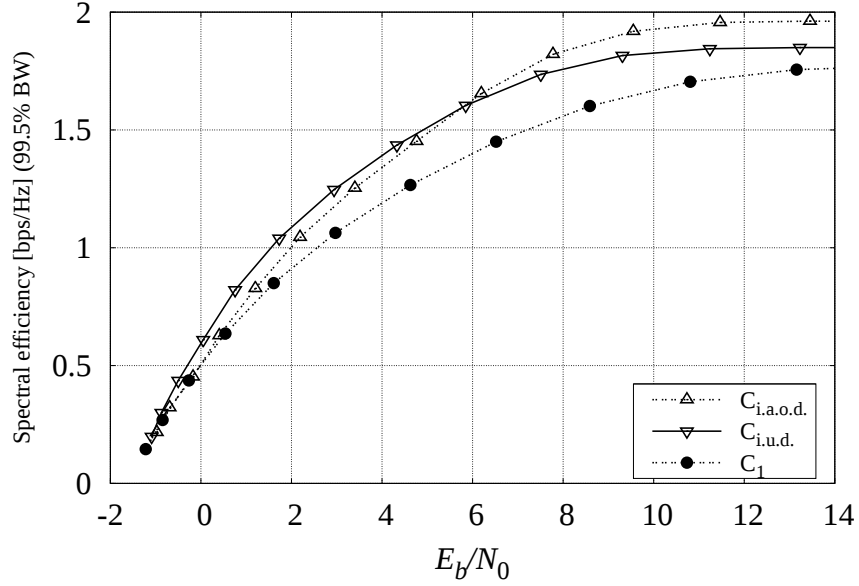


Figure 2.26: Autocorrelation function of the CPM transmitted signal for a quaternary 3-RC with $h = 2/7$.

(a) Carson's bandwidth ($K = 1.5$)

(b) 99.5% bandwidth

Figure 2.27: Spectral efficiency for an octal 2-RC with $h = 1/7$.

Table 2.8: Optimal 1st-order Markov source for an octal 2-RC with $h = 1/7$ at $E_b/N_0 \simeq -1$ dB.

(a) Transition probabilities ($P_x(i, j)$)								
(x_{n-1})	0	1	2	3	4	5	6	7
$x_n = 0$	.090	.087	.066	.046	.046	.028	.032	.090
$x_n = 1$	.245	.378	.279	.122	.046	.028	.032	.090
$x_n = 2$	.215	.322	.370	.289	.047	.028	.032	.090
$x_n = 3$	.090	.085	.173	.312	.092	.028	.032	.090
$x_n = 4$	.090	.032	.028	.092	.312	.173	.085	.090
$x_n = 5$	.090	.032	.028	.047	.289	.370	.322	.215
$x_n = 6$	.090	.032	.028	.046	.122	.279	.378	.245
$x_n = 7$	.090	.032	.028	.046	.046	.066	.087	.090

(b) State probabilities ($\mu_x(i)$)								
(x_{n-1})	0	1	2	3	4	5	6	7
	.056	.156	.180	.108	.108	.180	.156	.056

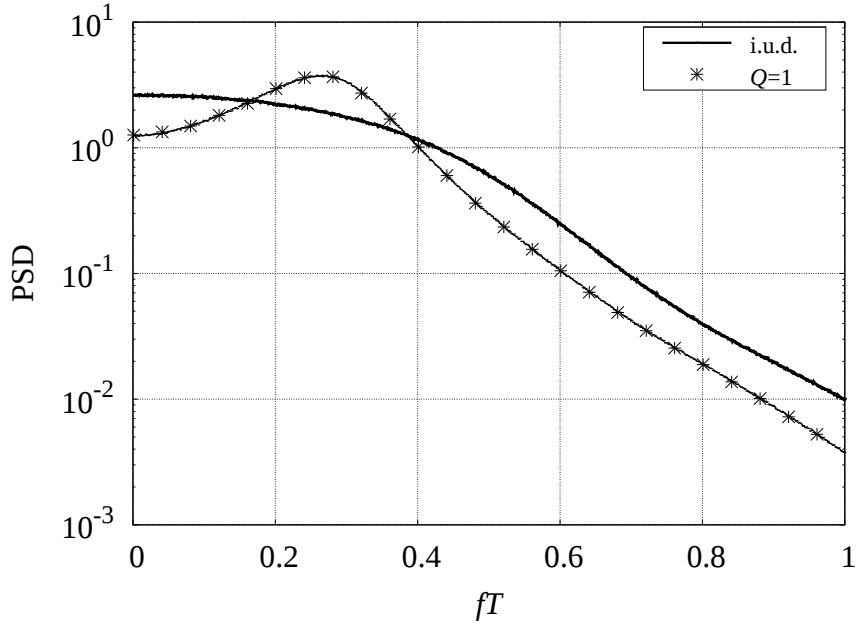
signal-to-noise ratio was large so that C_1 heavily outperformed the $C_{i.u.d.}$, in this case the signal-to-noise ratio is low and C_1 is slightly greater than $C_{i.u.d.}$. However, Table 2.8, in which the optimized source transition probabilities are shown for 1st-order Markov input, underlines that transitions associated to a reduction of the bandwidth start to be encouraged in, despite the small signal-to-noise ratio.

In Fig. 2.28, the PSD and the SPC of the transmitted signal with 1st-order optimized source and i.u.d. inputs are compared. We can observe that the PSD for correlated inputs tends to be concentrated around a peak at a normalized frequency of about $fT = 0.27$, which causes sidelobes in the relevant autocorrelation function (see Fig. 2.29). From the SPC functions we can understand that not only Carson's bandwidth is strongly reduced but also bandwidth definitions based on the SPC decrease. Table 2.9 confirms this

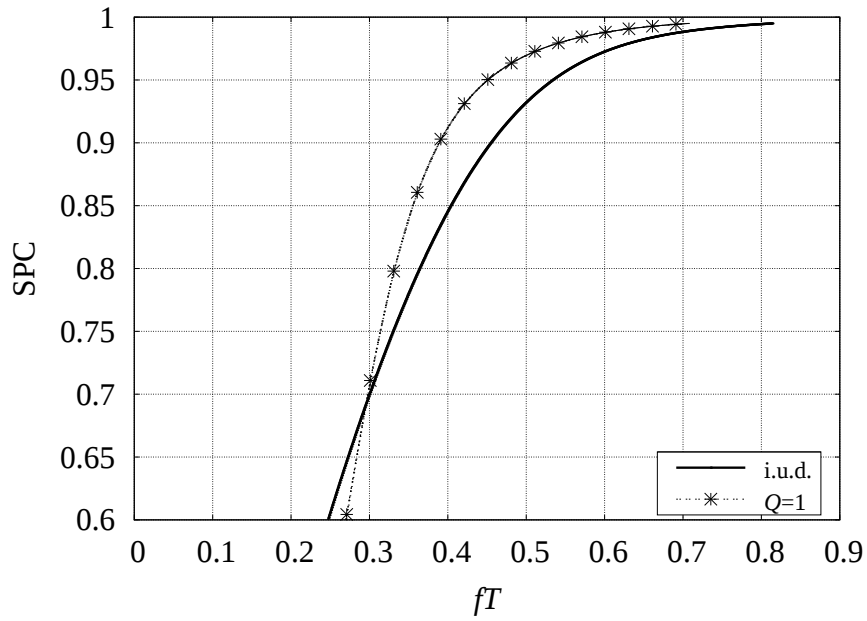
Table 2.9: Normalized bandwidth (under different definitions) for the i. u. d. case and the optimized correlated input for an octal 2-RC with $h = 1/7$.

	Carson ($\mathcal{K} = 1.5$)	90%	99.5%
i. u. d.	1.69	0.91	1.63
$Q = 1$	0.85	0.78	1.42

remark.



(a) Power Spectral Density



(b) Spectral Power Concentration

Figure 2.28: PSD and SPC of a CPM signal with i.i.d. input and with the optimized 1st-order Markov input described in Table 2.8.

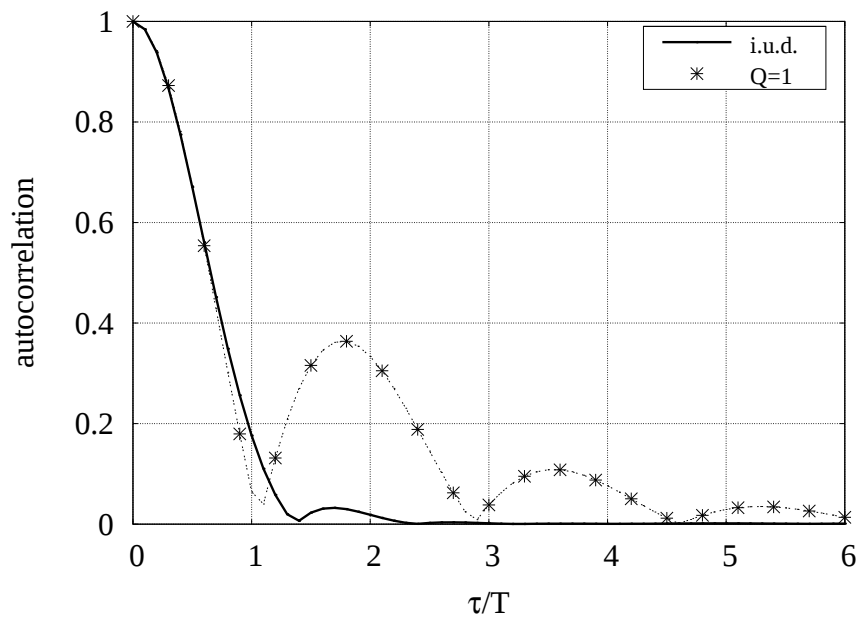


Figure 2.29: Autocorrelation function of the CPM transmitted signal for an octal 2-RC with $h = 1/7$.

Brute-Force Optimization of the Spectral Efficiency

In the following, we analyze a technique for the optimization of the spectral efficiency based on the brute-force approach. The maximization in (2.57) is carried out by choosing the maximum among all values obtained by computing the spectral efficiency function on a grid of points, corresponding to a discretization of the domain of the input probabilities $P_x(n, m)$. In particular, the information rate $I(\mathbf{s}; \mathbf{y})$ is computed by means of a Monte Carlo numerical average, following the algorithm developed in [21] and we can exploit a bandwidth definition based on spectral power concentration since this method does not require closed-form expressions for bandwidth derivatives. Clearly, the brute-force optimization has a very large computational complexity so that it is impractical in all those cases with a large number of source states. For reducing the amount of variables, we impose the following symmetry conditions (justified by the symmetry of the original problem): $P_x(n, m) = P_x(M^Q - n, M^Q - m)$ for each $n = 0, \dots, M^Q - 1$ and $m = 0, \dots, M^Q - 1$. Moreover, another reduction of the number of variables comes from the constraints: $P_x(n, m) = 1 - \sum_{k \neq m} P_x(n, k)$ for each $m = 0, \dots, M^Q - 1$.

In Fig. 2.30, we show the spectral efficiency results for the brute-force optimization with respect to the 99.5% bandwidth in the case of the binary CPM format considered in Fig. 2.20. In such a case, the number of variables $P_x(n, m)$ is 1 when $Q = 1$ and 2 when $Q = 2$. Comparing Fig. 2.30 with Fig. 2.20(b), it is clear that also the spectral efficiency defined with the 99.5% bandwidth can be increased exploiting the correlated inputs instead of i. u. d. inputs.

In Fig. 2.31 and in Fig. 2.32 the brute-force optimization with respect to the 99.5% bandwidth has been carried out for a quaternary scheme with $L = 2$, rectangular frequency pulse (2-REC) and with $h = 1/4$, and for a quaternary 3-RC with $h = 2/7$, respectively. Since when $Q = 2$ the number of variables to be optimized is too large, we just analyzed the case with $Q = 1$ characterized by six variables $P_x(n, m)$. From Fig. 2.31 it can be observed that the spectral efficiency defined with the 99.5% bandwidth can be increased

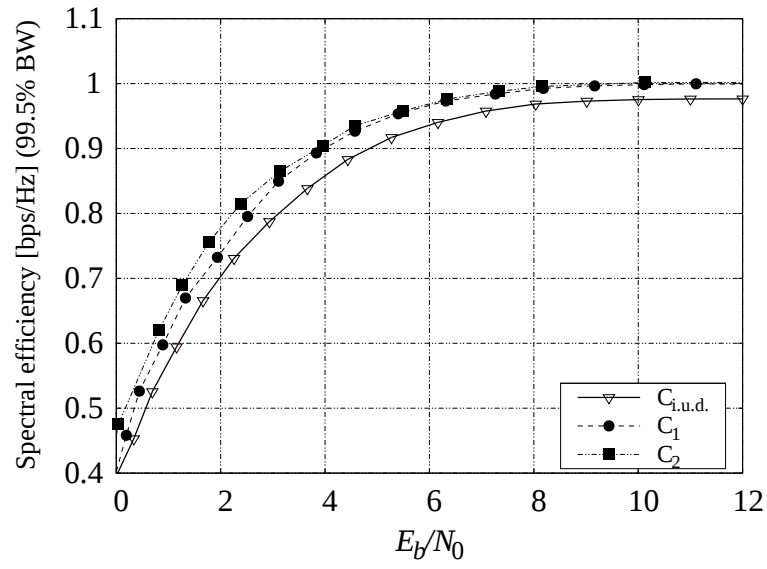


Figure 2.30: Spectral efficiency for a binary 2-RC with $h = 1/3$. Brute-force optimization w. r. t. the 99.5% bandwidth.

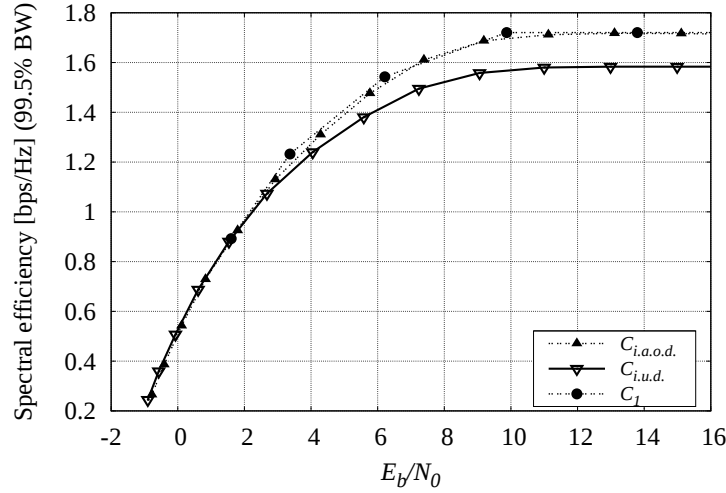


Figure 2.31: Spectral efficiency for a quaternary 2-REC with $h = 1/4$. Brute-force optimization w. r. t. the 99.5% bandwidth.

exploiting the correlated inputs, but it is not increased with respect to the independent asymptotically and optimally distributed inputs (optimized with the Carson's bandwidth but represented in Fig. 2.31 with respect to the 99.5% bandwidth). In Fig. 2.32 just a minor efficiency improvement, with respect to the independent asymptotically and optimally distributed inputs, is achieved at high values of the signal-to-noise ratio through correlated inputs.

Finally in Fig. 2.33, in Fig. 2.34 and in Fig. 2.35 it is shown the effect of the brute-force optimization of the spectral efficiency on the information rate I (i.e., the numerator of (2.57)), which is represented on the y-Axis. It is interesting to note that, as previously remarked, the largest impact of the proposed optimization method is on the bandwidth rather than on the information rate, i.e., the optimized source leads to a possibly large reduction of the denominator of (2.57), rather than an increase of the numerator (I_1 is always lower than $I_{i.u.d.}$).

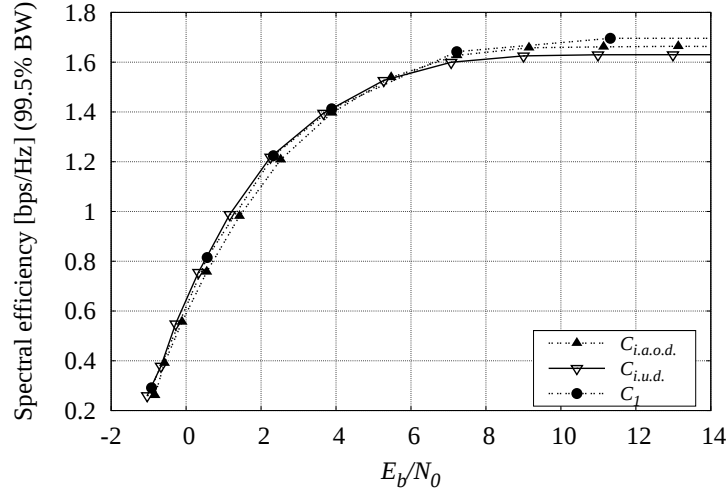


Figure 2.32: Spectral efficiency for a quaternary 3-RC with $h = 2/7$. Brute-force optimization w. r. t. the 99.5% bandwidth.

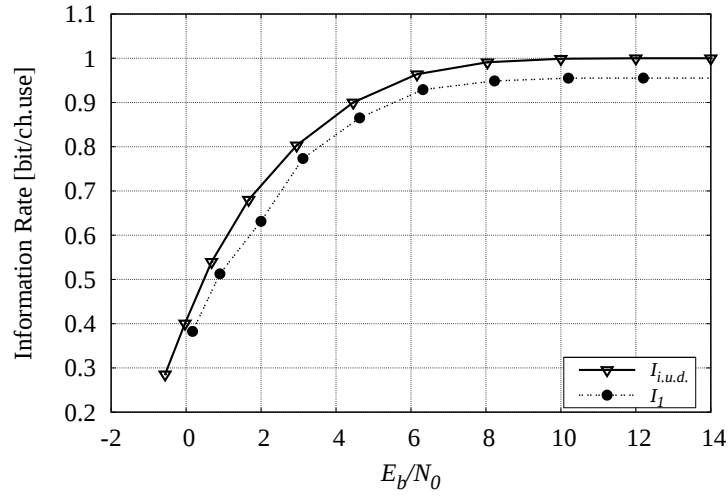


Figure 2.33: Information rate for a binary 2-RC with $h = 1/3$. Brute-force optimization w. r. t. the 99.5% bandwidth.

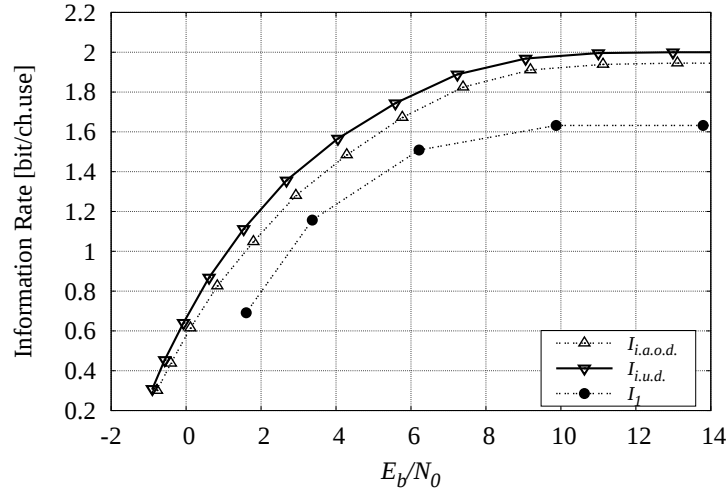


Figure 2.34: Information rate for a quaternary 2-REC with $h = 1/4$. Brute-force optimization w. r. t. the 99.5% bandwidth.

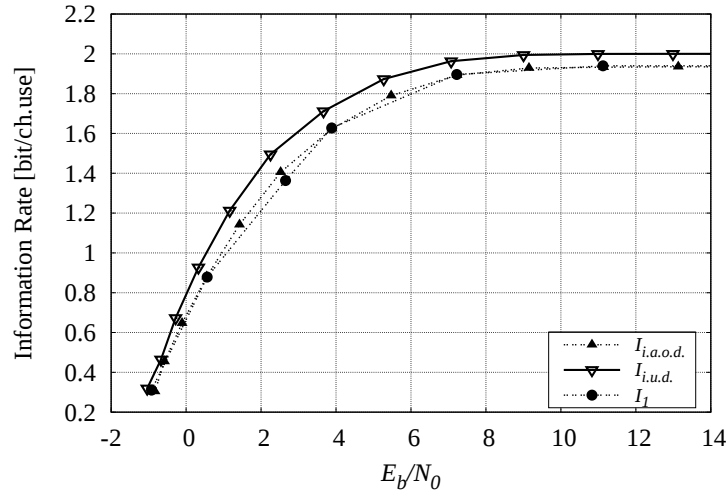


Figure 2.35: Information rate for a quaternary 3-RC with $h = 2/7$. Brute-force optimization w. r. t. the 99.5% bandwidth.

Chapter 3

CPM Reduced-Complexity Soft-Output Detection

We consider continuous phase modulations (CPMs) in iteratively decoded serially concatenated schemes. Although the overall receiver complexity mainly depends on that of the CPM detector, almost all papers in the literature consider the optimal maximum a posteriori (MAP) *symbol* detection algorithm described in Section (2.3.1) and only a few attempts have been made to design low-complexity suboptimal schemes. This problem is faced here by considering the case of an ideal coherent detection. We first investigate the possibility to reduce the complexity of the optimal algorithm by means of reduced-search techniques. Then, we consider two alternative approaches based on the approximation of the CPM signal as the superposition of linearly modulated signals. In particular, we will consider two extensions of the Laurent decomposition, originally proposed for binary CPM signals, to M -ary CPM formats, namely the Mengali and Morelli (MM) decomposition and the Green (G) decomposition. We will also consider the CPM representation based on orthogonal components proposed by Moqvist and Aulin (denoted in the following as MA). The combination of the reduced-search techniques for complexity reduction with those based on a proper decomposition of the CPM signal will be also investigated.

3.1 Introduction

Continuous phase modulations (CPMs) form a class of constant envelope signaling formats which are efficient in power and bandwidth [1]. Moreover, the recursive nature of the modulator described in Section 1.2, makes the CPM signaling formats attractive in serially concatenated schemes to be iteratively decoded [4, 37].

Several decomposition approaches for CPM signals, applied to the design of detection algorithms, were presented in the literature. For serially concatenated CPM signals with iterative decoding, the Rimoldi decomposition approach [2] is usually adopted to derive the optimal maximum a posteriori (MAP) *symbol* detection algorithm (e.g., see [4, 38]). However, for this approach an explicit technique for state reduction, such as that in [39], or reduced search, such as that in [40], should be employed to limit the receiver complexity [41]. A similar observation can be made for other approaches based on alternative representations of the CPM signal, such as those in [42–45].

On the other hand, the Laurent representation, originally devised in [46] for binary modulation formats, and one of its possible extensions to the general case of M -ary CPM signals, namely that proposed by Mengali and Morelli (MM) in [16], are more attractive from the point of view of the complexity reduction.¹ These representations allow to decompose the CPM signal as a superposition of linearly modulated signals. The observation that most of the signal power is contained in a limited number of linearly modulated components (the so-called *principal components*) allows to design a receiver based on these components only, simplifying the receiver front-end and automatically re-

¹The MM approach was also extended to the case of M -ary multi- h CPM signals in [47]. Multi- h schemes were originally studied to improve the distance properties of uncoded digital phase modulation, compared to mono- h digital phase modulation. The basic idea is to increase the distance by keeping distinct signals apart over longer intervals. For coded CPM systems however, the code restrictions help to keep signal traces separate. Also in the turbo code era, it is clear that the inner code or modulation should not be selected for high free distance. This removes the main motivation to examine multi- h signals.

ducing the number of trellis states, at least when MAP *sequence* detection and the Ungerboeck observation model are adopted [48].² In fact, for the Forney observation model, a more complex multidimensional whitened matched filter (WMF) must be adopted [50], and it can be also shown that the “automatic” state reduction does not happen, due to the memory introduced by the WMF.

The generalization of the approach in [48] to MAP *symbol* detection of CPM signals, even in a perfectly coherent setting, is not trivial. All the well known materialization of the MAP symbol detection strategy in the literature (e.g., see [6]) have been obtained by using a probabilistic derivation based on the chain rule and the properties of a Markov source observed through a discrete memoryless channel. Hence, this derivation cannot be directly extended to the Ungerboeck observation model. For linear modulations in the presence of inter-symbol interference, this problem has been recently solved in [52] by using a properly defined factor graph (FG) and the sum-product algorithm (SPA) [7]. This solution is extended in [53] to CPM signals, showing that the designed reduced-complexity detection algorithms entail only a minor performance degradation with respect to the optimal MAP symbol detectors when employed in serially concatenated schemes with iterative detection/decoding.³

An alternative (with respect to the MM approach) extension of the Laurent decomposition to the case of M -ary CPM signals, possibly multi- h , has been recently proposed by Green (G) in [56]. The use of this decomposition

²In analogy with the problem of detection in the presence of inter-symbol interference, for CPM signals a set of sufficient statistics obtained through a bank of filters matched to the pulses of the Laurent representation, as in [48], is said to form the Ungerboeck observation model [49]. On the contrary, when the multidimensional whitened matched filter front-end described in [50] is used, we say that the Forney observation model is employed for detection [51].

³We mention here that a decomposition very similar to that in [16] has been recently proposed in [54, 55]. With respect to the Mengali and Morelli approach, a better approximation of the pulses of the PAM components is provided based on the minimum mean square error criterion. We verified that, with respect to the Mengali and Morelli approach, it gives no advantages from the viewpoint of reducing the complexity of the detection algorithm for a given performance.

for complexity reduction purposes will be also investigated and compared with the previously mentioned solutions. Although this approach seems not to produce any “automatic” state-complexity reduction, it is attractive especially for the purpose of reduction of the front-end complexity. The investigation was also extended to the CPM representation based on orthogonal components, proposed by Moqvist and Aulin in [57], which seems to be interesting since a very reduced number of signal dimension can be effectively used to represent a general CPM signal.

3.2 Detector Model and Complexity Evaluation

We consider the transmission over an ideal coherent channel and, recalling (2.11), the complex envelope of the received signal can be written as

$$r(t) = s(t; \mathbf{x}) + w(t) \quad (3.1)$$

where $s(t, \mathbf{x})$ is the CPM signal (1.1) and $w(t)$ is a complex-valued additive white Gaussian noise (AWGN) process with independent components, each with two-sided power spectral density N_0 . We also assume that the information symbols $\{x_n\}$ are independent, so that the a priori probability $P(\mathbf{x})$ of the information sequence $\mathbf{x}$ can be factorized as

$$P(\mathbf{x}) = \prod_{n=0}^{N-1} P(x_n) \quad (3.2)$$

where $P(x_n)$ is the a priori probability of the information symbol x_n . We refer to Section 1.2 for an in deep description of the signal model $s(t, \mathbf{x})$.

Optimal MAP symbol detection is described in Section 2.3.1 and the resulting algorithm is denoted in the following as *Full-Complexity* algorithm (FC). By analyzing such an algorithm we see that the FC-based detector, as well as all the reduced-complexity detectors we will propose, can be decomposed as the cascade of two blocks, the front end stage (FE) and the detection algorithm (DA), see Fig 3.1. In particular, the FE stage is the block composed by

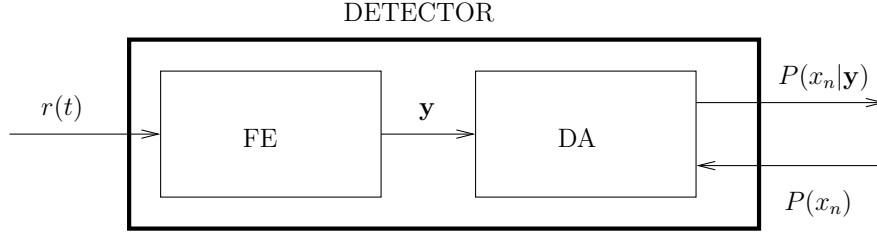


Figure 3.1: CPM detector block diagram.

all filters necessary to derive the sufficient statistics, denoted by $\mathbf{y}$ in Fig 3.1, from the received signal $r(t)$, while the DA block is the detection algorithm working on a trellis. In detail, we consider a soft-output DA which provides to a following error correcting code the a posteriori probabilities $P(x_n|\mathbf{y})$ for all information symbols x_n , employing the sufficient statistics supplied by the FE. If we do not take into account a SCCPM iterative scheme, the a priori symbols probabilities $P(x_n)$ employed by the DA are uniformly distributed and the DA operates just one time for each codeword. On the other hand, if a SCCPM iterative scheme is considered, then at each iteration the DA computes a posteriori symbol probabilities starting from the input symbol distribution provided by the code. In both cases, the FE stage works just one time for each codeword.

Thus, it is clear that the detector complexity can be seen as the sum of the complexity of its components block, the FE and the DA. In general, the elements affecting the DA complexity are the number of trellis states, the number of branches and the branch metric evaluation. However, since all detectors we will derive exhibit approximately the same number of branches and the same metric complexity, the major elements influencing the DA complexity is the number of trellis state, which is assumed in the following as the measure for the complexity of such a block. The FE complexity is evaluated in terms of total number of length- T intervals spanned by the matched filters. In fact, if we consider a digital implementation of the front end, for a given oversampling factor, the number of performed additions and multiplications

is just related to the total number of length- T intervals. Looking at the FC algorithm of Section 2.3.1, we derived that the FE is composed by M^L length- T filters $\{\tilde{s}(t; x_n, \omega_n)\}$ (1.16), one for each trial value of (x_n, ω_n) , where ω_n is the CPM correlative state (1.4). These filters are necessary to compute the sufficient statistics $y(x_n, \sigma_n)$ defined in (2.12) and collected in the vector $\mathbf{y}$. The number of trellis states in the DA is pM^{L-1} . Hence, since a lot of CPM formats which are attractive from a spectral efficiency point of view exhibit large values of parameters M , L , and p , the FC complexity becomes unfeasible for such modulation schemes. Hence, in this chapter we will provide a set of detectors which will be compared in term of DA complexity and FE complexity, with the aim of reducing the complexity with respect to the FC assuring a negligible performance degradation.

In conclusion, all detectors we describe in the following will be compared in terms of DA complexity and FE complexity. It is clear that for SCCPM iterative schemes, where FE operates just one time for each codeword while the DA operates one time for each iteration, the dominant factor in the overall detector complexity computation is the DA complexity. On the other hand, for all those schemes in which both the DA and the FE operates just one time for each codeword, both factors must be taken into account in the overall complexity evaluation.

3.3 Reduced-Complexity Algorithms Based on MM Decomposition

3.3.1 MM Decomposition

The complex envelope of the CPM signal (1.1) may be exactly expressed as [16]

$$s(t, \mathbf{x}) = \sum_{k=0}^{F-1} \sum_n a_{k,n} p_k(t - nT) \quad (3.3)$$

where $F = (M-1)2^{(L-1)\log_2 M}$ is the number of linearly modulated pulses $\{p_k(t)\}$, and $\{a_{k,n}\}$ are the so-called pseudo symbols (hereafter, simply referred to as

symbols). The expression of pulses $\{p_k(t)\}$ and that of symbols $\{a_{k,n}\}$ as a function of the modulation parameters and of the information symbols $\{x_n\}$ can be found in [16]. By truncating the summation in (3.3) to the first $K < F$ terms, we obtain the approximation

$$s(t, \mathbf{x}) \simeq \sum_{k=0}^{K-1} \sum_n a_{k,n} p_k(t - nT). \quad (3.4)$$

Most of the signal power is concentrated in the first $M - 1$ components, i.e., those associated with the pulses $\{p_k(t)\}$ with $0 \leq k \leq M - 2$, which are denoted as *principal components* [16]. As a consequence, a value of $K = M - 1$ may be used in (3.4) to attain a very good tradeoff between approximation quality and number of signal components. In fact, it was shown in [48] that MAP sequence detection only based on principal pulses often attain the same performance as the corresponding optimal detection. A nice feature of the principal components is that their symbols $\{a_{k,n}\}_{k=0}^{M-2}$ can be expressed as a function of x_n and $a_{0,n-1}$ only [16]. For example, the symbol $a_{0,n}$ can be computed as [16]

$$a_{0,n} = a_{0,n-1} e^{j\pi h x_n}. \quad (3.5)$$

From (3.5) it is clear that the pseudosymbol $a_{0,n-1}$ definition is very similar to the phase state π_n definition in (1.5); in particular:

$$a_{0,n} = e^{j\pi h \pi_n + L}. \quad (3.6)$$

Hence, symbols $\{a_{0,n}\}$, as the phase state π_n , can assume p different values per time epoch [16] (we recall that the modulation index is defined as $h = r/p$, where r and p are relatively prime integers). In particular, when n is odd, they belong to the alphabet $\mathcal{A}_o = \{e^{j2\pi h m}, m = 0, 1, \dots, p-1\}$, while, when n is even, they belong to $\mathcal{A}_e = \{e^{j\pi h} e^{j2\pi h m}, m = 0, 1, \dots, p-1\}$.⁴

We adopt the following equivalent representation for symbols x_n and $a_{0,n}$ [2]

$$x_n = 2\bar{x}_n - (M - 1) \quad (3.7)$$

$$a_{0,n} = e^{-j\pi h (M-1)(n+1)} e^{j2\pi h \psi_n} \quad (3.8)$$

⁴When r is even, $\mathcal{A}_o$ and $\mathcal{A}_e$ coincide.

so that $\bar{x}_n \in \{0, 1, \dots, M-1\}$ and $\psi_n \in \{0, 1, \dots, p-1\}$. The integer ψ_n can be recursively updated as follows

$$\psi_n = [\psi_{n-1} + \bar{x}_n]_p \quad (3.9)$$

where $[\cdot]_p$ denotes the “modulo p ” operator. In the following, we will exploit the one-to-one correspondence between the sequences $\{a_{0,n}\}$ and $\{\psi_n\}$.

3.3.2 MAP Symbol Detection Based on the MM Decomposition

We represent the received signal (3.1) onto an orthonormal base and denote by $\mathbf{r}$ its vector representation. Exploiting the constant envelope of the CPM signal, we can write [48]⁵

$$p(\mathbf{r}|\mathbf{x}) = p(\mathbf{r}|\mathbf{x}, \boldsymbol{\psi}) \tilde{\propto} \prod_n G_n(x_n, \psi_{n-1}) \quad (3.10)$$

where $\boldsymbol{\psi} = \{\psi_n\}$ and

$$G_n(x_n, \psi_{n-1}) = \exp \left\{ \frac{1}{N_0} \text{Re} \left[\sum_{k=0}^{M-2} p_{k,n} a_{k,n}^* \right] \right\} \quad (3.11)$$

$p_{k,n} = r(t) \otimes p_k(-t)|_{t=nT}$ being the output of a filter matched to the pulse $p_k(t)$, sampled at time nT . The symbol $\tilde{\propto}$ has been used in (3.10) to denote an approximate proportionality relationship. The approximation here is related to the fact that we are considering the principal components only. We notice that the functions $G_n(x_n, \psi_{n-1})$ are not probability density functions (pdfs) nor they are proportional to pdfs. In the arguments of G_n , we omitted the dependence on the received signal and exploited the above mentioned property that all symbols $\{a_{k,n}\}_{k=0}^{M-2}$ in (3.11) depend on x_n and ψ_{n-1} only. In order to implement the MAP sequence detection strategy, the maximization of the pdf $p(\mathbf{r}|\mathbf{x})$ can be performed by using the Viterbi algorithm with branch metrics

⁵In [48] the reader will find the so-called likelihood function, which is proportional to $\ln p(\mathbf{r}|\mathbf{x})$.

given by $\ln G_n(x_n, \psi_{n-1}) \propto \text{Re}[\sum_{k=0}^{M-2} p_{k,n} a_{k,n}^*]$. Hence, the resulting receiver works on a trellis whose state is defined by ψ_{n-1} , and thus the number of trellis states is p [48]. So, it is very interesting to note that the detector based on MM decomposition first of all reduces the FE complexity by approximating the signal with only its principal components and, as secondary effect of FE reduction, an automatic trellis state reduction also arises. Among all CPM signal decompositions we will present, the MM is the only one which attains a DA complexity reduction.

In the following, we extend the technique for complexity reduction of MAP sequence to MAP symbol detection (see also [53]). The joint a posteriori probability mass function (pmf) $P(\mathbf{x}, \boldsymbol{\psi} | \mathbf{r})$ can be expressed as

$$P(\mathbf{x}, \boldsymbol{\psi} | \mathbf{r}) \propto p(\mathbf{r} | \mathbf{x}, \boldsymbol{\psi}) P(\boldsymbol{\psi} | \mathbf{x}) P(\mathbf{x}). \quad (3.12)$$

By using (3.10) and observing that we can further factor the terms $P(\mathbf{x})$ and $P(\boldsymbol{\psi} | \mathbf{x})$ in (3.12) as

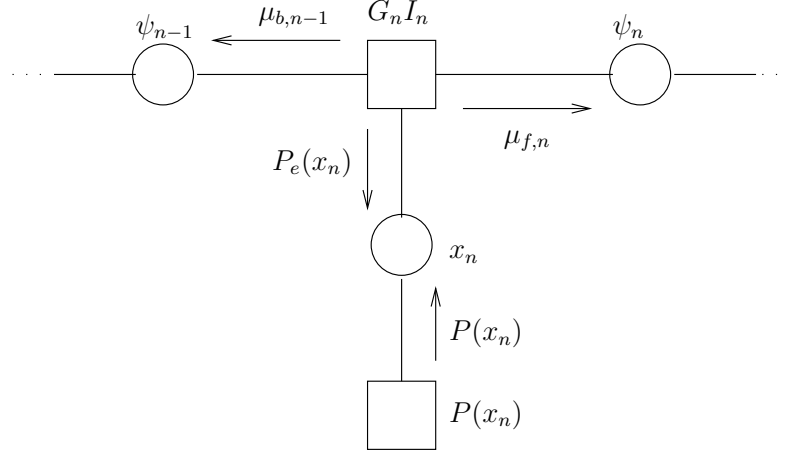
$$P(\mathbf{x}) = \prod_{n=0}^{N-1} P(x_n) \quad (3.13)$$

$$P(\boldsymbol{\psi} | \mathbf{x}) = P(\psi_{-1}) \prod_{n=0}^{N-1} I_n(\psi_n, \psi_{n-1}, x_n) \quad (3.14)$$

where $I_n(\psi_n, \psi_{n-1}, x_n)$ is an indicator function to one if x_n , ψ_{n-1} , and ψ_n respect the constraint (3.9) and equal to zero otherwise, we obtain

$$P(\mathbf{x}, \boldsymbol{\psi} | \mathbf{r}) \propto P(\psi_{-1}) \prod_{n=0}^{N-1} I_n(\psi_n, \psi_{n-1}, x_n) P(x_n) G_n(x_n, \psi_{n-1}). \quad (3.15)$$

The corresponding FG, depicted in Fig. 3.2, is cycle-free. Hence, the application of the SPA with a *non-iterative* forward-backward schedule, produces the *exact* marginal a posteriori probabilities (APPs) of symbols $\{x_n\}$ (except for the approximation related to the use of the principal components only). In the figure, we defined $P_e(x_n)$ as the extrinsic information on x_n , i.e., $P_e(x_n) = P(x_n | \mathbf{r}) / P(x_n)$. With reference to the messages in Fig. 3.2, by applying the


 Figure 3.2: Factor graph corresponding to eqn. (3.15), in the n -th time interval.

updating rules of the SPA, messages $\mu_{f,n}(\psi_n)$ and $\mu_{b,n}(\psi_n)$ can be computed by means of the following forward and backward recursions

$$\begin{aligned} \mu_{f,n}(\psi_n) &= \sum_{\psi_{n-1}} \sum_{x_n} \mu_{f,n-1}(\psi_{n-1}) G_n(x_n, \psi_{n-1}) I_n(\psi_n, \psi_{n-1}, x_n) P(x_n) \\ &= \sum_{x_n} \mu_{f,n-1}(\check{\psi}_{n-1}) G_n(x_n, \check{\psi}_{n-1}) P(x_n) \end{aligned} \quad (3.16)$$

$$\begin{aligned} \mu_{b,n-1}(\psi_{n-1}) &= \sum_{\psi_n} \sum_{x_n} \mu_{b,n}(\psi_n) G_n(x_n, \psi_{n-1}) I_n(\psi_n, \psi_{n-1}, x_n) P(x_n) \\ &= \sum_{x_n} \mu_{b,n}(\check{\psi}_n) G_n(x_n, \psi_{n-1}) P(x_n) \end{aligned} \quad (3.17)$$

where in (3.16)

$$\check{\psi}_{n-1} = [\psi_n - \bar{x}_n]_p$$

(i.e., given ψ_n and x_n , $\check{\psi}_{n-1}$ is such that $I_n(\psi_n, \check{\psi}_{n-1}, x_n) = 1$), whereas in (3.17)

$$\check{\psi}_n = [\psi_{n-1} + \bar{x}_n]_p$$

(i.e., given ψ_{n-1} and x_n , $\check{\psi}_n$ is such that $I_n(\check{\psi}_n, \psi_{n-1}, x_n) = 1$), and with the

following initial conditions

$$\mu_{f,-1}(\psi_{-1}) = P(\psi_{-1}) \quad (3.18)$$

$$\mu_{b,N-1}(\psi_{N-1}) = 1/p. \quad (3.19)$$

Finally, the extrinsic APPs of symbols $\{x_n\}$ can be computed by means of the following completion

$$\begin{aligned} P_e(x_n) &= \sum_{\psi_{n-1}} \sum_{\psi_n} \mu_{f,n-1}(\psi_{n-1}) \mu_{b,n}(\phi_n) I_n(\psi_n, \psi_{n-1}, x_n) G_n(x_n, \psi_{n-1}) \\ &= \sum_{\psi_n} \mu_{f,n-1}(\check{\psi}_{n-1}) \mu_{b,n}(\psi_n) G_n(x_n, \check{\psi}_{n-1}). \end{aligned} \quad (3.20)$$

Hence, as already stated, the MM detection algorithm not only reduces the FE complexity by employing just the filters matched to the principal pulses, but also the DA complexity is reduced by a factor equal to $pM^{L-1}/p = M^{L-1}$, that is the ratio between the number of states of the optimal receiver and that of the proposed receiver. Among all CPM signal decompositions, the MM is the only one bringing to a DA complexity reduction.

Extension of the Algorithm

When the considered CPM format is such that $L > 2$, several simulation results show that the reduced-complexity detector based on the principal components exhibits a significant performance loss with respect to the optimal detector. This is due to the presence of some non-principal components of the MM representation, denoted as *secondary components*, with a non negligible power. In other words, the performance degradation is not due to the detection algorithm but to the inaccurate signal approximation. In particular, for all considered CPM formats with $L = 3$ and $M = 4$, we found eight secondary pulses $\{p_k(t)\}$ with a non-negligible power, as we can observe from the example reported in Fig. 3.3. When these secondary components are taken into account, the functions G_n in (3.11) depend also on x_{n-1} , since symbols $\{a_{k,n}\}$ related

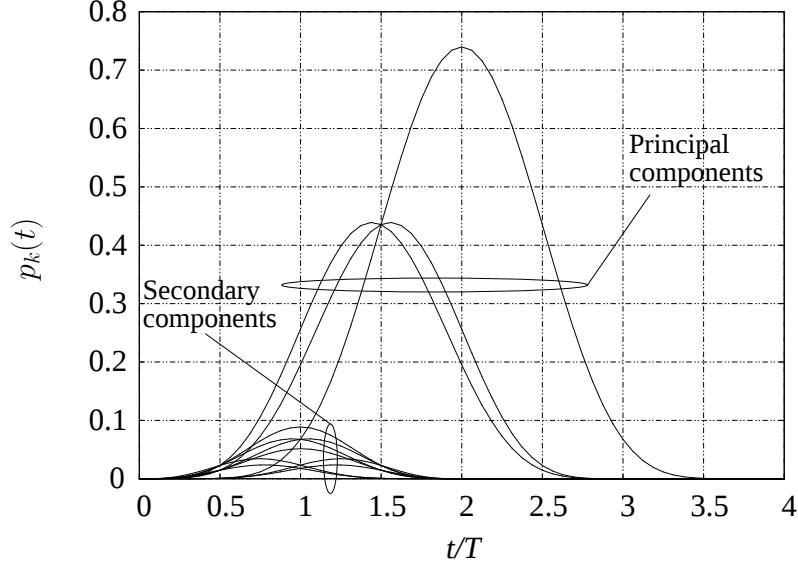


Figure 3.3: Signal pulses for a 3RC modulation with $h = 2/7$ and $M = 4$.

to the considered selected secondary components require the knowledge of the past information symbol x_{n-1} [16]. Hence, G_n in (3.11) is replaced by

$$G'_n(x_n, x_{n-1}, \psi_{n-1}) = \exp \left\{ \frac{1}{N_0} \text{Re} \left[\sum_{k \in K} p_{k,n} a_{k,n}^* \right] \right\} \quad (3.21)$$

where K is the set of all considered pulses, i.e., the $M - 2$ principal pulses and the 8 selected secondary pulses. Thus, extending the derivation described before, we can derive a forward-backward detection algorithm in which the trellis state is defined as the couple (ψ_{n-1}, x_{n-1}) and whose FG is provided in Fig 3.4. In this way, we obtain a soft-output detector working on pM states, with a reduction factor with respect to the optimal scheme equal to M^{L-2} . Hence, even if the FE and DA complexity reduction is not lower than that in the case of MM algorithm with just the principal pulses, MM algorithm with some secondary pulses allows us to obtain more or less the same performance of the FC detector when $L = 3$ with a complexity reduction in both the FE and DA.

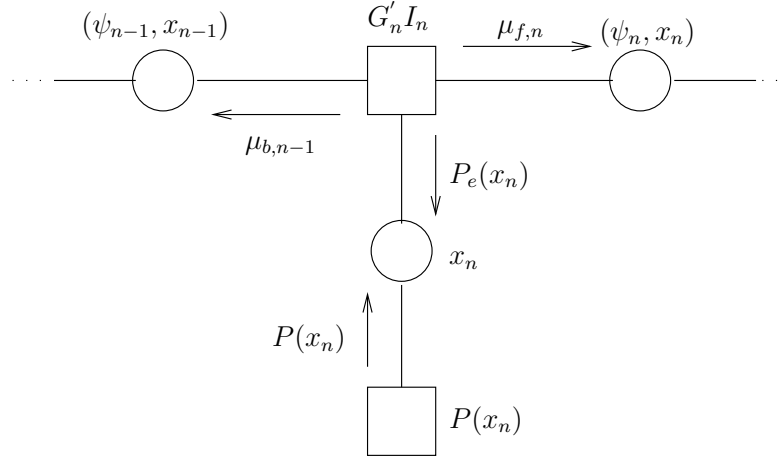


Figure 3.4: Factor graph corresponding to MM based algorithm with some selected secondary pulses, in the n -th time interval.

A further reduction can be obtained by resorting to a proper reduced-search technique, similar to the FT algorithm we will describe in Section 3.6, explained in the following. At each time epoch n , we select p forward metrics and explore only the paths extending from the related states. Then, only the paths selected in this way are considered while performing the backward recursion and the completion stage. In this case, the reduction factor with respect to the optimal scheme is equal to M^{L-1} , exactly as in the basic version of the algorithm, that is when only the principal components are considered. As a criterion for the reduced search, we found convenient not to select the p states (ψ_{n-1}, x_{n-1}) with the best forward metrics, but to select, for each of the possible p values of ψ_{n-1} , the value of x_{n-1} providing the best metric.

3.4 Complexity-Reduction Based on the G Decomposition

As already anticipated in Section 3.3.2, in contrast with the MM decomposition, the CPM decompositions in [42, 54, 56, 57] allow to reduce the front end complexity but not the state complexity of the DA. They have in common that the reduction in the front end complexity is reached by computing the sufficient statistics in (2.12) using alternative representations of the complex envelope of a CPM signal. The CPM detector, i.e., the DA, remains unchanged with respect to the FC detector.

3.4.1 The G Decomposition

Based on an extension of the Laurent decomposition described in [56], in the interval $[nT, (n+1)T]$ the complex envelope of the CPM signal (1.1) can be exactly expressed as⁶

$$s(t, \mathbf{x}) = \sum_{k=0}^{2^{L-1}-1} \sum_{m=0}^{L_k-1} b_{k,m,n} \prod_{i=0}^{L-1} u_{i+m+L\beta_{2k,i,n-[i+m]_L}}(\tau) \quad (3.22)$$

where $[\cdot]_L$ denotes the “modulo L ” operator, $t = nT + \tau$, $\tau \in [0, T)$ and $\beta_{k,i} \in \{0, 1\}$ are the coefficients of the binary representation of the index k , i.e.,

$$k = \beta_{k,0} + 2\beta_{k,1} + \dots + 2^{L-1}\beta_{k,L-1}. \quad (3.23)$$

The number of terms in the k th subgroup is equal to

$$L_k = \min_{i=1,\dots,L-1} \{L(2 - \alpha_{k,i}) - i\} \quad (3.24)$$

where $\alpha_{k,i}$ is the i th bit in the radix-2 representation of k , i.e., $k = \sum_{i=1}^{L-1} 2^{i-1} \alpha_{k,i}$, and $\alpha_{k,0} = 0$. The complex coefficients $\{b_{k,m,n}\}$ and the pulses $\{u_{i,n}(\tau)\}$ are

⁶In [56], the author explains that the decomposition (3.22) holds only when the product $h \cdot |x_n|$ is not an integer for any possible value of x_n .

data-dependent; in particular $b_{k,m,n}$ is a nonlinear function of the input sequence:

$$b_{k,m,n} = e^{j\pi_n} e^{j\pi h \sum_{i=0}^{L-1} (1-\beta_{X_k(m),i}) \alpha_{n-i}} \quad (3.25)$$

where $X_k(j)$ is defined as

$$\begin{aligned} X_k(0) &= 2k \\ X_k(j) &= 2X_k(j-1) + 1 \\ 0 \leq k &\leq 2^{L-1} - 1; \quad 1 \leq j \leq L_k - 1 \end{aligned} \quad (3.26)$$

and π_n is the CPM *phase state*, recursively defined in (1.5). The time-varying basic pulse $u_{i,n}(\tau)$ in (3.22) is of the form

$$u_{i,n}(\tau) = \begin{cases} \frac{\sin[2\pi h|x_n|q(\tau+iT)]}{\sin(\pi h|x_n|)}, & 0 \leq i \leq L-1 \\ \frac{\sin[\pi h|x_n| - 2\pi h|x_n|q(\tau+(i-L)T)]}{\sin(\pi h|x_n|)}, & L \leq i \leq 2L-1 \\ 0 & \text{otherwise.} \end{cases} \quad (3.27)$$

The key difference between this decomposition and that by MM described in Section 3.3.1 is that pulses $\{u_{i,n}(\tau)\}$ are now data-dependent; in particular on the n th symbol interval, the pulses which contribute to the CPM waveform in (3.22) depend on the magnitudes of the L most recent symbols: $\{|x_n|, \dots, |x_{n-(L-1)}|\}$. Therefore it is clear that the coherent receiver requires

$$\sum_{i=0}^{2^{L-1}-1} L_i \left(\frac{M}{2}\right)^L \quad (3.28)$$

matched filters of length T . However, most of the signal power is contained in the first terms of the outer summation and hence in many cases of practical interest, we can adopt the following signal approximation of the CPM signal in the interval $[nT, (n+1)T]$:

$$s(t, \mathbf{x}) \simeq \sum_{k=0}^{Q-1} \sum_{m=0}^{L_k-1} b_{k,m,n} \prod_{i=0}^{L-1} u_{i+m+L\beta_{2k,i},n-[i+m]_L}(\tau) \quad (3.29)$$

with $Q < 2^{L-1}$.

3.4.2 MAP Symbol Detection Based on the G Decomposition

The MAP symbol detection strategy based on the G decomposition can be carried out as the optimal MAP symbol detection strategy described in Section 2.3.1. In fact, from (3.22) we find that the necessary matched filters have length T . Hence, we can derive a BCJR algorithm in which the forward and backward recursions have the same expression and the same probabilistic meaning of the recursions described in Section 2.3.1. The only difference is in the computation of the vector $\mathbf{y}$ of sufficient statistics. In this case at hand, replacing (3.29) in (2.12) we find

$$\begin{aligned}
 y_n(x_n, \sigma_n) &\simeq \int_{nT}^{(n+1)T} r(t) \sum_{k=0}^{Q-1} \sum_{m=0}^{L_k-1} b_{k,m,n}^* \prod_{i=0}^{L-1} u_{i+m+L\beta_{2k,i,n-[i+m]_L}}(\tau) dt \\
 &= \sum_{k=0}^{Q-1} \sum_{m=0}^{L_k-1} b_{k,m,n}^* \int_{nT}^{(n+1)T} r(t) \prod_{i=0}^{L-1} u_{i+m+L\beta_{2k,i,n-[i+m]_L}}(\tau) dt \\
 &= \sum_{k=0}^{Q-1} \sum_{m=0}^{L_k-1} b_{k,m,n}^* \Gamma_{k,m,n}
 \end{aligned} \tag{3.30}$$

where we have defined $\{\Gamma_{k,m,n}\}$ as a output of the bank of length- T matched filters:

$$\Gamma_{k,m,n} \triangleq \int_0^T r(\tau + nT) \prod_{i=0}^{L-1} u_{i+m+L\beta_{2k,i,n-[i+m]_L}}(\tau) d\tau. \tag{3.31}$$

As previously mentioned, in many cases of practical interest we can compute a very good approximation of y_n by truncating the outer summation in (3.30) (i.e., over the variable k), taking a value of Q lower than 2^{L-1} . Hence, we find a suboptimal detector that with respect to the optimal FC detector (i.e., when $Q = 2^{L-1}$) has the same number of trellis states in the DA (pM^{L-1}) but a reduced number of front end filters.

When $Q = 1$, a further simplification can allow a reduction in the number of trellis states. Let us define

$$c_{k,n} \triangleq a_{0,n} e^{j\pi h \sum_{i=0}^{L-1} \beta_{2k,i} \alpha_{n-i}} \tag{3.32}$$

3.5. Complexity-Reduction Based on the MA Decomposition 109

where $a_{0,n}$ is described in (3.5). By some mathematical manipulations it can be proved that $b_{k,m,n} = c_{k,n-m}$. Hence, when $Q = 1$ (3.30) becomes

$$y_n(x_n, \sigma_n) \simeq \sum_{m=0}^{L_0-1} c_{0,n-m}^* \Gamma_{0,m,n} \quad (3.33)$$

where from (3.24) $L_0 = L + 1$ and $c_{0,n} = a_{0,n}$. By truncating the summation over the variable m considering $m = 0$ and $m = 1$, we obtain:

$$y_n(x_n, \sigma_n) = a_{0,n}^* \Gamma_{0,0,n} + a_{0,n-1}^* \Gamma_{0,1,n} \quad (3.34)$$

which just depends on the present symbol x_n , on ψ_n which is the integer representation of $a_{0,n}$ and which can assume p values, and on $\{|x_n|, \dots, |x_{n-(L-1)}|\}$ (necessary to describe $\Gamma_{0,m,n}$ and $\Gamma_{0,1,n}$). In conclusion, we can define a new signal state which belongs to a set of cardinality $p(M/2)^{L-1}$ instead of pM^{L-1} . Hence, in such a case a reduction in DA complexity is also possible, but, as we will see in Section 3.7, such a detector does not work due to the bad signal approximation. In all practical cases we need $Q > 1$ to have a good FE stage.

3.5 Complexity-Reduction Based on the MA Decomposition

3.5.1 The MA Decomposition

The method in [57] (denoted as *principal components (PC) method*) proposed to compute an efficient orthonormal base for the space spanned by a given signal set, can be applied to the finite set of CPM waveform provided in (1.16). We recall such expression

$$s_i(t) \triangleq \tilde{s}(t; x_n, \sigma_n) = \sqrt{\frac{2E_s}{T}} e^{j\pi h \pi_n} \exp \left\{ j2\pi h \sum_{m=0}^{L-1} x_{n-m} q(t + mT) \right\}, \quad 0 \leq t \leq T \quad (3.35)$$

110 Chapter 3. CPM Reduced-Complexity Soft-Output Detection

where $\sigma_n = (\omega_n, \pi_n)$ is defined in (1.4) and (1.5) and i is a given integer representation of the variables (x_n, σ_n) , $i = 0, 1, \dots, pM^L - 1$. The main advantage of the orthonormal base representation in [57] with respect to the well-known Gram-Schmidt technique, is that base functions are sorted in order of importance: the energy content in the first K functions is maximized.

Following this idea, we assume a set of linearly independent CPM waveform removing the contribution of phase state π_n from (3.35); thus we obtain the set $\{\bar{s}_k(t)\}_{k=0}^{M^L-1}$ (already defined in (1.17))

$$\bar{s}_k(t) \triangleq \bar{s}(t; x_n, \omega_n) = \exp \left\{ j2\pi h \sum_{l=0}^L x_{n-l} q(t + lT) \right\}, \quad 0 \leq t \leq T \quad (3.36)$$

$$k = 0, 1, \dots, M^L - 1$$

where k is a given integer representation of the variables $(x_n, \omega_n) = (\mathbf{x}_{n-L+1}^n)$.⁷ Initially we find the normalized cross-correlation Hermitian matrix $\mathbf{R}$ of dimension $M^L \times M^L$, whose element (i, j) is given by the following inner product $\langle \cdot, \cdot \rangle$

$$\mathbf{R}(i, j) \triangleq \langle \bar{s}_i(t), \bar{s}_j(t) \rangle = \int_0^T \bar{s}_i(t) \bar{s}_j^*(t) dt. \quad (3.37)$$

The matrix $\mathbf{R}$ can be uniquely decomposed by the so-called *singular value decomposition* as

$$\mathbf{R} = \mathbf{U} \mathbf{S}^2 \mathbf{U}^H \quad (3.38)$$

where $\mathbf{U}^H$ denotes the complex conjugate transpose of $\mathbf{U}$, $\mathbf{U}$ is an unitary matrix ($\mathbf{U} \mathbf{U}^H = \mathbf{U}^H \mathbf{U} = \mathbf{I}$) of dimension $M^L \times M^L$ and $\mathbf{S}^2$ is a diagonal matrix of dimension $M^L \times M^L$, $\mathbf{S}^2 = \text{diag}(\lambda_0, \lambda_1, \dots, \lambda_{M^L-1})$. Note that the diagonal entries of $\mathbf{S}^2$, $(\lambda_0, \lambda_1, \dots, \lambda_{M^L-1})$, are the non negative eigenvalues of $\mathbf{R}$, sorted in decreasing order. Finally from [57] it is proved that the pursued vector of M^L orthonormal base functions $\mathbf{b}(t)$,

$$\mathbf{b}(t) \triangleq [b_0(t), b_1(t), \dots, b_{M^L-1}(t)]^T \quad (3.39)$$

⁷In the following we will present a time continuous derivation; a practical implementation can be realized by sampling the set $\{\bar{s}_i(t)\}_{i=0}^{M^L-1}$ of CPM waveforms.

is given by

$$\mathbf{b}(t) = \mathbf{S}^{-1} \mathbf{U}^H \bar{\mathbf{s}}(t) \quad (3.40)$$

where $\bar{\mathbf{s}}(t) \triangleq [\bar{s}_0(t), \bar{s}_1(t), \dots, \bar{s}_{M^L-1}(t)]^T$ and

$$\mathbf{S}^{-1} = \text{diag}(1/\sqrt{\lambda_0}, 1/\sqrt{\lambda_1}, \dots, 1/\sqrt{\lambda_{M^L-1}}).$$

When compared to the Gram-Schmidt procedure, this method does not have the disadvantage of numerical instability and produces base functions that are optimal when a signal space reduction is attempted: for any truncation of the base $\mathbf{b}(t)$ to the $K < M^L$ first member functions, there is no other signal base with K members that carries an higher portion of the signal energy.

3.5.2 MAP Symbol Detection Based on the MA Decomposition

The orthonormal CPM base representation can be employed to implement the optimal MAP symbol detection strategy described in Section 2.3.1. In practice, instead of projecting the received signal $r(t)$ (reported in (3.1)) onto the original CPM set $\{\bar{s}(t; x_n, \omega_n)\}$ as in (2.12), one can project $r(t)$ and the M^L CPM waveforms $\bar{s}(t; x_n, \omega_n)$ onto the orthonormal set $\{b_i(t)\}_{i=0}^{M^L-1}$.

In fact, representing the received signal as

$$r(t) = \sum_{n=0}^{N-1} r_n(t)$$

where $r_n(t) \triangleq r(t) \text{rect}_T(t - nT)$,⁸ $r_n(t)$ and $s_i(t)$ can be exactly expressed as

$$r_n(t) = \sum_{j=0}^{M^L-1} r_{n,j} b_j(t) \quad n = 0, 1, \dots, N-1 \quad (3.41)$$

$$\bar{s}_i(t) = \sum_{j=0}^{M^L-1} \bar{s}_{i,j} b_j(t) \quad i = 0, 1, \dots, M^L-1 \quad (3.42)$$

⁸ $\text{rect}_T(t) = 1$ if $0 \leq t < T$, $\text{rect}_T(t) = 0$ elsewhere.

where the projections $r_{n,j}$ and $\bar{s}_{i,j}$ are defined as

$$r_{n,j} = \langle r_n(t), b_j(t) \rangle \quad (3.43)$$

$$\bar{s}_{i,j} = \langle \bar{s}_i(t), b_j(t) \rangle \quad (3.44)$$

$$i, j = 0, 1, \dots, M^L - 1$$

$$n = 0, 1, \dots, N - 1.$$

Thus replacing (3.41) and (3.42) into (2.12)

$$\begin{aligned} y_n(x_n, \sigma_n) &= e^{-j\pi_n} \int_{nT}^{(n+1)T} r(t) \bar{s}^*(t - nT; x_n, \omega_n) dt \\ &= e^{-j\pi_n} \int_0^T r_n(t - nT) \bar{s}_i^*(t) dt \\ &= e^{-j\pi_n} \int_0^T \sum_{j=0}^{M^L-1} r_{n,j} b_j(t) \sum_{k=0}^{M^L-1} \bar{s}_{i,k}^* b_k^*(t) dt \\ &= e^{-j\pi_n} \sum_{j=0}^{M^L-1} \sum_{k=0}^{M^L-1} r_{n,j} \bar{s}_{i,k}^* \int_0^T b_j(t) b_k^*(t) dt \\ &= e^{-j\pi_n} \sum_{j=0}^{M^L-1} r_{n,j} \bar{s}_{i,k}^* \end{aligned} \quad (3.45)$$

where we have exploited the orthonormality of $\{b_j(t)\}$ functions.

Since base functions produced through the method of principal components described in Section 3.5.1 are sorted in decreasing energy order, we can reduce the set of functions $b_j(t)$ employed in the computation of projections. In other words, we can approximate (3.45) as

$$y_n(x_n, \sigma_n) \simeq \sum_{j=0}^{K-1} r_{n,j} \bar{s}_{i,k}^* \quad (3.46)$$

reducing the set of front end filters from M^L to $K < M^L$. Thus, just as in the case of the G decomposition of Section 3.4, the MA decomposition allows a FE complexity reduction but the trellis of the detection algorithm remains the same on the FC detector.

3.6 Reduced-Search Algorithms

We now investigate some techniques aiming to a complexity reduction in the DA stage, without effects on the FE stage. In order to obtain a reduced-complexity DA, we can adopt the *reduced-search* technique [58]: the algorithm still works on a full-state trellis, that is no state reduction [39] is performed, but it explores only a subset of the possible paths on the trellis. In principle, we can apply this method to both type of trellises we have derived in previous Sections: the trellis of the full complexity algorithm (whatever is the FE, i.e. the optimal one or an approximated version obtained by the G or MA decompositions) or the trellis of the MM detector. Hence, we describe the reduced-search algorithm while operating on the the FC trellis and then, we try to apply such a technique to the DA of the MM detector (see Section 3.6.3).

3.6.1 Rationale

Since the lower the metric (either forward or backward) of a state, the more negligible its contribution to the summations of the recursions and completion, it is natural to explore only the paths extending from the states with the largest metrics. Let us suppose to keep memory, at each time epoch, of the S largest forward and the S largest backward metrics only, and to explore only the paths extending from the related states while performing the recursions. In the state metrics computation, the contribution corresponding to unexplored paths is considered null. If we consider the number S of saved metrics as a measure of the complexity of the algorithms⁹, the reduction factor with respect to the full-complexity BCJR algorithm is about pM^{L-1}/S .

By connecting the states whose forward metrics have been saved, a set of forward-selected-paths (FSPs) is progressively built, and the same can be

⁹ Although a more rigorous evaluation of the complexity would also take into account the additional computational load in implementing the various algorithms, the adopted choice is the most common in the literature. More detailed complexity comparisons can be found in [40].

done in order to define a set of backward-selected-paths (BSPs). The values of the forward (respectively, backward) metrics related to states which do not belong to the set of FSPs (respectively, BSPs) are not available during the completion stage. The unavailable metrics are traditionally replaced by zero, thus neglecting the terms containing them while computing the summations. In this case, any state can give a non-zero contribution to the completion only if it belongs to both FSPs and BSPs, so that the effective trellis is given by the intersection of FSPs and BSPs [40]. This algorithm, which will be referred to as *double-trellis* (DT) algorithm, can exhibit a poor performance since the sets of FSPs and BSPs are built independently of each other, and their intersection could thus be almost empty. For this reason, in [58] it is suggested not to build any set of BSPs and to perform the backward search over the set of FSPs, thus giving a predominant role to the forward recursion. We will refer to this solution as *forward-trellis* (FT) algorithm. From a computational viewpoint, the FT algorithm is simpler than the DT algorithm, since it requires to compare and sort the metrics in the forward recursion only, while this procedure is avoided in the backward recursion. A detailed discussion on the criterion for selecting the most convenient solution between the DT and the FT algorithm, based on the physical features of the considered finite-state machine, can be found in [40]. Following these arguments, we can state that, in the case of detection of CPM signals, there exists no physical reason to privilege one recursion instead of the other, so that the DT algorithm is expected to be more effective.¹⁰ The simulation results reported in Section 3.7 confirm this conjecture.

3.6.2 Optimization

A couple of techniques for improving the effectiveness of the DT algorithm are described in the following.

As stated before, a traditional completion replacing the unavailable metrics by null values works on a subset of paths given by the intersection of FSPs and

¹⁰For the same reason, the reduced-complexity algorithms based on state reduction [39] cannot be successfully applied.

BSPs. While in the case of the FT algorithm the intersection coincides with the set of FSPs, in the case of the DT algorithm the intersection could result almost empty, since the sets of FSPs and BSPs are built independently of each other. This issue is addressed in [59], where the authors propose a completion on a window of multiple trellis sections, thus implying a significant increase in the computational complexity of the completion stage. We propose a simpler solution allowing the completion stage to work not on the intersection but on the union of the sets of FSPs and BSPs, so that all the metrics saved during the recursions can give a contribution to the final result. The way to do this consists of replacing the unavailable metrics in (2.21) by proper non-zero values. For each time epoch n , let $\eta_{f,n}^{\text{MIN}}$ be the lowest metric saved during the forward recursion, and let $\eta_{b,n}^{\text{MIN}}$ be the lowest metric saved during the backward recursion (expressions for forward and backward recursions are provided in (2.17) and (2.18), respectively). When a given state σ_n does not belong to the set of FSPs at time epoch n , any non-zero value lower or equal to $\eta_{f,n}^{\text{MIN}}$ could be a reasonable choice for replacing the unavailable metric $\eta_{f,n}(\sigma_n)$ while performing the completion (2.21). Since we found, by means of extensive computer simulations, that overestimating the unavailable metrics provides a better performance than underestimating them, we choose the largest value in the allowed range. Hence, when the factor $\eta_{f,n}(\sigma_n)$ in (2.21) is not available, we replace it by $\eta_{f,n}^{\text{MIN}}$. Similarly, when $\eta_{b,n+1}(\sigma_{n+1})$ is not available, we replace it by $\eta_{b,n+1}^{\text{MIN}}$. When both factors are not available, we instead ignore the contribution of the corresponding path. This solution, which will be referred to as *non-zero* (NZ) completion, causes only a slight increase in computational complexity [40], but ensures a significant performance improvement (see the simulation results in Section 3.7).

A further optimization technique can be obtained by exploiting the probabilistic meanings of the state metrics, provided in (2.19) and (2.20) and here recalled:

$$\begin{aligned}\eta_{f,n}(\sigma_n) &\propto P(\sigma_n | \mathbf{y}_0^{n-1}) \\ \eta_{b,n}(\sigma_n) &\propto p(\mathbf{y}_n^{N-1} | \sigma_n).\end{aligned}$$

In particular, (2.19) and (2.20) imply that the FSPs are selected based on the MAP criterion, while the BSPs are selected based on the maximum-likelihood (ML) criterion. These different criteria, which result equivalent only when the modulation symbols are equally likely and the receiver does not perform iterative decoding, have a significant impact on the reliability of the selected paths, that is on the probability that such paths include the correct one [40]. It is intuitive to conjecture that the MAP approach is more reliable, and extensive computer simulations confirm this fact. As a consequence, the reduced-search forward recursion is more reliable than the reduced-search backward recursion. An effective solution for this asymmetry is presented in [40], with focus on the BCJR algorithm when employed for MAP symbol detection over channels affected by inter-symbol interference (ISI), and consists of modifying the definition of the backward recursion so that also the BSPs can be selected based on the MAP criterion. Since this modification ensures a significant performance improvement when ISI channels are considered, it is interesting to investigate the applicability of the algorithms in [40] to the problem at hand. Unfortunately, the recursive definition of the state σ_n implied by (1.5) makes the direct application of such algorithms impossible. An alternative solution for building a MAP-based set of BSPs is described in the following.

Let us temporarily assume to know, for each time epoch n and each state σ_n , the a priori probability $P(\sigma_n)$ that σ_n is the correct state, and let us define the modified backward metric $\tilde{\eta}_{b,n}(\sigma_n)$ as follows

$$\tilde{\eta}_{b,n}(\sigma_n) = \eta_{b,n}(\sigma_n)P(\sigma_n) . \quad (3.47)$$

Then, taking into account (2.20), we can write

$$\tilde{\eta}_{b,n}(\sigma_n) \propto p(\mathbf{y}_n^{N-1}|\sigma_n)P(\sigma_n) = p(\mathbf{y}_n^{N-1}, \sigma_n) \propto P(\sigma_n|\mathbf{y}_n^{N-1}) \quad (3.48)$$

showing that it is possible to build a MAP-based set of BSPs by working on the modified backward metrics $\{\tilde{\eta}_{b,n}\}$ instead of the classical ones $\{\eta_{b,n}\}$. In practice, while performing the reduced-search backward recursion at a given time epoch n , the algorithm executes the following steps:

1. it extends the BSPs from time epoch $n + 1$ to time epoch n for the computation of the metrics $\{\eta_{b,n}\}$;
2. it computes the metrics $\{\tilde{\eta}_{b,n}\}$ by scaling the metrics $\{\eta_{b,n}\}$ according to (3.47);
3. it includes the states σ_n with the best S metrics $\{\tilde{\eta}_{b,n}(\sigma_n)\}$ in the set of BSPs, saving the corresponding metrics;
4. it scales the saved metrics $\{\tilde{\eta}_{b,n}\}$ back to $\{\eta_{b,n}\}$ according to (3.47).

We will refer to this algorithm as *modified double-trellis* (MDT) algorithm. It is worth noting that, with respect to the DT algorithm, the forward recursion and the completion stage (either classical or non-zero) are exactly the same. Now, the point is how to compute the terms $\{P(\sigma_n)\}$ required in (3.47). Due to the recursive definition of the state σ_n , they can be computed as follows

$$P(\sigma_{n+1}) = \sum_{\alpha_n} \sum_{\sigma_n} T(\alpha_n, \sigma_n, \sigma_{n+1}) P(\sigma_n) P(\alpha_n) \quad (3.49)$$

after initializing the values of $\{P(\sigma_0)\}$ according to the available information on the actual first state $\bar{\sigma}_0$, as explained in the discussion related to the recursion (2.17). In practice, if we want to perform a MAP-based reduced-search backward recursion, we have to perform the additional full-search recursion (3.49) just for the computation of the terms $\{P(\sigma_n)\}$. Clearly, the latter recursion represents the bottleneck of the method from a complexity viewpoint, thus making the relevance of the MDT algorithm mainly theoretical.

In conclusion, among the various reduced-search algorithms described so far, the best choice in terms of performance/complexity tradeoff is expected to be the DT-NZ algorithm.

3.6.3 Application to the Trellis of MM Detector

The detection algorithm based on the MM decomposition and the optimal BCJR algorithm described in Section 2.3.1 are both characterized by a forward recursion, a backward recursion, and a completion stage. In principle, it

is thus possible to apply the reduced-search techniques described in Section 3.6 to both of them. On the other hand, while the application of such techniques to the optimal BCJR algorithm is justified by the probabilistic meanings of the state metric $\{\eta_{f,n}\}$ and $\{\eta_{b,n}\}$ (see the relevant discussion in Section 3.6), the same is no longer true when the detection algorithm based on the MM decomposition is considered. The key difference is that the state metric $\{\mu_{f,n}\}$ and $\{\mu_{b,n}\}$ does not have any probabilistic meaning, and thus the choice of exploring only the paths extending from the states with the largest metrics cannot be justified. This issue is not only theoretical, since it is fully confirmed by all simulation results, which definitely prove the ineffectiveness of the reduced-search techniques when applied to the recursions (3.16) and (3.17). An interesting exception arises when we consider the trellis associate to the extension of the MM detector, we described in Section 3.3.2.

3.7 Numerical Results

In this section, the performance of the described optimal and simplified detection schemes is assessed by means of computer simulations. To ensure a better numerical stability, the algorithms are implemented in the logarithmic domain [60]. First, we consider the case of uncoded CPM transmissions; then, we focus on the more interesting case of SCCPM schemes with iterative decoding.

In the key of the figures and in the related discussions, the following notation is adopted:

- FC: full-complexity BCJR algorithm (Section 2.3.1);
- MM-P: BCJR-like algorithm based on the principal components of the MM decomposition (Section 3.3.2);
- MM-PSS: BCJR-like algorithm based on the principal and some selected secondary components of the MM decomposition (Section 3.3.2);

- MM-PSS-RS: BCJR-like algorithm based on the principal and some selected secondary components of the MM decomposition, with the application of the reduced-search technique (Section 3.3.2);
- G: BCJR algorithm with the FC-DA stage and simplified FE, based on the G decomposition (Section 3.4.2);
- G-R: BCJR algorithm with the FC-DA stage and simplified FE, based on a subset of filters of the G decomposition, such that an automatic reduction in the number of trellis states is achieved (Section 3.4.2);
- MA: BCJR algorithm with the FC-DA stage and simplified FE, based on the MA orthogonal base components (Section 3.5.2);
- FT: forward-trellis technique applied to FC algorithm (Section 3.6.1);
- DT: double-trellis technique applied to FC algorithm (Section 3.6.1);
- DT-NZ: double-trellis technique with non-zero completion applied to FC algorithm (Section 3.6.2).

3.7.1 Uncoded CPM Transmissions

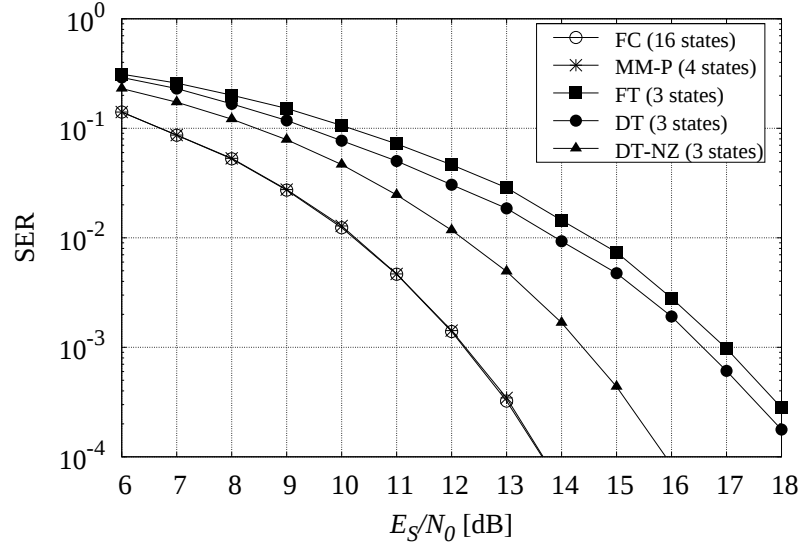
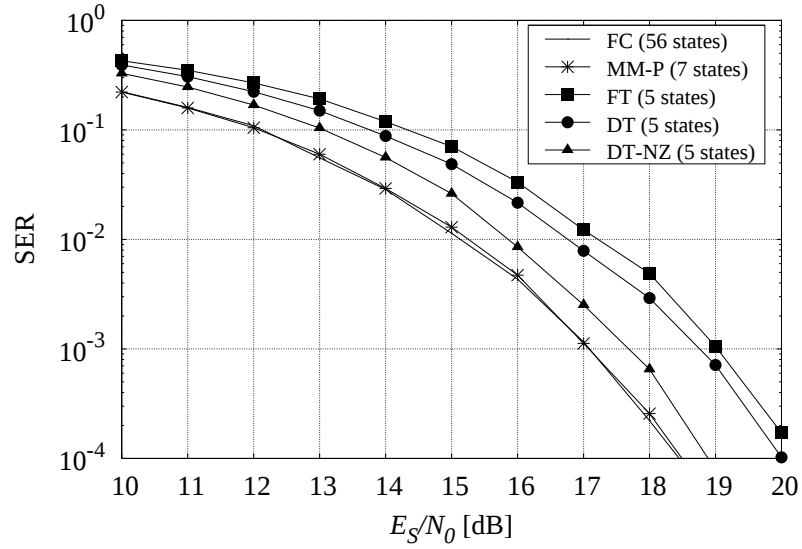
We consider uncoded CPM transmissions, showing the performance of various detection schemes in terms of symbol-error rate (SER) versus E_S/N_0 , N_0 being the one-sided noise power spectral density. Each transmitted frame contains 1000 symbols, and a minimum of 300 symbol errors are counted for computing each SER value.

In Fig. 3.5, a quaternary modulation with raised cosine frequency pulse, $L = 2$ (2RC), and $h = 1/4$ is considered. In this case, the BCJR-like algorithm based on the principal components of the MM decomposition and working on a 4-state trellis performs exactly as the optimal BCJR algorithm working on a 16-state trellis. Fig. 3.5 also reports the performance that can be obtained when the reduced-search techniques are applied to the optimal BCJR algorithm reducing the number of states to 3. As can be observed, the

double-trellis algorithm employing the non-zero completion is by far the most effective, as conjectured in the theoretical analysis carried out in Section 3.6, but the corresponding performance degradation with respect to the FC and MM-P schemes is significant. Other simulation results show that the degradation vanishes when the DT-NZ algorithm works on a 4-state trellis, exactly as the MM-P algorithm. On the other hand, the DT-NZ algorithm is more complex than the MM-P algorithm when they work on a trellis with the same number of states, since it requires additional operations while building the set of selected paths and performing the non-zero completion. On the other hand, it is possible to simplify the front-end by resorting to the G decomposition with a very low value of Q . A detailed analysis of the front-end complexity is reported in the next section devoted to the more interesting case of SCCPM schemes. However, we can anticipate that even from this point of view, the complexity of the MM-P algorithm is lower. Hence, in conclusion the most efficient algorithm for this scenario is definitely the BCJR-like algorithm based on the principal components of the MM decomposition.

In Fig. 3.6, an octal 2RC modulation with $h = 1/7$ is considered. As in the previous scenario, the MM-P algorithm provides the same performance as the optimal BCJR algorithm, even if it works on a trellis which is 8 times simpler. Fig. 3.6 also reports the performance of the algorithms based on reduced-search techniques, when they all work on a 5-state trellis, confirming that the DT-NZ algorithm is the most effective. Other simulation results show that, if we want a negligible performance degradation, the DT-NZ algorithm should work on a 7-state trellis, exactly as the MM-P algorithm. Hence, even for this scenario, the best performance/complexity tradeoff is ensured by the BCJR-like algorithm based on the principal components of the MM decomposition.

In Fig. 3.7, a quaternary 3RC modulation with $h = 2/7$ is considered. In this case, as expected according to the discussion carried out in Section 3.3.2, the BCJR-like algorithm based on the principal components of the MM decomposition exhibits a large performance degradation with respect to the optimal BCJR algorithm. If we compare the performance of the MM-P algo-

Figure 3.5: 2RC modulation with $h = 1/4$ and $M = 4$.Figure 3.6: 2RC modulation with $h = 1/7$ and $M = 8$.

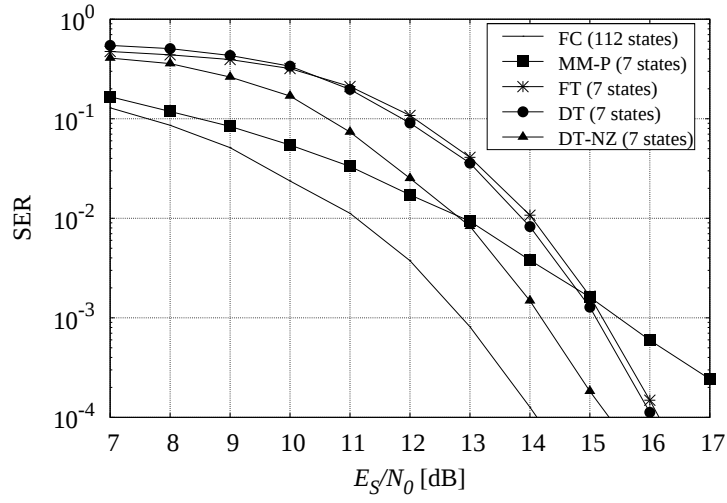


Figure 3.7: 3RC modulation with $h = 2/7$ and $M = 4$.

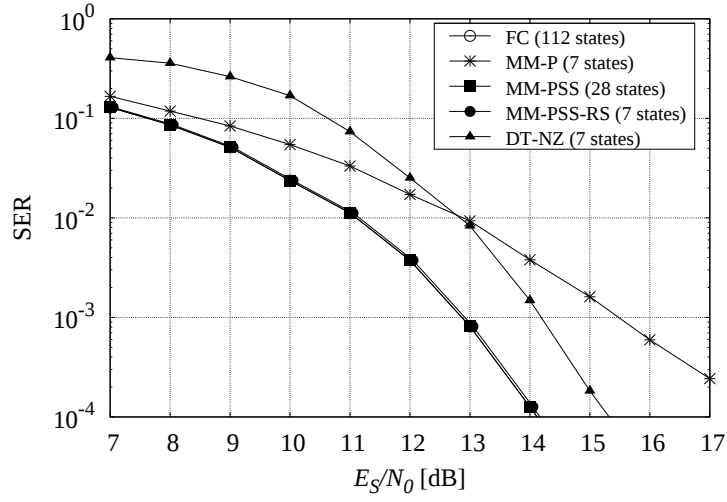


Figure 3.8: 3RC modulation with $h = 2/7$ and $M = 4$.

rithm and that of the reduced-search algorithms when they all work on a 7-state trellis, it is clear that the former solution is more effective at low values of E_S/N_0 , whereas the latter solutions are more effective at large values of E_S/N_0 . In particular, the DT-NZ algorithm outperforms all other detection schemes when E_S/N_0 is larger than 13 dB. On the other hand, Fig. 3.8 shows that we can achieve the optimal performance by taking into account also some selected secondary components of the MM decomposition, and applying the MM-PSS algorithm, which works on a 28-state trellis, or even the MM-PSS-RS algorithm, which works on a 7-state trellis. Hence, as in all other considered scenarios, the choice of resorting to detection schemes based on MM decomposition is more effective than applying reduced-search techniques to the optimal BCJR algorithm, whatever is the considered FE (an approximated version derived by G or MA decompositions, as well as the full complexity front end).

3.7.2 Iterative Decoding of SCCPM Schemes

We now consider SCCPM schemes with iterative decoding, reporting the performance in terms of bit error rate (BER) versus E_b/N_0 , E_b being the received energy per information bit. The relevant system model is depicted in Fig. 3.9. The considered concatenated schemes are completely defined by the outer code description, the codeword length, the type of interleaver (bit or symbol), the type of mapping, the CPM parameters, and finally the total number of iterations for the iterative decoding process. In our computer simulations, different SCCPM schemes, whose details are summarized in Tables 3.1, are considered. In all these schemes we consider a CPM modulation format concatenated with a non-recursive rate-1/2 convolutional code (CC) with generators $G_1 = 7$ and $G_2 = 5$ (octal notation) and the channel phase is considered known to the receiver. A dithered relative prime bit interleaved is used and coded bits are mapped into CPM symbols by using natural mapping.

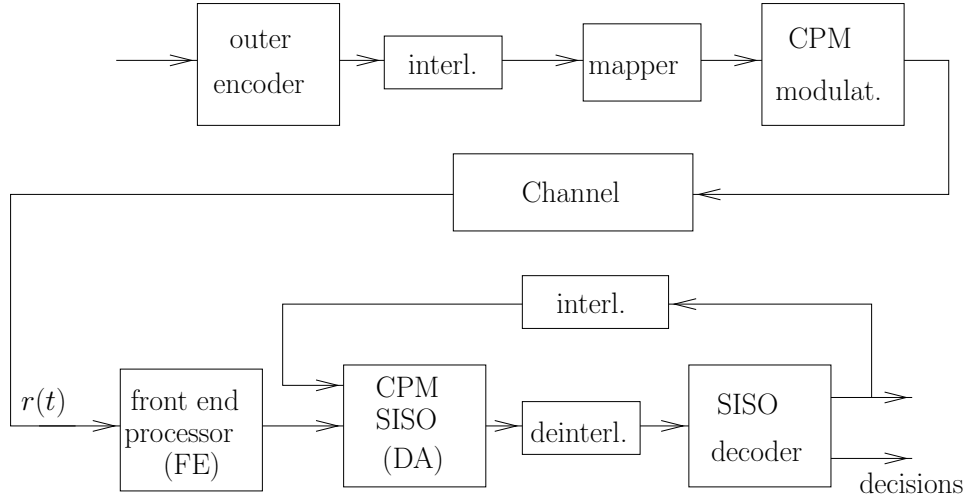


Figure 3.9: Transmitter and receiver structure for the considered SCCPM schemes.

Fig.	3.10	3.11 and 3.12	3.13 and 3.14
Pulse	2RC	2RC	3RC
M	4	4	4
h	1/4	1/4	2/7
Mapping	Gray	Natural	Natural
Outer code	CC (7,5) $r = 1/2$	CC (7,5) $r = 1/2$	CC (7,5) $r = 1/2$
Codeword length	2048	1760	2000
Interleaver	Bit or Symbol	Bit	Bit
Iterations	20	10	10

Table 3.1: Details of the considered SCCPM schemes.

SCCPM Schemes with $L=2$

In Fig. 3.10 we consider the concatenation of the above mentioned convolutional code with a quaternary 2RC modulation having $h = 1/4$. Two coded bits are mapped into one quaternary symbol by Gray mapping. The reduced-complexity soft-output detector based on the MM-P algorithm has only $p = 4$ states, whereas the optimal detector has $pM^{L-1} = 16$ states. A symbol interleaver of length 1024 symbols and a bit interleaver of length 2048 bits are considered, and 20 iterations are performed.¹¹ In [38] some considerations have been carried out about the advantages and disadvantages, in terms of convergence threshold and error floor, of symbol interleavers with respect to bit interleavers in SCCPM schemes. In particular, it has been shown that systems with symbol interleaving usually have a lower convergence threshold and a higher error floor. In our simulation, the error floor does not appear, but the result on the convergence threshold is confirmed. In any case, it can be observed that, irrespectively of the used interleaver, the MM-P algorithm, with 4 trellis states, exhibits a negligible performance loss with respect to the FC algorithm.

In Fig. 3.11 we analyze the same SCCPM scheme of Fig. 3.10, but now with a bit interleaver of length 1760 bits and natural mapping. As expected, the performance of the optimal FC algorithm with a FE based on the G decomposition and $Q = 2^{L-1} = 2$ is identical to that of the optimal BCJR algorithm with optimal FE. However, even when $Q = 1$, thus reducing the number of length- T matched filters from 16 to 12, there is no loss with respect to the optimal performance. Note that, in both cases, unless explicit reduced-search techniques are applied, there is no reduction in the number of trellis states. An automatic reduction in the number of trellis states may be obtained when the solution described in Section 3.4.2, and denoted as G-R, is adopted. As mentioned, this solution consists of taking into account only a subset of matched filters of the case $Q = 1$. In this case, the number of trellis states is halved from 16 to 8. In Fig. 3.11 the curve denoted as G-R with 16 states represent the

¹¹A dithered relative prime (DRP) interleaver [61] is used in both cases.

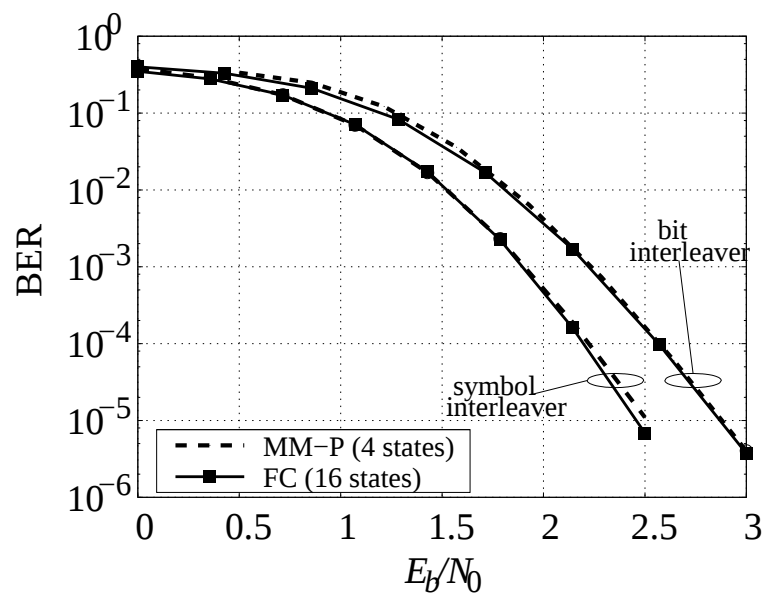


Figure 3.10: 2RC modulation with $h = 1/4$ and $M = 4$. Comparison between symbol and bit interleaver.

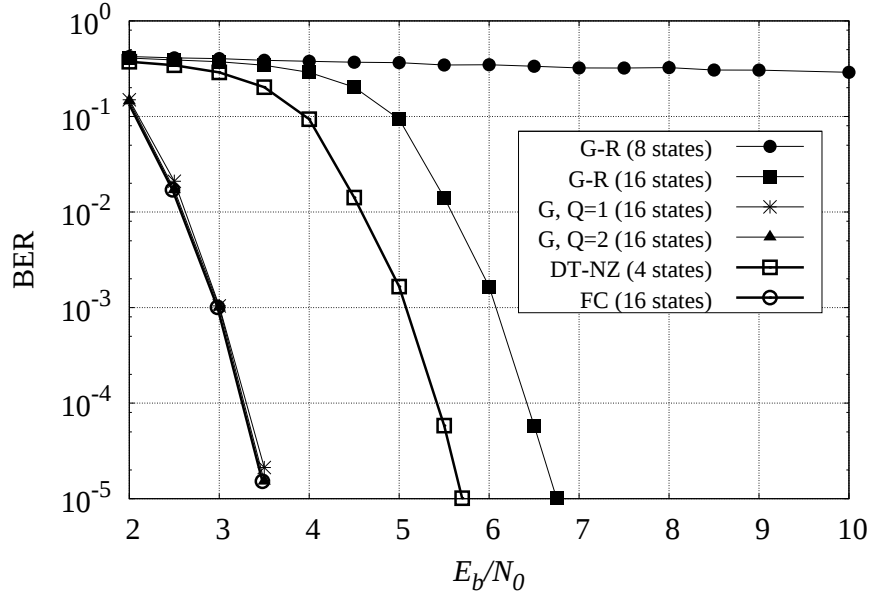


Figure 3.11: 2RC modulation with $h = 1/4$ and $M = 4$: FC algorithm with G-based FE and DT-NZ algorithm.

performance when, working on the full complexity trellis, among all matched filters of the set $Q = 1$, we consider the two filters which assure the automatic state reduction. The performance loss is significant and when we exploit the state reduction (curve denoted as G-R with 8 states) the algorithm does not work.

In Fig. 3.11, the curve related to the MM algorithm and working on a 4-state trellis is not shown since, as demonstrated in Fig. 3.10, it exhibits a negligible performance loss with respect to the FC algorithm. On the contrary, the performance of the best reduced-search algorithm, namely the DT-NZ, applied to the trellis of optimal BCJR algorithm, is also reported. Although the DT-NZ algorithm works on a 4-state trellis, the degradation is large.

In Fig. 3.12, CPM format and scenario are the same of Fig. 3.11 and per-

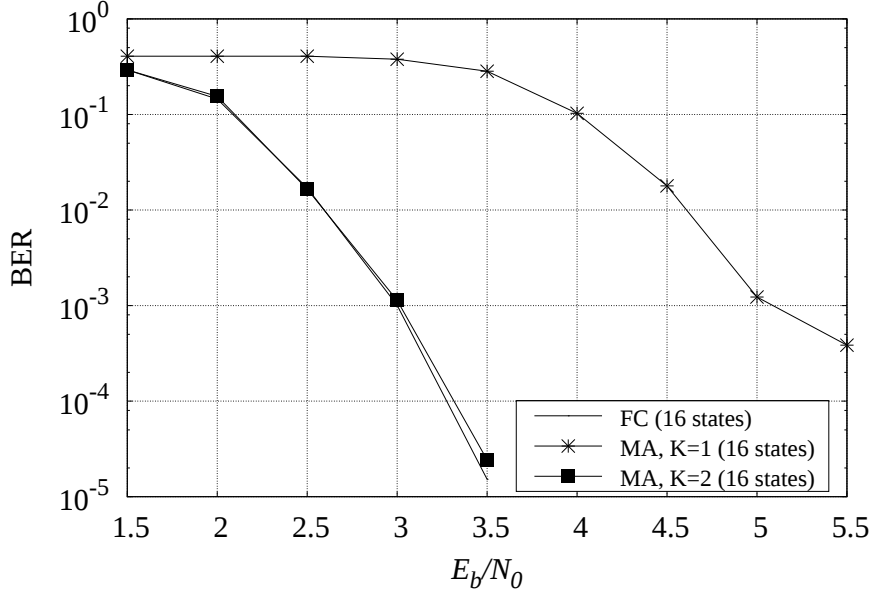


Figure 3.12: 2RC modulation with $h = 1/4$ and $M = 4$: FC algorithm with MA-based algorithm.

formance of FC algorithm with FE based on MA decomposition is analysed. While when $K = 1$ (i.e. the front end is composed by just one length- T filter) MA detector shows an heavy loss, for K values greater or equal to 2, it achieves the performance of FC algorithm with optimal FE. In other words, by means of MA-based FE the number of length- T filters can be reduced from $M^L = 16$ to 2: this solutions leads to a minimum in front end complexity although a reduction in the number of trellis states is not achieved.

It is interesting to recall that, for all considered CPM formats, the DT-NZ algorithm and the MM-P algorithm exhibit a similar performance when uncoded transmissions are considered (see Figs. 3.5-Fig. 3.8 and the relevant discussion). Hence, we can state that the DT-NZ algorithm is very effective in producing hard decisions, but definitely ineffective in producing soft deci-

sions to be exploited in iterative decoding. In Table 3.2 the number of trellis states and the front end complexity are compared for all considered detection algorithms. It is important to observe that since we are considering SCCPM schemes, the computational complexity is mainly determined by the number of trellis states since the front end stage processes each codeword just one time while the detection algorithm operates on each codeword one time every iteration.

Algorithm	Total number of length- T filters	Trellis states
FC	16	16
MM-P	7	4
G, $Q = 2$	16	16
G, $Q = 1$	12	16
MA, $K = 2$	2	16
MA, $K = 1$	1	16

Table 3.2: Front end complexity and number of trellis states for a 2RC modulation with $h = 1/4$ and $M = 4$.

In conclusion, while the FC algorithm with a FE based on the MA decomposition exhibits the lowest FE complexity (just two length- T filters are sufficient) without complexity reduction in the DA stage, the MM-P algorithm allows us to achieve the optimal performance with the minimum trellis complexity and a very reduced front end complexity. Hence, if the FC detector with a MA-based FE and the MM-P algorithm can be considered as two equally valid solutions if uncoded CPM transmissions are considered, MM-P algorithm is the best solution for all SCCPM schemes with $L = 2$.

SCCPM Schemes with $L=3$

In Fig. 3.13, we consider the concatenation of the above mentioned convolutional code with a quaternary 3RC modulation having $h = 2/7$. We adopt a DRP bit interleaver of length 2000 bits, and the coded bits are mapped into quaternary symbols by natural mapping. At the receiver side, iterative decoding is employed, performing 10 iterations. In this case, the MM-P algorithm provides, in terms of number of states, a reduction factor equal to 16 with respect to the FC algorithm but, as expected according to the discussion carried out in Section 3.3.2, the corresponding performance degradation is significant. Hence, to achieve the optimal performance, we have to take into account some selected secondary components and apply the MM-PSS algorithm, which provides a reduction factor equal to 4. Unlike the results in Fig. 3.8, related to an uncoded transmission of the same CPM format, Fig. 3.13 also shows that the application of the MM-PSS-RS algorithm causes a limited, but still significant, performance degradation. On the other hand, the MM-PSS-RS algorithm works on a trellis with only 7 states per time epoch, providing a reduction factor equal to 16 with respect to the optimal FC algorithm, and can be thus considered as a convenient tradeoff. Even in this case, the algorithms based on reduced-search techniques and directly applied to the optimal FC algorithm are not a convenient solution: the best of them, namely the DT-NZ algorithm, cannot achieve the performance of the MM-PSS-RS algorithm, even when it works on a larger trellis with 28 states.

In Fig. 3.14 we consider the same SCCPM scheme and the same scenario of Fig. 3.13 and we show the performance of the FC detection algorithm when we consider G-based and MA-based front end stages. Regarding the FE based on the G decomposition, since the solution with $Q = 1$ does not work, it is useless to assess the performance of the G-R algorithm (with reduced DA) and it is necessary to increase the value of Q . Fig. 3.14 demonstrates that the performance of the approximated G-based FE is close to the performance of the FC algorithm for $Q \geq 2$ (curves with $Q > 2$ are not reported in the figure since they coincide with the FC curve when optimal FE is considered). In other

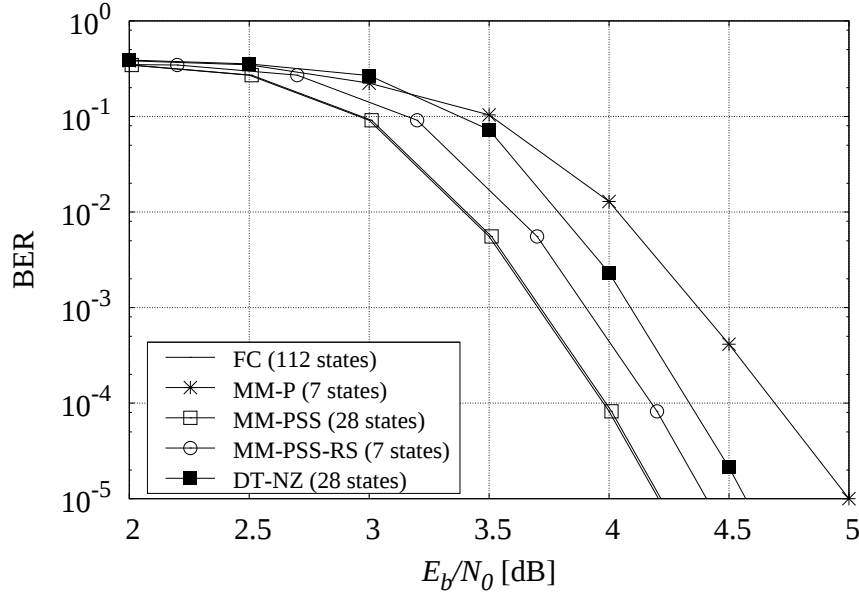


Figure 3.13: 3RC modulation with $h = 2/7$ and $M = 4$: performance of MM and DT-NZ algorithms.

words, as summarized in Table 3.3, the G decomposition in this case allows a reduction in the number of length- T matched filters from 64 to 48 with no reduction in the number of trellis states. Same conclusions hold for the MA decomposition: it is not possible to achieve a reduction in the number of trellis states but $K = 2$ length- T filters are sufficient in the front end. Also with this CPM format, MA algorithm is the detector which assures the minimum front end complexity.

In conclusion, for all the considered SCCPM schemes, the approach providing the simplest front end is that based on the CPM decomposition proposed by Moqvist and Aulin, while the most convenient solution in terms of trellis complexity results that based on the CPM decomposition proposed by Mengali and Morelli, which also reduces the FE complexity, possibly combined with

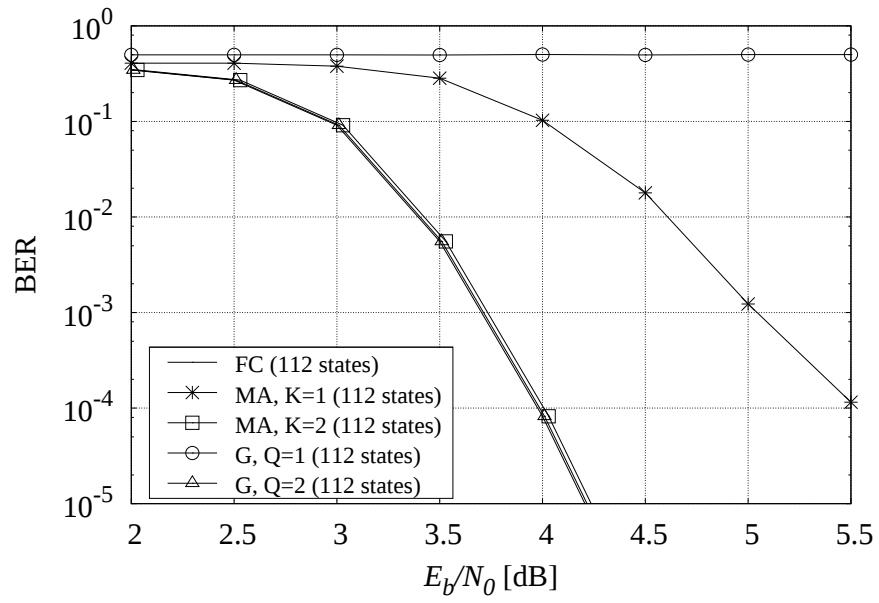


Figure 3.14: 3RC modulation with $h = 2/7$ and $M = 4$: performance of FC algorithm with G-based and MA-based FE.

Algorithm	Total number of length- T filters	Trellis states
FC	64	112
MM-P	10	7
MM-PSS	26	28
MM-PSS-RS	26	7
G, $Q = 4$	64	112
G, $Q = 3$	56	112
G, $Q = 2$	48	112
G, $Q = 1$	32	112
MA, $K = 2$	2	112
MA, $K = 1$	1	112

Table 3.3: Front end complexity and number of trellis states for a 3RC modulation with $h = 2/7$ and $M = 4$.

134 Chapter 3. CPM Reduced-Complexity Soft-Output Detection

proper techniques for reduced trellis search (see the MM-PSS-RS algorithm). For that reason, the choice of resorting to detection schemes based on MM decomposition ensures the best performance/complexity tradeoff.

Chapter 4

CPM Detection Algorithms in the Presence of Phase Noise

We consider continuous phase modulations (CPMs) in iteratively decoded serially concatenated schemes. The problem of designing low-complexity sub-optimal detection algorithms for maximum a posteriori symbol detection for CPMs, already described in Chapter 3 for the case of a coherent channel, is faced here for the case of a transmission over a typical satellite channel affected by phase noise. An ideal frequency synchronization is assumed. We describe a couple of different approaches for detection, which basically differ in the way they model the phase noise (PN) and perform the phase tracking. We also exploit the Mengali and Morelli decomposition (Section 3.3.1), which allows to approximate the CPM signal as the superposition of linearly modulated components, reducing the number of states required to describe the system.

4.1 Introduction

Two major problems in the design of practical systems employing CPM signals are the large complexity in the receiver (in terms of front end filters and number of states in the trellis) and the sensitiveness to inaccurate carrier syn-

chronization [1]. The former problem is addressed in Chapter 3 for the case of an ideal coherent receiver, showing that the algorithms ensuring the most convenient performance/complexity tradeoff are those based on the Mengali and Morelli (MM) decomposition [16, 46], which allows to reduce the number of states required to describe the system. In this Chapter, we address the latter problem, considering the transmission of a CPM signal over a typical satellite channel which introduces phase noise.

We focus on algorithms which can cope with strong phase noise, assuming an ideal frequency synchronization. Although several soft-input soft-output (SISO) detection algorithms suitable for iterative detection/decoding have been recently designed for linear modulations transmitted over channels affected by a time-varying phase (see for example [62–64] and references therein), less attention has been devoted to CPM signals. An exception is represented by [65, 66] where, based on the approach in [63, 67], joint detection and phase synchronization is performed by working on the trellis of the CPM signal or on an expanded trellis and using *multiple* phase estimators in a per-survivor fashion¹. We consider here various algorithms for MAP symbol detection, distinguishing two main approaches: the *Non-Bayesian* approach and the *Bayesian* approach. The non-Bayesian approach does not require any assumption on the statistical properties of the phase noise, since the phase noise is simply considered as a sequence of unknown parameters to be properly estimated—the above mentioned algorithms presented in [65] follow the Non-Bayesian approach. On the other hand, the rationale of the Bayesian approach consists of assuming a proper probabilistic model for the phase noise, for example the Wiener model [29, 30], and to exploit it for deriving algorithms for MAP symbol detection, as also done in [63] for the case of linear modulations. For some algorithms derived by such an approach, we also propose a trivial extension to the case of phase noise modeled by the sum of two first order auto-regressive Gaussian processes (SATMODE PN model proposed in Section 2.4.2). In this

¹As a particular case of this general approach, the use of a single phase estimator is also considered in [66] to trade performance against complexity.

4.2. Channel Model and Ideal Coherent MAP Symbol Detection 137

Chapter it is shown that Bayesian approach ensures a better performance, even if the actual phase noise does not match the assumed Wiener model. As in Chapter 3, we again resort to the MM decomposition to reduce the complexity of the front end filtering stage and the size of the trellis required to describe the modulator state. Moreover, for the derivation of the MAP symbol detection algorithm, we adopt the framework based on factor graphs (FGs) and the sum-product algorithm (SPA) [7] (see Section 1.3.2 for an overview), since the classical probabilistic arguments can not be exploited when the MM decomposition is employed (Section 3.3). In particular, to derive algorithms of practical complexity, we manage all continuous random variables in the FG by means of the *canonical distribution* approach [7,68], which allows to implement a Bayesian algorithm without dramatically increasing the complexity of the receiver with respect to the ideal coherent counterpart.

4.2 Channel Model and Ideal Coherent MAP Symbol Detection

We recall some concepts on CPM transmissions over channels affected by phase noise, already presented in Sections 2.4.1 and 2.4.3. When a transmission over a typical satellite channel is considered, the complex envelope of the received signal can be written as

$$r(t) = s(t; \mathbf{x})e^{j\theta(t)} + w(t) \quad (4.1)$$

where $s(t; \mathbf{x})$ is the CPM signal, whose complete model description is provided in Section 1.2, $\theta(t)$ is the phase noise introduced by the channel, and $w(t)$ is the complex-valued additive white Gaussian noise (AWGN) with independent components, each with two-sided power spectral density N_0 . In all cases of practical interest, the phase noise $\theta(t)$ is slowly varying with respect to the symbol interval T , so that we can approximate (4.1) in the n -th symbol interval

as follows

$$r(t) \simeq s(t; x_n, \sigma_n) e^{j\theta_n} + w(t) \quad (4.2)$$

$$= \tilde{s}(t + nT; x_n, \sigma_n) e^{j\theta_n} + w(t) \quad , \quad t \in [nT, nT + T) \quad (4.3)$$

where $\theta_n = \theta(nT)$, σ_n is defined in (1.18) and $\tilde{s}(t; x_n, \sigma_n)$ in (1.16). We also assume that the information symbols $\{x_n\}$ are independent, so that the a priori probability $P(\mathbf{x})$ of the information sequence $\mathbf{x}$ can be factorized as

$$P(\mathbf{x}) = \prod_{n=0}^{N-1} P(x_n) \quad (4.4)$$

where $P(x_n)$ is the a priori probability of x_n . Under the ideal hypothesis of a coherent receiver which perfectly knows the phase noise $\{\theta_n\}$, the optimal algorithm for MAP symbol detection turns out to be an instance of the well-known BCJR algorithm [6], and it is already described in Section 2.3.1. The only difference with respect to the scenario in which we derive optimal MAP symbol detection in Section 2.3.1, is the presence of the discrete-time phase noise θ_n ; however, since we assume an ideal coherent detection algorithm derivation, θ_n is known and it is sufficient to replace the branch metric in (2.15) by

$$F_n(x_n, \sigma_n) = \exp \left[\frac{1}{N_0} \text{Re} \left\{ y_n(x_n, \sigma_n) e^{-j\theta_n} \right\} \right] \quad (4.5)$$

necessary to take into account the channel phase.

The BCJR algorithm is characterized by the forward and backward recursions provided in (2.17) and (2.18), and by the completion stage (2.21). Let us to point out that, when the CPM modulation is such that $L > 1$, the following equivalent completion stage can be adopted [40]

$$P(x_n | \mathbf{y}) \propto \sum_{\sigma_{n+1}} S(x_n, \sigma_{n+1}) \eta_{f,n+1}(\sigma_{n+1}) \eta_{b,n+1}(\sigma_{n+1}) \quad (4.6)$$

where $S(x_n, \sigma_{n+1})$ is the trellis indicator function, equal to one if x_n and σ_{n+1} satisfy the trellis constraints and to zero otherwise. Let us point out that the

completion (4.6) is simpler than (2.21), since it does not require the computation of the branch metrics

Such algorithm will be considered as an ideal benchmark for testing the effectiveness of the algorithms presented in the remainder of this chapter, which have to perform detection in the absence of knowledge of the phase noise. In this case, among the various algorithms for MAP symbol detection, we distinguish two main approaches, namely the *non-Bayesian* one and the *Bayesian* one, whose formulations are described and discussed in Section 4.3 and Section 4.4, respectively. Detailed comparisons of the performance of the proposed algorithms will be then reported in Section 4.5.

4.3 Non-Bayesian Algorithms

4.3.1 Rationale

The non-Bayesian approach does not require any particular assumption on the statistical properties of the phase noise, since the channel phase is simply modeled as a deterministic and possibly time-varying parameter unknown to the receiver. The algorithms are based on the so-called *pseudo-coherent* strategy, which consists of resorting to the same detection algorithm as if the receiver were perfectly coherent, replacing the phase values $\{\theta_n\}$ by proper estimates $\{\hat{\theta}_n\}$ [29]. All algorithms for coherent detection described in Chapter 3 can be adopted for the pseudo-coherent approach, so that different non-Bayesian algorithms can be derived. In terms of complexity, the resulting algorithms are similar to their coherent counterparts, since the front end filters and the detection trellis are exactly the same, and the overhead computations required for phase tracking is often negligible [66]. In this work, we focus on algorithms derived from the optimal coherent receiver, whose formulation is reported in Section 4.2. The various non-Bayesian algorithms described in the following differ in the way they compute the estimates $\{\hat{\theta}_n\}$, used in the branch metrics (4.5) instead of the actual phase values $\{\theta_n\}$.

Several options for phase tracking, that is for computing the estimates $\{\hat{\theta}_n\}$,

are available [29]. The most simple option consists of generating, at each time epoch n , a unique estimate $\hat{\theta}_n$ used in the computation of the branch metric $F_n(x_n, \sigma_n)$ irrespectively of the trial values of the symbol x_n and the state σ_n . On the other hand, one can perform *per-state tracking*, propagating over the trellis one phase estimate $\hat{\theta}_n(\sigma_n)$ for each state σ_n , and computing the branch metric $F_n(x_n, \sigma_n)$ based on the related estimate $\hat{\theta}_n(\sigma_n)$ [29]. When the system is designed to work at low values of the signal-to-noise ratio and the phase noise is strong, the former approach is definitely ineffective, and it is mandatory to perform per-state tracking [29, 66]. In particular, this happens when serially concatenated schemes to be iteratively decoded are considered. In this case, another design option arises: one can use, in all iterations, the estimates evaluated during the first iteration, or, alternatively, update the estimates at every iteration, taking advantage of the information produced by the SISO decoder, so that the phase tracking is iteratively refined.

The complexity of the phase estimation is clearly increased when per-state tracking and/or iterative refining are performed. On the other hand, the complexity of whole receiver is mainly due to the recursions and the completion of the BCJR algorithm, and is thus not significantly affected by the non-Bayesian synchronization stage [66]. This motivates our choice of focusing on the algorithms which ensure the best performance, adopting both per-state tracking and iterative refining.

4.3.2 Proposed Algorithm

All proposed non-Bayesian algorithms produce the phase estimates $\{\hat{\theta}_n\}$ based on proper phase-locked loops (PLLs) [29]. We also considered the *vector tracker* strategy [69], that is the estimation of the phasor $e^{j\theta_n}$ instead of the scalar θ_n . Since we did not verify any performance improvement, as expected according to [69], we focus on PLL-based algorithms due to their lower complexity.

First, let us consider an ideal PLL, referred to as *genie-aided* (GA) PLL in the following, which knows the actual symbol x_n and the actual state σ_n

at each time epoch n . The first order² GA-PLL is governed by the following update rule [29]

$$\hat{\theta}_{n+1} = \hat{\theta}_n + \gamma \operatorname{Im} \left\{ y_n(x_n, \sigma_n) e^{-j\hat{\theta}_n} \right\} \quad (4.7)$$

where γ is the step size of the loop, and $y_n(x_n, \sigma_n)$ is defined in (2.12). We can employ a known preamble for each data symbol sequence, in order to initialize the recursion (4.7) to the value corresponding to the maximum-likelihood (ML) phase estimation over the preamble [29]. This option, with respect to a randomly-initialized PLL, ensures a much better performance [29]. Similarly to the *forward* update rule (4.7), we can define a *backward* update rule as follows

$$\hat{\theta}_{n-1} = \hat{\theta}_n + \gamma \operatorname{Im} \left\{ y_n(x_n, \sigma_n) e^{-j\hat{\theta}_n} \right\}. \quad (4.8)$$

In this case, since in many system standard (see for example the DVB-RCS) no known postamble is present after the data symbols, we can only randomly initialize the recursion (4.8). This motivates our choice of implementing the GA-PLL using the forward update rule (4.7) only. As a benchmark for all non-Bayesian algorithms, we can consider the BCJR algorithm described in Section 4.2, replacing the actual phase offsets $\{\theta_n\}$ in the branch metrics (4.5) by the estimates $\{\hat{\theta}_n\}$ produced by the GA-PLL. It is clear that any PLL-based algorithm cannot perform better than this ideal algorithm.

Let us now describe practical algorithms, considering first the implementation of the forward recursion (2.17). As stated before, we consider per-state phase tracking, computing one phase estimate $\hat{\theta}_n(\sigma_n)$ for each state σ_n at each time epoch n , and exploiting $\hat{\theta}_n(\sigma_n)$ in the computation of the branch metrics related to all trellis paths extending from σ_n . For all trial values of the initial state σ_0 , the estimate $\hat{\theta}_0(\sigma_0)$ is initialized to the same value, possibly the ML phase estimation over the preamble. The point is how to derive an update rule similar to (4.7), taking into account that the actual symbol x_n and the actual state σ_n are unknown to the receiver. At any given time epoch n ,

²It is known [29] that higher order PLLs do not provide any performance improvement when the frequency offset is not present, as in (4.3).

let $\check{\theta}_{n+1}(x_n, \sigma_n)$ be the phase estimate at time epoch $n + 1$ related to the trial symbol x_n and the trial state σ_n , defined as

$$\check{\theta}_{n+1}(x_n, \sigma_n) = \hat{\theta}_n(\sigma_n) + \gamma \operatorname{Im} \left\{ y_n(x_n, \sigma_n) e^{-j\hat{\theta}_n(\sigma_n)} \right\} \quad (4.9)$$

simply extending (4.7) to the case of per-state tracking.³ It is worth to notice that, for each value of the future state σ_{n+1} , every couple (x_n, σ_n) compatible with σ_{n+1} carries a different estimate of the actual phase offset θ_{n+1} . Now, the problem is how to combine the various estimates $\{\check{\theta}_{n+1}(x_n, \sigma_n)\}$ to obtain the estimate $\hat{\theta}_{n+1}(\sigma_{n+1})$ required to perform the forward recursion. In practice, we have to define a criterion for properly weighting the different estimates. A couple of different solutions are discussed in the following. Both of them require the definition, at each time epoch n and for each trial symbol x_n and trial state σ_n , of the coefficient

$$W_{n+1}(x_n, \sigma_n, \sigma_{n+1}) = T(x_n, \sigma_n, \sigma_{n+1}) F_n(x_n, \sigma_n) \eta_{f,n}(\sigma_n) \quad (4.10)$$

which equals the contribution given by the considered trellis path to the forward metric $\eta_{f,n+1}(\sigma_{n+1})$ related to the corresponding future state σ_{n+1} (see (2.17)). We point out that the computation of the terms (4.10) is done while performing the forward recursion (2.17), and is not to be considered an overhead due to phase tracking. The simplest solution, exploited in [66], consists of setting $\hat{\theta}_{n+1}(\sigma_{n+1})$ equal to the estimate $\check{\theta}_{n+1}(x_n, \sigma_n)$ related to the trellis path giving the largest contribution to the forward metric $\eta_{f,n+1}(\sigma_{n+1})$. Formally, for each given value of σ_{n+1} , we obtain

$$\hat{\theta}_{n+1}(\sigma_{n+1}) = \check{\theta}_{n+1}(\hat{x}_n, \hat{\sigma}_n) \quad (4.11)$$

with

$$(\hat{x}_n, \hat{\sigma}_n) = \arg \max_{(x_n, \sigma_n)} \{W_{n+1}(x_n, \sigma_n, \sigma_{n+1})\} . \quad (4.12)$$

³The phase estimates $\{\check{\theta}_{n+1}(x_n, \sigma_n)\}$ can be also used in the computation of the branch metrics $\{F_n(x_n, \sigma_n)\}$, instead of the estimates $\{\hat{\theta}_n(\sigma_n)\}$. In this case, *per-branch* phase tracking is performed, since different phase estimates for each trellis path are exploited.

Alternatively, we can use the terms $\{W_{n+1}(x_n, \sigma_n, \sigma_{n+1})\}$ as weight factors for the computation of the estimates $\{\hat{\theta}_{n+1}(\sigma_{n+1})\}$ as averages of the estimates $\{\check{\theta}_{n+1}(x_n, \sigma_n)\}$, as follows

$$\begin{aligned}\hat{\theta}_{n+1}(\sigma_{n+1}) &= \frac{\sum_{x_n} \sum_{\sigma_n} W_{n+1}(x_n, \sigma_n, \sigma_{n+1}) \check{\theta}_{n+1}(x_n, \sigma_n)}{\sum_{x_n} \sum_{\sigma_n} W_{n+1}(x_n, \sigma_n, \sigma_{n+1})} \\ &= \frac{\sum_{x_n} \sum_{\sigma_n} W_{n+1}(x_n, \sigma_n, \sigma_{n+1}) \check{\theta}_{n+1}(x_n, \sigma_n)}{\eta_{f,n+1}(\sigma_{n+1})}. \quad (4.13)\end{aligned}$$

According to the notation in [29] and widely adopted in the literature, both the update rules (4.11) and (4.13) are instances of trellis-based phase tracking, but the former is decision-directed, while the latter is soft-decision directed. Interestingly, the update rule (4.11) can be seen as an approximation of (4.13), obtained by neglecting all terms $\{W_{n+1}(x_n, \sigma_n, \sigma_{n+1})\}$ except the largest one.⁴ Hence, the latter strategy is expected to be more effective. On the other hand, the average operation in (4.13) clearly requires a computational overhead with respect to (4.11). The simulation results reported in Section 4.5 show that such an increase in complexity is often justified by a significant performance improvement. We will refer to the update rule (4.11) simply as PLL update, whereas we will refer to the update rule (4.13) as weighted PLL (W-PLL) update.

So far, we only considered the application of per-state phase tracking to the forward recursion (2.17), but the same strategies can be adopted while performing the backward recursion (2.18). It is indeed trivial to derive, based on the GA backward update rule (4.8), the backward counterparts of (4.9), (4.11), and (4.13). In practice, a backward recursion with joint phase tracking can be performed similarly to the forward recursion. It is worth noticing that this approach leads to a set of phase estimates produced during the backward recursion which are not constrained to be equal to those produced during the forward recursion. In this case, the completion stage (2.21) is not unambiguously defined, since the branch metrics $\{F_n(x_n, \sigma_n)\}$ can be computed based

⁴From a conceptual viewpoint, the above mentioned differences are similar to those occurring between the Viterbi algorithm and the BCJR algorithm [6, 70].

on the phase estimates produced during the forward recursion as well as those produced during the backward recursion. This issue has been addressed in [66], where the authors investigate several ways for combining the phase estimates produced in the recursions. We propose an alternative solution, which consists of performing the completion stage based on (4.6) instead of (2.21).⁵ The key point is that the implementation of (4.6) allows to perform the completion stage without computing the branch metrics, so that no phase estimate has to be exploited. Once computed the forward/backward state metrics, this approach does not introduce any further approximation in the completion stage, as instead done by all algorithms which perform the completion stage explicitly computing the branch metrics based on phase estimates. Hence, this is expected to be the most effective solution when the two recursions perform phase tracking independently each other.

As a final design option, we consider the possibility of exploiting the phase estimates produced during the forward recursion while performing the backward recursion and the completion stage, so that per-state phase tracking is performed only once. This approach is suggested in [66], where it is shown that the bidirectional phase estimation suffers the rotational invariance of the CPMs, so that *phase slips* occur more frequently.⁶ Then, even if we accept that we have to perform monodirectional phase tracking, it is natural to wonder if there exists a physical reason for doing this during the forward recursion and not in the backward recursion. A significant advantage in favour of performing forward phase tracking is due to the fact that, as stated before, the estimates can be reliably initialized to the value produced by a ML phase estimation over the preamble, while the same cannot be always done for the backward phase tracking since frequently no postamble follows the data stream. Hence, when

⁵As explained in Section 4.2, the completion (4.6) can be adopted only when the considered CPM format has correlation length larger than one. This condition holds for high spectral efficiency CPM formats we will consider.

⁶In [66], it is also shown that the bidirectional phase estimation is to be preferred when the phase offsets are time-invariant. Anyway, this is not the case of the satellite channel considered in this chapter.

considering monodirectional phase tracking, we will always refer to forward phase tracking.

4.4 Bayesian Algorithms

The rationale of the Bayesian approach consists of assuming a proper probabilistic model for the phase noise $\{\theta_n\}$, and of exploiting it for deriving algorithms for MAP symbol detection. The Bayesian approach does not require any particular decomposition of the CPM signal, and thus it could be directly applied to a receiver exploiting the exact signal model (1.1). In other words it can be applied considering the full number of CPM states. In particular, in Section 2.4.3 three algorithms have been derived following such an approach (denoted to as DP, D-DP and I-DP). However, even if such algorithms can be useful for information rate computation, they are not suitable to be implemented in practical detectors, due to the large complexity of the full-complexity trellis (see Chapter 3). Thus, we focus on the application of the Bayesian approach to a receiver which exploits the MM decomposition described in Section 3.3.1, based on the principal components only, when $L = 2$. When the CPM modulation is such that $L > 2$, the MM decomposition based on the principal components only is ineffective (Chapter 3). In this case, we will consider also some selected secondary components.

4.4.1 Derivation of the Algorithms

We model the phase noise as a first order process. In particular, we choose a discrete-time Wiener process [29, 30] with incremental variance over a symbol interval equal to σ_Δ^2 , implying the following update rule

$$\theta_{n+1} = \theta_n + \Delta_n \quad (4.14)$$

where $\{\Delta_n\}$ are independent and identically distributed Gaussian random variables with mean zero and standard deviation σ_Δ , and θ_0 is a random variable

uniformly distributed in $[0, 2\pi)$. It is worth noting that, having assumed zero-mean increments Δ_n , we practically assumed a perfect frequency synchronization, according to the channel model discussed in Section 4.2. We will show that, provided that the parameter σ_Δ^2 is properly set, the model (4.14) ensures a good approximation of several actual phase noise processes, so that the designed Bayesian algorithms are effective even when the phase noise is not a Wiener process, as proved by the simulation results reported in Section 4.5.

We focus on the application of the Bayesian approach to a receiver which exploits the described MM decomposition and in this case, a set of sufficient statistics for MAP symbol detection is given by [53]

$$p_{k,n} = \int_{-\infty}^{+\infty} r(t) p_k(t - nT) dt, \quad k \in \{0, 1, \dots, K\}, \quad n \in \{0, 1, \dots, N - 2\} \quad (4.15)$$

that is the output, sampled at symbol rate, of a bank of filters matched to the principal pulses $p_k(t)$ ⁷ ($K = M - 2$ is the number of principal pulses).

We now derive the algorithm for MAP symbol detection. To this purpose, we first factorize the joint distribution $p(\mathbf{x}, \boldsymbol{\psi}, \boldsymbol{\theta} | \mathbf{r})$ as

$$p(\mathbf{x}, \boldsymbol{\psi}, \boldsymbol{\theta} | \mathbf{r}) \propto P(\mathbf{x}) P(\boldsymbol{\psi} | \mathbf{x}) p(\boldsymbol{\theta}) p(\mathbf{r} | \mathbf{x}, \boldsymbol{\theta}, \boldsymbol{\psi}) \quad (4.16)$$

where $\boldsymbol{\theta} = \{\theta_n\}$ and $\mathbf{r}$ is the vector representation of the received signal onto an orthonormal base, while ψ_n , defined in (3.9), is the integer representation of the pseudosymbol $a_{0,n}$ which represents the state of the CPM signal when approximated by just the principal components of the MM decomposition (see Section 3.3.1). In particular, ψ_n belongs from the alphabet $\{0, 1, \dots, p-1\}$ and hence it can take p values. Then, after defining

$$H_n(x_n, \psi_{n-1}, \theta_n) = \exp \left\{ \frac{1}{N_0} \operatorname{Re} \left[\sum_{k=0}^{M-2} p_{k,n} a_{k,n}^* (x_n, \psi_{n-1}) e^{-j\theta_n} \right] \right\} \quad (4.17)$$

⁷Rigorously, the samples $\{p_{k,n}\}$ give a set of sufficient statistics only if θ_n can be considered constant over $L + 1$ symbol intervals [53]. Such an approximation holds in all scenarios of interest.

we can further factorize the terms in (4.16) as follows

$$P(\mathbf{x}) = \prod_{n=0}^{N-1} P(x_n) \quad (4.18)$$

$$P(\boldsymbol{\psi}|\mathbf{x}) = P(\psi_{-1}) \prod_{n=0}^{N-1} I_n(\psi_n, \psi_{n-1}, x_n) \quad (4.19)$$

$$p(\boldsymbol{\theta}) = p(\theta_0) \prod_{n=1}^{N-1} p(\theta_n|\theta_{n-1}) \quad (4.20)$$

$$p(\mathbf{r}|\mathbf{x}, \boldsymbol{\theta}, \boldsymbol{\psi}) \propto \prod_n H_n(x_n, \psi_{n-1}, \theta_n) \quad (4.21)$$

where $I_n(\psi_n, \psi_{n-1}, x_n)$ is an indicator function, equal to 1 if x_n , ψ_{n-1} , and ψ_n satisfy the constraint (3.9) and to zero otherwise. Finally, taking into account that $p(\theta_n|\theta_{n-1})$ is a Gaussian pdf in θ_n with mean θ_{n-1} and variance σ_Δ^2 , we obtain⁸

$$p(\mathbf{x}, \boldsymbol{\psi}, \boldsymbol{\theta}|\mathbf{r}) \propto P(\psi_{-1})p(\theta_0) \prod_{n=0}^{N-1} I_n(\psi_n, \psi_{n-1}, x_n)P(x_n) \\ H_n(x_n, \psi_{n-1}, \theta_n)g(\theta_{n-1}, \sigma_\Delta^2; \theta_n) . \quad (4.22)$$

The FG corresponding to (4.22) has cycles. However, by clustering [7] the variables θ_n and ψ_{n-1} into a global system state definition $\varepsilon_n = (\psi_{n-1}, \theta_n)$, we obtain the FG in Fig. 4.1, which is cycle-free. Hence, by applying to it the SPA with a *non-iterative* forward-backward schedule, we obtain, except for the above mentioned approximations, the exact a posteriori probabilities $P(x_n|\mathbf{r})$ necessary to implement the MAP symbol detection strategy. With reference to the messages in the figure, the resulting forward-backward algorithm is

⁸ A Gaussian pdf in the variable x , with mean η and variance ρ^2 , is denoted as $g(\eta, \rho^2; x)$. Note that, since the channel phase is defined modulo 2π , the probability density function $p(\theta_{n+1}|\theta_n)$ is not rigorously Gaussian, but it can practically be considered Gaussian since typically $\sigma_\Delta \ll 2\pi$.

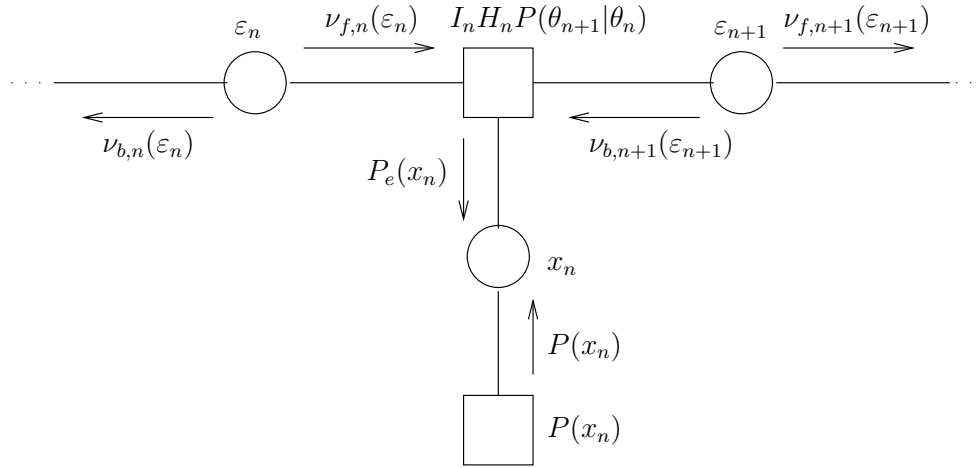


Figure 4.1: Factor graph corresponding to eqn. (4.22).

characterized by the following recursions and completion

$$\nu_{f,n+1}(\psi_n, \theta_{n+1}) = \sum_{x_n} P(x_n) \int \nu_{f,n}(\check{\psi}_{n-1}, \theta_n) H_n(x_n, \check{\psi}_{n-1}, \theta_n) g(\theta_n, \sigma_\Delta^2; \theta_{n+1}) d\theta_n \quad (4.23)$$

$$\nu_{b,n}(\psi_{n-1}, \theta_n) = \sum_{x_n} P(x_n) H_n(x_n, \psi_{n-1}, \theta_n) \int \nu_{b,n+1}(\check{\psi}_n, \theta_{n+1}) g(\theta_n, \sigma_\Delta^2; \theta_{n+1}) d\theta_{n+1} \quad (4.24)$$

$$P_e(x_n) = \sum_{\psi_n} \iint \nu_{f,n}(\check{\psi}_{n-1}, \theta_n) \nu_{b,n+1}(\psi_n, \theta_{n+1}) H_n(x_n, \check{\psi}_{n-1}, \theta_n) \cdot g(\theta_n, \sigma_\Delta^2; \theta_{n+1}) d\theta_n d\theta_{n+1} \quad (4.25)$$

where $\check{\psi}_{n-1} = [\psi_n - \bar{x}_n]_p$, $\check{\psi}_n = [\psi_{n-1} + \bar{x}_n]_p$ and $\bar{x}_n$ is the integer representation for symbol x_n we have introduced in (1.11). Since we do not assume to have any information on the initial/final modulator state nor channel phase, all metrics are initialized to the same value, which is irrelevant provided that it is positive.

A proof of the following properties is given in [53].

Property 1. For each $\ell = 0, \dots, p-1$,

$$\nu_{f,n}([\psi_{n-1} + \ell]_p, \theta_n) = \nu_{f,n}(\psi_{n-1}, \theta_n + 2\pi h\ell) \quad (4.26)$$

$$\nu_{b,n}([\psi_{n-1} + \ell]_p, \theta_n) = \nu_{b,n}(\psi_{n-1}, \theta_n + 2\pi h\ell). \quad (4.27)$$

Property 2. The extrinsic information $P_e(x_n)$ in (4.25) is given by the sum of p terms (one for each value of ψ_n), all assuming the same value, i.e., they do not depend on ψ_n , for any given x_n .

From the first property it follows that it is not necessary to evaluate and store all pdfs $\nu_{f,n}(\psi_{n-1}, \theta_n)$ and $\nu_{b,n}(\psi_{n-1}, \theta_n)$ for different values of ψ_{n-1} . It is for instance sufficient to evaluate $\bar{\nu}_{f,n}(\theta_n) = \nu_{f,n}(\psi_{n-1} = 0, \theta_n)$ and $\bar{\nu}_{b,n}(\theta_n) = \nu_{b,n}(\psi_{n-1} = 0, \theta_n)$. From the second property, it follows that only one term in (4.25) needs to be evaluated. The MAP symbol detection strategy can therefore be simplified as follows

$$\begin{aligned} \bar{\nu}_{f,n+1}(\theta_{n+1}) = & \sum_{x_n} P(x_n) \int \bar{\nu}_{f,n}(\theta_n - 2\pi h\bar{x}_n) H_n(x_n, \psi_{n-1} = [-\bar{x}_n]_p, \theta_n) \\ & \cdot g(\theta_n, \sigma_{\Delta}^2; \theta_{n+1}) d\theta_n \end{aligned} \quad (4.28)$$

$$\begin{aligned} \bar{\nu}_{b,n}(\theta_n) = & \sum_{x_n} P(x_n) H_n(x_n, \psi_{n-1} = 0, \theta_n) \int \bar{\nu}_{b,n+1}(\theta_{n+1} + 2\pi h\bar{x}_n) \\ & \cdot g(\theta_n, \sigma_{\Delta}^2; \theta_{n+1}) d\theta_{n+1} \end{aligned} \quad (4.29)$$

$$\begin{aligned} P_e(x_n) \propto & \iint \bar{\nu}_{f,n}(\theta_n) \bar{\nu}_{b,n+1}(\theta_{n+1} + 2\pi h\bar{x}_n) H_n(x_n, \psi_{n-1} = 0, \theta_n) \\ & \cdot g(\theta_n, \sigma_{\Delta}^2; \theta_{n+1}) d\theta_n d\theta_{n+1}. \end{aligned} \quad (4.30)$$

In practice, in each recursion the pdf of the channel phase is estimated, and these estimates are then exploited in the final completion.

This algorithm involves integration and computation of continuous pdfs in the recursions and in the completion stage, and it is thus not suited for

direct implementation. A solution for this problem is suggested in [68] and consists of the use of *canonical distributions*, i.e., the pdfs $\bar{\nu}_{f,n}(\theta_n)$ and $\bar{\nu}_{b,n}(\theta_n)$ computed by the algorithm are constrained to belong to proper “canonical” families, characterized by a simple parametrization. Hence, over the FG just the parameters of the pdf are propagated, rather than the pdf itself. Two algorithms based on this approach will be now described.

4.4.2 Proposed Algorithm

First Algorithm

A straightforward implementation of the algorithm is obtained by discretizing the channel phase: only the D phase values $\{2\pi i/D\}_{i=0}^{D-1}$ are considered [62,64]. In this way, the pdfs $\bar{\nu}_{f,n}(\theta_n)$ and $\bar{\nu}_{b,n}(\theta_n)$ become pmfs and the integrals in (4.28), (4.29), and (4.30) become summations. This algorithm will be referred to as “per-state discretized-phase algorithm” (*per-state DP-algorithm*). In order to distinguish it from that one described in Section 2.4.3 based on phase noise discretization and Rimoldi [2] CPM signal representation, we denote the algorithm as MM-DP algorithm (where MM suggests it is derived based on the MM decomposition). When the number D of discretization levels is large enough, the MM-DP-algorithm becomes optimal (in the sense that its performance approaches that of the exact algorithm), and it can thus be considered as a performance benchmark for any other Bayesian algorithm. Due to the nature of the CPM trellis, a simple implementation of the MM-DP-algorithm can be achieved only if D is constrained to be a multiple of $2p$ when r is odd, or a multiple of p when r is even. Moreover, to further simplify the algorithm, we implement it neglecting all terms $\{g(\theta_n, \sigma_\Delta^2; \theta_{n+1})\}$ which fall below a proper threshold value—in practice, we accept the most likely phase transitions only. In the simulation results reported in Section 4.5, the value of D is always constrained to be a multiple of $2p$, and the threshold value is set such that all but three phase transitions from each value of θ_n are neglected. In conclusion, the algorithm works on a trellis with D states per time epoch

and with $3M$ branches extending from each node.

Second Algorithm

By observing that the Tikhonov distribution ensures a very good performance with a low complexity when used as a canonical distribution in detection algorithms for phase noise channels [64], the pdfs $\bar{\nu}_{f,n}(\theta_n)$ and $\bar{\nu}_{b,n}(\theta_n)$ are constrained to have the following expressions

$$\bar{\nu}_{f,n}(\theta_n) = \sum_{m=0}^{p-1} q_{f,n}^{(m)} t(z_{f,n}; \theta_n - 2\pi h m) \quad (4.31)$$

$$\bar{\nu}_{b,n}(\theta_n) = \sum_{m=0}^{p-1} q_{b,n}^{(m)} t(z_{b,n}; \theta_n - 2\pi h m) \quad (4.32)$$

where, for each time index n , $\{q_{f,n}^{(m)}, m = 0, 1, \dots, p-1\}$ ($\{q_{b,n}^{(m)}, m = 0, 1, \dots, p-1\}$) and $z_{f,n}$ ($z_{b,n}$) are, respectively, p real coefficients and one complex coefficient, and $t(z; \theta)$ is a Tikhonov distribution with complex parameter z , defined as

$$t(z; \theta) = \frac{e^{\operatorname{Re}[ze^{-j\theta}]}}{2\pi \operatorname{I}_0(|z|)} \quad (4.33)$$

$\operatorname{I}_0(x)$ being the zero-th order modified Bessel function of the first kind. Note that, in order to obtain pdfs, the relation $\sum_{m=0}^{p-1} q_{f/b,n}^{(m)} = 1$ must hold.

Three approximations are now introduced for deriving the proposed detection algorithm.

i. The convolution of a Tikhonov and a Gaussian pdf is still a Tikhonov pdf, with a modified complex parameter [64], i.e.,

$$\int t(z; x) g(x, \rho^2; y) dx \simeq t\left(\frac{z}{1 + \rho^2|z|}; y\right). \quad (4.34)$$

ii. Since, for large arguments, $\operatorname{I}_0(x) \simeq e^x$, we approximate

$$e^{\operatorname{Re}[ze^{-j\theta}]} \simeq 2\pi e^{|z|} t(z; \theta). \quad (4.35)$$

iii. Let z be a complex number, $\{u_m, m = 0, 1, \dots, p-1\}$ a set of complex numbers, and $\{q_m, m = 0, 1, \dots, p-1\}$ a set of real numbers such that $\sum_m q_m = 1$, then the following approximation holds

$$\sum_m q_m t \left(z e^{j2\pi h m} + u_m; \theta \right) \simeq \sum_m q_m t \left(w e^{j2\pi h m}; \theta \right) \quad (4.36)$$

where $w = z + \sum_\ell q_\ell u_\ell e^{-j2\pi h \ell}$. The approximation (4.36) is good especially when $|z|$ is sufficiently larger than each $|u_m|$, or when there exists $\bar{m}$ such that $q_{\bar{m}} \gg q_m, \forall m \neq \bar{m}$.

In order to illustrate the derivation of the proposed algorithm, we consider the case of a binary modulation, i.e., $M = 2$, and hence $K = 1$.⁹ In this case

$$\frac{1}{N_0} \sum_{k=0}^{K-1} p_{k,n} a_{k,n}^* = \frac{1}{N_0} p_{0,n} a_{0,n}^* = y_n e^{-j2\pi h \psi_n} \quad (4.37)$$

where we define $y_n = \frac{1}{N_0} p_{0,n} e^{j\pi h(n+1)(M-1)}$. We now derive the reduced-complexity forward recursion. Substituting (4.17) in (4.28) and assuming that $\bar{\nu}_{f,n-1}(\theta_{n-1})$ has the canonical expression (4.31), we obtain

$$\begin{aligned} \bar{\nu}_{f,n+1}(\theta_{n+1}) = \sum_{x_n} P(x_n) \sum_{m=0}^{p-1} q_{f,n}^{(m)} \int g(\theta_n, \sigma_\Delta^2; \theta_{n+1}) \\ \cdot t(z_{f,n}; \theta_n - 2\pi h(\bar{x}_n + m)) e^{\text{Re}[y_n e^{-j\theta_n}]} d\theta_n. \end{aligned} \quad (4.38)$$

Now, changing the first summation index in $\ell = m + \bar{x}_n$, using (4.34) and (4.35), discarding irrelevant multiplicative factors, and neglecting $|y_n|$ with respect to $|z_{f,n}|$, we have

$$\begin{aligned} \bar{\nu}_{f,n+1}(\theta_{n+1}) = \sum_\ell \left[\sum_{x_n} P(x_n) q_{f,n}^{([\ell - \bar{x}_n]_p)} \right] \\ e^{|z_{f,n} + y_n e^{-j2\pi h \ell}|_t} \left(\frac{z_{f,n} e^{j2\pi h \ell} + y_n}{1 + \sigma_\Delta^2 |z_{f,n}|}; \theta_n \right). \end{aligned} \quad (4.39)$$

⁹The generalization to non-binary cases is straightforward from a conceptual viewpoint.

The resulting $\bar{\nu}_{f,n+1}(\theta_{n+1})$ is not in the constrained form (4.31). However, by applying the approximation (4.36), we obtain the following updating equations for the parameters of the canonical distribution (4.31)

$$q_{f,n+1}^{(\ell)} \propto \left[\sum_{x_n} P(x_n) q_{f,n}^{([\ell-\bar{x}_n]_p)} \right] e^{|z_{f,n} + y_n e^{-j2\pi h \ell}|} \quad (4.40)$$

$$z_{f,n+1} = \frac{z_{f,n} + y_n \sum_m q_{f,n+1}^{(m)} e^{-j2\pi h m}}{1 + \sigma_\Delta^2 |z_{f,n}|}. \quad (4.41)$$

It is worth noticing that, before computing the coefficient $z_{f,n+1}$, we have to normalize the p real coefficients $q_{f,n+1}^{(\ell)}$ so that their sum is unitary. Since we do not assume to have any information on the initial phase offset, the coefficients are initialized according to

$$\begin{aligned} q_{f,0}^{(\ell)} &= 1/p \\ z_{f,0} &= 0. \end{aligned}$$

In addition, since at the first step of the forward recursion the approximation (4.36) does not hold, we use the following values for the recursive coefficients at time $n = 1$

$$\begin{aligned} q_{f,1}^{(\ell)} &= \delta_\ell \\ z_{f,1} &= \frac{y_0}{1 + \sigma_\Delta^2 |y_0|} \end{aligned}$$

where δ_ℓ represents the Kronecker delta.

Similarly, it is also possible to find the backward recursive equations. We report the final expressions only:

$$\begin{aligned} u(x_n; m) &= \frac{P(x_n) q_{b,n+1}^{([m]_p)} e^{|z'_{b,n+1} + y_n e^{-j2\pi h m}|}}{\sum_\ell \sum_x P(x) q_{b,n+1}^{(\ell)} e^{|z'_{b,n+1} + y_n e^{-j2\pi h \ell}|}} \\ q_{b,n}^{(\ell)} &= \sum_{x_n} u(x_n; \ell + \bar{x}_n) \end{aligned} \quad (4.42)$$

$$z_{b,n} = z'_{b,n+1} + y_n \sum_m \left(\sum_{x_n} u(x_n; m) \right) e^{-j2\pi h m} \quad (4.43)$$

where $z'_{b,n+1} = \frac{z_{b,n+1}}{1+\sigma_{\Delta}^2|z_{b,n+1}|}$ and coefficients $u(\cdot; \cdot)$ have been introduced to simplify the notation (they do not need to be stored, since they are not involved in the completion stage). The initial values of the backward coefficients are

$$\begin{aligned} q_{b,N}^{(\ell)} &= 1/p & q_{b,N-1}^{(\ell)} &= \begin{cases} P(x_{N-1} = -1) & \ell = 0 \\ P(x_{N-1} = +1) & \ell = p-1 \\ 0 & \text{else} \end{cases} \\ z_{b,N} &= 0 & z_{b,N-1} &= y_{N-1} \end{aligned}$$

Finally, substituting (4.31) and (4.32) into (4.30) and discarding irrelevant constants, the extrinsic information is evaluated as

$$P_e(x_n) \propto \sum_{\ell} \sum_m q_{f,n}^{(\ell)} q_{b,n+1}^{(m)} \cdot \exp \left(\left| z_{f,n} e^{j2\pi h(\ell + \bar{x}_n)} + z'_{b,n+1} e^{j2\pi h m} + y_n \right| \right). \quad (4.44)$$

In summary, this detection algorithm is based on three steps: a forward recursion in which, for each time epoch n , one complex and p real coefficients are evaluated based on (4.40) and (4.41), a backward recursion, based on (4.42) and (4.43), which proceeds similarly, and finally a completion (4.44), which consists of the sum of p^2 terms (although only a small amount of them can be numerically significant). This algorithm will be referred to as *per-state CBC-algorithm* (MM-CBC).

4.4.3 Extension of the Algorithms

When the considered CPM format is such that $L > 2$, several simulation results of Chapter 3 show that the reduced-complexity detector based on the principal components only exhibits a significant performance loss with respect to the optimal detector when the ideal coherent receiver is considered. This is due to the presence of some non-principal components of the MM representation, denoted to *secondary components*, with a non-negligible power. In particular, for all considered CPM formats with $L=3$, we found eight secondary pulses $\{p_k(t)\}$ with a non-negligible power (Section 3.3.2). When these secondary components

are taken into account, the functions H_n in (4.17) depend also on x_{n-1} , since the symbols $\{a_{k,n}\}$ related to the considered selected secondary components require the knowledge of the past information symbol x_{n-1} [16]. Hence, extending the derivation described before, we obtain a forward-backward detection algorithm in which the trellis state is defined as $(\theta_n, \psi_{n-1}, x_{n-1})$. Again, a practical implementation of such an algorithm requires to resort to approaches based on canonical distributions. In particular, the phase discretization leads to the *extended per-state DP-algorithm*, while the Tikhonov parametrization leads to the *extended per-state CBC-algorithm*. While the MM-DP extension is trivial, this is not the case of the MM-CBC-algorithm. However we do not report the details of such algorithm since as we will see from numerical results of Section 4.5, it always exhibits worse performances. In the *extended per-state DP-algorithm* the number of trellis state is DM , since we have D phase levels for each possible value of x_{n-1} . On the other hand, the *extended per-state CBC-algorithm* at each time epoch n has to compute pM real coefficients and one complex coefficient per recursion. In conclusion, the extended versions of both algorithms are about M -times more complex than the related counterparts based on the principal components only. This increase in complexity can be faced by means of reduced-search techniques addressed to the MM decomposition, as explained in Section 3.3.2. In this case, the complexity is the same as that of the basic algorithms in Section 4.4.2, but some performance degradation is expected (Chapter 3).

4.4.4 Bayesian Algorithms based on double-AR1 Phase Noise Model

All Bayesian algorithms derived until now, are carried out by considering a first order phase noise model, i.e., a discrete-time Wiener model. On the other hand, sometimes channels of practical interest are affected by a phase noise which can be modeled as the sum of two first order auto-regressive filters. This is the case of the SATMODE phase noise, which impairs satellite communications characterized by low-cost equipment, and whose double-AR1 model is

described in Section 2.4.2. In Chapter 2, numerical results show that the detector derived until now, based on just a single-AR1 PN generation, exhibits a significant performance degradation when employed on double-AR1 PN generation (see Section 2.4.4). Thus, it can be useful to derive some Bayesian algorithms based on such double-AR1 PN model, as in the case of the D-DP-BCJR algorithm described in Section 2.4.3. Section 2.4.4, demonstrate that the information rate loss achieved by these algorithms with respect to the coherent case, is close to zero. However, the good result in performance is reached at the price of an increased trellis state complexity since the system state is composed by the union of the CPM state, with the state of both the two phase noise components. Thus these algorithms cannot be taken into account for systems of practical interest. Finally in Section 2.4.3, we also propose an improved version of the DP algorithm (the so called I-DP), suitable when considering double-AR1 PN with a very fast component, which can be approximated as jitter, independent from sample to sample. The trellis of the I-DP algorithm is based on the full-complexity CPM trellis based on the Rimoldi decomposition and thus is composed by DpM^{L-1} states. We can reduce the complexity by applying the same technique described in Section 2.4.3 to the MM-DP algorithm of Section 4.4.2, which is based on the MM decomposition and counts just D trellis states. The way to obtain the Improved-MM-DP (I-MM-DP) starting from MM-DP is very similar to that used to derive the I-DP in Section 2.4.3. In particular, firstly we just need to change the mean and variance of the Gaussian pdf in (4.22) representing phase noise probability transition $p(\theta_{n+1}|\theta_n)$, since the PN θ_n is no more modeled as a Wiener model but as an AR1 process. Secondly, in order to take into account also the second PN component $\theta_{f,n}$ (i.e., the faster component), we compute the branch metric $H_n(x_n, \psi_{n-1}, \theta_n)$ in (3.11) by averaging over such a component. Since such a component is a zero mean Gaussian random variable, with variance σ_f^2 , the

branch metric becomes:

$$H_n(x_n, \psi_{n-1}, \theta_n) = \int_{\theta_{f,n}} \exp \left\{ -\frac{\theta_{f,n}^2}{2\sigma_f^2} \right\} \cdot \exp \left\{ \frac{1}{N_0} \operatorname{Re} \left[\sum_{k=0}^{M-2} p_{k,n} a_{k,n}^*(x_n, \psi_{n-1}) e^{-j(\theta_n + \theta_{f,n})} \right] \right\} d\theta_{f,n} . \quad (4.45)$$

where the integral in (4.45) can be evaluated as a simple sum by discretizing the faster phase component $\theta_{f,n}$ as done for the slower component θ_n . Hence the I-MM-DP algorithm is obtained by just increasing the branch metric completion of the MM-DP algorithm. However, Section 2.4.4 show that improved-DP algorithm does not get significant information rate improvement with respect to the simple DP algorithm. We will verify this statement in the Section 4.5, by numerical results.

Detection Algorithm Based on Phase Noise Linear Prediction

In the following, we propose a Bayesian algorithm based on linear prediction of the phase noise process. In particular, we extend the approximate linear predictive approach described in [71], for a general time-varying phase process θ_n . We assume θ_n stationary and described by a given autocorrelation sequence of the phasor process $h_n = e^{j\theta_n}$, denoted by

$$R_h(k) = \mathbb{E} \left\{ e^{j\theta_{n+k}} e^{-j\theta_n} \right\} . \quad (4.46)$$

First of all, we focus on the vector $\mathbf{y}$ which collects the set $y_n(x_n, \sigma_n)$ of sufficient statistics of the received signal $r(t)$, for the ideal coherent detector (see Section 4.2). These statistics are generated as indicated in (2.12), i.e., by a bank of filters matched to all possible length- T CPM waveforms $\{\bar{s}(t; x_n, \omega_n)\}$. In the present Section, we propose a different set of sufficient statistics, collected in the vector $\mathbf{r}$, obtained projecting the received signal $r(t)$ over an alternative orthonormal base. In detail, the statistics are obtained by oversampling the received signal by an oversampling factor N_s which respects the anti-aliasing

condition; hence, by oversampling the signal $r(t)$ in (4.3) at $t_{n,k} = nT + kT/N_s$, we get

$$\begin{aligned} r_{n,k} &= \tilde{s}_{n,k}(x_n, \sigma_n) e^{j\theta_n} + w_{n,k}, & n &= 0, 1, \dots, N-1 \\ & & k &= 0, 1, \dots, N_s-1 \end{aligned} \quad (4.47)$$

where we have defined $r_{n,k} \triangleq r(t_{n,k})$, $\tilde{s}_{n,k}(x_n, \sigma_n) \triangleq \tilde{s}(t_{n,k}; x_n, \sigma_n)$ and $w_{n,k} \triangleq w(t_{n,k})$. In other words, n is the symbol index, while k is the oversampling index. It is trivial to demonstrate that $\{w_{n,k}\}_{n,k}$ are independent and uniformly distributed complex Gaussian random variables, with zero means and variance $\sigma_w^2 = 2N_0 N_s/T$. From that consideration, we find the following expression for the pdf $p(\mathbf{r}|\mathbf{x})$

$$p(\mathbf{r}|\mathbf{x}) = \prod_{n=0}^{N-1} \exp \left\{ -\frac{1}{\sigma_w^2} \sum_{k=0}^{N_s-1} \left| r_{n,k} - \tilde{s}_{n,k}(x_n, \sigma_n) e^{j\theta_n} \right|^2 \right\}. \quad (4.48)$$

We define

$$r'_{n,k}(x_n, \sigma_n) \triangleq \frac{r_{n,k}}{\tilde{s}_{n,k}(x_n, \sigma_n)} = r_{n,k} \tilde{s}_{n,k}^*(x_n, \sigma_n) \quad (4.49)$$

so that, replacing (4.49) in (4.47), when considering the transmitted sequence we get

$$r'_{n,k}(x_n, \sigma_n) = e^{j\theta_n} + w'_{n,k} \quad (4.50)$$

where $w'_{n,k} \triangleq w_{n,k} \tilde{s}_{n,k}^*(x_n, \sigma_n)$ is statistically equivalent to $w_{n,k}$. Hence, we can exploit the received samples (4.50) to perform $e^{j\theta_n}$ estimation, based on linear prediction filtering. We derive

$$e^{j\hat{\theta}_n} = \frac{\sum_{i=1}^C p_i r'_{n-i,0}(x_{n-i}, \sigma_{n-i})}{\left| \sum_{i=1}^C p_i r'_{n-i,0}(x_{n-i}, \sigma_{n-i}) \right|} \quad (4.51)$$

where C assumes the meaning of *prediction order* and $\{p_i\}_{i=1}^C$ are the predictor coefficients. The predictor coefficients can be computed by solving a Wiener-Hopf linear system $\mathbf{R}\mathbf{p} = \mathbf{b}$, where $\mathbf{p} \triangleq (p_1, p_2, \dots, p_C)^T$ is the unknown vector¹⁰, $\mathbf{b} = [R_h(1), R_h(2), \dots, R_h(C)]^T$ and $\mathbf{R}$ is a square $C \times C$ matrix

¹⁰The notation $(\cdot)^T$ indicates the transposition operator

whose elements are defined as the autocorrelation function of the samples $r'_{n,k}$, and has the following expression

$$\mathbf{R}(\ell, m) = \begin{cases} R_h(|\ell - m|) & \text{if } \ell \neq m \\ R_h(0) + \sigma_w^2 & \text{if } \ell = m. \end{cases} \quad (4.52)$$

Replacing (4.51) in (4.48), we derive

$$p(\mathbf{r}|\mathbf{x}) \simeq \prod_{n=0}^{N-1} \exp \left\{ -\frac{1}{\sigma_\epsilon^2} \sum_{k=0}^{N_s-1} |G_n(x_n, \dots, x_{n-C}, \sigma_n, \dots, \sigma_{n-C})|^2 \right\} \quad (4.53)$$

where σ_ϵ^2 is the mean square error prediction error, which can be expressed as [71]

$$\sigma_\epsilon^2 = R_h(0) + \sigma_w^2 - \sum_{i=0}^C p_i R_h(i). \quad (4.54)$$

and

$$G_n(x_n, \dots, x_{n-C}, \sigma_n, \dots, \sigma_{n-C}) \triangleq r_{n,k} - \tilde{s}_{n,k}(x_n, \sigma_n) \frac{\sum_{i=1}^C p_i r'_{n-i,0}(x_{n-i}, \sigma_{n-i})}{\left| \sum_{i=1}^C p_i r'_{n-i,0}(x_{n-i}, \sigma_{n-i}) \right|}. \quad (4.55)$$

By employing (1.16), we can write the $\tilde{s}_{n,k}(x_n, \sigma_n)$ definition as

$$\begin{aligned} \tilde{s}_{n,k}(x_n, \sigma_n) &= \bar{s}(t_{n,k}; x_n, \omega_n) e^{j\pi_n} \\ &= \bar{s}_{n,k}(x_n, \omega_n) e^{j\pi_n} \end{aligned} \quad (4.56)$$

where $\bar{s}(t_{n,k}; x_n, \omega_n)$ (1.17) is the component of the CPM waveform, depending on just the present symbol x_k and on the correlative state ω_n (1.4); π_n (1.5) is the phase state and we defined $\bar{s}_{n,k}(x_n, \omega_n) \triangleq \bar{s}(t_{n,k}; x_n, \omega_n)$. Thus, replacing (4.56) and (4.49) in (4.55), we find that the $G_n(x_n, \dots, x_{n-i}, \sigma_n, \dots, \sigma_{n-i})$

function becomes

$$\begin{aligned}
 G_n(x_n, \dots, x_{n-i}, \sigma_n, \dots, \sigma_{n-i}) &= \quad (4.57) \\
 &= r_{n,k} - \bar{s}_{n,k}(x_n, \sigma_n) \frac{\sum_{i=1}^C p_i r_{n-i,0} \bar{s}_{n-i,0}^*(x_{n-i}, \omega_{n-i}) e^{j(\pi_n - \pi_{n-i})}}{\left| \sum_{i=1}^C p_i r_{n-i,0} \bar{s}_{n-i,0}^*(x_{n-i}, \omega_{n-i}) \right|} \\
 &= r_{n,k} - \bar{s}_{n,k}(x_n, \omega_n) \frac{\sum_{i=1}^C p_i r_{n-i,0} \bar{s}_{n-i,0}^*(x_{n-i}, \omega_{n-i}) e^{j\pi h \sum_{m=0}^{i-1} x_{n-L-m}}}{\left| \sum_{i=1}^C p_i r_{n-i,0} \bar{s}_{n-i,0}^*(x_{n-i}, \omega_{n-i}) \right|}. \quad (4.58)
 \end{aligned}$$

Since the correlative state is composed by the $L - 1$ most recent past symbol, we understand that the function G_n just depends on the present symbol x_n and on a set of $L - 1 + C$ past symbols given by $(x_{n-1}, \dots, x_{n-(L-1+C)})$. Thus, by defining a new system state

$$\mu_n \triangleq (x_{n-1}, \dots, x_{n-(L-1+C)}) \quad (4.59)$$

we have

$$G_n(x_n, \dots, x_{n-i}, \sigma_n, \dots, \sigma_{n-i}) \equiv G_n(x_n, \mu_n), \quad (4.60)$$

and the pdf in (4.53) can be written as

$$p(\mathbf{r}|\mathbf{x}) \simeq \prod_{n=0}^{N-1} V_n(x_n, \mu_n) \quad (4.61)$$

where

$$V_n(x_n, \mu_n) \triangleq \exp \left\{ -\frac{1}{\sigma_\epsilon^2} \sum_{k=0}^{N_s-1} |G_n(x_n, \mu_n)|^2 \right\} \quad (4.62)$$

Finally, we can derive the MAP symbol detection strategy by marginalizing $p(\mathbf{x}, \boldsymbol{\mu}|\mathbf{y})$ (where vector $\boldsymbol{\mu}$ collects μ_n elements, with n from 0 to $N - 1$) by FG and SPA. In detail, by Bayes rule

$$\begin{aligned}
 p(\mathbf{x}, \boldsymbol{\mu}|\mathbf{r}) &\propto p(\mathbf{r}|\mathbf{x}, \boldsymbol{\mu}) P(\boldsymbol{\mu}|\mathbf{x}) P(\mathbf{x}) \\
 &= P(\mu_0) \prod_{n=0}^{N-1} V_n(x_n, \mu_n) I(x_n, \mu_n, \mu_{n+1}) P(x_n) \quad (4.63)
 \end{aligned}$$

where $I(x_n, \mu_n, \mu_{n+1})$ is the trellis indicator function, equal to one if x_n , μ_n , and μ_{n+1} satisfy the trellis constraints and to zero otherwise. It is clear that factorization (4.63) is equivalent to the factorization in (2.14) of the optimal coherent detector, where here the state definition μ_n replaces the CPM state σ_n (and thus, also the trellis indicator function is different), and where the branch metric $V_n(x_n, \mu_n)$ replaces $F_n(x_n, \sigma_n)$. Thus, forward recursion, backward recursion and completion stage are performed as described in (2.17), (2.18), (2.21), respectively. From (4.59), we know that the BCJR algorithm we have derived is based on a trellis of M^{L-1+C} states. So, it is clear that its computational complexity rapidly increase with the parameters L and C . In the present work, we do not address the problem of complexity reduction for the described algorithm, but we think that a possible solution is represented by reduced-search techniques (described in Section 3.6).

Predictor Coefficients for Wiener and double-AR1 PN Models

In order to compute the predictor coefficients $\mathbf{p}$ by the Wiener-Hopf linear system $\mathbf{R}\mathbf{p} = \mathbf{b}$, we need to assume a statistical model for the phase noise θ_n . From such model we can derive an expression for $R_h(k)$ defined in (4.46), necessary for the computation of matrix $\mathbf{R}$ (4.52).

If we model the phase noise as a discrete-time Wiener process of incremental variance σ_Δ^2 , it is easy to verify that

$$R_h(k) = e^{-|k|\sigma_\Delta^2/2} . \quad (4.64)$$

On the other hand, when we model the PN by the double-AR1 model described in Section 2.4.2, we can derive $R_h(k)$ as follows. First of all, we recall that the *characteristic function* of any random variable β is given by

$$\gamma_\beta(t) = \mathbb{E} \left\{ e^{jt\beta} \right\} . \quad (4.65)$$

Hence, we note from (4.46), that $R_h(k)$ is equivalent to the characteristic function of the real random variables $(\theta_{k+n} - \theta_n)$, computed for $t = 1$. The discrete-time phase noise process θ_n we are considering, is the sum of two Gaussian first

order auto-regressive processes,

$$\begin{aligned}\theta_{a,n} &= a \theta_{a,n-1} + v_{a,n} \\ \theta_{b,n} &= b \theta_{b,n-1} + v_{b,n}\end{aligned}$$

where $v_{a,n}$ and $v_{b,n}$ are two Gaussian zero-mean random variables of variance $\sigma_{v,a}^2$ and $\sigma_{v,b}^2$, respectively. $\theta_{a,n}$ and $\theta_{b,n}$ are still Gaussian random variables of variance σ_a^2 and σ_b^2 , respectively. Thus, it is easy to prove that $(\theta_{k+n} - \theta_n)$ is still a Gaussian variable, with mean zero and variance $2[R_\theta(0) - R_\theta(k)]$, where $R_\theta(k)$ is the autocorrelation sequence

$$R_\theta(k) = \mathbb{E}\{\theta_n \theta_{n+k}\} . \quad (4.66)$$

Thus, since the characteristic function of a zero-mean Gaussian random variable β with variance σ^2 is [72]

$$\gamma_\beta(t) = e^{-\sigma^2/2} \quad (4.67)$$

we derive that

$$R_h(k) = e^{[R_\theta(k) - R_\theta(0)]} . \quad (4.68)$$

Finally, since the autocorrelation function $R_\theta(k)$ is the sum of the autocorrelation functions of its AR1 components $\theta_{a,n}$ and $\theta_{b,n}$, we get

$$R_h(k) = \frac{\exp\{\sigma_a^2 a^{|k|} + \sigma_b^2 b^{|k|}\}}{\exp\{\sigma_a^2 + \sigma_b^2\}} . \quad (4.69)$$

4.5 Numerical results

4.5.1 Simulation environment

In this section, the performance of various detection schemes is assessed by means of computer simulations. We consider serially concatenated CPM (SC-CPM) schemes with iterative decoding, reporting the performance in terms of bit error rate (BER) versus E_b/N_0 , E_b being the received energy per information bit. The relevant system model is depicted in Fig. 3.9. In all considered

simulation scenarios, the information bits are encoded by means of a 4-states convolutional code with generators $(7, 5)_8$ and rate $1/2$, chosen following the guidelines in [3]. The obtained code bits are then interleaved¹¹ and mapped onto the M -ary modulation alphabet according to the natural mapping rule. At the receiver side, iterative decoding is performed: one instance of the decoder is executed after each instance of the detector, progressively refining the reliability of the decisions. The iterative process stops when the maximum number of allowed iterations, here set equal to ten, is achieved. The various simulation scenarios differ for the parameters of the concatenated CPM modulation, for the codeword length, and for the model of the phase noise affecting the channel. For all considered schemes, the bandwidth of the transmitted signal and the spectral efficiency is also reported.¹² To ensure better numerical stability, all detection/decoding algorithms are implemented in the logarithmic domain [60].

In the key of the figures and in the related discussions, the various algorithms are denoted as follows.

- FC-COH: full-complexity BCJR algorithm for ideal coherent detection (Section 2.3.1).
- MM-DP-P: per-state DP-algorithm based on the principal components of the MM decomposition (Section 4.4.2).
- MM-DP-PSS: extended per-state DP-algorithm based on the principal and some selected secondary components of the MM decomposition (Section 4.4.3).
- MM-DP-PSS-RS: extended per-state DP-algorithm based on the principal and some selected secondary components of the MM decomposition,

¹¹Pseudo-random interleavers are adopted.

¹²According to a rigorous definition, the bandwidth of a CPM signal is infinite. The reported values refer to the bandwidth which contains the 99.5% of the signal power, referred to as “99.5% bandwidth”. Hence, the spectral efficiency, that is the ratio between the actual information rate and the bandwidth, is defined with respect to the 99.5% bandwidth.

with the application of the reduced search (Section 4.4.3).

- MM-CBC: per-state CBC-algorithm based on the principal components of the MM decomposition (Section 4.4.2).
- MM-CBC-PSS: extended per-state CBC-algorithm based on the principal and some selected secondary components of the MM decomposition (Section 4.4.3).
- LP: Bayesian algorithm based on phase linear prediction (Section 4.4.4).
- GA-PLL: non-Bayesian algorithm performing phase tracking by means of a genie-aided PLL (Section 4.3.2).
- PLL: non-Bayesian algorithm performing per-state phase tracking by means of a PLL based on (4.11) (Section 4.3.2).
- W-PLL: non-Bayesian algorithm performing per-state phase tracking by means of a PLL based on (4.13) (Section 4.3.2).

All reported results related to non-Bayesian algorithms refer to monodirectional phase tracking, since it was found to ensure a better performance than bidirectional phase tracking in all investigated scenarios.

4.5.2 Channels affected by Wiener phase noise

All simulation results reported in this section are related to a codeword length of 2000 bits and a channel affected by Wiener phase noise with $\sigma_{\Delta} = 5$ degrees. As a comparison, the performance of the coherent benchmark FC-COH is also reported. The various parameters involved in the implementation of the algorithms have been optimized at a BER equal to 10^{-4} . In particular, the most convenient values of σ_{Δ} used by the MM-DP-P and the MM-CBC-P result larger than the actual value characterizing the phase noise.

First, we consider a concatenated scheme employing a minimum shift keying (MSK) modulation, that is a binary modulation with $h = 1/2$ and a rectangular frequency pulse of duration T . The 99.5% bandwidth of the transmitted

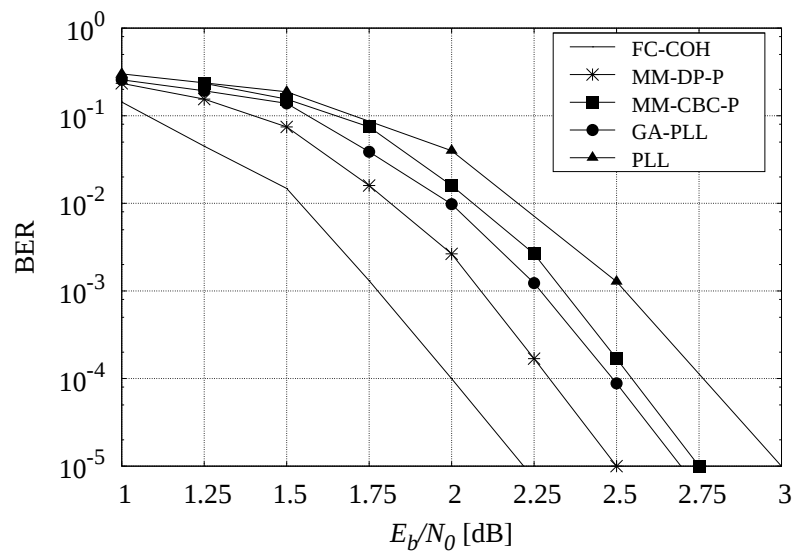


Figure 4.2: MSK modulation, codeword of 2000 bits.

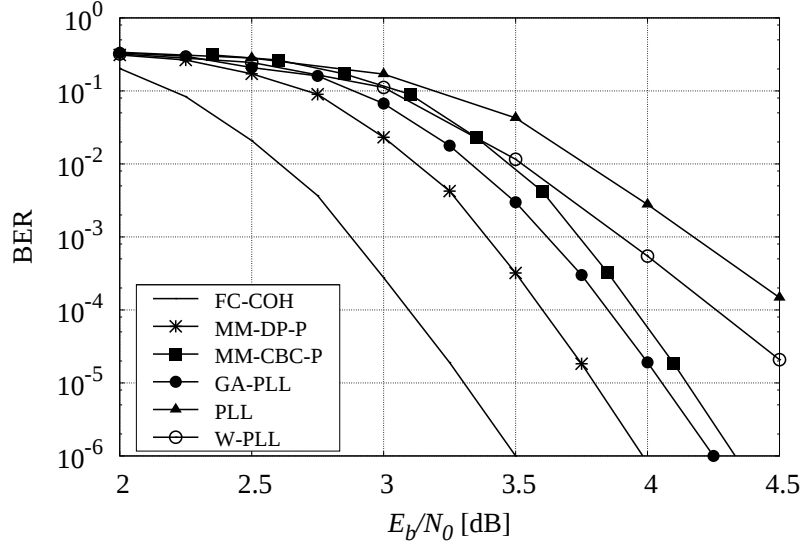


Figure 4.3: 2RC modulation with $h = 2/7$ and $M = 4$, codeword of 2000 bits.

signal is equal to $1.706/T$, and the spectral efficiency with respect to the 99.5% bandwidth is equal to 0.293 bps/Hz. Fig. 4.2 compares the performance of this coding scheme when different detection algorithms are considered, showing that the most effective one is the MM-DP-P, here implemented with $D = 12$ quantization levels, which exhibits a performance degradation of about 0.3 dB with respect to the ideal benchmark FC-COH. The other Bayesian algorithm, namely the MM-CBC-P, exhibits a further degradation of about 0.25 dB.¹³ On the other hand, Fig. 4.2 shows that the GA-PLL, which has been described in Section 4.3.2 to be just an ideal benchmark for any practical non-Bayesian algorithm, performs worse than the MM-DP-P, and only 0.06 dB better than the

¹³In the cases of some CPM formats, which include the MSK modulation, the MM-CBC-P allows a significant save in computational complexity with respect to the MM-DP-P [53], so that the MM-CBC-P can result more convenient in terms of performance/complexity tradeoff. Unfortunately, this conclusion does not hold for the CPM formats considered in the following, which were selected in [19] based on information theoretic arguments.

MM-CBC-P, so that we can state that the Bayesian approach is to be preferred. The poor performance of the PLL definitely confirms this conclusion.

We now consider a concatenated scheme employing a quaternary raised cosine (RC) modulation with $L = 2$, referred to as 2RC, and $h = 2/7$. The 99.5% bandwidth of the transmitted signal is equal to $1.561/T$, and the spectral efficiency with respect to the 99.5% bandwidth is equal to 0.641 bps/Hz. Fig. 4.3 shows that the most effective algorithm is again the MM-DP-P, here implemented with $D = 28$ quantization levels. The MM-CBC-P exhibits a degradation of about 0.3 dB, significantly better than those related to the PLL and the W-PLL. Hence, the Bayesian approach is to be preferred even in this case. Anyway, it is interesting to notice that the proposed W-PLL exhibits a gain of about 0.3 dB with respect to the PLL classically considered in the literature, thus confirming the conjectures reported in Section 4.3.2.

4.5.3 Channels affected by the SATMODE phase noise

All simulation results reported in this section are related to a channel affected by a phase noise generated according to the SATMODE mask described in Section 2.4.2. In particular, PN samples are symbol-time (ST) generated, following the double-AR1 model provided in Section 2.4.2. Among the various allowed transmission rates, the worst case from the viewpoint of the phase noise has been considered, that is the case of a transmission at 64 kbaud. The set of 4 parameters which describe the double-AR1 model at 64 kbaud, are reported in Table 2.3. As a comparison, the performance of the coherent benchmark FC-COH is also reported. The various parameters involved in the implementation of the algorithms have been optimized at a BER equal to 10^{-4} .

First, we consider a concatenated scheme employing an octal 2RC modulation with $h = 1/7$. The 99.5% bandwidth of the transmitted signal is equal to $1.622/T$, and the spectral efficiency with respect to the 99.5% bandwidth is equal to 0.925 bps/Hz. The performance of various detection algorithms is compared in Fig. 4.4, when codewords of 1760 bits are transmitted. Once again, the most effective solution is by far the MM-DP-P, here implemented

with $D = 28$ quantization levels. Unlike the previous scenarios, in this case the best non-Bayesian algorithm, namely the W-PLL, performs better than the MM-CBC-P. It is worth to notice that, at each time epoch and in each recursion, the MM-DP-P computes 28 state metrics, while the W-PLL computes 56 state metrics. Hence, besides being more effective, the MM-DP-P is also simpler.

We now consider a concatenated scheme employing a quaternary 3RC modulation with $h = 2/7$. The 99.5% bandwidth of the transmitted signal is equal to $1.227/T$, and the spectral efficiency with respect to the 99.5% bandwidth is equal to 0.815 bps/Hz. The performance of various detection algorithms is compared in Fig. 4.5 and in Fig. 4.6; the former figure refers to the transmission of codewords of 1760 bits, while the latter figure refers to the transmission of codewords of 3520 bits. These results show that the codeword length does not significantly affect the performance of the considered algorithms. It is interesting to note that, in this case, the Bayesian algorithms based on the principal components of the MM decomposition, namely the MM-DP-P and the MM-CBC-P, exhibit a poor performance. As explained in Section 4.4.3, this behavior was expected since the CPM format is such that $L > 2$. Hence, we have to resort to the extended version of the Bayesian algorithms, namely the MM-DP-PSS and the MM-CBC-PSS, which exploit also some selected secondary components—in this case, we consider eight secondary components. While the MM-CBC-PSS is definitely ineffective, the MM-DP-PSS, here implemented with $D = 28$ quantization levels, turns out to be the algorithm providing the best performance. On the other hand, the W-PLL exhibits a performance similar to that of the MM-DP-PSS, and thus the choice of the most convenient algorithm should account also for a complexity comparison. At each time epoch and in each recursion, both the MM-DP-PSS and the W-PLL compute 112 state metrics when implemented in the version which leads to the results reported in Fig. 4.5. On the other hand, as shown in Fig. 4.7, the complexity of the MM-DP-PSS can be significantly reduced at the expense of a limited performance degradation by lowering the value of D from 28 to 14

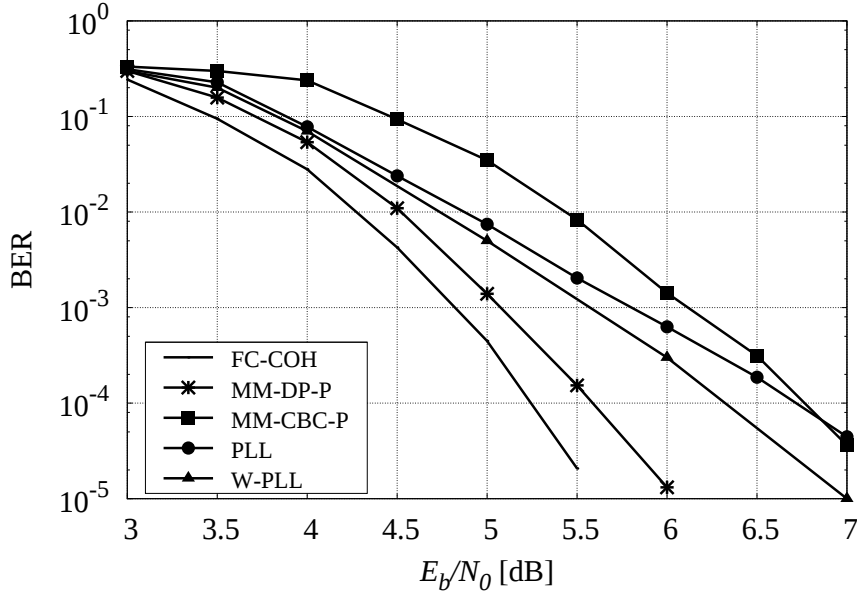


Figure 4.4: 2RC modulation with $h = 1/7$ and $M = 8$, codeword of 1760 bits.

(thus halving the complexity), or applying the MM-DP-PSS-RS (thus reducing the complexity by about four times). Since these two solutions provide nearly the same performance, the latter is clearly to be preferred. In practice, we can state that the MM-DP-PSS-RS, which performs similarly to the W-PLL but is four-times simpler, is the algorithm providing the best performance/complexity tradeoff.

In conclusion, for each considered scenario, we have selected an algorithm which ensures a limited performance degradation with respect to the ideal coherent benchmark. The selected algorithms are also fairly simple, since, in the worst case, a trellis with only 28 states per time epoch is required.

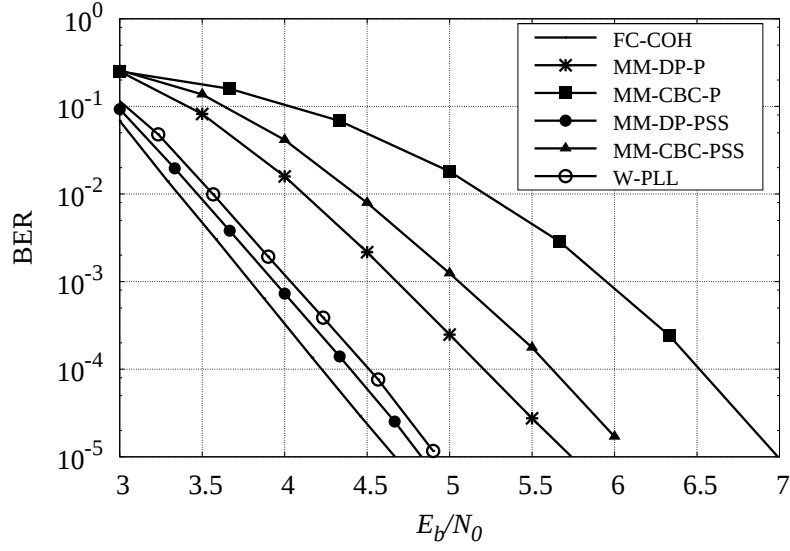


Figure 4.5: 3RC modulation with $h = 2/7$ and $M = 4$, codeword of 1760 bits.

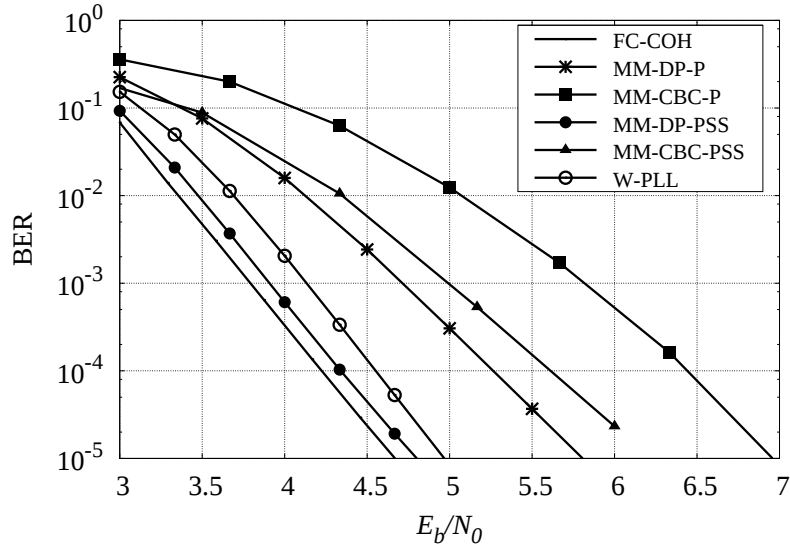


Figure 4.6: 3RC modulation with $h = 2/7$ and $M = 4$, codeword of 3520 bits.

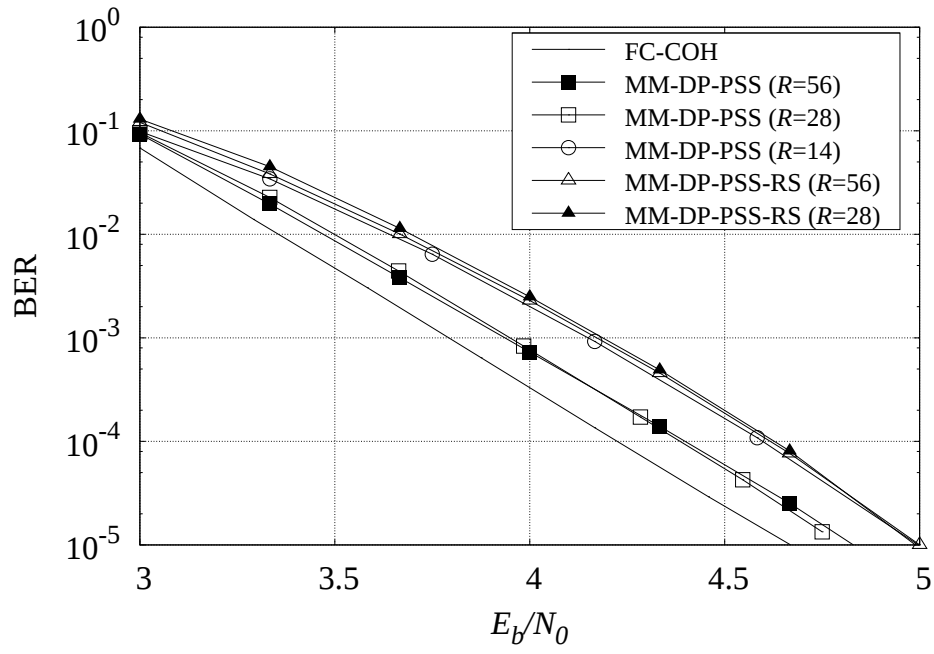


Figure 4.7: 3RC modulation with $h = 2/7$ and $M = 4$, codeword of 1760 bits.

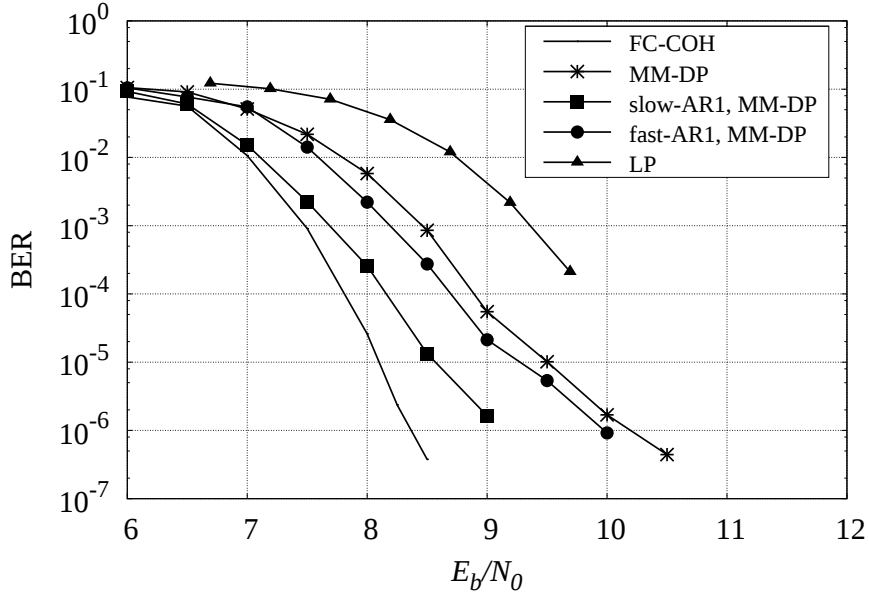


Figure 4.8: 2RC modulation with $h = 1/5$ and $M = 4$ and code rate $r = 0.88$.

4.5.4 SATMODE phase noise effects at large spectral efficiency

In the following Section, we show the BER performance for a SCCPM scheme characterized by a large spectral efficiency value. In particular, we choose a quaternary, 2-RC CPM format with $h = 1/5$, whose normalized 99.5% bandwidth is $BW = 1.005$, and a block code, the extended Bose, Ray-Chaudhuri, Hocquenghem (eBCH) provided in [73], with codeword of 1024 bits and rate $r = 0.88$. Hence, the spectral efficiency of the SCCPM scheme is $\eta = r \log_2 M = 1.76$ bps/Hz, absolutely larger than the spectral efficiency of SCCPM schemes proposed in Section 4.5.3.

In Fig. 4.8, the bit error rate of the MM-DP algorithm is represented. We employ such a detector since, from Section 4.5.3, it results to be the best one in term of BER performance, when $L = 2$. In this case however, also the performance of the MM-DP seems to be poor: we note a BER degradation of about

1.5 dB at $\text{BER} = 10^{-5}$ with respect to the FC algorithm. In Fig. 4.4, for a SC-CPM scheme with $\eta = 0.925$ bps/Hz, the same algorithm exhibits only 0.5 dB of loss with respect to the FC detector, and thus the considered algorithm gets worse performance when employed with a system of large spectral efficiency. In Section 2.4.4, it is demonstrated that, for the considered CPM format, all the algorithms based on a single phase noise discretization (i.e., algorithms like the DP in Section 2.4.3 or the MM-DP) and working on a channel affected by a double-AR1 PN, exhibit an IR loss with respect to the IR of the coherent case. Such a IR degradation is of about half dB at medium IR values, and almost 1 dB at large IR values. Hence, the large BER degradation we have noted, is only partially justified by information theoretic arguments. Probably, the BER loss is larger than what was expected because of the reduced number of quantization levels in the MM-DP we have employed in BER simulations ($D = 30$), with respect to the larger D value employed in the IR simulations. Moreover, in Fig. 2.14, we see that the IR curve constrained to the DP detector loses less than 0.5 dB at a spectral efficiency η of 0.925 bps/Hz, and approximately 0.8 dB when $\eta = 1.76$ bps/Hz; thus, the larger BER degradation when considering a more spectral efficient SCCPM scheme, is justified by IR arguments. In Fig. 4.8, we also provide the two BER curves corresponding to the application of the MM-DP algorithm when just the slow-AR1 component (or the fast-AR1 component) impairs the channel, in order to isolate the effects of the single PN processes. As expected from the IR results reported in Fig. 2.14, most of the BER degradation is due to the fast component, which is hard to be tracked.

In principle, there are two detectors by which we can recover the BER performance loss:

- the first one is to employ an algorithm based on a double PN discretization, one for each components (Section 2.4.3);
- the second one is to resort to an improved-DP algorithm (in Section 2.4.3).

The first way is not acceptable, due to the very large amounts of trellis state

necessary to describe two phase discretizations. Hence, we resort to the improved DP algorithms; in particular we can employ the I-DP algorithm (Section 2.4.3), based on the full-complexity trellis, as well as the MM-I-DP algorithm (Section 4.4.4), based on the trellis deriving from MM decomposition. However, from some BER simulations (not reported here), we found that the BER of the I-MM-DP and of the I-DP are overlapped to the BER of MM-DP in the scenario of Fig. 4.8. Thus, in this case BER results perfectly confirm the IR results of Fig. 2.14, where I-DP constrained IR is practically overlapped to the DP constrained IR.

Finally, in Fig. 2.14 we also show the performance of the LP algorithm described in Section 4.4.4, where the predictor coefficients are computed assuming a double-AR1 phase noise model and a prediction order $C = 4$. Even if such detector is matched to the channel, due to the approximations introduced in its derivation, it has a performance worse than that of the MM-DP algorithm.

Chapter 5

Multicarrier schemes over doubly-selective channels

In the following chapter we compare four different multicarrier schemes in a discrete-time, oversampled domain over doubly-selective channels. We find that all schemes can be implemented in a reduced complexity way, resorting to suitable fast transforms like DFT (Discrete Fourier Transform), DCT (Discrete Cosine Transform), or DST (Discrete Sine Transform). For all models we assume either a rectangular base pulse or a properly designed prototype pulse, well localized in both time and frequency domains, and we show that such a technique can be applied to our discrete-time block-based model. We highlight the similarities and the differences between some recently proposed multicarrier modulations based on DCT and DST to mathematical concepts such as the Wilson base, that in turn will be used to develop a novel and effective multicarrier format, ready to be employed in practical communication systems.

5.1 Introduction

Orthogonal Frequency-Division Multiplexing (OFDM) is an efficient modulation technique belonging to the wide class of multicarrier modulations, and is particularly suited for transmission over linear frequency-selective channels. It is in fact well known that, by means of OFDM, a linear time-invariant channel can be decomposed into a set of orthogonal, interference-free sub-channels [74], and exact maximum a posteriori (MAP) detection can be carried out on a symbol-by-symbol basis, thus without the need for complex equalization techniques. Other advantages of OFDM include, among others, improved spectral efficiency on frequency-selective channels by means of power allocation and easier resource allocation in multiuser scenarios. As a consequence, OFDM has been already implemented in both wireline applications (as in the digital subscriber line (DSL) standards) as well as in a wide range of wireless applications, ranging from the digital audio and video broadcasting (DAB-T, DVB-T, DVB-SH, DVB-H) standards, to the local and metropolitan area networks (WLANs and WMANs).

On the other hand, one of the OFDM most critical drawbacks, namely the increased sensitivity to the channel impulse response (CIR) time variations, has restricted the application of OFDM to scenarios characterized by sufficiently slow variations. In the presence of a rapidly time-varying CIR, where time selectivity stems for example from the Doppler effect or the oscillators' phase noise, the orthogonality between the subcarriers is destroyed, inter-carrier interference (ICI) appears [75, 76], and complex equalization techniques must be employed to cope with the latter effect [77, 78].

Recently, multicarrier modulations different from OFDM have been proposed to reduce the sensitivity to time-variations, in order to be able to effectively employ such modulations on doubly-selective channels. For example, some authors have proposed to use either the DCT or the DST as an alternative to the DFT in the modulator/demodulator implementation, for which fast implementations exist (the resulting modulation formats are denoted respec-

tively as DCT-OFDM and DST-OFDM, in order to distinguish them from the previously mentioned DFT-OFDM scheme). In particular in [79], the authors derive conditions on a guard sequence and a front-end prefilter for which a frequency selective channel can be diagonalized into orthogonal, interference-free, parallel sub-channels when DCT-OFDM is considered. Moreover, in [80] it is shown that DCT-OFDM outperforms DFT-OFDM system over AWGN channel affected by a carrier frequency offset and over a frequency-selective fast Rayleigh fading channel, while in [81] it is proved that DCT-OFDM and DST-OFDM are more robust to the interference, at the price of a dramatic decrease of the spectral efficiency due to a large overhead. It was also pointed out that such overhead can be avoided by interspersing the two above mentioned discrete trigonometric transforms (DTT) in the modulator stage (DTT-OFDM), by transmitting the DCT coefficients on the even subcarriers and the DST coefficients on the odd subcarriers [81]. However, this approach was not pursued further due to the fact that it leads to signals no longer orthogonal on linear time-invariant (LTI) channels (though orthogonality is instead ensured on non-dispersive channels). We remark that no scheme able to maintain orthogonality on a doubly-selective stochastic channel exists [82], thus orthogonality is not a realistic performance measure when doubly-selective channels are considered.

An alternative approach to combat ICI, recently pursued by several authors and denoted as pulseshape OFDM (see, for example, [83–86] and references therein), consists in a suitable design of the prototype pulse, so as to obtain a better time-frequency localization with respect to the rectangular pulse employed in standard OFDM, and thus reduce ICI. These approaches, that can be applied to any of the above mentioned multicarrier formats, allow low-complexity symbol-by-symbol detection, even on rapidly time-varying channels.

The aim of this chapter is threefold.

- Firstly, starting from a general filter-bank system, we propose an over-sampled discrete-time transmitter and receiver model aiming at the practical implementation of various multicarrier modulation formats in real-

istic communication systems.

- Secondly, we show that all the multicarrier modulation formats can be derived from such a discrete-time model, with a general prototype filter (rather than with the standard rectangular filter) and a general time and frequency spacing between the information symbols; in other word, we apply pulseshaping technique to all the schemes, so extending the DCT-OFDM, DST-OFDM and DTT-OFDM schemes already present in literature.
- Finally, inspired by the Wilson base [87, Section 4.2], that turns out to be a clever way to design well-localized orthogonal Gabor frames with high time-frequency efficiency that avoid the limiting factor due to the Balian-Low theorem [87], [88], we derive a practical multicarrier modulation with excellent performance on doubly-selective channels.

For all the above mentioned modulation schemes, we first assume a rectangular prototype pulse, as is usually done in the literature [74, 81], then we design a suitable prototype pulse, well localized in both the time and frequency domains, by means of the technique proposed in [89] or in [90].

5.2 System Model

We consider a linear modulation scheme as shown in Fig. 5.1, in which the information symbols $\{x_i\}$, generated by a source at a rate $R = (2Q + 1)/T$, are serial-to-parallel converted to obtain length- $(2Q + 1)$ blocks (also known as *OFDM blocks*), Q being an integer design parameter. We will use the following notation to denote the symbols:

$$x_n^m = x_{m(2Q+1)+n} \quad \text{with} \quad \begin{array}{l} n = -Q, -Q + 1, \dots, Q \\ m \in \mathbb{Z} \end{array}$$

where m represents the block index (also denoted as *time index*), n is the index identifying a symbol position inside a block, and T is the block period. The

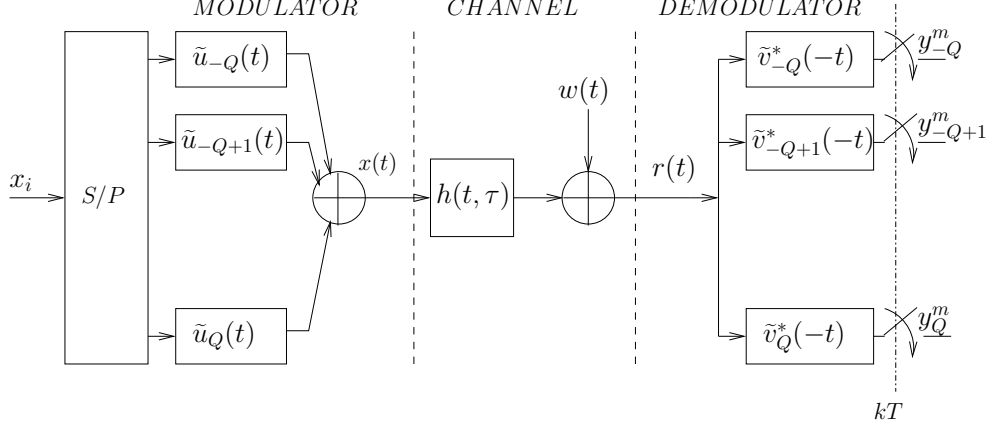


Figure 5.1: Ideal continuous-time filter-bank system.

transmitted symbols $\{x_i\}$ are zero-mean independent random variables (r.v.s) belonging to a given complex constellation. We define $\sigma_x^2 \triangleq \mathbb{E}\{|x_n|^2\}$. The transmitted signal, which is obtained by modulating a set of continuous-time filters $\{\tilde{u}_n(t)\}$, reads

$$x(t) = \sum_{m \in \mathbb{Z}} \sum_{n=-Q}^Q x_n^m \tilde{u}_n(t - mT). \quad (5.1)$$

Filters $\{\tilde{u}_n(t)\}$ will be chosen equal to a base pulse (either a rectangular pulse or an appropriately designed pulse, according to the approach described in [85]) with a suitable multiplexing in the frequency domain, as in the *uniform filter-bank* approach [91, Chapter 9]. Hence, index n is usually referred to as the *frequency* (or *subcarrier*) index. We will define the spectral efficiency η , expressed in symb/s/Hz, as the amount of code symbols that can be loaded on a time-frequency region characterized by a unitary bandwidth and timewidth. All the considered modulation formats will be designed so that they achieve the same spectral efficiency, in order to carry out a fair comparison.

After transmission over a noisy doubly-selective channel, the received signal

$r(t)$ is

$$r(t) = \int_{-\infty}^{+\infty} h(t, \tau) x(t - \tau) d\tau + w(t) \quad (5.2)$$

where the zero-mean Gaussian random process $h(t, \tau)$ denotes the time-varying channel impulse response (CIR). A wide-sense stationary uncorrelated scattering (WSSUS) channel is assumed, and $h(t, \tau) = 0$ if $\tau < 0$ or $\tau > \tau_{max}$, being τ_{max} the *maximal excess delay* of the channel. τ_{max} gives an indication about the channel spreading in the time domain. A similar parameter, denoted as ν_{max} (*maximal Doppler spread*), quantifies the channel spreading in the frequency domain [92]. $w(t)$ is a complex white Gaussian process with power spectral density $2N_0$.

At the receiver, the signal is processed by a bank of $2Q + 1$ filters. The r -th sample at the output of the k -th filter $\{\tilde{v}_k^*(-t)\}$ is

$$y_k^r \triangleq y_k(rT) = \int_{-\infty}^{+\infty} r(\tau) \tilde{v}_k^*(\tau - rT) d\tau, \quad k = -Q, \dots, Q. \quad (5.3)$$

In order to obtain a practical implementation model for the mentioned filter-bank scheme, we resort to a discrete-time model. In particular, each impulse response filter $\tilde{u}_n(t)$ (and similarly for $\tilde{v}_k(t)$) is approximately represented by

$$\tilde{u}_n(t) \simeq \sum_{i \in \mathbb{Z}} \tilde{u}_n(i) \Pi\left(\frac{t}{T_s} - i\right) \quad (5.4)$$

where $\tilde{u}_n(i) \triangleq \tilde{u}_n(iT_s)$, $T_s \triangleq T/N$ (N is the oversampling factor) and $\Pi(t)$ is a rectangular pulse defined as

$$\Pi(\alpha) \triangleq \begin{cases} 1 & \text{if } 0 \leq \alpha < 1 \\ 0 & \text{otherwise} \end{cases}. \quad (5.5)$$

The discrete-time representations of the filters, according to (5.4), is sufficiently accurate provided that a large enough oversampling factor N is employed. Under this assumption, an equivalent discrete-time representation of the transceiver is shown in Fig. 5.2, where the two square filters $\Pi(\cdot)$ play the role of digital-to-analog and analog-to-digital converters, respectively.

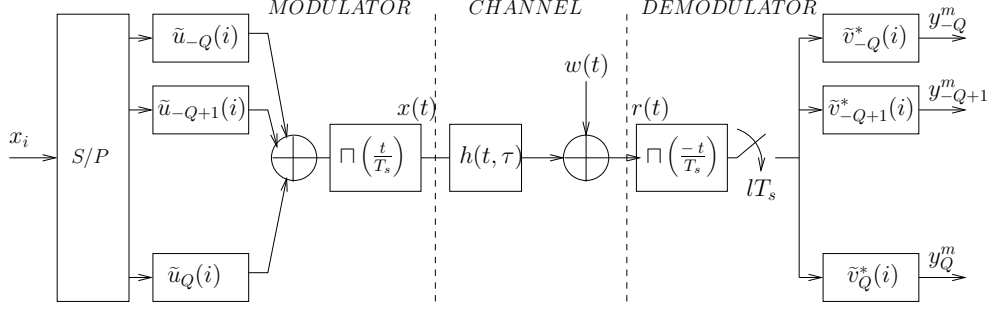


Figure 5.2: Practical oversampled discrete-time filter-bank system.

The discrete-time representation of the transmitted signal (5.1) is

$$x(t) = \sqrt{\frac{1}{T_s}} \sum_{m \in \mathbb{Z}} \sum_{n=-Q}^Q x_n^m \sum_i \tilde{u}_n(i) \Pi\left(\frac{t-mT}{T_s} - i\right) \quad (5.6)$$

where the constant $\sqrt{1/T_s}$ is an energy normalization term. The discrete-time received samples can be obtained by using (5.6) and (5.2) in (5.3), thus obtaining

$$\begin{aligned} y_k^r &= \sqrt{\frac{1}{T_s}} \sum_{l \in \mathbb{Z}} \tilde{v}_k^*(l) \int_{-\infty}^{\infty} r(t) \Pi\left(\frac{t-rT}{T_s} - l\right) dt \\ &= \sum_{m \in \mathbb{Z}} \sum_{n=-Q}^Q H_{k,n}^{r,m} x_n^m + n_k^r \quad \text{with} \quad \begin{matrix} k = -Q, \dots, Q \\ r \in \mathbb{N} \end{matrix} \end{aligned} \quad (5.7)$$

where

$$H_{k,n}^{r,m} \triangleq \sum_{l \in \mathbb{Z}} \tilde{v}_k^*(l - rN) \sum_{i \in \mathbb{Z}} \tilde{u}_n(i - mN) h_{l,l-i} \quad (5.8)$$

with

$$h_{l,i} \triangleq \frac{1}{T_s} \iint_{-\infty}^{\infty} h(t + lT_s, \tau) \Pi\left(\frac{t-\tau}{T_s} + i\right) \Pi\left(\frac{t}{T_s}\right) d\tau dt \quad (5.9)$$

and

$$n_k^r \triangleq \sqrt{\frac{1}{T_s}} \sum_{l \in \mathbb{Z}} \tilde{v}_k^*(l - rN) \int_{-\infty}^{\infty} w(t) \Pi\left(\frac{t}{T_s} - l\right) dt. \quad (5.10)$$



Figure 5.3: System model decomposed into three blocks: the modulator, the channel and the demodulator.

From (5.9), the discrete-time CIR $h_{l,i}$ has support over $0 \leq i \leq L$, where

$$L = \lfloor \tau_{\max}/T_s \rfloor + 1$$

and $\lfloor t \rfloor$ means round t down to the nearest integer.

Finally, we provide an alternative system description, aiming at the system separation into three different blocks: the transmitter, the channel, and the receiver. Replacing (5.8) in (5.7), with a few mathematical manipulations we obtain the following expression for the statistics y_k^r

$$y_k^r = \sum_{l \in \mathbb{Z}} \tilde{v}_k^*(l - rN) \left\{ \sum_{i \in \mathbb{Z}} \left[\sum_{m \in \mathbb{Z}} \sum_{n=-Q}^Q x_n^m \tilde{u}_n(i - mN) \right] h_{l,l-i} \right\} + n_k^r. \quad (5.11)$$

Hence we see that our system is composed of the following three stages, represented in Fig 5.3.

Modulator : computes the discrete-time signal $s(i)$

$$s(i) \triangleq \sum_{m \in \mathbb{Z}} \sum_{n=-Q}^Q x_n^m \tilde{u}_n(i - mN) \quad (5.12)$$

which is the sum of the output of a bank of filters whose input are the information symbols $\{x_n^m\}$. The signal $s(i)$ is then forwarded to the channel.

Channel : computes the output discrete-time signal $p(l)$ by the convolution of the input signal $s(i)$ and the channel coefficients $h_{l,i}$

$$p(l) \triangleq \sum_{i \in \mathbb{Z}} s(i) h_{l,l-i}. \quad (5.13)$$

From (5.9), the channel coefficient $h_{l,i}$ is given by the contribution of the time-varying channel impulse response and the two rectangular filters (the digital-to-analog and analog-to-digital converters).

Demodulator : derives the sufficient statistics y_k^r , by filtering the received signal $p(l)$ by the bank of filter $\{v_k(l)\}$

$$y_k^r = \sum_{l \in \mathbb{Z}} \tilde{v}_k^*(l - rN)p(l) + n_k^r. \quad (5.14)$$

Obviously the thermal noise is generated by the channel, but it is convenient for our purpose to see such noise as a Gaussian random variable n_k^r introduced by the demodulator.

The above general model is valid for all multicarrier schemes, which are based however on different modulator and demodulator stages. Hence, it will be sufficient to provide for the various schemes, the different expression for the signal $s(i)$ in (5.12) and for the statistic y_k^r in (5.14).

5.2.1 Uniform Filter-Bank

As, already anticipated in Section 5.2, all multicarrier modulation schemes we will consider, derive from the *uniform filter-banks* system (see [91], chapter 9): frequency responses of filters $\{\tilde{u}_n(i)\}$ and filters $\{\tilde{v}_n(i)\}$ are obtained by shifting the frequency response of the same real filter $u(t)$, denoted as *prototype filter*, by a spacing given by $F_n = nF$, where F is the spacing between the various subcarriers.

Since T is the OFDM symbol period and F the carrier separation, each coded symbols x_n^m can be associated to the point $(T_m, F_n) = (mT, nF)$ of a bidimensional grid in the time-frequency plane [85]. Hence, defining $\rho \triangleq FT$, the inverse of ρ can be seen as a measure of the *spectral efficiency* η (in terms of data symbols per seconds per Hertz), since higher ρ^{-1} values lead to a reduced space-frequency distance between symbols (i.e., a reduced distance between two adjacent grid points).

In particular for all employed modulation schemes, we will consider a rectangular prototype length- T pulse with $\rho = 1$ to obtain *Discrete Multitone* (DMT) schemes, and a suitable optimized pulse well localized in both time and frequency domains with $\rho \geq 1$ to achieve *pulseshape* (PS) schemes.

5.2.2 Symbol-by-Symbol Detection

Eqn. (5.7) shows that, unless $H_{k,n}^{r,m} = \delta_{n-k}\delta_{r-m}$ with probability one (being δ_ℓ the Kronecker delta), interference among the various received samples arises, both in time and frequency domains. In particular when $H_{k,n}^{r,r} \neq 0$ for $n \neq k$, ICI between various subcarriers appears, while when $H_{k,k}^{r,m} \neq 0$ for $r \neq m$, inter-block interference (IBI) is present among symbols belonging to different OFDM blocks. In other words, the sufficient statistics in (5.7) can be written as

$$y_k^r = H_{k,k}^{r,r}x_k^r + \sum_{m \neq r} \sum_{n \neq k} H_{k,n}^{r,m}x_n^m + n_k^r \quad (5.15)$$

where the term $H_{k,k}^{r,r}x_k^r$ represent the useful signal term, n_k^r is the thermal noise, while $\sum_{m \neq r} \sum_{n \neq k} H_{k,n}^{r,m}x_n^m$ is the term representing the signal interference. Hence, an optimal detection scheme would in principle involve a complex processing to cope with such an interference. Moreover, the discrete-time noise samples n_k^r can be correlated, thus further complicating the development of optimal receivers. In order to keep the computational complexity of the receiver low, filter-banks $\{\tilde{u}_n\}_n$ and $\{\tilde{v}_n\}_n$ can be designed with the aim of ensuring that $H_{k,n}^{r,m}$ is small, for example in a mean square error (MSE) sense [93], when $k \neq n$ or $r \neq m$. In this way, a low-complexity symbol-by-symbol receiver can be used without incurring in large performance penalties. In the following, a symbol-by-symbol receiver will always be assumed, although we point out that more involved equalization algorithms, for example based on the soft-interference cancellation (SIC) principle [94], could be employed. Such a receiver approximates the received statistics (5.15) as

$$y_k^r \simeq H_{k,k}^{r,r}x_k^r + n_k^r \quad (5.16)$$

and so it just computes the coefficients $H_{k,k}^{r,r}$ in order to implement a symbol-by-symbol detector. The approximation in (5.16) is an equality in perfect channel orthogonalization condition, i.e., when just the symbol x_k^r influences the output y_k^r of the k -th receiver filter at the r -th interval. As already stated, it is impossible to obtain such a condition over a doubly-selective channel [82], but all the multicarrier schemes we will derive, assure orthogonality over AWGN channels, with rectangular prototype pulse as well as with a properly optimized pulse. Since in such a case $h_{l,i} = \delta_i \forall l$, the expression for $H_{k,n}^{r,m}$ coefficients in (5.8) becomes

$$H_{k,n}^{r,m} \triangleq \sum_{l \in \mathbb{Z}} \tilde{v}_k^*(l - rN) \tilde{u}_n(l - mN) \quad (5.17)$$

which is just a scalar product between the two sequences $\tilde{v}_k(l - rN)$ and $\tilde{u}_n(l - mN)$. In other word, we say that for all the mulicarrier schemes, the two filter-banks $\{\tilde{u}_n\}_n$ and $\{\tilde{v}_n\}_n$ are biorthogonal. In our case, since we consider the receiver filters identical to the transmitter filters, we just need an orthogonal set of filters. That condition is important for two reasons. Firstly, it is important to totally suppress ISI when no dispersion is present; hence the effect of the doubly-dispersive channel can be seen as a drift away from the perfect channel orthogonalization. By increasing the channel dispersivity, the coefficients $H_{k,n}^{r,m}$, null on the AWGN channel, gradually increase and we just need to design prototype pulses which are well localized in both time and frequency domains in order to combat ICI. Secondly, if we consider an orthogonal set of received filters, from (5.10) we derive that the thermal noise samples n_k^r are white; hence they are independent for each r and k and identically distributed Gaussian random variables, with zero and variance $2N_0$.

We now show four different approaches to design transmitter and receiver filters. The last one has never been proposed for practical systems in the literature. We remark that all methods consist in a suitable multiplexing in the frequency domain of a prototype pulse: although rectangular pulses are commonly employed (e.g., the use of rectangular pulses have been proposed for DCT-OFDM [81]), their slow frequency-domain decay behaviour is responsi-

ble for their poor performance on doubly-selective channels, due to the strong interference they experience [75]. Hence, more localized pulses in the time-frequency domain have been recently proposed [83–86]. In this paper, we will consider the pulse stemming from the method proposed in [85] for all schemes, except the last one, based on Wilson base.

5.3 DFT-OFDM

We now derive the first modulation scheme based on DFT implementation. In this case, the frequency response of the filters is obtained by shifting the frequency response of a prototype filter by $F_n = nF$. In particular, denoting as $u(i)$ the oversampled version of the real prototype filter $u(t)$ at $t_i = iT_s = iT/N$, we find

$$\tilde{u}_n(i) = \tilde{v}_n(i) \triangleq u(i) e^{j2\pi F_n t_i} = u(i) e^{j2\pi \frac{ni}{N} FT}. \quad (5.18)$$

In order to derive modulation schemes for which a fast implementation exists, we just consider rational ρ values, hence $\rho = q/p$ where q and p are integers and $q \geq p$. From the above assumption, (5.18) becomes

$$\tilde{u}_n(i) = \tilde{v}_n(i) = u(i) e^{j2\pi \frac{niq}{Np}}. \quad (5.19)$$

Thus, replacing (5.19) in (5.12) and in (5.14), we find the following operation for the modulator and for the demodulator stages.

Modulator : for each OFDM symbols, the following Np -IDFT (inverse DFT) is computed

$$X_{Np}^m(i) \triangleq \frac{1}{\sqrt{Np}} \sum_{n=0}^{Np-1} x^m(n) e^{j2\pi \frac{ni}{Np}} \quad (5.20)$$

where we have defined $x^m(n)$ inserting some zeros in the sequence x_n^m :

$$x^m(n) \triangleq \begin{cases} x_{m(2Q+1)+n} & \text{if } 0 \leq n \leq Q \\ 0 & \text{if } Q+1 \leq n \leq Np-Q-1 \\ x_{m(2Q+1)+n-Np} & \text{if } Np-Q \leq n \leq Np-1. \end{cases} \quad (5.21)$$

Then the sequence $s(i)$ in (5.12) is

$$s(i) = \sqrt{Np} \sum_m X_{Np}^m[(i - mN)q] u(i - mN) \quad (5.22)$$

and it is passed to the channel.

Demodulator : substituting (5.19) in (5.14)

$$\begin{aligned} y_k^r &= \sqrt{Np} \sum_{l \in \mathbb{Z}} p(l) \frac{1}{\sqrt{Np}} e^{-j2\pi \frac{kq(l-rN)}{Np}} v(l - rN) + n_k^r \\ &= \sqrt{Np} \sum_{l \in \mathbb{Z}} v(l) p(l + rN) \frac{1}{\sqrt{Np}} e^{-j2\pi \frac{kql}{Np}} + n_k^r. \end{aligned} \quad (5.23)$$

Now, we define parameter α as

$$\alpha \triangleq \min_{\hat{\alpha}} \hat{\alpha} \quad \text{s.t.} \quad u(i) = 0 \quad \forall i \geq |\hat{\alpha}Np| \quad (5.24)$$

and (5.23) can be rewritten as

$$\begin{aligned} y_k^r &= \sqrt{Np} \sum_{l=-\alpha Np}^{\alpha Np-1} v(l) p(l + rN) \frac{1}{\sqrt{Np}} e^{-j2\pi \frac{kql}{Np}} + n_k^r \\ &= \sqrt{Np} \sum_{a=0}^{Np-1} \sum_{b=-\alpha}^{\alpha-1} v(a + bNp) p(a + bNp + rN) \frac{1}{\sqrt{Np}} e^{-j2\pi \frac{kqa}{Np}} \\ &\quad + n_k^r \\ &= \sqrt{Np} \sum_{a=0}^{Np-1} e^r(a) \frac{1}{\sqrt{Np}} e^{-j2\pi \frac{kqa}{Np}} + n_k^r \end{aligned} \quad (5.25)$$

where we defined $l \triangleq a + bNp$ and

$$e^r(a) \triangleq \sum_{b=-\alpha}^{\alpha-1} v(a + bNp) p(a + bNp + rN). \quad (5.26)$$

Hence for each OFDM symbols, the following Np -DFT is computed

$$E_{Np}^r(k) \triangleq \sum_{a=0}^{Np-1} e^r(a) \frac{1}{\sqrt{Np}} e^{-j2\pi \frac{ka}{Np}} \quad (5.27)$$

and then, when k spans from $-Q$ to Q

$$y_k^r = \sqrt{Np} E_{Np}^r ([kq]_{Np}) + n_k^r \quad (5.28)$$

where we adopt the operator $[c]_{Np}$ (which represents the module- Np of the integer c) since we exploit the periodicity of the Np -DFT transform.

In conclusion, for each transmitted OFDM block $\{x_n^m\}_n$, the modulator has to compute an Np -IDFT while the demodulator must perform an Np -DFT. From that, it is clear that when $p = 1$, as in the case of $\rho = 1$, we have a minimum in the (I)DFT dimension (which is N), while in all other cases (i.e, when ρ is not integer) the (I)DFT length is increased to Np . For all the following multicarrier schemes the same considerations hold true, where the DFT is replaced in turn by DCT or DST or both. For all employed modulation schemes, we will consider rectangular prototype length- T pulse to obtain DMT (*Discrete Multitone*) schemes, and a suitable optimized pulse well localized in both time and frequency domains [85] to achieve PS (*pulseshape*) schemes.

5.3.1 DFT-DMT

We now derive the standard OFDM scheme by choosing $\rho = 1$ and

$$u(i) = v(i) = \frac{1}{\sqrt{N}} \square_N(i) \quad (5.29)$$

where

$$\square_N(i) \triangleq \begin{cases} 1 & \text{if } 0 \leq i \leq N-1 \\ 0 & \text{elsewhere} \end{cases} \quad (5.30)$$

Replacing (5.29) and (5.19) in (5.8), from the expression of $H_{k,n}^{r,m}$ we can verify that in general, perfect channel orthogonalization is not achieved. A method to remove both the IBI and ICI can be reached for LTI channels exploiting a length- L *cyclic prefix* (we will denote such method as CP-DFT-DMT). In particular we choose:

- $h(t, \tau) \equiv h(\tau)$, from which (5.9) becomes $h_{l,i} = h_{l-i}$ with support over $[0, L]$;

- symbol period $T = N'T_s$ (instead of $T = NT_s$) where $N' = N + L$, while the frequency spacing is $F = 1/(NT_s)$; hence $\rho = FT = N'/N = 1 + L/N$;
- $\tilde{u}_n(i)$ has a longer support with respect to $\tilde{v}_k(l)$:

$$\tilde{u}_n(i) = \frac{\Pi_{N'}(i+L)}{\sqrt{N'}} e^{j2\pi nF iT_s} = \frac{\Pi_{N+L}(i+L)}{\sqrt{N+L}} e^{j2\pi \frac{ni}{N}} \quad (5.31)$$

$$\tilde{v}_k(l) = \frac{\Pi_N(i)}{\sqrt{N}} e^{j2\pi kF lT_s} = \frac{\Pi_N(i)}{\sqrt{N}} e^{j2\pi \frac{kl}{N}}. \quad (5.32)$$

Exploiting these assumptions, from (5.8) we find $H_{k,n}^{r,m} = 0$ if $m \neq r$ (i.e., IBI is suppressed); moreover coefficient $H_{k,n}^{r,r}$ can be seen as the (k,n) -th element of a circulant matrix. Since from the well-known DFT diagonalization property (see [95]) such a matrix is diagonal when pre- and post-multiplied by DFT and IDFT matrices, ICI suppression is also achieved. In conclusion, the perfect LTI channel orthogonalization achieved thanks to the CP, presents the following drawbacks:

- energy loss represented by the factor $\sqrt{N/(N+L)}$, due to the L symbols of the CP for each N symbols, added at the transmitter and discarded at the receiver;
- spectral efficiency loss represented by the increase of ρ from 1 to $1 + L/N$.

Such a diagonalization cannot be achieved over doubly-selective channel.

5.3.2 DFT-PS

Although DFT-DMT with CP is a very efficient scheme for LTI channels, its performance deeply degrades in presence of a time-varying channel due to the IBI and ICI presence. An approach adopted to cope with this problem and denoted as *pulseshape OFDM* is based on the design of a prototype pulse, well localized in time and frequency in order to reduced the interference. We can reduce interference by increasing the distance between the symbols x_n^n in the

190 Chapter 5. Multicarrier schemes over doubly-selective channels

time-frequency grid (i.e., by increasing T and F), but it is not acceptable since it also reduces the spectral efficiency ρ^{-1} of the system. Moreover, we would like to design a pulse such that the two banks of filters at the transmitter and at the receiver are biorthogonal, as already explained in Section 5.2. Since looking at the modulator/demodulator filter expression (5.18), we find that the transmitted signal (5.20) belongs to a Gabor set (verifying the well-known connection between OFDM systems and Gabor frames, see [83]), we can resort to Gabor theory. Hence, from the Balian-Low theorem [87] we know that it is not possible to simultaneously satisfy the three following conditions:

- orthogonal filters (to ensure white noise and exact reconstruction when the identity channel $h(t, \tau) \equiv \delta(\tau)$ is considered);
- pulse $u(i)$ well localized in time and frequency domains;
- maximum spectral efficiency $\rho^{-1} = 1$.

While in the DMT-based schemes, we have maximum spectral efficiency (i.e. $\rho^{-1} = 1$) but a bad localized pulse (the rectangular pulse has an infinite bandwidth with a slow decay), in pulshape technique we want to satisfy the first two conditions, and we slightly increase ρ so losing spectral efficiency in order to mitigate symbol interference.

In the following we take into account a real and symmetric pulse $u(t) = v(t)$ obtained through the pulshape technique described in [85].

5.4 DCT-OFDM

In the modulation scheme, if we employ DCT as an alternative to DFT [81], transmitter and receiver filters are obtained by modulating the prototype filter $u(t)$ by cosine functions. Sampling at $t_i = (i + 1/2)T_s$, we find

$$\tilde{u}_n(i) = \tilde{v}_n(i) = u(i)\sqrt{\beta_n} \cos[\pi F_n t_i] = u(i)\sqrt{Np} C_{Np}(nq, i) \quad (5.33)$$

where we recall that $FT = q/p$ and we denote by $C_N(n, i)$ the basis function of DCT-II [96]:

$$C_N(n, i) \triangleq \sqrt{\frac{\beta_n}{N}} \cos \left[\pi \frac{n(i + 1/2)}{N} \right] \quad (5.34)$$

and β_n is 1 if $n = 0$ and 2 if $n = 1, \dots, N - 1$. Moreover in all the following schemes we choose the index n , denoting the subcarriers, as varying from 0 to $2Q$ (instead that from $-Q$ to Q). We resort to the DCT since when IBI and ICI are present, DCT-based scheme can probably reach better performance with respect to the DFT-based scheme, due to the energy-compaction property of DCT transform [97] (the interference power is concentrated around a few low-index DCT coefficients).

Thus, replacing (5.33) in (5.12) and in (5.14), we find the following operation for the modulator and for the demodulator stages.

Modulator : for each OFDM symbols, the following Np -IDCT is computed

$$X_{c,Np}^m(i) \triangleq \sum_{n=0}^{Np-1} x_c^m(n) C_{Np}(nq, i) \quad (5.35)$$

where we have defined the length- Np sequence $x_c^m(n)$ by zero padding x_n^m in the following way:

$$x_c^m(n) \triangleq \begin{cases} x_{m(2Q+1)+i} & \text{if } n = iq \cap 0 \leq i \leq 2Q \\ 0 & \text{elsewhere} \end{cases} \quad (5.36)$$

Then the sequence $s(i)$ in (5.12) is

$$s(i) = \sqrt{Np} \sum_m X_{c,Np}^m(i - mN) u(i - mN) \quad (5.37)$$

and goes to the channel.

Demodulator : substituting (5.33) in (5.14), we find

$$y_k^r = \sqrt{Np} \sum_{a=0}^{Np-1} \sum_{b=-\alpha}^{\alpha-1} v(a+bNp) p(a+bNp+rN) C_{Np}(kq, a+bNp) + n_k^r. \quad (5.38)$$

Since it is trivial to show that $C_{Np}(k, a + bNp) = C_{Np}(k, a)$ when the subcarrier index k is even, we assume that only the even subcarriers transmit information symbols, while all odd subcarriers are suppressed. Thanks to this assumption, (5.38) becomes

$$y_k^r = \sqrt{Np} \sum_{a=0}^{Np-1} e^r(a) C_{Np}(kq, a) + n_k^r. \quad (5.39)$$

where $e^r(a)$ is defined as in (5.26). Hence, for each OFDM symbols the following Np -DCT is computed

$$E_{c,Np}^r(k) \triangleq \sum_{a=0}^{Np-1} e^r(a) C_{Np}(k, a) \quad (5.40)$$

and then, when k spans from 0 to $2Q$

$$y_k^r = \sqrt{Np} E_{c,Np}^r(kq) + n_k^r. \quad (5.41)$$

As in the DFT case, for each OFDM block the transmitter and the receiver carry out an Np -IDCT and an Np -DCT, respectively. However, as assumed in the derivation of (5.39), in order to employ the DCT-OFDM multicarrier scheme we need to suppress all odd subcarriers, thus increasing the frequency spacing among the information symbols (in particular the resulting ρ parameters is doubled and hence the spectral efficiency η is halved). Both the rectangular and the optimized pulse $u(i)$ can be employed and, as in the case of DFT-DMT, DCT-DMT can also reach perfect channel orthogonalization over LTI channels.

5.4.1 DCT-MAN

Choosing a rectangular prototype pulse as in (5.29) and $\rho = 1$, we derive a DCT-DMT scheme. By replacing the filter expression in (5.8), we can verify that perfect channel orthogonalization is not achieved as is for DFT-DMT described in 5.3.1. A method to completely remove interference in the case of

LTI channels is proposed by Mandyam in [81], where such a condition is obtained through the use of symmetric extension and cyclic prefix. Since it can be easily proved that N -IDCT with all odd inputs equal to zero is equivalent to a $N/2$ -IDCT whose output sequence $o(n)$ is divided by $\sqrt{2}$ and then symmetrically extended (i.e., $o(n) = o(N - 1 - n)$ when $n = N/2, \dots, N - 1$), the symmetric extension proposed by Mandyam is exactly equivalent to the odd subcarriers suppression we proposed in Section 5.4. In conclusion, channel orthogonalization is pursued through a reduction in the information throughput of a factor $1/2$ with respect to the DFT-DMT scheme and $\rho = 2(1 + L/N)$. We will denote the scheme exploiting rectangular pulse and symmetric extension as DCT-MAN.

5.4.2 DCT-PS

As in the DFT-OFDM case, we can face the strong interference affecting the DMT-based scheme (i.e., the DCT-MAN scheme) when employed on a doubly-selective channel, by resorting to a suited pulse well localized in both time and frequency domains. In particular, it can be shown that the pulse obtained in [89] ensures orthogonal DCT-based filter-banks, provided that just even subcarriers are employed. Hence, we will consider PS-DCT schemes with optimized pulse and spectral efficiency halved with respect to the DFT-PS case.

5.5 DTT-OFDM

In [81], Mandyam also proposed a scheme based on both DCT and DST, with a rectangular prototype pulse. In detail, the prototype filter $u(t)$ is modulated by a cosine function for the even subcarriers and by a sine function for the odd subcarriers. By sampling $\tilde{u}_n(t)$ and $\tilde{v}_n(t)$ at $t_i = (i + 1/2)T_s$, we obtain

$$\tilde{u}_n(i) = \begin{cases} u(i)\sqrt{Np} C_{Np}(nq, i) & \text{if } n \text{ even} \\ u(i)\sqrt{Np} S_{Np}(nq - 1, i) & \text{if } n \text{ odd} \end{cases} \quad (5.42)$$

194 Chapter 5. Multicarrier schemes over doubly-selective channels

and the same is for $\tilde{v}_k(i)$, where we have defined the basis function of DST-II [96]:

$$S_N(n, i) \triangleq \sqrt{\frac{\gamma_n}{N}} \sin \left[\pi \frac{(n+1)(i+1/2)}{N} \right] \quad (5.43)$$

and γ_n is 1 if $n = N - 1$ and 2 if $n = 0, \dots, N - 2$. Although in [81] this scheme is not further analyzed since it cannot lead to a complete interference cancellation over LTI channels, this is not a significant drawback over doubly-selective channels [83]. Thus, replacing (5.42) in (5.12) and in (5.14), we find the following operation for the modulator and for the demodulator stages.

Modulator : for each OFDM symbols, the following Np -IDCT and Np -IDST are computed

$$X_{c,Np}^m(i) \triangleq \sum_{n=0}^{Np-1} x_c^m(n) C_{Np}(nq, i) \quad (5.44)$$

$$X_{s,Np}^m(i) \triangleq \sum_{n=0}^{Np-1} x_s^m(n) S_{Np}(nq, i) \quad (5.45)$$

where we have defined the length- Np sequences $x_c^m(n)$ and $x_s^m(n)$ as

$$x_c^m(n) \triangleq \begin{cases} x_{m(2Q+1)+i} & \text{if } n = iq \cap i \text{ even} \cap 0 \leq i \leq 2Q \\ 0 & \text{elsewhere} \end{cases} \quad (5.46)$$

$$x_s^m(n) \triangleq \begin{cases} x_{m(2Q+1)+i} & \text{if } n = iq - 1 \cap i \text{ odd} \cap 0 \leq i \leq 2Q \\ 0 & \text{elsewhere} \end{cases} \quad (5.47)$$

Then the sequence $s(i)$ in (5.12) is

$$s(i) = \sqrt{Np} \sum_m [X_{c,Np}^m(i - mN) + X_{s,Np}^m(i - mN)] u(i - mN) . \quad (5.48)$$

Demodulator : substituting (5.42) in (5.14), we find

$$y_k^r = \begin{cases} \sqrt{Np} \sum_{a=0}^{Np-1} e^r(a) C_{Np}(kq, a) + n_k^r & \text{if } k \text{ even} \\ \sqrt{Np} \sum_{a=0}^{Np-1} e^r(a) S_{Np}(kq - 1, a) + n_k^r & \text{if } k \text{ odd} \end{cases} \quad (5.49)$$

where we have exploited the periodicity

$$C_{Np}(k, a + b Np) = C_{Np}(k, a) \quad \text{if } k \text{ even} \quad (5.50)$$

$$S_{Np}(k, a + b Np) = S_{Np}(k, a) \quad \text{if } k \text{ odd} \quad (5.51)$$

and where $e^r(a)$ is defined as in (5.26). Hence for each OFDM symbol, the demodulator computes the Np -DCT in (5.40) and the following Np -DST

$$E_{s,Np}^r(k) \triangleq \sum_{a=0}^{Np-1} e^r(a) S_{Np}(k, a) \quad (5.52)$$

and then, when k spans from 0 to $2Q$

$$y_k^r = \begin{cases} \sqrt{Np} E_{c,Np}^r(kq) + n_k^r & \text{if } k \text{ even} \\ \sqrt{Np} E_{s,Np}^r(kq - 1) + n_k^r & \text{if } k \text{ odd} . \end{cases} \quad (5.53)$$

Also, for this modulation scheme we can derive a DTT-DMT case by considering a rectangular prototype pulse and $\rho = 1$ (but perfect channel orthogonalization cannot still be achieved over LTI channel) and a more general scheme with $\rho > 1$ and a suitable optimized prototype pulse (DTT-PS scheme). In particular, for DTT-PS we can employ the well-localized pulse obtained in [85] and already employed for DFT-OFDM and DCT-OFDM, since it ensures orthogonality over AWGN channels.

5.6 ODTT-OFDM

In 1987 Wilson, in order to achieve good time-frequency localization, proposed the construction of a set of functions, based on sine and cosine functions, where the various subcarriers are characterized by functions with a suitable time offset (see [87], Sect. 4.2 and references therein). In particular, a continuous-time Wilson orthogonal basis was constructed by Daubechies, Jaffard and Journé in 1991 [90] using combination of elements of Gabor tight frame with redundancy 2. Subsequently, discrete-time Wilson function sets and frames are introduced

and discussed in [98, 99] and it is shown that a Wilson base can be derived from Weyl-Heisenberg sets. The construction of a Wilson basis for general time-frequency lattices whose generator matrix is in Hermite normal form is discussed in [100]. In [101], these results are extended on the general lattice of volume 1/2. Finally, in [102], the Wilson base construction is extended to Wilson sets and frames with arbitrary oversampling and redundancy.

In this Section, we derive discrete-time expressions for the impulse responses of a bank of filters, starting from the Wilson functions set, whose expression is provided in [87]; then, we derive an oversampled discrete-time multicarrier scheme based on such a basis as an instance of our practical oversampled discrete-time model. In such a way, we generalize the Wilson base (obtained for $\rho = 1$) to all ρ values greater than one and we address the problem of finding a well-localized pulse ensuring orthogonality over AWGN for such a set of function. We will denote such a scheme as ODTT-OFDM (i.e., *offset* DTT-OFDM), since it is very similar to the DTT schemes, but impulse response of some filters are shifted of a time offset equal to $T/2$. Finally, it is interesting to note that Wilson base construction, as well as the DCT-OFDM and DTT-OFDM schemes, is not a Gabor set and hence it allows to avoid the Balian-Low theorem: it can be an orthonormal set of functions with $FT = 1$ and employing simultaneously a well localized pulse.

5.6.1 Wilson Base Derivation

The Gabor function set is a windowed Fourier frame expressed as [87]

$$p_{n,m}(t) = u(t - mT)e^{j2\pi nFt}$$

where $u(t)$ is the window pulse, T is the time spacing, and F the frequency spacing. Filter-banks employed in DFT-OFDM scheme are based on such a function set (where $u(t)$ is the impulse response of the prototype pulse). In 1987, Wilson proposed to construct an orthonormal base with good time-frequency localization, not generated by a strict time-frequency lattice in order to avoid the Balian-Low theorem. Hence, Wilson suggested an orthonormal base $g_{n,m}(t)$

of the type

$$g_{n,m}(t) = z_n(t - mT), \quad n \in \mathbb{N}, m \in \mathbb{Z} \quad (5.54)$$

where the continuous time Fourier transform $Z_n(f)$ of the pulse $z_n(t)$ is a function with two peaks, situated near $n/2 F$ and $-n/2 F$.

A construction is suggested in [90] for which $n \in \mathbb{N} \setminus \{0\}$, $m \in \mathbb{Z}$ and

$$\begin{cases} Z_1(f) = U(f) \\ Z_n(f) = \frac{1}{\sqrt{2}} [U(f - \ell F) + (-1)^{\ell+k} U(f + \ell F)] e^{j\pi k f T} \end{cases} \quad \forall n > 1 \quad (5.55)$$

with $n = 2\ell + k$, $\ell \in \mathbb{N}$, $k = 0$ or 1 , and $\ell = 0$, $k = 0$ excluded. $U(f)$ is the Fourier transform of a pulse $u(t)$, which will play the role of the prototype pulse in the following derivation. We find the expression of $z_n(t)$ by antitransforming $Z_n(f)$ in (5.55) and replacing it in (5.54). By some mathematical manipulation, we find the following continuous-time expression for a Wilson function set:

$$g_{n,m}(t) = \begin{cases} u(t - mT) & \text{if } n = 1 \\ \sqrt{2} u(t - mT) \cos[\pi n F(t - mT)] & \text{if } \begin{cases} [n]_4 = 0 \\ n > 1 \end{cases} \\ \sqrt{2} u(t - mT + T/2) \sin[\pi(n-1)F(t - mT)] & \text{if } \begin{cases} [n]_4 = 1 \\ n > 1 \end{cases} \\ \sqrt{2} u(t - mT) \sin[\pi n F(t - mT)] & \text{if } [n]_4 = 2 \\ \sqrt{2} u(t - mT + T/2) \cos[\pi(n-1)F(t - mT)] & \text{if } [n]_4 = 3. \end{cases} \quad (5.56)$$

Hence, the proposed construction shows that the key to obtain a good time-frequency localization and orthonormality in the window Fourier framework is to use sines and cosines rather than complex exponentials.

5.6.2 Modulator and Demodulator Derivation

We now derive a multicarrier scheme belonging to the filter-bank system model described in Section 5.2 and based on the Wilson base. In particular, by sampling the continuous-time base $g_{n,m}(t)$ (5.56) at $t_i = (i + 1/2)T_s$ and by considering $\rho = q/p$, $q, p \in \mathbb{Z}$, we obtain the following discrete-time expression for

the impulse response of the filters

$$\tilde{u}_n(i) = \tilde{v}_n(i) = \begin{cases} u(i) & \text{if } n = 1 \\ \sqrt{Np} u(i) C_{Np}(nq, i) & \text{if } [n]_4 = 0 \cap n > 1 \\ \sqrt{Np} u(i + N/2) S_{Np}(nq - q - 1, i) & \text{if } [n]_4 = 1 \cap n > 1 \\ \sqrt{Np} u(i) S_{Np}(nq - 1, i) & \text{if } [n]_4 = 2 \\ \sqrt{Np} u(i + N/2) C_{Np}(nq - q, i) & \text{if } [n]_4 = 3. \end{cases} \quad (5.57)$$

Moreover, for each OFDM block we consider the frequency index n spanning from 1 to $2Q + 1$, instead that from 0 to $2Q$. Thus, replacing (5.57) in (5.12) and in (5.14), we find the following operation for the modulator and for the demodulator stages.

Modulator : for each OFDM symbol, it computes the Np -IDCT for the sequences $x_0^m(n)$ and $x_3^m(n)$ (denoted as $X_{0,Np}^m(i)$ and $X_{3,Np}^m(i)$ respectively) and the two the Np -IDST for the sequences $x_1^m(n)$ and $x_2^m(n)$ (denoted as $X_{1,Np}^m(i)$ and $X_{2,Np}^m(i)$ respectively), where

$$x_0^m(n) \triangleq \begin{cases} x_{m(2Q+1)+1} & \text{if } n = 0 \\ x_{m(2Q+1)+i} & \text{if } \begin{cases} n = iq \cap [i]_4 = 0 \\ 1 \leq i \leq 2Q + 1 \end{cases} \\ 0 & \text{elsewhere} \end{cases} \quad (5.58)$$

$$x_1^m(n) \triangleq \begin{cases} x_{m(2Q+1)+i} & \text{if } \begin{cases} n = iq - q - 1 \cap [i]_4 = 1 \\ 1 \leq i \leq 2Q + 1 \end{cases} \\ 0 & \text{elsewhere} \end{cases} \quad (5.59)$$

$$x_2^m(n) \triangleq \begin{cases} x_{m(2Q+1)+i} & \text{if } \begin{cases} n = iq - 1 \cap [i]_4 = 2 \\ 1 \leq i \leq 2Q + 1 \end{cases} \\ 0 & \text{elsewhere} \end{cases} \quad (5.60)$$

$$x_3^m(n) \triangleq \begin{cases} x_{m(2Q+1)+i} & \text{if } \begin{cases} n = iq - q \cap [i]_4 = 3 \\ 1 \leq i \leq 2Q + 1 \end{cases} \\ 0 & \text{elsewhere} \end{cases} \quad (5.61)$$

Then the sequence $s(i)$ in (5.12) is

$$s(i) = \sqrt{Np} \left\{ \sum_m [X_{0,Np}^m(i - mN) + X_{2,Np}^m(i - mN)] u(i - mN) \right. \\ \left. + \sum_m [X_{1,Np}^m(i - mN) + X_{3,Np}^m(i - mN)] u(i - mN + N/2) \right\}. \quad (5.62)$$

Demodulator : substituting (5.57) in (5.14), we find

- if $k = 1$

$$y_k^r = \sqrt{Np} E_{c,Np}^r(0) + n_k^r$$

- if $[k]_4 = 0 \cap k > 1$

$$y_k^r = \sqrt{Np} E_{c,Np}^r(kq) + n_k^r$$

- if $[k]_4 = 1 \cap k > 1$

$$y_k^r = \sqrt{Np} F_{s,Np}^r(kq - q - 1) \cos \left[\frac{\pi}{2} \rho(k - 1) \right] \\ - \sqrt{Np} F_{c,Np}^r(kq - q) \sin \left[\frac{\pi}{2} \rho(k - 1) \right] \epsilon_k + n_k^r$$

- if $[k]_4 = 2$

$$y_k^r = \sqrt{Np} E_{s,Np}^r(kq - 1) + n_k^r$$

- if $[k]_4 = 3$

$$y_k^r = \sqrt{Np} F_{c,Np}^r(kq - q) \cos \left[\frac{\pi}{2} \rho(k - 1) \right] \\ - \sqrt{Np} F_{s,Np}^r(kq - q - 1) \sin \left[\frac{\pi}{2} \rho(k - 1) \right] / \epsilon_k + n_k^r$$

where the frequency index k spans from 1 to $2Q+1$, $\epsilon_k \triangleq \sqrt{\gamma_{kq-q-1}/\beta_{kq-q}}$, $E_{c,Np}^r$ and $E_{s,Np}^r$ are defined in (5.40) and in (5.52) respectively and $F_{c,Np}^r$ and $F_{s,Np}^r$ are Np -DCT and Np -DST of the sequence

$$F^r(a) \triangleq \sum_{b=-\alpha}^{\alpha-1} v(a + bNp) p(a + bNp + rN - N/2). \quad (5.63)$$

Hence for each OFDM symbol, the demodulator computes two Np -DCT and two Np -DST.

5.6.3 ODTT-PS

Also for this modulation scheme we can derive an ODTT-DMT case by considering rectangular prototype pulse and $\rho = 1$ (but perfect channel orthogonalization cannot be achieved over LTI channel). We can also consider an improved scheme, ODTT-PS, based on a well localized prototype pulse. However in this case we believe that the pulse obtained in [85] and employed in all previous multicarrier schemes, does not ensure orthogonal ODTT filters. In particular, we cannot mathematically demonstrate orthogonality and numerical results confirm our assumption. Hence, we must cope with the problem of finding a well localized pulse, possibly with an exponential decay in both time and frequency domains, which also guarantees orthogonality over AWGN channel. In [90], the problem is solved for the special case of $\rho = 1$, which is the condition ensuring that the Wilson function set is an orthonormal base. In principle, since Wilson base avoid the Balian-Low theorem, we can employ ODTT-PS scheme with well localized prototype pulse and simultaneously $\rho = FT = 1$. However, that situation is not practicable since the distance between adjacent symbols in the time-frequency plane is so low that the system is affected by a strong performance degradation on a doubly-dispersive channel. Hence, we need to generalize the procedure used in [90] to find the expression of a well localized impulse ensuring orthogonality.

Unluckily such a problem is still an open issue. Actually we can just derive the conditions that the pulse must satisfy in order to provide orthogonality. Given a functions set $\{g_{n,m}(t)\}$ with $n \in \mathbb{N} \setminus \{0\}$, $m \in \mathbb{Z}$, orthogonality condition can be expressed as

$$\sum_{n=1}^{\infty} \sum_{m \in \mathbb{Z}} \langle p(t), g_{n,m}(t) \rangle \langle g_{n,m}(t), h(t) \rangle = \langle p(t), h(t) \rangle \quad (5.64)$$

where we have defined the scalar product between two square summable signals

$p(t)$ and $h(t)$ as

$$\langle p(t), h(t) \rangle \triangleq \int_{-\infty}^{\infty} p^*(t) h(t) dt . \quad (5.65)$$

Hence, by replacing the Wilson base expression (5.54) in (5.64), by some mathematical manipulations we find

$$\frac{1}{T} \sum_{n=1}^{\infty} Z_n(f) Z_n^* \left(f - \frac{k}{T} \right) = \delta_k, \quad \forall k \in \mathbb{Z} . \quad (5.66)$$

Finally, we replace the expression of $Z_n(f)$ given by (5.55) in (5.66) and we just consider rational ρ values; we get

$$\begin{cases} \frac{1}{T} \sum_{\ell \in \mathbb{Z}} U(f - \ell F) U(f - \frac{2i}{T} - \ell F) = \delta_i \\ \frac{1}{T} \sum_{\ell \in \mathbb{Z}} (-1)^\ell U(f - \ell F) U(f - \frac{2i}{T} - \frac{1}{T} + \ell F) = 0 \end{cases} \quad \forall i \in \mathbb{Z} . \quad (5.67)$$

In conclusion, we have found two orthogonal conditions (5.67) which are the same provided in [98]. So the open issue is the following: we want to find a prototype pulse with impulse response $u(t)$ as close as possible to a Gaussian pulse (which is the best localized function, with Gaussian decay in both the time and frequency domains) and whose Fourier transform $U(f)$ is subject to the two constraints in (5.67). Such a problem is solved in [90] for the case of $\rho = FT = 1$ but not yet for the general case of rational FT values.

5.7 Signal Spectrum and Bandwidth Computation

In order to have a bandwidth definition for the transmitted signal, we compute a closed-form expression for the spectrum of signal $x(t)$ (5.6) for each proposed multicarrier scheme. In detail, it can be easily shown that the autocorrelation function $R_x(\tau, t)$, defined as

$$R_x(\tau, t) = \mathbb{E}_x \{x(t) x^*(t + \tau)\} \quad (5.68)$$

is cycle-stationary with period T . Hence we can make it stationary by adding a random variable T_0 , uniformly distributed in $[0, T)$. Thus, we can define $R_x(\tau)$ as

$$R_x(\tau) = \mathbb{E}_{x, T_0} \{x(t - T_0) x^*(t - T_0 + \tau)\} \quad (5.69)$$

202 Chapter 5. Multicarrier schemes over doubly-selective channels

and by replacing (5.6) in (5.69), with some mathematical manipulation we find

$$R_x(\tau) = \frac{\sigma_x^2}{T} \sum_{i \in \mathbb{Z}} \sum_{\ell \in \mathbb{Z}} \Lambda\left(\frac{\tau}{T_s} + i - \ell\right) \sum_n \tilde{u}_n(i) \tilde{u}_n^*(\ell) \quad (5.70)$$

where we have defined

$$\Lambda(t) \triangleq \Pi(t) \otimes \Pi(-t) = \begin{cases} 1 - t & \text{if } 0 \leq t \leq 1 \\ 1 + t & \text{if } -1 \leq t \leq 0 \\ 0 & \text{elsewhere} \end{cases} \quad (5.71)$$

In (5.70), n spans from $-Q$ to Q for the DFT scheme, from 0 to $2Q$ for the DCT and DTT schemes and from 1 to $2Q + 1$ for the ODTT scheme. Thus, we find the power spectral density (PSD) of the stationary process $x(t)$ by applying the Fourier transform to $R_x(\tau)$ (5.70):

$$S_x(fT) = \frac{\sigma_x^2}{N} \text{sinc}^2\left(\frac{fT}{N}\right) \sum_n \left| \sum_{i \in \mathbb{Z}} \tilde{u}_n(i) e^{-j2\pi \frac{i}{N} fT} \right|^2. \quad (5.72)$$

We define $\tilde{U}_n^s(f)$ the sequence Fourier Transform of the prototype pulse $\tilde{u}_n(i)$

$$\tilde{U}_n^s(f) \triangleq \sum_{i \in \mathbb{Z}} \tilde{u}_n(i) e^{-j2\pi i \frac{T}{N} f} \quad (5.73)$$

and by substituting (5.73) in (5.72) we obtain

$$S_x(fT) = \frac{\sigma_x^2}{N} \text{sinc}^2\left(\frac{fT}{N}\right) \sum_n \left| \tilde{U}_n^s(f) \right|^2. \quad (5.74)$$

DFT-OFDM Spectrum

By replacing in (5.73) the expression (5.19) for the DFT-OFDM scheme, we get

$$\tilde{U}_n^s(f) = U^s(f - nF) \quad (5.75)$$

where we define $U^s(f)$ as the Fourier transform of the discrete-time prototype pulse $u(i)$. Thus the expression of the power spectral density of the DFT-OFDM signal $x(t)$ is

$$S_x(f) = \frac{\sigma_x^2}{N} \text{sinc}^2\left(\frac{fT}{N}\right) \sum_{n=-Q}^Q |U^s(f - nF)|^2. \quad (5.76)$$

From (5.76) we observe that the signal PSD is composed by the sum of $2Q + 1$ repetitions of the Fourier transform of the sequence $u(i)$, one for each subcarrier. In detail, the pulse $U^s(f)$ associated to the n -th subcarrier is centered around the frequency value $f = nF$, and hence we can verify that the distance among adjacent subcarriers is actually equivalent to the frequency separation F . We know that $U^s(f)$ is related to the Fourier transform $U(f)$ of the continuous-time pulse $u(t)$ by

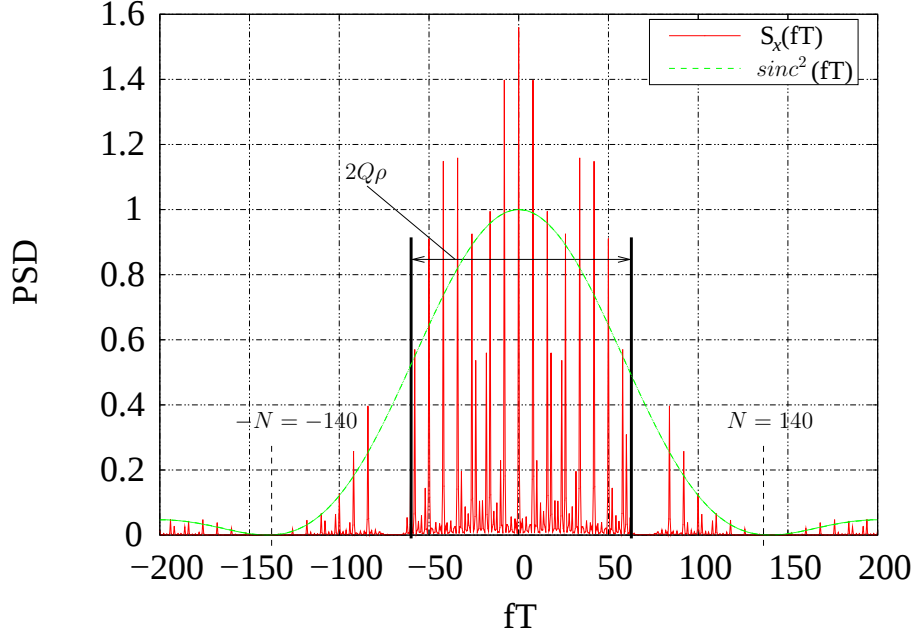
$$U^s(f) = \frac{N}{T} \sum_{i \in \mathbb{Z}} U\left(f - \frac{iN}{T}\right). \quad (5.77)$$

by which we see that $U^s(f)$ is periodic. As a consequence, for each subcarrier we have more than one peak (in detail, one peak for each repetition of $U(f)$). However, the factor $\text{sinc}^2\left(\frac{fT}{N}\right)$ deriving from the analog-to-digital converter $\Pi(t)$ in (5.6) plays the role of an envelope for the signal PSD; such an envelope strongly attenuates and distorts all peaks of $U^s(f)$ close to $f = \pm N/T$ and out of the range $[-N/T, N/T]$. In other words, the peaks which are not distorted are those close to the frequency zero.

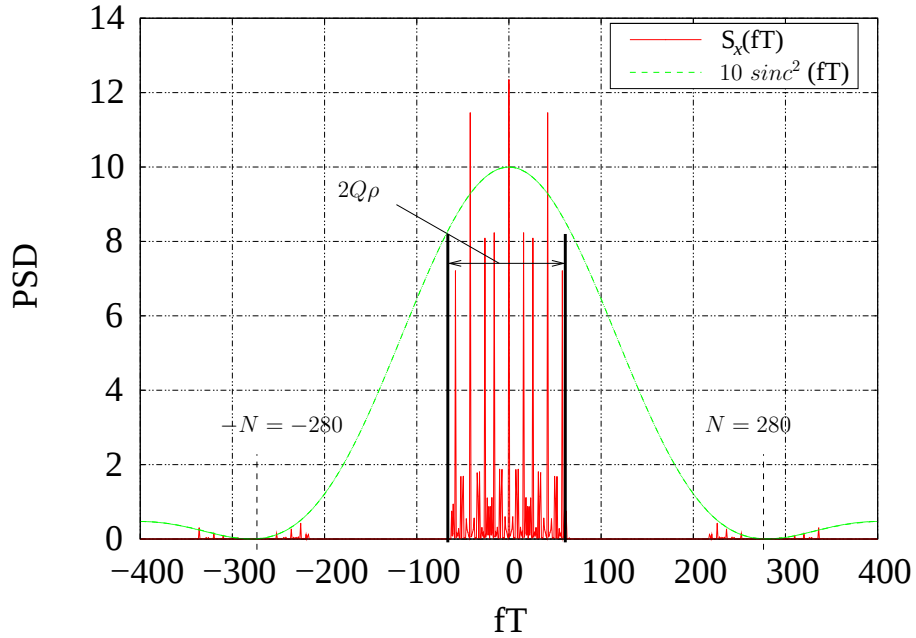
In Fig. 5.4(a), we represent $S_x(f)$ in (5.76) where $\rho = FT = 2$, $\lambda = 2 \cdot 10^{-1}$, $\tau_{max} = 2.14 \cdot 10^{-6}$, $Q = 31$ and $N = 140$. As explained in [85], it is suggested to choose T and F according to the channel parameters τ_{max} and ν_{max} so that

$$\frac{T}{F} = \frac{\tau_{max}}{\nu_{max}}. \quad (5.78)$$

Thus, we derive $T = 6.77 \cdot 10^{-6}$ and $F = 2.96 \cdot 10^5$. We consider $\sigma_x^2 = 1$ and the optimized pulse $u(i)$ provided in [85]. On the “ x ” axis we report the frequency f normalized to the OFDM period T , thus each subcarriers is centered around $fT = n\rho$. We also report the profile of the envelope $\text{sinc}^2\left(\frac{fT}{N}\right)$ in order to see its effects on the spectrum. Looking at the picture, we note that the transmitted process $x(t)$ is not bandlimited. Moreover, if we try to derive a bandwidth definition, based for example on the power spectral concentration, the bandwidth value B we obtain is greater than $fT = N = 140$; thus $T_s < 2B$ (where we know that $T_S = T/N$) and the Nyquist condition is not satisfied.



(a) $Q = 31$ and $N = 140$



(b) $Q = 31$ and $N = 280$

Figure 5.4: PSD of the transmitted signal $x(t)$ for DFT-OFDM.

Analyzing the spectrum, we note that most of the power is concentrated in the range $fT \in [-Q\rho, Q\rho]$ (where in our case $Q\rho = 62$). All peaks out of this range are generated by the periodicity of the function $U^s(f)$ (see (5.77)) and in particular, the most powerful ones are related to the most external subcarriers. Hence, in order to reduce the bandwidth we need to reduce the number Q of subcarriers or to increase the oversampling factor N . In other words, we need

$$\frac{Q\rho}{N} \ll 1. \quad (5.79)$$

Thus in Fig. 5.4(b) we double the oversampling factor N from 140 to 280. We can see that all secondary peaks are almost completely suppressed and hence a possible bandwidth definition for the process $x(t)$ is given by the frequency where the Q -th subcarrier is centered, increased by the bandwidth of a pulse $u(t)$; thus

$$BT \simeq Q\rho + B_u T \quad (5.80)$$

where B_u is a whichever bandwidth definition for the pulse $u(t)$. Since in general $Q\rho \gg B_u T$, we obtain

$$BT \simeq Q\rho. \quad (5.81)$$

We will show that, when (5.79) is satisfied, the bandwidth definition (5.81) will be the same for all the multicarrier modulation schemes we will consider.

We now considering the Nyquist condition,

$$\frac{1}{T_s} \geq 2B \quad (5.82)$$

and substituting the bandwidth definition (5.81) we get

$$Q\rho \leq \frac{N}{2} \quad (5.83)$$

which is certainly satisfied when (5.79) is satisfied.

DCT-OFDM Spectrum

We replace the filter expression (5.33) for the DCT-OFDM scheme in (5.74) and the PSD expression becomes

$$S_x(fT) = \frac{\sigma_x^2}{N} \text{sinc}^2\left(\frac{fT}{N}\right) \sum_{n=0}^{2Q} \frac{\beta_n}{4} \left| U^s\left(f - \frac{n}{2}F\right) + U^s\left(f + \frac{n}{2}F\right) e^{-j\pi \frac{n}{N}\rho} \right|^2. \quad (5.84)$$

Thus, in this case, two functions $U^s(f)$ are associated to each subcarrier, one at the positive frequency $f = n/2F$ and the other one at the opposite, i.e., $f = -n/2F$, except for the $n = 0$ subcarrier which has just one repetition $U^s(f)$ centered at frequency 0. Thus, since n spans from 0 to $2Q$, the most external $U^s(f)$ replicas are those associated to $n = 2Q$, placed at $f = \pm QF$ as is for the PSD of the DFT-OFDM case. Since also the envelope represented by the function $\text{sinc}^2\left(\frac{fT}{N}\right)$ is the same, we can derive that all the considerations carried out for DFT-OFDM spectrum are still valid. In particular, in order to respect the Nyquist condition, we need $\frac{Q\rho}{N} \ll 1$ and if this condition is satisfied, we can define the signal bandwidth as in (5.81). Finally, we note that when we suppress all odd subcarriers as suggested in Section 5.4, the bandwidth definition does not change.

DTT-OFDM Spectrum

The PSD for the DTT-OFDM signal is

$$S_x(fT) = \frac{\sigma_x^2}{N} \text{sinc}^2\left(\frac{fT}{N}\right) \cdot \left[\sum_{n=0, n \text{ even}}^{2Q} \frac{\beta_n}{4} \left| U^s\left(f - \frac{n}{2}F\right) + U^s\left(f + \frac{n}{2}F\right) e^{-j\pi \frac{n}{N}\rho} \right|^2 + \sum_{n=0, n \text{ odd}}^{2Q} \frac{\gamma_n}{4} \left| U^s\left(f - \frac{n}{2}F\right) - U^s\left(f + \frac{n}{2}F\right) e^{-j\pi \frac{n}{N}\rho} \right|^2 \right]. \quad (5.85)$$

It is very similar to the PSD of the DCT-OFDM, but here we have a slightly different behaviour between the even subcarriers and the odd ones. Also in this case the signal bandwidth is given by (5.81).

ODTT-OFDM Spectrum

Finally, by replacing the filter expression (5.57) for the ODTT-OFDM scheme in (5.74), we get the PSD expression

$$S_x(fT) = \frac{\sigma_x^2}{N} \text{sinc}^2\left(\frac{fT}{N}\right) \left[|U^s(f)|^2 + \sum_{n=2, [n]_4=0}^{2Q+1} A_n + \sum_{n=2, [n]_4=1}^{2Q+1} B_n + \sum_{n=1, [n]_4=2}^{2Q+1} C_n + \sum_{n=2, [n]_4=3}^{2Q+1} D_n \right] \quad (5.86)$$

where

$$\begin{aligned} A_n &= \frac{\beta_n}{4} \left| U^s\left(f - \frac{n}{2}F\right) + U^s\left(f + \frac{n}{2}F\right) e^{-j\pi \frac{n}{N}\rho} \right|^2 \\ B_n &= \frac{\gamma_n}{4} \left| U^s\left(f - \frac{n-1}{2}F\right) - U^s\left(f + \frac{n-1}{2}F\right) e^{j\pi \frac{(n-1)(N-1)}{N}\rho} \right|^2 \\ C_n &= \frac{\gamma_n}{4} \left| U^s\left(f - \frac{n}{2}F\right) - U^s\left(f + \frac{n}{2}F\right) e^{-j\pi \frac{n}{N}\rho} \right|^2 \\ D_n &= \frac{\beta_n}{4} \left| U^s\left(f - \frac{n-1}{2}F\right) + U^s\left(f + \frac{n-1}{2}F\right) e^{j\pi \frac{(n+1)(N-1)}{N}\rho} \right|^2. \end{aligned}$$

The PSD is represented in Fig 5.5 when $\lambda = 2 \cdot 10^{-1}$, $\tau_{max} = 2.14 \cdot 10^{-6}$, $\rho = 1$, $Q = 31$, and $N = 140$. In that case, since $\rho = 1$, with respect to the set of parameters of Fig. 5.4, we satisfy the Nyquist condition and we do not need to increase N (or, equivalently, to reduce Q). The pulse $u(t)$ we have employed to obtain Fig 5.5 is that derived in [90] and not the pulse of [85], employed in Fig. 5.4 for the DFT spectrum.

By analyzing the PSD in (5.86), we note that subcarriers are divided into four sets (A_n, B_n, C_n, D_n) , depending on the value of $[n]_4$. In detail, when $[n]_4$ spans from 0 to 3, the subcarriers belonging to A_n and B_n are overlapped at the

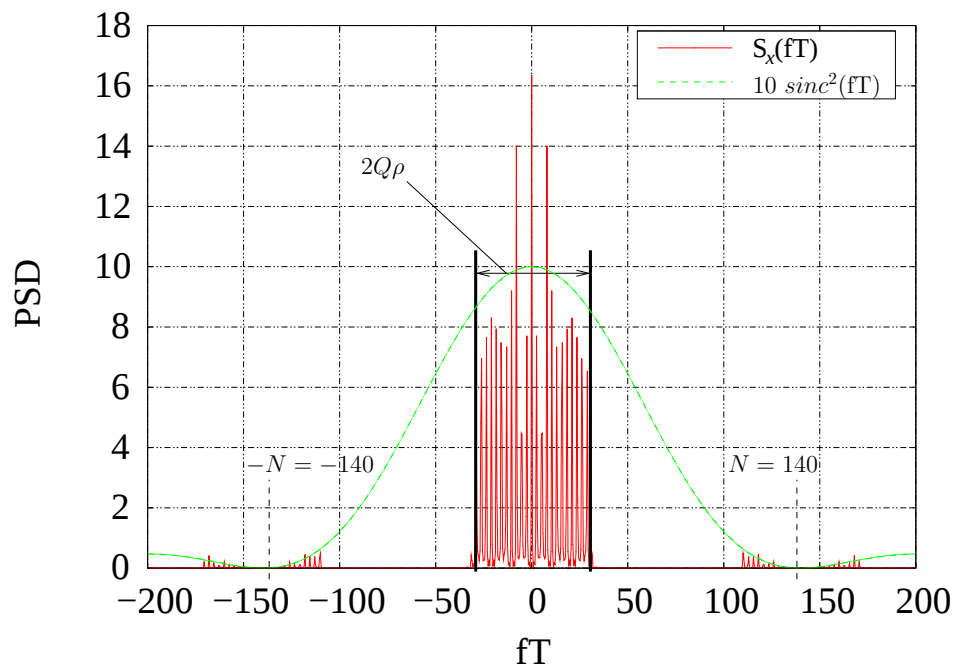


Figure 5.5: PSD of the transmitted signal $x(t)$ for ODTT-OFDM.

same frequency, and the same is for the subcarriers belonging to C_n and D_n . Finally, we see that the most external subcarrier has an odd index $n = 2Q + 1$; hence is a member of B_n or D_n and so it is centered at the frequency $f = QF$, as for all previous multicarrier format. Thus the bandwidth definition (5.81) is valid also for that format.

Conclusions

The work presented in this thesis deals with two modulation schemes, namely continuous phase modulations (CPMs) and multicarrier schemes. The purpose of this related research activity is to overcome the main drawbacks of these modulations, focusing on transmission over satellite channels (hence, affected by phase noise), for CPMs, and on doubly-selective channels, for multicarrier schemes.

In detail, by applying the Arnold and Loeliger method, we computed the CPM information rate over AWGN, as well as over channels affected by phase noise (PN). In particular, we considered a Wiener PN process and also a more practical phase noise process, typical of some satellite real channels (SAT-MODE PN), for which we derived a double first order auto-regressive model. Hence, we described the PN effect on the CPM information rate. Then, we have proposed a novel algorithm for the evaluation of the Markov capacity of CPM signals over the AWGN channel. Our goal was to maximize the overall spectral efficiency in bps/Hz, hence taking into account the signal bandwidth also, through optimization of the transition probabilities of a Markov input of given order. In order to ensure a mathematical tractability of the considered problem, we employed the Carson's rule for CPM bandwidth, which has been shown to possess good accuracy as well, although in the numerical results we have also considered a bandwidth definition related to the spectral power concentration. The numerical results showed that the spectral efficiencies achieved by inputs optimized with the proposed method outperform those obtained by

i.u.d. inputs, for both binary and non-binary CPMs, as well as those obtained by memoryless inputs optimized following a technique in the literature.

The problem of MAP *symbol* detection for CPM signals transmitted over an ideal coherent channel has been faced. First, we have recalled the optimal algorithm, showing that its complexity is unmanageable in several cases of practical interest. Then, we have presented some reduced-complexity algorithms, based on two different approaches. The first approach consists of applying reduced-search techniques to the optimal algorithm, while the second approach consists of resorting to decompositions of a CPM signal allowing to reduce the number of states required to describe the system. In particular, we have focused on the Mengali and Morelli decomposition, deriving a BCJR-like algorithm for MAP symbol detection by means of the factor graphs and the sum-product algorithm. Finally, we have shown several simulation results proving that, for each considered CPM format, there exists a receiver whose complexity is significantly lower than that of the optimal one, and whose performance is practically the same. In particular, the approach providing the simplest front end is that based on the CPM decomposition proposed by Moqvist and Aulin, while the most convenient solution in terms of trellis complexity results that based on the CPM decomposition proposed by Mengali and Morelli, possibly combined with proper techniques for reduced trellis search.

We have faced the problem of MAP symbol detection for CPM signals transmitted over a channel which also introduces phase noise. All design stages have been carried out with the aim of keeping the complexity of the proposed receivers fairly low. First, based on two different design approaches, several detection algorithms have been designed under the assumption of ideal frequency synchronization. Then, by means of computer simulations, we have selected the algorithms which provide the best performance/complexity tradeoff for various scenarios of interest. In particular, it has been shown that the selected algorithms, despite being relatively simple, ensure a performance very close to that of the optimal coherent receiver.

Finally, we have compared four multicarrier transmission systems in a

discrete-time, oversampled domain, over doubly-selective channels. We have shown that all schemes can be implemented by resorting to fast transforms (like DFT, DCT or DST), and for all schemes we have assumed either a rectangular prototype pulse or a suitable designed base pulse well localized in both time and frequency domains. Underlying the connection between OFDM systems and the Gabor communication theory, we derive a new scheme which can be very promising, since it is based on an instance of the Wilson base, which is a clever way to design well-localized and orthogonal frames in the windowed Fourier framework.

Bibliography

- [1] J.B. Anderson, T. Aulin, and C-E.W. Sundberg, *Digital Phase Modulation*, Plenum Press, New York, 1986.
- [2] B. E. Rimoldi, “A decomposition approach to CPM,” *IEEE Trans. Inform. Theory*, vol. 34, pp. 260–270, Mar. 1988.
- [3] P. Moqvist and T. Aulin, “Power and bandwidth efficient serially concatenated CPM with iterative decoding,” in *Proc. IEEE Global Telecommun. Conf.*, San Francisco, CA, Dec. 2000.
- [4] P. Moqvist and T. M. Aulin, “Serially concatenated continuous phase modulation with iterative decoding,” *IEEE Trans. Commun.*, vol. 49, pp. 1901–1915, Nov. 2001.
- [5] E. A. Pascale, “Advanced decoding techniques for serially concatenated continuous phase modulations,” Tech. Rep. EWP 2267, ESA-ESTEC, Noordwijk, Netherlands, Oct. 2004.
- [6] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, “Optimal decoding of linear codes for minimizing symbol error rate,” *IEEE Trans. Inform. Theory*, vol. 20, pp. 284–287, Mar. 1974.
- [7] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, “Factor graphs and the sum-product algorithm,” *IEEE Trans. Inform. Theory*, vol. 47, pp. 498–519, Feb. 2001.

- [8] F.R. Kschischang and B.J. Frey, "Iterative decoding of compound codes by probability propagation in graphical models," *IEEE J. Select. Areas Commun.*, pp. 219–231, February 1998.
- [9] S. Benedetto and G. Montorsi, "Design of parallel concatenated convolutional codes," *IEEE Trans. Commun.*, vol. 44, pp. 591–600, May 1996.
- [10] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: turbo-codes," in *Proc. IEEE Intern. Conf. Commun.*, Geneva, Switzerland, May 1993, pp. 1064–1070.
- [11] C. Berrou and A. Glavieux, "Near optimum error correcting coding and decoding: turbo-codes," *IEEE Trans. Commun.*, vol. 44, no. 10, pp. 1261–1271, October 1996.
- [12] I. Bar-David and S. Shamai, "On information transfer by envelope-constrained signals over the AWGN channel," *IEEE Trans. Inform. Theory*, vol. 34, pp. 371–379, May 1988.
- [13] M. Campanella and G. Mamola, "On the channel capacity for constant envelope signals with effective bandwidth constraints," *IEEE Trans. Commun.*, vol. 38, pp. 1164–1172, Aug. 1990.
- [14] D. Agrawal and U. Madhow, "Turbo coded CPM modulation," in *Proc. IEEE international Symposium on Information Theory and its Applications*, Honolulu, Hawaii, US, nov 2000.
- [15] K. C. Yu and A. Goldsmith, "Linear models and capacity bounds for continuous phase modulation," in *Proc. IEEE Intern. Conf. Commun.*, Apr/May 2002, vol. 2, pp. 722–726.
- [16] U. Mengali and M. Morelli, "Decomposition of M -ary CPM signals into PAM waveforms," *IEEE Trans. Inform. Theory*, vol. 41, pp. 1265–1275, Sept. 1995.

- [17] C. H. Kuo and K. M. Chugg, "On the bandwidth efficiency of CPM signals," in *Proc. IEEE Military Comm. Conf. (MILCOM)*, Oct.-Nov. 2004, pp. 218–224.
- [18] C. H. Kuo and K. M. Chugg, "Improving the bandwidth efficiency and of CPM signals via shaping and iterative detection," in *Proc. IEEE Military Comm. Conf. (MILCOM)*, Oct. 2005, pp. 596–602.
- [19] D. Delaruelle, "BSDT — Technical note TN10: Study of achievable information rate with CPM signals," Tech. Rep., Sept. 2006, ESA Contract No. 19370.
- [20] D. Arnold and H.-A. Loeliger, "On the information rate of binary-input channels with memory," in *Proc. IEEE Intern. Conf. Commun.*, June 2001, vol. 9, pp. 2692–2695.
- [21] D. M. Arnold, H.-A. Loeliger, P. O. Vontobel, A. Kavčič, and W. Zeng, "Simulation-based computation of information rates for channels with memory," *IEEE Trans. Inform. Theory*, vol. 52, no. 8, pp. 3498–3508, Aug. 2006.
- [22] Newtec, ESA, and SES ASTRA, "MODEM LAYER SPECIFICATION SMD-SP-SES-01000-1-2," July 2004.
- [23] A. Kavčič, "On the capacity of Markov sources over noisy channels," in *Proc. IEEE Global Telecommun. Conf.*, 2001, pp. 2997–3001.
- [24] R. Gallager, *Information Theory and Reliable Communication*, John Wiley & Sons, 1968.
- [25] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, John Wiley & Sons, Inc., New York, 1991.
- [26] H. D. Pfister, J. B. Soriaga, and P. H. Siegel, "On the achievable information rates of finite-state ISI channels," in *Proc. IEEE Global Telecommun. Conf.*, San Antonio, TX, 2001, pp. 2992–2996.

-
- [27] B. McMillan, "The basic theorems of information theory," in *Ann. Math. Statist.*, June 1953, vol. 24, pp. 196–219.
 - [28] Z. Zhang, T. M. Duman, and E. M. Kurtas, "Information rates of binary-input intersymbol interference channels with signal-dependent media noise," in *Ann. Math. Statist.*, January 2003, vol. 39, pp. 599–607.
 - [29] U. Mengali and A. N. D'Andrea, *Synchronization Techniques for Digital Receivers (Applications of Communications Theory)*, Plenum Press, 1997.
 - [30] H. Meyr, M. Moeneclaey, and S. A. Fechtel, *Digital communication receivers*, John Wiley & Sons, 1998.
 - [31] A. B. Carlson, *Communication Systems: An Introduction to Signals and Noise in Electrical Communication*, McGraw-Hill, 1986.
 - [32] A. Kavčič, X. Ma, and N. Varnica, "Matched information rate codes for partial response channels," *IEEE Trans. Inform. Theory*, vol. 51, no. 3, pp. 973–989, Mar. 2005.
 - [33] X. Ma, A. Kavčič, and M. Mitzenmacher, "Power spectral of good codes for partial response channels," submitted to *IEEE Trans. Commun.*, June 2003.
 - [34] K. B. Petersen and M. S. Pedersen, "The matrix cookbook," Tech. Rep., Technical University of Denmark, Feb. 2006.
 - [35] E. Seneta, *Non-Negative Matrices and Markov Chains*, Springer-Verlag, New York, 1981.
 - [36] A. Barbieri and G. Colavolpe, "Soft-output detection of CPM signals transmitted over channels affected by phase noise," in *Proc. 14th European Signal Processing Conf. (EUSIPCO'06)*, Florence, Italy, Sept. 2006.

- [37] K. R. Narayanan and G. L. Stüber, “Performance of trellis-coded CPM with iterative demodulation and decoding,” *IEEE Trans. Commun.*, vol. 49, pp. 676–687, Apr. 2001.
- [38] M. Xiao and T. M. Aulin, “Serially concatenated continuous phase modulation with symbol interleavers: performance, properties and design principles,” in *Proc. IEEE Global Telecommun. Conf.*, November–December 2004, pp. 179–183.
- [39] G. Colavolpe, G. Ferrari, and R. Raheli, “Reduced-state BCJR-type algorithms,” *IEEE J. Select. Areas Commun.*, vol. 19, no. 5, pp. 848–859, May 2001.
- [40] D. Fertonani, A. Barbieri, and G. Colavolpe, “Reduced-complexity BCJR algorithm for turbo equalization,” *IEEE Trans. Commun.*, 2007, to appear (available on www.tlc.unipr.it/fertonani).
- [41] R. Balasubramanian, M. P. Fitz, and J. K. Krogmeier, “Optimal and suboptimal symbol-by-symbol demodulation of continuous phase modulated signals,” *IEEE Trans. Commun.*, vol. 46, no. 12, pp. 1662–1668, Dec. 1998.
- [42] J. Huber and W. Liu, “An alternative approach to reduced-complexity CPM-receivers,” *IEEE J. Select. Areas Commun.*, vol. 7, pp. 1437–1449, Dec. 1989.
- [43] W. Tang and E. Shwedyk, “ML estimation of symbol timing and carrier phase for CPM in Walsh signal space,” *IEEE Trans. Commun.*, vol. 49, pp. 969–974, June 2001.
- [44] Q. Zhao and G. L. Stüber, “Robust time and phase synchronization for continuous phase modulations,” submitted to *IEEE Trans. Commun.*, 2005.

- [45] Q. Zhao and G. L. Stüber, “Joint time and phase recovery for cpm and its asymptotic behavior,” in *Proc. IEEE Global Telecommun. Conf.*, Nov.-Dec. 2005, pp. 2302–2306.
- [46] P. A. Laurent, “Exact and approximate construction of digital phase modulations by superposition of amplitude modulated pulses (AMP),” *IEEE Trans. Commun.*, vol. 34, pp. 150–160, Feb. 1986.
- [47] E. Perrins and M. Rice, “PAM decomposition of M -ary multi- h CPM,” *IEEE Trans. Commun.*, vol. 53, no. 12, pp. 2065–2075, Dec. 2005.
- [48] G. Colavolpe and R. Raheli, “Reduced-complexity detection and phase synchronization of CPM signals,” *IEEE Trans. Commun.*, vol. 45, pp. 1070–1079, Sept. 1997.
- [49] G. Ungerboeck, “Adaptive maximum likelihood receiver for carrier-modulated data-transmission systems,” *IEEE Trans. Commun.*, vol. com-22, pp. 624–636, May 1974.
- [50] G. Colavolpe and R. Raheli, “Noncoherent sequence detection of continuous phase modulations,” *IEEE Trans. Commun.*, vol. 47, pp. 1303–1307, Sept. 1999.
- [51] G. D. Forney, Jr., “Lower bounds on error probability in the presence of large intersymbol interference,” *IEEE Trans. Commun.*, vol. 20, pp. 76–77, Feb. 1972.
- [52] G. Colavolpe and A. Barbieri, “On MAP symbol detection for ISI channels using the Ungerboeck observation model,” *IEEE Commun. Letters*, vol. 9, no. 8, pp. 720–722, Aug. 2005.
- [53] A. Barbieri and G. Colavolpe, “Simplified soft-output detection of CPM signals over coherent and phase noise channels,” *IEEE Trans. Wireless Commun.*, vol. 6, no. 7, pp. 2486–2496, July 2007.

- [54] X. Huang and Y. Li, "MMSE-optimal approximation of continuous-phase modulated signal as superposition of linearly modulated pulses," *IEEE Trans. Commun.*, vol. 53, no. 7, pp. 1166–1177, July 2005.
- [55] X. Huang and Y. Li, "Simple CPM receivers based on a switched linear modulation model," *IEEE Trans. Commun.*, vol. 53, no. 7, pp. 1100–1103, July 2005.
- [56] M. P. Wylie-Green, "A new finite series expansion of continuous phase modulated waveforms," *IEEE Trans. Commun.*, vol. 55, no. 8, pp. 1547–1556, Aug. 2007.
- [57] P. Moqvist and T. Aulin, "Orthogonalization by principal components applied to CPM," *IEEE Trans. Commun.*, vol. 51, pp. 1838–1845, Nov. 2003.
- [58] V. Franz and J. B. Anderson, "Concatenated decoding with a reduced-search BCJR algorithm," *IEEE J. Select. Areas Commun.*, vol. 16, no. 2, pp. 186–195, Feb. 1998.
- [59] D. Bokolamulla, A. Hansson, and T. Aulin, "Low-complexity iterative detection based on bi-directional trellis search," in *Proc. IEEE International Symposium on Information Theory*, Yokohama, Japan, June-July 2003, p. 396.
- [60] P. Roberston, E. Villebrun, and P. Hoeher, "Optimal and sub-optimal maximum a posteriori algorithms suitable for turbo decoding," *European Trans. Telecommun.*, vol. 8, no. 2, pp. 119–125, March/April 1997.
- [61] S. Crozier and P. Guinand, "High-performance low-memory interleaver banks for turbo-codes," in *Proc. IEEE Vehicular Tech. Conf., VTC Fall 2001*, Atlantic City, NJ, USA, Oct. 2001, pp. 2394–2398.
- [62] M. Peleg, S. Shamai (Shitz), and S. Galán, "Iterative decoding for coded noncoherent MPSK communications over phase-noisy AWGN channel," *IEE Proc. Commun.*, vol. 147, pp. 87–95, Apr. 2000.

- [63] A. Anastasopoulos and K. M. Chugg, "Adaptive iterative detection for phase tracking in turbo coded systems," *IEEE Trans. Commun.*, vol. 49, pp. 2135–2144, Dec. 2001.
- [64] G. Colavolpe, A. Barbieri, and G. Caire, "Algorithms for iterative decoding in the presence of strong phase noise," *IEEE J. Select. Areas Commun.*, vol. 23, no. 9, pp. 1748–1757, Sept. 2005.
- [65] Q. Zhao, H. Kim, and G. L. Stüber, "Adaptive iterative phase synchronization for serially concatenated continuous phase modulation," in *Proc. IEEE Military Comm. Conf. (MILCOM)*, Oct. 2003, pp. 78–83.
- [66] Q. Zhao, H. Kim, and G. L. Stüber, "Innovation-based MAP estimation with application to phase synchronization for serially concatenated CPM," *IEEE Trans. Wireless Commun.*, vol. 5, no. 5, pp. 1033–1043, May 2006.
- [67] A. Anastasopoulos and K. M. Chugg, "Adaptive soft-input soft-output algorithms for iterative detection with parametric uncertainty," *IEEE Trans. Commun.*, vol. 48, no. 10, pp. 1638–1649, Oct. 2000.
- [68] A. P. Worthen and W. E Stark, "Unified design of iterative receivers using factor graphs," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 843–849, Feb. 2001.
- [69] A. N. D'Andrea, U. Mengali, and G. M. Vitetta, "Approximate ML decoding of coded PSK with no explicit carrier phase reference," *IEEE Trans. Commun.*, vol. 42, pp. 1033–1040, Feb./Mar./April 1994.
- [70] G. D. Forney, Jr., "The Viterbi algorithm," *Proc. IEEE*, vol. 61, pp. 268–278, Mar. 1973.
- [71] G. Ferrari, G. Colavolpe, and R. Raheli, "Finite memory: optimality and reality," in *Proc. Intern. Symp. on Turbo Codes & Relat. Topics*, Brest, France, Sept. 2003, pp. 1–8.

- [72] A. Papoulis, *Probability, Random Variables and Stochastic Processes*, McGraw-Hill, New York, NY, 1991.
- [73] A. G. Amat, C. A. Nour, and C. Douillard, “BSDT — Technical note TN05: Code selection for the cpm return link,” Tech. Rep., July 2007, ESA Contract No. 19370.
- [74] J. A. C. Bingham, “Multicarrier modulation for data transmission: An idea whose time has come,” *IEEE Commun. Mag.*, pp. 5–14, May 1990.
- [75] X. Cai and G. B. Giannakis, “Bounding performance and suppressing intercarrier interference in wireless mobile OFDM,” *IEEE Trans. Commun.*, vol. 51, no. 12, pp. 2047–2056, Dec. 2003.
- [76] L. Tomba, “On the effect of Wiener phase noise in OFDM systems,” *IEEE Trans. Commun.*, vol. 46, no. 5, pp. 580–583, May 1998.
- [77] A. Stamoulis, S. N. Diggavi, and N. Al-Dhahir, “Inter-carrier interference in MIMO OFDM,” *IEEE Trans. Signal Processing*, vol. 50, no. 10, pp. 2451–2464, Oct. 2002.
- [78] S. Hwang and P. Schniter, “Efficient communication over highly spread underwater acoustic channels,” in *Proc. of the ACM Intern. Workshop on UnderWater Networks (WUWNet)*, Sept. 2007.
- [79] N. Al-Dhahir, H.S. Minn, and S. Satish, “Optimum DCT-based multi-carrier transceivers for frequency-selective channels,” *IEEE Trans. on Communications*.
- [80] P. Tan and N. C. Beaulieu, “A comparison of DCT-based OFDM and DFT-based OFDM in frequency offset and fading channels,” *IEEE Trans. Commun.*, vol. 54, no. 11, pp. 2113–2125, Nov. 2006.
- [81] G. D. Mandyam, “Sinusoidal transforms in OFDM systems,” *IEEE Trans. on Broadcasting*.

- [82] W. Kozek and A. F. Molisch, "On the eigenstructure of underspread WS-SUS channels," in *Proc. of the IEEE Workshop on Signal Proc. Adv. in Wireless Comm. (SPAWC)*, Paris, France, Apr. 1997, pp. 325–328.
- [83] W. Kozek and A. F. Molisch, "Nonorthogonal pulseshapes for multicarrier communications in doubly dispersive channels," *IEEE J. Select. Areas Commun.*, vol. 16, no. 8, pp. 1579–1589, Oct. 1998.
- [84] G. Matz and F. Hlawatsch, "Time-varying communication channels: fundamentals, recent developments, and open problems," in *Proc. European Signal Processing Conference (EUSIPCO)*, Florence, Italy, Sept. 2006.
- [85] T. Strohmer and S. Beaver, "Optimal OFDM design for time-frequency dispersive channels," *IEEE Trans. Commun.*, vol. 51, no. 7, pp. 1111–1122, July 2003.
- [86] A. Vahlin and N. Holte, "Optimal finite duration pulses for OFDM," *IEEE Trans. Commun.*, vol. 44, no. 1, pp. 10–14, Jan. 1996.
- [87] I. Daubechies, *Ten Lectures on Wavelets*, SIAM J., 1991.
- [88] R. L. Allen and D. W. Mills, *Signal Analysis – time, frequency, scale, and structure*, John Wiley & Sons, 2004.
- [89] B. Le Floch, M. Alard, and C. Berrou, "Coded orthogonal frequency division multiplex," *Proc. IEEE*, vol. 83, no. 6, pp. 982–996, June 1995.
- [90] I. Daubechies, S. Jaffard, and J. L. Journé, "A simple Wilson orthonormal basis with exponential decay," *SIAM J. Math Anal.*, vol. 22, pp. 554–573, 1991.
- [91] N. Benvenuto and G. Cherubini, *Algorithms for Communications Systems and Their Applications*, John Wiley & Sons, 2002.
- [92] P. A. Bello, "Characterization of randomly time-variant linear channels," *IEEE Trans. Commun. Systems*, vol. 11, pp. 360–393, 1963.

- [93] A. Barbieri, G. Caire, and U. Mitra, "An approximate eigenmode decomposition for doubly-selective wireless channels," in *Proc. ICASSP*, Las Vegas, NV, Mar.-Apr. 2008.
- [94] J. Boutros and G. Caire, "Iterative multiuser joint decoding: unified framework and asymptotic analysis," *IEEE Trans. Inform. Theory*, vol. 48, no. 7, pp. 1772–1793, July 2002.
- [95] Z. Wang and G. B. Giannakis, "Wireless multicarrier communications," *IEEE Signal Processing Mag.*, pp. 29–48, May 2000.
- [96] J. L. Massey, "Information theory aspects of spread spectrum communications," in *Proc. IEEE Intn'l. Conf. Spread Spectrum Tech. and Applications*, Oulu, Finland, July 1994, vol. 1, pp. 16–21.
- [97] A. V. Oppenheim, R. W. Schaffer, and J. R. Buck, *Discrete-Time Signal Processing*, Prentice-Hall, Englewood Cliffs, New Jersey, 1998, 2nd ed.
- [98] H. Bölcskei, H. Feichtinger, K. Gröchenig, and F. Hlawatsch, "Discrete-time Wilson expansions," *IEEE Transactions on Time-Frequency and Time-Scale Analysis*, pp. 525–528, 1996.
- [99] H. Bölcskei and F. Hlawatsch, "Oversampled Wilson-type cosine modulated filter banks with linear phase," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, pp. 1057–1071, 1998.
- [100] G. Kutyniok and T. Strohmer, "Wilson bases for general time-frequency lattices," *Functional Analysis (math.FA)*, 2004.
- [101] P. Wojdyło, "Characterization of Wilson systems for general lattices," *International Journal of Wavelets, Multiresolution and Information Processing*, vol. 6, pp. 305–314, 2007.
- [102] H. Bölcskei, K. Gröchenig, F. Hlawatsch, and H. G. Feichtinger, "Oversampled Wilson expansions," *IEEE signal processing*, vol. 4, pp. 106–108, 1997.

Acknowledgements

This work was possible only with the support of many people. In the first place, I would like to thank my supervisor Giulio Colavolpe, for his stimulating guidance and for the continuous help. Moreover, I am particularly indebted to all my mate at the Department, especially to my colleague of the SPADiC Lab. It has been an absolute pleasure to work with you. Finally a special thanks to my family and to all my friends, for their infinite patience and continuous support.