



Turbo-Equalization for bandwidth-efficient digital communications over frequency-selective channels

Raphaël Le Bidan

► To cite this version:

Raphaël Le Bidan. Turbo-Equalization for bandwidth-efficient digital communications over frequency-selective channels. Signal and Image processing. INSA de Rennes, 2003. English. NNT: . tel-00110853

HAL Id: tel-00110853

<https://theses.hal.science/tel-00110853>

Submitted on 2 Nov 2006

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

N° d'ordre : D03-12

Thèse

présentée devant

l'Institut National des Sciences Appliquées de Rennes

pour obtenir le titre de

Docteur

spécialité : *Électronique*

Turbo-equalization for bandwidth-efficient digital communications over frequency-selective channels

par

Raphaël LE BIDAN

Soutenue le 7 novembre 2003 devant la commission d'examen :

Président :	J. Citerne	Professeur	IETR / INSA de Rennes
Rapporteurs :	H. Sari	Professeur	SUPELEC
	L. Vandendorpe	Professeur	Université Catholique de Louvain
Examineurs :	A. Glavieux	Professeur	GET / ENST Bretagne
	M. Hélard	Ingénieur R&D	France Télécom R&D
	C. Laot	Maître de Conférence	GET / ENST Bretagne
	D. Leroux	Maître de Conférence	GET / ENST Bretagne

*Travail réalisé au sein du département Signal et Communications de l'ENST Bretagne,
unité CNRS TAMCIC FRE 2658, avec le soutien de France Télécom R&D.*

Acknowledgments

I first wish to express my sincere gratitude to my supervisor, Christophe Laot. His invaluable insight and experience, guidance, encouragement, friendship and constant availability have had a great impact on this thesis. I consider myself to be truly fortunate to have had the opportunity to work so closely with such a talented person.

I express special thanks to Dominique Leroux, Joël Truibuil, André Goalic and Annie Godet for their fruitful advice that have been instrumental in completing the DSP implementation in chapter 5. It is also a real pleasure to thank Maryline H  lard, Charlotte Langlais and Rapha  l Visoz for their continuous interest in this work and for the stimulating discussions we had during our regular meetings over the last three years. The help of Janet Ormrod has greatly contributed to improving the overall quality of this document and is sincerely acknowledged.

I am extremely grateful to the two readers on my PhD dissertation committee, Prof. Hikmet Sari and Prof. Luc Vandendorpe, for their thoughtful comments and valuable suggestions. In addition, I would like to thank my two thesis directors, Prof. Jacques Citerne and Prof. Alain Glavieux, whose respective scientific reputations honor this work.

I wish to thank Prof. Ramesh Pyndiah for giving me the opportunity to undertake my graduate studies at ENST Bretagne. Acknowledgments are extended to all members of the Signal & Communications department for providing me with such an enjoyable and stimulating working environment.

I gratefully acknowledge France Telecom R&D DMR/DDH for their financial support under research contract CRE 011B032. I also thank Texas Instruments and the ELITE University Program for providing us with the necessary DSP development software.

On a personal note, I express my gratitude to my parents and my brother for their lifetime support, encouragement and love. Everything I have accomplished so far is due to them. I also wish to thank my grandparents who have always been a role model in my life.

Finally, I dearly thank Nathalie for her boundless love, emotional support, unending patience and for her many personal sacrifices over the past three years. I can only hope to play the same role in her life as she does in mine. To her I dedicate this thesis.

Contents

Acknowledgments	i
Glossary	vii
1 Introduction	1
2 Digital communications over frequency-selective channels	5
2.1 Coded modulation for band-limited channels	6
2.1.1 Preliminary definitions and notations	7
2.1.2 Trellis-coded modulation	9
2.1.3 Bit-interleaved coded modulation	11
2.1.4 Examples of practical coded modulation schemes	13
2.2 Equalization techniques for frequency-selective channels	18
2.2.1 A mathematical model for transmission over ISI channels	19
2.2.2 Matched filter bound	22
2.2.3 Maximum-likelihood sequence detection	23
2.2.4 Filtering-based equalizers	25
2.2.5 MMSE linear equalization	28
2.2.6 MMSE decision-feedback equalization	30
2.2.7 MMSE interference cancellation	34
2.2.8 Comparison of equalization schemes	37
2.3 Summary	38
3 Iterative equalization and decoding: Turbo-Equalization	41
3.1 On combined equalization and decoding	42
3.2 The MAP turbo-equalizer	43
3.2.1 Log-likelihood ratios	44
3.2.2 MAP turbo-equalization for TCM	45
3.2.3 MAP turbo-equalization for BICM	47
3.3 Asymptotic performance bounds for MAP turbo-equalization	49

3.4	Simulation results	53
3.4.1	Performance over a time-invariant channel	54
3.4.2	Performance over a fully-interleaved multipath Rayleigh fading channel	55
3.4.3	Performance over a quasi-static multipath Rayleigh fading channel	57
3.5	Convergence analysis using EXIT charts	61
3.5.1	The EXIT chart technique	62
3.5.2	EXIT chart analysis for BICM-ID	66
3.5.3	EXIT chart analysis of MAP turbo-equalization for BICM	70
3.5.4	Influence of the channel characteristics on the convergence	72
3.5.5	Influence of the inner code characteristics on the convergence	73
3.6	Concluding remarks	76
4	Low-complexity efficient MMSE Turbo-Equalizers	77
4.1	The general structure of SISO MMSE equalizers	79
4.1.1	The soft symbol mapper	80
4.1.2	The SISO symbol demapper	81
4.2	Finite-length MMSE equalization with <i>a priori</i> information	82
4.2.1	Preliminary notations and definitions	82
4.2.2	The time-varying solution	83
4.2.3	A time-invariant approximation	87
4.3	The MMSE IC-LE	88
4.3.1	Statistical properties of the soft data estimates	89
4.3.2	The infinite-length MMSE IC-LE	90
4.3.3	The finite-length MMSE IC-LE	91
4.3.4	A low-complexity procedure for computing the filter coefficients	93
4.3.5	Additional remarks	94
4.4	Asymptotic performance bounds for MMSE turbo-equalization	96
4.5	Performance results	97
4.5.1	Comparison between the different turbo-equalizers	97
4.5.2	Performance with high-order modulations over low to moderate ISI channels	99
4.5.3	Performance with high-order modulations over severe ISI channels	101
4.6	Frequency-domain MMSE turbo-equalization	105
4.6.1	Derivation of the frequency-domain MMSE IC-LE	105
4.6.2	Complexity issues	107
4.7	Concluding remarks	112
5	DSP implementation of the MMSE IC-LE Turbo-Equalizer	115
5.1	Description of the platform	116
5.2	Overview of the transmission scheme	118

5.2.1	Transmit processing	118
5.2.2	Channel and receiver front-end modeling	120
5.2.3	The turbo-equalization receiver	121
5.3	Implementation of the turbo-equalizer	122
5.3.1	Strategy of development	122
5.3.2	Data quantization	123
5.3.3	The soft mapping module	124
5.3.4	The MMSE IC-LE equalizer	124
5.3.5	The soft demapping module	125
5.3.6	The SISO decoder	125
5.4	System performance	127
5.4.1	Achievable bit-rates and storage requirements	127
5.4.2	Experimental results	128
5.5	Concluding remarks	130
6	Conclusions	133
A	Derivations of the equalizers of chapter 4	137
A.1	Derivation of the finite-length time-varying MMSE equalizer	137
A.2	Derivation of the infinite-length MMSE IC-LE	141
A.3	Derivation of the finite-length MMSE IC-LE	145
A.4	Derivation of the frequency-domain MMSE IC-LE	147
B	On the asymptotic efficiency of algorithms	151
C	The Forward-Backward algorithm	153
C.1	Preliminary definitions	153
C.2	Exposition of the algorithm	155
D	On fixed-point arithmetic	159
D.1	Unsigned fixed-point representation	159
D.2	Signed fixed-point representation	160
D.3	Arithmetic operations	161
D.4	Additional definitions	161
	Bibliography	163

Glossary

APP	<i>A Posteriori</i> Probability
AWGN	Additive White Gaussian Noise
BER	Bit-Error Rate
BICM	Bit-Interleaved Coded Modulation
BICM-ID	Bit-Interleaved Coded Modulation with Iterative Decoding
BPSK	Binary Phase-Shift Keying
DARAM	Dual-Access Random Access Memory
DFE	Decision-Feedback Equalizer
DFT	Discrete Fourier Transform
DSP	Digital Signal Processor
EDGE	Enhanced Data rate for GSM Evolution
EXIT	EXtrinsic Information Transfer
FD	Frequency-Domain
FDE	Frequency-Domain Equalization
FER	Frame-Error Rate
FIR	Finite Impulse Response
FFT	Fast Fourier Transform
GSM	Global System for Mobile communications
IC	Interference Canceller
IC-LE	Interference Canceller - Linear Equalizer
IIR	Infinite Impulse Response
ISI	Intersymbol Interference
LE	Linear Equalizer
LLR	Log-Likelihood Ratio
MAC	Multiply-Accumulate
MAP	Maximum <i>A Posteriori</i>
MFB	Matched-Filter Bound
ML	Maximum Likelihood
MLSD	Maximum-Likelihood Sequence Detection
MMSE	Minimum Mean-Square Error

MSE	Mean-Square Error
OFDM	Orthogonal Frequency-Division Multiplexing
PSK	Phase-Shift Keying
QAM	Quadrature-Amplitude Modulation
QPSK	Quaternary Phase-Shift Keying
RTDX	Real-Time Data eXchange
RSC	Recursive Systematic Convolutional (Code)
SISO	Soft-Input Soft-Output
SNR	Signal to Noise Ratio
SP	Set-Partitioning
TCM	Trellis-Coded Modulation
TD	Time-Domain
WMF	Whitening Matched Filter
ZF	Zero-Forcing

Chapter 1

Introduction

In his landmark 1948 paper which established the theoretical foundations of Information theory, Claude E. Shannon formulated elegantly the problem of communication systems design as follows

“The fundamental problem of communication is that of reproducing at one point either exactly or approximately a message selected at another point”.

In many (if not all) everyday life communications channels, the message picked up by the receiver is very likely to be somehow different from the signal that was sent by the transmitter. Appropriate signal processing techniques are thus required to recover the transmitted data from a noisy and possibly distorted observation at the channel output with the ultimate goal of minimizing the probability of selecting an incorrect message.

Nowadays, the ever-growing demand for multimedia services combined with the increasing commercial success of portable terminals such as mobile telephones confront the communication system designer with the conflicting requirements of transmitting information at higher data rates, with lower error probability and over channels affected by severe disturbance while maintaining low cost and low power consumption for the equipment. In addition, an efficient utilization of the spectrum resources is of premium importance to avoid future congestion in the frequency bands available for radio transmissions. As an illustration, even communication links for deep-space missions which were traditionally considered as severely power-limited and essentially unaffected by bandwidth limitations are expected to encounter such restrictions with the increasing number of simultaneous missions.

This challenging task calls for the development of more sophisticated solutions than those currently in use. At the same time, the complexity of the resulting transmissions systems must remain compatible with the current hardware and software technologies evolutions to be implementable in practice.

Context of this work

This work was motivated by the desire to achieve reliable communications by making efficient use of the available resource (bandwidth and power) at a reasonable cost over channels affected by severe disturbance. Specifically, we focus here on the class of *frequency-selective* channels, which are subject to *intersymbol interference*. Intersymbol interference arises when successive transmitted symbols are smeared together in time by the communication channel to the extent that they overlap at the receiver side. Such a phenomenon is commonly encountered over radio links for example, where the signal at the receiving unit is formed by the superposition of multiple propagations paths affected by different delays of arrival.

Single-carrier modulation combined with *equalization* techniques is a well-established technology to combat intersymbol interference. Other solutions exist (multi-carrier modulation for instance) but will not be considered here. The topic of equalization dates back to the early sixties, so that it would seem unlikely that anything new could be said about it at the current date. However, the introduction of *turbo-codes* and more generally *turbo-processing* has renewed considerable interest in the field.

Modern communication systems typically consist of a cascade of several subsystems, each optimized to perform a single task. Such a decoupling greatly simplifies the receiver design in practice. However, functions located higher up in the chain do not benefit from information derived by subsequent processing stages. With turbo-processing, an iterative exchange of information is established between two or more receiver functions in order to improve the overall system performance. An interesting example of turbo-processing is *turbo-equalization*, a concept introduced at ENST Bretagne in 1995. Turbo-equalization combines equalization with channel decoding in an efficient manner and offers significant gains with respect to the conventional approach where equalization and decoding are realized independently.

In this thesis, we focus on the combination of turbo-equalization with bandwidth-efficient coded modulation schemes, where channel coding and modulation are optimized in order to achieve significant coding gains over conventional uncoded transmission, but without compromising the spectral efficiency of the transmission system. Particular attention is devoted to the design of low-complexity receivers able to cope with multilevel modulations over intersymbol interference channels exhibiting long delay spreads.

Outline of the thesis

This dissertation is organized in 6 chapters. We expect the reader to be familiar with the basics of digital communications and channel coding.

Chapter 2 presents some relevant background material regarding the respective topics of coded modulation and equalization. The first section of this chapter offers a review of the code/modulation design problem for bandlimited channels. Specifically, we focus on two practical coded modulation schemes that have found widespread use in current transmission systems, namely *trellis-coded modulation* (TCM) and *bit-interleaved coded modulation* (BICM). Both solutions are exposed and compared by considering different typical transmission scenarios. In the second part of this chapter, we present a structured and detailed survey of classical equalization techniques. A theoretical performance comparison is provided and illustrated by a simple case-study.

We turn our attention in chapter 3 to the important problem of combining equalization and decoding in an efficient manner at the receiver side. After a brief survey of the conventional solutions advocated in the literature, we introduce the turbo-equalization scheme in its standard form for TCM and BICM transmission systems. This structure relies on a trellis-based so-called “BCJR-MAP” equalizer which is optimum in the sense of minimizing the symbol error probability. We examine the asymptotic performance of this receiver in the presence of perfect *a priori* information and present several simulation results over different transmission scenarios. We finally apply a semi-analytical analysis tool, the EXIT chart, to the turbo-equalization scheme in order to gain more insight into the convergence behavior of the iterative process.

Chapter 4 is devoted to the study of low-complexity turbo-equalizers optimized according to the minimum-mean square error criterion. These structures provide an interesting alternative to conventional MAP turbo-equalization for multilevel signaling over long delay spread channels, where the complexity of the latter solution is usually untractable. We review the major contributions in the field and introduce a novel equalization scheme which is derived both in infinite-length form as well as under finite-length realization constraints. Simulation results are presented in order to evaluate the performance of the proposed turbo-equalization scheme. A comparison with the optimum MAP turbo-equalizer is provided. Finally, we investigate the realization of the turbo-equalizer in the frequency-domain and examine the complexity reductions that may be achieved with respect to a conventional time-domain implementation over channels with a very large number of coefficients.

Chapter 5 explores the implementation of the low-complexity turbo-equalization scheme introduced in chapter 4 on a low-cost and low-power consumption digital signal processing (DSP) device. The development platform and the considered transmission scenario are described. The DSP implementation of the turbo-equalizer is then discussed, with an emphasis on the data representation and computation constraints arising from the use of fixed-point arithmetic. We examine the storage requirements as well as the maximum bit rate achievable on the DSP and present some experimental results.

Finally in chapter 6, we summarize this work and give some concluding remarks as well as suggestions for future research.

Thesis contributions and publications

We summarize below the main contributions of this dissertation:

- Theoretical proof of the convergence of a MAP turbo-equalizer towards the performance of the underlying system over an ISI-free channel (chapter 3)
- A rigorous derivation of the time-varying MMSE equalizer initially proposed by M. Tüchler (chapter 4)
- The introduction of a novel low-complexity MMSE equalizer with *a priori* information (the MMSE IC-LE, chapter 4)
- Presentation of a low-complexity approximate method to compute the MMSE IC-LE equalizer coefficients and relying on the Fast Fourier Transform (chapter 4)
- Implementation of a low-complexity MMSE turbo-equalization scheme on a fixed-point DSP device (chapter 5)

In particular, we hope that our theoretical treatment of MMSE equalization with *a priori* information in chapter 4 will help understand the potential and advantages offered by this approach, and ultimately inspire the design of new classes of low-complexity equalizers with improved performance.

At the present date, parts of the material presented in this thesis have led to the writing of three technical reports [105, 106, 107], three conference publications [104, 108, 109] and the submission of one journal paper [103].

Chapter 2

Digital communications over frequency-selective channels

Digital transmissions over band-limited channels will typically encounter two major impairments for reliable communications, namely *additive noise* and *intersymbol interference* (ISI). In addition, a third impediment may also arise over radio links, where the received signal power fluctuates randomly with time and sometimes encounters deep fades, a perturbation called *signal fading*.

Additive noise is a phenomenon of all practical transmission systems, and is usually mitigated through the use of appropriate *channel coding* techniques. ISI constitutes one of the major obstacles to reliable high-speed data transmission techniques over high signal-to-noise ratio (SNR) channels with limited bandwidth. A common solution to combat ISI involves deploying *equalization* techniques at the receiver side. Finally, signal fading is usually overcome through the use of *diversity* techniques, where the receiver is supplied with several independent replicas of the transmitted message. Then, the probability that all signal components will fade simultaneously is reduced considerably.

This chapter aims at providing the reader with some relevant background material regarding the subjects of channel coding and equalization. Interestingly, it turns out that properly designed coding and equalization schemes provide a built-in degree of diversity, and thus are robust against signal fading.

The first section of this chapter is devoted to the combination of channel coding with modulation, in order to achieve significant coding gains over conventional uncoded transmission, but without compromising the spectral efficiency of the transmission system. Two practical coded modulation schemes are considered in this thesis: *trellis-coded modulations* (TCM) and *bit-interleaved coded modulation* (BICM). These solutions were selected because of their widespread use in current communication systems. The two coded modulation schemes are exposed, and a comparison of performance is pro-

vided on the basis of simple codes, by considering different typical transmission scenarios that may be encountered in practice.

In the second part, we focus on the problem of intersymbol interference, and review classical equalization techniques which precisely aim at mitigating ISI, namely: maximum-likelihood sequence detection, linear equalization, decision-feedback equalization, and interference cancellation. A theoretical comparison of these equalizers is provided, and illustrated through a simple case study.

2.1 Coded modulation for band-limited channels

Channel coding essentially involves introducing some form of controlled redundancy in the transmitted message, which may then be exploited by the receiving terminal to detect and/or recover some of the transmission errors introduced by the noisy channel. Introducing redundancy reduces the effective transmission rate. Generally, there exist two possibilities to compensate for the rate loss: either increasing the modulation rate, if the channel permits bandwidth expansion, or enlarging the signal set of the modulation system if the channel is bandwidth-constrained. The latter solution necessarily leads to the use of nonbinary modulation, and comes at the expense of reduced noise immunity, since at constant average power, signal points are then spaced more closely together with respect to uncoded transmission. If modulation is considered independently of channel coding, the use of powerful codes (with large block or constraint length) is required to offset the performance loss caused by the expansion of the signal set. With the seminal work of Ungerboeck [179, 181] and Imai [90] in the early eighties, it was quickly recognized that modulation and coding could be advantageously optimized as a single entity to obtain significant coding gains with simpler codes, thus pioneering the field of *coded modulation*.

To understand the potential improvement to be expected from a clever combination of coding and modulation, it is insightful to consider the fundamental limits promised by information theory. In particular, information theory which dates back to the landmark paper of Shannon in 1948 [155] provides an upper bound on the maximum information rate that can be achieved with arbitrarily low error probabilities over physical channels, through the notion of *channel capacity*. Following the analysis of Forney and Ungerboeck [65], there exist coding schemes that can achieve arbitrarily low error probabilities over an ideal additive white Gaussian noise (AWGN) channel, at signal-to-noise ratios approximately 9 dB lower than those required to achieve error rates of the order of $10^{-5} - 10^{-6}$ with conventional quadrature-amplitude modulation (QAM). In fact, this gap of 9 dB can be separated into an effective coding gap of 7.5 dB and an asymptotic *shaping* gap of about 1.5 dB. The latter follows from the use of a uniform rather than Gaussian distribution over the signal set, and may be recovered independently of the coding gain by proper signal shaping techniques [61]. Similar

conclusions hold for the fully-interleaved Rayleigh fading channel¹, where significantly higher coding gains, more than 30 dB, could theoretically be achieved [23, chap. 13].

In the last two decades, practical coded modulation schemes have been devised which allow us to recover most of the SNR gap from capacity, both for the AWGN and Rayleigh channels. We focus exclusively in this thesis on two particular instances of coded modulation, which combine a binary convolutional encoder with a memoryless symbol mapping function, namely trellis-coded modulation (TCM) and bit-interleaved coded modulation (BICM). The philosophy behind the two approaches is quite different. TCM were originally designed for the AWGN channel, and generally prove to be less suited for the fully-interleaved Rayleigh fading channel. In contrast, BICM was introduced as a promising coding scheme for fading channels, but has been shown to perform close to the limits promised by information theory on the AWGN channel as well, with the use of appropriate codes and labelling rules combined with iterative decoding at the receiver side.

In this section, we first introduce some preliminary definitions and notations to describe and characterize band-pass transmission systems, employing some form of convolutional encoding and operating over bandlimited channels. Then, the fundamental principles of TCM and BICM are briefly reviewed, and examples of practical codes are exposed and compared. We should mention that since the combination of equalization with decoding is our main concern in this thesis, we have purposely restricted our attention to simple TCM and BICM schemes with low complexity and moderate coding gains. Better performance could be achieved by considering more elaborate coded modulation schemes such as *Turbo TCM* [145] or BICM with Turbo-Codes [110, 111] for example.

2.1.1 Preliminary definitions and notations

Consider the block diagram shown in figure 2.1. An ideal information source delivers a sequence of independent and identically distributed random information bits $\{b_k^i\}$. The information sequence enters a rate $R_c = k_c/n_c$ binary convolutional encoder, which delivers the corresponding coded sequence $\{c_k^i\}$. More precisely, we assume that an input sequence of k_c successive information bits $(b_k^1, \dots, b_k^{k_c})$ is encoded into a sequence of n_c coded bits $(c_k^1, \dots, c_k^{n_c})$ at time index k . The whole coded sequence is eventually shuffled by a bit interleaver. We recall that an interleaver is a one-to-one permutation $\pi : \mathbb{Z} \mapsto \mathbb{Z}$ of the integers, that rearranges the ordering of a sequence of symbols in a deterministic manner. The interleaved coded sequence is denoted $\{c_n^i\}$. The coded bits enter a memoryless symbol mapper. The symbol mapping operation is a one-to-one mapping which assigns an M -ary symbol x to an input group of m successive bits, with $M = 2^m$. We assume for convenience that $m = n_c$, so that n_c successive coded bits $(c_n^1, \dots, c_n^{n_c})$ are mapped onto the symbol x_n at time n . The symbols x_n are generally complex, and chosen from some finite discrete alphabet $\mathcal{S}$, with cardinality $|\mathcal{S}| = M$. We

¹The fully-interleaved Rayleigh fading channel refers to a flat-fading channel where successive transmitted symbols face uncorrelated attenuations with Rayleigh-distributed squared magnitude.

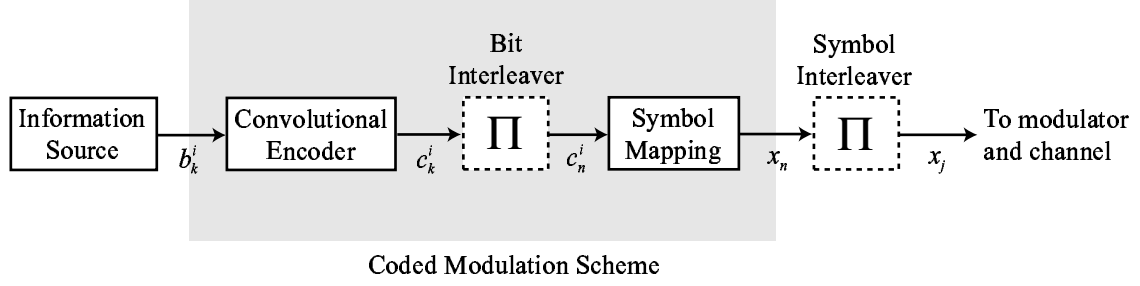


Figure 2.1: Block diagram of a generic pass-band transmitter in the presence of convolutional channel coding.

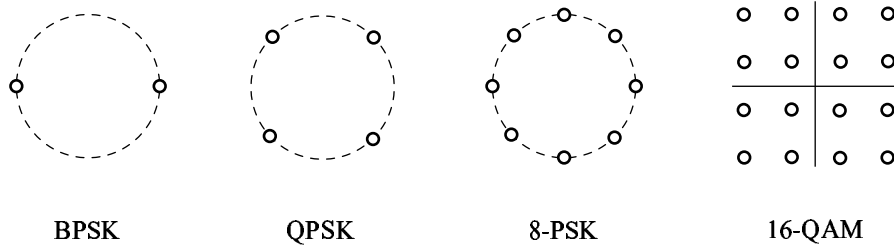


Figure 2.2: Signal sets for some common phase and amplitude/phase modulation.

shall denote by $\ell^i(s)$ the value of the i th bit of the label of $s \in \mathcal{S}$, with $i = 1, 2, \dots, m$. Accordingly, we shall denote by $\mathcal{S}_b^i$ the subset of all signals $s \in \mathcal{S}$ whose label $\ell^i(s)$ has value $b = \{0, 1\}$ in position i . Examples of common signal sets are shown in figure 2.2. A symbol interleaver may follow the signal mapping function.

As illustrated in figure 2.1, our definition of coded modulation encompasses the operations of convolutional encoding, symbol mapping, and possible bit-wise interleaving in between. We define the *code rate* R_{CM} of the coded modulation as the number of information bits conveyed per output symbol (or *channel use*), yielding

$$R_{CM} = R_c \log_2 M = mR_c \quad \text{information bits/symbol} \quad (2.1)$$

The discrete-time symbols $\{x_n\}$ are delivered to the signal modulator at the rate $R = 1/T$ symbol per second, where T is the symbol period.

In the rest of this thesis, we shall make the important assumption that the transmission system is constrained to use only a portion of the available spectrum resources, as a consequence of some physical or system design constraints. More precisely, we suppose that the transmission band is limited to some bandwidth W Hz. In view of this restriction, it is interesting to measure how efficiently a communication system makes use of the available spectrum resources. This is answered by considering the *spectral efficiency* η of the transmission, which basically tells us how many bits per second are

transmitted in a given bandwidth W , and is given by

$$\eta = \frac{R_c \log_2 M}{WT} = \frac{R_{CM}}{WT} \quad \text{information bits/s/Hz}$$

From Nyquist's theorem, we know that pass-band transmission of 1 signal every T seconds requires a minimum bandwidth $W = 1/T$ Hz in the absence of intersymbol interference (more about this in section 2.2). It follows that the maximum theoretical spectral efficiency that could be achieved by such a system is $\eta_{\max} = R_{CM}$.

2.1.2 Trellis-coded modulation

The concept of trellis-coded modulation was initially introduced by Ungerboeck in 1976 [181], and later developed in [179, 180]. TCM has undergone a rapid transition from theoretical investigations to practical applications, bringing in particular the field of wireline modems close to the capacity of the telephone channel. The subject of TCM is covered in depth in many textbooks (see e.g. [31, 91] or [152]), so that we shall only recall the basic principles in this section.

The conventional structure of a TCM encoder is shown in figure 2.3. It comprises a binary convolutional encoder followed by a special memoryless symbol mapper. Let us assume that we want to transmit m bits of information per encoder/modulator operation. Then, the signal constellation is enlarged from $M = 2^m$ to $M' = 2^{m+1}$ signals. $\tilde{m} \leq m$ information bits are expanded by the rate $R_c = \tilde{m}/(\tilde{m} + 1)$ binary convolutional encoder into $\tilde{m} + 1$ coded bits. These bits are used to select one of $2^{\tilde{m}+1}$ subsets of the M' -ary signal set. The remaining $m - \tilde{m}$ uncoded bits determine which of the $2^{m-\tilde{m}}$ signals in this subset is to be transmitted. The encoding operation thus directly occurs in signal space. The set of coded signal sequences may be conveniently described by a trellis diagram which has the same number of states as the considered convolutional encoder, and which presents the particularity that parallel transitions may occur, due to the presence of uncoded bits (when $\tilde{m} \neq m$).

Through the introduction of redundancy, coding aims at restricting the set of allowed signals at a given instant in time so as to maximize the difference between pairs of valid coded sequences. Considering transmission over the AWGN channel, this translates mathematically into the requirement that coded sequences should be located as far apart from each other as possible in the Euclidean signal space. TCM uses the fundamental concept of *mapping by set-partitioning*, which divides the signal constellation into smaller subsets with increasing Euclidean distance between signal points. A computer search is then performed to select the convolutional code, with a prescribed complexity, essentially yielding the maximum *free Euclidean distance* between pairs of coded sequences. Tables of good codes to be used with TCM can be found in [180]. Soft-decision decoding is performed at the receiver side, using a slightly modified version (to account for eventual parallel transitions) of the conventional Viterbi algorithm for decoding convolutional codes. TCM offer significant coding gains

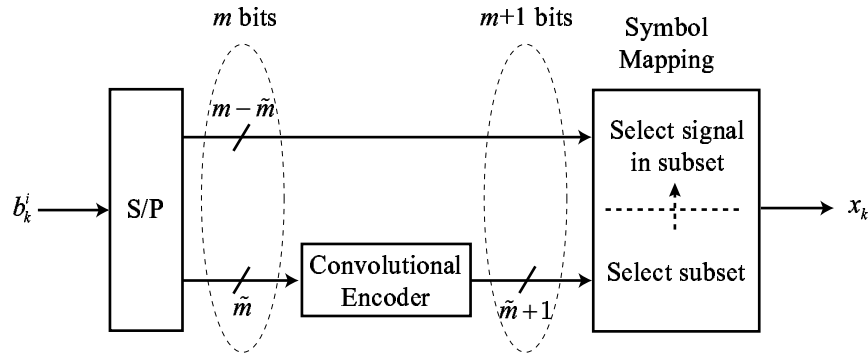


Figure 2.3: Block diagram of a TCM encoder.

over uncoded modulation at values of spectral efficiency above and including 2 bit/symbol. Roughly speaking, it is possible to gain 3 dB with 4 states, 4 dB with 8 states, nearly 5 dB with 16 states and up to 6 dB with 128 or more states.

TCM, however, suffers from two drawbacks. First, it proves necessary in many applications to adapt the spectral efficiency of the coding scheme as the channel conditions change over time, in order to maintain a given reliability for the transmission. Unfortunately, achieving optimum TCM code design in such a variable-rate environment requires implementing distinct encoders and decoders, with very different trellis structures. Capitalizing on some remarks by Clark and Cain [42, chap. 8], Viterbi *et al* [186] have proposed an elegant remedy to this problem, which was later refined by Wolf and Zehavi [194]. Called *pragmatic TCM*, their approach uses a fixed 64-state standard convolutional code with maximum free distance, combined with a different mapping rule than set partitioning. Rate adaptation is obtained by proper puncturing of the convolutional code. Although suboptimum in principle, this configuration yields performance very close to that obtained with the best Ungerboeck codes at similar encoder complexity, with the advantage of versatility.

But more importantly, TCM schemes optimized for the AWGN channel are generally not optimal for the fully-interleaved Rayleigh fading channel, a valid model for several important wireless communication systems. As suggested in figure 2.1, the use of TCM over fading channels requires symbol-wise interleaving at the output of the signal mapper, where the interleaver depth is typically larger than the maximum fade duration anticipated. This essentially results in a memoryless channel at the receiver side, where successive transmitted symbols undergo uncorrelated signal fadings. The suboptimality of Ungerboeck's TCM schemes stems from the fact that one of the most relevant parameters for code design over fully-interleaved fading channels turns out to be the minimum Hamming distance of the code², i.e. the number of symbols in which two valid coded sequences disagree, rather than the free Euclidean distance [160]. Classical TCM schemes generally have a low code diversity. Consequently, several approaches have been investigated in order to design efficient TCM schemes

² Sometimes called *diversity order* of the code in this context.

for the fading channel. A notable example is the use of *Multiple TCM* (MTCM), where more than one signal is assigned onto each transition in the trellis [31, chap. 9]. More recently, Al-Semari and Fuja have suggested using distinct encoders for the in-phase and quadrature components of the transmitted signal [10], an approach called *I-Q TCM*. Zehavi took a different look at this problem, and found that by employing bit-wise interleaving at the convolutional encoder output, and by using an appropriate soft-decision metric in the Viterbi decoder, the code diversity can be made equal to the smallest number of distinct bits (rather than symbols) between coded sequences [199]. Such a strategy, which essentially decouples coding and modulation, was later coined the name bit-interleaved coded modulation by Caire *et al* in [34].

2.1.3 Bit-interleaved coded modulation

The generic transmitter model for BICM is shown in figure 2.4, and usually consists of a serial concatenation of a binary convolutional encoder, a bit interleaver and a memoryless symbol mapper.

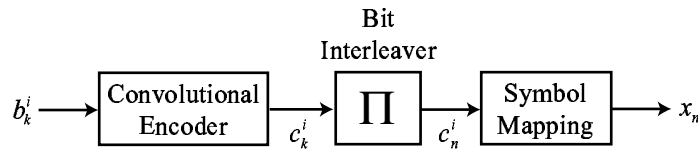


Figure 2.4: Block diagram of a BICM encoder.

The notable difference with a classical TCM encoder is the presence of the bit interleaver, which randomizes the mapping of binary coded sequences onto the signal set. Like pragmatic TCM, variable-rate transmissions are obtained by puncturing the output of the convolutional code. Two problems arise with such an approach.

1. The random mapping operation reduces the minimum Euclidean distance between pairs of coded sequences, by comparison with TCM where both functions are jointly optimized.
2. Exact maximum-likelihood (ML) decoding of BICM requires *joint* signal demapping and decoding at the receiver side, in order to account for the dependency between coded bits assigned to the same channel symbol which results from the signal labelling rule. However, this operation is rendered impractical by the bit interleaver.

Regarding these two points, Zehavi suggested the use of a signal set with *Gray labelling* instead of Set Partitioning, and proposed a two-step suboptimal decoding procedure, where simplified bit metrics are generated first by a soft-output demodulator, and then used in Viterbi decoding of the convolutional code after deinterleaving (see figure 2.5).

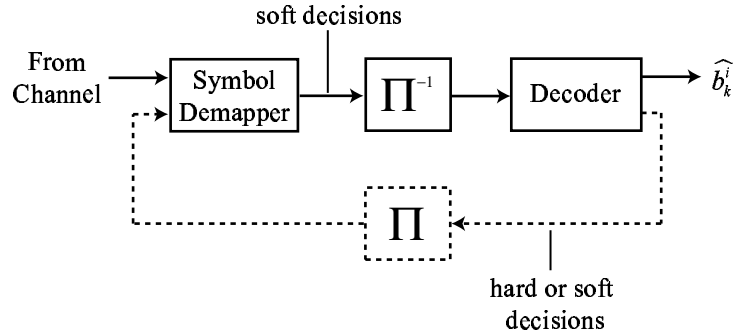


Figure 2.5: General form of a BICM decoder. The elements shown as dashed lines figure the feedback link used in iterative decoding.

A thorough analysis of the performance of BICM was later carried out by Caire *et al* [34]. They obtained tight upper bounds on the bit error probability for BICM over the AWGN and uncorrelated Rayleigh fading channels in the presence of ideal (infinite) interleaving, and suggested the following design guidelines: a) given a signal set, pick up a standard convolutional code with maximum free distance and code rate compatible with the desired spectral efficiency; b) select the signal labelling map according to some practical optimality criterion, which depends on the channel model and the decoding strategy, and essentially amounts to translating a large Hamming distance into a large Euclidean distance. In particular, Gray mapping was conjectured to be the best choice for the “one-shot” decoding procedure proposed by Zehavi.

In fact, the choice of the mapping rule has turned out to be of central importance for BICM design with the introduction of *iterative decoding* strategies, an approach called BICM-ID (Iterative Decoding). Iterative decoding was initially proposed by Li and Ritcey, who recognized the suboptimality of Zehavi’s approach and suggested the use of hard-decision feedback from the decoder [116, 117]. Their approach was later refined in [39, 115] and [121] to include soft-decision feedback, in accordance with the so-called *Turbo principle* [80]. A similar receiver was independently proposed in [165]. For convenience, the description of iterative decoding is deferred to the next chapter. It suffices to mention for the moment that the symbol demapper and the convolutional decoder exchange soft information about the transmitted message in an iterative manner (see figure 2.5). The key point of this approach is that iterative decoding resolves to some extent the problem of low minimum Euclidean distance encountered with conventional BICM schemes employing Gray mapping. In fact, assuming error-free feedback, it is possible to design labelling maps yielding higher free Euclidean distance than standard TCM schemes. However, as pointed out in [165] and [162], there is a trade-off to find between good asymptotic performance of the BICM-ID scheme, and sensitivity of the mapping rule to feedback errors, which is measured by the minimum SNR (also called *convergence threshold*) upon which successive iterations begin to improve the bit-error rate. A similar issue arises when considering iterative equalization and decoding, as we shall see in the next chapter. Hence, while Gray labelling is optimal for conventional (one-shot) BICM decoding, better labelling maps may be found for iterative

decoding, depending on the considered channel model and the desired performance. Design criterions have been proposed in the aforementioned papers as well as in [77] for optimizing the signal mapping rule. Significant improvements over both TCM and conventional BICM schemes have been obtained, but the subject still lacks some maturity at the present time.

2.1.4 Examples of practical coded modulation schemes

We introduce in this section three examples of coded modulation schemes, that achieve a maximal spectral efficiency of 2 bits/s/Hz with similar encoder complexity.

The first one is a TCM that combines an 8-PSK signal set with an 8-state rate $R_c = 2/3$ convolutional encoder. A possible realization of the TCM encoder is shown in figure 2.6. This code is known to achieve a theoretical asymptotic coding gain of 3.6 dB with respect to uncoded QPSK [179].

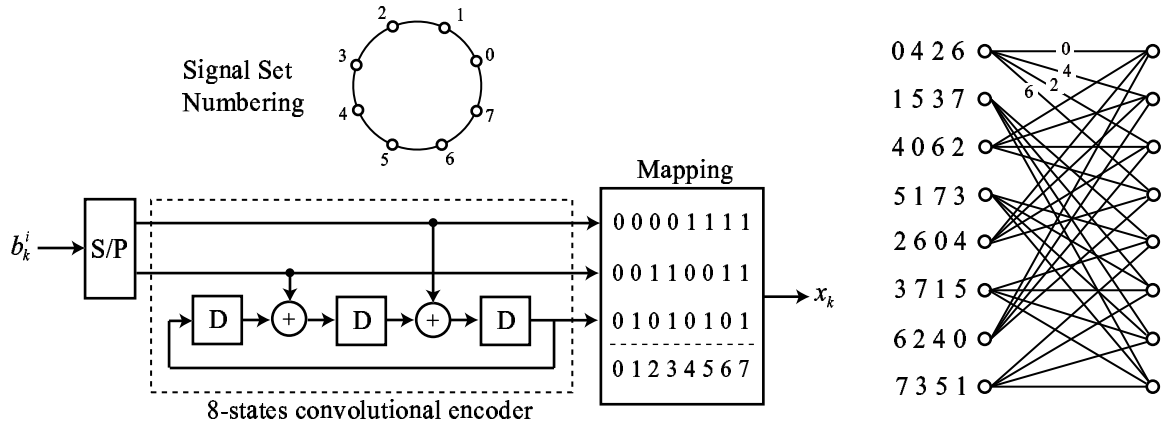


Figure 2.6: A possible realization of the encoder for the 8-state 8-PSK TCM, with the corresponding trellis representation.

The second scheme is a BICM that employs an 8-PSK constellation and an 8-state rate $R_c = 2/3$ convolutional code with maximum free distance $d_{\text{free}} = 4$. The transfer function matrix of this code was obtained from [92]

$$G(D) = \begin{pmatrix} 1+D & D & 1 \\ D^2 & 1 & 1+D+D^2 \end{pmatrix}$$

Finally, the third code is another BICM that combines a 16-QAM signal set with an 8-state rate $R_c = 1/2$ convolutional encoder with maximum free distance $d_{\text{free}} = 6$ and transfer function matrix

$$G(D) = \begin{pmatrix} 1+D+D^3 & 1+D+D^2+D^3 \end{pmatrix}$$

A set partitioning labelling map (see figure 2.7) and a random interleaver of size 8196 coded bits were considered for the two BICM schemes.

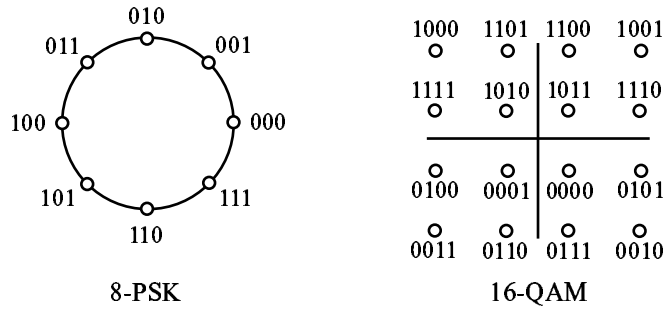


Figure 2.7: Set-partitioning labelling maps for 8-PSK and 16-QAM.

In order to illustrate the previous discussions, the three coded modulation schemes have been simulated by considering different typical transmission scenarios. We caution the reader that the purpose of these case-studies is certainly not to advocate which coded modulation scheme should be retained for a particular communication system, a problem for which no general solution exists, but simply to highlight the respective strengths and weaknesses of BICM-ID and TCM through simple examples.

Performance over the AWGN channel

Figure 2.8 presents the bit-error rate (BER) obtained by simulation of the three coded modulation schemes over an AWGN channel. Iterative decoding was used for the BICM. Accordingly, we present both the theoretical asymptotic performance achieved by a *genie* receiver which has perfect prior knowledge about the transmitted bits (ideal feedback from the decoder), and the practical performance obtained after 10 iterations. We first note that the TCM scheme provides an effective coding gain of 2.7 dB over conventional uncoded QPSK³ at an error probability of 10^{-5} , thereby showing the benefits of a clever combination of coding and modulation. Interestingly, the simulation results also show that BICM with iterative decoding is able to recover the loss in free Euclidean distance resulting from the separate optimization of coding and modulation. In fact, it even outperforms TCM for E_b/N_0 values in the range 3.5–7 dB, although the asymptotic slopes of the curves suggest that TCM remains the best choice at higher SNR. We emphasize however that labelling maps with improved asymptotic performance over set partitioning mapping exist for both 8-PSK and 16-QAM (see [39, 77, 115]). Finally, it is interesting to observe that the 16-QAM BICM performs better than its 8-PSK counterpart. This is a consequence of the combination of set-partitioning labelling with iterative decoding. In fact, the inverse phenomenon can be observed with a Gray labelling map (not shown here), where iterative decoding does not offer significant improvement over a conventional "one-shot" BICM receiver. But in the latter case, the BICM schemes only offer a marginal coding gain with respect to uncoded

³We recall that the theoretical coding gain of 3.6 dB is only valid asymptotically. In particular, depending on the TCM distance spectrum, very large SNR may sometimes be required to obtain such a coding gain.

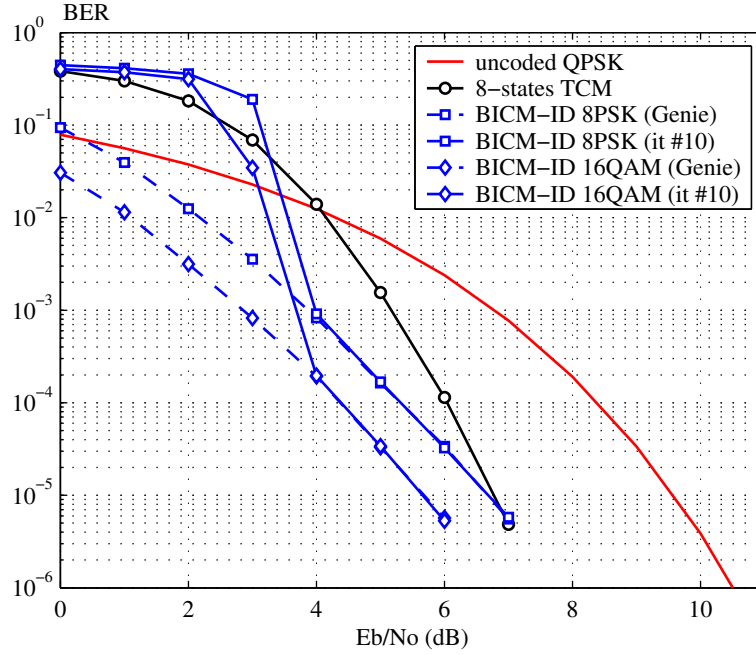


Figure 2.8: Performance of the considered coded modulations over the AWGN channel.

QPSK, and thus are significantly outperformed by the TCM at all SNR values.

Performance over the fully-interleaved Rayleigh flat-fading channel

Let us consider now transmission over a fully-interleaved Rayleigh flat-fading channel. Such a model assumes that the channel state varies independently between successive symbols. In practice, this assumption usually requires the presence of an interleaver whose depth exceeds the coherence time of the fading process, which is defined as the time beyond which successive samples can be considered as independent.

Simulation results for this channel model are shown in figure 2.9, where we have assumed ideal (infinite-depth) symbol interleaving, both in the TCM and BICM cases. We observe that thanks to the presence of the additional bit interleaver, the two BICM schemes benefit from an increased code diversity and thus offer significant coding gains over the TCM scheme, 5 and 7 dB respectively at a bit-error probability of 10^{-4} . Although this is not reported on the plots, similar simulations have been conducted for the two BICM schemes but with Gray labelling. We observed that the iterative process combined with set-partitioning offered respective gains of 2 dB for the 8-PSK BICM-ID and 4 dB for the 16-QAM BICM-ID with respect to the one-shot approach with Gray mapping over this channel.

At several occasions in this document, we shall be interested in evaluating the *diversity order* achieved by a transmission scheme. We recall that a system is said to achieve a diversity order L on

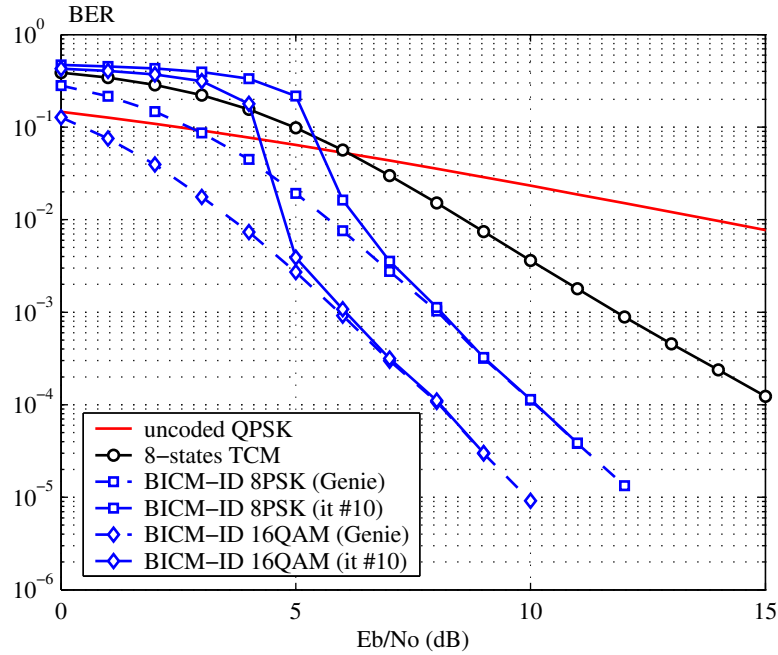


Figure 2.9: Performance of the considered coded modulations over the fully-interleaved Rayleigh flat-fading channel.

a fading channel, where L is an integer, if the probability of error decreases as the inverse of the L th power of the signal-to-noise ratio [23, chap. 13]. A simple method to measure the diversity order of a transmission scheme thus consists in superposing curves of the form $1/\text{SNR}^L$ to the plots of the system performance, and searching for the value L which matches the asymptotic slopes of the system performance curves. By applying such a technique with the simulation results of figure 2.9, with $\text{SNR} = R_{\text{CM}}E_b/N_0$, we were able to verify that the TCM scheme has an asymptotic diversity order $L = 2$ (the length of the smallest error event), whereas the 8-PSK and 16-QAM BICM schemes achieve respective asymptotic diversity orders $L = 4$ and $L = 6$ (the free distance of their respective inner convolutional encoders). This is consistent with the discussions in the previous sections.

Performance over a quasi-static Rayleigh flat-fading channel

In several practical applications, such as real-time speech transmission, a strict decoding delay is imposed at the receiver side. This precludes the use of large interleavers. In this case, depending on the channel coherence time, a transmitted data sequence may experience only a few different fading values, which makes the assumption of a memoryless channel, normally achieved by long-enough interleaving, no longer valid. We shall consider here a worst-case scenario where a whole data block is affected by the same random attenuation, which varies independently between successive blocks. Such a model is commonly called *quasi-static channel*. Since deep fades may cause the loss of entire

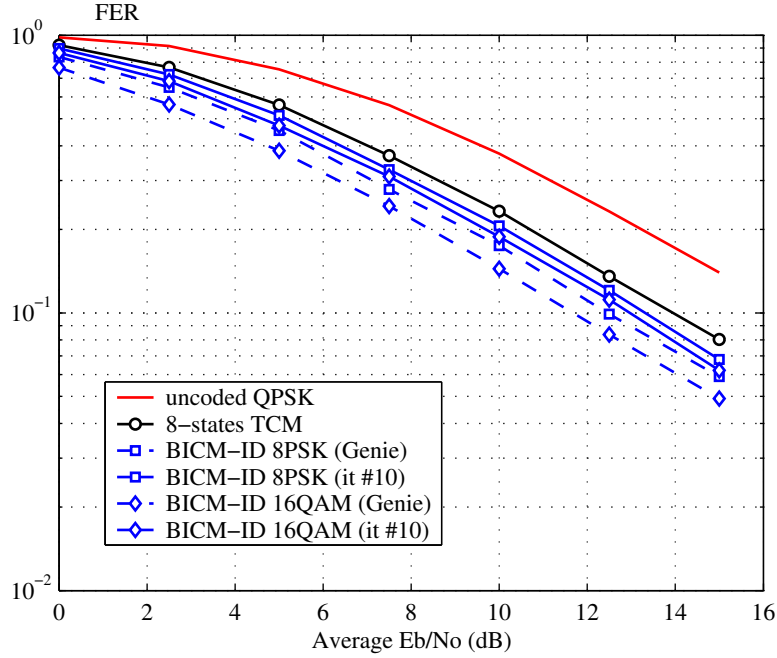


Figure 2.10: Performance of the considered coded modulations over a quasi-static Rayleigh flat-fading channel.

blocks, irrespective of the error-correction capabilities of the channel code, the relevant parameter for evaluating the system performance is no longer the bit-error rate, but rather the frame-error rate (FER) which measures the average number of received data blocks that have been decoded without errors.

In order to match the quasi-static model assumption, we have considered data blocks with a fixed size of 256 modulated symbols for each of the three coded modulation schemes, which seems a reasonable choice. Bit-interleavers with size 768 and 1024 coded bits were used for the 8-PSK and 16-QAM BICM schemes respectively.

Figure 2.10 presents the FER performance obtained with the three coded modulation schemes over a quasi-static Rayleigh flat-fading channel. On inspection of these curves, we first observe that the different schemes do not offer any diversity gain with respect to the uncoded transmission, since the asymptotic slopes are identical in all cases. The error-correction capabilities of the coded modulation schemes then simply translate into a horizontal shift of the performance curve towards lower signal-to-noise ratio at a given error probability. This comes from the fact that the channel can be considered as Gaussian, given the knowledge of the fading value affecting a block. Hence, codes optimized for the AWGN channel are likely to be good also for the quasi-static fading channel, and Euclidean minimum distance becomes the relevant parameter again. In particular, we note that in contrast with the previous transmission scenarios, BICM with iterative decoding is unable to reach the performance of the genie receiver. Hence, the BICM schemes only provide a marginal improvement with respect to the TCM. This is a consequence of the appearance of deep fades, which preclude the convergence

of the iterative process and thus penalize the frame-error rate.

It is clear that additional diversity techniques are required to achieve reliable communications on such severe channel models. In fact, an effective way of achieving a form of code diversity involves splitting a coded sequence into several blocks at the transmitted side, each block suffering from different channel attenuation. This transforms the quasi-static channel model into a *block-fading* channel model. This is precisely the solution adopted for the GSM telephony standard, where the blocks modulate 4 or 8 carriers whose spacing is larger than the coherence bandwidth of the channel (a diversity method called *frequency-hopping*), resulting in virtually uncorrelated blocks with constant channel coefficients at reasonable mobile speeds. Then, the sensible parameter governing the code design is no longer the Hamming distance between coded sequences, but rather the *block Hamming distance*, i.e. the number of blocks in which two coded sequences differ. We shall not delve into this issue any further in this work, and the interested reader is referred to [95, 96] for relevant discussions about this topic.

Discussion

The main purpose of these case-studies was to emphasize the fact that the choice of a coded modulation scheme is usually dictated by several considerations, including the channel model, the desired performance, and the complexity which is affordable at the transmitter and receiver side. We have seen in particular that with judicious code design, significant diversity gains can be obtained on fully-interleaved Rayleigh channels. In contrast, in more severe conditions, as arise for example with the class of quasi-static channels, additional diversity techniques may be required in order to obtain reliable transmissions at reasonable SNR values.

Having described solutions to realize reliable bandwidth-efficient transmissions in the presence of additive noise and possible signal fading, we now turn our attention to the important problem of combatting intersymbol interference arising in frequency-selective channels, by proper equalization techniques.

2.2 Equalization techniques for frequency-selective channels

All real channels exhibit some form of time dispersion. In radio links, this dispersion may be due to multi-path propagation, while in a telephone channel, it results from the imperfect transfer characteristics of the transmission system. In data transmission, this distortion causes successive transmitted symbols to be smeared in time and thus to overlap to the point that they may be no longer distinguishable as distinct pulses at the receiving terminal, a phenomenon known as *intersymbol interference*.

ence (ISI). From a frequency-domain perspective, the channel transfer function exhibits frequency-dependent attenuations and delays over the transmission bandwidth, hence the alternative designation of *frequency-selective* channels. The impairments caused by ISI become greater as successive symbols are spaced more closely together in time to increase the data rate, and as the bandwidth restrictions are more stringent (then requiring transmit pulses with larger temporal support, by virtue of the time-frequency duality). In fact, the effects of ISI may be so large on some pathological channels as to preclude reliable communication on the data system, even in the absence of noise. As an illustration, it is not uncommon today to encounter highly dispersive channels exhibiting delay spreads spanning up to a hundred symbol periods, as arise for example in the context of underwater communications [159] or broadband wireless access [15].

When the channel characteristics are known beforehand with a high degree of accuracy, it is theoretically possible to carefully design the transmit and receiver filters so as to eliminate ISI at the sampling instants, if the transmission rate R (in symbols per second) is less than the system bandwidth W (Hz) for a band-pass transmission. This condition is known as the *Nyquist criterion for zero ISI* [138, chap. 9]. In practice however, the frequency response of the channel is usually not known with sufficient precision, and appropriate methods must be deployed to compensate the ISI and achieve reliable communications.

We classically distinguish between three strategies to combat ISI. The first approach employs *spread-spectrum* signalling, where the transmission bandwidth W is much larger than the information rate R [138, 177]. Another solution involves using multicarrier transmission techniques, such as *Discrete Multi-Tone* (DMT) or *Orthogonal Frequency-Division Multiplexing* (OFDM) [18, 32]. This thesis focusses exclusively on a third approach which combines single-carrier transmission with *equalization* techniques at the receiver side, a solution dating back, in fact, to the early ages of digital communication [119].

After introducing the discrete-time model for data transmission over a frequency-selective channel, this section provides an overview and comparison of the classical equalization strategies that may be encountered in practical systems. Complementary material may be found e.g. in [139, 187] or [61, chap. 2], as well as in any textbook on digital communications.

2.2.1 A mathematical model for transmission over ISI channels

We shall consider in this work the following fundamental discrete-time baseband equivalent model for digital data transmission in the presence of intersymbol interference (see figure 2.11)

$$y_n = \sum_{\ell=0}^{L-1} h_{\ell} x_{n-\ell} + w_n \quad (2.2)$$

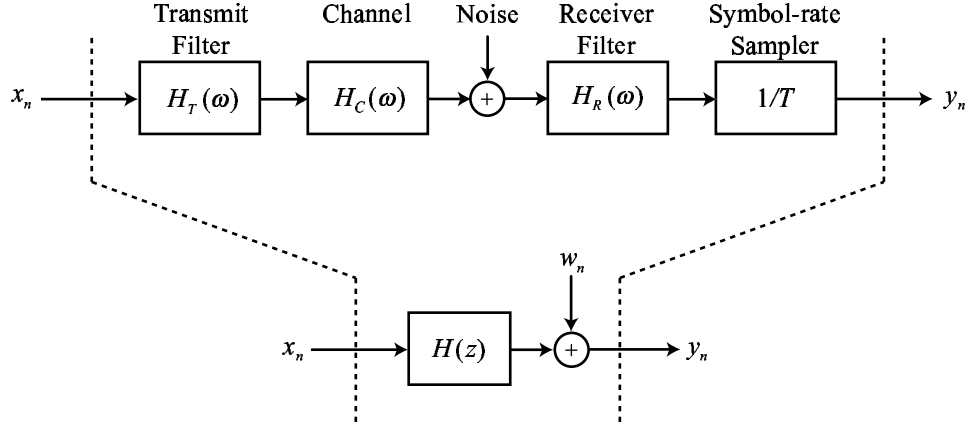


Figure 2.11: Discrete-time baseband equivalent channel model

Following the definitions and notations introduced in the previous section, the data symbols $\{x_n\}$ are modeled as a sequence of independent, identically distributed (i.i.d.) complex random variables x_n , with mean 0 and variance σ_x^2 , drawn from a finite alphabet $\mathcal{S}$, the signal constellation, with cardinality $M = |\mathcal{S}|$. While the assumption of uncorrelated data symbols is intuitively satisfying for BICM, it in fact also holds for many of Ungerboeck's TCM schemes (including the one considered in this thesis), as pointed out by Biglieri [30]. The data symbols are transmitted at the rate $R = 1/T$, where T is the symbol period (in seconds). The additive noise samples w_n are modeled as a zero-mean, wide-sense stationary, circularly-symmetric, white Gaussian complex random process with total variance σ_w^2 . The real and imaginary components of the noise are thus independent and identically distributed, with the same variance $\sigma_w^2/2$. Moreover, the noise is usually assumed to be uncorrelated with the transmitted data sequence $\{x_n\}$. The action of the ISI channel is modeled as a discrete-time finite-impulse response (FIR) filter with L complex coefficients $\{h_0, \dots, h_{L-1}\}$. As illustrated in figure 2.11, this representation includes the combined operations of transmit filtering, propagation over the continuous-time channel, receive filtering and symbol-spaced sampling, assuming perfect coherent demodulation and timing synchronization [112, 138].

The discrete-time equivalent channel model is conveniently described by its z -transform

$$H(z) = \sum_{\ell=0}^{L-1} h_{\ell} z^{-\ell} \quad (2.3)$$

The frequency response of the discrete-time channel is obtained by letting $z = e^{j\omega T}$ in the previous expression, yielding

$$H(\omega) = \sum_{\ell=0}^{L-1} h_{\ell} e^{-j\omega \ell T}, \quad |\omega| \leq \pi/T \quad (2.4)$$

Let us denote by $H_o(\omega)$ the frequency response of the underlying overall continuous-time baseband equivalent channel formed by the cascade of the transmit filter $H_T(\omega)$, channel $H_C(\omega)$ and receiver

filter $H_R(\omega)$

$$H_o(\omega) = H_T(\omega)H_C(\omega)H_R(\omega) \quad (2.5)$$

The frequency response of the discrete-time channel is related to the frequency response of the continuous-time channel by

$$H(\omega) = \frac{1}{T} \sum_m H_o\left(\omega - \frac{2m\pi}{T}\right) \quad , \quad |\omega| \leq +\pi/T \quad (2.6)$$

which shows that $H(\omega)$ is simply obtained as the *folded-spectrum* of $H_o(\omega)$ [139].

In order to complete the description of the transmission model, we shall finally introduce two additional quantities that will be useful in subsequent discussions, namely the *channel gain* and *channel signal-to-noise ratio (SNR) function*. The channel gain is written $\|h\|$ and defined by

$$\|h\| = \sqrt{R_{hh,0}} = \sqrt{\sum_{\ell=0}^{L-1} |h_\ell|^2} = \left(\frac{T}{2\pi} \int_{-\pi/T}^{+\pi/T} |H(\omega)|^2 d\omega \right)^{1/2} \quad (2.7)$$

where $R_{hh,0}$ denotes the sampled autocorrelation function of the discrete-time sequence $\{h_0, \dots, h_{L-1}\}$ evaluated at time $n = 0$. The channel SNR function $\text{SNR}_C(\omega)$, or *spectral SNR*, measures the signal-to-noise ratio at the output of the discrete-time equivalent channel, for a given angular frequency ω in the range $[-\pi/T, +\pi/T]$. It is defined as follows [61, 65]

$$\text{SNR}_C(\omega) = \frac{\sigma_x^2 |H(\omega)|^2}{\sigma_w^2} \quad (2.8)$$

This quantity naturally arises in the study of ISI channels, considering for example channel capacity calculation using standard water-pouring arguments [112, chap. 10], as well as in the evaluation of the theoretical performance of equalization schemes, as we shall see later in this chapter.

Let us now discuss the validity of the proposed discrete-time equivalent model for the transmission. Ideally, the receiver front end which comprises the receiver filter followed by the symbol-rate sampler should provide an information-lossless transition from continuous-time to T -spaced discrete-time. We know from classical estimation theory that the observations sampled at the symbol rate $1/T$ at the output of a receiver filter matched to the cascade of the transmit filter and channel $H_R(\omega) = H_T^*(\omega)H_C^*(\omega)$, provides a set of sufficient statistics for the estimation of the transmitted data sequence (see e.g. [63], [23, chap. 7] or [138, chap. 10] for a proof). However, the noise is usually correlated at the output of the matched filter, so that it is desirable to cascade the matched filter with an appropriate continuous-time noise whitening filter in order to simplify the analysis and receiver design. This particular receiver front end is called the *whitened matched filter* (WMF) [12, 63]. From the reversibility theorem⁴, it follows that the WMF, when it exists, also forms an information-lossless

⁴ The reversibility theorem essentially states that an invertible operation does not entail any loss of information since it

canonical receiver front end. Accordingly, the discrete-time model of equation 2.2 contains all the relevant information for the estimation of the transmitted sequence $\{x_n\}$, regardless of the criterion of optimality considered in the receiver design, when the receiver filter is chosen as the WMF.

Another interesting situation for which the model of equation 2.2 equivalently applies arises when the receiver filter is constrained to be a fixed square-root raised cosine filter [138, chap. 9]. Such a receiver filter is usually matched to the transmit pulse shape, and provides uncorrelated noise at the sampler input. It must be noted however that the receiver front end no longer delivers a set of sufficient statistics for the estimation of the transmitted sequence in this case, since it does not include a filter matched to the cascade $H_T^*(\omega)H_C^*(\omega)$ ⁵. Although sub-optimal, this solution is commonly adopted in practical transmission systems for several reasons. First, a fixed filter simplifies the receiver design, especially on time-varying channels where the synthesis of the optimum matched filter would require the use of continuous-time adaptive algorithms. Moreover, it has been shown in [73] that a fixed square-root Nyquist receiver filter with a carefully optimized roll-off factor may cause only a small degradation with respect to the optimum WMF. Finally, the square-root raised cosine filter acts as a low-pass filter. Consequently, sampling rates higher than the symbol rate $1/T$ may be employed so that no spectrum aliasing occurs, yielding a *fractionally spaced* receiver front end. The matched filter may then be moved to discrete-time without loss of optimality, which greatly simplifies the receiver design.

For the ease of exposition, we have thus limited ourselves in this work with the model given by equation 2.2, assuming uncorrelated data symbols and noise samples. We emphasize that unless the receiver filter is chosen as the WMF, an optimum receiver designed with respect to that discrete-time model will be sub-optimal with respect to the underlying continuous-time transmission model. The reader is referred to [112, chap. 10] for a thorough discussion about more general discrete-time models, which take into account colored noise and/or a colored data sequence, and the corresponding modifications that occur on the equalizers structure.

2.2.2 Matched filter bound

The matched filter bound (MFB) establishes a lower bound on the performance of any receiver in the presence of ISI. It is based on the idea that no receiver can have a lower error probability than the optimum receiver designed for the ISI-free transmission of a single data symbol x_0 , which simply consists of a discrete-time filter matched to $H(z)$ followed by a symbol-by-symbol decision device [112, 138]. The sample delivered by the matched filter is written $z_0 = R_{hh,0}x_0 + v_0$, with $R_{hh,0} = \|h\|^2$

can always be reversed [195].

⁵ Strictly speaking, this solution remains optimal as long as the impulse response $h_c(t)$ of the continuous-time channel consists of discrete pulses separated by multiples of the symbol period. For this particular case, the optimal matched filter may still be realized in T -spaced discrete-time without loss of information.

and where v_0 denotes the filtered noise sample at the output of the matched filter $H^*(z^{-*})$ at time 0. The effective signal variance is $\sigma_x^2 \|h\|^4$ and the effective noise variance is $\sigma_v^2 = \sigma_w^2 \|h\|^2$, so that the signal-to-noise ratio corresponding to the matched filter bound is

$$\text{SNR}_{\text{MFB}} = \frac{\sigma_x^2 \|h\|^2}{\sigma_w^2} = \frac{T}{2\pi} \int_{-\pi/T}^{+\pi/T} \text{SNR}_C(\omega) d\omega = \mathcal{A}(\text{SNR}_C(\omega)) \quad (2.9)$$

where $\mathcal{A}(\text{SNR}_C(\omega))$ denotes the *arithmetic mean* of the function channel SNR function $\text{SNR}_C(\omega)$. It is instructive to note that the matched filter collects the whole channel energy and maximizes the signal-to-noise ratio at the decision point. SNR_{MFB} thus places an upper bound on the SNR that can be achieved by any practical receiver at the input of the symbol-by-symbol detector. Since the noise is Gaussian at the output of the matched filter, the lower bound on the symbol error probability is simply obtained by applying standard formulas for uncoded modulations over an AWGN channel (see e.g. [138, chap. 5] or [23, chap. 5]), where the channel SNR is now given by SNR_{MFB} .

2.2.3 Maximum-likelihood sequence detection

Let us consider the transmission of a data sequence $\{x_n\}$ with finite length N , where N may be as large as desired. The corresponding discrete-time observation at the receiver side is denoted $\{y_n\}$ and will typically have a larger temporal support than the transmitted sequence, as a result of the time-dispersive propagation. The optimum equalization structure for a discrete data sequence in the presence of ISI generally is considered to be *maximum-likelihood sequence detection* (MLSD). The maximum-likelihood sequence detector determines the most probable transmitted sequence that resulted in a particular received signal. From a probabilistic point of view, MLSD selects the candidate sequence $\{\hat{x}_n\}$ that maximizes the likelihood $\Pr(\{y_n\}|\{\hat{x}_n\})$. When the transmitted sequences are all equally likely, a very common assumption, maximum-likelihood sequence estimation becomes equivalent to *maximum a-posteriori sequence detection*, where we search for the candidate sequence $\{\hat{x}_n\}$ that maximizes the *a posteriori* sequence probabilities $\Pr(\{\hat{x}_n\}|\{y_n\})$.

It must be emphasized that MLSD is not the optimum receiver from a digital transmission perspective, since sequences with a large number of errors are weighted the same as sequences with as small as one error. Consequently, MLSD does not minimize the probability of symbol error $P_S(\mathcal{E}) \triangleq \Pr(\hat{x}_n \neq x_n)$, which is obviously the most relevant criterion for the receiver design. Minimizing $P_S(\mathcal{E})$ in turn requires computing the set of *a posteriori* symbol probabilities $\Pr(\hat{x}_n|\{y_n\})$ for all admissible symbols at any time n , based upon the whole received observation, and selecting the symbols that maximize these quantities. This yields the *maximum a-posteriori symbol detector* [19, 85, 118]. In practice, comparable performance with the optimum symbol detector (at high SNR) as well as an elegant implementation and mathematical tractability in determining performance have made the MLSD an attractive receiver, in the absence of particular prior knowledge about the trans-

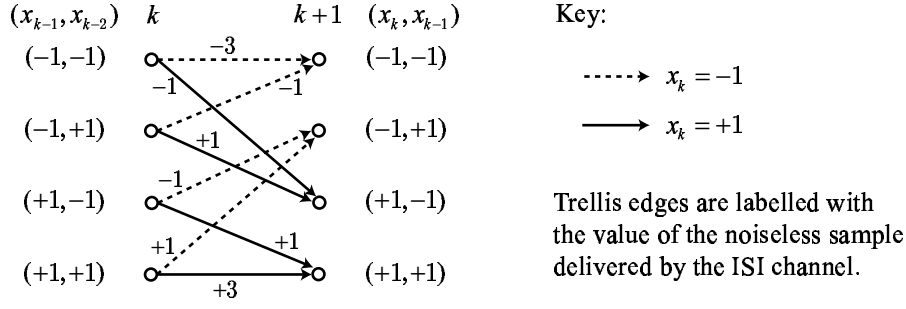


Figure 2.12: Example of trellis description for a BPSK transmission over the discrete-time 3-tap ISI channel $H(z) = 1 + z^{-1} + z^{-2}$.

mitted symbols. Otherwise, and as we shall see in chapters 3-4, there can be considerable benefit in employing equalizers that explicitly take into account *a priori* information about the data.

When the channel impulse response has finite length L , it can be modeled as a finite-state machine with M^{L-1} states, and thus admits an equivalent trellis representation (see figure 2.12). As shown by Forney [63], an M^{L-1} -state Viterbi algorithm may then be used to implement MLSD for such a system. The salient feature of the Viterbi algorithm is that optimum detection of an entire transmitted sequence is performed sequentially, with a computational cost growing linearly (and not exponentially as would happen with a “brute-force” approach) with the length N of the sequence. Forney’s receiver was originally derived for pulse-amplitude modulation (PAM) in the presence of white noise. Ungerboeck later extended this approach to work with complex signals, without the need for a noise whitening filter [178]. Both receivers have the same overall complexity. Tutorial expositions about the Viterbi algorithm and its application to MLSD may be found e.g. in [64, 84].

In addition to lending itself to a straightforward implementation, the Viterbi algorithm is relatively easy to analyze. Using the key concept of *error events*, Forney obtained upper and lower bounds for the symbol error rate, which are tight at moderate to high SNR [62, 63]. These bounds were further discussed and refined in [40, 66, 123, 130, 183], and show that the performance of MLSD depends principally on the minimum Euclidean distance between allowed noiseless transmitted sequences at the output of the ISI channel. In particular, for channels with small amounts of ISI, it is often the case that the minimum distance error event is a single symbol error. Consequently, the minimum distance between sequence pairs reduces to the minimum distance between any two uncoded modulation symbols, and MLSD achieves the effective SNR of the matched filter bound.

Although linear in the block length, the complexity of MLSD still remains exponential in the number M^{L-1} of states, and thus becomes rapidly prohibitive when the length L of the channel impulse response and/or the number M of points in the signal set increase. Thus, a significant amount of research has been devoted to alternative equalization methods that essentially retain the performance characteristics of MLSD but at reduced complexity. Early attempts essentially considered the use of

a prefilter to concentrate the energy in the first taps of the channels, so as to reduce the length L of the ISI span (see e.g. [60, 114] or [7]). Following the analogy with the problem of decoding long constraint-length convolutional codes, sequential detection was also proposed for long delay spread channels [197]. To date however, the most successful approach in retrospect has involved attacking directly the complexity of the Viterbi algorithm by reducing the number of searched paths in the trellis, employing truncation of the channel impulse response combined with state partitioning techniques and possible *per-survivor processing* [140]. An illuminating exposition of these algorithms is provided in [17], which encompasses as particular cases the *M algorithm* [13, 14], the *T algorithm* [157], *delayed decision-feedback sequence estimation* (DDFSE) [51], *reduced-state sequence estimation* (RSSE) [57, 94], or the *Generalized Viterbi Algorithm* (GVA) [83].

Rather than focussing on developing reduced-complexity variants of the optimum MLSD when the Viterbi algorithm becomes too complicated for implementation, one may as well consider simpler receiver structures, obtained by considering an alternative criterion to the direct minimization of the error probability, and by imposing particular constraints on the receiver design. This yields the general class of *filtering-based equalizers*. Although necessarily sub-optimal, such receivers often offer entirely adequate performance, and at a much lower cost than MLSD.

2.2.4 Filtering-based equalizers

Filtering-based equalizers employ linear filters to compensate for the channel distortions, so that the cascade of the channel with the equalizer ideally approaches a flat Nyquist folded spectrum (hence the name *equalizer*). Consequently, they turn the frequency-selective channel into an (almost) ISI-free equivalent channel. A memoryless symbol-by-symbol detector is then applied at the output of the equalizer (figure 2.13). As we shall see, such a transformation does not come without loss. Indeed, it usually results in noise enhancement, so that the signal-to-noise ratio at the slicer input remains lower than the ideal SNR of the matched filter bound.

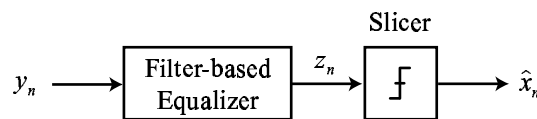


Figure 2.13: Generic block-diagram for filtering-based equalizer

Until recently, filtering-based equalizers were almost exclusively realized in *adaptive* form, operating in data-aided or decision-directed mode, and where the coefficients are adjusted iteratively from an error signal [139]. However, this approach may be unsuited to short packet transmission schemes due to the possible large number of training symbols (typically a few hundred) required for the adaptive equalizer to reach its steady-state. Therefore, we consider an alternative strategy in this thesis, where **the filter coefficients are computed from an estimate of the channel impulse re-**

sponse, which is turn is typically obtained using a known training sequence embedded in each packet. Moreover, in our simulation, we shall make the important assumptions of **ideal synchronization and channel estimation**.

Depending on the hypothesis considered in the equalizer design, we distinguish three different classes of filtering-based equalizers: linear equalization (LE), decision-feedback equalization (DFE), and interference cancellation (IC). Regardless of the particular structure of the equalizer, the filter coefficients may be optimized according to different criteria. Direct minimization of the symbol error probability leads to a set of nonlinear equations to solve, for which the solution may only be obtained by numerical techniques and with significant effort, thus precluding this approach for practical use [1, 154, 198]. Consequently, one must resort to more tractable but sub-optimal optimization criteria, such as the *zero-forcing* (ZF) or the *minimum mean-square error* (MMSE) criteria [112, 138]. ZF equalizers attempt to completely eliminate ISI at the slicer input, without regard for the noise. As a result, they essentially invert the channel frequency response, which may lead to severe noise enhancement for channels with spectral nulls or near-nulls. In contrast, the MMSE criterion allows residual ISI at the slicer input, and attempts to minimize conjointly the sum of ISI and noise. This reduces the variance of the noise and often procures a net advantage in SNR over ZF equalizers at the slicer input. Consequently, we shall focus only on MMSE-optimized equalizers in the following⁶.

The filter coefficients of MMSE-based equalizers are optimized to minimize the mean-square error

$$\varepsilon^2 \triangleq E \left(|z_n - x_n|^2 \right) \quad (2.10)$$

at the input of the symbol-by-symbol detector, where expectation is taken with respect to both the data and noise statistics. Unfortunately, performance analysis is complicated by the fact that minimization of the MSE does not necessarily translate into minimization of the symbol error probability since the noise is not strictly Gaussian at the slicer input. Consequently, an alternative figure of merit has to be defined in order to assess the performance of different MMSE equalization schemes. A common practice involves defining an equivalent signal-to-noise ratio at the equalizer output

$$\text{SNR}_E \triangleq \frac{\sigma_x^2}{\varepsilon^2} = \frac{\sigma_x^2}{E \left(|z_n - x_n|^2 \right)} \quad (2.11)$$

The intuitive justification for this choice is that we expect the equalizer with the highest output SNR to lead to the lowest error probability. This assumption holds rigorously true only for Gaussian additive distortion⁷, but offers insightful results in practice and leads to manageable mathematics. The situation is however further complicated by the fact that MMSE-optimized equalizers are inherently biased, i.e.

⁶ The corresponding ZF equalizers are easily deduced from the MMSE equalizers by letting $\sigma_w^2 \rightarrow 0$ in the expression of the filter coefficients.

⁷ In fact, such an assumption holds practically for low to moderate channel SNR, where the filtered noise usually dominates the residual ISI contribution at the equalizer output.

they deliver estimates of the form

$$z_n = g_0(x_n + \xi_n) \quad (2.12)$$

where g_0 is a real positive bias term, and ξ_n is an additive distortion term taking into account the residual ISI and the filtered noise at the equalizer output, that we suppose independent of x_n , with zero-mean and variance σ_ξ^2 . The presence of the bias has two consequences. First, it virtually increases the output SNR, so that SNR_E may exceed the matched filter bound SNR_{MFB} in some situations. Moreover, the decision rule are now biased and therefore sub-optimal with respect to the error probability. Consequently and as advocated in [41, 61], we will consistently use in the following the *unbiased SNR*, denoted $\text{SNR}_{E,U}$, as the relevant figure of merit to assess the performance of MMSE equalization schemes. The unbiased SNR is easily deduced from the knowledge of SNR_E by (see [41] for a proof of this general result)

$$\text{SNR}_{E,U} = \frac{\sigma_x^2}{E(|\xi_n|^2)} = \frac{\sigma_x^2 - \epsilon^2}{\epsilon^2} = \text{SNR}_E - 1 \quad (2.13)$$

and corresponds to the SNR obtained at the output of an unbiased MMSE equalizer, where the bias g_0 has been removed prior to the decision device (figure 2.14). In fact, removing the bias increases the MSE and reduces the SNR, but improves the error probability⁸.

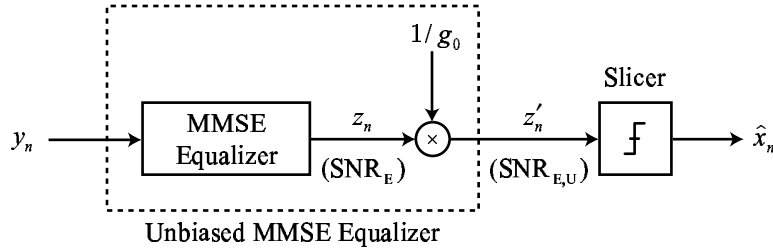


Figure 2.14: Block diagram of the unbiased MMSE equalizer

We now proceed to introduce the three classes of MMSE-optimized filtering-based equalizers. The equalizers are derived both in infinite-length form as well as under the constraint of a finite-length FIR realization, which is of great practical importance. In the latter case, we shall find it convenient to use the following model for the transmission, obtained from (2.2) by collecting N successive observations y_n in vector form

$$\mathbf{y}_n = \mathbf{H}\mathbf{x}_n + \mathbf{w}_n \quad (2.14)$$

⁸Obviously, another solution involves modifying the decision regions of the slicer to take the bias into account.

where we have defined

$$\mathbf{y}_n \triangleq [y_n, \dots, y_{n-N+1}]^T \quad (2.15)$$

$$\mathbf{w}_n \triangleq [w_n, \dots, w_{n-N+1}]^T \quad (2.16)$$

$$\mathbf{x}_n \triangleq [x_n, \dots, x_{n-L+1}]^T \quad (2.17)$$

and where $\mathbf{H}$ is the $N \times (N + L - 1)$ channel convolution matrix defined by

$$\mathbf{H} = \begin{pmatrix} h_0 & h_1 & \dots & h_{L-1} & 0 & \dots & 0 \\ 0 & h_0 & h_1 & \dots & h_{L-1} & & \vdots \\ \vdots & & \ddots & & & \ddots & \vdots \\ 0 & \dots & 0 & h_0 & h_1 & \dots & h_{L-1} \end{pmatrix} \quad (2.18)$$

where we have assumed that the channel coefficients are invariant at least over N successive symbol periods. Note that $\mathbf{H}$ is a Toeplitz matrix.

2.2.5 MMSE linear equalization

As shown in figure 2.15, the MMSE linear equalizer is simply obtained by placing a linear filter in the path of the received signal, followed by a memoryless decision device that selects the discrete symbol closest (in the minimum Euclidean distance sense) to the filter output.

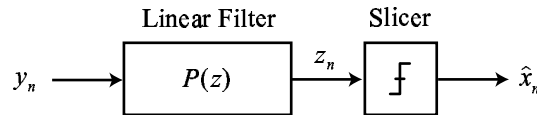


Figure 2.15: Block diagram of the MMSE linear equalizer

Infinite-length realization

The optimum transfer function $P(z)$ of the MMSE linear equalizer results from the minimization of the MSE $\varepsilon^2 = E(|z_n - x_n|^2)$. We obtain the following solution [112, 138]

$$P(z) = \frac{\sigma_x^2 H^*(z^{-*})}{\sigma_x^2 H(z) H^*(z^{-*}) + \sigma_w^2} \quad (2.19)$$

where the notation $H^*(z^{-*})$ is a shortcut for $H^*(1/z^*)$. We observe that the transfer function of the MMSE equalizer may be decomposed into a discrete-time matched filtering operation, followed by a purely recursive filter. Such a filter has infinite length and must be truncated in practice. The

corresponding minimum MSE is given by

$$\varepsilon_{\min, \text{MMSE-LE}}^2 = \frac{T}{2\pi} \int_{-\pi/T}^{+\pi/T} \frac{\sigma_x^2 \sigma_w^2}{\sigma_x^2 |H(\omega)|^2 + \sigma_w^2} d\omega \quad (2.20)$$

from which we deduce the biased SNR at the equalizer output

$$\text{SNR}_{\text{MMSE-LE}} = \frac{\sigma_x^2}{\varepsilon_{\min, \text{MMSE-LE}}^2} = \left(\frac{T}{2\pi} \int_{-\pi/T}^{+\pi/T} (\text{SNR}_C(\omega) + 1)^{-1} d\omega \right)^{-1} = \mathcal{H}(\text{SNR}_C(\omega) + 1) \quad (2.21)$$

We note from the last expression that the biased SNR is given by the *harmonic mean* $\mathcal{H}$ of the function $\text{SNR}_C(\omega) + 1$. Finally, the unbiased SNR for the MMSE linear equalizer is obtained as

$$\text{SNR}_{\text{MMSE-LE,U}} = \mathcal{H}(\text{SNR}_C(\omega) + 1) - 1 \quad (2.22)$$

Finite-length realization

We now derive the MMSE linear equalizer under the constraint that the filter $P(z)$ is FIR, with N coefficients. Let us express $P(z)$ in vector form: $\mathbf{p} = [p_0, \dots, p_{N-1}]^T$. Exploiting the notations of (2.14), the equalized sample at time n is then given by

$$z_n = \mathbf{p}^T \mathbf{y}_n \quad (2.23)$$

The filter coefficients are computed to minimize the MSE $\varepsilon^2 \triangleq E(|z_n - x_{n-\Delta}|^2)$, where $0 \leq \Delta \leq N + L - 2$ is a discrete delay introduced to approximate the anti-causality of the equalizer. This optimization problem yields the solution

$$\mathbf{p}^* = \mathbf{R}_{\mathbf{yy}}^{-1} \mathbf{R}_{\mathbf{yx}} \quad (2.24)$$

where we have introduced

$$\begin{aligned} \mathbf{R}_{\mathbf{yy}} &\triangleq E(\mathbf{y}_n \mathbf{y}_n^H) = \sigma_x^2 \mathbf{H} \mathbf{H}^H + \sigma_w^2 \mathbf{I} & (N \times N \text{ matrix}) \\ \mathbf{R}_{\mathbf{yx}} &\triangleq E(\mathbf{y}_n x_{n-\Delta}^*) = \sigma_x^2 \mathbf{h}_\Delta & (N \times 1 \text{ vector}) \end{aligned}$$

$\mathbf{h}_\Delta$ is defined as the product of the matrix $\mathbf{H}$ with the unit vector $\mathbf{e}_\Delta$ having a 1 in position Δ ⁹, and thus denotes the column Δ of $\mathbf{H}$. $\mathbf{I}$ is the identity matrix, and the superscript H denotes the hermitian transpose. It can be shown that in the presence of white noise ($\sigma_w^2 > 0$), the autocorrelation matrix $\mathbf{R}_{\mathbf{yy}}$ is strictly positive definite, and hence always invertible. Note also that $\mathbf{R}_{\mathbf{yy}}$ is Toeplitz. This structural property can be advantageously exploited to carry out the matrix inversion using computationally

⁹ We assume hereafter in this dissertation that indices for rows and columns are numbered starting from 0 (and not from 1, as is usual in mathematics).

efficient methods such as the Levinson-Durbin algorithm [76]. The corresponding MSE is given by

$$\varepsilon_{\min, \text{MMSE-LE}}^2 = \sigma_x^2 - \mathbf{R}_{yx}^H \mathbf{p}^* = \sigma_x^2 - \mathbf{R}_{yx}^H \mathbf{R}_{yy}^{-1} \mathbf{R}_{yx} \quad (2.25)$$

The output biased and unbiased SNR are easily deduced from the last expression using (2.11) and (2.13). We note that the MSE and hence the output SNR are a function of the delay Δ . In fact, this parameter adds a degree of freedom in the equalizer design and may be optimized to improve the performance, especially for small equalizer length N . The optimum restitution delay is given in closed-form by [107, app. C]

$$\Delta_{\text{opt}} = \arg \min_{0 \leq i \leq N+L-2} \mathbf{J}_{i,i} \quad , \quad \text{with } \mathbf{J} \triangleq \mathbf{H}^H \mathbf{R}_{yy}^{-1} \mathbf{H} \quad (2.26)$$

$\mathbf{J}_{i,j}$ denotes the (i, j) element of matrix $\mathbf{J}$. In practice, choosing $\Delta \approx (N+L)/2$ will generally provide good results if N is high enough. We would finally like to point out that the equivalence between finite-length and infinite-length equalizers can be established for $N \rightarrow \infty$ using the particular properties of Toeplitz matrices [79, 139].

2.2.6 MMSE decision-feedback equalization

The MMSE linear equalizer described above may suffer from noise enhancement which penalizes its performance, particularly at high SNR. The MMSE decision-feedback equalizer (DFE) is a simple non-linear alternative structure which attempts to somewhat circumvent this phenomenon by employing *tentative decisions*. The basic underlying idea is that if the receiver is operating with a low error-rate, then almost all past decisions are correct, and thus may be used to cancel the postcursor ISI by subtracting the past detected symbols with appropriate weighting from the equalizer output. The structure of the decision-feedback equalizer is presented in figure 2.16. It comprises a feedforward filter $P(z)$ which conceptually shapes the discrete-time channel model into a causal transfer function, and a strictly causal feedback filter $Q(z)$ which reconstructs the part of ISI from the present estimate caused by previous symbols. The equalized signal is the sum of the forward and feedback paths, and enters a memoryless threshold detector.

Infinite-length realization

Assume that the decisions on the past symbols are correct. As usual, the optimum transfer functions $P(z)$ and $Q(z)$ are obtained by minimizing the quantity $\varepsilon^2 = \text{E}(|z_n - x_n|^2)$. The derivation requires the introduction of the discrete-time spectral factorization theorem, summarized below (see [134] for a proof of this theorem).

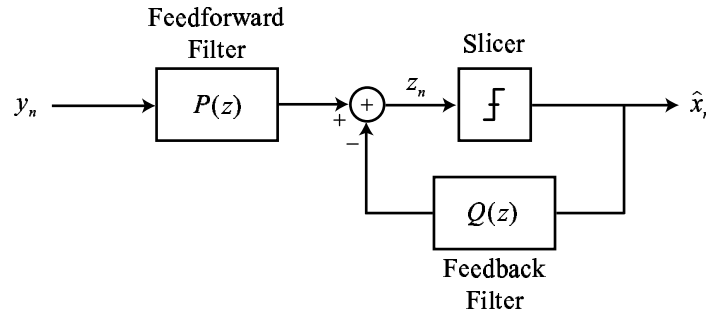


Figure 2.16: Block diagram of the MMSE decision-feedback equalizer

Theorem 2.1 (Discrete-time spectral factorization). *Let $S(z)$ be any discrete-time autocorrelation function with power spectrum $S(\omega)$ such that both $S(\omega)$ and $\log S(\omega)$ are integrable over the interval $-\pi \leq \omega \leq +\pi$ (Paley-Wiener conditions). Then there exists a unique, canonical (i.e. causal, monic¹⁰ and minimum-phase¹¹) discrete-time sequence with z-transform $F(z)$ and spectrum $F(\omega)$ such that*

$$S(z) = S_0 F(z) F^*(z^{-*}) \quad (2.27)$$

$$S(\omega) = S_0 |F(\omega)|^2 \quad (2.28)$$

where the real-valued constant $S_0 > 0$ is the geometric mean of the power spectrum $S(\omega)$

$$\log S_0 = \frac{1}{2\pi} \int_{-\pi}^{+\pi} \log S(\omega) d\omega \quad (2.29)$$

The logarithms may have any common base.

Consider now the z-transform $S_{yy}(z)$ of the autocorrelation of the received observations $\{y_n\}$, which is given by

$$S_{yy}(z) = \sigma_x^2 H(z) H^*(z^{-*}) + \sigma_w^2 \quad (2.30)$$

The previous expression is sometimes called the *key equation* [41]. $S_{yy}(z)$ is a polynomial in z with $2L - 1$ coefficients, whose frequency response satisfies $S_{yy}(\omega) > 0$ in the presence of noise. Hence, following theorem 2.1, $S_{yy}(z)$ may be factored into the product of a monic minimum-phase filter $F(z)$ with a monic maximum-phase filter $F^*(z^{-*})$

$$S_{yy}(z) = S_0 F(z) F^*(z^{-*}) \quad , \quad \text{with } S_0 > 0 \quad (2.31)$$

This factorization is unique and depends on the channel model and signal-to-noise ratio at the channel output. The MMSE-optimum infinite-length transfer functions for the feedforward and feedback

¹⁰ $f_0 = 1$.

¹¹ $F(z)$ has all its poles inside the unit circle, and all its zeros inside or on the unit circle.

filters are then given by (see [41, 112])

$$P(z) = \frac{\sigma_x^2 H^*(z^{-*})}{S_0 F^*(z^{-*})} \quad (2.32)$$

$$Q(z) = F(z) - 1 \quad (2.33)$$

Note that the feedforward filter $P(z)$ is anticausal, while the feedback filter is strictly causal. The corresponding figure of merits at the MMSE-DFE output are

$$\varepsilon_{\min, \text{MMSE-DFE}}^2 = \frac{\sigma_x^2 \sigma_w^2}{S_0} = \sigma_x^2 \exp - \left(\frac{T}{2\pi} \int_{-\pi/T}^{+\pi/T} \log(\text{SNR}_C(\omega) + 1) d\omega \right) \quad (2.34)$$

$$\text{SNR}_{\text{MMSE-DFE}} = \exp \left(\frac{T}{2\pi} \int_{-\pi/T}^{+\pi/T} \log(\text{SNR}_C(\omega) + 1) d\omega \right) = \mathcal{G}(\text{SNR}_C(\omega) + 1) \quad (2.35)$$

$$\text{SNR}_{\text{MMSE-DFE,U}} = \mathcal{G}(\text{SNR}_C(\omega) + 1) - 1 \quad (2.36)$$

where $\mathcal{G}(f)$ denotes the *geometric mean* of the function f .

Finite-length realization

Suppose now that the feedforward and feedback filters are finite-length FIR filters with N_p and N_q coefficients respectively. In vector form, we have $\mathbf{p} = [p_0, \dots, p_{N_p-1}]^T$ and $\mathbf{q} = [q_0, \dots, q_{N_q-1}]^T$. The feedback filter is strictly causal, with $0 \leq N_q \leq L - 1$. The optimum filters are obtained by minimizing the mean-square error $\varepsilon^2 = E(|y_n - x_{n-\Delta}|^2)$ where $0 \leq \Delta \leq N_p + L - 2$ is the parameterizable restitution delay of the feedforward filter. Our derivation follows the one by Al-Dhahir and Cioffi [5], which highlights the analogy with the infinite-length approach. Assuming correct past decisions, the equalized sample z_n at the input of the decision device at time n may be expressed as

$$z_n = \mathbf{p}^T \mathbf{y}_n - \mathbf{q}^T \mathbf{x}_{n-\Delta-1} \quad (2.37)$$

where $\mathbf{y}_n$ is the $N_p \times 1$ vector defined by $[y_n, \dots, y_{n-N_p+1}]^T$, and $\mathbf{x}_{n-\Delta-1}$ is the $N_q \times 1$ vector obtained by collecting the last N_q symbol estimates $[x_{n-\Delta-1}, \dots, x_{n-\Delta-N_q}]^T$ after a delay $\Delta + 1$. It proves to be mathematically convenient to introduce the augmented feedback vector $\tilde{\mathbf{q}} \triangleq [\mathbf{0}_{1 \times \Delta} \mathbf{1} \mathbf{q}^T \mathbf{0}_{1 \times s}]^T$, where s satisfies $N_p + L - 1 = \Delta + 1 + N_q + s$. We define the following Cholesky factorization [76], which is the finite-length equivalent of spectral factorization

$$\sigma_x^2 \mathbf{H}^H \mathbf{H} + \sigma_w^2 \mathbf{I} = \mathbf{L} \mathbf{D} \mathbf{L}^H \quad (2.38)$$

where $\mathbf{L}$ is an $(N_p + L - 1) \times (N_p + L - 1)$ lower triangular matrix, with 1's on the main diagonal, and where $\mathbf{D}$ is an $(N_p + L - 1) \times (N_p + L - 1)$ diagonal matrix, with positive real entries,

$\mathbf{D} \triangleq \text{diag}\{d_0, \dots, d_{N_p+L-2}\}$. The MMSE-optimum filters are then given by

$$\tilde{\mathbf{q}} = \mathbf{L}\mathbf{e}_\Delta \quad (2.39)$$

$$\mathbf{p}^* = \frac{\sigma_x^2 \mathbf{H}(\mathbf{L}^H)^{-1} \mathbf{e}_\Delta}{d_\Delta} \quad (2.40)$$

$\mathbf{H}$ is the $N_p \times (N_p + L - 1)$ Toeplitz channel convolution matrix, with first row $[h_0 \dots h_{L-1} \ \mathbf{0}_{1 \times (N_p-1)}]$. The minimum MSE at the equalizer output is

$$\varepsilon_{\min, \text{MMSE-DFE}}^2 = \frac{\sigma_x^2 \sigma_w^2}{d_\Delta} \quad (2.41)$$

from which we deduce the unbiased SNR, which admits a simple closed form

$$\text{SNR}_{\text{MMSE-DFE,U}} = \frac{d_\Delta}{\sigma_w^2} - 1 \quad (2.42)$$

Like for the finite-length MMSE LE, the MSE and hence the SNR depend on the delay Δ . Optimization of this parameter is especially important for short-length DFE. It has been shown in [6] that the optimum delay is given by

$$\Delta_{\text{opt}} = \arg \min_{0 \leq i \leq N_p+L-2} \{d_i\} \quad (2.43)$$

In practice, Δ may be safely set to $N_p - 1$ for most practical channels and noise scenarios without compromising performance, when N_p is of sufficient length. In this case, $\mathbf{p}$ is strictly anticausal. Obtaining the equalizer's coefficients essentially involves computing the Cholesky factorization (2.38). Efficient methods for realizing this task have been proposed in [4, 8, 9], both for the symbol-spaced as well as the fractionally-spaced cases.

Discussion

Worth mentioning is the fact that an alternative realization exists for the MMSE DFE, that was initially introduced by Belfiore and Park [22]. This structure, commonly called *noise-predictive DFE* (NP DFE), consists of an MMSE linear equalizer as the feedforward filter, and a linear noise predictor as the feedback filter, as shown in the configuration of figure 2.17. This structure is equivalent to the classical MMSE DFE under the condition that the filters have infinite length, and become sub-optimal otherwise. In spite of this fact, the NP DFE has been successfully used with trellis-coded signals, for which classical DFE do not apply directly because of system delay restrictions [56, 201].

The previous analysis has so far assumed correct past decisions for the sake of mathematical tractability, and thus leads to optimistic bounds regarding the performance of the MMSE DFE. In practice however, occasional incorrect decisions made by the slicer may propagate down the feedback

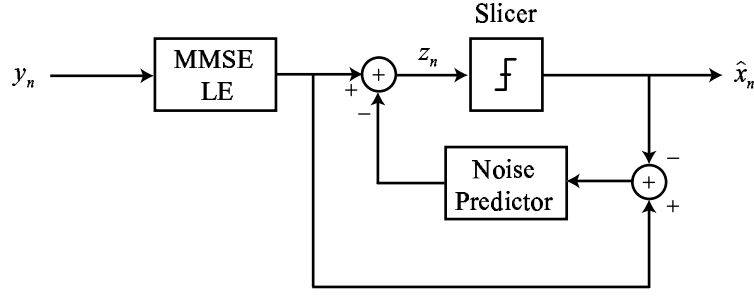


Figure 2.17: Block diagram of the noise-predictive interpretation of the MMSE DFE

section and alter the optimality of the equalizer's design. Analytical results on the probability of error for DFE in the presence of error propagation are available in the literature (see e.g. [11, 52, 158] and the references therein). To date however, the most efficient way to assess the influence of feedback errors has often been via measurements or Monte-Carlo simulations. On most practical channels, catastrophic error propagation rarely materializes, so that the MMSE DFE will generally (and often significantly) outperform the MMSE LE, except at very low SNR.

When the channel impulse response is known in advance, or in the presence of a reliable feedback link between the transmitter and receiver, efficient *signal precoding* techniques can be employed to mitigate error propagation as well as noise enhancement. Precoding is a transmitter equalization technique which essentially moves the feedback section of the DFE to the transmitter side. Modulo arithmetic is then required to limit the power enhancement that results. A comprehensive treatment of signal precoding techniques is available in the recent book by Fischer [61].

2.2.7 MMSE interference cancellation

The concept of decision feedback of past data symbols to cancel ISI can theoretically be extended to include future data symbols as well. If all past and future data symbols were assumed to be known exactly at the receiver side, then given a perfect model of the ISI process, all ISI could be cancelled exactly and without the noise enhancement phenomenon which affects linear as well as decision-feedback equalization. The resulting equalizer is called the MMSE interference canceller (IC). Initially suggested by Proakis [137], this equalization technique was later refined by Mueller and Salz [131] and Gersho and Lim [72]. The structure of the MMSE interference canceller is shown in figure 2.18. It includes a feedforward filter $P(z)$ in combination with a feedback filter $Q(z)$, which is in charge of reconstructing the precursor and postcursor ISI affecting the current sample, from an estimate $\{\bar{x}_n\}$ of the transmitted sequence. Unlike the DFE, the feedback filter may be noncausal. In the following analysis and like for the DFE, we assume perfect data estimates, i.e. $\bar{x}_n = x_n$.

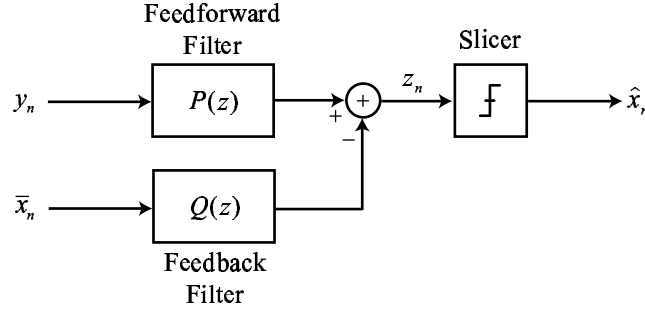


Figure 2.18: Block diagram of the MMSE interference canceller

Infinite-length realization

The optimum infinite-length transfer functions $P(z)$ and $Q(z)$ are obtained by minimizing the quantity $E(|z_n - x_n|^2)$ at the slicer input, under the additional constraint that $q_0 = 0$, q_0 being the reference tap of the feedback filter. This hypothesis is required to avoid the trivial solution $p_n = 0$ and $q_n = \delta_n$ which arises with the assumption of perfect decisions. With this constraint, the feedback filter is now restricted to the removal of ISI. Solving the optimization problem yields [101]

$$P(z) = \frac{\sigma_x^2}{\sigma_x^2 \|h\|^2 + \sigma_w^2} H^*(z^{-*}) \quad (2.44)$$

$$Q(z) = H(z)P(z) - g_0, \quad \text{with } g_0 = \frac{\sigma_x^2 \|h\|^2}{\sigma_x^2 \|h\|^2 + \sigma_w^2} \quad (2.45)$$

It is interesting to note that the optimum feedforward filter is a simple matched filter, up to a constant factor. In fact, the particular form of the optimum filters immediately suggests an alternative realization of the MMSE IC, shown in figure 2.19. The latter is equivalent to the structure of figure 2.18, but will require fewer coefficients and thus may be more suited to a practical implementation.

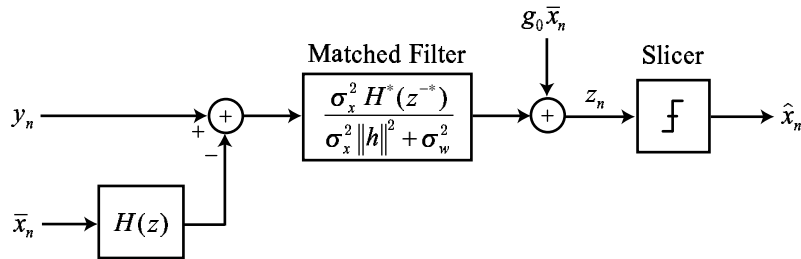


Figure 2.19: An alternative realization of the MMSE interference canceller

The minimum MSE at the MMSE IC output is given by

$$\epsilon_{\min, \text{MMSE-IC}}^2 = \frac{\sigma_x^2 \sigma_w^2}{\sigma_x^2 \|h\|^2 + \sigma_w^2} \quad (2.46)$$

from which we deduce the biased SNR

$$\text{SNR}_{\text{MMSE-IC}} = \frac{\sigma_x^2 \|h\|^2}{\sigma_w^2} + 1 = \frac{T}{2\pi} \int_{-\pi/T}^{+\pi/T} (\text{SNR}_C(\omega) + 1) d\omega = \mathcal{A}(\text{SNR}_C(\omega) + 1) \quad (2.47)$$

For this particular equalizer, the unbiased SNR admits an interesting closed form

$$\text{SNR}_{\text{MMSE-IC,U}} = \mathcal{A}(\text{SNR}_C(\omega) + 1) - 1 = \mathcal{A}(\text{SNR}_C(\omega)) = \text{SNR}_{\text{MFB}} \quad (2.48)$$

Thus, and in contrast with DFE or LE, the ideal MMSE IC is potentially able to eliminate all ISI without noise enhancement, since it theoretically achieves the matched filter bound. Note that the ISI-free noise samples at the input of the decision device are generally correlated.

Finite-length realization

The finite-length form of the MMSE IC follows immediately if one notes that the optimum unconstrained filters given by (2.44) are in fact finite-length FIR. Realization of the feedforward filter $P(z)$ indeed requires only L coefficients, and the feedback filter $Q(z)$ necessitates in turn $2L - 1$ coefficients. The optimal MMSE IC is thus naturally a finite-length equalizer when the discrete-time channel model can be approximated as FIR. The feedforward filter is strictly noncausal, so that a minimum restitution delay $\Delta \geq L - 1$ is required for the realization of the MMSE IC. For the sake of completeness, we give the corresponding filters in vector form

$$\mathbf{p} = \frac{\sigma_x^2}{\sigma_x^2 \|h\|^2 + \sigma_w^2} [h_{L-1}^*, \dots, h_0^*]^T \quad (2.49)$$

$$\mathbf{q} = \mathbf{H}^T \mathbf{p} - \frac{\sigma_x^2 \|h\|^2}{\sigma_x^2 \|h\|^2 + \sigma_w^2} \mathbf{e}_L \quad (2.50)$$

Discussion

The previous results have so far assumed perfect knowledge of the transmitted data. In practice however, application of the MMSE IC requires a first equalization step in order to provide preliminary decisions on the data, which typically involves a (possibly adaptive) MMSE LE [72]. Both Wesolowski [191] and Agazzi and Sheshadri [2] pointed out that the error performance of the canceller is critically dependent on the first stage decision device that generates the tentative decisions. The MMSE IC proves indeed to be more sensitive to error propagation than the DFE, a problem which has precluded its use in practical transmission systems. However, interference cancellation has recently met with renewed interest with the advent of “Turbo” processing techniques, as we shall see in chapter 4.

2.2.8 Comparison of equalization schemes

Theoretical comparison

We first provide a theoretical comparison of the performance of the aforementioned equalizers in infinite-length form, based on the unbiased SNR. For a positive argument f , the following inequalities hold between the arithmetic, geometric and harmonic means [190]

$$\mathcal{H}(f) \leq \mathcal{G}(f) \leq \mathcal{A}(f) \quad (2.51)$$

Combining the previous result with (2.9), (2.22), (2.36) and (2.48), it follows that

$$\text{SNR}_{\text{MMSE-LE,U}} \leq \text{SNR}_{\text{MMSE-DFE,U}} \leq \text{SNR}_{\text{MMSE-IC,U}} = \text{SNR}_{\text{MFB}} \quad (2.52)$$

Strict inequalities hold in all cases except on the ideal AWGN channel, for which $\text{SNR}_C(\omega) = \text{cte}$. Consequently, linear equalization is expected to perform worse than (ideal) decision-feedback equalization, which is surpassed in turn by (ideal) interference cancellation. Performance of maximum-likelihood sequence detection lies between the performance of DFE and the upper limit of the matched filter bound at high SNR.

A simple case-study

The four equalization schemes have been simulated in a BPSK transmission context over the 5-tap severe-ISI Proakis C channel model, with impulse response $[0.227, 0.460, 0.688, 0.460, 0.227]$ [138, chap. 10]. The filtering-based equalizers were realized in finite-length form. The number of coefficients and restitution delay were optimized to closely match the performance of the corresponding infinite-length equalizers over a wide SNR range, yielding $(N_p = 32, \Delta = 18)$ for the MMSE LE, $(N_p = 15, N_q = 4, \Delta = 14)$ for the MMSE DFE, and $(N_p = 5, N_q = 9, \Delta = 4)$ for the MMSE IC respectively. The DFE was simulated assuming perfect feedback as well as with tentative decisions, in order to assess the influence of error propagation. The same applied to the MMSE IC which was simulated both with perfect decisions and within a two-stage configuration comprising an MMSE LE as the first equalizer, delivering tentative decisions to the IC. The measured symbol-error rates (SER) are shown in figure 2.20 as a function of the normalized SNR per bit E_b/N_0 . We first note that linear equalization suffers from significant noise enhancement on this channel. As a consequence of the unreliable estimates delivered by the MMSE LE, the interference canceller with tentative decisions from the LE performs even worse. Consequently, decision-feedback equalization provides a significant advantage over the LE for this severe ISI channel, but is surpassed in turn by optimum MLSD. Finally and as expected, ideal interference cancellation effectively achieves the matched filter bound. This is not the case for MLSD, which presents a theoretical asymptotic performance loss of 5 dB on this channel

model, with respect to the ISI-free AWGN channel [146, chap. 2].

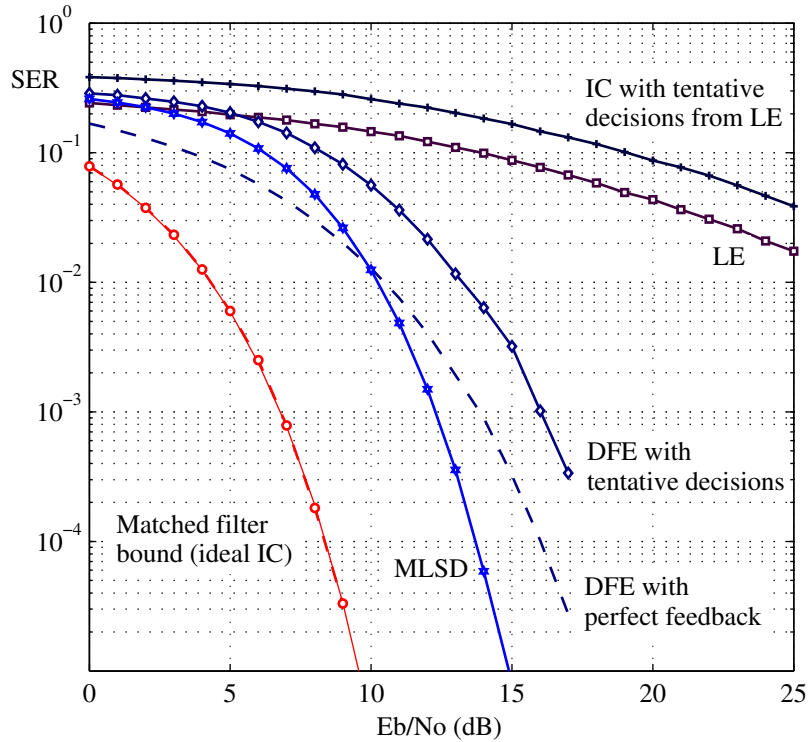


Figure 2.20: Symbol-error rate for the different equalizers with BPSK modulation over the Proakis C channel.

2.3 Summary

In this chapter, we have reviewed two functions of a digital transmission scheme that play an important role in the goal of achieving reliable communications in the presence of additive noise, intersymbol interference, and possibly signal fading, namely channel coding and equalization.

The first part of this chapter was concerned with the desire to combine channel coding techniques with modulation in order to increase the robustness of the system against noise, but without compromising the spectral efficiency of the transmission. Two coded modulation schemes (TCM and BICM) have been introduced which realize attractive solutions for practical transmission systems, depending on the channel characteristics and the complexity that is affordable at the receiver side.

The second part of the chapter was devoted to a survey of the classical equalization strategies that are usually considered for combating ISI over frequency-selective channels. In particular, it has been stressed that while MLSD is theoretically optimum from a communication system design point of view, sub-optimal filtering-based equalizers often offer satisfactory performance, and are considerably less complex to implement in practice.

We now turn our attention in the next chapter to the important problem of combining equalization and decoding in an efficient manner, such that each operation can take advantage of the result of the other in the process of recovering the transmitted message from a received observation.

Chapter 3

Iterative equalization and decoding: Turbo-Equalization

Building upon the tutorial exposition of chapter 2, it is now clear that reliable bandwidth-efficient data transmission over frequency-selective channels calls for some combination of equalization and decoding at the receiver side, in order to overcome the impairments caused by ISI and noise, and possibly mitigate the influence of signal fading.

We first present the general problem of combining equalization and decoding for TCM and BICM, and briefly review the classical solutions advocated as a response in the literature. Then, we introduce the turbo-equalization scheme, where equalization and decoding are performed in an iterative manner, by exchanging soft information at all stages of the process. Specifically, we focus in this chapter on a particular class of turbo-equalizers employing optimal BCJR-MAP symbol detectors and decoders.

The turbo-equalizer is described both in the BICM and TCM cases, and we rigorously prove that the resulting receiver converges towards the performance obtained over an ISI-free channel at high-enough signal-to-noise ratios, assuming sufficient interleaving. In particular, it is shown that the turbo-equalizer may achieve a maximum diversity order of $L \times L_c$ over fully-interleaved multipath Rayleigh fading channels, where L denotes the number of independent propagation paths and L_c is the diversity order of the inner code. Several simulation results are presented by considering different transmission scenarios, which confirm that turbo-equalization may offer significant improvements over the conventional approach where equalization and decoding are usually performed in a separate manner.

We finally apply a semi-analytical analysis tool, the EXIT chart, to the turbo-equalization scheme in order to gain more insight into the convergence behavior of the iterative process, so as to deduce some system design guidelines in return.

3.1 On combined equalization and decoding

Let us consider the general problem of recovering an information message from encoded data in the presence of noise and intersymbol interference. We know from classical estimation theory that the optimal (in the ML or MAP sense) receiver should then perform equalization and decoding jointly, by taking simultaneously into account the constraints introduced by the code and the ISI channel. This however requires us to consider a *super-trellis* whose overall complexity is proportional to the cartesian product of the individual code and ISI trellises.

As an illustration, consider the problem of detecting TCM signals on time-invariant channels in the presence of ISI. Assuming a TCM encoder with S states, an expanded signal set of M points and an ISI channel with L taps, it is known that the resulting super-trellis has $S(M/2)^{L-1}$ states, with $M/2$ transitions emerging from each state [38]. The complexity of the optimal receiver then rapidly becomes prohibitive for practical implementation as the channel memory increases. This is all the more true because TCMs usually employ multilevel signal sets with $M = 8$ points or more. Hence, several reduced-complexity MLSD algorithms have been suggested, which are similar to those mentioned in section 2.2.3 and operate on a reduced number of combined code and ISI states [38, 58]. In particular, an interesting receiver with moderate complexity arises when the super-trellis is reduced to the inner TCM code trellis. ISI is then not taken into account in the trellis states, but rather in the edge metrics, using principles similar to classical decision-feedback equalization. This yields the so-called *parallel decision-feedback decoder* (PDFD), that was first proposed in [192]. In spite of their tractable implementation and good performance, it must be stressed that these practical solutions remain inherently sub-optimal with respect to a full-blown ML receiver operating on the original super-trellis. In fact, a minimum SNR gap of 1–2 dB usually subsists with respect to the ideal performance promised by the matched filter bound. Considering now the detection of TCM signals in time-varying multipath environments, we know that symbol interleaving is usually required in order to combat correlated fadings. However, this results in a super-trellis whose complexity is *exponential* in the interleaver length. The optimal receiver is thus clearly untractable, and one has to resort to a suboptimal two-stage approach, where equalization and decoding are performed separately, preferably with an exchange of soft information in between the two functions. A similar problem is encountered when considering BICM transmissions over frequency-selective channels, where the systematic presence of the bit interleaver precludes a joint equalization and decoding approach, even on time-invariant ISI channels [97].

A major breakthrough occurred in 1995 in the field of combined equalization and decoding with the pioneering work of Douillard *et al.* on *Turbo-Equalization* [47, 50], building upon the principle of *Turbo-Coding* first introduced by Berrou *et al.* in 1993 [27]. Consider the canonical transmission schemes for BICM and TCM over frequency-selective channels depicted in figure 3.1. Note that we have purposely introduced a symbol interleaver in front of the discrete-time ISI channel, be it time-

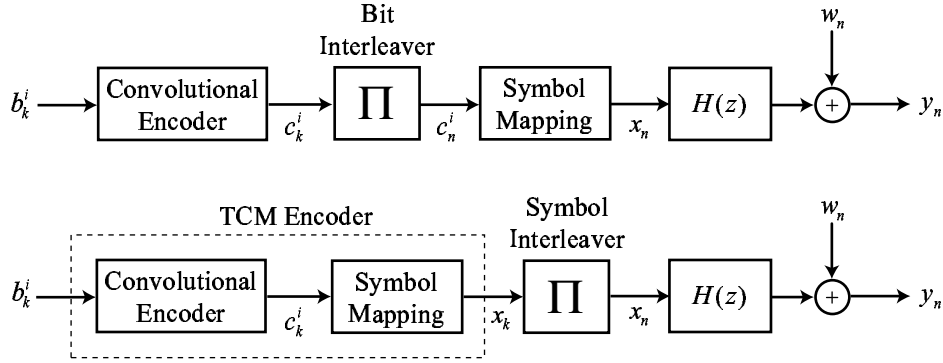


Figure 3.1: Respective block diagrams for BICM and TCM transmission over a discrete-time ISI channel.

varying or time-invariant, in the TCM case. These two transmission systems share striking similarities with a serial concatenation of codes with interleavers [23, chap. 11], where the ISI channel (combined with the mapping operation in the BICM case) acts as a pseudo inner code, whereas the outer code is the BICM convolutional code or a TCM respectively. In the light of this analogy, it thus seems natural to apply a turbo-decoding strategy, where equalization and decoding are performed iteratively with an exchange of soft information. It is expected that as the SNR increases, the performance of the iterative receiver will converge to the optimum performance of the ideal ML receiver operating on the super-trellis which takes into account the constraints imposed by the code, interleaver and ISI channel. Although we are still lacking formal proof of this result yet, the excellent performance obtained by simulation in fact supports such an assumption. Since its introduction, turbo-equalization has received increasing attention, in particular for mobile communications systems [20, 69] and in the field of equalization for magnetic recording channels (see [196] and references therein).

At the core of the turbo-equalizer, we find *Soft-Input Soft-Output* (SISO) modules, i.e. devices which accept and deliver soft decisions. We shall focus in this chapter on a particular form of turbo-equalizer where the SISO equalizer and SISO decoder are implemented using the BCJR-MAP algorithm, which is optimal in the sense of minimizing the symbol error rate (see [19] and the tutorial exposition in appendix C). Hence, the resulting turbo-equalization scheme will be called *MAP turbo-equalizer* in the following, to make the distinction between such a solution and the alternative low-complexity approaches introduced in the next chapter. We now describe in more detail the operations realized by the MAP turbo-equalizer.

3.2 The MAP turbo-equalizer

MAP turbo-equalization was initially proposed for BICM signals transmitted over ISI channels. However, the basic principles are fairly general, and we describe in this section the MAP turbo-equalization scheme as it applies both to TCM and BICM. In fact, the two receivers share the same basic functions,

namely an SISO equalizer and a SISO decoder, separated by an interleaving operation. However, they differ in the fact that the SISO modules operate at the symbol level in the TCM case, and at the bit level for BICM. We first define the concept of *log-likelihood ratio*, which will be used extensively in the subsequent discussions.

3.2.1 Log-likelihood ratios

Let us first consider a coded bit $c_n^i \in \{0, 1\}$. The binary log-likelihood ratio (LLR, or *L-value*) for c_n^i is denoted $L(c_n^i)$ and defined as follows

$$L(c_n^i) = \ln \frac{\Pr(c_n^i = 1)}{\Pr(c_n^i = 0)} \quad (3.1)$$

From this definition, one easily obtains that

$$\Pr(c_n^i = b) = \frac{\exp[b \cdot L(c_n^i)]}{1 + \exp[L(c_n^i)]}, \quad \text{with } b = \{0, 1\} \quad (3.2)$$

The generalization of these definitions to conditional probabilities is straightforward.

Consider now a signal set $\mathcal{S}$ with cardinality $|\mathcal{S}| = M = 2^m$, and assume that the sequence of m coded bits $\{c_n^1, \dots, c_n^m\}$ is mapped onto some symbol $x_n = s$ at time n , with $s \in \mathcal{S}$. The coded bits c_n^i take their value in $\{0, 1\}$ as a function of the considered symbol s and the labelling map. We define the symbol log-likelihood ratio $L_n(s)$ for s at time n as follows

$$L_n(s) = \ln \frac{\Pr(x_n = s)}{\Pr(x_n = s_0)} \quad (3.3)$$

where s_0 is some reference symbol in the signal set. One notices that we can define M different symbol LLRs at any time n , one for each symbol in the signal set. Moreover, we always have $L_n(s_0) = 0$. The symbol probabilities are obtained from the symbol LLRs as follows

$$\Pr(x_n = s) = K_n \times \exp[L_n(s)] \quad (3.4)$$

where the factor K_n is equal to $\Pr(x_n = s_0)$ and common to all symbol LLRs at time n .

The reference symbol s_0 may be chosen arbitrary. However, it proves particularly convenient to choose s_0 such that we have $c_n^i = 0$ for all $i = 1, \dots, m$ according to the signal labelling rule. This convention indeed raises some interesting properties. In particular, let us assume that coded bits c_n^i are i.i.d., owing to the presence of a bit interleaver, for example. Then we have

$$\Pr(x_n = s) = \prod_{i=1}^m \Pr(c_n^i = \ell^i(s)) \quad (3.5)$$

where, according to section 2.1.1, $\ell^i(s)$ denotes the value of the i th bit which is mapped onto the symbol s . Using (3.1), (3.3) and (3.5), we obtain the following interesting relationship between the symbol LLRs and binary LLRs on the coded bits

$$L_n(s) = \sum_{i=1}^m \ell^i(s) \times L(c_n^i) \quad , \quad \text{with } \ell^i(s) \in \{0, 1\} \quad (3.6)$$

Finally, the conversion from symbol LLRs to bit LLRs does not require any independence assumption, and is obtained by noting that

$$\Pr(c_n^i = b) = \sum_{s \in \mathcal{S}_b^i} \Pr(x_n = s) \quad , \quad \text{with } b = \{0, 1\} \quad (3.7)$$

We recall that $\mathcal{S}_b^i$ denotes the subset of all signals $s \in \mathcal{S}$ whose label $\ell^i(s)$ has value $b = \{0, 1\}$ in position i . This yields

$$L(c_n^i) = \ln \frac{\sum_{s \in \mathcal{S}_b^i} \exp[L_n(s)]}{\sum_{s \in \mathcal{S}_b^0} \exp[L_n(s)]} \quad (3.8)$$

3.2.2 MAP turbo-equalization for TCM

Consider the turbo-equalization scheme for TCM as depicted in figure 3.2. As exposed in chapter 2, we recall that the transmission of m information bits per modulated symbol requires an expanded signal set with $M = 2^{m+1}$ points. In the sequel, we shall consider packet transmission of finite-length messages $\{x_0, \dots, x_{N-1}\}$. The corresponding sequence of N observations $\{y_n\}$ at the channel output will be denoted hereafter by $\mathbf{Y} = \{y_0, \dots, y_{N-1}\}$. The following exposition makes use of the notations introduced in section 2.1.1 and figure 3.1. We note that a similar derivation has been proposed in [120], but where symbol probabilities are used instead of symbol LLRs.

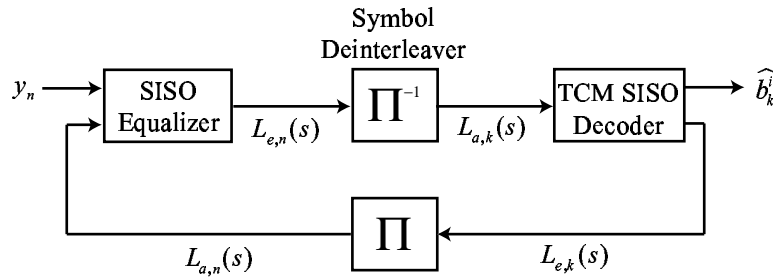


Figure 3.2: Turbo-equalization scheme for TCM.

On the basis of the observation $\mathbf{Y}$ and the *a priori* symbol LLRs $L_{a,n}(s)$ available at its input, the

SISO symbol equalizer internally computes the *a posteriori* symbol LLRs $L_n(s)$ defined by

$$L_n(s) = \ln \frac{\Pr(x_n = s | \mathbf{Y}, \{L_{a,n}(s)\})}{\Pr(x_n = s_0 | \mathbf{Y}, \{L_{a,n}(s)\})} \quad (3.9)$$

Let us introduce the $\max^*$ operator, defined by

$$\max^*(x, y) = \ln(e^x + e^y) = \max(x, y) + \ln(1 + e^{-|x-y|}) \quad (3.10)$$

Application of the BCJR-MAP algorithm yields the computation procedure [24]

$$L_n(s) = \max_{v \rightarrow v': s}^* \{A_{n-1}(v) + \Gamma_n(v \rightarrow v') + B_n(v')\} - \max_{v \rightarrow v': s_0}^* \{A_{n-1}(v) + \Gamma_n(v \rightarrow v') + B_n(v')\} \quad (3.11)$$

where the notation $v \rightarrow v' : s$ denotes the subset of all transitions labelled with the symbol s between pairs of states (v, v') in the ISI trellis at time n . The quantity $\Gamma_n(v \rightarrow v')$ is precisely the branch metric associated with those transitions, and reads

$$\Gamma_n(v \rightarrow v') = -\frac{1}{\sigma_w^2} \left| y_n - h_0 s - \sum_{\ell=1}^{L-1} h_\ell \hat{x}_{n-\ell}(v) \right|^2 + L_{a,n}(s) \quad (3.12)$$

where the $L-1$ symbol estimates $\hat{x}_{n-\ell}(v)$ are obtained from the knowledge of the starting state v . The partial state metrics $A_n(v)$ and $B_n(v')$ are computed in a recursive manner, using the relations

$$A_n(v') = \max_{v \rightarrow v'}^* \{A_{n-1}(v) + \Gamma_n(v \rightarrow v')\} + K_1 \quad (3.13)$$

$$B_n(v) = \max_{v \rightarrow v'}^* \{\Gamma_{n+1}(v \rightarrow v') + B_{n+1}(v')\} + K_2 \quad (3.14)$$

where K_1 and K_2 are suitable normalization constants satisfying

$$\max_{\text{all } v}^* \{A_n(v)\} = \max_{\text{all } v}^* \{B_n(v)\} = 0 \quad , \quad \text{at any time } n \quad (3.15)$$

We suppose that the trellis begins in the zero-state, but is left untruncated, so that we have the respective initial and final conditions

$$\begin{cases} A_0(0) = 0, A_0(v) = 0 \text{ for } v > 0 \\ B_N(v) = -\ln M \text{ for all } v \end{cases} \quad (3.16)$$

The SISO equalizer delivers extrinsic symbols LLRs $L_{e,n}(s)$, given by

$$L_{e,n}(s) = L_n(s) - L_{a,n}(s) \quad (3.17)$$

The set of extrinsic symbol LLRs is deinterleaved, yielding the shuffled sequence $L_{a,k}(s)$, which is

then sent to the SISO TCM decoder. It is interesting to note that for an ISI-free channel, the SISO symbol equalizer reduces to a symbol soft-output demodulator, delivering the metrics required for MAP/ML decoding of the TCM in LLR form.

The SISO TCM decoder uses the *a priori* information $L_{a,k}(s)$ to evaluate in turn new *a posteriori* symbols LLRs $L_k(s)$, defined by

$$L_k(s) = \ln \frac{\Pr(x_n = s | \{L_{a,k}(s)\})}{\Pr(x_n = s_0 | \{L_{a,k}(s)\})} \quad (3.18)$$

The application of the BCJR-MAP algorithm yields

$$L_k(s) = \max_{v \rightarrow v': s}^* \{A_{k-1}(v) + \Gamma_k(v \rightarrow v') + B_k(v')\} - \max_{v \rightarrow v': s_0}^* \{A_{k-1}(v) + \Gamma_k(v \rightarrow v') + B_k(v')\} \quad (3.19)$$

where the $\max^*$ operation is still performed over the subset of transitions $v \rightarrow v'$ labelled with the desired symbols, but in the TCM trellis. The partial state metrics $A_k(v)$ and $B_k(v)$ are recursively computed using (3.13) and (3.14), except that the trellis is now assumed to begin and end in the zero-state

$$\begin{cases} A_0(0) = 0, A_0(v) = 0 \text{ for } v > 0 \\ B_N(0) = 0, B_N(v) = 0 \text{ for } v > 0 \end{cases} \quad (3.20)$$

The major difference occurs with the branch metrics $\Gamma_k(v \rightarrow v')$ which are now simply given by [24]

$$\Gamma_k(v \rightarrow v') = L_{a,k}(s) \quad (3.21)$$

assuming that s is the symbol labelling the considered transition. The SISO decoder finally delivers hard decisions $\hat{b}_k^i$ on the information bits, as well as updated extrinsic LLRs on the modulated symbols

$$L_{e,k}(s) = L_k(s) - L_{a,k}(s) \quad (3.22)$$

These extrinsic LLRs are interleaved again, and sent back to the SISO equalizer where they are exploited as updated *a priori* information for a new equalization attempt at the next iteration. This completes the description of the system in the symbol-level case.

3.2.3 MAP turbo-equalization for BICM

We now present the turbo-equalization in its most classical form, as it applies to BICM transmissions (see figure 3.3). We recall that in contrast to the previous scheme, both the SISO equalizer and the SISO decoder operate here at the bit-level.

Several detailed derivations of the SISO equalizer at the bit-level are available in the literature

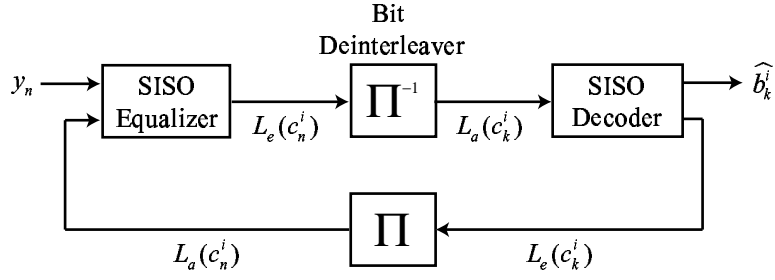


Figure 3.3: Turbo-equalization scheme for BICM.

(see e.g. [21]). However we find it more insightful to consider the alternative interpretation shown in figure 3.4, which is naturally suggested by the fact that the SISO equalizer necessarily works at the symbol-level internally.

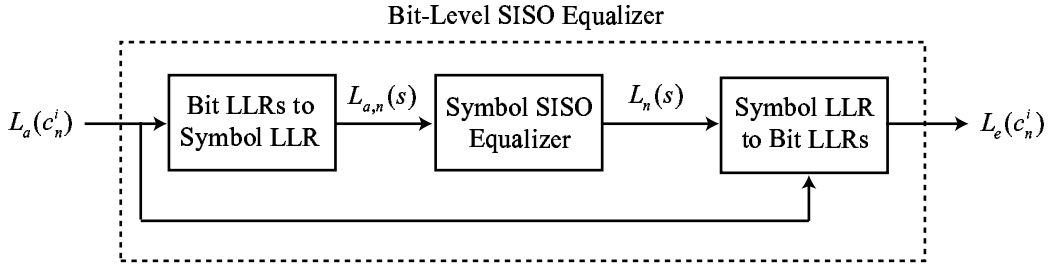


Figure 3.4: An alternative interpretation for the bit-level SISO equalizer.

The inner SISO symbol equalizer operates exactly in the same way as for the TCM case to compute the *a posteriori* LLRs $L_n(s)$ on the modulation symbols. However, a first operation is required at the equalizer input, to convert the *a priori* LLRs $L_a(c_n^i)$ on the coded bits into *a priori* LLRs $L_{a,n}(s)$ on the modulated symbols. To be precise, assuming that successive coded bits are independent due to the presence of the bit interleaver, we obtain using (3.6)

$$L_{a,n}(s) = \sum_{i=1}^m \ell^i(s) \times L_a(c_n^i) \quad (3.23)$$

Conversely, the *a posteriori* LLRs $L(c_n^i)$ on the coded bits are evaluated from the knowledge of the *a posteriori* symbol LLRs $L_n(s)$ delivered by the symbol equalizer using (3.8)

$$L(c_n^i) = \max_{s \in S_b^1} \{L_n(s)\} - \max_{s \in S_b^0} \{L_n(s)\} \quad (3.24)$$

The extrinsic LLRs $L_e(c_n^i)$ are finally obtained by subtracting the *a priori* information relative to the considered coded bit, yielding

$$L_e(c_n^i) = L(c_n^i) - L_a(c_n^i) \quad (3.25)$$

The extrinsic LLRs are deinterleaved, yielding the shuffled sequence $L_a(c_k^i)$ which is then sent to the

SISO convolutional decoder. Note that for an ISI-free channel, the SISO bit-level equalizer reduces to a conventional SISO symbol demapper, delivering the metrics required for the MAP/ML decoding of the outer convolutional code in LLR form.

On the basis of the *a priori* information $L_a(c_k^i)$, the SISO decoder computes *a posteriori* LLRs $L(c_k^i)$ on the coded bits defined as follows

$$L(c_k^i) = \ln \frac{\Pr(c_k^i = 1 | \{L_a(c_k^i)\})}{\Pr(c_k^i = 0 | \{L_a(c_k^i)\})} \quad (3.26)$$

Applying again the BCJR-MAP algorithm yields the computation procedure

$$L(c_k^i) = \max_{v \rightarrow v': c_k^i=1}^* \{A_{k-1}(v) + \Gamma_k(v \rightarrow v') + B_k(v')\} - \max_{v \rightarrow v': c_k^i=0}^* \{A_{k-1}(v) + \Gamma_k(v \rightarrow v') + B_k(v')\} \quad (3.27)$$

where the $\max^*$ operation is performed over all transitions $v \rightarrow v'$ in the convolutional code trellis labelled with the desired value (0 or 1) for the coded bit c_k^i at time k . The trellis is assumed to begin and end in the zero-state. The partial state metrics $A_k(v)$ and $B_k(v)$ are computed in a recursive manner using (3.13) and (3.14). The branch metric $\Gamma_k(v \rightarrow v')$ admits the following expression

$$\Gamma_k(v \rightarrow v') = \sum_{i=1}^{n_c} c^i \cdot L_a(c_k^i) \quad (3.28)$$

where the coded bits c^i take their value in $\{0, 1\}$ as a function of the considered transition. The SISO decoder finally delivers hard decisions $\hat{b}_k^i$ on the information message, as well as updated extrinsic LLRs $L_e(c_k^i)$ on the coded bits, given by

$$L_e(c_k^i) = L(c_k^i) - L_a(c_k^i) \quad (3.29)$$

These extrinsic LLRs are interleaved again and sent back to the SISO equalizer for the next iteration.

3.3 Asymptotic performance bounds for MAP turbo-equalization

We establish in this section simple lower bounds on the performance of turbo-equalization for TCM and BICM, in the limit of perfect feedback from the decoder. The latter assumption equivalently translates into the hypothesis of ideal *a priori* information at the equalizer input.

These bounds are of great practical importance since, as the SNR increases, the soft decisions delivered by the decoder tend to become more and more reliable. As a result, the turbo-equalizer effectively converges towards these bounds, which are in fact quite intuitive, and easily suggested by simulations. However, we are not aware of any previous formal proof of these results in the literature.

We first consider the TCM case.

Proposition 3.1 (Asymptotic bound for TCM turbo-equalization). *In the limit of perfect a priori knowledge about the transmitted symbols, turbo-equalization for TCM approaches the performance of MAP/ML decoding of TCM over an ISI free channel with equivalent signal-to-noise ratio SNR_{MFB} , independently of the ISI channel characteristics.*

Proof. Consider the SISO symbol equalizer operating in the presence of perfect *a priori* knowledge about the transmitted data sequence. In such conditions, the *a posteriori* LLR about the symbol s at time n is given by

$$L_n(s) = \ln \frac{\Pr(x_n = s | \{y_k\}, \{x_k\}_{k \neq n})}{\Pr(x_n = s_0 | \{y_k\}, \{x_k\}_{k \neq n})} \quad (3.30)$$

Using the chain rule of conditional probabilities¹, the previous expression may be equivalently rewritten

$$L_n(s) = \underbrace{\ln \frac{\Pr(x_n = s)}{\Pr(x_n = s_0)}}_{=L_{a,n}(s)} + \underbrace{\ln \frac{P(\{x_k\}_{k \neq n} | x_n = s)}{P(\{x_k\}_{k \neq n} | x_n = s_0)}}_{=0} + \ln \frac{P(\{y_k\} | \{x_k\}_{k \neq n}, x_n = s)}{P(\{y_k\} | \{x_k\}_{k \neq n}, x_n = s_0)} \quad (3.31)$$

where the cancellation of the second member comes from the fact that successive coded symbols are supposed independent owing to the presence of the symbol interleaver. We deduce that the extrinsic LLR on the symbol s at time n admits the following expression

$$L_{e,n}(s) = L_n(s) - L_{a,n}(s) = \ln \frac{P(\{y_k\} | \{x_k\}_{k \neq n}, x_n = s)}{P(\{y_k\} | \{x_k\}_{k \neq n}, x_n = s_0)} \quad (3.32)$$

Assume an ISI channel model with L taps h_ℓ . Then, the observations y_k for $k < n$ and $k \geq L$ are irrelevant to x_n given the knowledge of the partial data sequence $\{x_k\}_{k \neq n}$. Under the assumption of independent noise samples, $L_{e,n}(s)$ simplifies into

$$L_{e,n}(s) = \ln \frac{\prod_{i=0}^{L-1} P(y_{n+i} | x_n = s, \{x_{n+i-\ell}\}_{\ell=0, \dots, L-1; \ell \neq i})}{\prod_{i=0}^{L-1} P(y_{n+i} | x_n = s_0, \{x_{n+i-\ell}\}_{\ell=0, \dots, L-1; \ell \neq i})} \quad (3.33)$$

with

$$P(y_{n+i} | x_n = s, \{x_{n+i-\ell}\}_{\ell=0, \dots, L-1; \ell \neq i}) = \frac{1}{\sigma_w^2} \exp \left[- \frac{|y_{n+i} - h_i s - \sum_{\ell \neq i} h_\ell x_{n+i-\ell}|^2}{\sigma_w^2} \right] \quad (3.34)$$

The previous expression can be rewritten in more compact form as follows

$$L_{e,n}(s) = \frac{1}{\sigma_w^2} \sum_{i=0}^{L-1} \left[|y_{n+i} - h_i s_0 - \bar{y}_{n+i}|^2 - |y_{n+i} - h_i s - \bar{y}_{n+i}|^2 \right] \quad (3.35)$$

¹ $f(x_1, \dots, x_N) = f(x_1)f(x_2|x_1) \cdots f(x_N|x_1, \dots, x_{N-1})$.

where we have introduced the quantity $\bar{y}_{n+i} = \sum_{\ell \neq i} h_\ell x_{n+i-\ell}$. We recall that y_n is given by (see (2.2))

$$y_n = \sum_{\ell=0}^{L-1} h_\ell x_{n-\ell} + w_n \quad (3.36)$$

It follows that $y_{n+i} - \bar{y}_{n+i} = h_i x_n + w_{n+1}$. Inserting this expression into equation (3.35) yields

$$L_{e,n}(s) = \frac{1}{\sigma_w^2} \sum_{i=0}^{L-1} \left[|h_i(x_n - s_0) + w_{n+i}|^2 - |h_i(x_n - s) + w_{n+i}|^2 \right] \quad (3.37)$$

$$= \frac{1}{\sigma_w^2} \sum_{i=0}^{L-1} \left[|h_i|^2 (|x_n - s_0|^2 - |x_n - s|^2) + 2 \operatorname{Re} \{ h_i(s - s_0) w_{n+i}^* \} \right] \quad (3.38)$$

$$= \frac{1}{\sigma_w^2} \sum_{i=0}^{L-1} \left[|h_i|^2 (|s_0|^2 - |s|^2 + 2 \operatorname{Re} \{ (s - s_0) x_n^* \}) + 2 \operatorname{Re} \{ h_i(s - s_0) w_{n+i}^* \} \right] \quad (3.39)$$

Let $\|h\|^2 = \sum_{\ell=0}^{L-1} |h_\ell|^2$, and define

$$z_n = \|h\|^2 x_n + \sum_{\ell=0}^{L-1} h_\ell^* w_{n+\ell} \quad (3.40)$$

Then, $L_{e,n}(s)$ can be equivalently rewritten

$$L_{e,n}(s) = \frac{1}{\sigma_w^2} \left[\|h\|^2 (|s_0|^2 - |s|^2) + 2 \operatorname{Re} \{ (s - s_0) z_n^* \} \right] \quad (3.41)$$

$$= \frac{1}{\|h\|^2 \sigma_w^2} \left[\left| z_n - \|h\|^2 s_0 \right|^2 - \left| z_n - \|h\|^2 s \right|^2 \right] \quad (3.42)$$

$$= \ln \frac{P(z_n | x_n = s)}{P(z_n | x_n = s_0)} \quad (3.43)$$

where we have identified z_n with an observation at the output of an ISI free channel with gain factor $\|h\|^2$ and additive colored Gaussian noise with total variance $\|h\|^2 \sigma_w^2$. Hence, it looks as if the equalizer had converted the original ISI channel into an equivalent ISI free channel with signal-to-noise ratio $\text{SNR} = \text{SNR}_{\text{MFB}} = \sigma_x^2 \|h\|^2 / \sigma_w^2$, from the viewpoint of the TCM inner SISO decoder. The final expression obtained for $L_{e,n}(s)$ corresponds precisely to the *a priori* LLRs required for MAP/ML decoding of the TCM over this equivalent ISI free channel. This concludes the proof. $\square$

Regarding now the turbo-equalization scheme for BICM, we have the following proposition.

Proposition 3.2 (Asymptotic bound for BICM turbo-equalization). *In the limit of perfect a priori knowledge about the transmitted coded bits, turbo-equalization for BICM approaches the performance of iterative demapping and decoding with perfect feedback from the decoder (the genie receiver of section 2.1.4) over an ISI free channel with equivalent signal-to-noise ratio SNR_{MFB} , independently of the ISI channel characteristics.*

Proof. The proof essentially follows along the same lines as for the TCM case. Hence, we do not develop all the calculations, but rather highlight the main points in the derivations.

Consider the SISO bit-level equalizer operating in the presence of perfect *a priori* information about the transmitted coded bits. The *a posteriori* LLR $L(c_n^i)$ computed at time n by the equalizer is defined as

$$L(c_n^i) = \ln \frac{P(c_n^i = 1 | \{y_k\}, \{x_k\}_{k \neq n}, \{c_n^j\}_{j \neq i})}{P(c_n^i = 0 | \{y_k\}, \{x_k\}_{k \neq n}, \{c_n^j\}_{j \neq i})} \quad (3.44)$$

where we have used the fact that there is a one-to-one correspondence between the set of coded bits and the transmitted data sequence $\{x_k\}$. Using again the chain rule of conditional probabilities, one obtains the following expression for the extrinsic LLR about c_n^i

$$L_e(c_n^i) = \ln \frac{P(\{y_k\} | \{x_k\}_{k \neq n}, \{c_n^j\}_{j \neq i}, c_n^i = 1)}{P(\{y_k\} | \{x_k\}_{k \neq n}, \{c_n^j\}_{j \neq i}, c_n^i = 0)} \quad (3.45)$$

Now let $s^{(1)}$ and $s^{(0)}$ denote the two symbols in the signal set $\mathcal{S}$ which are perfectly defined by the knowledge of the $m - 1$ coded bits $\{c_n^j\}_{j \neq i}$ at time n , and which differ only by the value taken by c_n^i (1 or 0). Then, $L_e(c_n^i)$ can be equivalently rewritten as

$$L_e(c_n^i) = \ln \frac{P(\{y_k\} | \{x_k\}_{k \neq n}, x_n = s^{(1)})}{P(\{y_k\} | \{x_k\}_{k \neq n}, x_n = s^{(0)})} \quad (3.46)$$

Exploiting the results derived in the TCM case, we finally obtain

$$L_e(c_n^i) = \frac{1}{\|h\|^2 \sigma_w^2} \left[\left| z_n - \|h\|^2 s^{(0)} \right|^2 - \left| z_n - \|h\|^2 s^{(1)} \right|^2 \right] = \ln \frac{P(z_n | x_n = s^{(1)})}{P(z_n | x_n = s^{(0)})} \quad (3.47)$$

where we have introduced the quantity z_n , defined as follows

$$z_n = \|h\|^2 x_n + \sum_{\ell=0}^{L-1} h_\ell^* w_{n+\ell} \quad (3.48)$$

Hence, it again looks as if the SISO equalizer had turned the ISI channel into an equivalent ISI free channel with signal-to-noise ratio $\text{SNR} = \text{SNR}_{\text{MFB}} = \sigma_x^2 \|h\|^2 / \sigma_w^2$. Moreover, one recognizes in (3.47) the extrinsic LLRs that would be delivered with a SISO demapper operating over this equivalent channel, *in the presence of perfect a priori information about the transmitted coded bits*. This corresponds precisely to the first stage of the *genie* receiver for BICM-ID introduced in section 2.1.4, and thus completes the proof. $\square$

Discussion

The two previous propositions have important practical consequences.

First, they show that the turbo-equalization scheme is theoretically able to completely eliminate ISI at high enough SNRs, thereby *achieving the ideal matched filter bound at the equalizer's output*. In particular, we have seen that the turbo-equalizer approaches an ideal iterative demapping and decoding scheme in the BICM case. This suggests that, for BICM transmissions over frequency-selective channels, any signal set labelling optimization is worth having a look. In particular, Gray mapping may not be the optimum choice, depending on the desired bit-error rate and convergence threshold.

Now, let us consider the situation where the channel tap coefficients $\{h_\ell\}$ are time-varying independent complex random variables with Rayleigh-distributed square magnitude. We have shown that the turbo-equalizer ideally converts the ISI channel into an equivalent ISI free channel with average signal-to-noise ratio

$$\text{SNR}_{\text{MFB}} = \frac{\sigma_x^2}{\sigma_w^2} E(\|h\|^2) = \frac{\sigma_x^2}{\sigma_w^2} \sum_{\ell=0}^{L-1} E(|h_\ell|^2) \quad (3.49)$$

But this is exactly the signal-to-noise ratio obtained at the output of a *maximum-ratio combiner*, in the presence of L independent (and ISI free) diversity branches, with respective average signal-to-noise ratio [23, sec. 13.4]

$$\text{SNR}_\ell = \frac{\sigma_x^2}{\sigma_w^2} E(|h_\ell|^2) \quad (3.50)$$

on each branch. Hence, we conclude that in the limit of perfect *a priori* information, *the turbo-equalization scheme may achieve a maximum asymptotic diversity order L at the equalizer's output, if the L taps have equal average power*. Provided that the inner coding scheme itself provides a code diversity of order L_c , we may thus expect the turbo-equalizer to provide an *overall maximum diversity order $L \times L_c$, over a fully-interleaved multipath Rayleigh fading channel with taps of equal average power, where coded symbols (or bits) face independent realizations of the channel coefficients*.

As a final remark, it is interesting to note that the proofs of these bounds provide explicit support for the use of extrinsic information in iterative decoding schemes, a quantity that was initially introduced in a somewhat heuristic manner for turbo-codes [26]. Moreover, an exact expression for extrinsic information has been obtained in the limit of perfect *a priori* information. We believe that interesting results could be obtained by applying similar arguments to parallel and serial turbo-codes.

3.4 Simulation results

In order to support the previous discussions, we now present some performance results obtained by Monte-Carlo simulations of the MAP turbo-equalizer for the three examples of coded modulation

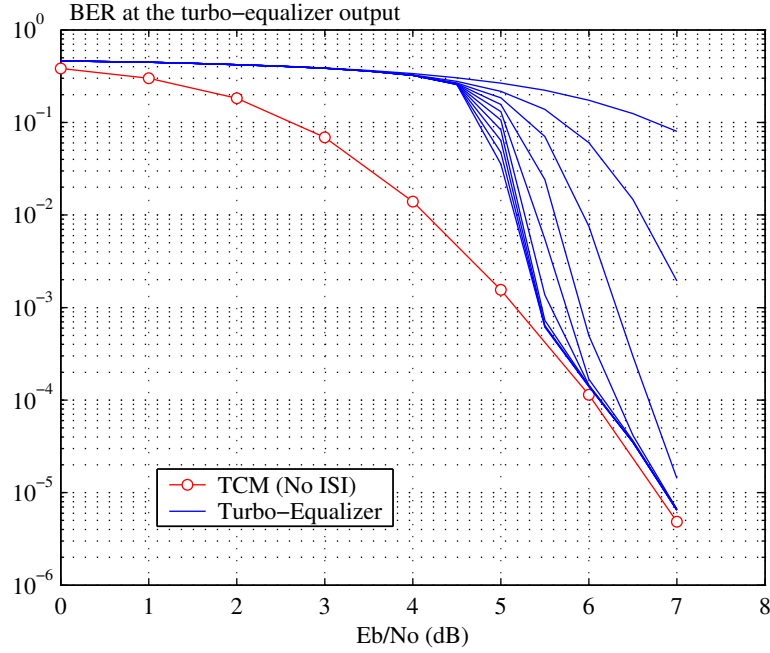


Figure 3.5: BER performance of the MAP turbo-equalizer for the 8-PSK TCM scheme, with 10 iterations, over the Proakis B channel model.

schemes introduced in section 2.1.4.

3.4.1 Performance over a time-invariant channel

We first present the performance obtained over the 3-tap severe-ISI time-invariant channel model Proakis B, obtained from [138, chap. 10], and with impulse response $[0.407, 0.815, 0.407]$. For all three codes, a random interleaver was used with size 8196 (coded symbols for TCM, or coded bits for BICM), and 10 iterations were performed. 100 erroneous frames were totalized for each simulation point.

Figure 3.5 shows the system bit-error rate (BER) of the TCM scheme for the different iterations and as a function of the normalized signal-to-noise ratio per information bit E_b/N_0 (in decibels). We first observe that the turbo-equalizer essentially reaches the performance obtained with MAP decoding of the TCM code over an AWGN channel for SNR values equal to or greater than 5.5 dB. This result confirms that the MAP turbo-equalizer can effectively achieve the lower bound presented in the previous section in practice, at sufficiently high SNRs. In addition, only 5-6 iterations are necessary. Not shown here but worth mentioning is the fact that a significant performance gain, about 4 dB, is obtained with respect to the first iteration, where equalization does not benefit from any prior information from the decoder.

Figures 3.6 and 3.7 show the system BER obtained respectively with the 8-PSK and 16-QAM BICM schemes. We observe in both cases a convergence of the turbo-equalizer toward the performance obtained with the genie receiver of section 2.1.4 (over an AWGN channel), once the SNR reaches a given minimum threshold (of 7.2 dB for the 8-PSK code, and 9.25 dB for the 16-QAM code). We note by extrapolating the curves that significant performance gains are then obtained with respect to the first iteration. This validates the second lower bound introduced in section 3.3.

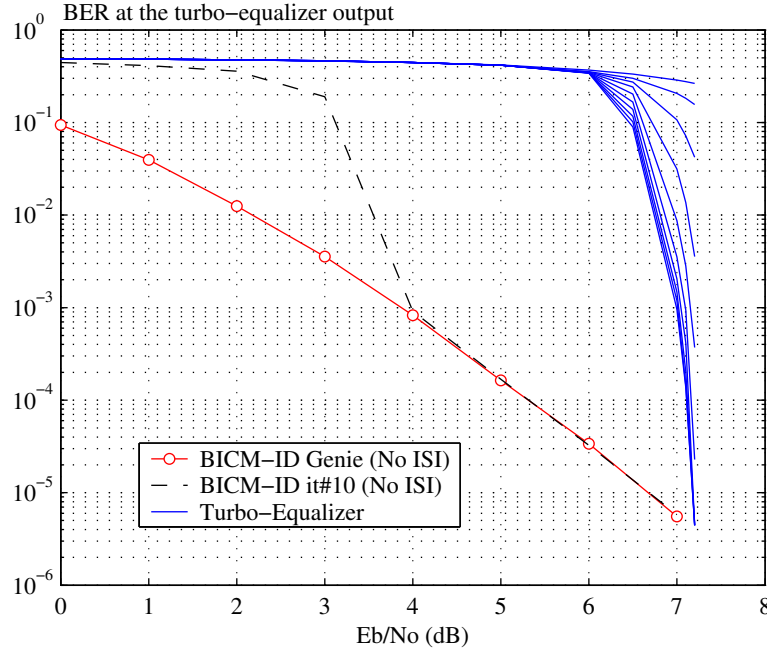


Figure 3.6: BER performance of the MAP turbo-equalizer for the 8-PSK BICM scheme, with 10 iterations, over the Proakis B channel model.

If we finally compare the performance of the three schemes, the TCM clearly appears as the most interesting solution for this channel model. This conclusion stems from the fact that for the BICM schemes, the convergence only occurs at SNR values where they are outperformed by the TCM code over an ISI-free channel.

3.4.2 Performance over a fully-interleaved multipath Rayleigh fading channel

The same transmission systems have been simulated over a discrete-time multipath Rayleigh fading channel, consisting of two symbol-spaced taps of equal average power. We call this model EQ2. More generally, the name EQX will in this document designate a multipath Rayleigh fading channel with X taps of equal power, spaced T seconds apart where T is the symbol period.

In fact, a fast-fading channel was considered, characterized by a normalized Doppler bandwidth

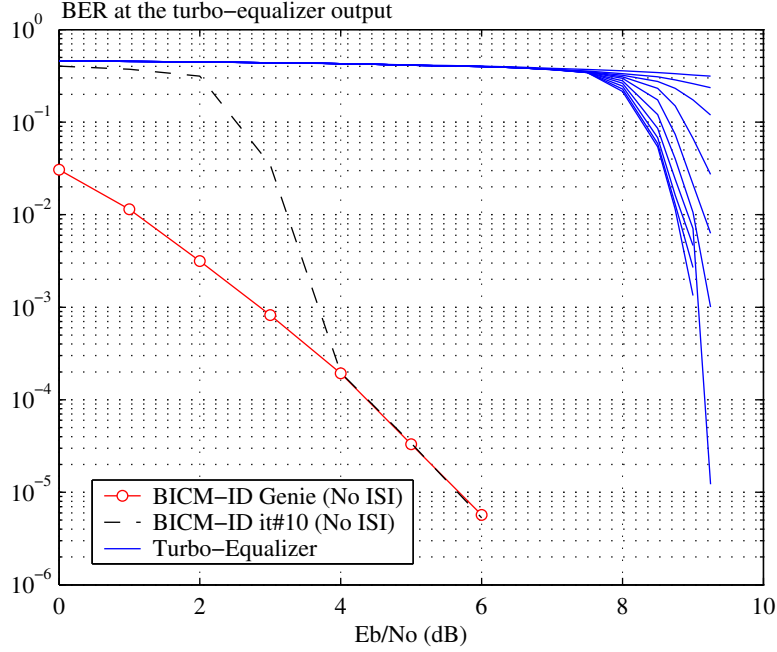


Figure 3.7: BER performance of the MAP turbo-equalizer for the 16-QAM BICM scheme, with 10 iterations, over the Proakis B channel model.

$B_d T = 10^{-2}$, where B_d is the two-sided Doppler bandwidth. In these conditions, we found that an interleaver size of 8196 coded symbols (or coded bits for BICM) was in fact sufficient to satisfy the hypothesis of a memoryless channel at the receiver input. The multipath Rayleigh fading channel was simulated according to the method described in [200]. 10 iterations were performed in each case.

The simulation results for the three coded modulation schemes are shown in figures 3.8 to 3.10. We have also included the curves obtained in chapter 2 over the AWGN channel and the one-path (EQ1) fully-interleaved Rayleigh channel for reference purposes. Using the technique introduced in section 2.1.4, we measured the total diversity order L_{tot} at the turbo-equalizer output and obtained $L_{\text{tot}} = 2 \times 2 = 4$ for the TCM, $L_{\text{tot}} = 2 \times 4 = 8$ for the 8-PSK BICM-ID, and $L_{\text{tot}} = 2 \times 6 = 12$ for the 16-QAM BICM-ID respectively. This confirms our previous claim that on fully-interleaved multipath Rayleigh channels with taps of equal average power, the turbo-equalizer is able to simultaneously exploit the full diversity offered both by the multipath channel and the inner code, thereby providing significant performance improvement over the flat-fading case.

Moreover, the following observations can be drawn from inspection of the simulation results.

- The performance gain obtained with respect to the flat-fading channel is all the more important that the coding scheme has a low diversity order. As an illustration, we observe at a BER of 10^{-4} a 4 dB gain for the TCM, a 2.5 dB gain for the 8-PSK BICM-ID, and a 1.4 dB gain only

for the 16-QAM BICM-ID.

- The best asymptotic performance is obtained with the coding schemes having the highest diversity order. In particular, the performance of the 16-QAM BICM-ID at a BER of $2 \cdot 10^{-5}$ over the EQ2 channel is only 1.5 dB away from the performance over the AWGN channel.

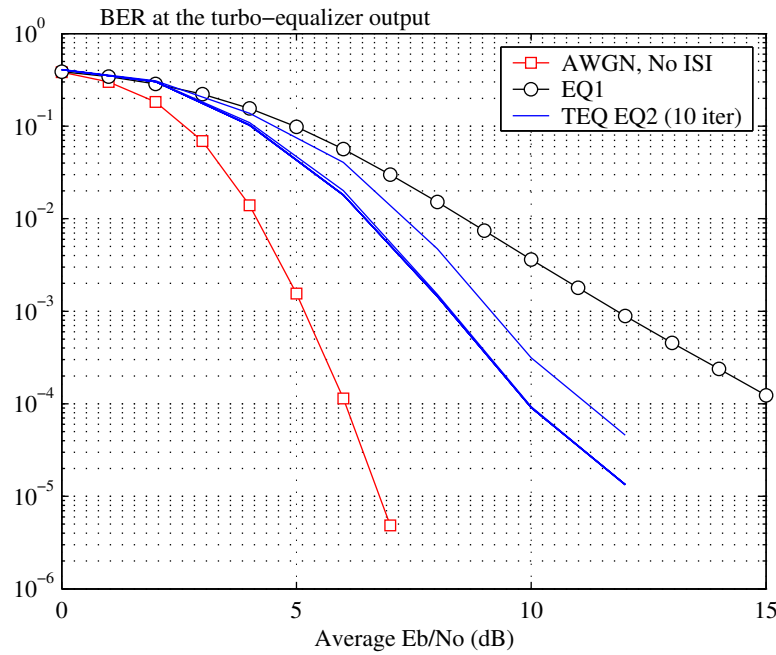


Figure 3.8: BER performance of the MAP turbo-equalizer for the 8-PSK TCM scheme, with 10 iterations, over the fully-interleaved EQ2 multipath Rayleigh fading channel model.

To summarize, the code diversity order clearly plays an even more important role on fully-interleaved multipath Rayleigh fading channels than it does for flat-fading channels, assuming turbo-equalization at the receiver side in the former case. Hence, it should not come as a surprise that the BICM-ID schemes outperform the TCM in this context.

3.4.3 Performance over a quasi-static multipath Rayleigh fading channel

To conclude this study, we finally simulated the coded modulation schemes over a quasi-static EQ2 multipath Rayleigh fading channel. Data blocks of 256 modulated symbols were considered, to remain consistent with the quasi-static assumption. 10 iterations were performed in each case. The frame-error rate for the three transmission systems is shown in figures 3.11 to 3.13. The curves obtained in chapter 2 over a one-path (EQ1) quasi-static channel have been included for reference purposes. We have also shown the ideal performance that could be reached over the EQ2 channel with a genie-aided MAP equalizer, supplied with perfect feedback from the decoder (dashed lines).

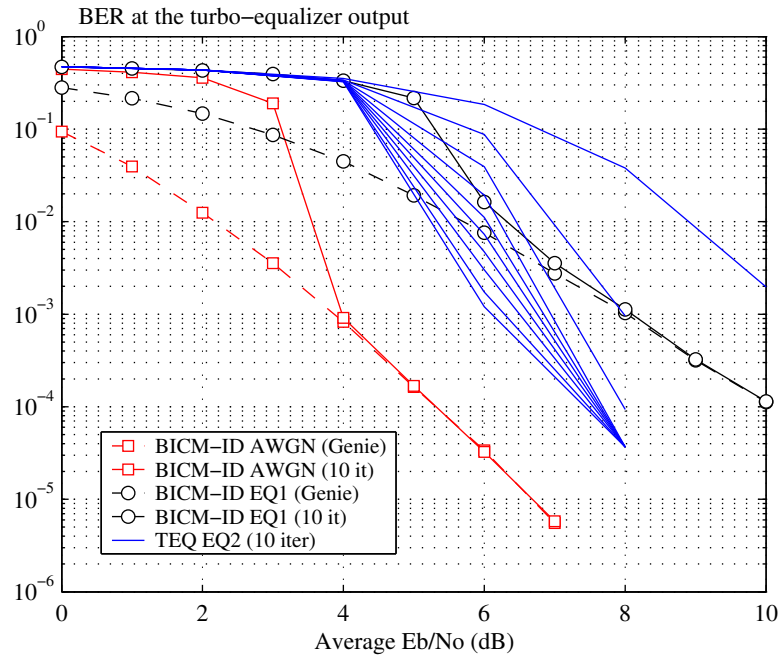


Figure 3.9: BER performance of the MAP turbo-equalizer for the 8-PSK BICM scheme, with 10 iterations, over the fully-interleaved EQ2 multipath Rayleigh fading channel model.

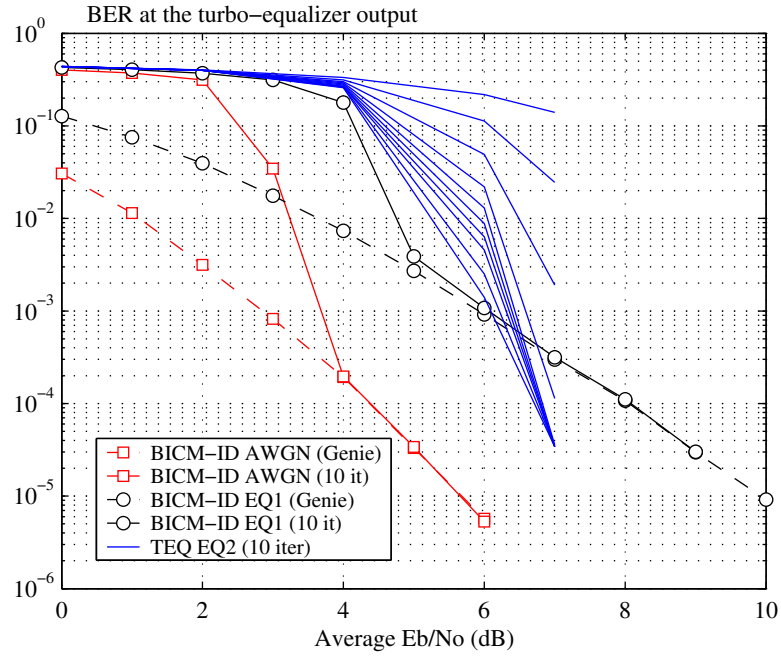


Figure 3.10: BER performance of the MAP turbo-equalizer for the 16-QAM BICM scheme, with 10 iterations, over the fully-interleaved EQ2 multipath Rayleigh fading channel model.

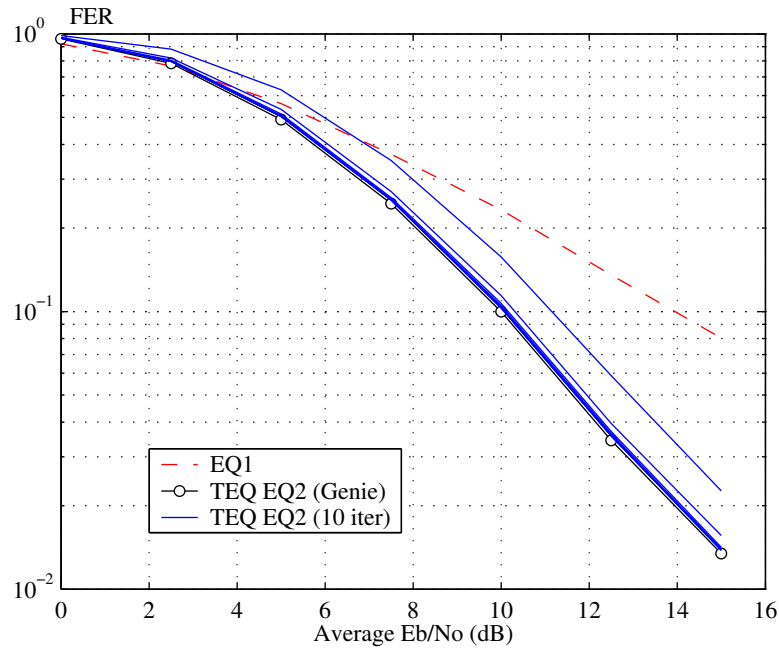


Figure 3.11: FER performance of the MAP turbo-equalizer for the 8-PSK TCM scheme, with 10 iterations, over the quasi-static EQ2 multipath Rayleigh fading channel model.

We first notice that the three systems appear to achieve an overall diversity order of 2. In fact, simulations at lower FER would be required to strictly confirm this assertion. In other words, the turbo-equalizer may only benefit from the diversity offered by the multipath channel. This is consistent with the observation made in section 2.1.4 that the coding schemes do not offer any diversity gain over quasi-static channels. Moreover, the three coded modulation schemes essentially present similar performance after 10 iterations over this channel model. In particular, the TCM offers the best trade-off between complexity and performance since the performance of the genie-aided equalizer is reached with only 3–4 iterations. In contrast, the performance of the two BICM-ID schemes remain several dB away from the genie receiver over the EQ2 channel, although a net gain is observed after 4–5 iterations with respect to the conventional disjoint approach (at the first iteration).

Two conclusions should be drawn from these results. First, codes optimized for the AWGN channel also perform well on a quasi-static channel when turbo-equalization is used at the receiver side. In addition, diversity techniques such as time-hopping or frequency-hopping, for example, are clearly required to improve the system performance on such severe channel models.

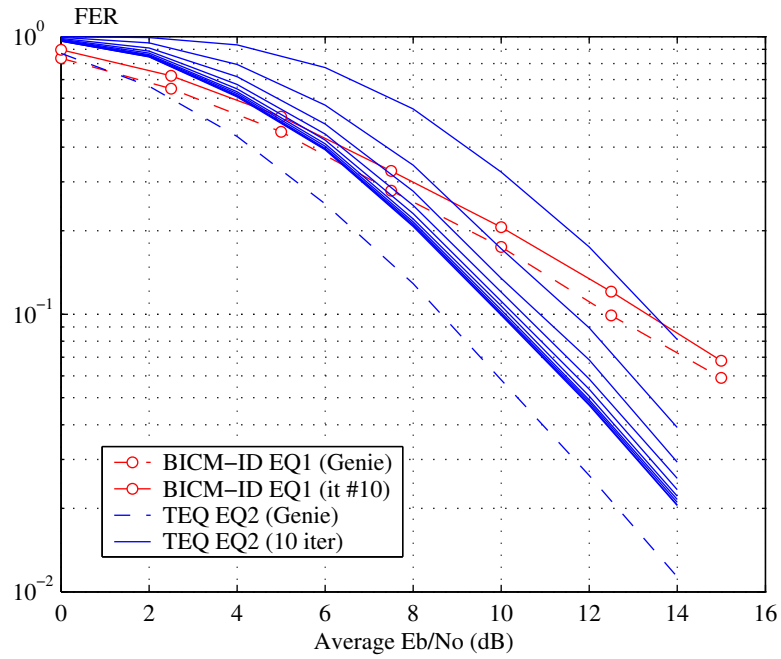


Figure 3.12: FER performance of the MAP turbo-equalizer for the 8-PSK BICM scheme, with 10 iterations, over the quasi-static EQ2 multipath Rayleigh fading channel model.

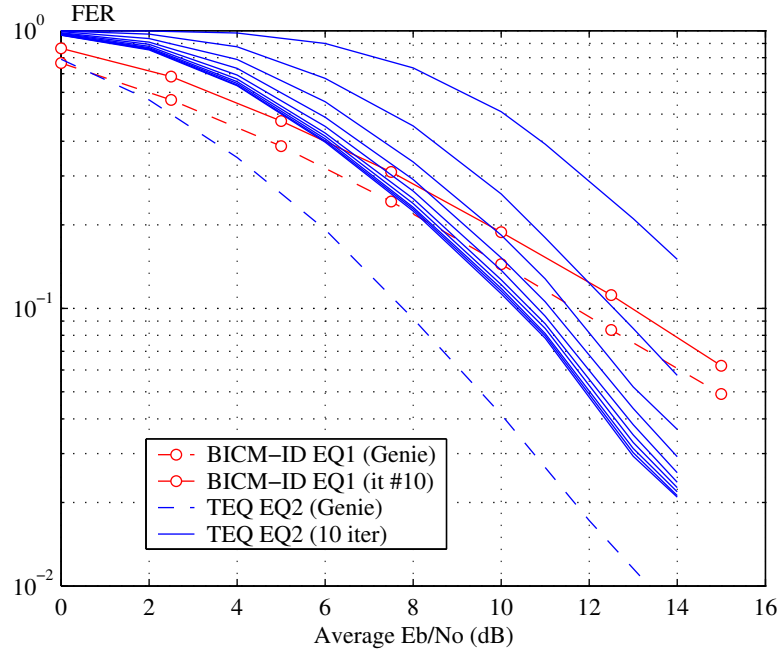


Figure 3.13: FER performance of the MAP turbo-equalizer for the 16-QAM BICM scheme, with 10 iterations, over the quasi-static EQ2 multipath Rayleigh fading channel model.

3.5 Convergence analysis using EXIT charts

From the simulation results exposed in the previous section, we notice that the performance of the turbo-equalization scheme is characterized by the existence of a minimum SNR value, the *convergence threshold*, beyond which successive iterations begin to bring improvements in the bit-error rate. The system then progressively converges towards the lower bounds developed in section 3.3 as the SNR increases, when a sufficient number of iterations is considered. Experience shows that the value of the threshold greatly depends on various parameters such as the code properties, the interleaver characteristics, the channel model, and the detection and decoding algorithms. It is thus of particular importance to understand the exact impact of these parameters on the convergence of the turbo-equalization scheme, so as to develop some general design guidelines in response.

Appropriate techniques have been recently developed to deal with such issues. An iterative decoding process (including turbo-equalization) can be viewed as a dynamical non-linear system with feedback. The basic idea behind the convergence analysis techniques involves studying the evolution of the extrinsic information messages as they evolve across the iterations, but from a probabilistic point of view since the component SISO decoders are by nature probabilistic devices. Moreover, to get rid of the interleaver influence and simplify the analysis, an ideal (infinite-length) interleaver is usually assumed. The most sophisticated technique is called *density evolution* and was in fact first suggested by Gallager in [70, p. 48]. Density evolution precisely tracks the probability density functions (densities) of the exchanged messages from iteration to iteration. Detailed presentations of density evolution are available in [142, 143] as well as in [136] and [184]. Albeit general and accurate in practice, density evolution requires significant computation efforts since one has to track densities, which are by definition multi-dimensional (theoretically infinite) quantities. Several simplified analyses have thus been suggested, which essentially propose tracking a single-dimensional parameter rather than an entire density from iteration to iteration (see e.g. [48, 54]).

In this section, we apply the *EXtrinsic Information Transfer* (EXIT) chart to the analysis of turbo-equalization schemes. Introduced by ten Brink in [161], the EXIT chart measures the evolution of mutual information across the iterative process, and constitutes a valuable and general graphical tool for convergence analysis and design of iterative decoding systems. Motivations behind our choice include the fact that mutual information is an accurate robust measure for predicting the convergence, as verified in [163, 176], which possesses interesting provable properties [16].

In the sequel, we first review the basic principles of the EXIT chart method. Then, as a tutorial introduction, we consider its application to the convergence analysis of the BICM-ID schemes exposed in chapter 2. The analysis is extended to the convergence study of MAP turbo-equalization of BICM over time-invariant channels. Finally, we examine the influence of the ISI channel and inner convolutional code characteristics on the convergence.

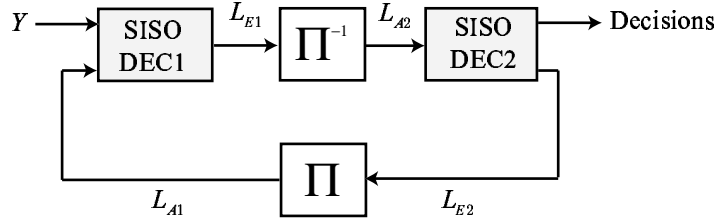


Figure 3.14: Model of an iterative decoder for a generic serially concatenated coding scheme.

For simplicity, we have restricted ourselves to bit-level iterative decoding schemes in this work, since EXIT charts were originally introduced in this context. We nevertheless believe that the method could be extended to the analysis of symbol-based turbo-equalization as well, using for example the techniques proposed in [151] or [78].

3.5.1 The EXIT chart technique

Consider a generic iterative decoder model for serially concatenated coding schemes as shown in figure 3.14. For convenience, we shall restrict ourselves to BCJR-MAP SISO detectors and decoders as exposed at the beginning of the chapter. Depending on the nature of the inner and outer SISO decoders (labelled SISO DEC1 and SISO DEC2 respectively), such a model may hold for serially concatenated turbo-codes, turbo-equalization (TCM or BICM), or iterative demapping and decoding (BICM-ID) as particular examples. For each iteration, the inner decoder takes channel observations Y and *a priori* LLRs L_{A1} , and outputs extrinsic LLRs L_{E1} which are then passed through the bit interleaver, to become *a priori* information A_2 for the outer decoder. The latter feeds back in turn extrinsic LLRs L_{E2} , which are reinterleaved and become new *a priori* information for the inner decoder.

In essence, the EXIT chart analysis proceeds as follows. First, mutual information is used to characterize the component decoders separately. We recall that mutual information between two random variables measures the average amount of information conveyed by the knowledge of one variable about the other, and reciprocally. Let I_{A1} denote the mutual information between the *a priori* LLRs L_{A1} and the transmitted coded bits at the input of the inner decoder. Furthermore, let I_{E1} denote the mutual information associated with the extrinsic LLRs L_{E1} at the decoder output. The action of inner decoder may then be described by a transfer function of the form

$$I_{E1} = T_1(I_{A1}, \text{channel}) \quad (3.51)$$

where we have made explicit the fact that the output mutual information I_{E1} is necessarily a function of channel-dependent parameters (signal-to-noise ratio, ISI channel model, etc.). Similarly, the action of the outer decoder may be described by a second transfer function, depending solely on the average

mutual information at the decoder input (the outer decoder is not connected to the channel)

$$I_{E_2} = T_2(I_{A_2}) \quad (3.52)$$

Such functions are called the *extrinsic information transfer characteristics* of the component decoders. The EXIT chart is obtained in a second step by superposing these two characteristics in the same graph, as shown in figure 3.15. Note that since $I_{A_1} = I_{E_2}$ and $I_{A_2} = I_{E_1}$, the abscissas and ordinates of the outer decoder transfer function have to be swapped in the EXIT chart. The exchange of extrinsic information is then visualized as a “zig-zag” decoding trajectory in the EXIT chart. In particular, the iterative process reaches a *saturation point* (or *fixed point*) when the two characteristics intersect, since in this case, no innovation can be gained from either one of the component decoders. This allows the prediction of the convergence threshold of the system, under the assumption of ideal interleaving.

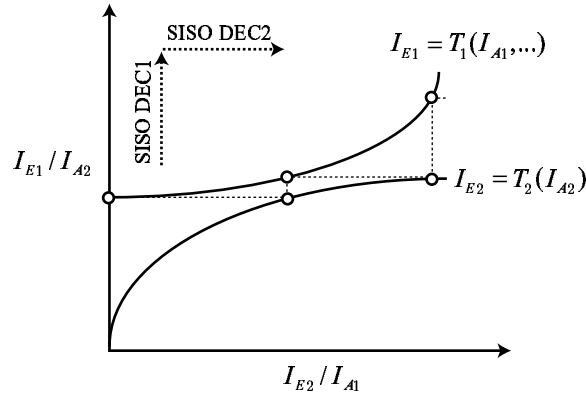


Figure 3.15: Typical diagram of an EXIT chart. The dashed lines figure the trajectory of the iterative decoding process, assuming infinite-length interleaving.

We now give general expressions for mutual information in the binary case. We denote by C the transmitted coded bits, and by L the corresponding (*a priori* or extrinsic) binary LLRs. For convenience, we assume that the coded bits take their value in $\{-1, +1\}$ rather than in $\{0, 1\}$, according to the mapping rule $0 \mapsto -1$ and $1 \mapsto +1$. Then, assuming that the transmitted code bits are i.i.d., it is known that (see e.g. [71])

$$I(L, C) = \frac{1}{2} \sum_{C=\pm 1} \int_{-\infty}^{+\infty} P(L|C) \log_2 \left[\frac{P(L|C)}{(1/2)[P(L|C=-1) + P(L|C=+1)]} \right] dL \quad (3.53)$$

For most practical channels as well as for BCJR-MAP decoders, the conditional densities $P(L|C)$ are symmetric with respect to 0, i.e. $P(L|C=+1) = P(-L|C=-1)$. This yields the following simpler

expression for the mutual information

$$I(L, C) = 1 - \int_{-\infty}^{+\infty} P(L|C = -1) \log_2 \left[1 + \frac{P(L|C = +1)}{P(L|C = -1)} \right] dL \quad (3.54)$$

$$= 1 - E_{L|-1} \left\{ \log_2 \left[1 + \frac{P(L|C = +1)}{P(L|C = -1)} \right] \right\} \quad (3.55)$$

Furthermore, the following inequalities are satisfied [71]

$$0 \leq I(L, C) \leq H(C) = 1 \quad (3.56)$$

Mutual information of 0 means that the random variables L and C are independent. Consequently, the knowledge of L does not bring any information about the value of C . Conversely, L perfectly identifies C in a deterministic manner when $I(L, C) = 1$.

Analytical expressions for the transfer functions T_1 and T_2 have not been found yet, except for some very simple codes. Consequently, these characteristics have to be evaluated numerically for each component decoder. We now present practical methods that can be used to obtain the transfer characteristics T_1 and T_2 .

We begin by describing the computation of the mutual information I_A in the input of the SISO modules. From equation (3.55), we notice that the knowledge of the conditional density $P(L|C = -1)$ is in fact required. This is problematic since we do not have exact analytical expressions for this density. However, experimental evidence suggests that the extrinsic LLRs exchanged across the iterative process are well approximated by independent and identically distributed Gaussian random variables [48, 106, 193]. Moreover, under the assumption of large interleaving, it is usual to assume that under certain mild hypothesis about the channel model, the density of extrinsic information is asymptotically *consistent*, i.e. it satisfies an exponential symmetry condition which translates into (see [142])

$$\frac{P(L|C = +1)}{P(L|C = -1)} = \exp[L] \quad (3.57)$$

Application of the consistency condition to a Gaussian density with parameters $\mathcal{N}(\pm\mu, \sigma^2)$ given the knowledge of a binary random variable in $\{\pm 1\}$ yields the result $\sigma^2 = 2|\mu|$ (the variance is twice the mean). Hence, the conditional density $P(L|C = -1)$ can be conveniently modeled by a Gaussian distribution with mean $-\mu_A$ and variance $2\mu_A$. The corresponding mutual information in input I_A finally reads

$$I_A = J(\mu_A) = 1 - E_{L_A|-1} \{ \log_2(1 + \exp[L_A]) \} \quad (3.58)$$

where expectation is taken over the conditional density

$$P(L_A|C = -1) = \frac{1}{2\sqrt{\pi\mu_A}} \exp - \left[\frac{(L_A + \mu_A)^2}{4\mu_A} \right] \quad (3.59)$$

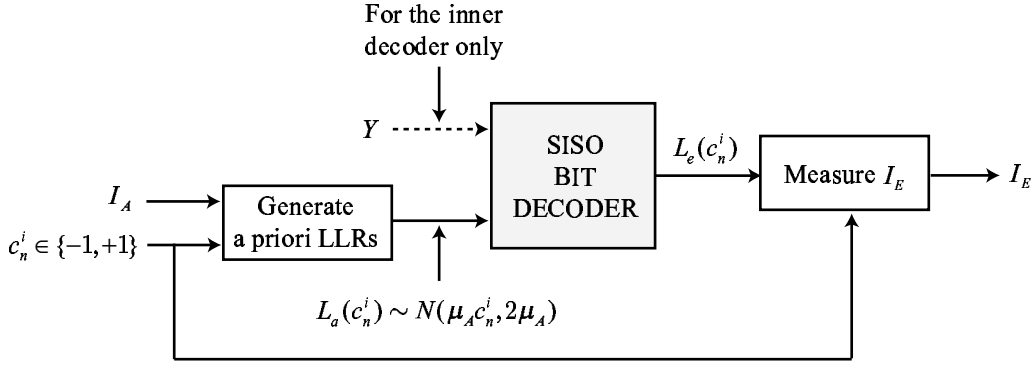


Figure 3.16: Graphical illustration of the method used to obtain the transfer characteristics of the component decoders in the bit-level case.

The resulting function $J(\mu_A)$ is easily evaluated by numerical integration. It is monotonically increasing and thus reversible, with limits

$$\lim_{\mu_A \rightarrow 0} J(\mu_A) = 0, \quad \lim_{\mu_A \rightarrow \infty} J(\mu_A) = 1 \quad (3.60)$$

Now, to calculate the transfer function $I_E = T(I_A, \text{channel})$ in response to an input mutual information value of I_A (in combination with other channel parameters for the inner decoder), one simply has to apply independent Gaussian random variables with parameter $\mu_A = J^{-1}(I_A)$ at the decoder input, conditionally on the knowledge of the transmitted coded bits, and then measure the mutual information I_E obtained in response. By repeating the previous procedure for different values of I_A , one finally obtains an estimate of the extrinsic information characteristics T_1 and T_2 . The method is illustrated in figure 3.16.

In practice, I_E can be accurately computed from expression (3.55), using numerical integration techniques and an estimate of the conditional densities $P(L|C)$ at the decoder output. The latter estimates are typically from histograms. A simpler approach was however suggested in [81]. Let us again apply the consistency condition (3.57) to expression (3.55). We obtain

$$I_E = 1 - E_{L_E|=-1} \{ \log_2(1 + \exp[L_E]) \} \quad (3.61)$$

Hence, by invoking the ergodic assumption, I_E can be determined experimentally from N realizations of $(c_n^i, L_e(c_n^i))$ using the sample mean estimator and applying a correction for positive c_n^i , yielding

$$I_E \approx 1 - \frac{1}{N} \sum_{n=0}^{N-1} \log_2(1 + \exp[-c_n^i L_e(c_n^i)]) \quad (3.62)$$

In such a way, it is not necessary to determine the density of the output extrinsic LLRs, which considerably reduces the computational cost of the analysis.

3.5.2 EXIT chart analysis for BICM-ID

Having described the fundamentals of EXIT charts, we first apply this technique to the convergence analysis of the two BICM-ID schemes introduced in section 2.1.4.

Figure 3.17 shows the transfer characteristics of the SISO demapper for an 8-PSK signal set (labelled DEM), as a function of the channel SNR and signal labelling map (Set-Partitioning, labelled SP, or Gray labelling). We observe that the curves are almost horizontal for Gray labelling. This confirms that Gray labelling does not benefit from the iterative process. In contrast, the mutual information in the output progressively increases for the set-partitioning labelling map, which suggests intuitively that set-partitioning may lead to better asymptotic performance than standard Gray labelling. We mention that similar results have been observed with a 16-QAM signal set (not reported here).

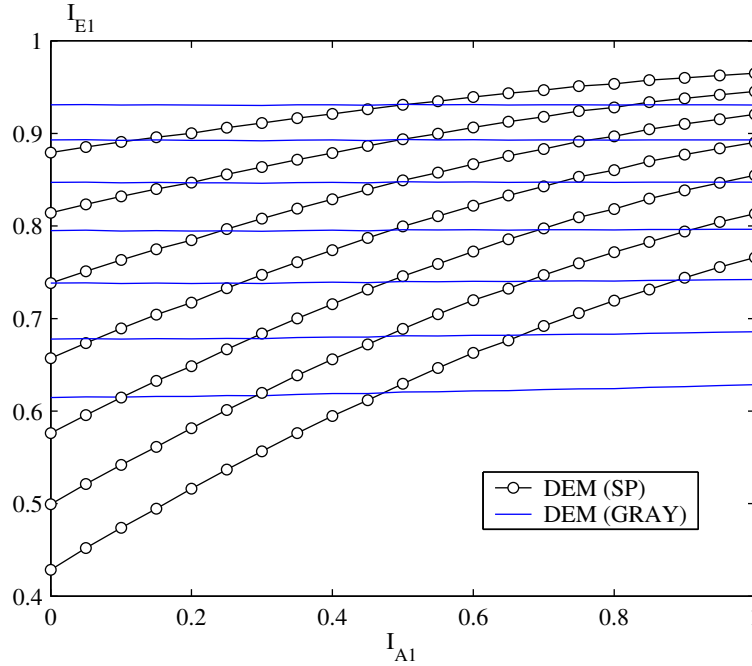


Figure 3.17: Transfer characteristics of the SISO demapper for different signal labelling maps. The characteristics are presented for E_b/N_0 values ranging from 2 dB (bottom) to 8 dB (top) in steps of 1 dB.

The EXIT charts for the 8-PSK and 16-QAM BICM-ID schemes are shown in figures 3.19 and 3.20. The demapper characteristics are presented for different values of the channel SNR. We have also figured as dashed lines the transfer characteristics obtained with a genie-aided demapper, with perfect feedback from the decoder. These curves are in fact horizontal, since the transfer functions then only depend on the *a priori* LLR on the current coded bit, information which is subtracted at the demapper output to produce the extrinsic LLRs.

A few definitions are in order before commenting the EXIT charts. We define a *saturation point* of

the system as the intersection of the inner (the SISO demapper here) and outer decoders characteristics. Convergence is then measured by the mutual information gap (taken at the demapper output / decoder input) between the saturation point obtained with the genie-aided demapper, and the saturation point reached by the iterative process at a given channel SNR value. *We say that convergence occurs when the two saturation points come close enough together that they progressively become indistinguishable in the EXIT chart.* In our context here, this is tantamount to saying that the iterative decoding scheme has essentially reached the ideal performance obtained with a genie-aided demapper. This concept of mutual information gap and convergence is illustrated in figure 3.18 at an SNR value $E_b/N_0 = 2$ dB.

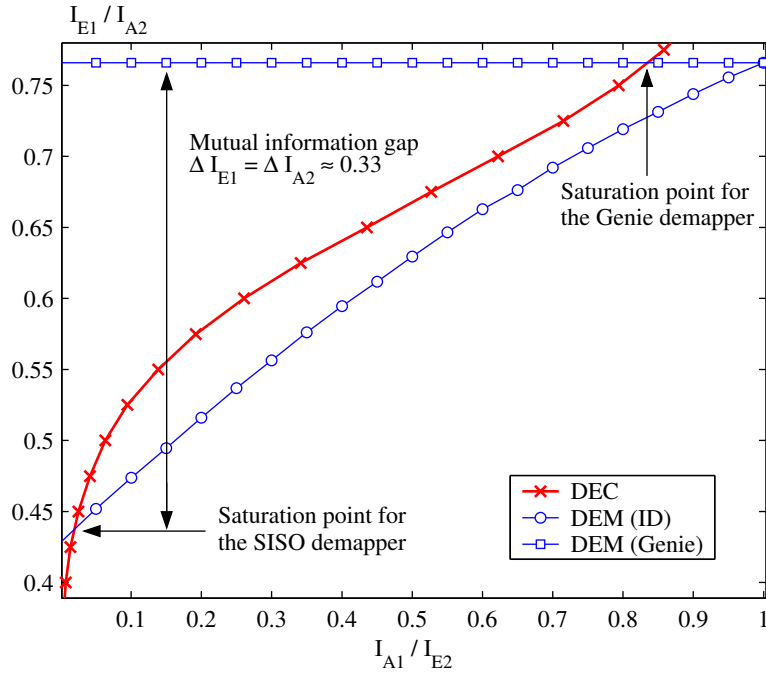


Figure 3.18: Zoom of figure 3.19 at $E_b/N_0 = 2$ dB in order to illustrate the concept of mutual information gap with respect to the performance of the genie-aided demapper.

Inspection of the EXIT charts of figures 3.19 and 3.20 reveals that once the channel reaches a given minimum SNR value (of about 3 dB for the 8-PSK scheme, and between 3 and 4 dB in the 16-QAM case), a tunnel is progressively opened between the transfer characteristics of the demapper and decoder, which allows for a progressive convergence of the iterative process towards the performance of the genie-aided demapper. This phenomenon precisely characterizes the *convergence threshold* of the system. Regarding now the simulation results presented in section 2.1.4 (figure 2.8), we obtained convergence thresholds of 4 dB, both for the 8-PSK and 16-QAM BICM-ID schemes. These values do not match the predictions of the EXIT chart exactly, but one has to recall that EXIT chart analysis only holds rigorously in the presence of large interleaving.

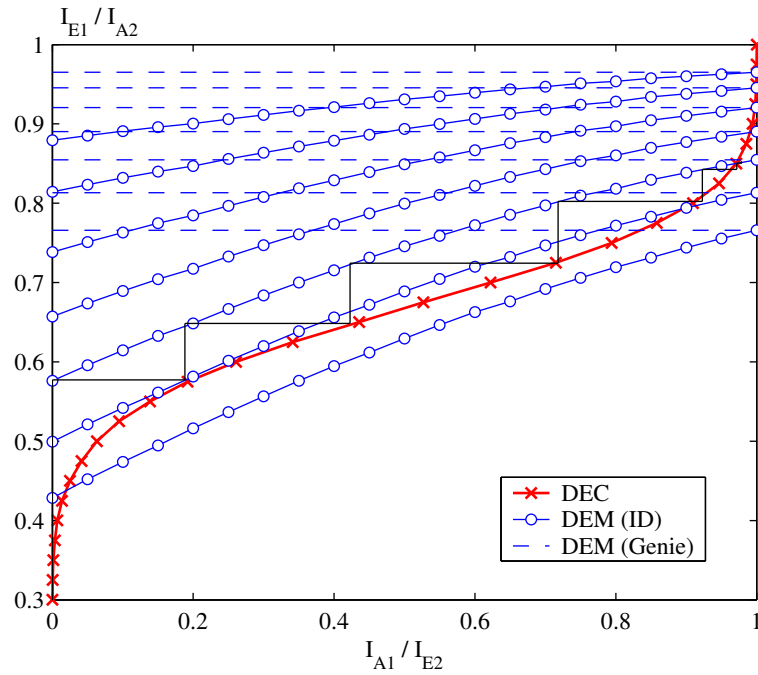


Figure 3.19: EXIT chart for the 8-PSK BICM-ID scheme. The demapper characteristics are presented for E_b/N_0 values ranging from 2 dB (bottom) to 8 dB (top) in steps of 1 dB. The experimental trajectory of the iterative decoding process is also figured for $E_b/N_0 = 4$ dB.

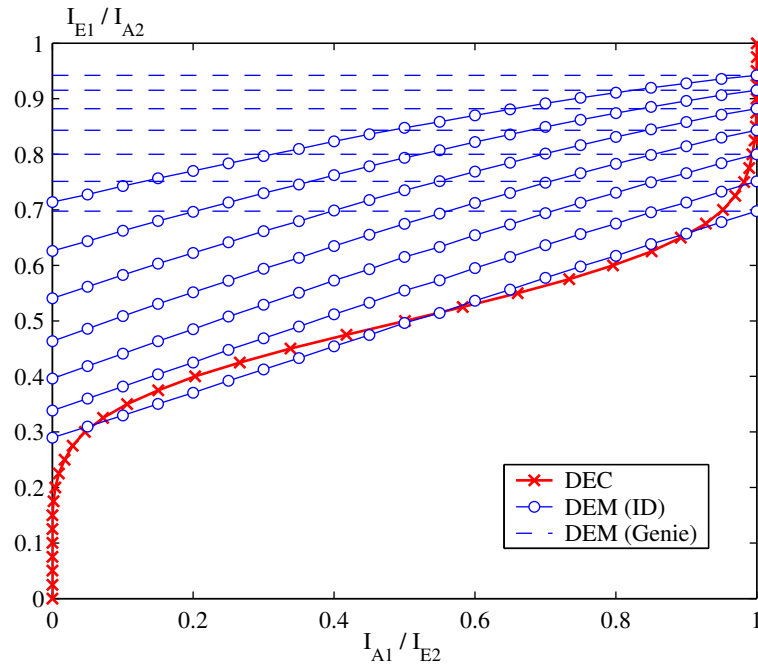


Figure 3.20: EXIT chart for the 16-QAM BICM-ID scheme. The demapper characteristics are presented for E_b/N_0 values ranging from 2 dB (bottom) to 8 dB (top) in steps of 1 dB.

In order to further verify the accuracy of the EXIT chart method, we simulated the *whole* iterative decoding system for the 8-PSK BICM-ID scheme, with an interleaver size of $2 \cdot 10^5$ coded bits and at an SNR value $E_b/N_0 = 4$ dB, and measured the mutual information at the output of the SISO modules at each iteration. We obtained an experimental trajectory for the iterative process that has been reported in figure 3.19. We observe that the experimental trajectory closely follows the convergence behavior predicted by the EXIT chart. This confirms the accuracy of the EXIT chart analysis when large interleavers are used.

In addition to offering a pertinent graphical interpretation of the convergence behavior (obtained at a low computational cost), the EXIT chart may also be used to predict the BER that could be obtained under the assumption of ideal interleaving. It suffices to introduce a new transfer characteristic for the outer SISO decoder, giving the BER as a function of the mutual information I_{A2} in input. Such characteristics are shown in figure 3.21 for the convolutional codes used in the two BICM-ID schemes considered. Then, we simply have to transfer the ordinate of the saturation point obtained in the EXIT chart to the abscissa of the BER characteristics, and we obtain the corresponding BER. As an illustration, consider the EXIT chart characteristic of the 8-PSK BICM-ID scheme in figure 3.19 at an SNR value of 4 dB. The saturation point occurs for a value $I_{A2} \approx 0.86$, which when transferred to the graph in figure 3.21 gives us a BER value of about 10^{-3} . This is consistent with the simulation results of figure 2.8.

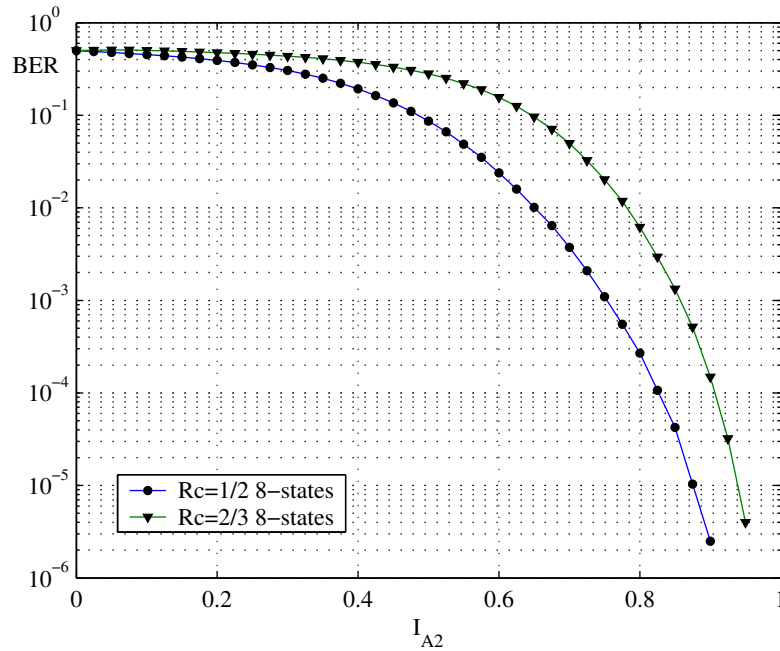


Figure 3.21: BER characteristics for the two convolutional codes considered in the BICM schemes.

3.5.3 EXIT chart analysis of MAP turbo-equalization for BICM

We now apply the EXIT chart technique to the analysis of MAP turbo-equalization for BICM-ID. Specifically, we focus on the transmission scenario introduced in section 3.4.1 over the severe-ISI Proakis B channel model. The corresponding EXIT charts obtained with the two BICM schemes over this static channel are shown in figures 3.22 and 3.23 respectively. The transfer characteristics of the SISO MAP equalizer (labelled by EQ) are presented for different values of the channel SNR. We have also included the curves obtained with a genie-aided demapper over an AWGN channel for reference purposes (dashed lines). These curves correspond to the lower bound established in section 3.3.

By inspection of the plots, we observe that convergence theoretically occurs between 6 and 7 dB for the 8-PSK BICM-ID, and around 8 dB for the 16-QAM BICM-ID. Moreover, once the convergence threshold is reached, the turbo-equalizer is expected to converge quickly (i.e. with few iterations) towards the performance of the genie-aided demapper in both situations. Interestingly, this analysis demonstrates again the ability of the turbo-equalizer to effectively achieve in practice the theoretical lower bounds established previously (at least under the assumption of large interleaving), once the channel SNR is high enough. Note however that the simulation results obtained in section 3.4.1 show that the convergence occurs in fact at SNR values about 1.0 dB higher than those predicted by the EXIT chart. This mainly stems from the moderate interleaver size (8196 coded bits) considered in our simulations. Finally, the system BER can be predicted quite accurately using the BER transfer functions of figure 3.21, using the method described in the previous subsection.

On the basis of these results, it is clear that the shape of the transfer characteristics of the inner and outer decoder has a strong influence on the convergence behavior of the iterative process. Hence, it seems interesting to study the impact of some particular system parameters on these transfer functions. More precisely, we shall now consider the influence of the channel characteristics and outer convolutional code parameters on the convergence behavior of the turbo-equalization scheme².

²Parts of this study were presented at the 18th GRETSI Symposium [109].

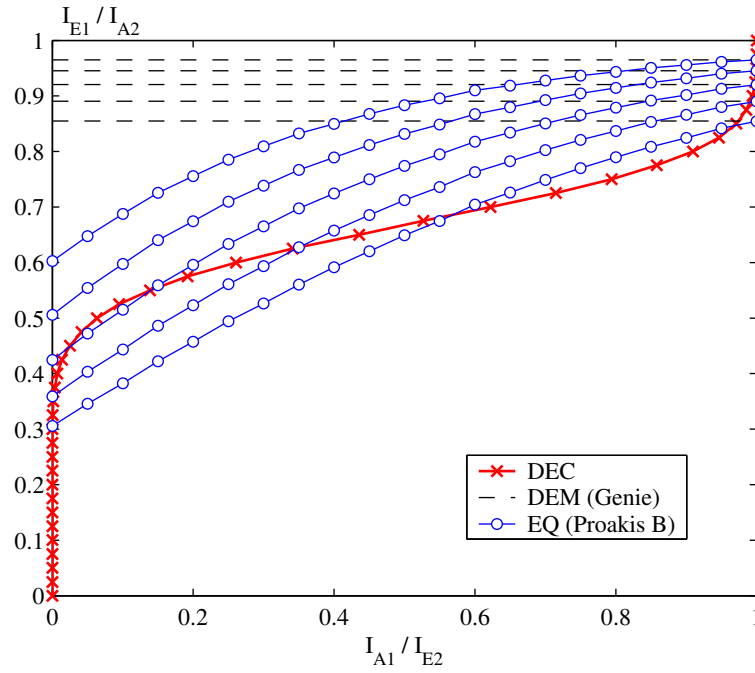


Figure 3.22: EXIT chart for MAP turbo-equalization of the 8-PSK BICM-ID scheme over the Proakis B channel model. The equalizer characteristics are presented for E_b/N_0 values ranging from 4 dB (bottom) to 8 dB (top) in steps of 1 dB.

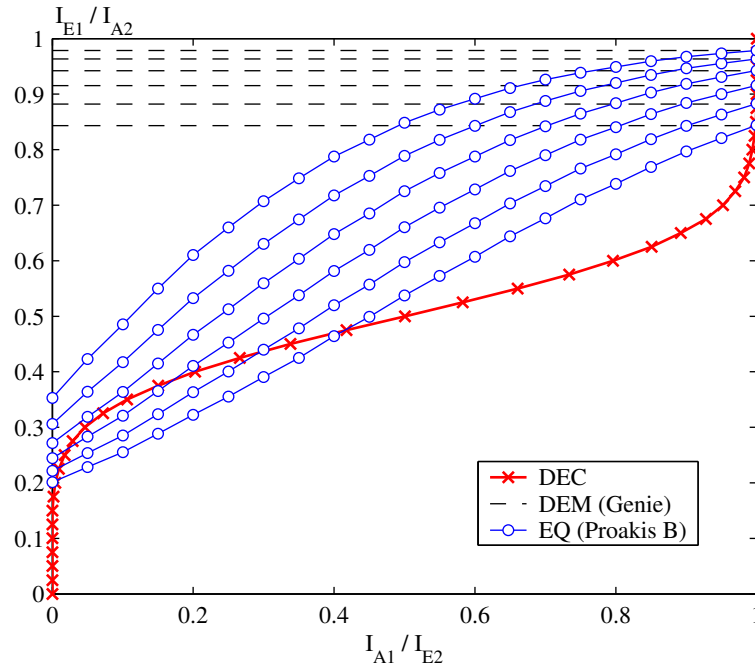


Figure 3.23: EXIT chart for MAP turbo-equalization of the 16-QAM BICM-ID scheme over the Proakis B channel model. The equalizer characteristics are presented for E_b/N_0 values ranging from 5 dB (bottom) to 10 dB (top) in steps of 1 dB.

3.5.4 Influence of the channel characteristics on the convergence

Inspecting the EXIT charts in the previous subsection, we immediately notice that as the channel SNR increases, the transfer characteristics of the MAP equalizer are shifted upwards. Hence, the iterative process progressively requires fewer iterations to reach the optimal performance, and the convergence is faster. In fact, the transfer functions approach the straight line $I_{E1} = 1$ in the limit of infinite SNR. This is intuitively satisfying since in such an ideal situation, the presence of prior information about the transmitted message becomes irrelevant to the decision.

We now investigate the influence of the severity of the ISI channel on the equalizer transfer function. For convenience, we shall restrict ourselves to an 8-PSK signal set. The extrinsic information transfer characteristic for the MAP equalizer has been computed for several time-invariant ISI channel models, including

- the AWGN channel
- the Proakis B channel model
- the CRIT3 channel model, with impulse response $[0.5, 0.71, 0.5]$, that was introduced in [138, chap. 10] as a worst-case 3-taps ISI model for trellis-based equalizers
- the EQUI2 and EQUI3 models, consisting of 2 and 3 taps of equal magnitude respectively
- an almost ISI-free channel with impulse response $[1.0, 0.3, 0.2]$, that we called TEST3

BER simulations of the SISO equalizer over these various ISI channels models revealed that they may be classified as follows, in order of increasing severity

$$\text{AWGN} < \text{TEST3} < \text{EQUI2} < \text{EQUI3} < \text{PROAKIS B} < \text{CRIT3}$$

The resulting transfer characteristics for the equalizer are shown in figure 3.24. These curves were obtained at $E_b/N_0 = 5$ dB. We observe that the more severe the channel is, the lower is the mutual information at the equalizer output in the absence of particular prior information ($I_{A1} = 0$), and thus the higher the convergence threshold will be. However, this also implies that the slopes of the characteristics increase with the severity of the ISI channel. Clearly, in the neighborhood of the convergence threshold, a transfer function with a steep slope is more likely to require fewer iterations to reach the optimal performance than a transfer function which approaches a straight line,

To summarize, we conclude that a channel with small or moderate ISI will have a low convergence threshold, but will also offer small improvements across the iterative process. Conversely, a channel

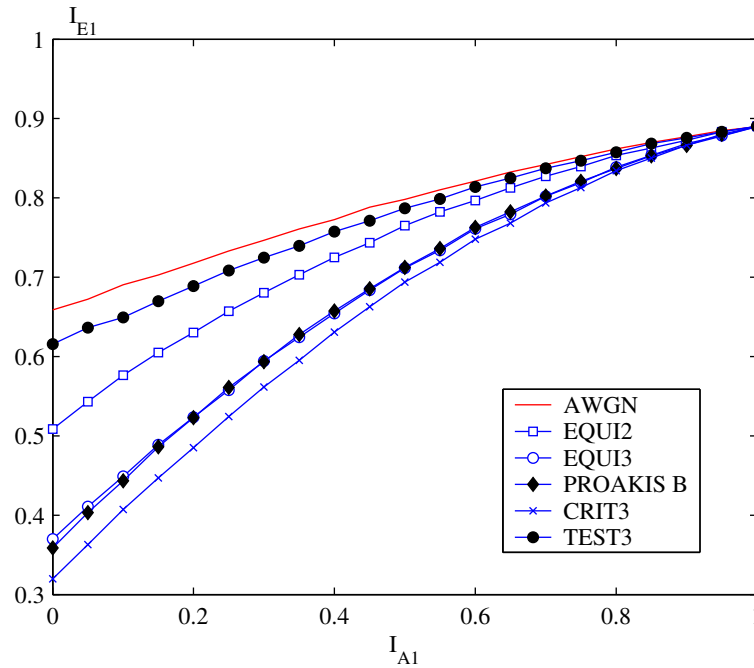


Figure 3.24: Transfer characteristics of the SISO MAP equalizer for different ISI channel models, at an SNR value $E_b/N_0 = 5$ dB.

with severe ISI will require higher SNR values to converge, but is expected to offer significant performance improvements with turbo-equalization once the convergence threshold is reached. We mention that similar conclusions were obtained in [162].

3.5.5 Influence of the inner code characteristics on the convergence

We finally investigate the influence of the parameters of the outer convolutional code on the shape of the extrinsic information characteristic of the outer SISO decoder. For the sake of conciseness, we only consider rate $R_c = 1/2$ convolutional codes. However, our conclusions apply to other code rates as well.

Figure 3.25 shows the extrinsic information transfer characteristics of the outer decoder for some rate $1/2$ maximum free distance convolutional codes with different memory values. It clearly appears that the code memory has a strong influence on the shape of the characteristics. The two extremes are obtained with the simple repetition code, for which $I_{E2} = I_{A2}$, and a convolutional code with infinite memory respectively [163]. For the latter code, the extrinsic transfer characteristic tends to approach

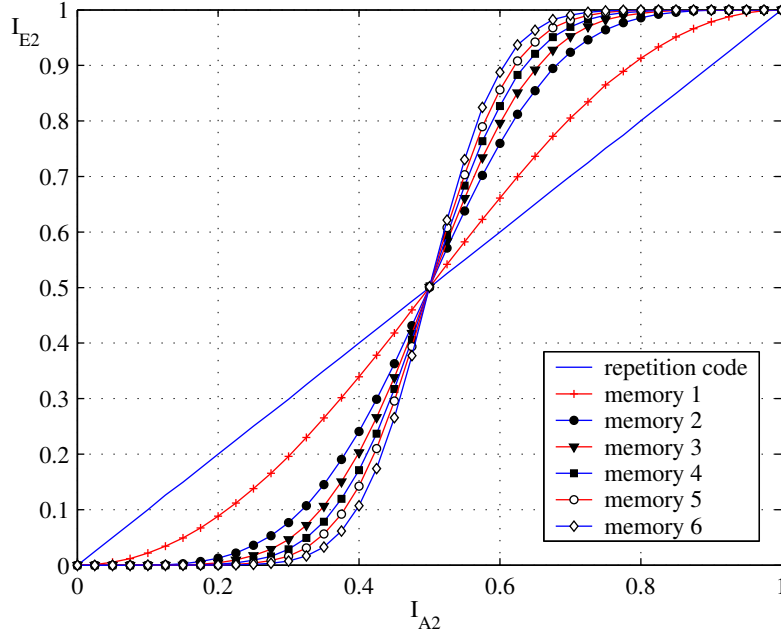


Figure 3.25: Extrinsic information transfer characteristics of the APP outer bit-level decoder for some rate 1/2 maximum free distance convolutional codes with different memories.

a step function

$$I_{E2} = \begin{cases} 0 & 0 \leq I_{A2} < 0.5 \\ 0.5 & I_{A2} = 0.5 \\ 1 & 0.5 < I_{A2} \leq 1 \end{cases} \quad (3.63)$$

As shown in figures 3.26 and 3.27 however, it appears that using different generator polynomials, or switching from a non-recursive non-systematic (NRNS) encoder to a recursive systematic (RS) encoder has little influence on the shape of the characteristics³. Thus, from this analysis, it becomes evident that the code memory is the single most important parameter of the outer code, together with the code rate, from a convergence behavior point of view.

An additional important observation is in order here. The use of convolutional codes with a high memory is required to improve the asymptotic performance of the overall iterative scheme, at least for BICM-ID and turbo-equalization. However, the shape of the transfer functions in figure 3.25 suggests that codes with a small memory are very likely to have a smaller convergence threshold than codes with a high memory, regardless of the shape of the characteristic for the inner SISO decoder. Hence, there is a trade-off to find between good asymptotic performance and early convergence of the iterative process. Such a phenomenon was mentioned briefly in chapter 2 when discussing BICM-ID. More generally, this may be a symptomatic dilemma of every iterative decoding scheme relying on the Turbo principle. As an illustration, we refer the reader to the discussion about convergence issues

³These modifications may however have a significant impact on the BER performance of the system.

versus minimum distance for turbo-codes in [25].

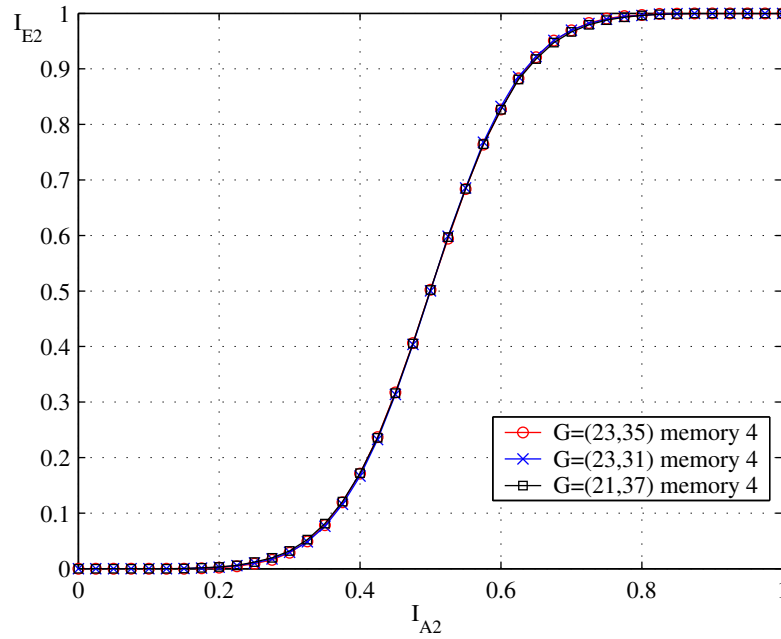


Figure 3.26: Influence of the generator polynomials on the shape of extrinsic information transfer characteristic for the APP outer bit-level decoder.

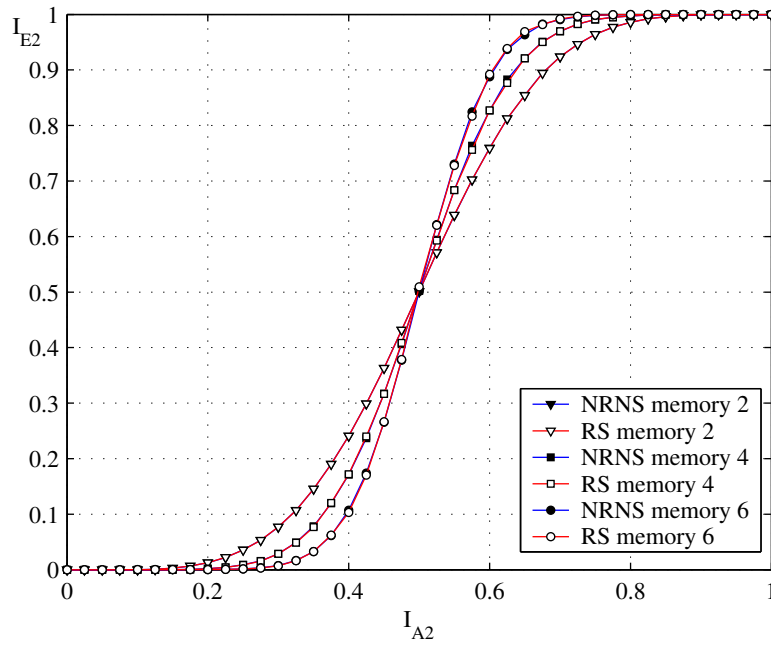


Figure 3.27: Influence of the encoder structure (Non-Recursive Non-Systematic or Recursive Systematic) on the shape of extrinsic information transfer characteristic for the APP outer bit-level decoder.

3.6 Concluding remarks

This chapter was devoted to the problem of combining equalization and decoding in an efficient manner at the receiver side, in the presence of interleaving. Traditional systems usually address this issue by performing one-time equalization and decoding. However, such a strategy is sub-optimal from a system performance point of view. Turbo-equalization offers an alternative approach, where an iterative exchange of soft information is established between the equalizer and the decoder. It is expected that as the SNR increases, the performance of the overall iterative scheme will converge towards the performance of an ideal receiver taking simultaneously into account the constraints imposed by the coding scheme, the interleaver and the ISI channel.

The turbo-equalization scheme has been introduced both for TCM and BICM transmissions. Lower-bounds on the performance of the turbo-equalizer have been established assuming large enough interleaving and ideal feedback from the decoder. On the basis of these assumptions, we have proved that the turbo-equalizer essentially converts the ISI channel into an equivalent ISI-free channel, thereby achieving the ideal matched-filter bound. For the particular case of a fully-interleaved multipath Rayleigh fading channel, we have shown that the turbo-equalizer is able to exploit the diversity offered both by the multipath propagation and the inner coding scheme.

Simulation results have been presented for different transmission scenarios. We have verified that the turbo-equalization scheme provides significant performance gains over the conventional approach, and effectively reaches in practice the lower bounds established previously at high enough SNR. In order to gain more insight into the convergence behavior of the iterative process, the EXIT chart tool has been introduced and applied to the analysis of the turbo-equalization scheme. We have examined the influence of the channel and outer convolutional code characteristics, and observed in particular that there exists a trade-off to find between good asymptotic performance and an early convergence of the turbo-equalization process.

In view of these remarks, turbo-equalization may appear as an attractive receiver in the context of bandwidth-efficient transmissions over ISI channels. However, it was assumed in this chapter that both the equalizer and decoder are implemented using the BJCR-MAP algorithm. While this algorithm is optimal in the sense of minimizing the symbol error probability, its complexity becomes rapidly untractable as the number of channel taps increases, especially when high-order modulations are considered (which is typical of bandwidth-efficient transmission schemes). Hence, the desire to transmit at higher data-rates over ISI channels calls for the introduction of reduced-complexity turbo-equalization schemes. This is the subject of the next chapter.

Chapter 4

Low-complexity efficient MMSE Turbo-Equalizers

The SISO equalizers presented in the previous chapter have a computational complexity which increases exponentially as a function of the dimension of the signal set and the length of the discrete-time channel impulse response. This precludes their practical use in broadband wireless transmission systems, where multilevel signalling is usually required and where long-delay spread ISI channels may be encountered. Research efforts have thus been devoted to the design of reduced-complexity SISO equalizers conserving performance as close as possible to the optimal BCJR-MAP equalizer, in view of their inclusion in a turbo-equalization scheme.

The proposed solutions usually fall into two main classes, whether they rely on reduced-states trellis-based algorithms, or filtering-based equalizers. The two approaches essentially explore the issue of accommodating classical equalization structures, as exposed in chapter 2, with the support of reliability information, i.e. soft inputs and soft outputs. Among the class of trellis-based equalizers, we mention the notable contributions of Penther [135], Berthet and Visoz [28, 29, 185], Colavolpe *et al* [43], and Fragouli *et al* [67], who each extended different reduced-complexity MLSD algorithms to the SISO case. We shall however be concerned in this chapter with the alternative solution offered by filtering-based equalizers, optimized according the minimum-mean square error criterion.

Glavieux *et al* were the first to propose the use of MMSE equalizers in a turbo-equalization scheme [75, 101, 102]. They considered an MMSE interference cancellation structure supplied with soft estimates on the transmitted data computed from the soft decisions delivered by the decoder. The equalizer coefficients were updated in adaptive way using the standard *least-mean square* (LMS) algorithm. Interestingly, it was observed that this adaptive configuration in fact approached a classical MMSE linear equalizer at the first iteration, and then progressively converged in an adaptive manner towards an ideal MMSE interference canceller as the number of iterations and the SNR increased,

with large enough interleavers. Adaptive MMSE turbo-equalization has been extensively studied by Langlais in [99]. In particular, it has been successfully applied to High-Frequency (HF) transmissions over severe-ISI ionospheric channels (see also [100]). An improved realization of the adaptive turbo-equalizer was recently proposed in [88] where a blind equalizer is used at the first iteration in order to improve the spectral efficiency of the transmission by suppressing the need for a training sequence.

A significant breakthrough occurred in the field with the work of Tüchler *et al.* Building upon previous work of Wang and Poor [189] (see also [141]), Tüchler obtained a closed-form expression for a time-varying MMSE linear equalizer where the *a priori* knowledge available from the decoder is explicitly taken into account into the calculation of the filter coefficients [172, 174, 175]. In particular, Tüchler formally proved that the resulting equalizer was respectively equivalent to a classical MMSE linear equalizer in the absence of particular knowledge about the transmitted symbols, and to an ideal MMSE interference canceller in the presence of perfect *a priori* information. This equalizer was successfully integrated in a turbo-equalizer in [175]. The extension to higher-level signal sets was studied in more details by Dejonghe and Vandendorpe in [45] and [46].

Tüchler restricted his attention to finite-length equalizers. Capitalizing on his results and on previous work from Chan and Wornell [35, 36], Laot *et al* obtained an MMSE interference canceller in infinite-length form, termed MMSE IC-LE¹ [103], which presents a self-reconfigurable structure switching progressively from a classical MMSE linear equalizer to an ideal MMSE interference canceller as the reliability of the data estimates increases. Interestingly, it was found later that under finite-length realization constraints, the proposed equalizer in fact reduces to the low-complexity approximate implementation of the previous time-varying equalizer, that was suggested in a heuristic manner by Tüchler in [175, section III.B]. Our approach has, nonetheless, the merit of providing a rigorous derivation for this solution.

As we shall see in the sequel, these equalizers generally achieve good performance, sometimes falling very close to the optimum BCJR-MAP equalizer. Moreover, they maintain a reasonable complexity, growing almost independently of the dimension of the signal set and linearly with the length of the channel impulse response. Finally, they lend themselves readily to an adaptive implementation in order to track eventual channel variations. Hence, this class of equalizers constitutes an attractive candidate for broadband transmission over long-delay spread channels, especially when spectrally-efficient channel coding schemes are considered at the transmitter side.

This chapter is organized as follows. We first examine the general structure shared by the SISO MMSE equalizers considered in this dissertation. Then, we review the time-varying finite-length MMSE equalizer proposed by Tüchler, as well as its low-complexity approximate implementation. Building upon these results but considering the problem from a different perspective, we introduce the MMSE IC-LE which constitutes our major contribution in this work. This equalizer is derived

¹The acronym IC-LE stands for *Interference Canceller - Linear Equalizer*.

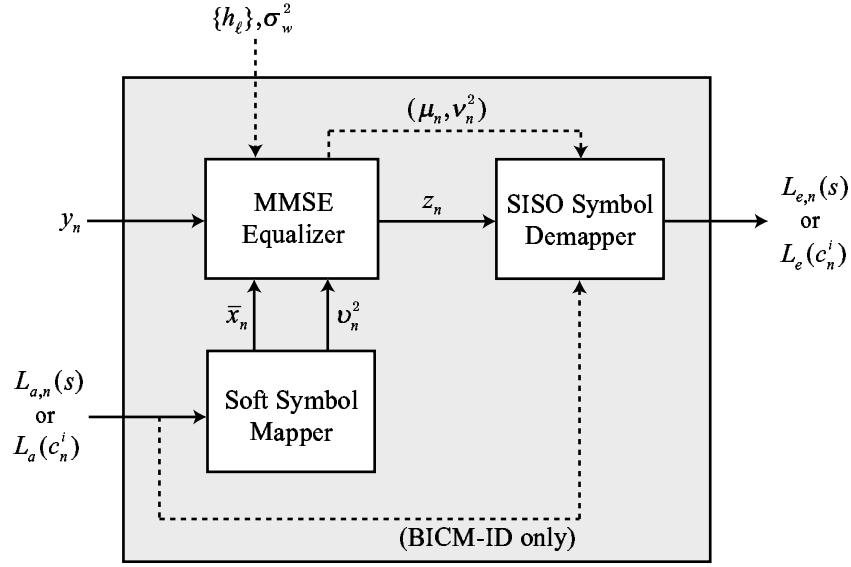


Figure 4.1: Block diagram of a generic SISO MMSE equalizer.

both in infinite-length and finite-length form. The equivalence of the latter with the above-mentioned time-invariant equalizer is established. A low-complexity method for computing the equalizer coefficients is suggested and analyzed. We briefly examine the asymptotic performance obtained with MMSE turbo-equalization schemes, and show that they achieve the same lower bounds as their MAP counterpart under the assumption of perfect *a priori* information if the inner MMSE equalizer is properly designed. Several simulation results are presented over different transmission scenarios. It is shown that the MMSE IC-LE realizes an interesting tradeoff between complexity and performance over long delay spread channels with low to moderate ISI. Finally, we derive the frequency-domain MMSE IC-LE turbo-equalizer and take a close look at the complexity savings that may result in comparison with a time-domain turbo-equalizer, over channels with very long impulse responses.

4.1 The general structure of SISO MMSE equalizers

The general structure shared by the SISO MMSE equalizers considered in this thesis is depicted in figure 4.1.

On the basis of the *a priori* symbols LLRs (for TCM) or bits LLRs (for BICM) available at its input, a soft symbol mapper first computes soft symbol estimates $\bar{x}_n$ on the transmitted data sequence, as well as a reliability measure denoted v_n^2 for each of these estimates. These quantities are sent to an MMSE equalizer, which exploits this *a priori* information together with the channel observations y_n to produce an estimate z_n of a transmitted symbol $x_{n-\Delta}$, assuming that the equalizer introduces

an overall restitution delay Δ . Finally, given the knowledge of the equalizer characteristics, a SISO demapper calculates updated extrinsic LLRs from the equalized sample z_n . Note that for the particular case of BICM-ID with non-binary modulation, the *a priori* LLRs $L_a(c_n^i)$ on the coded bits are usually required in the demapping process, as shown by the dashed line in figure 4.1.

This structure is intuitively satisfying since, over an AWGN channel, it is well known that MMSE equalization reduces to a multiplication of the received observation by a constant factor (the so-called MMSE bias). Then, the overall SISO equalizer of figure 4.1 takes the form of a soft demodulator in the symbol-level case, or a SISO symbol demapper in the bit-level case respectively. This is consistent with the behavior observed in chapter 3 with BJCR-MAP equalizers.

Before discussing the realization of the inner MMSE equalizer, which is the subject of the next two sections, it is instructive to describe in more detail the calculations realized by the soft symbol mapper and the SISO demapper respectively.

4.1.1 The soft symbol mapper

The role of the soft symbol mapper is twofold: computing the soft estimates $\bar{x}_n$ on the basis of the prior LLRs available in the input, and providing a measure of the reliability of these estimates, information that we shall later exploit to derive the optimum equalizer coefficients.

Let us denote by $\mathcal{L}_{a,n}$ the set of (symbol or bit) *a priori* LLRs relative to the transmitted symbol x_n at time n . Then, the soft estimate $\bar{x}_n$ is given by the expectation of x_n conditioned to $\mathcal{L}_{a,n}$, viz.

$$\bar{x}_n = E(x_n | \mathcal{L}_{a,n}) \quad (4.1)$$

This expression is evaluated in practice using *a priori* symbol probabilities $P_{a,n}(s)$ computed from the prior LLRs. In the symbol-level case, these prior probabilities are directly obtained from the prior symbol LLRs delivered by the decoder at the previous iteration using relation (3.3). In the bit-level case, they are computed on the basis of the *a priori* LLRs $L_a(c_n^i)$ on the coded bits using expressions (3.5) and (3.2). The data estimate $\bar{x}_n$ is finally obtained as

$$\bar{x}_n = \sum_{s \in \mathcal{S}} s \times P_{a,n}(s) \quad (4.2)$$

This estimator has a number of valuable properties that we shall exploit in the derivation of the MMSE IC-LE in section 4.3. For the moment, it suffices to mention that this choice minimizes the mean-square error between x_n and $\bar{x}_n$ [150], and is particularly consistent with the definition of a *soft* estimator. Under the common assumption of i.i.d transmitted symbols, we indeed have $\bar{x}_n = 0$. Conversely, for perfect prior information, $\bar{x}_n$ identifies x_n in a deterministic manner so that $\bar{x}_n \rightarrow x_n$.

Regarding now the reliability measure v_n^2 , it is calculated as the variance of x_n conditioned to $\mathcal{L}_{a,n}$, and reads

$$v_n^2 = E \left(|x_n - \bar{x}_n|^2 | \mathcal{L}_{a,n} \right) = \sum_{s \in \mathcal{S}} |s|^2 \times P_{a,n}(s) - |\bar{x}_n|^2 \quad (4.3)$$

Note that this definition conforms to the requirements of a reliability measure in the sense that we have $v_n^2 \rightarrow 0$ in the limit of perfect *a priori* information about x_n , and $v_n^2 = \sigma_x^2$ under the general but not really informative assumption of i.i.d. transmitted symbols.

Motivations behind these formulations of $\bar{x}_n$ and v_n^2 should become clearer as we derive the form of the different equalizers in sections 4.2 and 4.3. We now turn our attention to the operations realized by the SISO demapper.

4.1.2 The SISO symbol demapper

The SISO symbol demapper has to compute extrinsic LLRs on the transmitted symbols or coded bits on the basis of the data estimates delivered by the equalizer. One can show that for filtering-based equalizers, assuming a restitution delay Δ , the equalized sample z_n delivered at time n can always be written as a sum of two terms

$$z_n = \mu_n x_{n-\Delta} + \eta_n \quad (4.4)$$

where the first quantity denotes the desired signal, μ_n being a bias term introduced by the equalizer, and where η_n is a zero-mean additive interference term encompassing filtered noise and residual ISI at the equalizer output, which is independent of the data sample x_n at time n . The SISO demapping module operates by assuming that the perturbation η_n follows a (usually complex) Gaussian distribution, with parameters $\mathcal{N}(0, v_n^2)$ at time n . This is tantamount to assuming that the equalizer has turned the ISI channel into an equivalent AWGN channel². As we shall see later in this chapter, the parameters (μ_n, v_n^2) are easily determined from the knowledge of the equalizer filter coefficients.

On the basis of the equivalent AWGN channel assumption at the equalizer output, the *a posteriori* probability of having transmitted symbol s at time n conditioned to the observation z_n reads

$$P(x_n = s | z_n) \propto P(z_n | x_n = s) P_{a,n}(s) \quad (4.5)$$

From the definition of the extrinsic probability relative to the symbol s at time n , we obtain

$$P_{e,n}(s) = \frac{P(x_n = s | z_n)}{P_{a,n}(s)} \propto P(z_n | x_n = s) \quad (4.6)$$

²We recall that such an assumption only rigorously holds on ISI channels in the presence of an ideal MMSE interference canceller with perfect *a priori* knowledge about the transmitted symbols. However, it offers good results in practice.

where the conditional probability $P(z_n|x_n = s)$ is given by

$$P(z_n|x_n = s) = \frac{1}{\pi v_n^2} \exp \left[-\frac{|z_n - \mu_n s|^2}{v_n^2} \right] \quad (4.7)$$

In the symbol-level case (e.g. for TCM), the extrinsic symbol LLRs follow from a direct application of relation (3.3) once the symbol extrinsic probabilities have been evaluated. An additional step is required in the bit-level case, where we have to convert the symbol extrinsic probabilities into extrinsic LLRs on the coded bits. The *a posteriori* LLR on a coded bit c_n^i is defined by

$$L(c_n^i) = \ln \frac{\sum_{s:c_n^i=1} P(z_n|x_n=s) P_{a,n}(s)}{\sum_{s:c_n^i=0} P(z_n|x_n=s) P_{a,n}(s)} = \ln \frac{\sum_{s:c_n^i=1} P_{e,n}(s) \prod_{j=1}^m P_a(c_n^j)}{\sum_{s:c_n^i=0} P_{e,n}(s) \prod_{j=1}^m P_a(c_n^j)} \quad (4.8)$$

Consequently, the extrinsic LLR on c_n^i is finally given by

$$L_e(c_n^i) = \ln \frac{\sum_{s:c_n^i=1} P_{e,n}(s) \prod_{j=1, j \neq i}^m P_a(c_n^j)}{\sum_{s:c_n^i=0} P_{e,n}(s) \prod_{j=1, j \neq i}^m P_a(c_n^j)} = L(c_n^i) - L_a(c_n^i) \quad (4.9)$$

where we have used definition (3.2) to obtain the last equality.

4.2 Finite-length MMSE equalization with *a priori* information

In this section, we review the time-varying finite-length MMSE equalizer proposed by Tüchler, as well as its low-complexity time-invariant approximation. Our exposition, however, differs from the one originally adopted by Tüchler. Specifically, building upon the presentation of similar receivers in [53] and [153], the equalizers are derived here in the general framework of MMSE interference cancellation. This yields a rigorous formulation for the equalizer coefficients, and also provides a theoretical justification for the definition of the quantities $\bar{x}_n$ and v_n^2 as introduced previously.

4.2.1 Preliminary notations and definitions

Our derivation is based on the following vector model for data transmission over an ISI channel with L taps (see section 2.2.4)

$$\mathbf{y}_n = \mathbf{H}\mathbf{x}_n + \mathbf{w}_n \quad (4.10)$$

where the vectors $\mathbf{y}_n$, $\mathbf{x}_n$ and $\mathbf{w}_n$ have respective sizes $N \times 1$, $J \times 1$ and $N \times 1$, and where $\mathbf{H}$ is an $N \times J$ channel convolution matrix, with $J \triangleq N + L - 1$. We shall be interested in the sequel in isolating the contribution of a particular data symbol $x_{n-\Delta}$. In this case, it is convenient to modify the previous

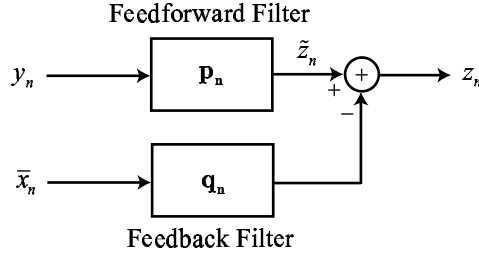


Figure 4.2: Block diagram of a finite-length MMSE interference canceller.

expression for y_n as follows

$$y_n = \mathbf{h}_\Delta x_{n-\Delta} + \mathbf{H}\tilde{\mathbf{x}}_n + \mathbf{w}_n \quad (4.11)$$

where $\mathbf{h}_\Delta$ denotes column Δ of matrix $\mathbf{H}$, and where $\tilde{\mathbf{x}}_n$ is the vector $\mathbf{x}_n$ whose component Δ has been set to zero, viz.

$$\tilde{\mathbf{x}}_n = \mathbf{x}_n - x_{n-\Delta} \mathbf{e}_\Delta \quad (4.12)$$

$\mathbf{e}_\Delta$ being the unit vector having a 1 in position Δ .

We now introduce a few definitions that will play an important role in the following discussions. We denoted in section 4.1.1 by $\mathcal{L}_{a,n}$ the set of *a priori* LLRs (on symbols or coded bits) relative to the transmitted symbol x_n at time n . Similarly, we now define by $\mathcal{L}_a$ the set of **all** *a priori* LLRs available at the SISO equalizer input. Finally, we denote by $\mathcal{L}_{a,[n-\Delta]}$ the set of **all** *a priori* LLRs available in the input, **with the exception of the LLRs relative to the symbol** $x_{n-\Delta}$. Formally, we can write

$$\mathcal{L}_{a,[n-\Delta]} = \mathcal{L}_a \setminus \mathcal{L}_{a,n-\Delta} \quad (4.13)$$

The reader is recommended to become familiar with these definitions since they will appear on several occasions in this chapter as well as in the derivations of appendix A.

4.2.2 The time-varying solution

Consider a general finite-length interference cancellation structure as in figure 4.2. Without loss of generality, we assume that the feedforward filter coefficients vector $\mathbf{p}_n$ has size N , and is allowed to change with time, hence the subscript n . Assuming that the feedforward filter introduces a restitution delay Δ , the filtered sample $\tilde{z}_n$ at time n is an estimate of the data symbol $x_{n-\Delta}$ transmitted at time $n - \Delta$. Using (4.12), we may decompose the expression of $\tilde{z}_n$ as a sum of three terms

$$\tilde{z}_n = \mathbf{p}_n^T \mathbf{y}_n = \underbrace{\mathbf{p}_n^T \mathbf{h}_\Delta}_{\mu_n} x_{n-\Delta} + \underbrace{\mathbf{p}_n^T \mathbf{H}}_{i_n} \tilde{\mathbf{x}}_n + \underbrace{\mathbf{p}_n^T}_{\zeta_n} \mathbf{w}_n \quad (4.14)$$

The first quantity is the desired signal, up to a time-varying bias factor μ_n . The second quantity i_n is the residual interference at the filter output. The third term ζ_n is a filtered noise sample. On the basis of some kind of estimate $\bar{x}_n$ on the transmitted data, the feedback filter $\mathbf{q}_n$ then aims at reconstructing an estimate of the residual interference $\hat{i}_n$, which is then subtracted from $\tilde{z}_n$ to obtain an improved estimate z_n of the data symbol $x_{n-\Delta}$, namely

$$z_n = \tilde{z}_n - \hat{i}_n \quad (4.15)$$

The standard MMSE equalization problem then involves finding the optimum values for the feedforward filter $\mathbf{p}_n$ and the interference estimate i_n which minimize the error signal $z_n - x_{n-\Delta}$ in the mean-square error sense.

In fact, in the presence of *a priori* information about the transmitted data, in LLR form for instance, we can do better. The philosophy is the following: just as we use *a priori* probabilities derived from extrinsic LLRs to improve detection/decoding with the BCJR-MAP algorithm, we may take advantage of these *a priori* probabilities to improve equalization structures relying on the MMSE criterion as well. Hence, we propose here to jointly optimize the feedforward filter coefficients and the interference estimate so as to minimize the mean-square error between the data symbol $x_{n-\Delta}$ and its estimate z_n , *conditioned to the set $\mathcal{L}_{a,[n-\Delta]}$ of all *a priori* LLRs available in the input which are not related to the current symbol of interest $x_{n-\Delta}$* . The omission of the LLRs relative to $x_{n-\Delta}$ is motivated by the desire to conform to the so-called Turbo Principle, which essentially states that any *a priori* information used to process a given symbol should not be propagated to subsequent decoding stages [80]. This optimization problem may thus be formally stated as follows

$$(\mathbf{p}_n, i_n) = \arg \min_{(\mathbf{p}_n, i_n)} E \left(|z_n - x_{n-\Delta}|^2 \mid \mathcal{L}_{a,[n-\Delta]} \right) \quad (4.16)$$

The solution is given in appendix A.1, and the results are summarized in table 4.1 below. ε_n^2 denotes the (conditional) minimum mean-square error at the equalizer output at time n .

We first observe that the optimum interference estimate i_n has the form (see appendix A.1)

$$i_n = \mathbf{q}_n^T E(\mathbf{x}_n \mid \mathcal{L}_{a,[n-\Delta]}) \quad (4.17)$$

where $\mathbf{q}_n$ is the coefficient vector for the feedback filter whose expression is given in table 4.1. Note that this filter has been constructed so that the central coefficient $q_{n,\Delta}$ is zero, as required for interference cancellation. As mentioned in the appendix, we can safely assume that the *a priori* LLRs relative to different data symbols x_n are independent, owing to the presence of interleavers in a turbo-equalization scheme. Then, from the expression of the optimum interference estimate, one obtains

Parameters	Dimensions
$\mathbf{C}_{\mathbf{xx},n} = \text{diag} \{v_n^2, \dots, v_{n-J+1}^2\}$	$J \times J$ real matrix
$\mathbf{S}_n = \mathbf{H}\mathbf{C}_{\mathbf{xx},n}\mathbf{H}^H + \sigma_w^2 \mathbf{I}$	$N \times N$ complex matrix
$\tilde{\mathbf{p}}_n^* = \sigma_x^2 \mathbf{S}_n^{-1} \mathbf{h}_\Delta$	$N \times 1$ complex vector
$\tilde{\mu}_n = \tilde{\mathbf{p}}_n^T \mathbf{h}_\Delta$	real scalar
$\lambda_n = \sigma_x^2 / (\sigma_x^2 + (\sigma_x^2 - v_{n-\Delta}^2) \tilde{\mu}_n)$	real scalar
$\mathbf{p}_n = \lambda_n \tilde{\mathbf{p}}_n$	$N \times 1$ complex vector
$\mu_n = \mathbf{p}_n^T \mathbf{h}_\Delta = \lambda_n \tilde{\mu}_n$	real scalar
$\mathbf{q}_n = \mathbf{H}^T \mathbf{p}_n - \mu_n \mathbf{e}_\Delta$	$J \times 1$ complex vector
$v_n^2 = \sigma_x^2 \mu_n (1 - \mu_n)$	real scalar
$\varepsilon_n^2 = \sigma_x^2 (1 - \mu_n)$	real scalar

Table 4.1: Parameters for the time-varying finite-length MMSE equalizer with *a priori* information.

that the optimum data estimates $\bar{x}_n$ are in fact given by

$$\bar{x}_n = E(x_n | \mathcal{L}_{a,n}) \quad (4.18)$$

This is precisely the form that was presented without proof in subsection 4.1.1, but it finds here a rigorous justification. Similarly, we note from table 4.1 that the computation of the optimum feedforward filter $\mathbf{p}_n$ makes use of the reliability measures v_n^2 as defined in subsection 4.1.1.

In order to gain more insight into the behavior of this equalizer, it is instructive to study the limiting forms obtained under the assumption of uniform and perfect prior knowledge about the transmitted symbols respectively. Assuming i.i.d. transmitted symbols, as we usually do at the first iteration in a turbo-equalization process, we have $\bar{x}_n = 0$ and $v_n^2 = \sigma_x^2$. In this case, the feedback branch is of no use and can be safely ignored. Then, the optimum feedforward filter becomes time-invariant and has the following expression

$$\mathbf{p}_n^* = \sigma_x^2 [\sigma_x^2 \mathbf{H}\mathbf{H}^H + \sigma_w^2 \mathbf{I}]^{-1} \mathbf{h}_\Delta \quad (4.19)$$

We recognize here the form of a classical finite-length MMSE linear equalizer, as presented in chapter 2, section 2.2.5. Conversely, under the assumption of perfect *a priori* knowledge about the data symbols x_n , we have $\bar{x}_n \rightarrow x_n$ and $v_n^2 \rightarrow 0$. It follows from table 4.1 that in this case,

$$\mathbf{p}_n \longrightarrow \frac{\sigma_x^2}{\sigma_x^2 \|h\|^2 + \sigma_w^2} \mathbf{h}_\Delta^* \quad , \text{ with } \|h\|^2 \triangleq \mathbf{h}_\Delta^H \mathbf{h}_\Delta \quad (4.20)$$

$$\mathbf{q}_n \longrightarrow \mathbf{H}^T \mathbf{p} - \frac{\sigma_x^2 \|h\|^2}{\sigma_x^2 \|h\|^2 + \sigma_w^2} \mathbf{e}_\Delta \quad (4.21)$$

Again, we observe that the optimum filters become time-invariant, and we recognize the form of an ideal finite-length MMSE interference canceller supplied with perfect estimates about the transmitted

data (see chapter 2, section 2.2.7). To summarize, this equalizer adapts its equalization strategy as a function of the reliability of the data estimates $\bar{x}_n$, on a *per-symbol basis*.

When discussing MMSE interference cancellation in chapter 2, we showed that the resulting equalizer could be realized in two different ways. In the classical structure, the interference is subtracted at the output of the feedforward filter, as shown in figure 4.2 for instance. However there is another option which involves reconstructing the ISI contribution and subtracting it from the channel observation, in front of the feedforward filter. This alternative structure is shown in figure 4.3 and can be identified with the equalization scheme originally proposed by Tüchler. The equivalence between the two structures is straightforward to establish. Note, however, that the equalizer of figure 4.3 requires fewer coefficients for the cancellation process (L instead of $J = L + M - 1$), and thus may be preferred for a practical implementation.

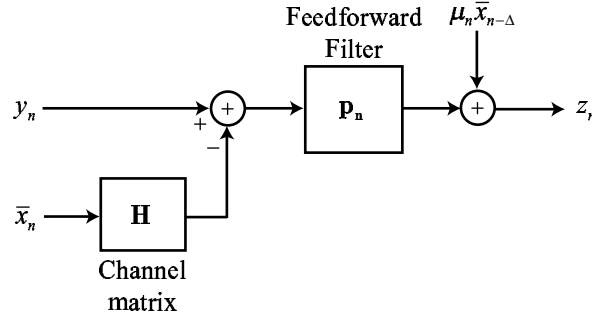


Figure 4.3: Equivalent block diagram of the finite-length MMSE equalizer with *a priori* information.

A discussion about the complexity of implementation is finally in order here. The computation of the equalizer coefficients is a computationally demanding task, since it requires the inversion of the $N \times N$ matrix $\mathbf{S}_n$ for every symbol. Direct matrix inversion is an $O(N^3)$ process in the general case³. However, noting the time coherence of the matrix $\mathbf{S}_n^{-1}$, Tüchler developed an efficient time-recursive method to compute $\mathbf{S}_n^{-1}$ from $\mathbf{S}_{n-1}^{-1}$, thereby reducing the computational load to $O(N^2 + L^2)$ operations per symbol [172, 175]. While this achievement may offer an appreciable speed gain over standard matrix inversion, this equalizer remains merely of theoretical interest at the present date, in view of the current hardware/software capabilities. For this reason, several approximate implementations with a much lower computational cost were suggested in [174, 175], and we shall now examine one of these solutions that we find particularly attractive.

³We refer the interested reader to appendix B for a discussion about the meaning of the $O(\cdot)$ notation.

4.2.3 A time-invariant approximation

Clearly, the key issue in reducing the complexity of the previous approach involves devising a time-invariant equalizer, whose coefficients need only be computed once per processed data block. In particular, Tüchler suggested in [175] computing the time average $\bar{v}^2$ over the set of conditional variances $\{v_n^2\}$

$$\bar{v}^2 = \frac{1}{N_S} \sum_{n=0}^{N_S-1} v_n^2 \quad (N_S: \text{number of symbols in a data block}) \quad (4.22)$$

and then substituting the time-varying covariance matrix $\mathbf{C}_{\mathbf{x}\mathbf{x},n}$ in the calculation of the optimum filters with the time-invariant average covariance matrix $\bar{\mathbf{C}}_{\mathbf{x}\mathbf{x}}$, defined by

$$\bar{\mathbf{C}}_{\mathbf{x}\mathbf{x}} = \bar{v}^2 \mathbf{I} \quad (4.23)$$

This yields a time-invariant equalizer, whose parameters are summarized in table 4.2. Again, the quantity ε^2 denotes the minimum mean-square error at the equalizer output.

Parameters	Dimensions
$\bar{\mathbf{C}}_{\mathbf{x}\mathbf{x}} = \bar{v}^2 \mathbf{I}$	$J \times J$ real matrix
$\mathbf{S} = \mathbf{H} \bar{\mathbf{C}}_{\mathbf{x}\mathbf{x}} \mathbf{H}^H + \sigma_w^2 \mathbf{I}$	$N \times N$ complex matrix
$\tilde{\mathbf{p}}^* = \sigma_x^2 \mathbf{S}^{-1} \mathbf{h}_\Delta$	$N \times 1$ complex vector
$\tilde{\mu} = \tilde{\mathbf{p}}^T \mathbf{h}_\Delta$	real scalar
$\lambda = \sigma_x^2 / (\sigma_x^2 + (\sigma_x^2 - \bar{v}^2) \tilde{\mu})$	real scalar
$\mathbf{p} = \lambda \tilde{\mathbf{p}}$	$N \times 1$ complex vector
$\mu = \mathbf{p}^T \mathbf{h}_\Delta = \lambda \tilde{\mu}$	real scalar
$\mathbf{q} = \mathbf{H}^T \mathbf{p} - \mu \mathbf{e}_\Delta$	$J \times 1$ complex vector
$v^2 = \sigma_x^2 \mu (1 - \mu)$	real scalar
$\varepsilon^2 = \sigma_x^2 (1 - \mu)$	real scalar

Table 4.2: Parameters for the time-invariant finite-length MMSE equalizer with *a priori* information.

This solution may be viewed as an average realization of the previous time-varying equalizer. Consequently, the equalization strategy is adapted here on a per data block basis rather than for every processed symbol, and the resulting equalizer is thus naturally expected to offer lower performance. How much exactly is lost will be examined by simulation in section 4.5. This approximate solution features, nonetheless, a number of attractive properties.

First, it is straightforward to verify from the parameters in table 4.2 that this time-invariant approximation has the same limiting forms as the time-varying solution in the presence of uniform and perfect prior knowledge about the transmitted symbols respectively. The last observation implies in

particular that both equalizers are expected to offer the same asymptotic performance when integrated in a turbo-equalization scheme, as the SNR increases. In addition, this time-invariant equalizer is particularly well suited for a practical implementation from a complexity point of view. Only one matrix inversion is in fact required per data block. In addition, for symbol-spaced equalizers and transmissions over time-invariant, quasi-static or block-fading channels, the matrix $\mathbf{H}$ is Toeplitz. In this case, $\mathbf{S}$ is also Toeplitz and its inversion can be carried out using computationally efficient methods such as the Levinson-Durbin algorithm [76] in $O(N^2)$ operations. In fact, computing the optimum coefficients for this equalizer should not require more operations than computing the optimum settings for a classical MMSE linear or decision-feedback equalizer.

We conclude this exposition by mentioning that like for the time-varying solution, this equalizer can be implemented in practice using either the structure in figure 4.2 or the one in figure 4.3.

4.3 The MMSE IC-LE

In this section, we introduce a novel form of MMSE equalizer, called the MMSE IC-LE, that was inspired by the respective works of Laot *et al* [102], Tüchler [175], and Chan and Wornell [36]. In contrast with the approach adopted in section 4.2, we take here a different look at the problem of designing MMSE equalizers in the presence of *a priori* information. Specifically, we start from a classical MMSE interference cancellation structure, but where we do explicitly account for the fact that the supplied data estimates may not be perfect. In other words, we provide the interference canceller with some form of *a priori* knowledge about the reliability of the data estimates, information that is in fact captured by the variance of these estimates. We obtain a time-invariant equalizer, which can be derived both in infinite-length form, as well as under finite-length realization constraints.

The MMSE IC-LE shares striking similarities with the equalizer proposed by Tüchler. In particular, it also presents a self-reconfigurable structure, which switches progressively from a classical MMSE linear equalizer to an ideal MMSE interference canceller as the reliability of the soft estimates increases. This analogy culminates with the derivation of the finite-length MMSE IC-LE, which is shown to be equivalent to the time-invariant approximation introduced in section 4.2.3. However, this equalizer is given here a rigorous definition, on the basis of adequate assumptions about the statistics of the data estimates.

We begin by examining the statistical properties of the sequence of soft data estimates delivered to the interference canceller. Then, we derive the MMSE IC-LE both in infinite-length and finite-length form. In the latter case, the equivalence with the equalizer of section 4.2.3 is established. Finally, we expose a low-complexity approximate method for computing the equalizer coefficients, which relies on the Fast Fourier Transform.

4.3.1 Statistical properties of the soft data estimates

We consider an MMSE interference canceller supplied with soft data estimates $\bar{x}_n$ that are computed as described in subsection 4.1.1, viz.

$$\bar{x}_n = E(x_n | \mathcal{L}_{a,n}) \quad (4.24)$$

where we recall that the notation $\mathcal{L}_{a,n}$ refers to the set of *a priori* LLRs that pertain to the data symbol x_n at time n . Derivation of the optimum filters for the MMSE interference canceller requires the knowledge of both the auto-correlation $R_{\bar{x}\bar{x},k}$ of the sequence $\{\bar{x}_n\}$ and the cross-correlation $R_{x\bar{x},k}$ between the sequences $\{x_n\}$ and $\{\bar{x}_n\}$. We shall now show how to model these quantities from the statistical properties of $\bar{x}_n$.

Since $\bar{x}_n$ is defined as the MMSE estimate of the random variable x_n given the observation of the set of random variables $\mathcal{L}_{a,n}$, the following equalities hold

$$E(\bar{x}_n) = E(E(x_n | \mathcal{L}_{a,n})) = E(x_n) \quad (4.25)$$

$$E(\bar{x}_n x_n^*) = E(\bar{x}_n \bar{x}_n^*) = E(|\bar{x}_n|^2) \quad (4.26)$$

We refer the reader to [150, chap. 2] or [93, app. 3.A] for detailed proofs of these very general results. We suppose in the sequel that the transmitted symbols x_n have zero mean and are uncorrelated, a valid model for many practical transmission systems. Then, it follows from the previous equalities that

$$E(\bar{x}_n) = 0 \quad (4.27)$$

$$E(\bar{x}_n x_n^*) = E(|\bar{x}_n|^2) = \sigma_{\bar{x}}^2 \quad (4.28)$$

where $\sigma_{\bar{x}}^2$ denotes the variance of the data estimates $\bar{x}_n$. Some remarks are in order here. Assuming uniform prior knowledge about the transmitted data symbols, we have seen in section 4.1.1 that the soft estimates $\bar{x}_n$ are identically zero. Clearly then, we have $E(\bar{x}_n x_n^*) = 0$ and thus $\sigma_{\bar{x}}^2 = 0$. In contrast, assuming that $\bar{x}_n$ is a perfect estimate of x_n , we observe from property (4.28) that $\sigma_{\bar{x}}^2$ becomes equal to the variance σ_x^2 of the transmitted data (just substitute $\bar{x}_n$ with x_n in the equation to obtain this result). In other words, *the variance $\sigma_{\bar{x}}^2$ of the soft data estimates is an indicator of the reliability of these estimates*. Finally, owing to the presence of an interleaver in a turbo-equalization scheme, it seems reasonable to suppose that the data estimates are uncorrelated, so that we obtain the following model for the auto- and cross-correlation functions

$$R_{\bar{x}\bar{x},k} = E(x_n \bar{x}_{n-k}^*) = E(\bar{x}_n \bar{x}_{n-k}^*) = R_{\bar{x}\bar{x},k} = \sigma_{\bar{x}}^2 \delta_k \quad (4.29)$$

We now describe two methods to estimate $\sigma_{\bar{x}}^2$ in practice.

Since the data estimates $\bar{x}_n$ have zero mean, their variance $\sigma_{\bar{x}}^2$ can be calculated on a per data block

basis using the sample mean estimator

$$\sigma_{\bar{x}}^2 \approx \frac{1}{N_S} \sum_{n=0}^{N_S-1} |\bar{x}_n|^2 \quad (N_S: \text{number of symbols in a data block}) \quad (4.30)$$

There is however another solution which involves computing $\sigma_{\bar{x}}^2$ from the time average $\bar{v}^2$ of the conditional variances v_n^2 delivered by the soft symbol mapper, a quantity that was introduced in section 4.2.3. In fact, this second method is suggested by noting that the variance $\sigma_{\bar{x}}^2$ of the data estimates is theoretically related to the expected value $E(v_n^2)$ of the quantities v_n^2 as follows

$$E(v_n^2) = E\left(E(|x_n - \bar{x}_n|^2 | \mathcal{L}_{a,n})\right) \quad (4.31)$$

$$= E\left(E(|x_n|^2 | \mathcal{L}_{a,n})\right) - E(|\bar{x}_n|^2) \quad (4.32)$$

$$= E(|x_n|^2) - E(|\bar{x}_n|^2) \quad (4.33)$$

$$= \sigma_x^2 - \sigma_{\bar{x}}^2 \quad (4.34)$$

Since $\bar{v}^2$ is by definition an estimate of $E(v_n^2)$, we obtain the approximate equality

$$\sigma_{\bar{x}}^2 \approx \sigma_x^2 - \bar{v}^2 \quad (4.35)$$

This expression will be used later to prove the equivalence between the finite-length MMSE IC-LE and the time-invariant equalizer of section 4.2.3. We considered the two solutions (4.30) and (4.35) for estimating $\sigma_{\bar{x}}^2$ in our simulations and found that they perform nearly equally well. Method (4.30) should be preferred in a practical implementation since it requires fewer computations.

4.3.2 The infinite-length MMSE IC-LE

Consider now an infinite-length MMSE interference cancellation structure as shown in chapter 2, figure 2.18. We propose to determine the form of the optimum feedforward filter $P(\omega)$ and feedback filter $Q(\omega)$ which minimize the mean-square error between the data symbol x_n and its estimate z_n , on the basis of the correlation properties of the soft data estimates established in the previous subsection. This optimization problem can be stated formally as follows

$$(P(\omega), Q(\omega)) = \arg \min_{(P(\omega), Q(\omega))} E(|z_n - x_n|^2) \quad (4.36)$$

where we impose the condition $q_0 = 0$, q_0 being the reference tap of the feedback filter, since $Q(\omega)$ aims only at cancelling the residual ISI at the output of the feedforward filter. This condition furthermore conforms to the Turbo Principle. This optimization problem is solved in appendix A.2, and we

obtain

$$P(\omega) = \lambda \frac{\sigma_x^2 H^*(\omega)}{(\sigma_x^2 - \sigma_x^2) |H(\omega)|^2 + \sigma_w^2} \quad (4.37)$$

$$Q(\omega) = H(\omega)P(\omega) - \mu \quad (4.38)$$

where we have introduced the following scalar parameters

$$\tilde{\mu} = \frac{1}{2\pi} \int_{-\pi}^{+\pi} \frac{\sigma_x^2 |H(\omega)|^2}{(\sigma_x^2 - \sigma_x^2) |H(\omega)|^2 + \sigma_w^2} d\omega \quad (4.39)$$

$$\lambda = \frac{\sigma_x^2}{1 + \tilde{\mu} \sigma_x^2} \quad (4.40)$$

$$\mu = \lambda \tilde{\mu} = \frac{1}{2\pi} \int_{-\pi}^{+\pi} H(\omega)P(\omega) d\omega \quad (4.41)$$

Note that μ is defined as the reference tap of the overall filter formed by the cascade of the channel with the feedforward filter $P(\omega)$, and thus corresponds to the bias factor introduced by the equalizer. We recall that this parameter is required by the SISO demapper (see section 4.1.2), as well as the variance v^2 of the equivalent noise at the equalizer output, which is given by (see appendix A.2)

$$v^2 = \sigma_x^2 \mu (1 - \mu) \quad (4.42)$$

Finally, the minimum mean-square error ε^2 at the equalizer output admits the following expression

$$\varepsilon^2 = \sigma_x^2 (1 - \mu) \quad (4.43)$$

This completes the description of the infinite-length MMSE IC-LE.

It is straightforward to verify that in the presence of uniform *a priori* information, i.e. $\sigma_x^2 = 0$, the MMSE IC-LE reduces to a classical infinite-length MMSE linear equalizer (section 2.2.5). Conversely, assuming perfect knowledge of the data symbols, we have $\sigma_x^2 \rightarrow \sigma_x^2$ and the MMSE IC-LE converges towards an ideal MMSE interference canceller supplied with perfect estimates (section 2.2.7). Hence, like for the finite-length equalizers introduced by Tüchler, we observe that the MMSE IC-LE adapts its equalization strategy as a function of the reliability of the data estimates, information which is captured by the variance of these estimates.

4.3.3 The finite-length MMSE IC-LE

The filters derived previously have infinite-length and thus need to be truncated for a practical implementation. A better approach then involves deriving the optimum filters under the constraint of a finite-length realization. Accordingly, we consider now a finite-length MMSE interference can-

cellation structure as in figure 4.2, but where the feedforward filter $\mathbf{p}$ and the feedback filter $\mathbf{q}$ are time-invariant. We furthermore suppose that $\mathbf{p}$ has size N . Assuming that the discrete-time channel model spans L symbol periods, the convolution of the channel with the feedforward filter cannot span more than $J = N + L - 1$ symbol periods. Consequently, the feedback filter $\mathbf{q}$ requires J coefficients at most. Assuming that the feedforward filter introduces a restitution delay Δ , the optimization problem here involves finding the optimum finite-length filters $\mathbf{p}$ and $\mathbf{q}$ which minimize the mean-square error between $x_{n-\Delta}$ and its estimate z_n at time n

$$(\mathbf{p}, \mathbf{q}) = \arg \min_{(\mathbf{p}, \mathbf{q})} E \left(|z_n - x_{n-\Delta}|^2 \right) \quad (4.44)$$

under the constraint that the reference tap q_Δ of the feedback filter is zero, and taking into account the correlation properties of the data estimates $\bar{x}_n$ established in section 4.3.1. This optimization problem is solved in appendix A.3, and the parameters of the resulting equalizer are summarized in table 4.3. ε^2 denotes the minimum mean-square error at the equalizer output.

Parameters	Dimensions
$\mathbf{S} = (\sigma_x^2 - \sigma_{\bar{x}}^2) \mathbf{H}\mathbf{H}^H + \sigma_w^2 \mathbf{I}$	$N \times N$ complex matrix
$\tilde{\mathbf{p}}^* = \mathbf{S}^{-1} \mathbf{h}_\Delta$	$N \times 1$ complex vector
$\tilde{\mu} = \tilde{\mathbf{p}}^T \mathbf{h}_\Delta$	real scalar
$\lambda = \sigma_x^2 / (1 + \sigma_{\bar{x}}^2 \tilde{\mu})$	real scalar
$\mathbf{p} = \lambda \tilde{\mathbf{p}}$	$N \times 1$ complex vector
$\mu = \mathbf{p}^T \mathbf{h}_\Delta = \lambda \tilde{\mu}$	real scalar
$\mathbf{q} = \mathbf{H}^T \mathbf{p} - \mu \mathbf{e}_\Delta$	$J \times 1$ complex vector
$v^2 = \sigma_x^2 \mu (1 - \mu)$	real scalar
$\varepsilon^2 = \sigma_x^2 (1 - \mu)$	real scalar

Table 4.3: Parameters for the finite-length MMSE IC-LE.

Inspecting tables 4.3 and 4.2, we observe that the finite-length MMSE IC-LE is very similar to the time-invariant approximate equalizer presented in section 4.2.3. In fact, the equivalence between the two solutions can be readily established from relation (4.35) and by performing a few changes of variables. Since these equalizers are rigorously equivalent, they share the same properties and the remarks introduced in section 4.2.3 thus apply to the finite-length MMSE IC-LE as well. In particular, the coefficients of the MMSE IC-LE need only be calculated once per data block, and their computation is an $O(N^2)$ process in the most favorable case.

4.3.4 A low-complexity procedure for computing the filter coefficients

We finally describe an approximate but rather simple method that can be used to obtain the MMSE IC-LE filter coefficients in finite-length form. This method constitutes an alternative to the matrix inversion required by the computation procedure exposed in the previous subsection. It involves evaluating the frequency response $P(\omega)$ in equation (4.37) at the set of N uniformly spaced discrete frequencies $\{\omega_k = 2\pi k/N\}$. Such a computation can be carried out efficiently in $O(N \log_2 N)$ operations using the FFT if N is chosen as a power of 2. Then, performing an inverse FFT on the sampled frequency response returns the impulse response $\{p_k\}$ of the feedforward filter in the time-domain. The impulse response $\{q_k\}$ of the feedback filter is finally obtained as the convolution of the feedforward filter response $\{p_k\}$ with the channel impulse response $\{h_k\}$, and by setting $q_\Delta = 0$. Note that by virtue of the FFT properties, the resulting equalizer introduces a fixed restitution delay $\Delta = N/2$. This procedure is summarized in table 4.4. We mention that similar approaches have already been proposed in the literature, but for computing the settings of conventional equalization structures (see for example [148] or [113]).

-
1. Compute the FFT $\{H_k\}$ of $\{h_k\}$ on N points
 2. Compute $P'_k = H_k^* / [(\sigma_x^2 - \sigma_x^2) |H_k|^2 + \sigma_w^2]$ for $k = 0..N-1$
 3. Compute $\tilde{\mu} = (1/N) \sum_{k=0}^{N-1} H_k P'_k$
 4. Compute $\lambda = \sigma_x^2 / (1 + \sigma_x^2 \tilde{\mu})$, and $\mu = \lambda \tilde{\mu}$
 5. Compute $P_k = \lambda P'_k$ for $k = 0..N-1$
 6. Take the IFFT of $\{P_k\}$ on N points to get the impulse response $\{p_k\}$
 7. Compute $\{q_k\}$ as the convolution of $\{p_k\}$ with $\{h_k\}$ and set $q_0 = 0$
-

Table 4.4: Low-complexity approximate method for computing the MMSE IC-LE filter coefficients.

Let us first examine the reduction in complexity resulting from the use of the FFT method with respect to the optimal computation procedure of table 4.3 relying on the Levinson-Durbin algorithm. We propose to measure the computational cost of the two methods in terms of *flops*, where one *flop* stands for one basic floating-point operation (addition, subtraction, multiply or divide between real numbers). We stress however that flop counting is a crude approach to measuring the efficiency of an algorithm. In particular, it assigns the same weight to different elementary operations such as addition and division, which do not really have the same cost in hardware. Hence, one should not infer too much from a comparison of flop counts. We focus here on the computation of the feedforward filter, and disregard the other operations. From [76], we obtain that the Levinson-Durbin approach has an overall cost of $4N^2$ flops. In contrast, the FFT method only requires one FFT and one IFFT to obtain $\{p_k\}$, resulting in a global cost of $2 \times 5N \log_2 N = 10N \log_2 N$ flops [132]. A simple numerical evaluation of the previous costs then suggests that the FFT method may potentially offer significant

computation savings for $N \geq 15$. However, we recall that this solution is only a sub-optimal approach, and that we also have to examine the accuracy of the resulting set of coefficients.

Mathematically, the FFT method amounts to approximating the Toeplitz matrix $\mathbf{S}$ in table 4.3 with its corresponding *circulant* matrix $\mathbf{S}_C$ where each column is obtained as a “downshifted” version of its predecessor. It is well known (see e.g. [79, chap. 4]) that the matrices $\mathbf{S}$ and $\mathbf{S}_C$ are asymptotically equivalent when $N \rightarrow \infty$. Hence, the accuracy of the FFT method with respect to an optimum calculation of the finite-length filters will greatly depend on the number N of coefficients considered in the FFT calculations. In order to find the minimum value for N required to closely match the performance obtained with the optimum approach, we undertook the following experiment. Both methods were simulated over 5000 random realizations of a quasi-static multipath Rayleigh fading channel with 5 taps of equal average power. The average mean-square error at the equalizer output was measured for increasing values of N and different values of the channel SNR E_s/N_0 . The parameter σ_x^2 was set to zero, so that the MMSE IC-LE reduces to a classical MMSE IC-LE. This constitutes a worst-case study since as the reliability of the data estimates increases, the MMSE IC-LE converges towards an ideal MMSE IC which generally requires fewer coefficients than an MMSE LE. The simulation results are shown in figure 4.4. We observe that choosing $N = 32$ provides performance close to optimum in this context, except at high SNR. In the latter case, the MMSE LE in fact approaches a Zero-Forcing LE and more coefficients are required with the FFT method to compensate for deep attenuations in the channel frequency response. On the other hand, the MMSE IC-LE is very likely to converge quickly towards the ideal MMSE IC at such SNR values, so that a small loss at the first iteration should not dramatically affect the overall performance. In contrast, the FFT computation method delivers highly unreliable coefficients for a very small equalizer length.

From the previous observations, we conclude that for $N \leq 16$ coefficients, the optimum approach (matrix inversion) should be preferred both from a performance and complexity point of view. In contrast, the FFT method may realize an interesting alternative for $N \geq 16$. In our experience, choosing $N = 32$ usually guarantees good performance with the latter approach.

4.3.5 Additional remarks

We conclude this presentation of SISO equalization structures optimized according to the MMSE criterion by discussing two issues of practical importance, and that concern both the MMSE IC-LE and the time-varying equalizer proposed by Tüchler.

We first take a look at the problem of choosing the adequate number of coefficients N and restitution delay Δ for a finite-length implementation. As explained in the previous subsection, it is generally a good idea to optimize these parameters so that the resulting equalizer will offer good performance at the first iteration (where it reduces in fact to a classical MMSE LE). In particular, the restitution

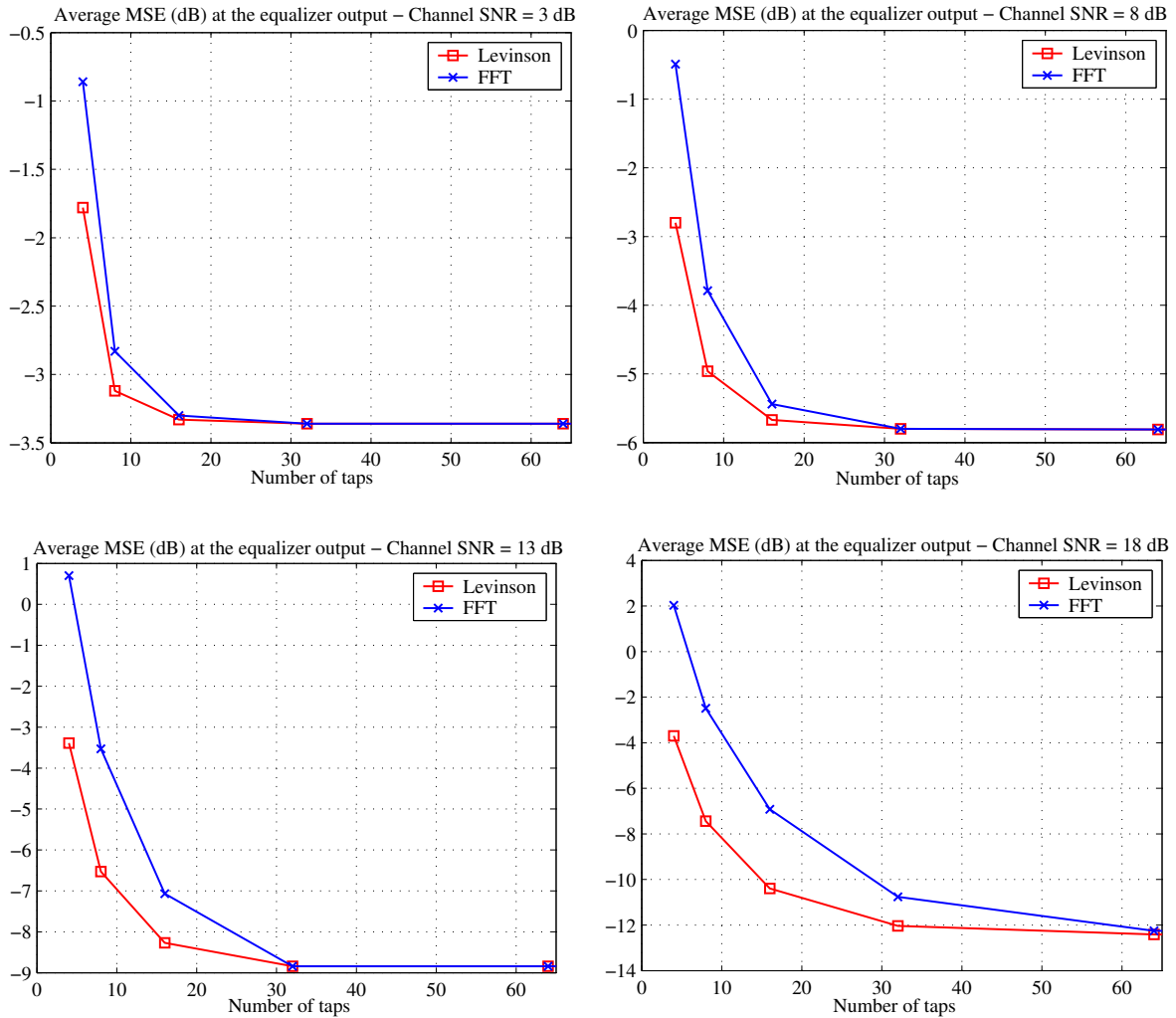


Figure 4.4: Average mean-square error obtained at the MMSE IC-LE output with both computation methods and measured over 5000 realizations of a quasi-static channel with 5 taps of equal average power.

delay Δ may be carefully adjusted using the methods described in chapter 2, especially for a small equalizer length. On the other hand, Δ should be chosen large enough to allow for the realization of the matched-filter without truncation, as required with an ideal MMSE interference canceller. In other words, it is wise to set $\Delta \geq L - 1$ to guarantee optimum performance at high SNR over a discrete-time channel model with L taps.

Consider now the situation of a transmission over a time-varying multipath Rayleigh fading channel which does not obey the quasi-static assumption. In this context, the channel variations have to be tracked and the equalizer coefficients have to be adjusted accordingly. While the time-varying equalizer proposed by Tüchler may be well-suited in such a scenario, it remains computationally-intensive to implement in practice. On the other hand, the time-invariant equalizers studied previously im-

explicitly suppose that the channel coefficients remain approximatively constant over a data block and thus are not directly applicable in this context. A possible solution would then involve computing the equalizer coefficients using a channel estimate at the beginning of the data block, and then updating the resulting coefficients on a per-symbol basis using standard adaptive algorithm (LMS or RLS). The resulting turbo-equalizer will then be almost identical to the solution initially proposed by Glavieux *et al.* This issue goes beyond the scope of this dissertation, but would merit attention for certain applications.

4.4 Asymptotic performance bounds for MMSE turbo-equalization

Just as we did in chapter 3 with the MAP turbo-equalizer, it is instructive to examine here the asymptotic performance reached by the different MMSE turbo-equalizers introduced in sections 4.2 and 4.3, under the assumption of perfect *a priori* information about the transmitted coded bits or symbols.

We have pointed out that in the presence of perfect data estimates, all the MMSE equalizers considered converge towards an ideal MMSE interference cancellation scheme. The signal at the output of an ideal MMSE IC has the form

$$z_n = \|h\|^2 x_{n-\Delta} + \sum_{\ell=0}^{L-1} h_{-\ell}^* w_{n-\ell} \quad (4.45)$$

where we have defined $\|h\|^2 = \sum_{\ell=0}^{L-1} |h_{-\ell}|^2 = \mathbf{h}_{\Delta}^H \mathbf{h}_{\Delta}$. In fact, from expression (4.45), we observe that an ideal MMSE IC converts the ISI channel into an ISI-free channel with gain factor $\|h\|^2$, perturbed by additive correlated Gaussian noise. The signal-to-noise ratio at the equalizer output then reads

$$\text{SNR} = \frac{\sigma_x^2 \|h\|^4}{\sigma_w^2 \|h\|^2} = \frac{\sigma_x^2 \|h\|^2}{\sigma_w^2} = \text{SNR}_{\text{MFB}} \quad (4.46)$$

where the last equality comes from definition (2.9). This expression demonstrates that the MMSE equalizers achieve the matched-filter bound for perfect *a priori* information. Now considering the general structure of MMSE SISO equalizers shown in figure 4.1, it follows that MMSE turbo-equalizers then achieve the same asymptotic performance as MAP turbo-equalizers. This fundamental result is summarized in the following proposition, which holds both for TCM and BICM MMSE turbo-equalization schemes.

Proposition 4.1 (Asymptotic bound for MMSE turbo-equalization). *In the limit of perfect a priori knowledge about the transmitted symbols (or coded bits), turbo-equalization structures optimized according to the MMSE criterion have the same asymptotic performance as the corresponding MAP turbo-equalizers if the inner MMSE equalizer converges towards an ideal MMSE interference canceller.*

In summary, MMSE turbo-equalizers converge towards the performance of the underlying coded modulation scheme over an equivalent AWGN channel in the limit of perfect *a priori* information. Similar conclusions were also raised by Langlais in [99] and Dejonghe and Vandendorpe in [46].

4.5 Performance results

We examine in this section the performance obtained with the MMSE turbo-equalization structures. For the sake of conciseness and due to time limitations, we restrict ourselves to time-invariant ISI channel models. As mentioned briefly in subsection 4.3.5, fully-interleaved Rayleigh fading channels require additional studies that are beyond the scope of this work since some kind of adaptive algorithm is required to track the channel variations. On the other hand, simulation results over quasi-static fading channels can be found in [103], where we have investigated the application of the MMSE IC-LE turbo-equalizer to the EDGE wireless telephony standard and obtained encouraging results. Note finally that the finite-length MMSE equalizers have been computed using exact matrix inversion techniques rather than the FFT method in all the following simulations.

4.5.1 Comparison between the different turbo-equalizers

We first propose to compare the performance obtained with a MAP equalizer, the time-varying MMSE equalizer proposed by Tüchler and the MMSE IC-LE respectively. Motivated by the case study presented in [175], we considered a simple BICM transmission system with BPSK modulation and a recursive systematic convolutional encoder with transfer function $G(D) = \left[1, \frac{1+D^2}{1+D+D^2}\right]$. A random interleaver of size 65536 bits was used. The system was simulated over the 5-tap severe-ISI Proakis C channel model, with impulse response $[0.227, 0.460, 0.688, 0.460, 0.227]$ [138]. The MMSE equalizers were realized using $N = 15$ coefficients and a restitution delay $\Delta = 9$. Each receiver performed 15 iterations. The results are shown in figure 4.5, where the label TV MMSE-LE stands for Time-Varying MMSE linear equalizer (section 4.2).

We observe that the turbo-equalizer relying on the time-varying MMSE equalizer comes very close to the performance offered by optimum MAP turbo-equalization on this severe channel. In addition, the MMSE IC-LE only suffers a convergence SNR penalty of about 1 dB with respect to the time-varying equalizer in this case. Since the latter has a much higher complexity in practice, we shall only consider the MMSE IC-LE in the following simulations since it realizes the best tradeoff between performance and computational cost. We mention that extensive simulations have been carried out over a wide range of transmission scenarios in [107] in order to quantify precisely the performance loss resulting from the use of the MMSE IC-LE with respect to the time-varying solution. We measured a gap of 1 dB at most, the worst-cases occurring with severe-ISI channels.

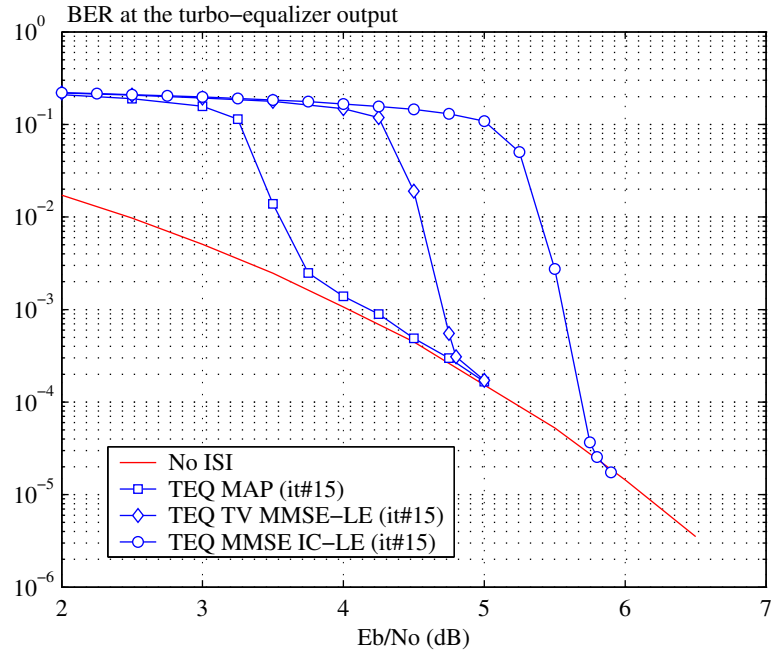


Figure 4.5: BER performance obtained with different turbo-equalization schemes after 15 iterations and over the severe-ISI 5-taps time-invariant Proakis C channel model.

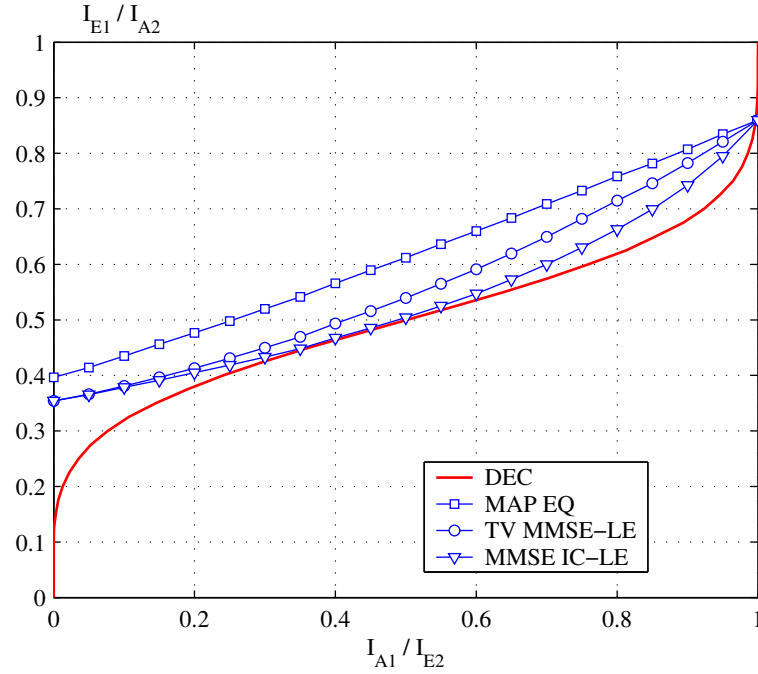


Figure 4.6: EXIT chart for different turbo-equalization schemes over the Proakis C channel at $E_b/N_0 = 5$ dB.

Figure 4.6 shows the corresponding EXIT charts for the three turbo-equalizers. They were computed at a signal-to-noise ratio $E_b/N_0 = 5$ dB. We observe that the characteristic of the time-varying equalizer opens a wider tunnel than the transfer function of the MMSE IC-LE with respect to the decoder curve. Hence, the time-varying equalizer enables an earlier and faster convergence, as shown by the simulation results of figure 4.5. This difference between the two characteristics stems from the fact that the time-varying equalizer reconfigures itself for every processed symbol, whereas the MMSE IC-LE only operates on the basis of an average reliability measure about the data estimates.

4.5.2 Performance with high-order modulations over low to moderate ISI channels

In order to demonstrate the ability of the MMSE IC-LE turbo-equalizer to deal efficiently with configurations untractable with a standard MAP turbo-equalizer, we simulated the three coded modulation schemes of chapter 2 over the 11-taps Proakis A channel model, with impulse response $[0.04, -0.05, 0.07, -0.21, -0.5, 0.72, 0.36, 0, 0.21, 0.03, 0.07]$ [138].

Considering either an 8-PSK or a 16-QAM signal set, applying a MAP equalizer in this context would require working with trellises with more than 10^9 states, a configuration which is clearly prohibitive for practical systems. In contrast, we used here an equalizer with only $N = 15$ coefficients and a restitution delay $\Delta = 10$. As in the previous chapter, random interleavers of size 8196 (symbols for the TCM, or coded bits for the BICM schemes) were used. 10 iterations were performed in each case. The simulation results are shown in figures 4.7 to 4.9. We observe that the MMSE IC-LE turbo-equalizer is able to eliminate all ISI at moderate SNR values and thus reaches the ideal performance of the ISI-free channel both with the TCM and the BICM-ID schemes. We have to mention that similar results were presented in [46]. These results suggest that the MMSE IC-LE turbo-equalizer may realize an attractive low-complexity receiver offering good performance for bandwidth-efficient transmissions with coded modulation over ISI channels. However, we must emphasize that in spite of its long impulse response, the Proakis A channel model is relatively easy to equalize. Hence, the previous conclusion should be nuanced for more severe ISI channels, as we shall see in the next subsection.

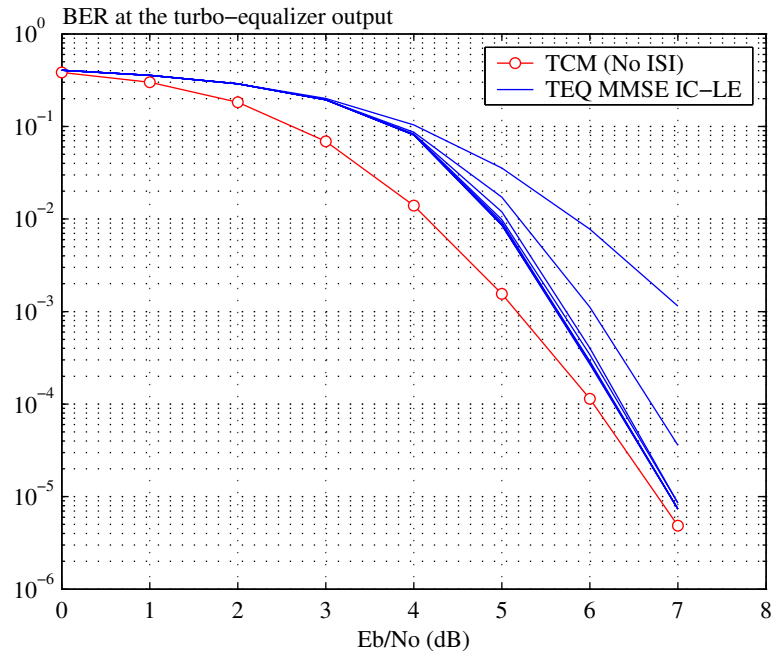


Figure 4.7: BER performance for MMSE turbo-equalization with the 8-PSK TCM scheme over the Proakis A channel model (10 iterations).

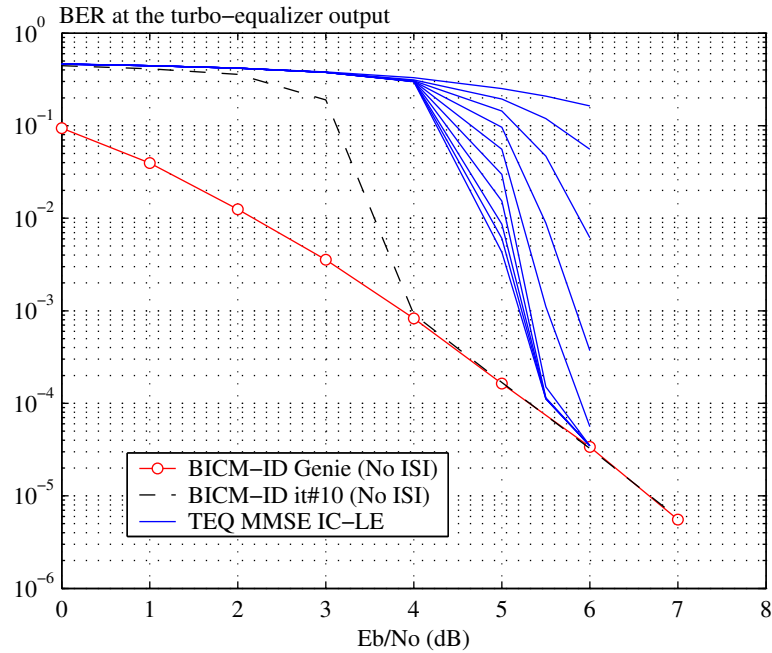


Figure 4.8: BER performance for MMSE turbo-equalization with the 8-PSK BICM-ID scheme over the Proakis A channel model (10 iterations).

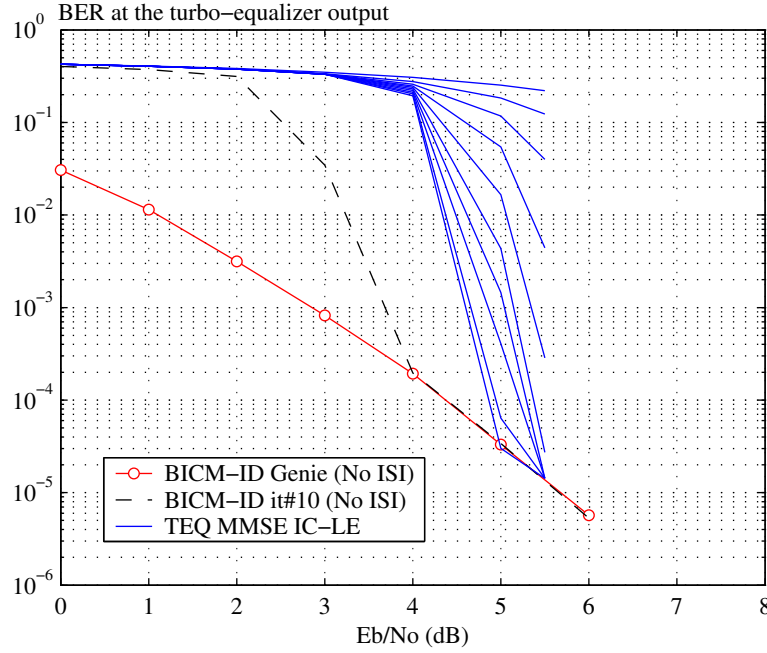


Figure 4.9: BER performance for MMSE turbo-equalization with the 16-QAM BICM-ID scheme over the Proakis A channel model (10 iterations).

4.5.3 Performance with high-order modulations over severe ISI channels

In order to measure the efficiency of the MMSE IC-LE turbo-equalizer over severe ISI channels, we finally simulated the two BICM-ID schemes of chapter 2 over the 3-tap Proakis B channel, a model that was already considered for MAP turbo-equalization schemes in chapter 3, section 3.4. We used an equalizer with $N = 15$ coefficients and a restitution delay $\Delta = 8$. Again, a random interleaver of size 8196 coded bits was considered, and 10 iterations were performed. The simulation results are shown in figures 4.10 and 4.11. For comparison purposes, the performance obtained in section 3.4 with a MAP turbo-equalizer after 10 iteration has also been reported.

From these simulation results, we observe an important SNR gap of 10 dB for the 8-PSK BICM-ID and 8 dB for the 16-QAM BICM-ID respectively, at a BER of 10^{-4} , between the performance obtained with the MMSE IC-LE turbo-equalizer and that obtained with a MAP turbo-equalizer. This gap is supported by the EXIT chart analysis shown in figures 4.12 and 4.13, where we note that the convergence theoretically occurs at E_b/N_0 values between 14 and 15 dB for the 8-PSK BICM-ID, and between 12 and 13 dB for the 16-QAM BICM-ID.

Several remarks are in order here. First, the MMSE IC-LE is penalized by the poor performance of the MMSE LE at the first iteration on this severe channel. This penalty is all the more important

that high-order modulations are considered, as shown by the EXIT charts. In addition, **the choice of set-partitioning labelling for the underlying BICM-ID schemes delays the convergence of the iterative process**. To illustrate this phenomenon, we show in figure 4.14 the EXIT charts obtained with a MAP turbo-equalizer and the MMSE IC-LE turbo-equalizer over the Proakis B channel respectively, but with an 8-PSK BICM scheme with Gray Mapping. We observe that the convergence gap between MAP turbo-equalization and MMSE turbo-equalization reduces from 10 dB to only 4 dB in this case, thanks to the use of Gray mapping. Finally, it must be stressed that the Proakis B is merely a theoretical channel model, and such models seldom arise in many practical transmission scenarios. In fact, our simulations performed in the EDGE context have shown that the MMSE IC-LE offers attractive performance over a wide range of channel conditions [103].

In summary, MMSE turbo-equalization presents an interesting tradeoff between complexity and performance, especially for coded transmissions with non-binary modulations and over long delay spread channels with low to moderate ISI. These conclusions should be nuanced over severe ISI channels. In this case, **it may be desirable to employ BICM with standard Gray mapping in order to accelerate the convergence of the iterative process**.

More generally, designing low-complexity turbo-equalization schemes that are robust to severe ISI channels and able to cope with long delay spread channels as well still constitutes an open research topic.

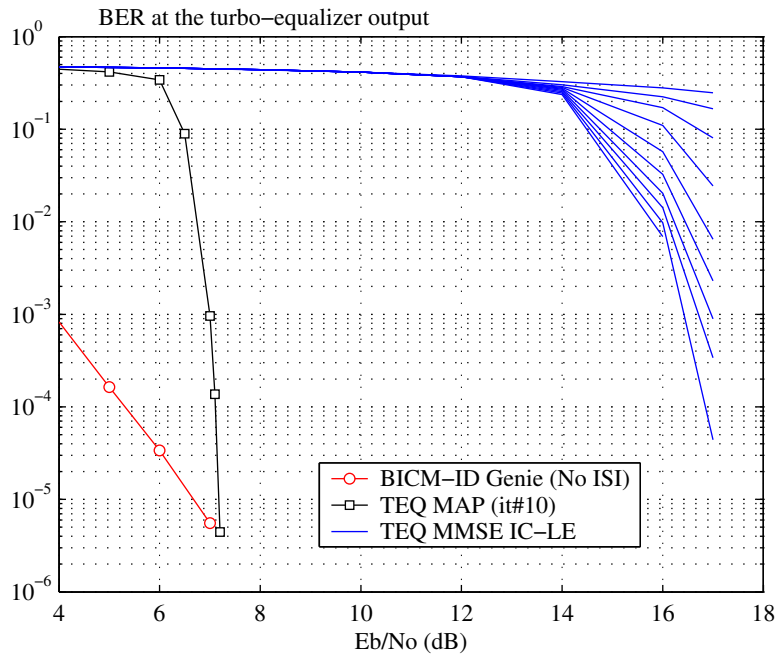


Figure 4.10: BER performance for MMSE turbo-equalization of the 8-PSK BICM-ID scheme over the Proakis B channel model, with 10 iterations.

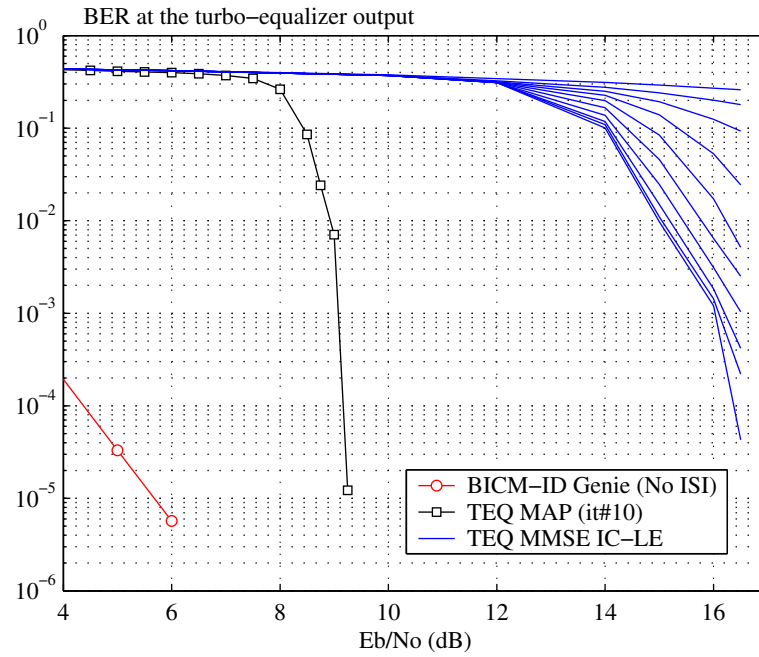


Figure 4.11: BER performance for MMSE turbo-equalization of the 16-QAM BICM-ID scheme over the Proakis B channel model, with 10 iterations.

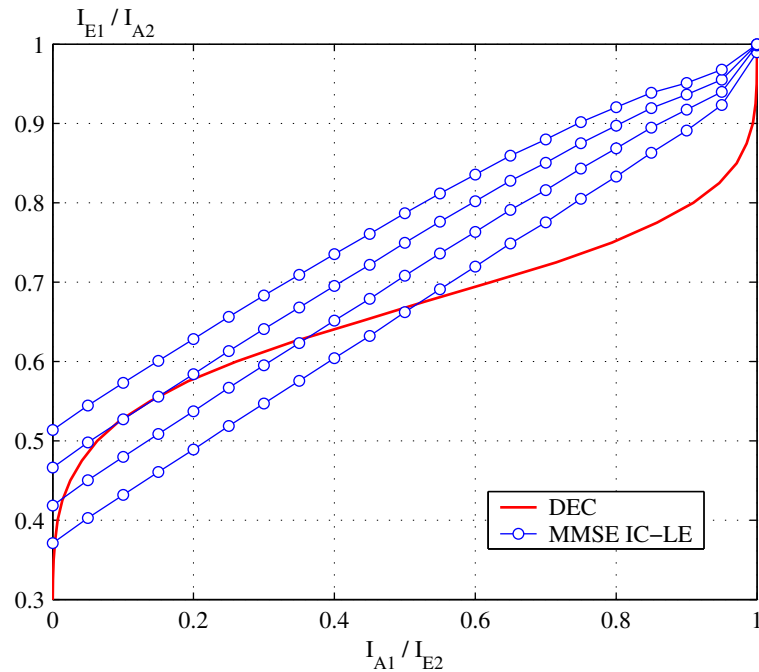


Figure 4.12: EXIT chart for MMSE turbo-equalization of the 8-PSK BICM-ID scheme over the Proakis B channel model. The equalizer characteristics are presented for E_b/N_0 values ranging from 10 dB (bottom) to 16 dB (top) in steps of 2 dB.

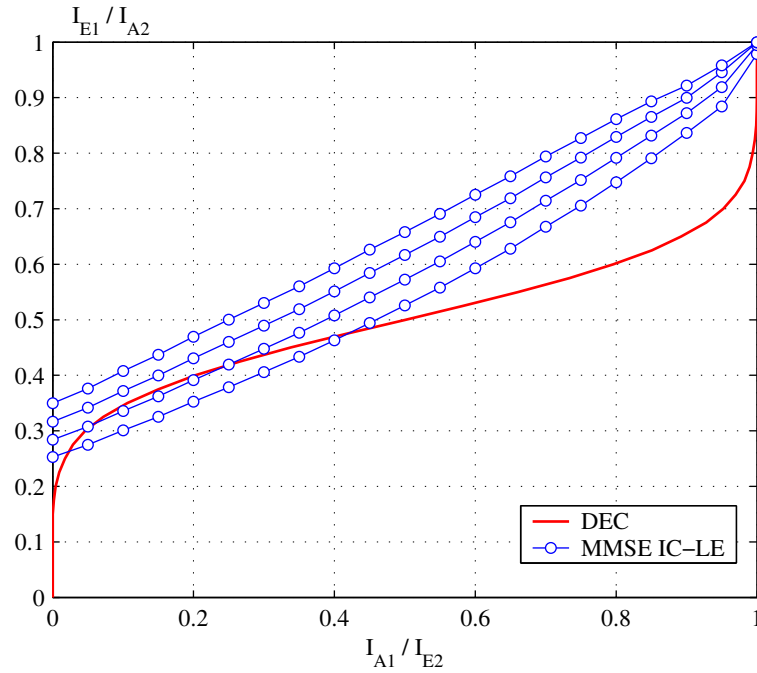


Figure 4.13: EXIT chart for MMSE turbo-equalization of the 16-QAM BICM-ID scheme over the Proakis B channel model. The equalizer characteristics are presented for E_b/N_0 values ranging from 10 dB (bottom) to 16 dB (top) in steps of 2 dB.

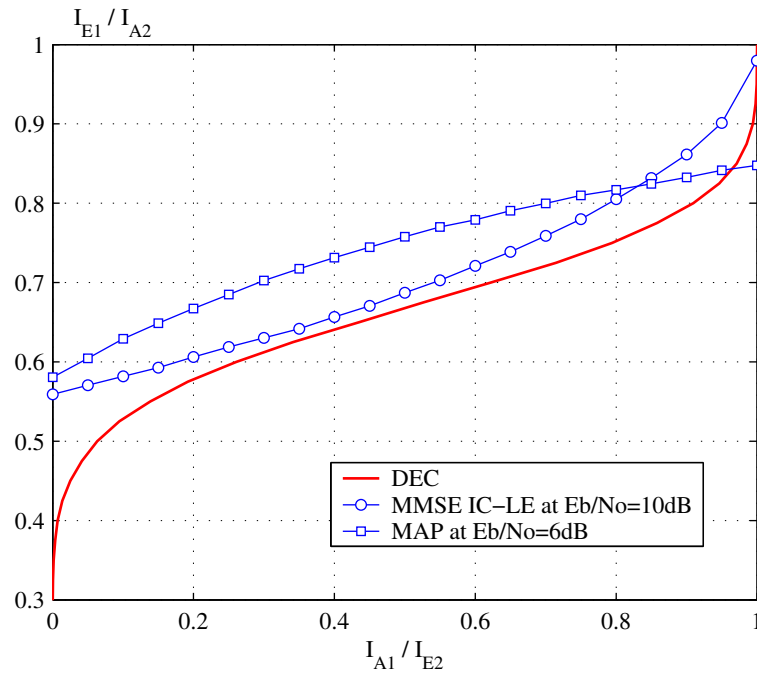


Figure 4.14: EXIT chart for MAP and MMSE turbo-equalization of the 8-PSK BICM-ID scheme with Gray mapping over the Proakis B channel model.

4.6 Frequency-domain MMSE turbo-equalization

It is not uncommon today to encounter multipath channels spanning several tens of symbol periods. Such a situation arises for example in broadband cellular wireless access systems where transmission occurs at rates of several megabits per second, and where multiple echoes of the transmitted signal may be received with delay spreads of up to tens of microseconds [55]. In this case, single-carrier transmission with time-domain equalization may require costly filtering operations, involving 50 or more coefficients to deal with the resulting ISI, and thus become unattractive with respect to the solution offered by multicarrier modulation. However, it has been pointed out by several researchers that single-carrier transmission with frequency-domain equalization (SC-FDE) may offer a competitive alternative to multicarrier modulation methods with essentially the same overall complexity (see for example [149] and [59]). As an illustration, the IEEE 802.16 standardization group has recently approved a physical layer specially designed for SC-FDE in the context of broadband wireless metropolitan area networks (MANs) operating in the 2-11 GHz band and over non-line-of-sight conditions [89]. Such transmission scenarios may constitute potential applications for low-complexity MMSE turbo-equalizers.

In this section, we first derive the frequency-domain version of the MMSE IC-LE turbo-equalizer. Then, we take a close look at the computational savings that may be achieved with a full frequency-domain implementation in the presence of long-delay spread channels, by comparison with the more traditional time-domain realization considered in the previous sections. Our analysis extends some results previously reported by Tüchler in [173].

4.6.1 Derivation of the frequency-domain MMSE IC-LE

The philosophy of frequency-domain equalization is the following. Rather than computing the equalizer coefficient from a channel estimate and then performing the equalization task by filtering in the time-domain, the whole procedure may be realized entirely in the frequency-domain. As with OFDM, frequency-domain equalization requires the insertion of a cyclic prefix at the beginning of each transmitted block. The cyclic prefix is a simple copy of the last data symbols in a block (figure 4.15). Its length should be chosen long enough to cope with the maximum delay spread of the channel. The role of the cyclic prefix is twofold. It first acts as a guard interval and prevents inter-block interference (IBI) between successive blocks at the receiver side. In addition, it makes the received block appear to be periodic with a period equal to the block size, so that the time-domain linear convolution with the channel impulse response is turned into a circular convolution, which corresponds to multiplications in the frequency-domain. The cyclic prefix is finally discarded at the receiver input before the equalization process.

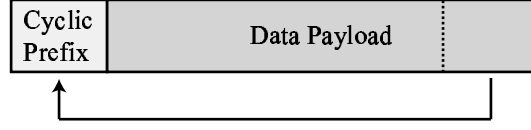


Figure 4.15: Construction of the cyclic prefix.

Let us consider the transmission of blocks of size N data symbols. Mathematically, the corresponding channel observations can be written in vector form as follows at the receiver input, after suppression of the cyclic prefix

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{w} \quad (4.47)$$

where $\mathbf{y}$, $\mathbf{x}$ and $\mathbf{w}$ are $N \times 1$ vectors, and where $\mathbf{H}$ is the square $N \times N$ *circulant* matrix given by

$$\mathbf{H} = \begin{pmatrix} h_0 & h_1 & \dots & h_{L-1} & 0 & \dots & 0 \\ 0 & h_0 & h_1 & \dots & h_{L-1} & & \vdots \\ \vdots & & \ddots & & & \ddots & \vdots \\ h_1 & \dots & h_{L-1} & \dots & \dots & \dots & h_0 \end{pmatrix} \quad (4.48)$$

where each row is deduced from a right-shift of its predecessor. We have dropped for convenience the time subscript n in our vector notation since we operate here on a data block basis.

Suppose now that we take the N -point Discrete Fourier Transform (DFT) of the received sequence $\{y_n\}$. It is well known that every circulant matrix $\mathbf{H}$ can be diagonalized by the Discrete Fourier Transform (DFT) matrix, i.e. it holds that $\mathbf{H}_F = \mathbf{F}\mathbf{H}\mathbf{F}^H$, where $\mathbf{F}$ is the DFT matrix of size N

$$\mathbf{F}_{i,k} = \frac{1}{\sqrt{N}} \exp - \left[\frac{j2\pi ik}{N} \right] \quad i, k = 0, \dots, N-1 \quad (4.49)$$

which satisfies $\mathbf{F}\mathbf{F}^H = \mathbf{I}$, and where $\mathbf{H}_F$ is an $N \times N$ diagonal matrix whose eigenvalues are the samples $\{H_k\}$ of the DFT of the first row of $\mathbf{H}$. If we define the frequency-transformed vectors

$$\mathbf{y}_F = \mathbf{F}\mathbf{y}, \quad \mathbf{x}_F = \mathbf{F}\mathbf{x}, \quad \mathbf{w}_F = \mathbf{F}\mathbf{w} \quad (4.50)$$

that is, $\{\mathbf{y}_F, \mathbf{x}_F, \mathbf{w}_F\}$ are the DFTs of the vectors $\{\mathbf{y}, \mathbf{x}, \mathbf{w}\}$, then it is straightforward to verify that the following relation holds

$$\mathbf{y}_F = \mathbf{H}_F \mathbf{x}_F + \mathbf{w}_F \quad (4.51)$$

or, equivalently,

$$Y_k = H_k X_k + W_k \quad k = 0, \dots, N-1 \quad (4.52)$$

where we have denoted by $\{Y_k\}, \{X_k\}, \{W_k\}$ the respective N -points DFT of the discrete-time sequences $\{y_n\}, \{x_n\}$, and $\{w_n\}$.

Operating in the frequency-domain on a block per block basis, MMSE interference cancellation takes the following form

$$Z_k = P_k Y_k - Q_k \bar{X}_k \quad k = 0, \dots, N-1 \quad (4.53)$$

where $\{Z_k\}$ is the DFT of the equalized sequence, $\{\bar{X}_k\}$ is the DFT of the sequence of soft data estimates, and $\{P_k\}$ and $\{Q_k\}$ are the respective DFTs of the feedforward and feedback filters. The time-domain equalized sequence $\{z_n\}$ is finally obtained by taking the IDFT of $\{Z_k\}$. Taking into account the statistical properties established in section 4.3.1, we thus have to find the optimum discrete-frequency responses $\{P_k\}$ and $\{Q_k\}$ which minimize the mean-square error between the sequences $\{Z_k\}$ and $\{X_k\}$, that is

$$(P_k, Q_k) = \arg \min_{(P_k, Q_k)} E \left(|Z_k - X_k|^2 \right) \quad \text{for all } k = 0, \dots, N-1 \quad (4.54)$$

This optimization problem is solved in appendix A.4. We summarize the parameters of the resulting frequency-domain equalizer in table 4.5. ε^2 denotes the total minimum mean-square error at the equalizer output.

Parameters
$\tilde{P}_k = H_k^* / [(\sigma_x^2 - \sigma_{\bar{x}}^2) H_k ^2 + \sigma_w^2]$ for $k = 0, \dots, N-1$
$\tilde{\mu} = (1/N) \sum_{k=0}^{N-1} H_k \tilde{P}_k$
$\lambda = \sigma_x^2 / (1 + \sigma_{\bar{x}}^2 \tilde{\mu})$
$P_k = \lambda \tilde{P}_k$ for $k = 0, \dots, N-1$
$\mu = \lambda \tilde{\mu}$
$Q_k = H_k P_k - \mu$ for $k = 0, \dots, N-1$
$v^2 = \sigma_x^2 \mu (1 - \mu)$
$\varepsilon^2 = \sigma_x^2 (1 - \mu)$

Table 4.5: Parameters for the frequency-domain MMSE IC-LE.

4.6.2 Complexity issues

In order to compare the respective complexity of the time- and frequency-domain implementation of the MMSE IC-LE equalizer, a theoretical estimation of the computational cost has been conducted for the two approaches. We focus here only on the computation of the equalizer coefficients and the filtering operations, and therefore purposely omit the soft mapping and demapping operations since they are identical for the two solutions. In the following discussions, the parameters N , L and M denote the size of a data block, the number of channel taps and the number of coefficients for the feedforward filter $\mathbf{p}$ respectively. The latter only appears with the time-domain implementation.

The frequency-domain approach

Table 4.6 summarizes the main steps of the frequency-domain (FD) MMSE IC-LE algorithm, and provides an estimation of the corresponding number of operations (real additions, multiplications and reciprocal computation⁴). Note that these results were obtained by considering that a complex FFT on N points requires $2N\log_2 N$ real multiplications and $3N\log_2 N$ real additions [132].

Building upon the contents of this table, a raw estimation of the overall computational cost can be obtained by flop counting. This estimation is targeted towards an implementation on a digital signal processing (DSP) device, so that we assign a default unitary cost of 1 flop to real additions and multiplications, which are both DSP elementary operations. In contrast, division is not a native DSP instruction and usually has to be emulated in software. Consequently, we assign a cost of 40 flops to divisions. This choice is consistent with the number of cycles required to perform a division on current fixed-point DSP devices [169]. These approximations yield the following estimation for the implementation's cost $\mathcal{C}_{FD}$ of the frequency-domain equalizer

$$\mathcal{C}_{FD} = N(20\log_2 N + 66) + 84 \quad \text{flops} \quad (4.55)$$

We observe that the complexity is a function of the sole parameter N . More precisely, the computational cost grows asymptotically like $O(N\log_2 N + N)$ with respect to N .

The time-domain approach

A similar analysis has been carried out for the time-domain implementation of the MMSE IC-LE. The equalizer is realized according to the structure shown in figure 4.3 since it requires fewer coefficients than the standard interference cancellation structure. In addition, computation of the filter coefficients takes advantage of the low-complexity approximate method exposed in section 4.3.4, relying on the FFT. Table 4.7 summarizes the main steps of the time-domain (TD) implementation of the MMSE IC-LE, together with the corresponding number of operations. We mention that the following proposition has been exploited to estimate the number of operations required by complex FIR filtering.

Proposition 4.2. *Filtering a complex discrete-time sequence of length N by a FIR with M complex coefficients requires $4NM$ real multiplications and $2N(2M - 1)$ real additions.*

Proof. Assuming that the FIR memory is properly initialized, the filtering operation requires NM complex multiplies and $N(M - 1)$ complex additions. Accounting for the fact that a complex addition involves 2 real additions and that a complex multiply involves 4 real multiplies and 2 real additions, we obtain the claimed result. $\square$

⁴The reciprocal of x is $1/x$.

Using the results of table 4.7 and again assigning a default cost of 1 flop to real additions and multiplications, and 40 flops to reciprocal computations, we obtain the following estimation for the overall cost $\mathcal{C}_{TD}$ of the time-domain approach

$$\mathcal{C}_{TD} = M(10\log_2 M + 8N + 52) + N(8L - 2) + 84 \quad \text{flops} \quad (4.56)$$

We note that the complexity now depends on the three system parameters N , M and L . In particular, the computational cost grows linearly with respect to N and L , and like $O(M\log_2 M + M)$ with respect to parameter M .

Comparison and discussion

In order to compare the complexity of the time-domain and frequency-domain implementation, we have computed the theoretical number of flops required by the two approaches using (4.55) and (4.56) for increasing values of the length L of the channel impulse response. A fixed data blocksize of 512 symbols was assumed. For the time-domain implementation, the number of coefficients M was set to $2L$ for the feedforward filter. This is a conservative choice that should realize a satisfying tradeoff between performance and complexity over a wide range of channel conditions. The results are shown in figure 4.16.

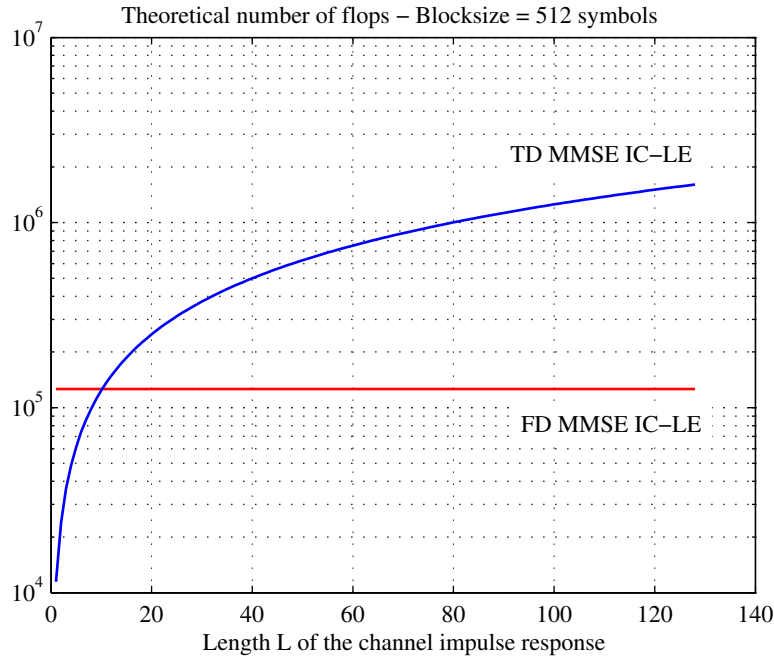


Figure 4.16: Complexity comparison between the time-domain and frequency-domain realizations of the MMSE IC-LE.

• **Inputs:**

- N sampled observations $\{y_n\}$ and N soft estimates $\{\bar{x}_n\}$
- the noise variance σ_w^2 and L channel taps $\{h_\ell\}$
- the average variance σ_x^2 of the data estimates

• **Algorithm:**

1. Frequency-domain transposition

- (a) $\{Y_n\} = \text{FFT}_N\{y_n\}$
- (b) $\{H_n\} = \text{FFT}_N\{h_n\}$
- (c) $\{\bar{X}_n\} = \text{FFT}_N\{\bar{x}_n\}$

2. Feedforward filter and bias computation

- (a) $P'_n = H_n^* / [(\sigma_x^2 - \sigma_{\bar{x}}^2) |H_n|^2 + \sigma_w^2], \quad n = 0, \dots, N-1$
- (b) $\tilde{\mu} = \frac{1}{N} \sum_{n=0}^{N-1} H_n P'_n$
- (c) $\lambda = 1 / (1 + \tilde{\mu} \sigma_{\bar{x}}^2)$
- (d) $P_n = \lambda P'_n, \quad n = 0, \dots, N-1$
- (e) $\mu = \lambda \tilde{\mu}$

3. Feedback filter computation

$$Q_n = H_n P_n - \mu, \quad n = 0, \dots, N-1$$

4. Frequency-domain equalization

$$Z_n = P_n Y_n - Q_n \bar{X}_n, \quad n = 0, \dots, N-1$$

5. Time-domain transposition

$$\{z_n\} = \text{IFFT}_N\{Z_n\}$$

• **Outputs:**

- the equalized sequence $\{z_n\}$ and the equalizer bias μ

- **Note:** (IFFT) $_N$ = complex (inverse) FFT on N points

Step	#Add.	#Mul.	#1/x
1(a)-1(c)	$9N \log_2 N$	$6N \log_2 N$	-
2(a)	$2N + 1$	$5N$	$N + 1$
2(b)	N	$2N$	-
2(c)	1	1	1
2(d)	-	$2N$	-
2(e)	-	1	-
3	$2N$	$2N$	-
4	$4N$	$6N$	-
5	$3N \log_2 N$	$2N \log_2 N$	-
Total	$N(12 \log_2 N + 9) + 2$	$N(8 \log_2 N + 17) + 2$	$N + 2$

Table 4.6: Computation steps and theoretical number of operations for the FD MMSE IC-LE.

• **Inputs:**

- N sampled observations $\{y_n\}$ and N soft estimates $\{\bar{x}_n\}$
- the noise variance σ_w^2 and L channel taps $\{h_\ell\}$
- the variance σ_x^2 of the soft estimates

• **Algorithm:**

1. Frequency-domain transposition of the channel taps

$$\{H_n\} = \text{FFT}_M\{h_n\}$$

2. Feedforward filter and bias computation

$$(a) P'_n = H_n^* / [(\sigma_x^2 - \sigma_{\bar{x}}^2) |H_n|^2 + \sigma_w^2], \quad n = 0, \dots, M-1$$

$$(b) \tilde{\mu} = \frac{1}{M} \sum_{n=0}^{M-1} H_n P'_n$$

$$(c) \lambda = 1 / (1 + \tilde{\mu} \sigma_{\bar{x}}^2)$$

$$(d) P_n = \lambda P'_n, \quad n = 0, \dots, M-1$$

$$(e) \mu = \lambda \tilde{\mu}$$

3. Feedforward filter's impulse response

$$\{p_n\} = \text{IFFT}_M\{P_n\}$$

4. Time-domain equalization

$$(a) \tilde{z}_n = y_n - \sum_{\ell=0}^{L-1} h_\ell \bar{x}_{n-\ell}, \quad n = \Delta, \dots, N + \Delta - 1$$

$$(b) z_n = \sum_{k=0}^{M-1} p_k \tilde{z}_{n-k}, \quad n = \Delta, \dots, N + \Delta - 1$$

• **Outputs:**

- the equalized sequence $\{z_n\}$ and the equalizer bias μ

• **Notes:**

- $(\text{I})\text{FFT}_M$ = complex (inverse) FFT on M points
- The equalizer introduces a restitution delay $\Delta = M/2$ samples

Step	#Add.	#Mul.	#1/x
1	$3M \log_2 M$	$2M \log_2 M$	-
2(a)	$2M + 1$	$5M$	$M + 1$
2(b)	M	$2M$	-
2(c)	1	1	1
2(d)	-	$2M$	-
2(e)	-	1	-
3	$3M \log_2 M$	$2M \log_2 M$	-
4(a)	$4NL$	$4NL$	-
4(b)	$2N(2M - 1)$	$4NM$	-
Total	$M(6 \log_2 M + 4N + 3) + N(4L - 2) + 2$	$M(4 \log_2 M + 4N + 9) + 4NL + 2$	$M + 2$

Table 4.7: Computation steps and theoretical number of operations for the TD MMSE IC-LE.

As can be seen from (4.55), the cost of the frequency-domain is a sole function of the blocksize N and thus does not depend on the number of channel taps. We observe that the frequency-domain approach begins to offer complexity savings with respect to the time-domain implementation for channels with 15 coefficients or more. With 100 coefficients, the frequency-domain solution is one order of magnitude less complex to implement than its time-domain counterpart.

This analysis demonstrates the potential savings that may be realized by implementing the MMSE IC-LE turbo-equalizer in the frequency-domain in the presence of channels with very long delay spreads. We stress however that our analysis is merely theoretical in the sense that it does not capture all the factors that may influence the choice of a solution for a practical system. In particular, important issues such as the degree of parallelism or the data quantization requirements should be carefully examined to determine the best approach for hardware or software implementation.

4.7 Concluding remarks

This chapter was devoted to the study of low-complexity turbo-equalization structures that may represent an interesting alternative to conventional MAP turbo-equalization for multilevel signaling over long delay spread channels, where the complexity of the latter solution is usually intractable. Specifically, we focused on filtering-based turbo-equalizers where the inner equalization scheme is optimized according to the MMSE criterion.

Several equalizers have been introduced, which share the common property of accounting explicitly for the presence of *a priori* information about the transmitted data (available from the iterative process) to adapt the equalization strategy accordingly. A novel equalizer, the MMSE IC-LE, has been derived both in infinite-length form as well as under finite-length realization constraints. We have furthermore established the equivalence of the finite-length implementation with another similar equalizer introduced earlier in a somewhat heuristic manner in the literature.

Several simulation results have been presented, which demonstrate that MMSE turbo-equalization structures constitute attractive receivers which realize an interesting tradeoff between performance and complexity for bandwidth-efficient communications over long delay spread channels with low to moderate ISI. We have also noticed that such receivers may suffer from performance losses over severe-ISI channels when non-binary modulations are used, depending on the bit-labelling strategy considered. In particular, Gray mapping may be used to reduce the resulting performance gap. Although such channels are seldom encountered in many practical transmission systems, further research work is required to address this issue, either by improving the above-mentioned MMSE equalization structures, or by considering the alternative offered by reduced-state trellis-based SISO equalizers.

Since it is not uncommon today to encounter transmission scenarios where ISI may span up to several tens of symbol periods, we have finally examined the possibility of realizing the turbo-equalization process entirely in the frequency-domain. Having derived the frequency-domain form of the MMSE IC-LE, we have conducted a theoretical estimation of the computational cost required both by a time-domain and frequency-domain implementation of the filtering operations. Our results show that the frequency-domain equalizer may offer significant complexity savings over its time-domain counterpart over very long delay spread channels.

Chapter 5

DSP implementation of the MMSE IC-LE Turbo-Equalizer

Having described and analyzed several receiver structures which combine equalization with decoding in an efficient manner, we now turn our attention to the important problem of realizing such solutions in practice. This chapter exposes the implementation of the low-complexity MMSE IC-LE turbo-equalizer studied in section 4.3 on a fixed-point digital signal processor (DSP). This work was conducted as a proof of concept, motivated by the desire to show that efficient low-complexity turbo-equalization structures are actually manageable on current hardware. Our study constitutes the first step towards the realization of a real-time demonstration platform. This work was supported in part by Texas Instruments ELITE university program.

This chapter is organized as follow. We first describe the development platform that was considered in this study. Then, we provide a general overview of the transmission scenario retained for our application. Recalling Lao Tzu's precept that "*The longest journey starts with but a single step*", we shall restrict ourselves here to a simple bit-interleaved coded modulation scheme combining a 4-state rate one-half convolutional code with QPSK modulation. The DSP implementation of the turbo-equalizer is discussed, with an emphasis on the data representation and computation constraints arising from the use of fixed-point arithmetic. Finally, we examine the storage requirements as well as the maximum bit rate achievable on the DSP. Some experimental results are provided. They show that our receiver suffers no real performance loss with respect to an ideal (unquantized) floating-point implementation.

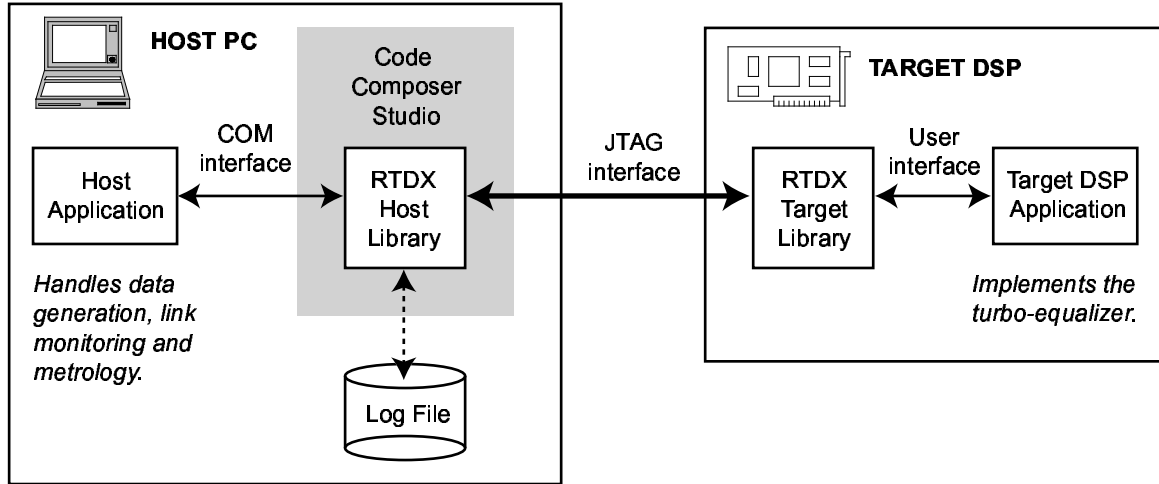


Figure 5.1: Block diagram of the demonstration platform

5.1 Description of the platform

The demonstration platform is composed of a host PC communicating with a target DSP evaluation module, as depicted in figure 5.1. The host PC runs a monitoring application which is responsible for the tasks of configuring the transmission link and the DSP board, generating the data at the channel output, sending the resulting signal to the DSP which implements equalization and decoding, retrieving the processed data, and updating the link metrology (bit error rate, symbol error rate and frame error rate).

The DSP board, shown in figure 5.2, is a stand-alone evaluation card from Spectrum Digital, Inc. It features in particular a Texas Instrument TMS320VC5509 DSP device operating at 120 MHz, 4 megawords of 16-bits on-board DRAM memory, an embedded IEEE 1149.1 JTAG emulator, and various I/O capabilities including a USB port, an ADC/DAC stereo codec, as well as serial and parallel port interfaces.

The TI C5509 device is a high performance low-cost 16-bits fixed-point DSP with low-power dissipation. It is typically targeted towards embedded communications applications such as 2G – 3G cell phones, wireless modems, and multimedia applications (digital audio and still cameras, voice recognition, etc.). Table 5.1 summarizes the main features of the DSP device. Detailed coverage of the DSP capabilities are provided in [167, 168, 170]. This processor manipulates 16-bit fixed-point numbers, and actually supports fixed-point computations with an internal precision of up to 40 bits, which is the maximum length of the accumulator registers. Appendix D provides a review of fixed-point arithmetic. It also introduces useful notations to characterize fixed-point data types that we shall exploit in subsequent sections of this chapter.

C5509 Key Features	
- CPU speed: 120 Mhz (240 MIPS, cycle time = 8.33 ns)	
- Variable instruction length (from 8 bits up to 48 bits)	
- Unified data/program memory architecture	
- Up to 3 memory reads and 2 memory writes in a single CPU cycle	
- 32K words (1 word = 16 bits) of on-chip Dual-Access RAM (DARAM)	
- 96K words of on-chip Single-Access RAM (SARAM)	
- 32K words of on-chip ROM	
- A 40-bit arithmetic logic unit (ALU) in parallel with a second 16-bit ALU	
- Dual multiply-accumulate (MAC) unit (17-bit x 17-bit multipliers with 40-bit adders)	
- 4 40-bit accumulators	
- Dedicated units for Viterbi Add-Compare-Select computation	

Table 5.1: Overview of the key features of the TI C5509 DSP device provided by the EVM

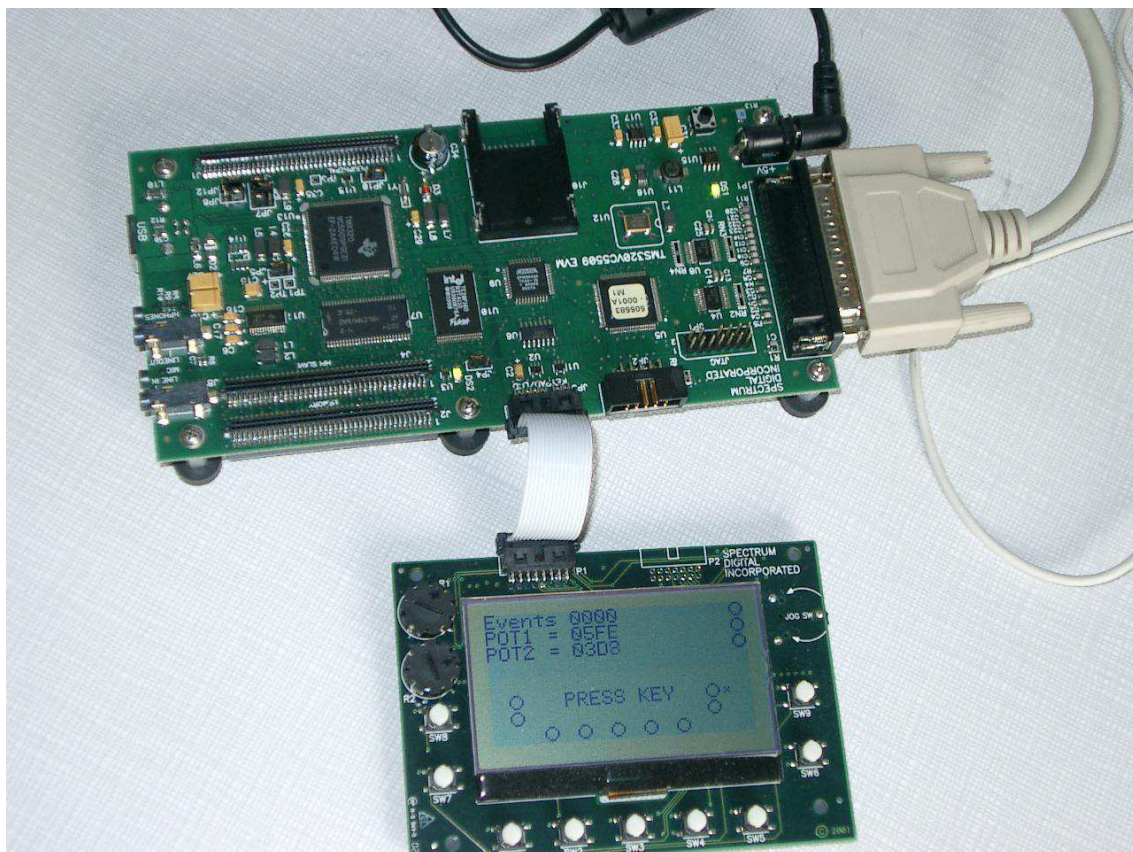


Figure 5.2: Spectrum Digital, Inc. TMS320VC5509 evaluation module

Communication between the DSP board and the host PC uses the *Real-Time Data eXchange* (RTDX) protocol offered by Texas Instrument. RTDX allows developers to transfer data between the host computer and DSP devices throughout the JTAG emulation link, without stopping the target application. A block diagram of the RTDX communication protocol is included in figure 5.1. An RTDX target library runs on the target application. Function calls to this library's application programming interface (API) allow the DSP to transfer data to/from the host in the background, while the target application keeps running. On the host platform, an RTDX host library operates in conjunction with the Code Composer Studio development environment and provides a simple API relying on the Windows Component Object Model (COM) interface to obtain and/or to send data to the target application. RTDX offers transfer data rates typically ranging from 30 to 50 Kb/s, the latest improvement (*High-Speed RTDX*) allowing up to 2Mb/s. Reference [166] provides a comprehensive overview of the RTDX technology.

5.2 Overview of the transmission scheme

Figure 5.3 presents the block diagram of the communication scenario that was considered in our application. This section provides a general overview of the transmitter and receiver functions, as well as discussions about the assumptions made about the channel model for the implementation of the turbo-equalizer.

5.2.1 Transmit processing

The transmitter functions implement a simple bit-interleaved coded modulation scheme. Frames of N_i information bits $\{b_k\}$ are encoded by a rate $1/2$ recursive systematic convolutional (RSC) encoder with memory 2, free distance $d_{\text{free}} = 5$, and generator polynomials $(1, 1 + D^2/1 + D + D^2)$. 2 tailbits are appended at the end of the information sequence in order to force zero-state trellis termination at the end of the message. The effective code rate is thus

$$R_c = \frac{1}{2} \frac{N_i}{N_i + 2} \quad (5.1)$$

This convolutional code was selected because of its low decoding complexity (4 states only). Additionally, the systematic property of the code simplifies SISO decoding since the APP on the code bits are required solely to extract both the hard decisions on the information bits and the extrinsic information on the coded bits in the same pass. The $N_c = 2N_i + 4$ coded bits $\{c_k^i\}$ are interleaved according to a pre-computed S -random permutation function [49] with parameter $S = \lfloor \sqrt{N_c/2} \rfloor$, and mapped onto discrete-time complex QPSK symbols $\{x_n\}$ with zero mean and unit variance $\sigma_x^2 = 1$. A Gray labelling map was selected (figure 5.4).

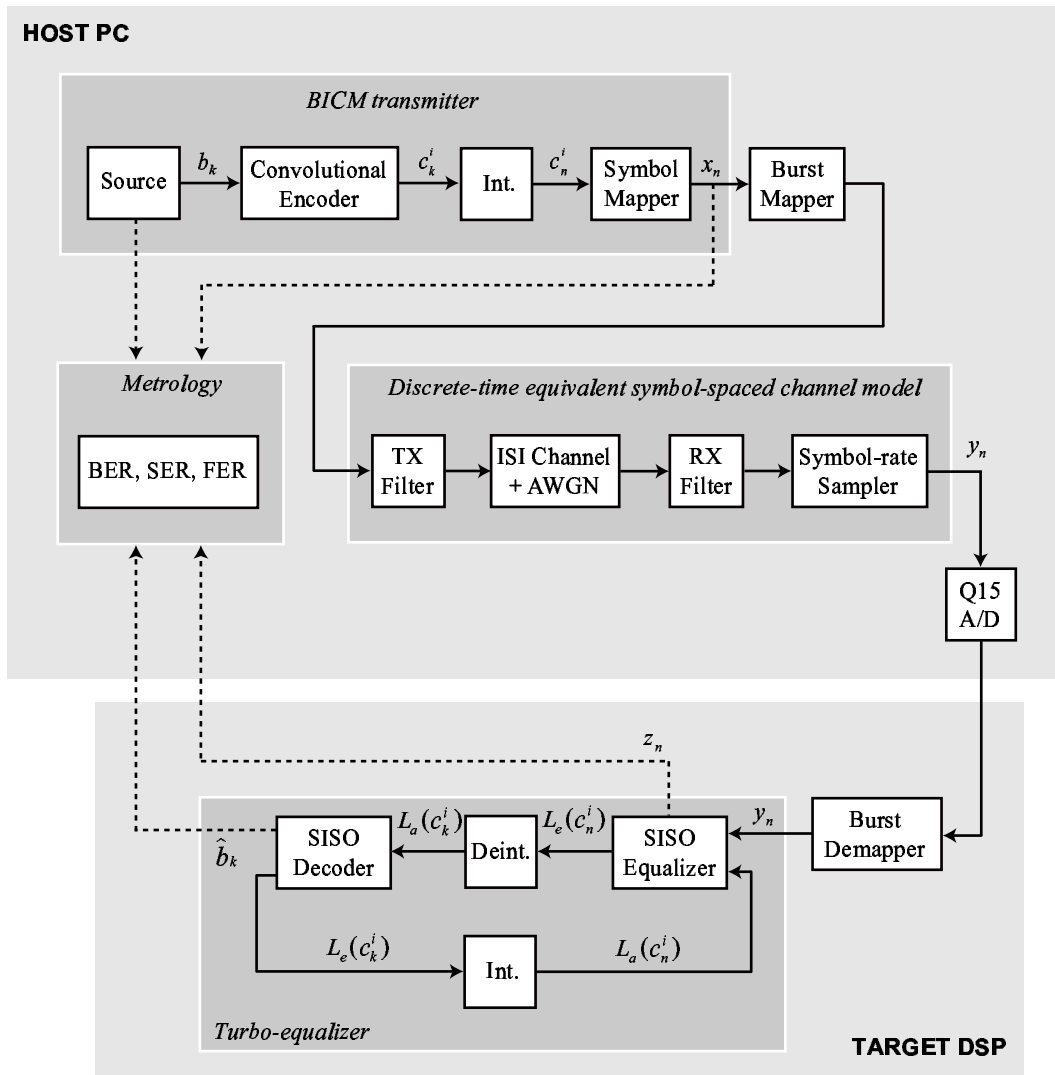


Figure 5.3: Block diagram of the transmission scheme

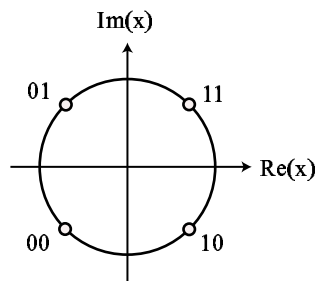


Figure 5.4: QPSK signal set with Gray labelling

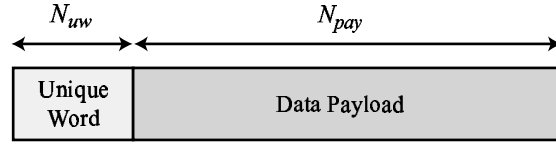


Figure 5.5: Description of the burst format

The discrete-time modulated symbols enter a burst mapping module which formats the incoming data into transmission bursts. The general structure of a burst is described in figure 5.5, and comprises a data payload section of N_{pay} information symbols, prefixed by a *unique word* (UW) with length N_{uw} symbols. The unique word is inserted for channel estimation and frame synchronization purposes. It derives from Frank-Zadoff sequences [68] and thus possesses constant amplitude zero autocorrelation (CAZAC) properties [128]. For best performance, the length of the unique word should be at least as long as the expected channel maximum delay spread. As we shall see in the next subsection, we currently limit ourselves to channel models spanning a maximum of 16 symbol periods. Hence, N_{uw} was set to 16, and the corresponding unique word is

$$UW = \{+1, +1, +1, +1, +1, +i, -1, -i, +1, -1, +1, -1, +1, -i, -1, +i\}$$

In our application, the data payload size was set to $N_{pay} = 512$ QPSK symbols. The corresponding values for the other parameters at the transmitter side are summarized in table 5.2.

Modulation	QPSK
Number of information bits per burst	$N_i = 510$
Number of coded bits per burst	$N_c = 1024$
Number of payload symbols per burst	$N_{pay} = 512$
Size of the unique word	$N_{uw} = 16$

Table 5.2: Parameters of the transmission link

5.2.2 Channel and receiver front-end modeling

Following the conventions introduced in chapter 2, we assume here a coherent demodulator, i.e. perfect carrier and phase offset compensation, as well as perfect timing synchronization. Hence, the discrete-time baseband channel resulting from the cascade of the transmit filter, ISI channel, receive filter and symbol-rate sampler can be modeled as a FIR filter with L symbol-spaced coefficients. The

observation y_n received at time n at the turbo-equalizer input is then given by

$$y_n = \sum_{\ell=0}^{L-1} h_{\ell} x_{n-\ell} + w_n \quad (5.2)$$

where the noise samples w_n are zero-mean, circularly-symmetric, wide-sense stationary uncorrelated Gaussian random variables with total variance σ_w^2 . The receiver front-end is shown in figure 5.3 and consists of a fixed filter matched to the transmit pulse shape, followed by a symbol-rate sampler, a quantizer delivering fixed-point numbers in Q15 representation (see appendix D), and a burst demapper. We currently assume in our application that the discrete-time baseband channel model is time-invariant and spans a maximum number of 16 symbol periods (16 coefficients).

Any practical receiver would include an amplitude gain control (AGC) device, whose role is to maintain the average received power at a desired level by properly adjusting the received signal amplitude. Such an operation is typically realized in continuous-time on analog signals by hardware devices [127]. The choice of the desired average power P_{des} at the input of the quantization device is an important issue for the design of the receiver front-end. The Q15 data representation format applies to real numbers in the range $[-1, +1)$, and values outside this interval must be saturated. It follows that P_{des} must be chosen high enough so that the received data may be quantized with enough precision, while avoiding excess saturation in order for the receiver to operate in a linear A/D conversion regime most of the time. It has been found by simulation that the value $P_{des} = 0.25$ provides a good balance between these two objectives. It is assumed hereafter in this chapter that the AGC scaling factor is tacitly taken into account into the definition of the channel coefficients $\{h_{\ell}\}$ and noise variance σ_w^2 in equation (5.2).

5.2.3 The turbo-equalization receiver

Depicted in figure 5.3, the turbo-equalization scheme is implemented on the C5509 target device. The host PC provides the DSP with the sampled data $\{y_n\}$ composing the received burst, quantized in Q15 representation. In spite of the presence of a training sequence, both perfect channel estimation and frame synchronization are currently assumed at the receiver side.

Before the turbo-equalization scheme, the burst demapping module discards the training sequence and provides the SISO equalizer with the payload data symbols. The SISO equalizer implements the finite-length form of the MMSE IC-LE structure introduced in chapter 4, section 4.3. Regarding the computation of the equalizer coefficients, we suppose that the noise variance σ_w^2 is perfectly known at the receiver side. At the present date, the turbo-equalizer is designed to operate with signal-to-noise ratios in the 0–15 dB range.

For the ease of implementation, a fixed number of 5 iterations was considered in our application. However, a practical system could benefit from the use of an appropriate early stopping criterion, which essentially halts the iterative process when the estimated data reach a given reliability. As the SNR increases, this usually results in substantial throughput gains (or reduced power consumption) without significant performance degradation with respect to the case where the number of iterations is fixed in advance. The design of a proper early stopping criterion is discussed in [21] and [122] for example.

The equalized sequence $\{z_n\}$ as well as hard decisions $\{\hat{b}_k\}$ on the information message are fed back to the host PC for metrology purposes (BER, SER and FER measurements) at each iteration. Table 5.3 summarizes the design parameters that were used in the realization of the turbo-equalizer.

Parameters	Value	Comments
Complex channel impulse response length	16	System parameter
Channel impulse response	-	Assumed available
Transmitted power σ_x^2	1.0	System parameter
Noise variance σ_w^2	-	Assumed available
Number of iterations	5	System parameter

Table 5.3: Design parameters for the turbo-equalizer

5.3 Implementation of the turbo-equalizer

This section describes the fixed-point implementation of the turbo-equalizer on the C5509 device¹. We first expose the general development strategy that was adopted for the implementation. Then, we examine the number of bits required to represent the different quantities involved in the turbo-equalization process with sufficient accuracy. Finally, we discuss the implementation of the different operations realized by the turbo-equalizer, with the exception of the interleaving/deinterleaving functions that are realized using simple look-up table consultation in the DSP RAM.

5.3.1 Strategy of development

We have chosen to implement the turbo-equalizer using the C language solely, in order to speed-up the development process. As a result, our implementation is certainly not optimal from a code performance point of view. In fact, we can expect speed improvements by a factor two or more with a receiver entirely realized in optimized assembly language.

¹Parts of this study have been presented in [108].

The Texas Instruments C55x C compiler provides several *intrinsics*. Intrinsics are special C functions that map directly onto inlined specific C55x assembly language instructions. Examples of intrinsics include saturated additions and subtractions, multiplies, multiply-accumulate (MAC) operations, etc. [169]. Although interesting from a code optimization perspective, the use of intrinsics has been voluntarily kept to a minimum to favor code portability and reusability. In contrast, we have made use whenever possible of the general purpose digital signal processing functions provided by Texas Instrument Digital Signal Processing Library (DSPLIB) [171]. This library includes optimized assembly routines such as FFT, bit-reversing, convolution, FIR and IIR filtering functions, and is currently available for a wide range of TI DSP devices.

5.3.2 Data quantization

The received samples $\{y_n\}$ and channel coefficients $\{h_\ell\}$ are delivered in Q15 representation from the receiver front-end. However, finding the right number of bits required to represent the other quantities involved in the turbo-equalization scheme without sacrificing performance turns out to be a crucial issue in the receiver design. This problem has been addressed in two steps.

First, a campaign of simulations was performed with a floating-point C model of the turbo-equalizer in order to evaluate the dynamics of the values exchanged between the receiver functions. The corresponding fixed-point formats used in our implementation are summarized in table 5.4. We refer the reader to appendix D for a description of the $S(a, b)$ notation.

Quantity	Fixed-point format
Received symbols y_n	Q15
Channel taps h_ℓ	Q15
Noise variance σ_w^2	Q15
Soft symbol estimates $\bar{x}_n$	Q15
Equalized symbols z_n	$S(3, 12)$
Extrinsic LLRs $L_e(c_n^i)$ at the SISO equalizer output	$S(4, 5)$
Extrinsic LLRs $L_e(c_k^i)$ at the SISO decoder output	$S(3, 5)$

Table 5.4: Data quantization table

The previous step has allowed us to specify adequate representations for the quantities in the input and output of each signal processing block in the turbo-equalizer. Second, additional studies regarding the representation of the data manipulated internally by these different functions were then performed case by case.

We now describe in greater details the implementation of the different signal processing blocks composing the turbo-equalizer.

5.3.3 The soft mapping module

We recall that the soft mapping module is in charge of computing the soft symbol estimates $\{\bar{x}_n\}$ from the set of prior LLRs $\{L_a(c_n^i)\}$ delivered by the SISO decoder at the previous iteration. The soft estimates are computed assuming a signal set with unit power $\sigma_x^2 = 1$. Hence, they fit naturally well into the DSP native Q15 representation. The mapping equations are summarized in the upper part of table 5.5.

Soft mapping equations	$\begin{aligned}\operatorname{Re}(\bar{x}_n) &= \frac{1}{\sqrt{2}} \tanh\left(\frac{L_a(c_n^1)}{2}\right) \\ \operatorname{Im}(\bar{x}_n) &= \frac{1}{\sqrt{2}} \tanh\left(\frac{L_a(c_n^2)}{2}\right)\end{aligned}$
Soft demapping equations	$\begin{aligned}L_e(c_n^1) &= \frac{4}{\sqrt{2(1-\mu)}} \operatorname{Re}(z_n) \\ L_e(c_n^2) &= \frac{4}{\sqrt{2(1-\mu)}} \operatorname{Im}(z_n)\end{aligned}$

Table 5.5: Soft mapping and demapping equations for QPSK modulation with Gray labelling.

We note that the mapping operation makes use of the non linear mathematical $\tanh(\ell/2)$ function. Since this function does not exist in the DSP instruction set, it has to be tabulated in RAM. Experiments have shown that a quantization range $\ell \in [-8.0, \dots, +8.0)$ with a quantization step $\Delta_\ell = 1/32$ yield virtually no performance degradation with respect to the unquantized implementation. In addition, this choice has the advantage of matching directly the fixed-point format of the extrinsic LLRs delivered by the SISO decoder, which are quantized in $S(3,5)$ representation. We then obtain a look-up table with 512 entries for the $\tanh(\ell/2)$ function.

The soft symbol mapper also calculates the variance $\sigma_{\bar{x}}^2$ of the soft symbol estimates $\{\bar{x}_n\}$, a quantity which actually measures the reliability of these estimates. $\sigma_{\bar{x}}^2$ is computed using the sample variance estimator, assuming $E(\bar{x}_n) = 0$ and $\sigma_x^2 = 1$. We obtain

$$\sigma_{\bar{x}}^2 = \frac{1}{N_{\text{pay}}} \sum_{n=0}^{N_{\text{pay}}-1} |\bar{x}_n|^2 \quad (5.3)$$

By definition (see section 4.3.1), $\sigma_{\bar{x}}^2$ satisfies the inequalities $0 \leq \sigma_{\bar{x}}^2 \leq 1$. Hence, it is quantized in Q15 representation.

5.3.4 The MMSE IC-LE equalizer

Implementing the MMSE IC-LE equalizer proved to be a challenging task. Obtaining a fixed-point implementation exhibiting a small degradation with regard to an ideal unquantized receiver was all the more important that the overall turbo-equalization scheme critically relies on the equalizer perfor-

mance. A detailed analysis was conducted in order to select the most promising approach for a DSP implementation, between a time-domain and frequency-domain realization of the equalizer [105]. It was found that the two systems have approximately the same complexity in our context. We first investigated the frequency-domain approach. However, accuracy problems were encountered with divisions, owing to important scaling operations required by the (I)FFT operations. Hence, we finally turned our attention to the time-domain implementation of the MMSE IC-LE.

The MMSE IC-LE equalizer was realized using $N_p = 32$ taps for the feedforward filter and $N_q = 32 + 16 - 1 = 47$ taps for the feedback filter respectively. The coefficients were computed using the low-complexity method described in section 4.3.4 and relying on the Fast Fourier Transform. In this case, the MMSE IC-LE introduces a restitution delay $\Delta = 16$. Table 4.7 in the previous chapter summarizes the main steps of the equalization process. Simulations were performed with a floating-point C model of the equalizer in order to determine the number of bits required for the filter taps $\{p_k\}$ and $\{q_k\}$. We found that the $S(3, 12)$ fixed-point format was well-suited to accurately represent these quantities.

5.3.5 The soft demapping module

The soft demapping module extracts extrinsic LLRs $L_e(c_n^i)$ from the equalized samples $\{z_n\}$. The demapping equations are shown in the lower part of table 5.5. Note that the *a priori* LLRs $L_a(c_n^i)$ are not required for a QPSK signal set. In fact, the extrinsic LLRs are computed using 16 bits and finally saturated in $S(4, 5)$ representation to match the fixed-point format expected at the decoder input.

5.3.6 The SISO decoder

The SISO convolutional decoder implements the Max-Log-MAP decoding algorithm [144]. The notations used in this subsection closely follows those introduced in chapter 3 section 3.2.3 to describe the turbo-equalization scheme for BICM.

One-shot decoding is performed here on the whole coded sequence, thus maximizing computation efficiency at the expense of higher storage requirements with respect to a sliding-window implementation. The decoder proceeds in three steps. The backward recursion is performed first, and the resulting backward state metrics $B_k(v)$ at all nodes v in the trellis are stored in RAM. Then, the decoder performs simultaneously the forward recursion and the calculation of the *a posteriori* LLRs $L(c_k^i)$ on the coded bits. During the forward recursion, only the forward path metrics for the previous and current state $A_{k-1}(v)$ and $A_k(v)$ need to be stored. This requires only 8 memory words in our application. Finally, extrinsic LLRs $L_e(c_k^i)$ on the coded bits as well as hard decisions $\hat{b}_k$ on the information bits (thanks to the systematic property of the code) are extracted from the *a posteriori* LLRs. It is worth

noting that no additional storage is required for the transitions metrics $\Gamma_k(v \rightarrow v')$ since they admit here a simple expression and are thus recomputed at each trellis step during the backward and forward recursion.

Since the state metrics $A_k(v)$ and $B_k(v)$ are represented by finite fixed-point values and accumulate during the forward and backward recursions, care must be taken to avoid overflows during the computation and storage of the metrics. Several remedies are known to this problem, the most common approach involving subtracting the value of one metric (usually the one with the largest value) to the other metrics at each step in the trellis. We considered here an interesting alternative approach which has the advantage of not requiring any explicit normalization operation at all.

This method extends techniques used for the Viterbi algorithm to the Max-Log-MAP algorithm. It is well known that for the Viterbi algorithm, the difference between state metrics remains bounded in magnitude by a fixed quantity δ_M at each time instant k (see e.g. [87, 156]). The key idea here is not to invest to avoid overflow, but rather to accommodate the overflow so that it does not affect the accuracy of the results. This is accomplished by using *two's complement* arithmetic (see appendix D) with a number of bits sufficient to represent the maximum difference δ_M . Similar properties have been shown to hold for the Max-Log-MAP and Log-MAP algorithms in [129] and [33] respectively. This solution is particularly attractive since the C5509 fixed-point DSP intrinsically uses two's complement arithmetic over 16 bits for its computations. Hence, the problem here involves finding the minimum number of bits necessary to represent the quantities used in the decoding process so as to make sure that they fit within the 16-bit DSP format. The latter condition prevents overflows during metric accumulation and LLR computation.

Several upper bounds are known for δ_M , which depend on the code characteristics. However, they are usually not tight. Here, we used the experimental method proposed by Boutillon *et al* in [33] to find the exact maximum values assumed by the partial state metrics, *a posteriori* LLRs and extrinsic LLRs during decoding respectively. The following properties were obtained.

Properties: let the input prior LLRs $L_a(c_k^i)$ be in the range $-M \leq L_a(c_k^i) < M$. Then we have

$$-3M \leq A_k(v) < 3M - 1 \quad (5.4)$$

$$-3M \leq B_k(v) < 3M - 1 \quad (5.5)$$

$$-5M \leq L(c_k^i) < 5M - 1 \quad (5.6)$$

$$-4M \leq L_e(c_k^i) < 4M - 1 \quad (5.7)$$

In other words, if n bits are required to represent the input *a priori* LLRs (sign included), the partial state metrics $A_k(v)$ and $B_k(v)$ require in turn $n + 2$ bits, the *a posteriori* LLRs $L(c_k^i)$ require $n + 3$ bits, and the extrinsic $L_e(c_k^i)$ finally require $n + 2$ bits to avoid any overflow during the decoding process.

We stress that these properties only hold for the convolutional code considered in our application. Similar analysis should be carried out for other codes.

Now if we choose to represent the input LLRs with 10 bits, the computation of the *a posteriori* LLRs require 13 bits, a value which remains lower than the 16-bit accuracy offered by the DSP. In this case, normalization is accomplished in a transparent way by the processor. This justifies our choice to quantize the *a priori* LLRs using only 10 bits, in $S(4.5)$ representation, at the decoder input. In this case, the updated extrinsic LLRs delivered by the decoder are in fact internally computed in $S(6.5)$ format. Hence, these quantities have to be clipped to the range $[-9.0, +9.0)$ to match the $S(3.5)$ fixed-point format expected at the input of the soft mapping module.

5.4 System performance

We finally examine in this section the storage requirements as well as the maximum bit rate achievable by our DSP implementation. Some experimental results are also provided in order to measure the performance loss obtained with respect to an ideal (unquantized) floating-point receiver.

5.4.1 Achievable bit-rates and storage requirements

Table 5.6 summarizes the average number of DSP cycles required to perform the different signal processing functions in the turbo-equalizer. These measurements were obtained using the optimization level -o3 of Code Composer Studio C Compiler. The total number of cycles per iteration is also shown.

Function	Average number of cycles
SISO mapping	17258 T_c
Time-domain equalization	115965 T_c
SISO demapping	21835 T_c
Interleaving/Deinterleaving	6157 T_c each
SISO decoding	126977 T_c
Total	294349 T_c

Table 5.6: Average number of DSP cycles per subfunction of the turbo-equalization scheme.

We observe that one iteration of the turbo-equalizer requires in average 294349 DSP cycles. Since one DSP cycle is executed in 8.33 ns (table 5.1), one iteration is performed in about 2.45 milliseconds. Taking into account the fact that one burst conveys 510 information bits, we obtain a maximum achievable information bit-rate of about 207 Kbits/s per iteration, or, equivalently, 41 Kbits/s with 5

iterations. To the best of the author's knowledge, this is the first DSP implementation result reported so far in the literature.

A close look at these timings reveals that the SISO equalizer and decoder account for 53% and 43% of the total running time per iteration respectively. Higher data rates are thus to be expected with proper assembly language programming. In particular, it would be worth optimizing the decoder implementation by taking advantage of the dedicated Add-Compare-Select instructions offered by the C5509 DSP.

Regarding now the storage requirements, our implementation has a code size of 3747 words (we recall that 1 word = 16 bits) and uses an overall amount of 10118 words in data memory. These values are fully compatible with the 32 Kwords of on-chip DARAM available on the C5509 device (table 5.1). We emphasize that no particular attempt was made in order to optimize the storage requirements. In particular, unused memory is not shared between separate processing functions. A clever reuse of memory could yield some savings.

5.4.2 Experimental results

In order to compare the performance of our fixed-point DSP implementation with an ideal (unquantized) floating-point C model of the receiver, simulations were conducted with the two systems over three different channel models:

- the moderate-ISI time-invariant Porat channel model, with complex impulse response

$$\mathbf{h} = \{2 - 0.4j, 1.5 + 1.8j, 1, 1.2 - 1.3j, 0.8 + 1.6j\}$$

- the severe-ISI time-invariant Proakis C channel model, with impulse response

$$\mathbf{h} = \{0.227, 0.460, 0.688, 0.460, 0.227\}$$

- the time-varying quasi-static EQ6 channel model, with 6 taps of equal average power

The results are shown in figures 5.6 to 5.8 at the first and fifth iterations respectively. We observe that our fixed-point DSP receiver exhibits virtually no performance degradation in comparison with the floating-point implementation. These results were further confirmed by additional simulations over other channel models. We recall that better performance could be obtained over the Proakis C channel by considering larger interleavers (see for example the simulation results in section 4.5.1).

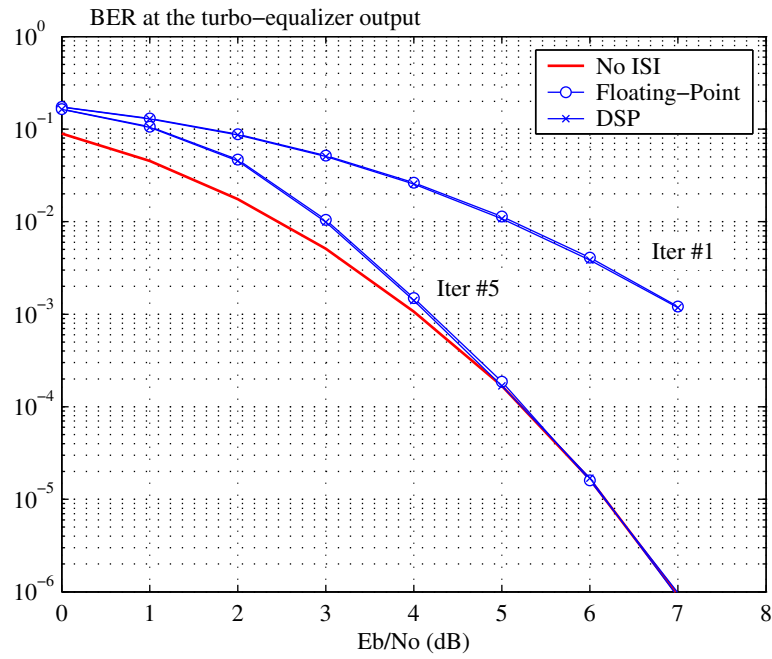


Figure 5.6: BER performance with 5 iterations over the Porat static channel model.

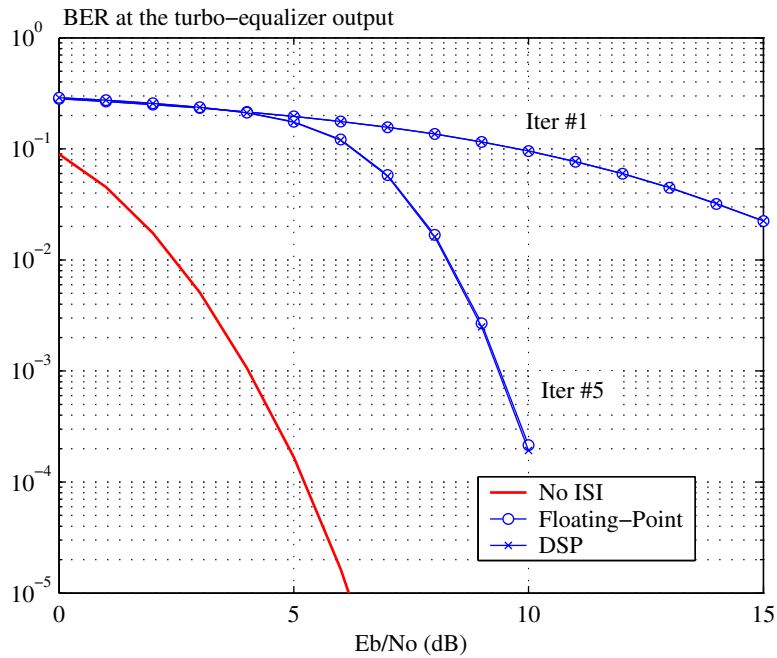


Figure 5.7: BER performance with 5 iterations over the Proakis C static channel model.

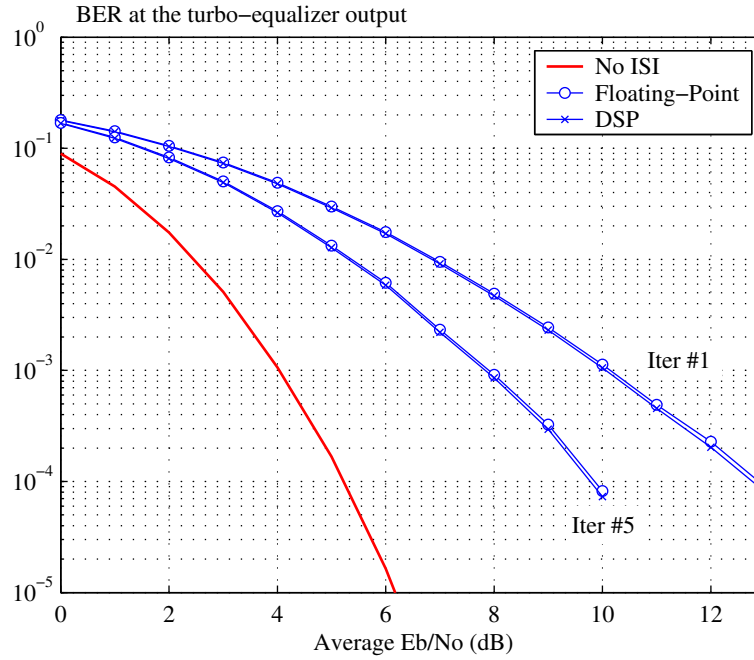


Figure 5.8: BER performance with 5 iterations over the EQ6 quasi-static channel model.

5.5 Concluding remarks

We have described in this chapter the implementation of the MMSE IC-LE turbo-equalizer on a low-cost fixed-point DSP with low power consumption. Using only C programming, a promising data rate of 41 Kbits/s has been achieved with 5 iterations. In addition, the fixed-point implementation does not exhibit any performance loss with respect to an ideal unquantized receiver. These results clearly demonstrate the possibility of implementing low-complexity turbo-equalization structures in practice on portable terminals.

This study constitutes the first step towards the realization of a demonstration platform. However, much work remains to complete this objective. In particular, a carefully optimized assembly language implementation will be necessary in order to further improve the data rate. Estimation of the channel characteristics (noise variance and channel coefficients) as well as frame synchronization will also have to be included.

The implementation of this turbo-equalizer on Field Programmable Gate Arrays (FPGA) constitutes another interesting perspective with a view to achieving higher data rates compatible with the requirements of broadband wireless transmissions. A flexible solution in this context involves realizing the computation of the equalizer coefficients on a general purpose DSP core, and then performing the costly filtering operations directly in hardware [147, chap. 5]. The DSP core may then be used to

perform other modem functions as well. However, hardware complexity is an increasing function of the number of bits used to represent the data. This calls for additional studies in order to determine the minimum wordlength that is really required at each stage of the turbo-equalization process without compromising the overall performance too much.

Chapter 6

Conclusions

Summary

This work was motivated by the desire to achieve reliable transmission with bandwidth-efficient coded modulation schemes over frequency-selective channels.

We have seen in chapter 3 that an optimal receiver realizing equalization and decoding in a joint manner is prohibitively complex to implement in the presence of interleaving. Hence, the conventional approach involves performing the equalization task first, followed by the decoding operation and preferably with an exchange of soft information in between the two functions. We have then introduced the turbo-equalization scheme, where equalization and decoding are combined in an iterative process so that each operation benefits from the information delivered by the other. Significant performance gains have been observed with respect to the conventional approach. We have investigated the theoretical asymptotic performance of the turbo-equalizer in the presence of perfect *a priori* information (an assumption that holds at high SNR) and formally proved that the iterative scheme then essentially converts the ISI channel into an ISI-free channel without noise enhancement, thereby achieving the ideal matched-filter bound. For the particular case of a fully-interleaved multipath Rayleigh fading channel, we have also shown that the turbo-equalizer is able to exploit the diversity offered both by the multipath propagation and by the outer coding scheme. We have finally applied a semi-analytical analysis tool, the EXIT chart, to the turbo-equalization scheme in order to gain more insight into the convergence behavior of the iterative process. We have examined in particular the influence of the channel and outer convolutional code characteristics, and observed that there exists a trade-off to find between good asymptotic performance and early convergence of the turbo-equalization process.

Although the turbo-equalizers studied in chapter 3 constitute attractive receivers in the context of bandwidth-efficient transmissions over ISI channels, they rely on trellis-based optimum MAP equaliz-

ers whose complexity becomes rapidly untractable as the number of channel taps increases, especially when high-order modulations are considered (which is typical of bandwidth-efficient transmission schemes). Hence, reduced-complexity turbo-equalization schemes have been investigated in chapter 4. Several filtering-based equalizers optimized according to the MMSE criterion have been introduced, which share the common property of accounting explicitly for the presence of *a priori* information about the transmitted data (available from the iterative process) to adapt the equalization strategy accordingly. A novel equalizer, the MMSE IC-LE, has been derived both in infinite-length form as well as under finite-length realization constraints. We have furthermore established the equivalence of the finite-length implementation with another similar equalizer introduced earlier in a somewhat heuristic manner in the literature. Our analyses have shown that MMSE turbo-equalizers offer an attractive alternative to MAP turbo-equalizers for multilevel coded transmissions over long delay spread channels. In addition, channels with a large (50 or more) number of coefficients can be efficiently addressed by a frequency-domain realization of the turbo-equalizer. We have also noticed that MMSE turbo-equalizers may suffer from performance losses over some severe ISI channels, depending in particular on the bit-labelling strategy considered (for BICM). However, it should be noted that such channels are seldom encountered in many practical transmission systems.

We have finally described the implementation of a low-complexity MMSE turbo-equalizer on a low-cost fixed-point DSP which is typically targeted towards portable terminals such as 2.5G–3G mobile phones. Using only C programming, a promising data rate of 41 Kbits/s has been achieved with 5 iterations. In addition, the fixed-point implementation does not exhibit any performance loss with respect to an ideal unquantized receiver. These results demonstrate the possibility of implementing low-complexity turbo-equalization structures on portable terminals in practice, and also offer an interesting perspective for the realization of an FPGA prototype able to work at significantly higher data rates. The latter may constitute an attractive receiver solution for broadband wireless systems operating in non-line-of-sight environments.

Future directions

A natural direction of future research is the extension of the MMSE IC-LE turbo-equalization scheme to multiple-input multiple-output (MIMO) systems, where the turbo-equalizer could fully exploit the diversity offered by a rich-scattering multipath environment.

We have purposely disregarded the problem of channel estimation in our work. However, this issue should not be overlooked in practice, and it is desirable to examine in particular the robustness of MMSE turbo-equalizers with respect to a mismatched estimation. Another interesting research topic involves investigating the possibility of taking explicitly advantage of *a priori* information in the channel estimation process, just as we do in the derivation of the equalizer. Solutions in this direction

have been reported for example in [133], where the authors propose an improved form of the standard RLS adaptive algorithm suitable for fast-varying fading channels.

We have seen that MMSE turbo-equalizers may exhibit performance losses over severe ISI channels, especially when high-order modulations are used. It is the author's belief that the potential of MMSE equalization in the presence of *a priori* information has not been fully exploited yet, and that better equalizers remain to be found. Another approach also exists, which involves transforming a severe channel into a channel well-suited for MMSE turbo-equalization. In particular, a form of channel-independent precoding was suggested in [37], where frequency-domain interleaving is performed at the transmitted side to appropriately condition the channel. Frequency-domain equalization is then used at the receiving unit. It would be interesting to measure precisely the performance gains (if any) that could be obtained with this approach for frequency-domain MMSE turbo-equalizers operating over severe channels.

A novel turbo-equalizer able to deal with multilevel modulations and long delay spread channels has recently been proposed in [82]. This turbo-equalizer relies on a LISt-Sequential (LISS) SISO equalizer using a modified stack algorithm operating on a tree. It would be interesting to further investigate this solution and examine in particular how it compares with MMSE turbo-equalization schemes, both from a performance and complexity point of view, in the context of broadband wireless transmissions.

Finally, we have seen from the EXIT chart analysis that the parameters of the outer error-correcting code involved in the turbo-equalization scheme have an important impact on the convergence behavior of the overall iterative equalization and decoding process. Given a channel model and a SISO equalization algorithm, it would be desirable to optimize the characteristics of the outer code so as to yield an earlier convergence threshold without affecting the asymptotic performance too much. This issue was briefly examined in [174]. A more general framework for this problem was introduced in [164] where it is shown that code optimization can be interpreted as a curve-fitting procedure on EXIT charts. Several solutions have been suggested in order to construct coding schemes with prescribed convergence properties. Applying this method in a turbo-equalization context constitutes an interesting topic of research when the channel characteristics are known in advance, as happens for example with magnetic recording systems.

Appendix A

Derivations of the equalizers of chapter 4

This appendix groups together the derivations of the MMSE equalizers introduced in chapter 4. We shall on several occasions use the rules of derivation with respect to complex-valued vectors. For convenience, these rules are summarized in table A.1. We recall that $\mathbf{z}^*$ and $\mathbf{z}^H$ denote the conjugate and hermitian transpose of a vector $\mathbf{z}$ respectively.

$f(\mathbf{z})$	$\frac{\partial f(\mathbf{z})}{\partial \mathbf{z}}$	$\frac{\partial f(\mathbf{z})}{\partial \mathbf{z}^*}$
$\mathbf{a}^T \mathbf{z} = \mathbf{z}^T \mathbf{a}$	$\mathbf{a}$	$\mathbf{0}$
$\mathbf{a}^T \mathbf{z}^* = \mathbf{z}^H \mathbf{a}$	$\mathbf{0}$	$\mathbf{c}$
$\mathbf{z}^H \mathbf{z} = \mathbf{z}^T \mathbf{z}^*$	$\mathbf{z}^*$	$\mathbf{z}$
$\mathbf{z}^H \mathbf{M} \mathbf{z} = \mathbf{z}^T \mathbf{M}^T \mathbf{z}^*$	$\mathbf{M}^T \mathbf{z}^*$	$\mathbf{M} \mathbf{z}$

Table A.1: Derivatives of some important functions with respect to a complex-valued vector.

A.1 Derivation of the finite-length time-varying MMSE equalizer

We have to solve the following optimization problem

$$(\mathbf{p}_n, i_n) = \arg \min_{(\mathbf{p}_n, i_n)} E \left(|z_n - x_{n-\Delta}|^2 | \mathcal{L}_{a,[n-\Delta]} \right) \quad (\text{A.1})$$

An important observation is in order here. Since the set of *a priori* LLRs $\mathcal{L}_{a,[n-\Delta]}$ does not include the LLRs relative to the considered data symbol $x_{n-\Delta}$ at time n , we have

$$E(x_{n-\Delta} | \mathcal{L}_{a,[n-\Delta]}) = E(x_{n-\Delta}) = 0 \quad (\text{A.2})$$

$$\text{var}(x_{n-\Delta} | \mathcal{L}_{a,[n-\Delta]}) = \text{var}(x_{n-\Delta}) = \sigma_x^2 \quad (\text{A.3})$$

In order to simplify the notations, we define

$$E_a(x_n) \triangleq E(x_n | \mathcal{L}_{a,[n-\Delta]}) \quad (\text{A.4})$$

for the following calculations in this section.

We begin by developing the expression of the cost function to minimize

$$\varepsilon_n^2 = E_a(|z_n - x_{n-\Delta}|^2) = E_a(|\mathbf{p}_n^T \mathbf{y}_n - i_n - x_{n-\Delta}|^2) \quad (\text{A.5})$$

$$= \mathbf{p}_n^H E_a(\mathbf{y}_n \mathbf{y}_n^H) \mathbf{p}_n - \mathbf{p}_n^T E_a(\mathbf{y}_n (i_n + x_{n-\Delta})^*) - \mathbf{p}_n^H E_a(\mathbf{y}_n^* (i_n + x_{n-\Delta})) + E_a(|i_n + x_{n-\Delta}|^2) \quad (\text{A.6})$$

Let us develop the different terms that appear in the cost function. Using (4.11), we first obtain

$$E_a(\mathbf{y}_n \mathbf{y}_n^H) = E_a([\mathbf{h}_\Delta x_{n-\Delta} + \mathbf{H} \tilde{\mathbf{x}}_n + \mathbf{w}_n][\mathbf{h}_\Delta x_{n-\Delta} + \mathbf{H} \tilde{\mathbf{x}}_n + \mathbf{w}_n]^H) \quad (\text{A.7})$$

$$= \sigma_x^2 \mathbf{h}_\Delta \mathbf{h}_\Delta^H + \mathbf{H} E_a(\tilde{\mathbf{x}}_n \tilde{\mathbf{x}}_n^H) \mathbf{H}^H + \sigma_w^2 \mathbf{I} \quad (\text{A.8})$$

where the second line comes from definition (4.12) and property (A.3), and where we have assumed that the noise samples are uncorrelated and independent of the transmitted data. Regarding the second and third terms, we have using (4.11) and (A.3)

$$E_a(\mathbf{y}_n (i_n^* + x_{n-\Delta}^*)) = E_a([\mathbf{h}_\Delta x_{n-\Delta} + \mathbf{H} \tilde{\mathbf{x}}_n + \mathbf{w}_n](i_n + x_{n-\Delta})^*) \quad (\text{A.9})$$

$$= \sigma_x^2 \mathbf{h}_\Delta + \mathbf{H} E_a(\tilde{\mathbf{x}}_n) i_n^* \quad (\text{A.10})$$

Using finally (A.2) and (A.3) the fourth term simplifies into

$$E_a(|i_n + x_{n-\Delta}|^2) = |i_n|^2 + \sigma_x^2 \quad (\text{A.11})$$

The optimization problem (A.1) is solved using standard minimization techniques. Setting $\partial \varepsilon_n^2 / \partial i_n = 0$, we first obtain

$$\boxed{i_n = \mathbf{p}_n^T \mathbf{H} E_a(\tilde{\mathbf{x}}_n)} \quad (\text{A.12})$$

Now setting $\partial \varepsilon_n^2 / \partial \mathbf{p}_n = 0$ and exploiting the previous result yields

$$[\mathbf{H}(E_a(\tilde{\mathbf{x}}_n \tilde{\mathbf{x}}_n^H) - E_a(\tilde{\mathbf{x}}_n) E_a(\tilde{\mathbf{x}}_n^H)) \mathbf{H}^H + \sigma_x^2 \mathbf{h}_\Delta \mathbf{h}_\Delta^H + \sigma_w^2 \mathbf{I}] \mathbf{p}_n^* = \sigma_x^2 \mathbf{h}_\Delta \quad (\text{A.13})$$

Let us introduce the $J \times J$ covariance matrix $\mathbf{C}_{\tilde{\mathbf{x}}\tilde{\mathbf{x}},n}$ of the vector $\tilde{\mathbf{x}}_n$ at time n conditioned to the LLRs set $\mathcal{L}_{a,[n-\Delta]}$. This matrix admits the following expression

$$\mathbf{C}_{\tilde{\mathbf{x}}\tilde{\mathbf{x}},n} \triangleq \text{Cov}(\tilde{\mathbf{x}}_n \tilde{\mathbf{x}}_n^H | \mathcal{L}_{a,[n-\Delta]}) = E_a(\tilde{\mathbf{x}}_n \tilde{\mathbf{x}}_n^H) - E_a(\tilde{\mathbf{x}}_n) E_a(\tilde{\mathbf{x}}_n^H) \quad (\text{A.14})$$

$$= \text{diag}\{v_n^2, \dots, v_{n-\Delta+1}^2, 0, v_{n-\Delta-1}^2, \dots, v_{n-J+1}^2\} \quad (\text{A.15})$$

where the second line follows from definitions (4.3) and (4.12), and from the assumption that *a priori* LLRs belonging to different data symbols x_k and x_n are independent¹, viz.

$$E(x_n x_k^* | \mathcal{L}_{a,[n-\Delta]}) = \begin{cases} E(x_n | \mathcal{L}_{a,n}) E(x_k^* | \mathcal{L}_{a,k}) = \bar{x}_n \bar{x}_k^* & \text{for } k \neq n \\ E(|x_n|^2 | \mathcal{L}_{a,n}) = v_n^2 + |\bar{x}_n|^2 & \text{for } k = n \end{cases} \quad (\text{A.16})$$

From (A.13) and (A.14), we finally obtain the following expression for the optimum feedforward filter

$$\mathbf{p}_n^* = \sigma_x^2 [\mathbf{H} \mathbf{C}_{\tilde{\mathbf{x}},n} \mathbf{H}^H + \sigma_x^2 \mathbf{h}_\Delta \mathbf{h}_\Delta^H + \sigma_w^2 \mathbf{I}]^{-1} \mathbf{h}_\Delta \quad (\text{A.17})$$

To complete the description of the equalizer, we need to derive the value of the minimum mean-square error at the equalizer output, as well as the expression of the parameters (μ_n, v_n^2) required by the SISO symbol demapping device. We recall that the last two parameters are obtained by considering that the equalized sample z_n at time n may be written as (see section 4.1.2)

$$z_n = \mu_n x_{n-\Delta} + \eta_n \quad (\text{A.18})$$

where η_n is an additive perturbation term encompassing filtered noise and residual ISI, that we suppose uncorrelated with $x_{n-\Delta}$ and Gaussian-distributed with parameters $\mathcal{N}(0, v_n^2)$ at time n . From the discussion in section 4.2, we readily obtain that the time-varying bias factor μ_n is given by

$$\mu_n = \mathbf{h}_\Delta^T \mathbf{p}_n = \mathbf{p}_n^T \mathbf{h}_\Delta \quad (\text{A.19})$$

Finding v_n^2 is a little more involved. First note that using property (A.3) we can write

$$v_n^2 = E_a(|z_n - \mu_n x_{n-\Delta}|^2) = E_a(|z_n|^2) - \sigma_x^2 \mu_n^2 \quad (\text{A.20})$$

For the subsequent calculations, it is convenient to express z_n as follow

$$z_n = \mathbf{p}_n^T (\mathbf{y}_n - \mathbf{H} E_a(\tilde{\mathbf{x}}_n)) \quad (\text{A.21})$$

where we have used expression (A.12) to obtain the previous result. Developing the term $E_a(|z_n|^2)$ then yields

$$E_a(|z_n|^2) = \mathbf{p}_n^T E_a([\mathbf{y}_n - \mathbf{H} E_a(\tilde{\mathbf{x}}_n)][\mathbf{y}_n - \mathbf{H} E_a(\tilde{\mathbf{x}}_n)]^H) \mathbf{p}_n^* \quad (\text{A.22})$$

$$= \mathbf{p}_n^T [\mathbf{H} (E_a(\mathbf{x}_n \mathbf{x}_n^H) - E_a(\tilde{\mathbf{x}}_n) E_a(\tilde{\mathbf{x}}_n^H)) \mathbf{H}^H + \sigma_w^2 \mathbf{I}] \mathbf{p}_n^* \quad (\text{A.23})$$

$$= \mathbf{p}_n^T [\mathbf{H} \mathbf{C}_{\tilde{\mathbf{x}},n} \mathbf{H}^H + \sigma_x^2 \mathbf{h}_\Delta \mathbf{h}_\Delta^H + \sigma_w^2 \mathbf{I}] \mathbf{p}_n^* \quad (\text{A.24})$$

$$= \mathbf{p}_n^T (\sigma_x^2 \mathbf{h}_\Delta) = \sigma_x^2 \mu_n \quad (\text{A.25})$$

¹Such an assumption is justified by the presence of interleavers in a turbo-equalization scheme.

The second line follows from definitions (4.11) and (4.12). The third line follows from (A.14) and the fourth line from (A.17) and (A.19). We obtain the following closed-form result for v_n^2

$$\boxed{v_n^2 = \sigma_x^2 \mu_n (1 - \mu_n)} \quad (\text{A.26})$$

To obtain the minimum MSE ε_n^2 , we finally note that combining (A.18) with (A.5) yields

$$\varepsilon_n^2 = E_a \left(|(1 - \mu_n)x_{n-\Delta} + \eta_n|^2 \right) = \sigma_x^2 |1 - \mu_n|^2 + v_n^2 \quad (\text{A.27})$$

From (A.26), we readily obtain the result

$$\boxed{\varepsilon_n^2 = \sigma_x^2 (1 - \mu_n)} \quad (\text{A.28})$$

This completes the description of the equalizer.

In fact, we can go a little step further and express the optimum feedforward filters coefficient vector $\mathbf{p}_n$ as a function of the covariance matrix $\mathbf{C}_{\mathbf{xx},n}$ of the vector $\mathbf{x}_n$, conditioned to the **full** set of *a priori* LLRs denoted by $\mathcal{L}_a$. Note that this new set includes the LLRs relative to the symbol $x_{n-\Delta}$ at time n . The resulting solution will prove to be useful in identifying the similarities between this equalizer and the alternative solution proposed in section 4.3.

We begin by introducing a new coefficient vector $\tilde{\mathbf{p}}_n$ defined by

$$\boxed{\tilde{\mathbf{p}}_n^* = \sigma_x^2 [\mathbf{H}\mathbf{C}_{\mathbf{xx},n}\mathbf{H}^H + \sigma_w^2 \mathbf{I}]^{-1} \mathbf{h}_\Delta} \quad (\text{A.29})$$

where the covariance matrix $\mathbf{C}_{\mathbf{xx},n}$ is given by

$$\mathbf{C}_{\mathbf{xx},n} \triangleq \text{Cov}(\mathbf{x}_n \mathbf{x}_n^H | \mathcal{L}_a) = \text{diag} \{ v_n^2, \dots, v_{n-J+1}^2 \} \quad (\text{A.30})$$

Following the approach of [175], we shall now show that the coefficients vectors $\mathbf{p}_n$ and $\tilde{\mathbf{p}}_n$ are in fact equivalent up to a time-varying scalar factor λ_n . We first observe that the following relationship holds between the matrices $\mathbf{C}_{\mathbf{xx},n}$ and $\mathbf{C}_{\tilde{\mathbf{x}}\tilde{\mathbf{x}},n}$

$$\mathbf{C}_{\mathbf{xx},n} = \mathbf{C}_{\tilde{\mathbf{x}}\tilde{\mathbf{x}},n} + v_{n-\Delta}^2 \mathbf{e}_\Delta \mathbf{e}_\Delta^H \quad (\text{A.31})$$

where we recall that $\mathbf{e}_\Delta$ is a unit vector having a 1 in position Δ . Inserting this expression into the form of the optimum filter $\mathbf{p}_n$ given by equation (A.17) yields

$$\mathbf{p}_n^* = \sigma_x^2 [\mathbf{H}\mathbf{C}_{\mathbf{xx},n}\mathbf{H}^H + \sigma_w^2 \mathbf{I} + (\sigma_x^2 - v_{n-\Delta}^2) \mathbf{h}_\Delta \mathbf{h}_\Delta^H]^{-1} \mathbf{h}_\Delta \quad (\text{A.32})$$

Invoking Woodward's identity² on the quantity in brackets, using (A.29) and after some algebraic

² $(\mathbf{A} + \mathbf{u}\mathbf{v}^H)^{-1} = \mathbf{A}^{-1} - \frac{\mathbf{A}^{-1}\mathbf{u}\mathbf{v}^H\mathbf{A}^{-1}}{1 + \mathbf{v}^H\mathbf{A}^{-1}\mathbf{u}}.$

manipulations, one finally obtains the desired result, viz.

$$\mathbf{p}_n = \lambda_n \tilde{\mathbf{p}}_n \quad (\text{A.33})$$

with

$$\lambda_n = \frac{\sigma_x^2}{\sigma_x^2 + (\sigma_x^2 - v_{n-\Delta}^2) \tilde{\mu}_n} \quad (\text{A.34})$$

and where we have introduced the scalar quantity $\tilde{\mu}_n$, defined by analogy with expression (A.19) as

$$\tilde{\mu}_n = \tilde{\mathbf{p}}_n^T \mathbf{h}_\Delta \quad (\text{A.35})$$

The other equalizer parameters are left unchanged.

A.2 Derivation of the infinite-length MMSE IC-LE

We have to solve the following optimization problem

$$(P(\omega), Q(\omega)) = \arg \min_{(P(\omega), Q(\omega))} E(|z_n - x_n|^2) \quad (\text{A.36})$$

under the constraint $q_0 = 0$, q_0 being the reference tap of the feedback filter $Q(\omega)$. Using (2.2), the equalized sample z_n at time n is given by

$$z_n = \sum_k p_k y_{n-k} - \sum_k q_k \bar{x}_{n-k} = \sum_k g_k x_{n-k} - \sum_k q_k \bar{x}_{n-k} + \sum_k p_k w_{n-k} \quad (\text{A.37})$$

where $\bar{x}_n$ is the soft estimate of the transmitted symbol x_n , and where we have introduced the transfer function $G(\omega)$ with coefficients $\{g_k\}$ defined by

$$G(\omega) = H(\omega)P(\omega) \quad (\text{A.38})$$

Equivalently, we can write

$$z_n = g_0 x_n + \eta_n \quad (\text{A.39})$$

where η_n denotes the residual interference and filtered noise at the equalizer output. Note that the parameters (μ, v^2) required by the SISO demapper (see section 4.1.2) are related to the quantities g_0 and η_n by $\mu = g_0$ and $v^2 = \text{var}(\eta_n)$. We can then define the signal to interference + noise ratio (SINR) at the equalizer output as follows

$$\text{SINR} = \frac{|\mu|^2 \sigma_x^2}{v^2} \quad (\text{A.40})$$

Following [36] (see also [74]), we propose here to determine the optimum filters $P(\omega)$ and $Q(\omega)$ which maximize the SINR at the equalizer output. In fact, the Max-SINR criterion constitutes a generalization of the standard MMSE criterion, as will be apparent later, and results from the fact that a filtering-based equalizer can be viewed as a device in charge of converting an ISI channel into an equivalent AWGN channel.

We first note that since the quantities x_n , w_n and $\bar{x}_n$ have zero mean (see section 4.3.1 for a proof of the latter claim), we have

$$v^2 = \text{var}(\eta_n) = E(|\eta_n|^2) \quad (\text{A.41})$$

Assuming that $P(\omega)$ is fixed, we search for the optimum form for $Q(\omega)$ which maximizes expression (A.40), or equivalently minimizes v^2 . Developing the expression of v^2 , we obtain

$$v^2 = E \left(\left| \sum_{k \neq 0} g_k x_{n-k} - \sum_{k \neq 0} q_k \bar{x}_{n-k} + \sum_k p_k w_{n-k} \right|^2 \right) \quad (\text{A.42})$$

Defining $\|p\|^2 = \sum_k |p_k|^2$ and assuming that the data x_n as well as their soft estimates $\bar{x}_n$ are independent of the noise samples w_n yields

$$v^2 = E \left(\left| \sum_{k \neq 0} g_k x_{n-k} - \sum_{k \neq 0} q_k \bar{x}_{n-k} \right|^2 \right) + \sigma_w^2 \|p\|^2 \quad (\text{A.43})$$

Recalling that the following property holds between x_n and $\bar{x}_n$ (see section 4.3.1)

$$R_{x\bar{x},k} = E(x_n \bar{x}_{n-k}^*) = E(\bar{x}_n x_{n-k}^*) = R_{\bar{x}x,k} = \sigma_{\bar{x}}^2 \delta_k \quad (\text{A.44})$$

we obtain from (A.43)

$$v^2 = \sigma_x^2 \sum_{k \neq 0} |g_k|^2 + \sigma_{\bar{x}}^2 \sum_{k \neq 0} |q_k|^2 - 2\sigma_{\bar{x}}^2 \text{Re} \left\{ \sum_{k \neq 0} g_k q_k^* \right\} + \sigma_w^2 \|p\|^2 \quad (\text{A.45})$$

The last equation can be conveniently rewritten as follows

$$v^2 = (\sigma_x^2 - \sigma_{\bar{x}}^2) \sum_{k \neq 0} |g_k|^2 + \sigma_{\bar{x}}^2 \sum_{k \neq 0} |g_k - q_k|^2 + \sigma_w^2 \|p\|^2 \quad (\text{A.46})$$

We observe that v^2 is minimum when $\sum_{k \neq 0} |g_k - q_k|^2$ is minimum, yielding the following solution for $Q(\omega)$

$$\boxed{Q(\omega) = G(\omega) - \mu = H(\omega)P(\omega) - \mu} \quad (\text{A.47})$$

Note that the constraint $q_0 = 0$ is respected.

Having found the expression of $Q(\omega)$, we now search for the optimum feedforward filter $P(\omega)$

that maximizes expression (A.40). Substituting (A.47) into (A.46), we obtain

$$E(|\eta_n|^2) = (\sigma_x^2 - \sigma_{\bar{x}}^2)(\|g\|^2 - |\mu|^2) + \sigma_w^2 \|p\|^2 \quad (\text{A.48})$$

where we have defined $\|g\|^2 = \sum_k |g_k|^2$. Consequently, using (A.41) and (A.48) with (A.40) yields the following SINR expression to maximize

$$\text{SINR} = \frac{|\mu|^2 \sigma_x^2}{(\sigma_x^2 - \sigma_{\bar{x}}^2)(\|g\|^2 - |\mu|^2) + \sigma_w^2 \|p\|^2} \quad (\text{A.49})$$

Applying Parseval's theorem, we have

$$\|p\|^2 = \frac{1}{2\pi} \int_{-\pi}^{+\pi} |P(\omega)|^2 d\omega \quad (\text{A.50})$$

$$\|g\|^2 = \frac{1}{2\pi} \int_{-\pi}^{+\pi} |H(\omega)P(\omega)|^2 d\omega \quad (\text{A.51})$$

In addition, it follows from definition (A.38) that

$$\mu = g_0 = \frac{1}{2\pi} \int_{-\pi}^{+\pi} H(\omega)P(\omega) d\omega \quad (\text{A.52})$$

Let us furthermore introduce the following spectral factorization (see chapter 2, theorem 2.1)

$$|F(\omega)|^2 = (\sigma_x^2 - \sigma_{\bar{x}}^2) |H(\omega)|^2 + \sigma_w^2 \quad (\text{A.53})$$

which always exists for $\sigma_w^2 > 0$. Using (A.50)–(A.53) with (A.49), it is straightforward to show that the SINR admits the following equivalent expression

$$\text{SINR} = \frac{\sigma_x^2}{\frac{1}{\tilde{\mu}} + (\sigma_x^2 - \sigma_{\bar{x}}^2)} \quad , \quad \text{with } \tilde{\mu} = \frac{\left| \frac{1}{2\pi} \int_{-\pi}^{+\pi} H(\omega)P(\omega) d\omega \right|^2}{\frac{1}{2\pi} \int_{-\pi}^{+\pi} |F(\omega)P(\omega)|^2 d\omega} \quad (\text{A.54})$$

Hence, maximizing SINR requires maximizing $\tilde{\mu}$. Applying the Cauchy-Schwartz inequality³ to the numerator of $\tilde{\mu}$ yields

$$\left| \frac{1}{2\pi} \int_{-\pi}^{+\pi} H(\omega)P(\omega) d\omega \right|^2 = \left| \frac{1}{2\pi} \int_{-\pi}^{+\pi} \frac{H(\omega)}{F(\omega)} F(\omega)P(\omega) d\omega \right|^2 \quad (\text{A.55})$$

$$\leq \left(\frac{1}{2\pi} \int_{-\pi}^{+\pi} |F(\omega)P(\omega)|^2 d\omega \right) \left(\frac{1}{2\pi} \int_{-\pi}^{+\pi} \left| \frac{H(\omega)}{F(\omega)} \right|^2 d\omega \right) \quad (\text{A.56})$$

It follows that $\tilde{\mu}$ (and hence SINR) is maximum when $P(\omega)F(\omega) = \lambda H^*(\omega)/F^*(\omega)$, where λ is a

³ $|\int AB|^2 \leq (\int |A|^2)(\int |B|^2)$ with equality iff $A = \lambda B^*$.

positive constant to be determined later. This yields the following solution for $P(\omega)$

$$P(\omega) = \lambda \frac{H^*(\omega)^2}{|F(\omega)|} = \lambda \frac{H^*(\omega)}{(\sigma_x^2 - \sigma_{\bar{x}}^2) |H(\omega)|^2 + \sigma_w^2} \quad (\text{A.57})$$

The maximum value of SINR is then given by

$$\text{SINR}_{\max} = \frac{\sigma_x^2}{\frac{1}{\tilde{\mu}} - (\sigma_x^2 - \sigma_{\bar{x}}^2)} \quad , \quad \text{with } \tilde{\mu} = \frac{1}{2\pi} \int_{-\pi}^{+\pi} \frac{|H(\omega)|^2}{(\sigma_x^2 - \sigma_{\bar{x}}^2) |H(\omega)|^2 + \sigma_w^2} d\omega \quad (\text{A.58})$$

Note finally from (A.52), (A.57) and (A.58) that the following relation holds between μ and $\tilde{\mu}$

$$\mu = g_0 = \lambda \tilde{\mu} \quad (\text{A.59})$$

Different equalizers are obtained depending on the value of λ . In particular, setting $\lambda = 1$ yields the unbiased MMSE IC-LE. Here, we are interested in deriving the standard MMSE IC-LE and thus search for the parameter λ that minimizes the mean-square error between z_n and x_n . We first write the mean-square error ε^2 at the equalizer output as follows using (A.39) and (A.59)

$$\varepsilon^2 = E(|z_n - x_n|^2) = E(|(\lambda \tilde{\mu} - 1)x_n + \eta_n|^2) \quad (\text{A.60})$$

$$= \sigma_x^2 |\lambda \tilde{\mu} - 1|^2 + v^2 \quad (\text{A.61})$$

The second line comes from definition $v^2 = E(|\eta_n|^2)$ and from the statistical properties of the soft data estimates $\bar{x}_n$ established in section 4.3.1. From definition (A.40) and result (A.58), we have

$$v^2 = \frac{|\lambda \tilde{\mu}|^2 \sigma_x^2}{\text{SINR}_{\max}} = \lambda^2 \tilde{\mu} [1 - \tilde{\mu}(\sigma_x^2 - \sigma_{\bar{x}}^2)] \quad (\text{A.62})$$

Combining (A.61) and (A.62) yields the following expression of ε^2 as a function of λ

$$\varepsilon^2 = \lambda^2 \tilde{\mu} [1 - \tilde{\mu}(\sigma_x^2 - \sigma_{\bar{x}}^2)] + \sigma_x^2 |\lambda \tilde{\mu} - 1|^2 \quad (\text{A.63})$$

Setting $\partial \varepsilon^2 / \partial \lambda = 0$, we finally obtain the solution

$$\lambda = \frac{\sigma_x^2}{1 + \tilde{\mu} \sigma_{\bar{x}}^2} \quad (\text{A.64})$$

Substituting λ with its value in (A.63) and using (A.59), the minimum mean-square error at the equalizer output is given by

$$\varepsilon^2 = \sigma_x^2 (1 - \mu) \quad (\text{A.65})$$

Combining (A.65) with (A.61) finally yields

$$\boxed{v^2 = \text{var}(\eta_n) = \sigma_x^2 \mu (1 - \mu)} \quad (\text{A.66})$$

This completes the derivation of the infinite-length MMSE IC-LE.

A.3 Derivation of the finite-length MMSE IC-LE

We have to solve the following optimization problem

$$(\mathbf{p}, \mathbf{q}) = \arg \min_{(\mathbf{p}, \mathbf{q})} E(|z_n - x_{n-\Delta}|^2) \quad (\text{A.67})$$

where the equalized datum z_n at the equalizer output at time n is given by

$$z_n = \mathbf{p}^T \mathbf{y}_n - \mathbf{q}^T \tilde{\mathbf{x}}_n, \quad \text{with } \tilde{\mathbf{x}}_n = \bar{\mathbf{x}}_n - \bar{\mathbf{x}}_n \mathbf{e}_\Delta \quad (\text{A.68})$$

We recall that $\mathbf{e}_\Delta$ is the unit vector having a 1 in position Δ . Hence, $\tilde{\mathbf{x}}_n$ is simply the vector $\bar{\mathbf{x}}_n$ of soft estimates at time n but whose component Δ has been set to zero. The optimization problem (A.67) can be solved using the Max-SINR criterion, just as we did for the infinite-length MMSE IC-LE in the previous section. It suffices to replace the spectral factorization (A.53) by a Cholesky factorization (see [107]). For the sake of simplicity however, we rather resort here to standard minimization techniques.

Expression (A.68) can be expressed in more compact form as follows

$$z_n = \mathbf{a}^T \mathbf{b}_n \quad (\text{A.69})$$

where we have introduced the vectors

$$\mathbf{a} = \begin{pmatrix} \mathbf{p} \\ -\mathbf{q} \end{pmatrix} \quad \text{and} \quad \mathbf{b}_n = \begin{pmatrix} \mathbf{y}_n \\ \tilde{\mathbf{x}}_n \end{pmatrix} \quad (\text{A.70})$$

Developing the expression of the mean-square error $\varepsilon^2 = E(|z_n - x_{n-\Delta}|^2)$, we obtain

$$\varepsilon^2 = \sigma_x^2 + \mathbf{a}^H E(\mathbf{b}_n^* \mathbf{b}_n^T) \mathbf{a} - \mathbf{a}^T E(\mathbf{b}_n x_{n-\Delta}^*) - \mathbf{a}^H E(\mathbf{b}_n^* x_{n-\Delta}) \quad (\text{A.71})$$

Setting $\partial \varepsilon^2 / \partial \mathbf{a} = 0$ yields the solution

$$E(\mathbf{b}_n \mathbf{b}_n^H) \mathbf{a}^* = E(\mathbf{b}_n x_{n-\Delta}^*) \quad (\text{A.72})$$

or equivalently the expanded system

$$\begin{pmatrix} E(\mathbf{y}_n \mathbf{y}_n^H) & E(\mathbf{y}_n \tilde{\mathbf{x}}_n^H) \\ E(\tilde{\mathbf{x}}_n \mathbf{y}_n^H) & E(\tilde{\mathbf{x}}_n \tilde{\mathbf{x}}_n^H) \end{pmatrix} \begin{pmatrix} \mathbf{p}^* \\ -\mathbf{q}^* \end{pmatrix} = \begin{pmatrix} E(\mathbf{y}_n x_{n-\Delta}^*) \\ E(\tilde{\mathbf{x}}_n x_{n-\Delta}^*) \end{pmatrix} \quad (\text{A.73})$$

Using expression (2.2), one readily shows that

$$E(\mathbf{y}_n \mathbf{y}_n^H) = \sigma_x^2 \mathbf{H} \mathbf{H}^H + \sigma_w^2 \mathbf{I} \quad (\text{A.74})$$

$$E(\mathbf{y}_n x_{n-\Delta}^*) = \sigma_x^2 \mathbf{h}_\Delta \quad (\text{A.75})$$

where $\mathbf{h}_\Delta$ denotes column Δ of convolution matrix $\mathbf{H}$. In addition, exploiting the correlation properties of the data estimates $\tilde{x}_n$ established in section 4.3.1, we have

$$E(\tilde{\mathbf{x}}_n \tilde{\mathbf{x}}_n^H) = \sigma_x^2 (\mathbf{I} - \mathbf{e}_\Delta \mathbf{e}_\Delta^H) \quad (\text{A.76})$$

$$E(\mathbf{y}_n \tilde{\mathbf{x}}_n^H) = \sigma_x^2 \mathbf{H} (\mathbf{I} - \mathbf{e}_\Delta \mathbf{e}_\Delta^H) \quad (\text{A.77})$$

where the expression $\mathbf{e}_\Delta \mathbf{e}_\Delta^H$ defines a square matrix with a 1 in position (Δ, Δ) and zeros everywhere else. The linear system (A.73) may then be rewritten

$$\begin{pmatrix} \sigma_x^2 \mathbf{H} \mathbf{H}^H + \sigma_w^2 \mathbf{I} & \sigma_x^2 \mathbf{H} (\mathbf{I} - \mathbf{e}_\Delta \mathbf{e}_\Delta^H) \\ \sigma_x^2 (\mathbf{I} - \mathbf{e}_\Delta \mathbf{e}_\Delta^H) \mathbf{H}^H & \sigma_x^2 (\mathbf{I} - \mathbf{e}_\Delta \mathbf{e}_\Delta^H) \end{pmatrix} \begin{pmatrix} \mathbf{p}^* \\ -\mathbf{q}^* \end{pmatrix} = \begin{pmatrix} \sigma_x^2 \mathbf{h}_\Delta \\ \mathbf{0} \end{pmatrix} \quad (\text{A.78})$$

from which we obtain the solutions

$$\mathbf{q} = \mathbf{H}^T \mathbf{p} \quad (\text{A.79})$$

$$\mathbf{p}^* = \sigma_x^2 [(\sigma_x^2 - \sigma_x^2) \mathbf{H} (\mathbf{I} - \mathbf{e}_\Delta \mathbf{e}_\Delta^H) \mathbf{H}^H + \sigma_w^2 \mathbf{I}]^{-1} \mathbf{h}_\Delta \quad (\text{A.80})$$

Let us define the vector $\tilde{\mathbf{p}}$

$$\tilde{\mathbf{p}}^* = \sigma_x^2 [(\sigma_x^2 - \sigma_x^2) \mathbf{H} \mathbf{H}^H + \sigma_w^2 \mathbf{I}]^{-1} \mathbf{h}_\Delta \quad (\text{A.81})$$

Then invoking Woodward's identity on the quantity in brackets in (A.80) and using (A.81) just as we did in section A.1, we find that the optimum feedforward filter $\mathbf{p}$ can be equivalently rewritten as

$$\mathbf{p} = \lambda \tilde{\mathbf{p}} \quad \text{with } \lambda = \frac{\sigma_x^2}{1 + \sigma_x^2 \tilde{\mu}} \quad (\text{A.82})$$

where we have introduced the quantity

$$\tilde{\mu} = \mathbf{h}_\Delta^T \tilde{\mathbf{p}} \quad (\text{A.83})$$

In addition, by defining

$$\mu = \lambda \tilde{\mu} \quad (\text{A.84})$$

we obtain the following expression for the feedback filter $\mathbf{q}$ where the condition $q_\Delta = 0$ has been imposed

$$\mathbf{q} = \mathbf{H}^T \mathbf{p} - \mu \mathbf{e}_\Delta \quad (\text{A.85})$$

The parameters v^2 required by the SISO demapper and the minimum mean-square error ε^2 at the equalizer output are calculated in exactly the same way as in section A.1 and we obtain

$$v^2 = \sigma_x^2 \mu (1 - \mu) \quad (\text{A.86})$$

and

$$\varepsilon^2 = \sigma_x^2 (1 - \mu) \quad (\text{A.87})$$

This completes the derivation of the finite-length MMSE IC-LE.

A.4 Derivation of the frequency-domain MMSE IC-LE

Consider the optimization problem

$$(P_k, Q_k) = \arg \min_{(P_k, Q_k)} E \left(|Z_k - X_k|^2 \right) \quad \text{for all } k = 0, \dots, N-1 \quad (\text{A.88})$$

subject to the constraint

$$q_0 = \frac{1}{N} \sum_{k=0}^{N-1} Q_k = 0 \quad (\text{A.89})$$

where q_0 is the reference tap of the feedback filter with Discrete Fourier Transform (DFT) coefficients $\{Q_k\}$. We recall that the k th DFT coefficient Z_k of the equalized sequence $\{z_n\}$ is given by

$$Z_k = P_k Y_k - Q_k \bar{X}_k, \quad \text{with } Y_k = H_k X_k + W_k \quad (\text{A.90})$$

$$= H_k P_k X_k - Q_k \bar{X}_k + P_k W_k \quad (\text{A.91})$$

The total mean-square error ε^2 after returning in the time-domain can be expressed as

$$\varepsilon^2 = \frac{1}{N} \sum_{k=0}^{N-1} \varepsilon_k^2, \quad \text{where we have defined } \varepsilon_k^2 = E \left(|Z_k - X_k|^2 \right) \quad (\text{A.92})$$

To minimize ε^2 , we simply need to minimize ε_k^2 for each component k separately.

We first write the cost function to minimize as follows

$$\varepsilon_k^2 = E \left(|(H_k P_k - 1) X_k - Q_k \bar{X}_k + P_k W_k|^2 \right) + \beta \left(\frac{1}{N} \sum_{n=0}^{N-1} Q_n \right) \quad (\text{A.93})$$

where we have used (A.91) and introduced the Lagrange multiplier β to account for constraint (A.89). Assuming that the transmitted data symbols as well as their soft estimates are independent from the noise samples, we obtain

$$\varepsilon_k^2 = E \left(|(H_k P_k - 1)X_k - Q_k \bar{X}_k|^2 \right) + \sigma_w^2 |P_k|^2 + \beta \left(\frac{1}{N} \sum_{n=0}^{N-1} Q_n \right) \quad (\text{A.94})$$

Now taking into account the statistical properties of the soft estimates (see section 4.3.1), we have

$$\varepsilon_k^2 = \sigma_x^2 |H_k P_k - 1|^2 + \sigma_{\bar{x}}^2 \left[|Q_k|^2 - H_k Q_k^* - H_k^* Q_k + Q_k^* + Q_k \right] + \sigma_w^2 |P_k|^2 + \beta \left(\frac{1}{N} \sum_{n=0}^{N-1} Q_n \right) \quad (\text{A.95})$$

Setting $\partial \varepsilon_k^2 / \partial Q_k = 0$ in the previous expression yields

$$Q_k = H_k P_k - 1 - \frac{\beta}{\sigma_{\bar{x}}^2 N} \quad (\text{A.96})$$

Applying constraint (A.89) we obtain the following value for β

$$\beta = \sigma_{\bar{x}}^2 \left[\sum_{n=0}^{N-1} H_n P_n - N \right] \quad (\text{A.97})$$

It follows that the optimum DFT coefficients of the feedback filter are given by

$$\boxed{Q_k = H_k P_k - \mu \quad \text{with} \quad \mu = \frac{1}{N} \sum_{n=0}^{N-1} H_n P_n} \quad (\text{A.98})$$

Setting now $\partial \varepsilon_k^2 / \partial P_k = 0$ in expression (A.95) yields the following relation, after simple algebra,

$$P_k^* \left[(\sigma_x^2 - \sigma_{\bar{x}}^2) |H_k|^2 + \sigma_w^2 \right] = H_k \left[\sigma_x^2 - \sigma_{\bar{x}}^2 \mu \right] \quad (\text{A.99})$$

where we have used definition (A.98) of μ . Since μ depends on P_k , solving the previous equation for P_k requires introducing the quantities $\tilde{P}_k$ and $\tilde{\mu}$ defined respectively by

$$\boxed{\tilde{P}_k = \frac{H_k^*}{(\sigma_x^2 - \sigma_{\bar{x}}^2) |H_k|^2 + \sigma_w^2}} \quad (\text{A.100})$$

and

$$\boxed{\tilde{\mu} = \frac{1}{N} \sum_{n=0}^{N-1} H_n \tilde{P}_n} \quad (\text{A.101})$$

Then equation (A.99) can be rewritten as follows

$$\boxed{P_k = \lambda \tilde{P}_k} \quad (\text{A.102})$$

where λ is a scalar value given by

$$\lambda = \sigma_x^2 - \sigma_x^2 \mu \quad (\text{A.103})$$

By noting from (A.101) and (A.102) that the following relation holds between μ and $\tilde{\mu}$

$$\boxed{\mu = \lambda \tilde{\mu}} \quad (\text{A.104})$$

we can express λ as a sole function of $\tilde{\mu}$ (and hence $\tilde{P}_k$) using (A.103), yielding

$$\boxed{\lambda = \frac{\sigma_x^2}{1 + \sigma_x^2 \tilde{\mu}}} \quad (\text{A.105})$$

To obtain the minimum mean-square error value ε^2 at the frequency-domain equalizer output, we first substitute P_k and Q_k with expressions (A.102) and (A.98) in equation (A.93). This yields after simplifications

$$\varepsilon_k^2 = \lambda \frac{|H_k|^2}{(\sigma_x^2 - \sigma_x^2 \mu) |H_k|^2 + \sigma_w^2} [\lambda + 2\sigma_x^2 - 2\sigma_x^2 \mu] + \mu^2 \sigma_x^2 - 2\mu \sigma_x^2 + \sigma_x^2 \quad (\text{A.106})$$

Inserting the previous expression into (A.92) and using (A.98) and (A.104), it follows that

$$\varepsilon^2 = \frac{1}{N} \sum_{k=0}^{N-1} \varepsilon_k^2 = \sigma_x^2 - 2\sigma_x^2 \mu + \mu(\lambda + \mu \sigma_x^2) \quad (\text{A.107})$$

Then using (A.105), we finally come up with the desired result, viz.

$$\boxed{\varepsilon^2 = \sigma_x^2 - \sigma_x^2 \mu = \sigma_x^2 (1 - \mu)} \quad (\text{A.108})$$

We also have to determine the parameter v^2 required by the SISO demapper. We recall that the equalized sample z_n at time n in the time-domain is given by

$$z_n = \mu x_n + \eta_n \quad (\text{A.109})$$

with $\text{var}(\eta_n) = E(|\eta_n|^2) = v^2$. Hence, the minimum mean-square error ε^2 at the equalizer output admits the following expression

$$\varepsilon^2 = E(|z_n - x_n|^2) = E(|(\mu - 1)x_n + \eta_n|^2) = \sigma_x^2 (1 - \mu)^2 + v^2 \quad (\text{A.110})$$

Combining (A.108) with the previous expression, we readily obtain the result

$$\boxed{v^2 = \varepsilon^2 - \sigma_x^2(1 - \mu)^2 = \sigma_x^2\mu(1 - \mu)} \quad (\text{A.111})$$

This completes the derivation of the frequency-domain MMSE IC-LE.

Appendix B

On the asymptotic efficiency of algorithms

On several occasions in this dissertation we use the “big oh” O -notation to describe the asymptotic complexity of an algorithm. The precise meaning of this notation deserves some clarification. This discussion follows the exposition given by Cormen *et al.* in [44, chap. 3].

Let $T(n)$ be some measure of the complexity (typically an estimate of the running time) of an algorithm with respect to the size n of some input parameter. We are concerned with how the complexity of the algorithm increases as the size n of the input increases without bounds. Usually, an algorithm that is asymptotically more efficient is expected to be the best choice for all but small input sizes.

The O -notation gives an asymptotic upper-bound on the complexity of an algorithm, to within a constant factor. For a given function $g(n)$, we define by $O(g(n))$ the set of cost functions $T(n)$ for which there exist positive constants c and n_0 such that

$$0 \leq T(n) \leq cg(n) \quad \text{for all } n \geq n_0$$

The upper-bound provided by the O -notation may or may not be asymptotically tight, and describes the worst-case complexity of an algorithm.

Appendix C

The Forward-Backward algorithm

The so-called “BCJR-MAP” algorithm is an optimum symbol-by-symbol MAP decoding procedure, originally devised by Balh *et al.* in 1974 [19]. It has recently evolved into a basic tool of central importance for the design of efficient *soft-input soft-output* (SISO) decoders to be used in iterative “Turbo” receivers [24]. Thorough expositions of this algorithm and its common variants are now available in many textbooks (see e.g. [23, 86, 188]). Hence, this appendix does not cover the BCJR-MAP in its classical form, but rather attempts to provide an original exposition of the Forward-Backward computation procedure, which lies at the core of the BCJR-MAP. The Forward-Backward algorithm (FBA) is presented as a generic and efficient method for computing *flows* in a special kind of directed graph called a *weighted trellis*. This particular framework avoids the need to manipulate probabilities, and leads instead to a simple, graphical derivation of the algorithm, which builds upon the standard terminology for directed graphs (see e.g. [44, app. B]). Our exposition closely follows some thought-provoking lecture notes by McEliece [125]. For the interested reader, we mention that other insightful presentations and generalizations of this algorithm can be found in [3, 98, 126].

C.1 Preliminary definitions

The following definitions build upon the tutorial expositions on the trellis structure of codes provided by McEliece [124] and Vardy [182].

A trellis $T(V, E)$ of *rank* n is a directed graph consisting of a set V of *vertices*, and a set E of couples (v, v') with $v, v' \in V$ called *edges*. Every vertex is assigned a “depth”¹ in the range $\{0, 1, \dots, n\}$.

¹ One may think of the depth as a discrete time index in the context of decoding or sequence detection.

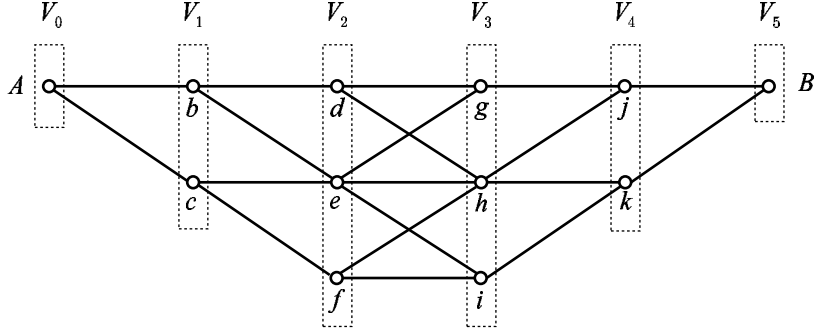


Figure C.1: An example of trellis with rank 5. Here, $V_0 = \{A\}$, $V_1 = \{b, c\}$, $E_{0,1} = \{(A, b), (A, c)\}$, etc.

The set of vertices at depth i is denoted by V_i , and the whole vertex set V can be partitioned as

$$V = V_0 \cup V_1 \cup \dots \cup V_n$$

Each edge $(v, v') \in E$ connects a vertex v at depth $i-1$ to a vertex v' at depth i for some $i = 1, 2, \dots, n$. Multiple edges between pairs of vertices are allowed. The set of edges connecting vertices at depth $i-1$ to those at depth i is denoted by $E_{i-1,i}$, so that E admits the partition

$$E = E_{0,1} \cup E_{1,2} \cup \dots \cup E_{n-1,n}$$

If $e \in E$ is an edge connecting the vertices v and v' , we call v the *initial* vertex and v' the *final* vertex of e , and write $\text{init}(e) = v$ and $\text{fin}(e) = v'$. For the ease of exposition, we shall assume that there is single vertex of depth 0 called the *source*, denoted by A , as well as a single vertex of depth n called the *sink*, denoted by B . These assumptions however may be safely relaxed to encompass more general trellis structures. An trellis example is shown in figure C.1.

A *path* P of *length* k in the trellis from a vertex $v_0 \in V$ to a vertex $v_k \in V$ is an ordered set of connected edges $P = \{e_1, e_2, \dots, e_k\}$, such that e_1 begins at v_0 , e_k ends at v_k , and each consecutive pair of edges (e_i, e_{i+1}) shares a common vertex v_i at which e_i ends and e_{i+1} begins. If P is such a path, we write for short

$$P : v_0 \mapsto v_k$$

If in addition the path P passes through the intermediate vertex v_i at some depth i , we write

$$P : v_0 \xrightarrow{v_i} v_k$$

Finally, if the path P contains the edge $e_i \in E_{i-1,i}$ for some depth i , we write

$$P : v_0 \xrightarrow{e_i} v_k$$

Let us assume now that each edge e in the trellis is assigned a *weight* $w(e)$, that we suppose to be a real number. If $P = \{e_1, e_2, \dots, e_k\}$ is a path of length k in T , its weight $w(P)$ is defined as the product of the individual weights of the component edges

$$w(P) = w(e_1)w(e_2) \cdots w(e_k)$$

The fundamental quantities computed by the Forward-Backward algorithm are the *flows*. If v_0 and v_k are vertices in a trellis, the *flow from* v_0 *to* v_k , denoted $\mu(v_0, v_k)$, is defined as the sum of the weights of all paths connecting v_0 to v_k

$$\mu(v_0, v_k) = \sum_{P: v_0 \mapsto v_k} w(P)$$

If there are no such paths, $\mu(v_0, v_k)$ is defined to be zero. Similarly, if v_0 , v_i and v_k are vertices in T , the *constrained flow from* v_0 *to* v_k *through* v_i is denoted $\mu_{v_i}(v_0, v_k)$ and defined as

$$\mu_{v_i}(v_0, v_k) = \sum_{P: v_0 \mapsto^{v_i} v_k} w(P)$$

Finally, if v_0 and v_k are vertices and e_i is an edge in T , the *constrained flow from* v_0 *to* v_k *through* e_i is denoted $\mu_{e_i}(v_0, v_k)$ and defined as

$$\mu_{e_i}(v_0, v_k) = \sum_{P: v_0 \mapsto^{e_i} v_k} w(P)$$

C.2 Exposition of the algorithm

The Forward-Backward algorithm offers a computational efficient solution to the following three problems

1. Compute the flow from A to B , i.e.

$$\mu(A, B) = \sum_{P: A \mapsto B} w(P)$$

2. For a given vertex $v_i \in V_i$, compute the flow from A to B through v_i .

$$\mu_{v_i}(A, B) = \sum_{P: A \mapsto^{v_i} B} w(P)$$

3. For a given edge $e_i \in E_{i-1,i}$, compute the flow from A to B through e_i

$$\mu_{e_i}(A, B) = \sum_{P: A \mapsto^{e_i} B} w(P)$$

It is instructive to note that

$$\sum_{e_i \in E_{i-1,i}} \mu_{e_i}(A, B) = \sum_{v_i \in V_i} \mu_{v_i}(A, B) = \mu(A, B) \quad (\text{C.1})$$

since each path from A to B must pass through exactly one of the vertices in V_i , or equivalently traverse exactly one of the edges in $E_{i-1,i}$. Consequently, the solution to problem 1 is immediately obtained as a by-product of the solutions to problems 2 and 3, which essentially involve in turn devising an efficient method to compute the constrained flows $\mu_{v_i}(A, B)$ and $\mu_{e_i}(A, B)$.

Given some intermediate vertex $v_i \in V$, it will prove to be mathematically convenient to define the *partial flows*

$$\alpha_i(v_i) = \mu(A, v_i) \quad , \quad \beta_i(v_i) = \mu(v_i, B)$$

By convention, we define $\alpha_0(A) = \beta_n(B) = 1$. Then, the following two theorems completely describe the Forward-Backward algorithm.

Theorem C.1 (Factorization of constrained flows). *Given some vertex $v_i \in V$, we have*

$$\mu_{v_i}(A, B) = \mu(A, v_i) \cdot \mu(v_i, B) = \alpha_i(v_i) \cdot \beta_i(v_i) \quad (\text{C.2})$$

Similarly, given some edge $e_i \in E_{i-1,i}$ with $\text{init}(e) = v_{i-1}$ and $\text{fin}(e) = v_i$, we have

$$\mu_{e_i}(A, B) = \mu(A, v_{i-1}) \cdot w(e_i) \cdot \mu(v_i, B) = \alpha_{i-1}(v_{i-1}) \cdot w(e_i) \cdot \beta_i(v_i) \quad (\text{C.3})$$

Proof. Let us prove (C.2). Suppose that there are m paths, say $F_1, F_2, \dots, F_m$, from A to v_i , and n paths, say $G_1, G_2, \dots, G_n$, from v_i to B . Then we count exactly mn paths from A to B , of the form $F_i \bowtie G_j$ where the operator $\bowtie$ denotes path concatenation, and we have

$$\begin{aligned} \mu_{v_i}(A, B) &= \sum_{P: A \rightarrow^{v_i} B} w(P) \\ &= \sum_{i=1}^m \sum_{j=1}^n w(F_i \bowtie G_j) \\ &= \sum_{i=1}^m \sum_{j=1}^n w(F_i) \cdot w(G_j) \\ &= \left(\sum_{i=1}^m w(F_i) \right) \left(\sum_{j=1}^n w(G_j) \right) \\ &= \mu(A, v_i) \cdot \mu(v_i, B) \end{aligned}$$

The proof for (C.3) is similar. □

Theorem C.2 (Backward and forward recursions). *The partial flow $\alpha_i(v_i)$ can be recursively com-*

puted using the forward recursion

$$\alpha_i(v_i) = \sum_{\substack{e \in E_{i-1,i} \\ \text{fin}(e)=v_i}} \alpha_{i-1}(\text{init}(e)) \cdot w(e) \quad (\text{C.4})$$

Similarly, the partial flow $\beta_i(v_i)$ can be obtained recursively using the backward recursion

$$\beta_i(v_i) = \sum_{\substack{e \in E_{i,i+1} \\ \text{init}(e)=v_i}} w(e) \cdot \beta_{i+1}(\text{fin}(e)) \quad (\text{C.5})$$

Proof. We shall only prove (C.4) since a similar derivation holds for (C.5). Each path from A to v_i in the trellis must traverse exactly one edge $e \in E_{i-1,i}$, connecting some vertex v_{i-1} at depth $i-1$ to the target vertex v_i at depth i . Consequently, the partial flow $\alpha_i(v_i) = \mu(A, v_i)$ may be obtained as

$$\alpha_i(v_i) = \sum_{\substack{e \in E_{i-1,i} \\ \text{fin}(e)=v_i}} \mu_e(A, v_i)$$

Let P be any path from A to the initial vertex $v_{i-1} = \text{init}(e)$ of one of the edges considered above. By definition of the constrained flow $\mu_e(A, v_i)$, we have

$$\begin{aligned} \mu_e(A, v_i) &= \sum_{P: A \mapsto \text{init}(e)} w(P) \cdot w(e) \\ &= \left(\sum_{P: A \mapsto \text{init}(e)} w(P) \right) w(e) \\ &= \mu(A, \text{init}(e)) \cdot w(e) \\ &= \alpha_{i-1}(\text{init}(e)) \cdot w(e) \end{aligned}$$

which completes the proof. □

Depending on the meaning assigned to the weights and flows in the trellis, different applications of the Forward-Backward algorithm may arise. In particular, the original paper by Bahl *et al* [19] presents the application of the FBA to the problem of estimating the *a posteriori* probabilities (APP) of the states and transitions of a Markov source observed through a discrete memoryless channel.

Appendix D

On fixed-point arithmetic

In digital hardware, real numbers are stored in binary words which are finite-length sequences of binary digits (1's and 0's). Interpretation of the binary word is described by the data type. Binary words are commonly represented as either fixed-point or floating-point data types. Floating-point data types are characterized by a sign bit, a fraction (or mantissa) field, and an exponent field. The length of these fields is well standardized¹ so that a number can be identified without ambiguity from its binary word representation. Fixed-point data types are characterized by the word size in bits, the location of the radix (binary) point, and whether the numbers are signed or unsigned. In contrast with floating-point numbers, they do not explicitly include any scaling information (exponent), so that the programmer is left with the interpretation of the number. A fixed-point data type is capable of finer resolution than floating-point for the same binary word length, because of the extra bits available for the mantissa. On the other hand, fixed-point numbers are limited in that they cannot simultaneously represent numbers with very large or very small magnitude using a reasonable word length, so that computations may fail to maintain sufficient accuracy. This appendix introduces some handy notations to characterize fixed-point numbers, and exposes some fundamental rules of fixed-point arithmetic.

D.1 Unsigned fixed-point representation

We define the notation $U(a, b)$ to indicate a fixed-point unsigned number with a bits of dynamic (integer part) and b bits of precision (fractional part). When $b = 0$, we have an integer number. When $a = 0$, the number is purely fractional. For $a \neq 0$ and $b \neq 0$, we have a *mixed-mode* fixed-point number. Representing a $U(a, b)$ fixed-point number requires $N = a + b$ bits. The value of a particular N -bit

¹A famous example is the IEEE Standard 754-1985 for binary floating-point arithmetic.

number $x = (x_0, \dots, x_{N-1})$ in $U(a, b)$ representation is given by the expression

$$x = \frac{1}{2^b} \left[\sum_{n=0}^{N-1} x_n 2^n \right]$$

where x_n represents the value of bit n of x . We use the convention that x_0 and x_{N-1} denote the least significant bit (LSB) and most significant bit (MSB) of the binary word respectively. The range of a $U(a, b)$ representation is therefore

$$0 \leq x \leq 2^a - 2^{-b}$$

The resolution of the data type (smallest strictly positive representable number) is $R(x) = 2^{-b}$.

D.2 Signed fixed-point representation

We distinguish between 3 possible binary encodings of signed numbers, namely: sign/magnitude, one's complement and two's complement. We tacitly assume hereafter a two's complement encoding. We recall that two's complement results from a bit inversion (one's complement) followed the addition of a 1.

The notation $S(a, b)$ refers to a fixed-point signed number with a bits of dynamic and b bits of precision. Representing an $S(a, b)$ fixed-point number requires $N = a + b + 1$ bits since one additional bit is necessary to encode the sign in two's complement format. The value of a particular N -bit number $x = (x_0, \dots, x_{N-1})$ in $S(a, b)$ representation is given by

$$x = \frac{1}{2^b} \left[-2^{N-1} x_{N-1} + \sum_{n=0}^{N-2} x_n 2^n \right]$$

The corresponding range is then

$$-2^a \leq x \leq +2^a - 2^{-b}$$

with a resolution of $R(x) = 2^{-b}$. Note that with two's complement representation, one cannot discern whether numbers are signed or unsigned by simple inspection since this information is not explicitly encoded within the binary word.

The particular purely fractional number formats $S(0, 15)$ and $S(0, 31)$ are frequently encountered when dealing with fixed-point DSP. They are called Q15 and Q31 representation, and have the respective ranges

$$-1.0 \leq x_{Q15} \leq +1.0 - 2^{-15} \quad , \quad -1.0 \leq x_{Q31} \leq +1.0 - 2^{-31}$$

D.3 Arithmetic operations

Table D.1 lists some fundamental rules of fixed-point arithmetic. When a rule applies both to unsigned and signed representation, we use the notation $X(a,b)$ where X stands either for U or S . The number of bits required to represent the final result of an operation is specified, assuming that $X(a,b)$ and $X(c,d)$ are M - and N -bit numbers respectively.

Description	Operation	Resulting format	Resulting wordlength
Addition, subtraction	$X(a,b) + X(a,b)$	$X(a+1,b)$	$M+1$ bits
	$X(a,b) - X(a,b)$	$X(a+1,b)$	$M+1$ bits
Multiplication	$U(a,b) \times U(c,d)$	$U(a+c,b+d)$	$M+N$ bits
	$S(a,b) \times S(c,d)$	$S(a+c+1,b+d)$	$M+N$ bits
Division	$U(a,b) / U(c,d)$	$U(a+d,b+c)$	$M+N$ bits
	$S(a,b) / S(c,d)$	$S(a+d+1,b+c)$	$M+N$ bits
Shifting by m positions (to reinterpret scaling)	$X(a,b) \gg m$	$X(a+m,b-m)$	M bits
	$X(a,b) \ll m$	$X(a-m,b+m)$	M bits
Shifting by m positions (to modify scaling)	$X(a,b) \gg m$	$X(a,b-m)$	$M-m$ bits
	$X(a,b) \ll m$	$X(a,b+m)$	$M+m$ bits

Table D.1: Fixed-point arithmetic rules

This table deserves some remarks. We first note that addition and subtraction only make sense for numbers with compatible fixed-point representation. Moreover, shifting operations may take one of two possible interpretations depending on the context. Right/left shifting amounts to a “power-of-2” multiplication/division respectively. The shifting operation can be implicit: the binary word is left unchanged but the radix point location is tacitly moved to the left or to the right, so that the interpretation of the number is modified. We call this *shifting to reinterpret scaling*. When the shifting is explicit, the binary word is shifted to the right or to the left, some bits are discarded, and appropriate bits (set to 0 or resulting from sign bit extension) are inserted to fill in the gap. The fixed-point representation is left unchanged by the shift. We refer to this second operation as *shifting to modify scaling*. Explicit right shifting is commonly used to increase the precision of a fixed-point number just before performing a computation.

D.4 Additional definitions

We finally introduce a few notions that frequently arise when discussing fixed-point arithmetic issues.

First, fixed-point numbers approximate arbitrarily precise real-world values with a fixed number of bits. Consequently, fixed-point representation introduces quantization noise in the computations.

Moreover, the result of most operations on a fixed-point number typically requires a longer word length than the number's original format. When the result has to be put back into the original representation, one must resort to rounding techniques which increase in turn the computational noise. We usually identify 4 distinct rounding strategies, namely *round towards zero*, *round towards nearest*, *ceil rounding* and *floor rounding*. Each one has its own benefits and disadvantages, depending on the implementation's context.

An *underflow* (respectively *overflow*) occurs when the result of a computation is too small (resp. too large) so that it does not fit into the precision (resp. dynamic range) of the number representation. While underflows typically produce numbers equal to zero, exceeding the number representation range can lead to undesirable results. Fixed-point computation units usually treat this problem by operating either in number *saturation mode* (the result is clipped to the maximum representable value), or number *wrapping mode* (a positive number exceeding the allowed dynamic range becomes negative, and vice-versa).

Bibliography

- [1] M. Aaron and D. W. Tufts, "Intersymbol interference and error probability," *IEEE Trans. Inform. Theory*, vol. IT-12, no. 1, pp. 26–34, Jan. 1966.
- [2] O. E. Agazzi and N. Seshadri, "On the use of tentative decisions to cancel intersymbol interference and nonlinear distortion (with application to magnetic recording channels)," *IEEE Trans. Inform. Theory*, vol. 43, no. 2, pp. 394–408, Mar. 1997.
- [3] S. M. Aji and R. J. McEliece, "The generalized distributive law," *IEEE Trans. Inform. Theory*, vol. 46, no. 2, pp. 325–343, Mar. 2000.
- [4] N. Al-Dhahir and J. M. Cioffi, "Fast computation of channel-estimate based equalizers in packet data transmission," *IEEE Trans. Signal Processing*, vol. 43, no. 11, pp. 2462–2473, Nov. 1995.
- [5] —, "MMSE decision-feedback equalizers: Finite-length results," *IEEE Trans. Inform. Theory*, vol. 41, no. 4, pp. 961–975, July 1995.
- [6] —, "Efficient computation of the delay-optimized finite length MMSE-DFE," *IEEE Trans. Signal Processing*, vol. 44, no. 5, pp. 1288–1292, May 1996.
- [7] —, "Efficiently computed reduced-parameter input-aided MMSE equalizers for ML detection: A unified approach," *IEEE Trans. Inform. Theory*, vol. 42, no. 3, pp. 903–915, May 1996.
- [8] N. Al-Dhahir and A. H. Sayed, "CORDIC-based MMSE-DFE coefficient computation," *Digital Signal Processing*, vol. 9, no. 3, pp. 178–194, July 1999.
- [9] —, "The finite-length multi-input multi-output MMSE-DFE," *IEEE Trans. Signal Processing*, vol. 48, no. 10, pp. 2921–2936, Oct. 2000.
- [10] S. A. Al-Semari and T. E. Fuja, "I-Q TCM: Reliable communication over the Rayleigh fading channel close to the cutoff rate," *IEEE Trans. Inform. Theory*, vol. 43, no. 1, pp. 250–262, Jan. 1997.
- [11] S. A. Altekari and N. C. Beaulieu, "Upper bounds to the error probability of decision feedback equalization," *IEEE Trans. Inform. Theory*, vol. 39, no. 1, pp. 145–156, Jan. 1993.

- [12] I. N. Andersen, "Sample-whitened matched filters," *IEEE Trans. Inform. Theory*, vol. IT-19, no. 5, pp. 653–650, Sept. 1973.
- [13] J. B. Anderson and S. Mohan, "Sequential coding algorithm: A survey and cost analysis," *IEEE Trans. Commun.*, vol. COM-32, no. 2, pp. 169–176, Feb. 1984.
- [14] ———, *Source and Channel Coding – An Algorithmic Approach*. Norwell, MA: Kluwer Academic Publishers, 1991.
- [15] S. L. Ariyavistakul and L. J. Greenstein, "Reduced-complexity equalization techniques for broadband wireless channels," *IEEE J. Select. Areas Commun.*, vol. 15, no. 1, pp. 5–15, Jan. 1997.
- [16] A. Ashikhmin, G. Kramer, and S. ten Brink, "Extrinsic information transfer functions: Model and erasure channel properties," *Submitted to IEEE Trans. Inform. Theory*, Mar. 2003.
- [17] T. M. Aulin, "Breadth-first maximum likelihood sequence detection: Basics," *IEEE Trans. Commun.*, vol. 47, no. 2, pp. 208–216, Feb. 1999.
- [18] A. R. S. Bahai and B. R. Saltzberg, *Multi-Carrier Digital Communications – Theory and Applications of OFDM*. Norwell, MA: Kluwer Academic Publishers, 1999.
- [19] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate (Corresp.)," *IEEE Trans. Inform. Theory*, vol. IT-20, no. 2, pp. 284–287, Mar. 1974.
- [20] G. Bauch and V. Franz, "Iterative equalization and decoding for the GSM system," in *Proc. IEEE Veh. Technol. Conf. VTC'98*, Ottawa, Ontario, Canada, May 1998, pp. 2262–2266.
- [21] G. Bauch, H. Khorram, and J. Hagenauer, "Iterative equalization and decoding in mobile communications systems," in *Proc. 2nd European Personal Mobile Commun. Conf. EPMCC'97*, Bonn, Germany, Sept.-Oct. 1997, pp. 307–312.
- [22] C. A. Belfiore and J. H. Park, Jr, "Decision feedback equalization," *Proc. IEEE*, vol. 67, no. 8, pp. 1143–1156, Aug. 1979.
- [23] S. Benedetto and E. Biglieri, *Principles of Digital Transmission – With Wireless Applications*. New-York, NY: Kluwer Academic / Plenum Publishers, 1999.
- [24] S. Benedetto, G. Montorsi, D. Divsalar, and F. Pollara, "A soft-input soft-output maximum a posteriori (MAP) module to decode parallel and serial concatenated codes," Jet Propulsion Laboratory, Pasadena, CA, JPL TDA Progress Report vol. 42-127, Nov. 1996.
- [25] C. Berrou, "The ten-year-old turbo codes are entering into service," *IEEE Commun. Mag.*, vol. 41, no. 8, pp. 110–116, Aug. 2003.

- [26] C. Berrou and A. Glavieux, "Reflections on the prize paper: "Near optimum error-correcting coding and decoding: Turbo-codes"," *IEEE Inform. Theory Soc. Newsletter*, vol. 48, no. 2, p. 1, June 1998.
- [27] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near shannon limit error-correcting coding and decoding: Turbo codes," in *Proc. IEEE Int. Conf. Commun. ICC'93*, vol. 2, Geneva, Switzerland, 23-26 May 1993, pp. 1064–1070.
- [28] A. O. Berthet, "Méthodes itératives appliquées au décodage efficace de combinaisons de codes en treillis," Ph.D. dissertation, ENST, Paris, France, Dec. 2001.
- [29] A. O. Berthet, B. S. Ünal, and R. Visoz, "Iterative decoding of convolutionally encoded signals over multipath rayleigh fading channels," *IEEE J. Select. Areas Commun.*, vol. 19, no. 9, pp. 1729–1743, Sept. 2001.
- [30] E. Biglieri, "Ungerboeck codes do not shape the signal power spectrum (Corresp.)," *IEEE Trans. Inform. Theory*, vol. IT-32, no. 4, pp. 595–596, July 1986.
- [31] E. Biglieri, D. Divsalar, P. J. McLane, and M. K. Simon, *Introduction to Trellis-Coded Modulation with Applications*. New-York, NY: McMillan Publishing Company, 1991.
- [32] J. A. C. Bingham, "Multicarrier modulation: An idea whose time has come," *IEEE Commun. Mag.*, vol. 28, no. 5, pp. 5–14, May 1990.
- [33] E. Boutillon, W. J. Gross, and P. G. Gulak, "VLSI architectures for the MAP algorithm," *IEEE Trans. Commun.*, vol. 51, no. 2, pp. 175–185, Feb. 2003.
- [34] G. Caire, G. Taricco, and E. Biglieri, "Bit-interleaved coded modulation," *IEEE Trans. Inform. Theory*, vol. 44, no. 3, pp. 927–946, May 1998.
- [35] A. M. Chan, "A class of batch-iterative methods for equalization of intersymbol interference channels," Master's thesis, Dept. EECS, Massachusetts Institute of Technology, Boston, MA, Sept. 1999.
- [36] A. M. Chan and G. W. Wornell, "A class of block-iterative equalizers for intersymbol interference channels: Fixed channel results," *IEEE Trans. Commun.*, vol. 49, no. 11, pp. 1966–1976, Nov. 2001.
- [37] —, "Approaching the matched-filter bound using iterated-decision equalization with frequency-interleaved encoding," in *Proc. IEEE Global Telecommun. Conf. GLOBECOM'02*, vol. 1, Taipei, Taiwan, 17-21 Nov. 2002, pp. 297–301.
- [38] P. R. Chevillat and E. Eleftheriou, "Decoding of trellis-encoded signals in the presence of intersymbol interference and noise," *IEEE Trans. Commun.*, vol. 37, no. 7, pp. 669–676, July 1989.

- [39] A. Chindapol and J. A. Ritcey, "Design, analysis, and performance evaluation for BICM-ID with square QAM constellations in rayleigh fading channels," *IEEE J. Select. Areas Commun.*, vol. 19, no. 5, pp. 944–957, May 2001.
- [40] K. M. Chugg and A. Anastasopoulos, "On symbol error probability bounds for ISI-like channels," *IEEE Trans. Commun.*, vol. 49, no. 10, pp. 1704–1709, Oct. 2001.
- [41] J. M. Cioffi, G. P. Dudevoir, M. V. Eyuboglu, and G. D. Forney, Jr, "MMSE decision-feedback equalizers and coding – Part I: Equalization results," *IEEE Trans. Commun.*, vol. 43, no. 10, pp. 2582–2594, Oct. 1995.
- [42] G. C. Clark, Jr and J. B. Cain, *Error-Correction Coding for Digital Communications*. New-York, NY: Plenum Press, 1981.
- [43] 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.
- [44] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, 2nd ed. Cambridge, MA: MIT Press, 2001.
- [45] A. Dejonghe and L. Vandendorpe, "Turbo-equalization for multilevel modulation: An efficient low-complexity scheme," in *Proc. IEEE Int. Conf. Commun. ICC'02*, vol. 3, New York City, NY, 28 Apr.-2 May 2002, pp. 1863–1867.
- [46] ———, "A comparison of bit and symbol interleaving in MMSE turbo-equalization," in *Proc. IEEE Int. Conf. Commun. ICC'03*, vol. 4, Anchorage, Alaska, 11-15 May 2003, pp. 2928–2932.
- [47] P. Didier, "La turbo-égalisation et son application aux communications radiomobiles," Ph.D. dissertation, Univ. de Bretagne Occidentale, Brest, France, Dec. 1996, (in French).
- [48] D. Divsalar, S. Dolinar, and F. Pollara, "Iterative turbo decoder analysis based on density evolution," Jet Propulsion Laboratory, Pasadena, CA, JPL TMO Progress Report vol. 42-144, Feb. 2001.
- [49] S. Dolinar and D. Divsalar, "Weight distributions for turbo-codes using random and nonrandom permutations," Jet Propulsion Laboratory, Pasadena, CA, JPL TDA Progress Report vol. 42-122, Aug. 1995.
- [50] C. Douillard, M. Jézéquel, C. Berrou, A. Picart, P. Didier, and A. Glavieux, "Iterative correction of intersymbol interference: Turbo-equalization," *European Trans. Telecommun.*, vol. 6, no. 5, pp. 507–511, Sept.-Oct. 1995.
- [51] A. Duel-Hallen and C. Heegard, "Delayed decision-feedback sequence estimation," *IEEE Trans. Commun.*, vol. 37, no. 5, pp. 428–436, May 1989.

- [52] D. L. Duttweiler, J. E. Mazo, and D. G. Messerschmitt, "An upper bound on the error probability in decision-feedback equalization," *IEEE Trans. Inform. Theory*, vol. IT-20, no. 4, pp. 490–497, July 1974.
- [53] H. El Gamal and E. Geraniotis, "Iterative multiuser detection of coded CDMA signals in AWGN and fading channels," *IEEE J. Select. Areas Commun.*, vol. 18, no. 1, pp. 30–41, Jan. 2000.
- [54] H. El Gamal and A. R. Hammons, Jr, "Analyzing the turbo decoder using the gaussian approximation," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 671–686, Feb. 2001.
- [55] V. Erceg, D. G. Michelson, S. S. Ghassemzadeh, L. J. Greenstein, A. J. Rustako, Jr, P. B. Guerlain, M. K. Dennison, R. S. Roman, D. J. Barnickel, S. C. Wang, and R. R. Miller, "A model for the multipath delay profile of fixed wireless channels," *IEEE J. Select. Areas Commun.*, vol. 17, no. 3, pp. 399–410, Mar. 1999.
- [56] M. V. Eyuboglu, "Detection of coded modulation signals on linear, severely distorted channels using decision-feedback noise prediction with interleaving," *IEEE Trans. Commun.*, vol. 36, no. 4, pp. 401–409, Apr. 1988.
- [57] M. V. Eyuboglu and S. U. H. Qureshi, "Reduced-state sequence estimation with set partitioning and decision-feedback," *IEEE Trans. Commun.*, vol. 36, no. 1, pp. 13–20, Jan. 1988.
- [58] ———, "Reduced-state sequence estimation for coded modulation on intersymbol interference channels," *IEEE J. Select. Areas Commun.*, vol. 7, no. 6, pp. 989–995, Aug. 1989.
- [59] D. D. Falconer, S. L. Ariyavisitakul, A. Benyamin-Seeyar, and B. Eidson, "Frequency domain equalization for single-carrier broadband wireless systems," *IEEE Commun. Mag.*, vol. 40, no. 4, pp. 58–66, Apr. 2002.
- [60] D. D. Falconer and F. R. Magee, Jr, "Adaptive memory truncation for maximum-likelihood sequence estimation," *Bell Sys. Tech. J.*, vol. 59, no. 9, pp. 1541–1562, Nov. 1973.
- [61] R. F. H. Fischer, *Precoding and Signal Shaping for Digital Transmission*. New-York, NY: John Wiley & Sons, 2002.
- [62] G. D. Forney, Jr, "Lower bounds on error probability in the presence of large intersymbol interference (Corresp.)," *IEEE Trans. Commun. Technol.*, vol. COM-20, no. 2, pp. 76–77, Feb. 1972.
- [63] ———, "Maximum-likelihood sequence estimation of digital sequences in the presence of intersymbol interference," *IEEE Trans. Inform. Theory*, vol. IT-18, no. 3, pp. 363–378, May 1972.
- [64] ———, "The Viterbi algorithm," *Proc. IEEE*, vol. 61, no. 3, pp. 268–278, Mar. 1973.

- [65] G. D. Forney, Jr and G. Ungerboeck, "Modulation and coding for linear gaussian channels," *IEEE Trans. Inform. Theory*, vol. 44, no. 6, pp. 2384–2415, Oct. 1998.
- [66] G. J. Foschini, "Performance bound for maximum-likelihood reception of digital data," *IEEE Trans. Inform. Theory*, vol. IT-21, no. 1, pp. 47–50, Jan. 1975.
- [67] C. Fragouli, N. Al-Dhahir, S. N. Diggavi, and W. Turin, "Prefiltered space-time M-BCJR equalizer for frequency-selective channels," *IEEE Trans. Commun.*, vol. 50, no. 5, pp. 742–753, May 2002.
- [68] R. L. Frank and S. A. Zadoff, "Phase shift pulse codes with good periodic correlation properties (Corresp.)," *IRE Trans. Inform. Theory*, vol. IT-8, no. 6, pp. 381–382, Oct. 1962.
- [69] V. Franz and G. Bauch, "Turbo-detection for enhanced data for GSM evolution," in *Proc. IEEE Veh. Technol. Conf. VTC'99 Fall*, Amsterdam, The Netherlands, 19-22 Sept. 1999, pp. 2954–2958.
- [70] R. G. Gallager, *Low Density Parity Check Codes*. Cambridge, MA: MIT Press, 1963.
- [71] ———, *Information Theory and Reliable Communication*. New-York, NY: John Wiley & Sons, 1968.
- [72] A. Gersho and T. L. Lim, "Adaptive cancellation of intersymbol interference for data transmission," *Bell Sys. Tech. J.*, vol. 60, no. 11, pp. 1997–2021, Nov. 1981.
- [73] W. Gerstacker and R. Schober, "Equalization concepts for EDGE," *IEEE Trans. Wireless Commun.*, vol. 1, no. 1, pp. 190–199, Jan. 2002.
- [74] W. H. Gerstacker and J. B. Huber, "Maximum SNR decision-feedback equalization with FIR filters: Filter optimization and a signal processing application," in *Proc. IEEE Int. Conf. Commun. ICC'96*, vol. 2, Dallas, TX, 23-27 June 1996, pp. 1188–1192.
- [75] A. Glavieux, C. Laot, and J. Labat, "Turbo equalization over a frequency selective channel," in *Proc. Int. Symp. on Turbo Codes & Related Topics*, Brest, France, 3-5 Sept. 1997, pp. 96–102.
- [76] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 3rd ed. Baltimore, MD: The Johns Hopkins University Press, 1996.
- [77] A. Gorokhov and M. van Dijk, "Optimized bit labeling maps for bit-interleaved transmission with turbo demodulation," in *Proc. IEEE Veh. Technol. Conf. VTC'01 Spring*, vol. 2, Rhodes, Greece, 6-9 May 2001, pp. 1459–1463.
- [78] A. J. Grant, "Convergence of non-binary iterative decoding," in *Proc. IEEE Global Telecommun. Conf. GLOBECOM'01*, vol. 2, San Antonio, TX, 25-29 Nov. 2001, pp. 1058–1062.

- [79] R. M. Gray, "Toeplitz and circulant matrices: A review," Information and Systems Lab., Dept. of Electrical Eng., Stanford Univ., Stanford, CA, Tech. Rep., Revised Aug. 2002, [Online]. Available: <http://ee.stanford.edu/~gray/>.
- [80] J. Hagenauer, "The turbo principle: Tutorial introduction and state of the art," in *Proc. Int. Symp. on Turbo Codes & Related Topics*, Brest, France, 3-5 Sept. 1997, pp. 1-11.
- [81] —, "Log-likelihood ratios, mutual information and EXIT charts – A primer," Lecture at the 2002 12th Joint Conf. on Commun. and Coding, Saas Fe, Switzerland, 5 mar. 2002.
- [82] J. Hagenauer and C. Kuhn, "Turbo equalization for channels with high memory using a list-sequential (LISS) equalizer," in *Proc. 3rd Int. Symp. on Turbo Codes & Related Topics*, Brest, France, 1-5 Sept. 2003, pp. 9-13.
- [83] T. Hashimoto, "A list-type reduced-constraint generalization of the Viterbi algorithm," *IEEE Trans. Inform. Theory*, vol. IT-33, no. 6, pp. 866-876, Nov. 1987.
- [84] J. F. Hayes, "The Viterbi algorithm applied to digital data transmission," *IEEE Commun. Mag.*, vol. 40, no. 5, pp. 26-32, May 2002.
- [85] J. F. Hayes, T. M. Cover, and J. B. Riera, "Optimal sequence detection and optimal symbol-by-symbol detection: Similar algorithms," *IEEE Trans. Commun.*, vol. 30, no. 1, pp. 152-157, Jan. 1982.
- [86] C. Heegard and S. B. Wicker, *Turbo Coding*. Norwell, MA: Kluwer Academic Publishers, 1999.
- [87] A. P. Hekstra, "An alternative to metric rescaling in Viterbi decoders," *IEEE Trans. Commun.*, vol. 37, no. 11, pp. 1220-1222, Nov. 1989.
- [88] M. H  lard, P.-J. Bouvet, C. Langlais, Y.-M. Morgan, and I. Siaud, "On the performance of a turbo-equalizer including blind equalizer over time and frequency-selective channel. Comparison with an OFDM system," in *Proc. 3rd Int. Symp. on Turbo Codes & Related Topics*, Brest, France, 1-5 Sept. 2003, pp. 419-422.
- [89] *IEEE Std 802.16a-2003 – Amendment 2 to IEEE Std 802.16-2001: Medium Access Control Modifications and Additional Physical Layer Specifications for 2-11 GHz*, IEEE-SA Standards Board, New-York, NY, USA, Apr. 2003, [Online]. Available: <http://www.wirelessman.org>.
- [90] H. Imai and S. Hirakawa, "A new multilevel coding method using error-correcting codes," *IEEE Trans. Inform. Theory*, vol. IT-23, no. 3, pp. 371-377, May 1977.
- [91] S. H. Jamali and T. Le-Ngoc, *Coded-Modulation Techniques for Fading Channels*. Norwell, MA: Kluwer Academic Publishers, 1994.

- [92] R. Johannesson and K. S. Zigangirov, *Fundamentals of Convolutional Coding*. Piscataway, NJ: IEEE Press, 1999.
- [93] T. Kailath, A. H. Sayed, and B. Hassibi, *Linear Estimation*. Upper Saddle River, NJ: Prentice Hall, 2000.
- [94] R. E. Kamel and Y. Bar-Ness, "Reduced-complexity sequence estimation using set partitioning," *IEEE Trans. Commun.*, vol. 44, no. 9, pp. 1057–1063, Sept. 1996.
- [95] R. Knopp, "Coding and multiple-access over fading channels," Ph.D. dissertation, EPFL / Institut Eurécom, Lausanne, Switzerland, 1997.
- [96] R. Knopp and P. A. Humblet, "On coding for block fading channels," *IEEE Trans. Inform. Theory*, vol. 46, no. 1, pp. 189–205, Jan. 2000.
- [97] W. Koch and A. Baier, "Optimum and sub-optimum detection of coded data disturbed by time-varying intersymbol interference," in *Proc. IEEE Global Telecommun. Conf. GLOBECOM'90*, vol. 3, San Diego, CA, 2-5 Dec. 1990, pp. 1679–1684.
- [98] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 498–519, Feb. 2001.
- [99] C. Langlais, "Étude et amélioration d'une technique de réception itérative: Turbo-égalisation," Ph.D. dissertation, Univ. de Rennes I, Rennes, France, Nov. 2002, (in French).
- [100] C. Langlais and M. Héland, "Optimization of the equalization and the interleaving in turbo-equalization for a frequency-selective fading channel," in *Proc. IEEE Int. Conf. Commun. ICC'02*, vol. 3, New York City, NY, 28 Apr.-2 May 2002, pp. 1868–1872.
- [101] C. Laot, "Égalisation autodidacte et turbo-égalisation – Application aux canaux sélectifs en fréquence," Ph.D. dissertation, Univ. de Rennes I, Rennes, France, July 1997, (in French).
- [102] C. Laot, A. Glavieux, and J. Labat, "Turbo-equalization: Adaptive equalization and channel decoding jointly optimized," *IEEE J. Select. Areas Commun.*, vol. 19, no. 9, pp. 1744–1752, Sept. 2001.
- [103] C. Laot, R. Le Bidan, and D. Leroux, "Low complexity linear turbo equalization: A possible solution for EDGE," *Submitted to IEEE Trans. Wireless Commun. (under revision)*, Aug. 2002, [Online]. Available: <http://www-sc.enst-bretagne.fr/~laot>.
- [104] —, "Multiple-input turbo-equalization over time-varying frequency selective channels," in *Proc. XI European Signal Proc. Conf. EUSIPCO 2002*, Toulouse, France, 3-6 Sept. 2002.
- [105] R. Le Bidan, "Étude préliminaire relative à l'implémentation du turbo-égaliseur MMSE IC-LE sur plate-forme DSP," Dept. S&C, ENST Bretagne, Brest, France, Rapport Interne, Dec. 2002, (in French).

- [106] R. Le Bidan, C. Laot, and D. Leroux, "La turbo-détection: Principe et performances," Dept. S&C, ENST Bretagne, Brest, France, Rapport de Recherche, Contrat France Télécom R&D CRE 011B032, Oct. 2001, (in French).
- [107] —, "Nouvelles structures d'égalisation linéaire à entrée et sortie pondérée – Application à la turbo-égalisation pour les transmissions en mode paquet," Dept. S&C, ENST Bretagne, Brest, France, Rapport de Recherche, Contrat France Télécom R&D CRE 011B032, Sept. 2002, (in French).
- [108] —, "Fixed-point implementation of an efficient low-complexity turbo-equalization scheme," in *Proc. 3rd Int. Symp. on Turbo Codes & Related Topics*, Brest, France, 1-5 Sept. 2003, pp. 415–418.
- [109] R. Le Bidan, C. Laot, D. Leroux, and A. Glavieux, "Analyse de la convergence en turbo-détection," in *Actes du 18e Colloque GRETSI'01*, Toulouse, France, 10-13 Sept. 2001, (in French).
- [110] S. Le Goff, "Signalling constellations for power-efficient bit-interleaved coded modulation schemes," *IEE Proc. Commun.*, vol. 150, no. 3, pp. 141–148, June 2003.
- [111] S. Le Goff, A. Glavieux, and C. Berrou, "Turbo-codes and high spectral efficiency modulation," in *Proc. IEEE Int. Conf. Commun. ICC'94*, New Orleans, LA, 1-5 May 1994, pp. 645–649.
- [112] E. A. Lee and D. G. Messerschmitt, *Digital Communication*, 2nd ed. Norwell, MA: Kluwer Academic Publisher, 1994.
- [113] I. Lee and J. M. Cioffi, "A fast computation algorithm for the decision feedback equalizer," *IEEE Trans. Commun.*, vol. 43, no. 11, pp. 2742–2749, Nov. 1995.
- [114] W. U. Lee and F. S. Hill, Jr, "A maximum-likelihood sequence estimator with decision-feedback equalizer," *IEEE Trans. Commun.*, vol. COM-25, no. 9, pp. 971–979, Sept. 1977.
- [115] X. Li, A. Chindapol, and J. A. Ritcey, "Bit-interleaved coded modulation with iterative decoding and 8PSK signaling," *IEEE Trans. Commun.*, vol. 50, no. 8, pp. 1250–1257, Aug. 2002.
- [116] X. Li and J. A. Ritcey, "Bit-interleaved coded modulation with iterative decoding," *IEEE Commun. Lett.*, vol. 1, no. 6, pp. 169–171, Nov. 1997.
- [117] —, "Trellis-coded modulation with bit interleaving and iterative decoding," *IEEE J. Select. Areas Commun.*, vol. 17, no. 4, pp. 715–724, Apr. 1999.
- [118] Y. Li, B. Vucetic, and Y. Sato, "Optimum soft-output detection for channels with intersymbol interference," *IEEE Trans. Inform. Theory*, vol. 41, no. 3, pp. 704–713, May 1995.

- [119] R. W. Lucky, J. Salz, and E. J. Weldon, Jr, *Principles of Data Communication*. New-York, NY: McGraw-Hill, 1968.
- [120] P. Magniez, P. Duhamel, A. Roumy, and I. Fijalkow, "Turbo-equalization applied to trellis-coded modulation," in *Proc. IEEE Veh. Technol. Conf. VTC'99 Fall*, vol. 5, Amsterdam, The Netherlands, 19-22 Sept. 1999, pp. 2556–2560.
- [121] P. Magniez, B. Muquet, P. Duhamel, V. Buzenac, and M. de Courville, "Optimal decoding of bit-interleaved modulations: Theoretical aspects and practical algorithms," in *Proc. 2nd Int. Symp. on Turbo Codes & Related Topics*, Brest, France, 4-7 Sept. 2000, pp. 169–172.
- [122] A. Matache, S. Dolinar, and F. Pollara, "Stopping rules for turbo decoders," Jet Propulsion Laboratory, Pasadena, CA, JPL TMO Progress Report vol. 42-142, Aug. 2000.
- [123] J. E. Mazo, "A geometric derivation of Forney's upper bound," *Bell Sys. Tech. J.*, vol. 54, pp. 1087–1094, July-Aug. 1975.
- [124] R. J. McEliece, "On the BCJR trellis for linear block codes," *IEEE Trans. Inform. Theory*, vol. 42, no. 4, pp. 1072–1092, July 1996.
- [125] ———, "The forward-backward algorithm," Lecture Notes for EE-127C, Caltech, CA, Apr. 2001, [Online]. Available: <http://www.systems.caltech.edu/EE/Courses/EE127/EE127C/>.
- [126] R. J. McEliece, D. J. C. MacKay, and J.-F. Cheng, "Turbo decoding as an instance of Pearl's "belief propagation" algorithm," *IEEE J. Select. Areas Commun.*, vol. 16, no. 2, pp. 140–152, Feb. 1998.
- [127] H. Meyr and G. Ascheid, *Synchronization in Digital Communications Vol. 1: Phase-, Frequency-Locked Loops, and Amplitude Control*. New-York, NY: John Wiley & Sons, 1990.
- [128] A. Milewski, "Periodic sequences with optimal properties for channel estimation and fast start-up equalization," *IBM J. Res & Dev.*, vol. 27, no. 5, pp. 426–431, Sept. 1983.
- [129] G. Montorsi and S. Benedetto, "Design of fixed-point iterative decoders for concatenated codes with interleavers," *IEEE J. Select. Areas Commun.*, vol. 19, no. 5, pp. 871–882, May 2001.
- [130] P. Moqvist and T. M. Aulin, "Improved lower bounds on the symbol error probability for ISI channels," *IEE Proc. Commun.*, vol. 147, no. 6, pp. 351–354, Dec. 2000.
- [131] M. S. Mueller and J. Salz, "A unified theory of data-aided equalization," *Bell Sys. Tech. J.*, vol. 60, no. 11, pp. 2023–2038, Nov. 1981.
- [132] A. V. Oppenheim, R. W. Schaffer, and J. R. Buck, *Discrete-Time Signal Processing*, 2nd ed. Upper Saddle River, NJ: Prentice Hall, 1999.

- [133] R. Otnes and M. Tüchler, "Soft iterative channel estimation for turbo equalization: Comparison of channel estimation algorithms," in *Proc. IEEE Int. Conf. on Commun. Systems ICSS'02*, Singapore, 25-28 Nov. 2002.
- [134] A. Papoulis, *Signal Analysis*. New-York, NY: McGraw-Hill, 1977.
- [135] B. Penther, D. Castelain, and H. Kubo, "A modified turbo-detector for long-delay spread channels," in *Proc. 2nd Int. Symp. on Turbo Codes & Related Topics*, Brest, France, 4-7 Sept. 2000, pp. 295–298.
- [136] O. Pothier, "Codes composites construits à partir de graphes et leur décodage itératif," Ph.D. dissertation, ENST, Paris, France, Jan. 2000.
- [137] J. G. Proakis, "Adaptive nonlinear filtering techniques for data transmission," in *Proc. IEEE Symp. on Adaptive Processes, Decision and Control*, 1970, pp. XV.2.1–XV.2.5.
- [138] ———, *Digital Communications*, 4th ed. New-York, NY: McGraw-Hill, 2000.
- [139] S. U. H. Qureshi, "Adaptive equalization," *Proc. IEEE*, vol. 73, no. 9, pp. 1349–1387, Sept. 1985.
- [140] R. Raheli, A. Polydoros, and C.-K. Tzou, "Per-survivor processing: A general approach to MLSE in uncertain environments," *IEEE Trans. Commun.*, vol. 43, no. 2/3/4, pp. 354–364, Feb.-Apr. 1995.
- [141] D. Reynolds and X. Wang, "Low-complexity turbo-equalization for diversity channels," *Signal Proc.*, vol. 81, no. 5, pp. 989–995, May 2001.
- [142] T. J. Richardson and R. L. Urbanke, "An introduction to the analysis of iterative coding systems," in *Proc. 1999 IMA Summer Program: Code Systems and Graphical Models*, Minneapolis, MN, 2-6 Aug. 1999, pp. 12–25.
- [143] ———, "The capacity of low-density parity-check codes under message-passing decoding," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 599–618, Feb. 2001.
- [144] P. Robertson, E. Villebrun, and P. Hoeher, "A comparison of optimal and sub-optimal MAP decoding algorithms operating in the log domain," in *Proc. IEEE Int. Conf. Commun. ICC'95*, vol. 2, Seattle, WA, 18-22 June 1995, pp. 1009–1013.
- [145] P. Robertson and T. Wörz, "Bandwidth-efficient turbo trellis-coded modulation using punctured components codes," *IEEE J. Select. Areas Commun.*, vol. 16, no. 2, pp. 206–218, Feb. 1998.
- [146] A. Roumy, "Égalisation et décodage conjoints: Méthodes turbo," Ph.D. dissertation, Univ. de Cergy-Pontoise, Cergy-Pontoise, France, Oct. 2000, (in French).

- [147] T. M. Sailer, "Decision feedback equalization for powerline and HIPERLAN," Ph.D. dissertation, Swiss Federal Institute of Technology, Zürich, Switzerland, Feb. 2001.
- [148] H. Sari, "Algorithmes d'égalisation adaptative d'un canal dispersif," Ph.D. dissertation, ENST, Paris, France, Oct. 1980, (in French).
- [149] H. Sari, G. Karam, and I. Jeanclaude, "Transmission techniques for digital terrestrial TV broadcasting," *IEEE Commun. Mag.*, vol. 33, no. 2, pp. 100–109, Feb. 1995.
- [150] A. H. Sayed, *Fundamentals of Adaptive Filtering*. New-York, NY: IEEE Press / John Wiley & Sons, 2003.
- [151] B. Scanavino, G. Montorsi, and S. Benedetto, "Convergence properties of iterative decoders working at bit and symbol level," in *Proc. IEEE Global Telecommun. Conf. GLOBECOM'01*, vol. 2, San Antonio, TX, 25–29 Nov. 2001, pp. 1037–1041.
- [152] C. Schlegel, *Trellis Coding*. Piscataway, NJ: IEEE Press, 1997.
- [153] M. Sellathurai and S. Haykin, "TURBO-BLAST for wireless communications: Theory and experiments," *IEEE Trans. Signal Processing*, vol. 50, no. 10, pp. 2538–2546, Oct. 2002.
- [154] E. Shamash and K. Yao, "On the structure and performance of a linear decision feedback equalizer based on the minimum error probability," in *Proc. IEEE Int. Conf. Commun. ICC'74*, June 1974, pp. 25F1–25F5.
- [155] C. E. Shannon, "A mathematical theory of communications," *Bell Sys. Tech. J.*, vol. 27, pp. 379–423 (Part I) and 623–656 (Part II), July 1948.
- [156] C. B. Shung, P. H. Siegel, G. Ungerboeck, and H. K. Thapar, "VLSI architectures for metric normalization in the Viterbi algorithm," in *Proc. IEEE Int. Conf. Commun. ICC'90*, vol. 4, Atlanta, GA, 16–19 Apr. 1990, pp. 1723–1728.
- [157] S. J. Simmons, "Breadth-first trellis decoding with adaptive effort," *IEEE Trans. Commun.*, vol. 38, no. 1, pp. 3–12, Jan. 1990.
- [158] J. H. Smee and N. C. Beaulieu, "Error-rate evaluation of linear equalization and decision feedback equalization with error propagation," *IEEE Trans. Commun.*, vol. 46, no. 5, pp. 656–665, May 1998.
- [159] M. Stojanovic, "Recent advances in high-speed underwater acoustic communications," *IEEE J. Oceanic Eng.*, vol. 21, no. 2, pp. 125–136, Apr. 1996.
- [160] C.-E. W. Sundberg and N. Seshadri, "Coded modulations for fading channels: An overview," *European Trans. Telecommun.*, vol. 4, no. 3, pp. 309–324, May–June 1993.

- [161] S. ten Brink, "Convergence of iterative decoding," *Electron. Lett.*, vol. 35, no. 13, pp. 1117–1119, June 1999.
- [162] —, "Designing iterative decoding schemes with the extrinsic information transfer chart," *AEÜ Int. J. Electron. Commun.*, vol. 54, no. 6, pp. 389–398, Nov. 2000.
- [163] —, "Code characteristics matching for iterative decoding of serially concatenated codes," *Ann. Télécommun.*, vol. 56, no. 7-8, pp. 394–408, 2001.
- [164] S. ten Brink and G. Kramer, "Turbo processing for scalar and vector channels," in *Proc. 3rd Int. Symp. on Turbo Codes & Related Topics*, Brest, France, 1-5 Sept. 2003, pp. 23–30.
- [165] S. ten Brink, J. Speidel, and R.-H. Yan, "Iterative demapping and decoding for multilevel modulation," in *Proc. IEEE Global Telecommun. Conf. GLOBECOM'98*, vol. 1, Sidney, Australia, 8-12 Nov. 1998, pp. 579–584.
- [166] H. Thampi S. and J. Govindarajan, "DSP/BIOS, RTDX and host-target communications," Texas Instruments, Dallas, TX, TI Application Report SPRA895, Feb. 2003.
- [167] "TMS320VC5509 fixed-point DSP data manual," Texas Instruments, Dallas, TX, TI User Guide SPRS163A, July 2001.
- [168] "TMS320C55x DSP functional overview," Texas Instruments, Dallas, TX, TI User Guide SPRU312, June 2000.
- [169] "TMS320C55x DSP programmer's guide – Preliminary draft," Texas Instruments, Dallas, TX, TI User Guide SPRS376A, July 2001.
- [170] "TMS320C55x DSP technical overview," Texas Instruments, Dallas, TX, TI User Guide SPRU393, Feb. 2000.
- [171] "TMS320C55x DSP library programmer's reference," Texas Instruments, Dallas, TX, TI User Guide SPRU422F, Oct. 2002.
- [172] M. Tüchler, "Iterative equalization and decoding using priors," Master's thesis, Univ. of Illinois, Urbana-Champaign, IL, May 2000.
- [173] M. Tüchler and J. Hagenauer, "Turbo equalization using frequency domain equalizers," in *Proc. 38th Annual Allerton Conf. on Commun., Control and Computing*, Monticello, IL, 4-6 Oct. 2000.
- [174] M. Tüchler, R. Koetter, and A. C. Singer, "Turbo-equalization: Principles and new results," *IEEE Trans. Commun.*, vol. 50, no. 5, pp. 754–767, May 2002.
- [175] M. Tüchler, A. C. Singer, and R. Koetter, "Minimum mean squared error equalization using a priori information," *IEEE Trans. Signal Processing*, vol. 50, no. 3, pp. 673–683, Mar. 2002.

- [176] M. Tüchler, S. ten Brink, and J. Hagenauer, "Measures for tracing convergence of iterative decoding algorithms," in *Proc. 4th Intern. ITG Conf. on Source and Channel Coding*, Berlin, Germany, Jan. 2002, pp. 53–60.
- [177] G. L. Turin, "Introduction to spread-spectrum anti-multipath techniques and their application to urban digital radio," *Proc. IEEE*, vol. 68, no. 3, pp. 328–353, Mar. 1980.
- [178] G. Ungerboeck, "Adaptive maximum-likelihood receiver for carrier-modulated data-transmission systems," *IEEE Trans. Commun.*, vol. 22, no. 5, pp. 624–636, May 1974.
- [179] —, "Channel coding with multilevel/phase signals," *IEEE Trans. Inform. Theory*, vol. IT-28, no. 1, pp. 55–67, Jan. 1982.
- [180] —, "Trellis-coded modulations with redundant signal sets. Part I: Introduction. Part II: State of the art," *IEEE Commun. Mag.*, vol. 25, no. 2, pp. 5–21, Feb. 1987.
- [181] G. Ungerboeck and I. Csajka, "On improving data-link performance by increasing channel alphabet and introducing sequence coding," in *Proc. IEEE Int. Symp. Inform. Theory*, Ronneby, Sweden, June 1976.
- [182] A. Vardy, "Trellis structure of codes," in *Handbook of Coding Theory – Vol. II*, V. S. Pless and W. C. Huffman, Eds. North-Holland: Elsevier Science, 1998, pp. 1989–2117.
- [183] S. Verdú, "Maximum likelihood sequence detection for intersymbol interference channels: A new upper bound on error probability," *IEEE Trans. Inform. Theory*, vol. IT-33, no. 1, pp. 62–68, Jan. 1987.
- [184] S. Vialle, "Construction et analyse de nouvelles structures de codage adaptées au traitement itératif," Ph.D. dissertation, ENST, Paris, France, Dec. 2000, (in French).
- [185] R. Visoz, "Iterative and joint processing for wireless mobile systems," Ph.D. dissertation, ENST, Paris, France, Mar. 2002.
- [186] A. J. Viterbi, J. K. Wolf, E. Zehavi, and R. Padovani, "A pragmatic approach to trellis-coded modulation," *IEEE Commun. Mag.*, vol. 27, no. 7, pp. 11–19, July 1989.
- [187] G. M. Vitetta, B. D. Hart, A. Mämmelä, and D. P. Taylor, "Equalization techniques for single carrier, unspread digital modulations," in *Wideband Wireless Digital Communications*, A. F. Molish, Ed. Upper Saddle River, NJ: Prentice Hall, 2001, pp. 155–308.
- [188] B. Vucetic and J. Yuan, *Turbo Codes: Principles and Applications*. Norwell, MA: Kluwer Academic Publishers, 2000.
- [189] X. Wang and H. V. Poor, "Iterative (turbo) soft interference cancellation and decoding for coded CDMA," *IEEE Trans. Commun.*, vol. 47, no. 7, pp. 1046–1061, July 1999.

- [190] E. W. Weisstein, *CRC Concise Encyclopedia of Mathematics*, 2nd ed. Boca Raton, FL: CRC Press, 2002, [Online]. Available: <http://mathworld.wolfram.com/>.
- [191] K. Wesolowski, "On the performance and convergence of the adaptive canceller of intersymbol interference in data transmission," *IEEE Trans. Commun.*, vol. 33, no. 5, pp. 425–432, May 1985.
- [192] —, "Efficient digital receiver structure for trellis-coded signals transmitted through channels with intersymbol interference," *Electron. Lett.*, vol. 23, no. 24, pp. 1265–1267, 19th Nov. 1987.
- [193] N. Wiberg, "Codes and iterative decoding on general graphs," Ph.D. dissertation, Dept. Elec. Eng., Linköping Univ., Linköping, Sweden, 1996.
- [194] J. K. Wolf and E. Zehavi, "P2 codes: Pragmatic trellis codes utilizing punctured convolutional codes," *IEEE Commun. Mag.*, vol. 33, no. 2, pp. 94–99, Feb. 1995.
- [195] J. M. Wozencraft and I. M. Jacobs, *Principles of Communication Engineering*. New-York, NY: John Wiley & Sons, 1965.
- [196] Z.-N. Wu and J. M. Cioffi, "Low-complexity iterative decoding with decision-aided equalization for magnetic recording channels," *IEEE J. Select. Areas Commun.*, vol. 19, no. 4, pp. 699–708, Apr. 2001.
- [197] F. Xiong, A. Zerik, and E. Shwedyk, "Sequential sequence estimation for channels with intersymbol interference of finite or infinite length," *IEEE Trans. Commun.*, vol. 38, no. 6, pp. 795–804, June 1990.
- [198] K. Yao, "On minimum average probability of error expression for a binary pulse-communication system with intersymbol interference (Corresp.)," *IEEE Trans. Inform. Theory*, vol. 18, no. 4, pp. 528–531, July 1972.
- [199] E. Zehavi, "8-PSK trellis codes for a Rayleigh channel," *IEEE Trans. Commun.*, vol. 40, no. 5, pp. 873–884, May 1992.
- [200] Y. R. Zheng and C. Xiao, "Improved models for the generation of multiple uncorrelated Rayleigh fading waveforms," *IEEE Commun. Lett.*, vol. 6, no. 6, pp. 256–258, June 2002.
- [201] K. Zhouh, J. G. Proakis, and F. Ling, "Decision-feedback equalization of time-dispersive channels with coded modulation," *IEEE Trans. Commun.*, vol. 38, no. 1, pp. 18–24, Jan. 1990.

AVIS DU JURY SUR LA REPRODUCTION DE LA THESE SOUTENUE

Titre de la thèse : Application de la Turbo-Egalisation pour les transmissions avec modulations codées sur canaux sélectifs en fréquence. Etude théorique et implémentation sur processeur DSP de turbo-égaliseurs à complexité réduite.

Nom Prénom de l'auteur : LE BIDAN Raphaël

Membres du jury : Monsieur SARI
Monsieur VANDENDORPE
Monsieur CITERNE
Monsieur GLAVIEUX
Madame HELARD
Monsieur LAOT
Madame LEROUX

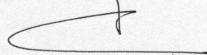
Président du jury : *Jacques CITERNE*

Date de la soutenance : 07/11/2003

Reproduction de la thèse soutenue :

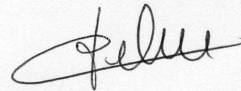
- ☒ Thèse pouvant être reproduite en l'état
☐ Thèse ne pouvant être reproduite
☐ Thèse pouvant être reproduite après corrections suggérées

Le Directeur,


P. FLEISCHMANN

Rennes, le 07/11/2003

Signature du Président du jury



Résumé

Cette thèse traite du problème de l'égalisation des modulations codées pour les transmissions haut-débit sur canaux sélectifs en fréquence, sujets au phénomène d'interférence entre symboles. Nous considérons plus particulièrement la Turbo-Égalisation, qui instaure un échange réciproque d'information entre l'égaliseur et le décodeur, et ce de manière itérative. Nous étudions dans un premier temps le turbo-égaliseur MAP, qui utilise un égaliseur optimal au sens de la minimisation de la probabilité d'erreur par symbole. Nous montrons que ce récepteur offre des gains de performances importants en comparaison avec les récepteurs conventionnels où les opérations d'égalisation et décodage sont généralement effectuées de manière disjointe. En contrepartie, la complexité du turbo-égaliseur MAP devient rapidement prohibitive en présence de modulations à grand nombre d'états et sur des canaux présentant des étalements temporels importants. En conséquence, nous nous intéressons à une seconde classe de turbo-égaliseurs de moindre complexité et reposant sur des égaliseurs à base de filtres linéaires, optimisés selon le critère MEQM. La nouveauté consiste ici à prendre en compte explicitement la présence d'information *a priori* dans le calcul des coefficients des filtres, ce qui conduit à des structures bien plus performantes que les égaliseurs MEQM classiques. Nos études montrent que ce type de récepteur constitue une solution attractive pour les transmissions à grande efficacité spectrale sur canaux sélectifs en fréquence. Finalement, nous présentons la mise en oeuvre d'un turbo-égaliseur MEQM sur un DSP virgule-fixe et faible coût, le TMS320VC5509, typiquement destiné aux terminaux mobiles. Nous obtenons ainsi un débit utile de 42 Kbits/s après 5 itérations avec une implémentation en langage C, ce qui démontre la faisabilité de tels récepteurs avec les moyens technologiques actuels.

Mots clés : égalisation, codage de canal, modulations codées, turbo-égalisation.

Abstract

This thesis deals with the problem of combining equalization with decoding for high data rate bandwidth-efficient transmissions over frequency-selective channels. Specifically, we focus on Turbo-Equalization which establishes an exchange of soft information between the equalizer and the decoder in an iterative manner. We first consider MAP turbo-equalizers which rely on trellis-based equalizers optimum in the sense of minimizing the symbol error probability. We show that these receivers achieve significant performance gains over the conventional approach where equalization and decoding are performed separately. However, the complexity of MAP turbo-equalizers becomes rapidly prohibitive for transmissions with high-order modulations over long delay spread channels. Hence, we then investigate reduced-complexity turbo-equalizers relying on filtering-based equalizers optimized according to the MMSE criterion. These equalizers share the common property of accounting explicitly for the presence of *a priori* information in the computation of the optimum filters coefficients and thus outperform classical MMSE equalizers. Our analysis shows that MMSE turbo-equalizers offer an attractive alternative to MAP turbo-equalizers for multilevel coded transmissions over long delay spread frequency-selective channels. We finally describe the implementation of an MMSE turbo-equalizer on the low-cost fixed-point TMS320VC5509 DSP device which is typically targeted at portable terminals. Using only C programming, a promising data rate of 42 Kbits/s has been achieved with 5 iterations. This result demonstrates the possibility of realizing such receivers in practice with current hardware capabilities.

Keywords : equalization, channel coding, coded modulation, turbo-equalization.