Reduced Receivers for Faster-than-Nyquist Signaling and General Linear Channels

Prlja, Adnan

2013

Thesis

Reduced Receivers for Faster-than-Nyquist Signaling and General Linear Channels

Adnan Prlja

Lund 2013

Department of Electrical and Information Technology
Lund University
Box 118, SE-221 00 LUND
SWEDEN

This thesis is set in Computer Modern 10pt
with the LaTeX Documentation System

Series of licentiate and doctoral theses
No. 48
ISSN 1654-790X
ISBN 978-91-7473-427-0

Printed in Sweden by *Tryckeriet i E-huset*, Lund.
January 2013.

To my daughter...

Sammanfattning

Dagens samhälle är allt mer beroende av elektroniska hjälpmedel såsom smarta telefoner, läsplattor, datorer osv. Prestandakraven ökar i en allt högre takt och därför måste även den bakomliggande teknologin följa samma trend. De flesta av oss vill att den nya mobiltelefonen ska ha så hög kameraupplösning som möjligt men få av oss tänker på att detta medför en större mängd data. Den nya 4G telefonen förväntas också kunna ladda upp dessa högupplösta foton och videoklipp på Internet snabbare än den gamla slitna GSM telefonen. Den ska dessutom vara lika billig i inköp. Man inser snabbt att detta ställer stora krav på den bakomliggande kommunikationsteknologin. I denna avhandling analyseras därför de praktiska utmaningarna hos en potentiell lösning.

Snabb och tillförlitlig dataöverföring tillsammans med hög bandbreddseffektivitet är viktiga designaspekter i ett modernt kommunikationssystem som t.ex. 3G ,WiFi och LTE. Bandbreddseffektivitet är i grova drag ett mått på hur mycket data ett kommunikationssystem kan överföra per tidsenhet och hertz (Hz). Idag baseras all konventionell teknologi på att de olika informationsbitarna, ettorna och nollorna, ska kunna behandlas oberoende av varandra på mottagarsidan. Denna avhandling undersöker en väsentligt annorlunda metod, nämligen att interferens mellan bitarna införs avsiktligt på sändarsidan med hjälp av den så kallade faster-than-Nyquist (FTN) tekniken. Detta medför i sin tur att bitarna stör varandra vilket resulterar i att mottagaren inte kan behandla dem var för sig. Denna signaleringsteknik introducerades redan 1975 av James Mazo, forskare på Bell Laboratories i USA, och har sedan dess utökats i många riktningar. Tidigare arbete inom området har påvisat signifikanta vinster i bandbreddseffektivitet men också påpekat att mottagaren blir alltför komplex för praktisk realisering. I denna avhandling föreslår vi ett antal lågkomplexitetslösningar för mottagning av denna typ av självstörande signaler. Vår slutsats är att de teoretiska vinsterna i bandbreddseffektivitet, som tidigare påvisats, är fullt möjliga att uppnå i praktiken då våra lågkomplexitetsmottagare har mycket god prestanda för praktiska komplexitetsnivåer.

Den första delen av avhandlingen behandlar lågkomplexitetsmottagare och relaterade stabilitetsproblem. Nya algoritmer presenteras tillsammans med en rad viktiga förbättringar vilka tillsammans radikalt reducerar mottagarens komplexitet utan att nämnvärt öka antalet felaktigt mottagna bitar.

Den andra delen analyserar effekten av de mottagarinterna beräkningarna på prestandan, kvoten mellan antalet felaktigt mottagna bitar och det totala antalet bitar, hos två i grunden olika mottagarmodeller. En av dessa två standarmodeller är välundersökt i litteraturen. Även om deras slutresultat är identiska vid optimal mottagning, så är de interna beräkningarna i allmänhet annorlunda för de två modellerna. Icke optimala mottagare, dvs. mottagare som utför ett mindre antal beräkningar, behöver därför inte generera samma slutresultat. Färre beräkningar medför i regel viktiga energibesparingar hos batteridrivna enheter samt billigare produktionskostnader. Utöver detta så föreslås och utvärderas nya typer av mottagarmodeller som arbetar emellan de två standardmodellerna.

Den sista delen av avhandlingen ägnas åt ett annat tillvägagångssätt för att reducera antalet beräkningar. Så kallade kanalkortningsmottagare optimeras ur ett icke konventionellt perspektiv. Istället för att reducera antalet beräkningar genom en förbättrad inre mottagarstruktur, så försöker en kanalkortningsmottagare att neutralisera effekterna av omgivningen (kanalen) och därefter arbeta med en förenklad kanalmodell. Ramverket som används för kanalkortning i denna avhandling är mer generell än vad som tidigare använts inom området.

Abstract

Fast and reliable data transmission together with high bandwidth efficiency are important design aspects in a modern digital communication system. Many different approaches exist but in this thesis bandwidth efficiency is obtained by increasing the data transmission rate with the faster-than-Nyquist (FTN) framework while keeping a fixed power spectral density (PSD). In FTN consecutive information carrying symbols can overlap in time and in that way introduce a controlled amount of intentional intersymbol interference (ISI). This technique was introduced already in 1975 by Mazo and has since then been extended in many directions.

Since the ISI stemming from practical FTN signaling can be of significant duration, optimum detection with traditional methods is often prohibitively complex, and alternative equalization methods with acceptable complexity-performance tradeoffs are needed. The key objective of this thesis is therefore to design reduced-complexity receivers for FTN and general linear channels that achieve optimal or near-optimal performance. Although the performance of a detector can be measured by several means, this thesis is restricted to bit error rate (BER) and mutual information results. FTN signaling is applied in two ways: As a separate uncoded narrowband communication system or in a coded scenario consisting of a convolutional encoder, interleaver and the inner ISI mechanism in serial concatenation. Turbo equalization where soft information in the form of log likelihood ratios (LLRs) is exchanged between the equalizer and the decoder is a commonly used decoding technique for coded FTN signals.

The first part of the thesis considers receivers and arising stability problems when working within the white noise constraint. New M-BCJR algorithms for turbo equalization are proposed and compared to reduced-trellis VA and BCJR benchmarks based on an offset label idea. By adding a third low-complexity M-BCJR recursion, LLR quality is improved for practical values of M. M here

measures the reduced number of BCJR computations for each data symbol. An improvement of the minimum phase conversion that sharpens the focus of the ISI model energy is proposed. When combined with a delayed and slightly mismatched receiver, the decoding allows a smaller M without significant loss in BER.

The second part analyzes the effect of the internal metric calculations on the performance of Forney- and Ungerboeck-based reduced-complexity equalizers of the M-algorithm type for both ISI and multiple-input multiple-output (MIMO) channels. Even though the final output of a full-complexity equalizer is identical for both models, the internal metric calculations are in general different. Hence, suboptimum methods need not produce the same final output. Additionally, new models working in between the two extremes are proposed and evaluated. Note that the choice of observation model does not impact the detection complexity as the underlying algorithm is unaltered.

The last part of the thesis is devoted to a different complexity reducing approach. Optimal channel shortening detectors for linear channels are optimized from an information theoretical perspective. The achievable information rates of the shortened models as well as closed form expressions for all components of the optimal detector of the class are derived. The framework used in this thesis is more general than what has been previously used within the area.

Preface

This Ph.D. thesis is based on the results of my research at the Department of Electrical and Information Technology (EIT) at Lund University. The material has partly appeared in the following journal and conference papers:

[1] A. Prlja and J.B. Anderson, "Reduced-complexity receivers for strongly narrowband intersymbol interference introduced by faster-than-Nyquist signaling," *IEEE Transactions on Communications*, vol. 60, no. 9, pp. 2591–2601, September 2012.

[2] A. Prlja, F. Rusek, M. Lončar, "A Comparison of Ungerboeck and Forney Models for Reduced-Complexity Equalization," in submission *Transactions on Emerging Telecommunications Technologies*.

[3] F. Rusek and A. Prlja, "Optimal channel shortening for MIMO and ISI channels," *IEEE Transactions on Wireless Communications*, vol. 11, no. 2, pp. 810–818, February 2012.

[4] F. Rusek, M. Lončar, A. Prlja, "A Comparison of Ungerboeck and Forney Models for Reduced-Complexity ISI Equalization," in *Proc. IEEE Global Communications Conference (GLOBECOM)*, Washington DC, December 2007.

[5] A. Prlja, J.B. Anderson, F. Rusek, "Receivers for faster-than-Nyquist signaling with and without turbo equalization," in *Proc. IEEE International Symposium on Information Theory (ISIT)*, Toronto, Canada, July 2008.

[6] J.B. Anderson, A. Prlja, F. Rusek, "New reduced state space BCJR algorithms for the ISI channel," in *Proc. IEEE International Symposium on Information Theory (ISIT)*, Seoul, South Korea, June 2009.

[7] J.B. Anderson, A. Prlja, "Turbo Equalization and an M-BCJR Algorithm for Strongly Narrowband Intersymbol Interference," in *Proc. IEEE International Symposium on Information Theory and its Applications (ISITA)*, Taichung, Taiwan, October 2010.

The following paper, which is not included in the thesis, has also been published by the author:

[8] F. Rusek, A. Prlja, D. Kapetanović, "Faster-than-Nyquist signaling based on short finite pulses," Radiovetenskaplig konferens 2008 (RVK'08), Växjö, Sweden.

This work was supported by the Swedish Research Council (VR) through Grant 621-2003-3210.

Acknowledgements

The last five years have been a great journey and an unforgettable experience for me. During my PhD studies I met and collaborated with many people who helped me to improve myself, both professionally and personally. I would like to take this opportunity to express my gratitude to all of them.

First and foremost, my utmost gratitude goes to my main supervisor, John B Anderson, for giving me this great opportunity. He believed in me and was very supportive during the whole research. Without his valuable guidance and moral help this thesis would not be possible. His enthusiasm, encouragement and motivation always helped me to proceed and take the next step. I would also like to thank my second supervisor, Fredrik Rusek, for sharing his knowledge and experience with me. His ability to present problems in a simple and interesting ways inspired me many times. His deep technical insight, attention for details and excellent ideas helped me and facilitated during difficult times. Writing several papers together and having long discussions about different problems was very invaluable for me.

Besides my supervisors I would like to thank all my friends who were very supportive during this period. Spending time with them after working hours helped me to relax and gather new energy to continue with my studies. I would also like to thank all PhD students and employees at the EIT department who made my working environment nice and friendly. It was a pleasure working with such wonderful people.

Last but not least, I would like to thank my family, my mom, dad and sister for their infinite support and encouragement. I am very grateful to them for leading me onto the right path and teaching me many good things about life. Without their love and caring all this would not be possible. Finally, I would like to thank my wife, Nidzara, who always has the right word of advice and knows how to make me laugh. I appreciate her patience and support when I needed it the most. This thesis is dedicated to my one year old daughter, Lamija, who is my biggest inspiration in life and always puts a smile on my face. She truly makes my world a better place.

Adnan Prlja

Contents

Chapter 1

Introduction

A tremendous progress in communication technology has been made in the last two decades. For example mobile telephony, which was primarily meant for voice-based communication, has evolved rapidly and today non-voice services are overtaking voice-based communications. Advances in computer networking on the other hand laid the foundation for the largest global medium for information exchange, the Internet. As a result, there is an ever increasing need for improving the existing technologies which more efficiently can exploit the available resources.

Modulation theory has, since the pioneering work of Nyquist [1], been mainly based on the concept of memoryless transmission which greatly simplifies the receiver design and the theoretical analysis. The symbols in different time intervals were independent and they were transmitted in a fashion such that there was no intersymbol interference (ISI) present at the receiver. In 1948 and 1949 Shannon [2, 3] brought a radical change to communications with the development of information theory. He discovered that highly reliable communication is possible if the symbols are encoded in groups. He also proved that such a construction is possible if the time signals are generated using sinc pulses. Most communication technologies therefore maintained the memoryless assumption in the modulation part (see Figure 2.2). Although this assumption can be made optimal in theory, in practice it can lead to significant capacity penalties due to non-ideal components. Therefore, in this thesis, ISI is intentionally introduced using the concept of Faster-than-Nyquist (FTN) signaling. FTN provides improved spectral efficiency that cannot be reached by communication systems based on orthogonal (Nyquist) signaling. Section 2.5 of this thesis reviews the FTN signaling concept.

Decoding of signals using trellises or trees plays a crucial role in digital communications. In fact, many signal detection problems in wireless communications can be approached with trellis and tree decoding techniques. One such example is the detection of FTN signals. In FTN intersymbol interference is introduced by transmitting signals at a higher signaling rate than allowed by the Nyquist orthogonality criterion. Each received signal can in the presence of ISI be represented as a function of the most recent input symbol and the past L input symbols, where L is the length or memory of the ISI sequence. This signal structure can be modeled by a finite state machine (FSM) process and consequently be described using a trellis [4]. MIMO and frequency selective communication channels are other examples where tree and trellis detection can be employed.

A well-known algorithm that operates on a trellis is the Viterbi algorithm (VA) developed in 1967 [5]. Due to its high computational complexity for long ISI responses and large constellation sizes, the Viterbi algorithm is often impractical. Instead, the M-algorithm by Anderson in 1969 [6] can be used. It explores only a part of the tree/trellis and in that way the overall computational effort is reduced. For a description of the M-algorithm, see Section 2.3.

The invention of turbo codes [7, 8] was a major step forward in communications. The turbo processing principle developed by Hagenauer [9] has been applied to concatenated communication systems in order to improve their overall system performance. This iterative exchange of soft information between two soft-input soft-output component decoders will be frequently used in this thesis. Turbo equalization [10], also known as iterative equalization and decoding, is one such application. Instead of using conventional hard-output component decoders, turbo equalization uses a soft-input soft-output ISI equalizer and a soft-input soft-output outer decoder which produce and exchange log likelihood ratios. In this way the turbo equalizer can approach the optimal performance, the performance of joint equalization and decoding, with practical complexity levels. More details about the basic principles of turbo equalization are given in Section 2.7.

Some well-known soft-output algorithms are the Bahl-Cocke-Jelinek-Raviv (BCJR) algorithm [11], also known as the maximum a posteriori (MAP) decoder and the soft-output Viterbi algorithm (SOVA) [12]. Both algorithms require a realization of the complete trellis and are therefore often not practical to implement. In this thesis, reduced-complexity trellis and tree based soft-input soft-output algorithms for FTN signaling and general linear channels are proposed. The objective is to design low-complexity receivers which can produce reliable log likelihood ratios.

A reduced-complexity soft-output algorithm for iterative detection is the well-known M-BCJR algorithm proposed by Franz and Anderson [13]. By using the M-algorithm, they reduced the complexity of the BCJR algorithm. A similar algorithm called the T-BCJR algorithm was also proposed in [13]. In Chapter 3 of this thesis a new improved M-BCJR algorithm is proposed. Another soft-input soft-output algorithm is the soft-output M-algorithm (SOMA) [14] which is a reduced complexity variant of SOVA. A soft-output sequential decoder known as the LIST-sequential (LISS) decoder was proposed by Kuhn and Hagenauer in [15]. A popular reduced-complexity technique for turbo detection of MIMO channels is the sphere decoder proposed by Hochwald and ten Brink [16]. Note that the impressive amount of literature on reduced-complexity techniques prevents a full treatment in this thesis.

In Chapter 4 the performance of reduced-complexity algorithms based on two different discrete-time observation models is studied. Even though the final output of a full-complexity detector is identical for both models, the internal metric calculations are different and hence reduced-complexity methods based on the two models need not produce the same final output. Chapter 5 considers channel shortening detectors for linear channels, optimized from an information theoretical perspective. Chapter 6 summarizes the thesis and gives a discussion on possible future work.

Chapter 2

Basic Principles of Linear Modulation

In this chapter, an introduction to time-continuous linear modulation systems is given. Mathematical models for the linear modulation system and the communication channel are formulated and some basic detection algorithms, giving rise to equivalent discrete-time models of the communication system, are presented. Note that this chapter only introduces results required for the remaining part of the thesis. For more details and complete derivations, the reader is referred to [17, 18, 19, 20].

2.1 Single Carrier Linear Modulation

The signal transmission method considered in this thesis is the simple and practical linear modulation whose baseband form can expressed as

$$s(t) = s_{\boldsymbol{a}}(t) \stackrel{\triangle}{=} \sum_{k=0}^{\infty} a_k h(t - kT) \tag{2.1}$$

where $\boldsymbol{a} = \{a_0, a_1, a_2, \ldots\}$ is the information carrying symbol sequence (possibly complex-valued) and $h(t)$ is a real-valued continuous modulation pulse. In order to satisfy frequency assignment requirements on the system, the spectrum of (2.1) is modulated to a carrier frequency f_c before transmission, yielding the radio frequency (RF) representation

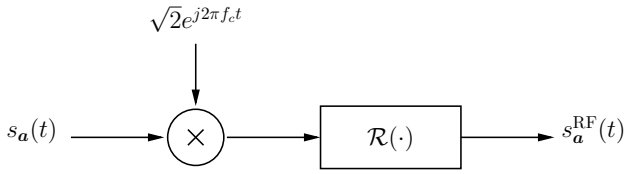

Figure 2.1: A simple device for information transmission via carrier modulation.

$$s_{\boldsymbol{a}}^{\mathrm{RF}}(t) = \sqrt{2}\mathcal{R}\{s_{\boldsymbol{a}}(t)e^{j2\pi f_c t}\}. \tag{2.2}$$

Here $\mathcal{R}\{\cdot\}$ denotes the real part of a complex number, f_c is the carrier frequency in Hz and the superscript "RF" denotes a modulated signal. Note that the baseband signal (2.2) has its frequency support concentrated around $f = 0$. A simple model of the device that generates the bandpass signal $s_{\boldsymbol{a}}^{\mathrm{RF}}(t)$ from the baseband signal $s_{\boldsymbol{a}}(t)$ is shown in Figure 2.1. It is also assumed that the signal $s(t)$ is bandlimited to W positive Hz, where W is referred to as the bandwidth of $s(t)$. This bandlimitation is achieved by bandlimiting the modulation pulse $h(t)$ to W Hz. Additionally, in order to avoid frequency overlaps in the transmitted signal, it is assumed that $W \ll f_c$.

Now consider the communication system in Figure 2.2. The binary information sequence $\boldsymbol{u}$ is encoded by an encoder with code rate R_c producing the binary sequence $\boldsymbol{v}$. The length of $\boldsymbol{v}$ is given by the length of $\boldsymbol{u}$ divided by R_c. The encoding introduces a structured dependence among the encoded bits which in general improves the communication performance. The mapper in Figure 2.2 now maps the binary codeword $\boldsymbol{v}$ onto the sequence $\boldsymbol{a}$ consisting of symbols from the symbol alphabet Ω. A modulator uses the sequence $\boldsymbol{a}$ as input in order to produce a sequence of analog signal waveforms $s_{\boldsymbol{a}}(t)$ to be transmitted. Finally, an additive white Gaussian noise (AWGN) channel with noise $n(t)$ follows resulting in the received signal $r(t)$, i.e., $r(t) = s_{\boldsymbol{a}}(t) + n(t)$. The symbols $\{a_k\}$ in this thesis do not need to be independent. However, it is assumed that the encoder/mapper combinations are such that they generate uncorrelated output streams which can be expressed as

$$\mathbb{E}[a_k a_m^*] = \sigma_a^2 \delta[k-m] \tag{2.3}$$

where $\mathbb{E}$ denotes the expectation operator, $*$ denotes complex conjugation and $\delta[\cdot]$ is the Kronecker delta function. These notations will be used throughout

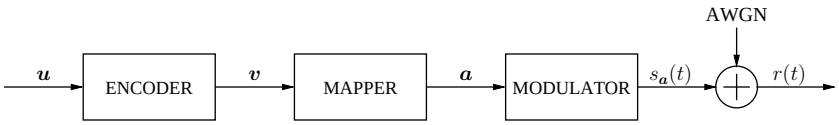

Figure 2.2: A system model of a communications system in additive white Gaussian noise (AWGN). After encoding and mapping, the information carrying signal is formed by (2.1).

the thesis. Another requirement is that the data symbols at their output are equiprobable, i.e.,

$$\Pr(a_k = a') = \frac{1}{|\Omega|}. \quad a' \in \Omega \tag{2.4}$$

In (2.4) $\Pr(\cdot)$ denotes a probability mass function (PMF) while a probability density function (PDF) will be denoted $p(\cdot)$ throughout. In this thesis convolutional and low-density parity-check codes (LDPC) are used for encoding. The symbol alphabet Ω is assumed to be time-invariant. It is also a balanced one, that is

$$\sum_{a_k \in \Omega} a_k = 0.$$

2.1.1 Bit and Block Error Rate

Since in general the received signal $r(t)$ is distorted and noisy, the receiver will in a random manner produce erroneous decisions. In order to quantify this as a communication performance measure we next define *the average number of information bit errors per detected information bit*, the bit error rate P_b (alternatively BER, bit error ratio or bit error probability). The uncoded sequence $\boldsymbol{u}$ consisting of N information bits is to be communicated across a linear channel. As in Figure 2.2 these bits are in general encoded which produces a longer sequence of bits, $\boldsymbol{v}$. After mapping onto the discrete alphabet Ω the symbol sequence $\boldsymbol{a}$ is sent across the channel as analog waveforms. After filtering and sampling the receiver observes the sequence $\boldsymbol{y}$ and produces an estimate $\hat{\boldsymbol{u}}$ of the information sequence $\boldsymbol{u}$. This is usually made in two stages, demodulation and decoding. Note that demapping is often included in the demodulation process. The demodulation stage converts the received analog signal into a sequence of information carrying numbers containing both distortion and noise.

This sequence is then fed to the decoder's input which, by following a certain decoding rule, produces the estimate $\hat{\boldsymbol{u}}$ of the binary uncoded sequence $\boldsymbol{u}$.

Consider now the sequence $\boldsymbol{u}$ and let u_k denote its kth information bit and let $P_k \triangleq \Pr(\hat{u}_k \neq u_k)$ be its error probability. The bit error rate P_b can now be defined as

$$P_b \triangleq \frac{\sum_{k=0}^{N-1} P_k}{N}. \tag{2.5}$$

In the same manner we can define the block error rate (BLER), denoted P_b^N, as *the probability that the receiver outputs an incorrect sequence* $\hat{\boldsymbol{u}}$, that is

$$P_b^N \triangleq \Pr(\hat{\boldsymbol{u}} \neq \boldsymbol{u}). \tag{2.6}$$

Even though these quantities should be as small as possible, in many cases and due to different constraints one needs to specify a desired value. Typical desired values of P_b are in the range $10^{-2} - 10^{-9}$ depending on the application. From (2.6) we have that

$$P_b^N = \Pr\left(\bigcup_{0\leq k\leq N-1} \{\hat{u}_k \neq u_k\}\right) \tag{2.7}$$

and by using the union bound, we obtain the following upper limit

$$P_b^N = \Pr\left(\bigcup_{0\leq k\leq N-1} \{\hat{u}_k \neq u_k\}\right) \leq \sum_{k=0}^{N-1} \Pr\left(\hat{u}_k \neq u_k\right) = NP_b. \tag{2.8}$$

Moreover, we have that

$$P_b^N \geq \Pr\left(\hat{u}_k \neq u_k\right), \quad k = 0, \ldots, N-1.$$

We can now write

$$\sum_{k=0}^{N-1} P_b^N \geq \sum_{k=0}^{N-1} P_k$$

which gives

$$P_b^N \geq P_b. \tag{2.9}$$

Finally P_b^N can be bounded by combining (2.8) and (2.9):

$$P_b \leq P_b^N \leq NP_b. \tag{2.10}$$

2.1.2 Bandwidth Properties

The power spectral density (PSD), denoted $\Phi_{s_{\boldsymbol{a}}}(f)$, of the wide-sense cyclostationary process $s_{\boldsymbol{a}}(t)$ is a function describing the distribution of the power as a function of frequency. It is given by the Fourier transform of the autocorrelation of $s_{\boldsymbol{a}}(t)$. In order to proceed with the bandwidth (and Euclidean distance) calculations we introduce the autocorrelation function of $h(t)$, denoted $\lambda(t)$. It is defined as

$$\lambda(t) \triangleq \int_{-\infty}^{\infty} h(\tau)h^*(\tau - t)\,\mathrm{d}\tau \tag{2.11}$$

or alternatively

$$\lambda(t) = h(t) \star h^*(-t), \tag{2.12}$$

where $\star$ is the convolution operator. From (2.11) it follows that the modulation pulse energy, denoted E_p, is given by

$$E_p \triangleq \int_{-\infty}^{\infty} |h(t)|^2\,\mathrm{d}t = \lambda(0). \tag{2.13}$$

The autocorrelation of $s_{\boldsymbol{a}}(t)$ is

$$\begin{aligned}
\phi_{s_{\boldsymbol{a}}}(\tau + t, t) &\triangleq \mathbb{E}[s_{\boldsymbol{a}}(\tau + t)s_{\boldsymbol{a}}^*(t)] \\
&= \sum_{j=0}^{\infty}\sum_{k=0}^{\infty} h(\tau + t - jT)h^*(t - kT)\mathbb{E}[a_j a_k^*] \\
&= \sigma_a^2 \sum_{k=0}^{\infty} h(\tau + t - kT)h^*(t - kT).
\end{aligned} \tag{2.14}$$

Since $s_{\boldsymbol{a}}(t)$ is a wide-sense cyclostationary process with period T, its time-average autocorrelation function is

$$\begin{aligned}\bar{\phi}_{s_{\boldsymbol{a}}}(\tau) &\triangleq \frac{1}{T}\int_0^T \phi_{s_{\boldsymbol{a}}}(\tau+t,t)\,\mathrm{d}t \\ &= \frac{\sigma_a^2}{T}\int_0^T \sum_{k=0}^{\infty} h(\tau+t-kT)h^*(t-kT)\,\mathrm{d}t \\ &= \frac{\sigma_a^2}{T}\int_{-\infty}^{\infty} h(\tau+t)h^*(t)\,\mathrm{d}t \\ &= \frac{\sigma_a^2}{T}\lambda(t). \end{aligned} \tag{2.15}$$

By now taking the Fourier transform of (2.15), we obtain the power spectral density of $s_{\boldsymbol{a}}(t)$:

$$\begin{aligned}\Phi_{s_{\boldsymbol{a}}}(f) &\triangleq \mathcal{F}\{\bar{\phi}_{s_{\boldsymbol{a}}}(\tau)\} \\ &= \frac{\sigma_a^2}{T}\Lambda(f), \quad |f|<W \end{aligned} \tag{2.16}$$

where, according to (2.12), $\Lambda(f)=\mathcal{F}\{\lambda(t)\}$ is given by

$$\Lambda(f) = H(f)H^*(f) = |H(f)|^2. \tag{2.17}$$

Combining (2.16) and (2.17) results in

$$\Phi_{s_{\boldsymbol{a}}}(f) = \frac{\sigma_a^2}{T}|H(f)|^2. \tag{2.18}$$

Note that $|H(f)|^2$ is symmetric around f = 0 since the modulation pulse $h(t)$ is real-valued. The bandwidth is in this thesis defined as the smallest single scalar number W such that

$$\Phi_{s_{\boldsymbol{a}}}(f) = 0, \quad |f|>W \tag{2.19}$$

This is shown schematically in Figure 2.3. From the average power P of $s_{\boldsymbol{a}}(t)$ given by

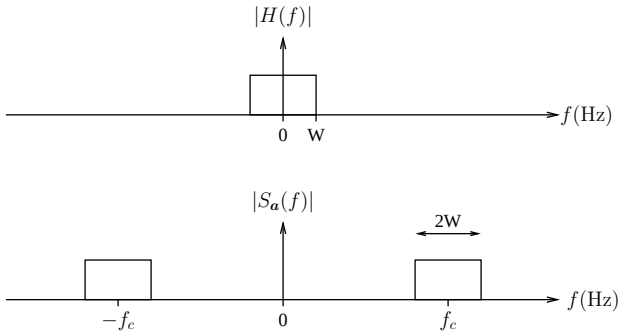

Figure 2.3: An example of the frequency content in the bandpass signal $s_{\boldsymbol{a}}^{\mathrm{RF}}(t)$.

$$P = \bar{\phi}_{s_{\boldsymbol{a}}}(0) = \frac{\sigma_a^2}{T} E_p \tag{2.20}$$

we can now obtain the average symbol energy E_s according to

$$E_s \triangleq T\bar{\phi}_{s_{\boldsymbol{a}}}(0) = \sigma_a^2 \lambda(0) = \sigma_a^2 E_p. \tag{2.21}$$

The average energy per information bit is given by

$$E_b \triangleq \frac{E_s}{R_c \log_2 |\Omega|} = \frac{\sigma_a^2 E_p}{R_c \log_2 |\Omega|}. \tag{2.22}$$

If not otherwise stated, in this thesis it is assumed that $h(t)$ is unit energy, i.e.,

$$E_p = \int_{-\infty}^{\infty} |h(t)|^2 \, \mathrm{d}t = 1$$

so that (2.21) and (2.22) become

$$\begin{aligned} E_s &= \sigma_a^2 \\ E_b &= \frac{\sigma_a^2}{R_c \log_2 |\Omega|}. \end{aligned}$$

Finally we introduce the normalized bandwidth W_{norm} so that different communication setups can compared. It is defined as the ratio between *the total*

consumed bandwidth and the total information bit rate. If N_{dim} denotes the number of dimensions spanned by (2.1), the normalized bandwidth becomes

$$W_{\text{norm}} \triangleq \frac{N_{\text{dim}} WT}{R_c \log_2 |\Omega|}, \quad \text{Hz} - \text{s/bit}. \tag{2.23}$$

A real $s_{\boldsymbol{a}}(t)$ gives $N_{\text{dim}} = 1$ while $N_{\text{dim}} = 2$ for complex $s_{\boldsymbol{a}}(t)$. The physical data bits carried by a communication system is the product of its bandwidth W and its time T, divided by W_{norm}. For example, a 1 MHz width system working for 2 seconds carries $2 \times 10^6 / W_{\text{norm}}$ bits.

2.1.3 Frequency Selective and Flat Channels

Intersymbol interference (ISI) is in this thesis introduced either by a frequency selective communication channel or by filtering and pulse shaping at the transmitter. Section 2.5 considers an example of the last, called faster-than-Nyquist (FTN) signaling where the signals are intentionally allowed to overlap in the time-domain. If instead the radio frequency modulated signal $s_{\boldsymbol{a}}^{\text{RF}}(t)$ from (2.2) is exposed to a multipath environment, represented by its real-valued impulse response $c^{\text{RF}}(t)$, the received signal $r_{\boldsymbol{a}}^{\text{RF}}(t)$ equals [17]

$$\begin{aligned} r_{\boldsymbol{a}}^{\text{RF}}(t) &= \int_{-\infty}^{\infty} c^{\text{RF}}(\tau) s_{\boldsymbol{a}}^{\text{RF}}(t-\tau)\, \mathrm{d}\tau + n^{\text{RF}}(t) \\ &= \mathcal{R}\left\{ \left(\int_{-\infty}^{\infty} c^{\text{RF}}(\tau) e^{-j2\pi f_c \tau} s_{\boldsymbol{a}}^{\text{RF}}(t-\tau)\, \mathrm{d}\tau \right) e^{j2\pi f_c t} \right\} + n^{\text{RF}}(t) \end{aligned} \tag{2.24}$$

where $n^{\text{RF}}(t)$ is additive white Gaussian noise with mean $\mathbb{E}[n^{\text{RF}}(t)] = 0$ and autocorrelation $\mathbb{E}[n^{\text{RF}}(t) n^{\text{RF}}(t+\tau)] = N_0 \delta(\tau)$. The noise $n^{\text{RF}}(t)$ can be expressed as

$$n^{\text{RF}}(t) = \sqrt{2}\mathcal{R}\{n(t) e^{j2\pi f_c t}\} \tag{2.25}$$

where $n(t)$ is complex-valued AWGN with mean $\mathbb{E}[n(t)] = 0$ and autocorrelation $\mathbb{E}[n(t) n^*(t+\tau)] = N_0 \delta(\tau)$. By further defining

$$c(\tau) \triangleq c^{\text{RF}}(\tau) e^{-j2\pi f_c \tau} \tag{2.26}$$

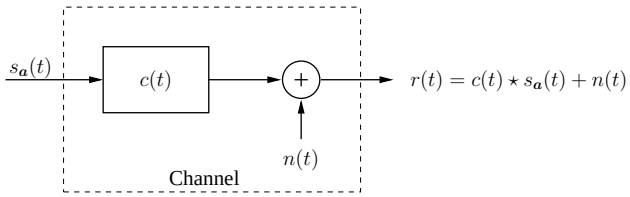

Figure 2.4: A baseband model of a communications system exposed to a multipath environment.

we note that the integral in (2.24) represents convolution of $s_{\boldsymbol{a}}(t)$ with a complex baseband channel impulse response (CIR) $c(\tau)$. A complex-valued baseband model of (2.24) equals

$$\begin{aligned} r(t) &= c(t) \star s_{\boldsymbol{a}}(t) + n(t) \\ &= \sum_{k=0}^{\infty} a_k \Big(c(t) \star h(t - kT) \Big) + n(t) \\ &= \sum_{k=0}^{\infty} a_k b(t - kT) + n(t) \end{aligned} \tag{2.27}$$

where

$$b(t) \stackrel{\triangle}{=} c(t) \star h(t). \tag{2.28}$$

Since the spectrum of $s_{\boldsymbol{a}}(t)$ is changed by the channel (the Fourier transform of $c(t) \star s_{\boldsymbol{a}}(t)$ is $C(f)S_{\boldsymbol{a}}(f)$ where $C(f) = \mathcal{F}\{c(t)\}$ and $S_{\boldsymbol{a}}(f) = \mathcal{F}\{s_{\boldsymbol{a}}(t)\}$), $c(t)$ represents a *frequency selective* channel. Figure 2.4 shows a simple baseband model of the multipath environment channel $c(t)$ with additive noise. A *non-frequency selective* channel, also known as a *flat* channel is obtained if $c(t) = \delta(t)$, i.e., there is no multipath in the environment. Note that, in this thesis, it is always assumed that the receiver has perfect knowledge of the channel $c(t)$.

2.1.4 The Squared Euclidean Distance

In order to detect data reliably it is relevant to investigate how different two analog signals, corresponding to data sequences $\boldsymbol{a}_0$ $\boldsymbol{a}_1$, are. It is easier for the receiver to distinguish the two signals if the difference is large which eventually

leads to a smaller probability of error in optimal detection. One important measure of the difference, closely related to the bit error rate, is the *squared Euclidean distance* defined as

$$\begin{aligned} D^2(\boldsymbol{a}_0, \boldsymbol{a}_1) &\triangleq \int_{-\infty}^{\infty} |s_{\boldsymbol{a}_0}(t) - s_{\boldsymbol{a}_1}(t)|^2 \, \mathrm{d}t \\ &= \int_{-\infty}^{\infty} \left| \sum_{k=0}^{\infty} (a_{0,k} - a_{1,k}) h(t - kT) \right|^2 \mathrm{d}t \\ &= \int_{-\infty}^{\infty} \left| \sum_{k=0}^{\infty} e_k h(t - kT) \right|^2 \mathrm{d}t \\ &= \sum_{j=0}^{\infty} \sum_{k=0}^{\infty} e_j e_k^* \int_{-\infty}^{\infty} h(t - kT) h^*(t - jT) \, \mathrm{d}t \end{aligned} \tag{2.29}$$

where $\boldsymbol{e}$ is an error event defined as $\boldsymbol{e} \triangleq \boldsymbol{a}_0 - \boldsymbol{a}_1$ and where the notation $a_{0,k}$ denotes the kth symbol in the sequence $\boldsymbol{a}_0$. Since (2.29) only depends on the difference $\boldsymbol{a}_0 - \boldsymbol{a}_1$ we can define

$$D^2(\boldsymbol{e}) \triangleq D^2(\boldsymbol{a}_0, \boldsymbol{a}_1) \tag{2.30}$$

where the error symbols e in an error event belong to an error symbol alphabet $\mathcal{E}$. In the simple case $\Omega = \{+1, -1\}$ we have that $\mathcal{E} = \{+2, 0, -2\}$. An alternative expression of the the squared Euclidean distance is obtained if (2.11) is substituted into (2.29):

$$\begin{aligned} D^2(\boldsymbol{e}) &= \sum_{j=0}^{\infty} \sum_{k=0}^{\infty} e_j \lambda((j-k)T) e_k^* \\ &= \sum_{j=0}^{\infty} \sum_{k=0}^{\infty} e_j g_{j-k} e_k^* \end{aligned} \tag{2.31}$$

where g_k is the baud rate (signaling rate) sampled autocorrelation function of $h(t)$, i.e.,

$$g_k \triangleq \lambda(kT). \tag{2.32}$$

The sampled matched filter outputs, which define the so-called Ungerboeck observation model in the following chapters, depend directly on the samples g_k. Therefore the same notation as in (2.32) will be used throughout the thesis.

Since $D^2(\boldsymbol{e})$ in (2.31) depends on the average symbol energy E_s it is not possible to compare communication systems with different pulse shapes and symbol alphabets if the value of E_s varies. Therefore a more appropriate measure is the *normalized squared Euclidean distance* defined as

$$d^2(\boldsymbol{e}) \triangleq \frac{D^2(\boldsymbol{e})}{2E_b} = \frac{R_c \log_2 |\Omega|}{2\sigma_a^2} D^2(\boldsymbol{e}) \tag{2.33}$$

The asymptotic error probability of any linear signaling, neglecting multiplicities, depends strongly on the *normalized minimum squared Euclidean distance* [20] defined as

$$d^2_{\min} \triangleq \min_{\boldsymbol{e}\neq 0}\{d^2(\boldsymbol{e})\}. \tag{2.34}$$

This important measure is found by performing a minimization over all error events allowed by the outer code. In the uncoded case the minimization in (2.34) is instead performed over all possible error events. Henceforth, $d^2_{\min}$ will be referred to as minimum distance. Note that $d^2_{\min}$ in (2.34) depends on $h(t)$.

2.1.5 T-Orthogonal Pulses

A pulse $h(t)$ is said to be T-orthogonal (or orthogonal under T-shifts) if it satisfies

$$\int_{-\infty}^{\infty} h(t)h(t-kT)\,\mathrm{d}t = 0, \quad k = \pm 1, \pm 2, \ldots, \tag{2.35}$$

where T, here and throughout, is the symbol interval. Note that this implies that the sampled autocorrelation function g_k in (2.32) is $g_k = \delta[k]$. Since a T-orthogonal pulse is uncorrelated with a shift of itself by any multiple of T, it is possible to find any symbol a_k in the noise-free signal $s_{\boldsymbol{a}}(t)$ by performing the correlation integral

$$\int_{-\infty}^{\infty} s_{\boldsymbol{a}}(t)h(t-kT)\,\mathrm{d}t = \int_{-\infty}^{\infty}\left(\sum_{j=0}^{\infty} a_j h(t-jT)\right)h(t-kT)\,\mathrm{d}t = a_k \int_{-\infty}^{\infty} |h(t-kT)|^2\,\mathrm{d}t. \tag{2.36}$$

If in (2.13) $E_p = 1$, the right-hand side of (2.36) equals a_k. In fact, if the signal $s_{\boldsymbol{a}}(t)$ is fed to a matched filter (MF), i.e., a filter with transfer function $H^*(f)$, and followed by sampling each T seconds, the whole sequence $\boldsymbol{a}$ is obtained.

If $g_k \neq \delta[k]$, intersymbol interference (ISI) is present. The memory of the ISI response $\{g_k\}$, given by the smallest L such that

$$g_k = 0, \quad |k| > 0 \tag{2.37}$$

determines the complexity of a tree/trellis based detection algorithm. An ISI channel of memory L and a modulation alphabet Ω is represented by a size-$|\Omega|^L$ trellis. Since a more challenging receiver design is undesirable a first objective is to choose a pulse $h(t)$ that fulfills $g_k = \delta[k]$ with as small bandwidth as possible. There is however a tradeoff between the two; reducing the bandwidth will in general increase the length of the ISI response.

The narrowest bandwidth of any T-orthogonal pulse is $1/2T$ Hz and the corresponding pulse is the sinc pulse:

$$h_{\text{sinc}}(t) = \frac{\sin(\pi t/T)}{\pi t/T}. \tag{2.38}$$

Its Fourier transform is in fact a square pulse, i.e.,

$$H_{\text{sinc}}(f) = \begin{cases} \sqrt{T}, & |f| \leq 1/2T \\ 0, & |f| > 1/2T. \end{cases} \tag{2.39}$$

In order to reduce the amplitude variations in the signal $s_{\boldsymbol{a}}(t)$ and the temporal tails of the sinc pulse, a common class of T-orthogonal pulses with a smoother spectra, the root raised cosine (root RC) class, can be used instead. A pulse from this class is defined by its Fourier transform which satisfies

$$|H(f)|^2 = \begin{cases} T, & |f| \leq (1-\beta)/2T \\ T\cos^2\left(\frac{\pi T}{2\beta}\left(|f| - \frac{1-\beta}{2T}\right)\right), & (1-\beta)/2T < |f| \leq (1+\beta)/2T \\ 0, & |f| > (1+\beta)/2T. \end{cases} \tag{2.40}$$

The extra bandwidth is defined through the parameter β, $0 \leq \beta \leq 1$, also known as the "rolloff" or excess bandwidth factor. A root RC pulse is bandlimited to $(1+\beta)/2T$, i.e., its bandwidth is a fraction β greater than the

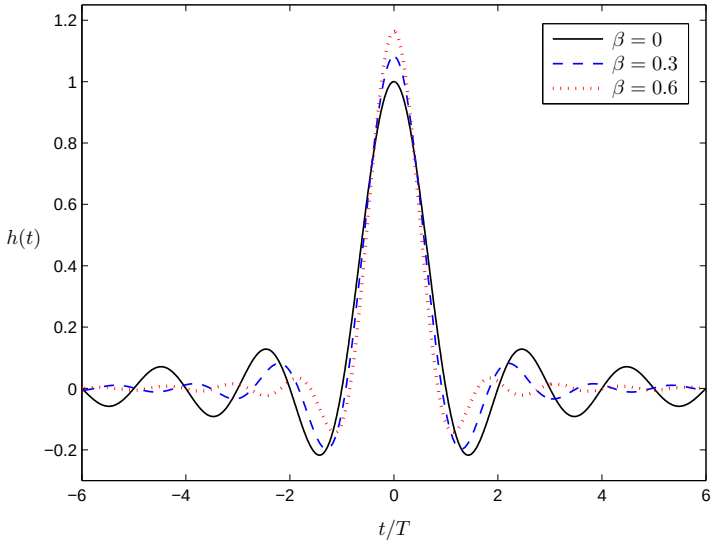

Figure 2.5: Root RC pulses with three different excess bandwidths β.

bandwidth of a sinc. By setting $\beta = 0$ we in fact obtain a sinc pulse. Nyquist showed [1] that a sufficient condition for T-orthogonality is that the Fourier transform of the pulse is antisymmetric around the point $f = 1/2T$. This condition is satisfied by the root RC family whose time domain expression is

$$h(t) = \begin{cases} \frac{1}{\sqrt{T}} \frac{\sin(\pi(1-\beta)t/T) + (2\beta t/T)\cos(\pi(1+\beta)t/T)}{(\pi t/T)(1-(4\beta t/T)^2)}, & t \neq 0, \pm\frac{T}{4\beta} \\ \frac{1}{\sqrt{T}}\left(1-\beta+\frac{4\beta}{\pi}\right), & t = 0 \\ \frac{\beta}{\sqrt{2T}}\left(1+\frac{2}{\pi}\right)\sin\left(\frac{\pi}{4\beta}\right) + \left(1-\frac{2}{\pi}\right)\cos\left(\frac{\pi}{4\beta}\right), & t = \pm\frac{\pi}{4\beta}. \end{cases} \quad (2.41)$$

Figure 2.5 shows the time domain representation of $h(t)$ as root RC pulses with three different excess bandwidths β, $\beta = 0, 0.3, 0.6$. The solid curve corresponds to $\beta = 0$, i.e., the sinc pulse. It is clear that a larger excess bandwidth results in a narrower pulse with much smaller amplitude oscillations. Since root RC pulses have infinite time support, they must be truncated in practice. It is however important to assure that the truncation is not made too early which could improve the receiver error rate and give a false test result.

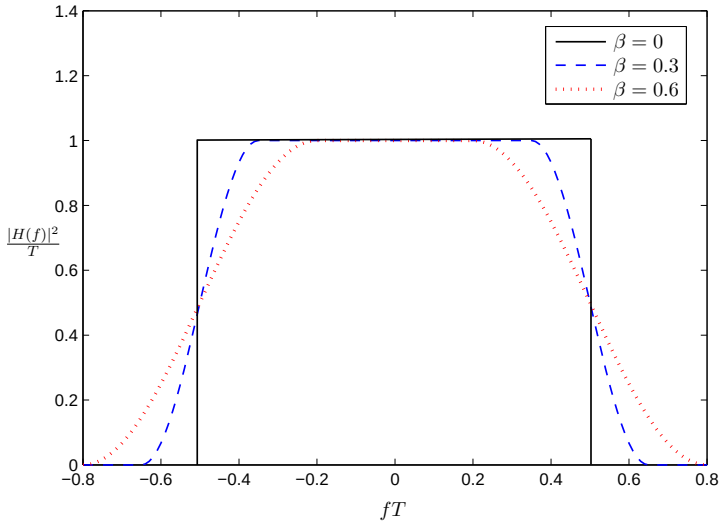

Figure 2.6: Squared Fourier transforms of root RC pulses with different β.

In Figure 2.6 squared Fourier transforms of root RC pulses with the same three values of β as in Figure 2.5 are shown. Even though a larger β simplifies the implementation of a linear modulation system (by increasing the bandwidth), it does not change the error performance of communication systems signaling at the rate $1/T$, for frequency flat channels.

2.1.6 Symbol Alphabets

In this thesis three different symbol alphabets Ω are considered. If the information is placed only in the amplitude the modulation method is called pulse amplitude modulation (PAM). Even though other amplitude alternatives are possible, the conceptually simplest choice is to let the symbols a_k in (2.1) be real and taken from the balanced and equispaced M-PAM (M-ary PAM) alphabet defined as

$$\Omega_{M-\mathrm{PAM}} = \{-(M-1), -(M-3), \ldots, (M-3), (M-1)\}$$

where $|\Omega| = M$ and M is usually a power of two, i.e., $M = 2^k$ where k is an integer. The corresponding error symbol alphabet $\mathcal{E}$ is given by

$$\mathcal{E}_{M-\mathrm{PAM}} = \{-2(M-1), -2(M-2), \ldots, 2(M-2), 2(M-1)\}.$$

Consider now the normalized minimum Euclidean distance of an M-PAM alphabet in orthogonal signaling with a T-orthogonal modulation pulse and a flat channel. The error event that results in the smallest Euclidean distance is in fact an error event consisting of the single error symbol $e_k = 2$ for all possible M. It is easy to show that the corresponding normalized minimum Euclidean distance, often referred to as the matched filter bound in the literature, equals

$$d^2_{\min} = d^2_{\mathrm{MF}} \triangleq \frac{6\log_2(M)}{M^2-1}. \tag{2.42}$$

The normalized bandwidth W_{norm} is derived next. Since the M-PAM alphabet is real-valued, $N_{\dim}$ in (2.23) is $N_{\dim} = 1$. If further a T-orthogonal root RC pulse with excess bandwidth β is assumed, the normalized bandwidth of an uncoded system ($R_c = 1$) is given by

$$W_{\mathrm{norm}} = \frac{1+\beta}{2\log_2(M)}.$$

In case of a complex-valued Ω, there are two standard alphabets: phase shift keying (PSK) and quadrature amplitude modulation (QAM). In contrast to PAM, the information in PSK is placed only in the phase. The M-PSK alphabet is defined as

$$\Omega_{M-\mathrm{PSK}} = \{e^{j2\pi n/M}, \quad 0 \le n \le M-1\}$$

where M is again a power of two, i.e., $M = 2^k$, k an integer and j is the imaginary unit throughout the thesis. Note that all symbols within an M-PSK alphabet have equal energy. This property is important for efficient hardware implementation since the transmitted signal has smaller amplitude variations. The normalized minimum Euclidean distance of an M-PSK alphabet in orthogonal signaling equals

$$d^2_{\min} = 2\log_2(M)\sin^2(\pi/M). \tag{2.43}$$

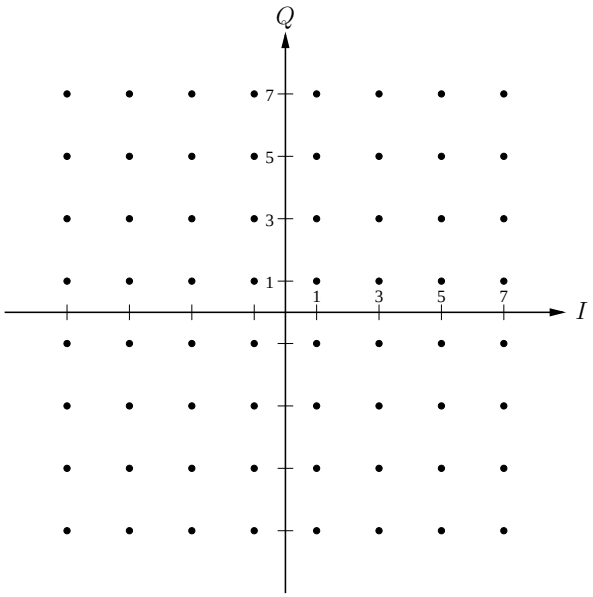

Figure 2.7: An I/Q diagram of a 64-QAM constellation.

In QAM the information to be transmitted is placed as amplitude values on two orthogonal signals, often referred to as the *in-phase* (I) component and *quadrature* (Q) component. The M-QAM alphabet is defined as

$$\Omega_{M-\mathrm{QAM}} = \{A + jB \quad A, B \in \Omega_{\sqrt{M}-PAM}\}$$

where it is assumed that $M = 2^{2k}$, k an integer. The definition of M implies that some values of M cannot be reached, i.e., there exists for example no standard 32-QAM. However for $M = 2^{2k+1}$, k an integer, so-called cross-constellations can be used instead [20]. An example of an I/Q diagram for a 64-QAM constellation is shown in Figure 2.7. Note that 4-QAM and 4-PSK are identical except for a rotation. In this thesis they are both referred to as QPSK (quadrature PSK).

The normalized minimum Euclidean distance of an M-QAM alphabet in orthogonal signaling is given by

$$d^2_{\min} = \frac{3\log_2(M)}{M-1}. \tag{2.44}$$

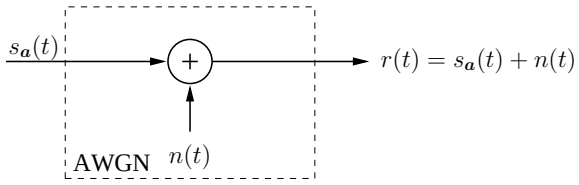

Figure 2.8: A model of the AWGN channel.

By comparing (2.42) with (2.44), we realize that $d^2_{\min}$ of a $\sqrt{M}$-PAM alphabet equals that of an M-QAM alphabet. The normalized bandwidth for both M-QAM and M-PSK in an uncoded system with $h(t)$ taken as a root RC pulse with excess bandwidth β is

$$W_{\text{norm}} = \frac{2(1+\beta)}{2\log_2(M)} = \frac{1+\beta}{\log_2(M)}.$$

2.2 Maximum-Likelihood Sequence Estimation

Whenever ISI is present in the received signal, sequence detection can be performed. This section considers the maximum-likelihood sequence estimation (MLSE) algorithm when the communication channel is assumed to be the AWGN channel, i.e., when the received signal $r(t)$ can be expressed as

$$r(t) = s_{\boldsymbol{a}}(t) + n(t). \tag{2.45}$$

As in Section 2.1.3, it is assumed that $n(t)$ is a complex-valued white Gaussian process with mean $\mathbb{E}[n(t)] = 0$ and autocorrelation

$$\mathbb{E}[n(t)n^*(t+\tau)] = N_0\delta(\tau). \tag{2.46}$$

Additionally, it is assumed that the data transmission is uncoded, i.e., $\boldsymbol{u} = \boldsymbol{v}$. Even if there is no multipath in the environment, that is $c(t) = \delta(t)$, there can still be need for sequence detection if the ISI is intentionally introduced in the transmitter as will often be the case in this thesis.

In order to detect the data symbols $\boldsymbol{a}$ from the received signal $r(t)$, MLSE can be applied at the receiver. The MLSE decoding rule is

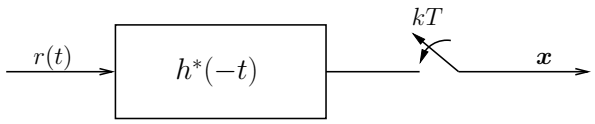

Figure 2.9: An efficient way to generate the sequence $\boldsymbol{x}$ from the received signal $r(t)$.

$$\hat{\boldsymbol{a}} \stackrel{\triangle}{=} \arg\max_{\boldsymbol{a}} p(r(t)|\boldsymbol{a}), \tag{2.47}$$

where $p(r(t)|\boldsymbol{a})$ is the conditional probability density function (PDF) of $r(t)$ given that the sequence $\boldsymbol{a}$ is sent. It is possible to show [17] that (2.47) is optimal if and only if all symbol sequences $\boldsymbol{a}$ are equiprobable. Further, it is well-known [17] that for an AWGN channel, the optimization in (2.47) is equivalent to minimizing the Euclidean distance between the received signal and the estimated signal, i.e.,

$$\begin{aligned} \hat{\boldsymbol{a}} &= \arg\min_{\boldsymbol{a}} \int_{-\infty}^{\infty} |r(t) - s_{\boldsymbol{a}}(t)|^2 \,\mathrm{d}t \\ &= \arg\min_{\boldsymbol{a}} \int_{-\infty}^{\infty} |r(t)|^2 - 2\mathcal{R}\{r(t)s_{\boldsymbol{a}}^*(t)\} + |s_{\boldsymbol{a}}(t)|^2 \,\mathrm{d}t. \end{aligned} \tag{2.48}$$

Note that the term $\int |r(t)|^2 \,\mathrm{d}t$ has no impact on the minimization (does not depend on $\boldsymbol{a}$) and can therefore be omitted. By inserting (2.1) in (2.48), the minimization of the Euclidean distance reduces to the maximization

$$\begin{aligned} \hat{\boldsymbol{a}} &= \arg\max_{\boldsymbol{a}} \int_{-\infty}^{\infty} \left(\mathcal{R}\{r(t)s_{\boldsymbol{a}}^*(t)\} - \frac{1}{2}|s_{\boldsymbol{a}}(t)|^2\right) \mathrm{d}t \\ &= \arg\max_{\boldsymbol{a}} \int_{-\infty}^{\infty} \left(\mathcal{R}\left\{r(t)\sum_{k=0}^{\infty} a_k^* h^*(t-kT)\right\} - \frac{1}{2}|s_{\boldsymbol{a}}(t)|^2\right) \mathrm{d}t \\ &= \arg\max_{\boldsymbol{a}} \sum_{k=0}^{\infty} \mathcal{R}\{a_k^* x_k\} - \int_{-\infty}^{\infty} \frac{1}{2}|s_{\boldsymbol{a}}(t)|^2 \,\mathrm{d}t, \end{aligned} \tag{2.49}$$

where

$$x_k \triangleq \int_{-\infty}^{\infty} r(t)h^*(t-kT)\,\mathrm{d}t. \tag{2.50}$$

The sequence $\boldsymbol{x} = [x_0, x_1, x_2, \ldots]$ can be obtained by applying a matched filter $h^*(-t)$ together with baud rate sampling at the receiver. This is shown schematically in Figure 2.9. Furthermore, the sequence $\boldsymbol{x}$ is a set of *sufficient statistics* for detecting $\boldsymbol{a}$. By inserting the expression for $r(t)$ in (2.50), the samples x_k become

$$\begin{aligned} x_k &= \sum_{j=0}^{\infty} a_j \int_{-\infty}^{\infty} h(t-jT)h^*(t-kT)\,\mathrm{d}t + \int_{-\infty}^{\infty} n(t)h^*(t-kT)\,\mathrm{d}t \\ &= \sum_{j=0}^{\infty} a_j g_{k-j} + \eta_k, \end{aligned} \tag{2.51}$$

where the sequence $\boldsymbol{g} = [g_{-L}, \ldots, g_0, \ldots, g_L]$ is ISI if $g_k \neq \delta[k]$. Likewise, the noise sequence $\boldsymbol{\eta}$ is a colored sequence if $g_k \neq 0$ for $k \neq 0$. An equivalent discrete-time model of (2.45) is therefore

$$\boldsymbol{x} = \boldsymbol{a} \star \boldsymbol{g} + \boldsymbol{\eta}. \tag{2.52}$$

The model in (2.52) is in this thesis referred to as the Ungerboeck observation model [21]. The noise sequence $\boldsymbol{\eta}$ is Gaussian with zero mean and autocorrelation

$$\phi_{\boldsymbol{\eta}}(j, j+k) = N_0 g_k. \tag{2.53}$$

A so-called Forney observation model is often preferred to the Ungerboeck model due to the whiteness of the noise at its output. Forney proposed [4] that the sampled MF outputs could be modeled as a trellis structure as follows. The outputs could be filtered with a discrete-time whitening filter (see Figure 2.10) in order to produce the sequence $\boldsymbol{y}$ given by

$$\boldsymbol{y} = \boldsymbol{a} \star \boldsymbol{f} + \boldsymbol{w}. \tag{2.54}$$

The sequence $\boldsymbol{f}$ is a causal $(L+1)$-tap long ISI response sequence with autocorrelation $\boldsymbol{g}$ while $\boldsymbol{w}$ is a random Gaussian sequence with zero mean and autocorrelation

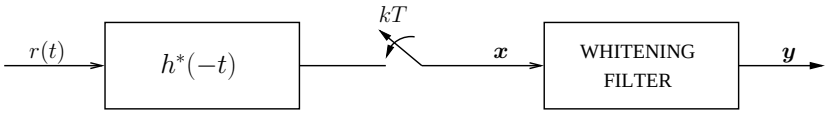

Figure 2.10: Forney generation of the sequence $\boldsymbol{y}$ from the received signal $r(t)$.

$$\phi_{\boldsymbol{w}}(j, j+k) = N_0\delta[k]. \tag{2.55}$$

The sequence $\boldsymbol{y}$ also forms a set of sufficient statistics, i.e., knowing $\boldsymbol{y}$ is sufficient to perform MLSE. However, a practical implementation of the Forney observation model can in some cases suffer from filter stability problems. This issue is considered in Chapter 3 where modifications and improvements of the discrete-time models are proposed.

2.2.1 Spectral Factorization

The whitening filter in Figure 2.10 can be obtained by so-called spectral factorization, briefly explained in this section. The transmitted signal for a data symbol sequence $\boldsymbol{a}$ and a discrete-time causal ISI response $\boldsymbol{f}$ (Forney observation model) can be expressed as

$$s_k = \sum_{j=0}^{\infty} a_k f_{k-j}. \tag{2.56}$$

The autocorrelation of the discrete sequence $\boldsymbol{f}$ is according to previous section

$$g_k = \sum_{j=-\infty}^{\infty} f_j f_{j+k}. \tag{2.57}$$

In [22] it is shown that the z-transform of g_k, denoted $G(z)$, can be expressed as

$$G(z) = c_{\mathrm{n}} c_{\mathrm{n}}^* \prod_{i=1}^{N_z} (1 - \epsilon_i z^{-1})(1 - \epsilon_i^* z) \tag{2.58}$$

where c_{n} is a normalization constant while ϵ_i and ϵ_i^* are the zeros of $G(z)$. Furthermore, it is always possible to choose

$$V(z) = c_{\mathrm{n}} \prod_{i=1}^{N_z} (1 - \epsilon_i z^{-1}) \tag{2.59}$$

satisfying

$$V^*(1/z^*) = c_{\mathrm{n}}^* \prod_{i=1}^{N_z} (1 - \epsilon_i^* z) \tag{2.60}$$

which together result in

$$G(z) = V(z)V^*(1/z^*). \tag{2.61}$$

Note that there are many different ways to construct $V(z)$. If v_j denotes the output sequence from the inverse z-transform of $V(z)$, i.e., $v_j = \mathcal{Z}^{-1}\{V(z)\}$, it is obvious that in general v_j is different from f_j. Since $\mathcal{Z}^{-1}\{V^*(1/z^*)\}$ equals v_{-j}^*, an alternative expression of the autocorrelation in (2.57) is

$$g_k = \sum_{j=-\infty}^{\infty} v_j v_{j+k}. \tag{2.62}$$

A minimum phase sequence, denoted v_j^{MP} is obtained by taking the inverse z-transform of a $V(z)$ constructed from (2.59) with $|\epsilon_i| \leq 1$, i.e., with zeros on or inside the unit circle. The corresponding whitening filter equals $1/V^*(1/z^*)$. It should be pointed out that all sequences $\{v_j\}$ produce the same minimum Euclidean distance (depends on the autocorrelation g_k) and have equivalent detection properties with optimal detection.

This section ends with a simple example. Assume the following unit-energy discrete-time causal ISI response $\boldsymbol{f}$

$$\boldsymbol{f} = \frac{1}{\sqrt{8}} [1,\ 0,\ 1,\ 2,\ 1,\ 0,\ 1]\,. \tag{2.63}$$

The corresponding autocorrelation sequence $\boldsymbol{g}$ is given by

$$\boldsymbol{g} = \frac{1}{8} [1,\ 0,\ 2,\ 4,\ 3,\ 4,\ 8,\ 4,\ 3,\ 4,\ 2,\ 0,\ 1]\,. \tag{2.64}$$

By performing spectral factorization of $G(z)$ and choosing $V(z)$ such that $|\epsilon_i| \leq 1$ we obtain the corresponding minimum phase discrete-time sequence

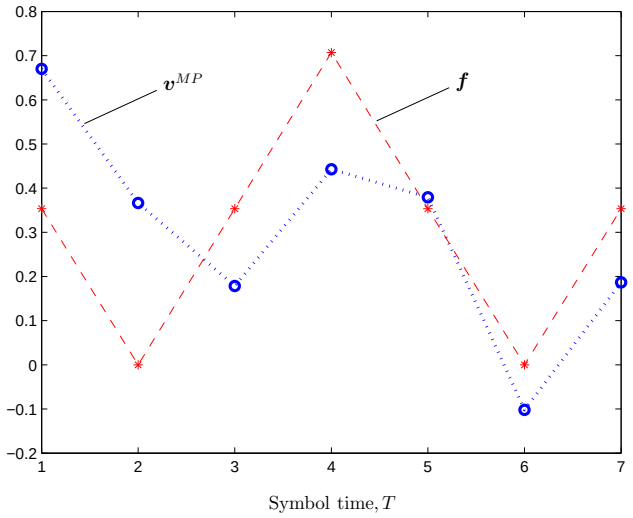

Figure 2.11: Example of two different causal ISI responses having the same autocorrelation $\boldsymbol{g}$.

$$\boldsymbol{v}^{MP} = [.670,\ .366,\ .178,\ .443,\ .379,\ -.102,\ .187]\,. \tag{2.65}$$

Both ISI responses are plotted in Figure 2.11. Although they look different, they have the same PSD, minimum Euclidean distance and optimal detection properties. Minimum phase concentrates energy to the early taps, in particular

$$\sum_{k=0}^{R} \left|v_k^{MP}\right|^2 \geq \sum_{k=0}^{R} |f_k|^2 \tag{2.66}$$

for any R.

2.2.2 The Recursive Structure of the MLSE

This section discusses the recursive structure of the MLSE algorithm, implemented by the Viterbi algorithm (VA). The assumptions are an uncoded data sequence $\boldsymbol{a}$ and that the reader is familiar with the trellis structure of the Forney-based algorithm, i.e., an algorithm based on (2.54). Nonetheless a brief overview is given next.

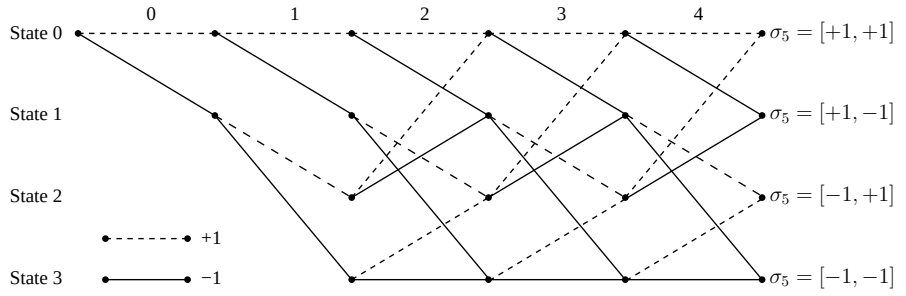

Figure 2.12: An example of a 4-state binary trellis.

Assume for simplicity a length-N binary ($\Omega = \{+1, -1\}$) data sequence $\boldsymbol{a}$ and a causal ISI response $\boldsymbol{f}$ such that

$$f_k = 0, \quad k > L \tag{2.67}$$

The binary setup above can be associated with a 2^L-state trellis of depth N where each state corresponds to a distinct combination of the L most recent symbols, that is

$$\sigma_k \triangleq [a_{k-L}, a_{k-L+1}, \ldots, a_{k-1}] \tag{2.68}$$

where σ_k denotes a state at depth k. An example of a 4-state binary trellis is shown in Figure 2.12 that is, $L = 2$. A state σ_k is connected to two different states at depth $k + 1$. These two states can be uniquely identified by the symbol pattern corresponding to the origin state at depth k and the transition symbol at time k. Let $\sigma_k = [a_{k-L}, a_{k-L+2}, \ldots, a_{k-1}]$ be a state in the trellis and $a_k \in \{+1, -1\}$ be the transition symbol at time k. The two states that are connected to σ_k are now given by $[a_{k-L+1}, a_{k-L+2}, \ldots, a_{k-1}, +1]$ and $[a_{k-L+1}, a_{k-L+2}, \ldots, a_{k-1}, -1]$ for input $+1$ and -1, respectively. Clearly, the succession of states is Markovian. Furthermore a line that connects two states is called a branch while a sequence of connected states is denoted a trellis path. An MLSE algorithm selects the most probable data symbol sequence that maximizes (2.47). If we assume that all paths begin from the so-called *all-zero* state, $\sigma_0 = [+1, +1, ..., +1]$, there exist in total 2^N different paths through the trellis. Since every path represents a distinct symbol sequence the decoding problem is equivalent to finding the most probable trellis path.

Consider now the alternative expression of the last term in (2.49) given by

$$\int_{-\infty}^{\infty} \frac{1}{2}|s_{\boldsymbol{a}}(t)|^2 \,\mathrm{d}t = \frac{1}{2}\sum_{j=0}^{\infty}\sum_{k=0}^{\infty} a_j a_k^* g_{j-k}. \tag{2.69}$$

From (2.67) we get that $\boldsymbol{g}$ is a finite ISI sequence , i.e., $g_k = 0$ when $|k| > L$. By introducing the likelihood function (compare with (2.49))

$$\Theta(\boldsymbol{a}) = \sum_{k=0}^{\infty} \mathcal{R}\{a_k^* x_k\} - \frac{1}{2}\sum_{j=0}^{\infty}\sum_{k=0}^{\infty} a_j a_k^* g_{j-k}, \tag{2.70}$$

we notice that $\Theta(\boldsymbol{a})$ can be recursively computed as

$$\Theta(\ldots, a_{k-1}, a_k) = \Theta(\ldots, a_{k-2}, a_{k-1}) + \mathcal{R}\left\{a_k^*\left(x_k - \frac{1}{2}g_0 a_k - \sum_{l=1}^{L} g_l a_{k-l}\right)\right\}. \tag{2.71}$$

Since the term $\sum_{l=1}^{L} g_l a_{k-l}$ is a contribution from the past L symbols, the same trellis state representation as in the Forney-based algorithm can be used. Even though the length of the autocorrelation sequence $\boldsymbol{g}$ is $2L+1$, the number of states in an Ungerboeck-based algorithm is $|\Omega|^L$, that is exponential in L. The trellises have the same branching structure but since the branch labels and computation rules are different, the performance of reduced-complexity algorithms based on the two observation models is in general different. This is investigated in Chapter 4 of this thesis.

By using the same notation as in (2.68), we can now define the so-called *survivor metric* of the VA as

$$\tilde{\Theta}(\sigma_{k+1}) \triangleq \max_{[a_0,\ldots,a_{k-L}]} \Theta(\ldots, a_{k-1}, a_k). \tag{2.72}$$

Finally, combining (2.71) and (2.72) results in

$$\tilde{\Theta}(\sigma_{k+1}) = \mathcal{R}\{a_k^* x_k\} + \max_{\sigma_k \to \sigma_{k+1}} \tilde{\Theta}(\sigma_k) - \frac{1}{2}a_k^* g_0 a_k - \mathcal{R}\{a_k^* \sum_{l=1}^{L} g_l a_{k-l}\}. \tag{2.73}$$

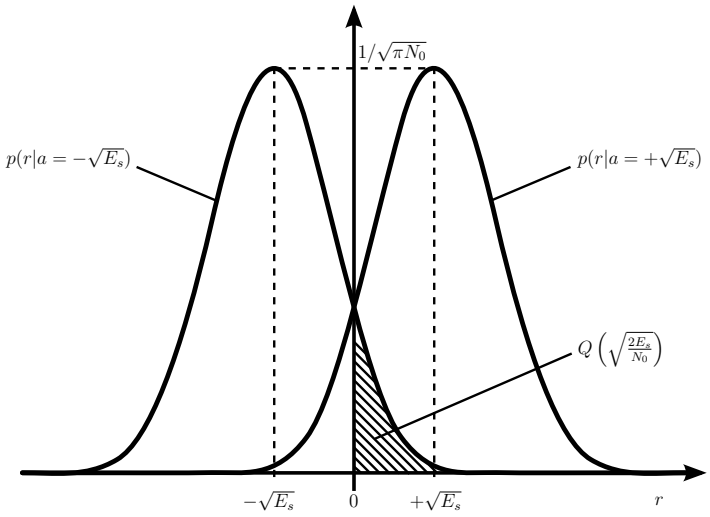

Figure 2.13: Conditional output PDFs of the AWGN channel with binary inputs. This channel can be characterized by $|\Omega|$ conditional PDFs $p(r|a)$.

2.2.3 MLSE Error Performance

The error performance of the MLSE algorithm is considered in this section. Let us begin by defining the probability of a symbol error as

$$P_s \triangleq \Pr(\hat{a}_k \neq a_k) \tag{2.74}$$

where $\hat{a}_k$ is an MLSE estimated symbol at depth k. Since there exists no closed form expression of P_s in the case of ISI, upper bounds must used. In order to proceed with the results we introduce the complementary Gaussian distribution function, also known as the Gaussian tail function, defined as

$$Q(x) \triangleq \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-t^2/2}\,\mathrm{d}t. \tag{2.75}$$

An upper bound to P_s is now given by [4, 23]

$$P_s \leq \sum_{\boldsymbol{e}\in\mathcal{X}_{\boldsymbol{e}}} Q\left(\sqrt{d^2(\boldsymbol{e})\frac{E_b}{N_0}}\right) m_{\boldsymbol{e}} d_H(\boldsymbol{e}). \tag{2.76}$$

In (2.76) $\mathcal{X}_{\boldsymbol{e}}$ denotes the set of all possible error events while $m_{\boldsymbol{e}}$ and $d_H(\boldsymbol{e})$ are the multiplicity and Hamming weight of the error event $\boldsymbol{e}$, respectively. In the case of a binary symbol alphabet, $\Omega = \{+1, -1\}$, it can be shown that the multiplicities $m_{\boldsymbol{e}}$ equal

$$m_{\boldsymbol{e}} = 2^{-d_H(\boldsymbol{e})}. \tag{2.77}$$

Furthermore, by exploiting that $d^2(\boldsymbol{e}) = d^2(-\boldsymbol{e})$, $d_H(\boldsymbol{e}) = d_H(-\boldsymbol{e})$ and $m_{\boldsymbol{e}} = m_{-\boldsymbol{e}}$, the summation in (2.76) can be done only over those events $\boldsymbol{e} = [e_0, e_1, \ldots]$ which have $e_0 = 2$ with $m_{\boldsymbol{e}}$ given by

$$m_{\boldsymbol{e}} = 2^{1-d_H(\boldsymbol{e})}. \tag{2.78}$$

For more general cases the reader is referred to [17, 19].

Since, according to Figure 2.13, $Q(x)$ has a steep descent towards 0, the dominating terms in (2.76) are those corresponding to the minimum distance. In fact, Forney showed in [4] that if the sum in (2.76) converges, there exist constants K_1 and K_2 such that

$$K_1 Q\left(\sqrt{d_{\min}^2 \frac{E_b}{N_0}}\right) \leq P_s \leq K_2 Q\left(\sqrt{d_{\min}^2 \frac{E_b}{N_0}}\right). \tag{2.79}$$

The first error probability P_{ee}, that is the probability that an error event starts at depth k, given that there are no errors up to this depth, can be defined as

$$P_{ee} \triangleq \Pr(\hat{a}_k \neq a_k | \hat{a}_{k-L} = a_{k-L}, \ldots, \hat{a}_{k-1} = a_{k-1}) \tag{2.80}$$

By omitting the factor $d_H(\boldsymbol{e})$ in (2.76), an upper bound to P_{ee} is obtained.

2.3 The M-algorithm

Even though the MLSE algorithm from previous section is the optimal sequence detector, its complexity is exponential in the length of the ISI response L. If the size of the underlying trellis, $|\Omega|^L$, becomes too large a realization of the MLSE algorithm is not practical. Therefore, it is important to consider suboptimum reduced-search trellis decoders. One well-established technique, introduced by

Anderson in 1969 [6], is the M-algorithm which is briefly described next. This decoder will be frequently used throughout the thesis.

The M-algorithm is a suboptimal trellis-search technique which reduces the complexity of trellis decoding by traversing only a part of the trellis. At each depth k, only the M most likely states (survivors),

$$\{\sigma_k^1,\ \sigma_k^2,\ \ldots,\ \sigma_k^M\} \tag{2.81}$$

with the highest cumulative metric values $\tilde{\Theta}(\cdot)$ are extended to depth $k+1$ while the remaining paths are discarded. The M-algorithm is therefore also known as list decoding since the set of M subpaths form a list of size M. Furthermore, no branch is to be extended from a discarded state. The retained states are the states that lie closest in Euclidean distance to the received signal. When a new signal vector is received, the M-algorithm extends the M retained states to the next trellis interval, generating up to $M|\Omega|$ new states. In this thesis, duplicates in the list are not allowed. The algorithm then identifies the survivor for each new state and sorts the set of new paths according to their cumulative metric values. The most promising M are retained while the rest are discarded. The M-algorithm repeats this process until the end of the trellis is reached. A path that reaches the end of the trellis with the highest cumulative metric $\tilde{\Theta}_{\max}(\cdot)$ is denoted the approximated ML path.

The main advantage of the M-algorithm over decoding techniques such as the T-algorithm and the stack algorithm is that it performs the same number of computations at each trellis depth k. It is therefore relatively easy to specify the parameter M in order to meet a desired computational complexity level. The total number of branch metric computations in a depth N trellis, when extending only the best M survivors from one depth to the next, is reduced from $|\Omega|^L|\Omega|N$ to $M|\Omega|N$. However sorting of $M|\Omega|$ states according to their metric value is required at each depth k. In fact, there is no need for a complete sorting. Instead it is enough to find the M best values which is a linear operation in M [24].

In [25, 26] it was shown that the M-algorithm is optimal in the sense of minimizing the probability of correct path loss among the constant-complexity breadth-first search decoders. Variants of the algorithm can be found in [27, 28, 29].

Suboptimal decoders either traverse a small part of a full trellis or all possible paths in a reduced size trellis. Decoders like the M-algorithm that only move in the forward direction are called breadth-first decoders while decoders that allow backward motion are called backtracking decoders. The breadth-first trellis class of decoders can be further classified into one-way decoders

and two-way decoders. One-way decoders like the VA perform only a single recursion while two-way decoders like the BCJR in Section 2.4.1 perform one forward and one backward recursion. Examples of one-way decoders from the first class are the very popular soft output VA (SOVA) presented by Hagenauer and Hoeher in [12] and its reduced complexity variant, the soft output M-algorithm (SOMA), proposed in [14]. Other reduced complexity decoders from the breadth-first class can be found in [30, 31]. Two examples from the backtracking class are the Fano and the stack algorithm. A complete treatment of decoders from both classes is given in [32, 33].

2.4 Maximum a Posteriori Symbol-by-Symbol Decoding

Optimal methods for minimizing the bit error rate and the block error rate are nonlinear and based on maximum-likelihood (ML) estimation. In the presence of *a priori* information about the transmitted data $\boldsymbol{a}$ this turns into maximum *a posteriori* probability (MAP) estimation. This section considers the MAP symbol-by-symbol trellis decoder proposed by Bahl, Cocke, Jelinek and Raviv in 1974 [11]. It is commonly referred to as the BCJR algorithm.

Let $\Pr(\hat{\boldsymbol{a}} = \boldsymbol{a}) = 1 - \Pr(\hat{\boldsymbol{a}} \neq \boldsymbol{a})$ be the probability of a correct decision of the transmitted symbol sequence at the receiver. Further, let $p(\boldsymbol{y})$ be the PDF of the received sequence $\boldsymbol{y} = [y_0, y_1, y_2, \ldots]$ from (2.54). Then, the probability that the decision $\hat{\boldsymbol{a}}$ is correct can be expressed as

$$\Pr(\hat{\boldsymbol{a}} = \boldsymbol{a}) = \int_{\boldsymbol{y}} \Pr(\hat{\boldsymbol{a}} \text{ sent}|\boldsymbol{y}) p(\boldsymbol{y})\, d\boldsymbol{y}. \tag{2.82}$$

The objective of an optimal decoder is to minimize the error probability or, equivalently, maximize the probability of a correct decision. The right-hand side of (2.82) is maximized when the term $\Pr(\hat{\boldsymbol{a}} \text{ sent}|\boldsymbol{y})$ is maximized for each $\boldsymbol{y}$. Thus, upon observing the received signal $\boldsymbol{y}$, the optimal decision rule is

$$\hat{\boldsymbol{a}} = \arg\max_{\boldsymbol{a}} \Pr(\boldsymbol{a} \text{ sent}|\boldsymbol{y}). \tag{2.83}$$

The decision rule in (2.83) is known as the *maximum a posteriori* (MAP) rule. This receiver minimizes the probability of detecting an erroneous message. We can alternatively write

$$\Pr(\boldsymbol{a} \text{ sent}|\boldsymbol{y}) = \frac{p(\boldsymbol{y}|\boldsymbol{a} \text{ sent}) \Pr(\boldsymbol{a} \text{ sent})}{p(\boldsymbol{y})}. \tag{2.84}$$

Since $p(\boldsymbol{y})$ is independent of $\boldsymbol{a}$ it can be omitted in the maximization. If, additionally, all symbol sequences $\boldsymbol{a}$ are equiprobable, the maximization in (2.83) is equivalent to

$$\hat{\boldsymbol{a}} = \arg\max_{\boldsymbol{a}} p(\boldsymbol{y}|\boldsymbol{a} \text{ sent}). \tag{2.85}$$

The term $p(\boldsymbol{y}|\boldsymbol{a})$ is called the *likelihood* of $\boldsymbol{a}$ while a decoder defined by (2.85) is, according to Section 2.2, known as a *maximum likelihood* (ML) decoder. This decoder is optimal in the case of equiprobable symbol sequences $\boldsymbol{a}$. In this thesis, we in fact assume that $\boldsymbol{a}$ is uniformly distributed.

Consider now a MAP *sequence* equalizer which finds the most probable data sequence $\hat{\boldsymbol{a}}$ according to (2.83). Since both (2.52) and (2.54) are sufficient statistics for optimal detection we can write

$$\hat{\boldsymbol{a}} = \arg\max_{\boldsymbol{a}} \Pr(\boldsymbol{a}|\boldsymbol{x}) = \arg\max_{\boldsymbol{a}} \Pr(\boldsymbol{a}|\boldsymbol{y})$$

where $\boldsymbol{x}$ and $\boldsymbol{y}$ are received observations from (2.52) and (2.54), respectively. Furthermore, $\Pr(\boldsymbol{a})$ in (2.84) is known as the a priori sequence probability and it is possibly provided from a convolutional code decoder in an iterative turbo loop. If independent data symbols can be assumed, $\Pr(\boldsymbol{a})$ factorizes into

$$\Pr(\boldsymbol{a}) = \prod_{k=0}^{N-1} \Pr(a_k)$$

where N is the sequence length. One of the main reasons why the Forney observation model is often preferred over the Ungerboeck model is the whiteness of the noise samples at the receiver which, together with the independence assumption, allows the following factorization

$$p(\boldsymbol{y}|\boldsymbol{a}) = \prod_{k=0}^{N-1} p(y_k|\boldsymbol{a}). \tag{2.86}$$

Since each term in (2.86) is given by

$$p(y_k|\boldsymbol{a}) \propto \exp\Big(-\frac{1}{N_0}\Big|y_k - \sum_{l=0}^{L} f_l a_{k-l}\Big|^2\Big) \tag{2.87}$$

the VA branch metric at kth trellis stage is proportional to $\Pr(a_k)p(y_k|\boldsymbol{a})$.

In contrast to the Forney model, the received observations $\boldsymbol{x}$ of the Ungerboeck model are corrupted by colored noise which prohibits the factorization (2.86). Instead the likelihood $\Pr(\boldsymbol{x}|\boldsymbol{a})$ can be factorized [21] as

$$p(\boldsymbol{x}|\boldsymbol{a}) \propto \prod_{k=0}^{N} \varphi(x_k, \boldsymbol{a}) \tag{2.88}$$

where $\varphi(x_k, \boldsymbol{a})$ is given by

$$\varphi(x_k, \boldsymbol{a}) \triangleq \exp\left(\frac{2}{N_0} a_k^* \left(x_k - \frac{g_0}{2} a_k - \sum_{l=1}^{L} g_l a_{k-l}\right)\right). \tag{2.89}$$

2.4.1 The BCJR Algorithm

A MAP *symbol* decoder decides in favor of the symbol $\hat{a}_k$, using the following decision rule

$$\hat{a}_k = \arg\max_{a_k} \Pr(a_k|\boldsymbol{x}) = \arg\max_{a_k} \Pr(a_k|\boldsymbol{y}). \tag{2.90}$$

This decoder also provides soft information about the symbols, in the form of logarithmic a posteriori (APP) ratios, sometimes referred to as L-values, defined as

$$L(a_k|\boldsymbol{y}) \stackrel{\triangle}{=} \log\left(\frac{\Pr(a_k = +1|\boldsymbol{y})}{\Pr(a_k = -1|\boldsymbol{y})}\right) = \log\left(\frac{\sum_{\boldsymbol{a}:a_k=+1} \Pr(\boldsymbol{a}|\boldsymbol{y})}{\sum_{\boldsymbol{a}:a_k=-1} \Pr(\boldsymbol{a}|\boldsymbol{y})}\right). \tag{2.91}$$

In (2.91) we have, for simplicity, assumed a binary PAM (2-PAM) alphabet, $\Omega = \{+1, -1\}$, and the Forney observation model. The APP ratio can further be expressed as

$$L(a_k|\boldsymbol{y}) = \log\left(\frac{\sum_{(\sigma,\sigma')\in\mathcal{S}^+} p(\sigma_k = \sigma, \sigma_{k+1} = \sigma', \boldsymbol{y})}{\sum_{(\sigma,\sigma')\in\mathcal{S}^-} p(\sigma_k = \sigma, \sigma_{k+1} = \sigma', \boldsymbol{y})}\right) \tag{2.92}$$

where $\mathcal{S}^+$ and $\mathcal{S}^-$ are the sets of trellis state pairs (σ_k, σ_{k+1}) at depth k that correspond to $a_k = +1$ and $a_k = -1$, respectively. Note that in (2.92) we have also assumed time-invariant trellises.

The BCJR algorithm computes probabilities of states and paths in a trellis, given the channel outputs $\boldsymbol{y} = [y_0, y_1, \ldots, y_{N-1}]$ and the a priori data

probabilities. It efficiently computes logarithmic APP ratios by exploiting the factorization

$$p(\sigma_k = \sigma, \sigma_{k+1} = \sigma', \boldsymbol{y}) = \alpha_k(\sigma)\gamma_k(\sigma, \sigma')\beta_{k+1}(\sigma'). \tag{2.93}$$

Here, the recursively calculated forward and backward trellis metrics of the state σ at kth trellis depth are denoted $\alpha_k(\sigma)$ and $\beta_k(\sigma)$, respectively. The metric of the branch connecting the states (σ, σ'), denoted $\gamma_k(\sigma, \sigma')$, can in the case of the Forney model, be expressed as

$$\gamma_k(\sigma, \sigma') = p(\sigma, y_k|\sigma') = \Pr(a_k)p(y_k|\boldsymbol{a}) \tag{2.94}$$

where the likelihoods $p(y_k|\boldsymbol{a})$ are given by (2.87) while, in the Ungerboeck model,

$$\gamma_k(\sigma, \sigma') = \Pr(a_k)\varphi(x_k|\boldsymbol{a}) \tag{2.95}$$

where $\varphi(x_k, \boldsymbol{a})$ is given by (2.89).

Starting from the initial all-zero state at the root of the trellis, the forward metric is computed recursively in a forward trellis pass according to

$$\alpha_{k+1}(\sigma') = \sum_{\sigma \in \mathcal{S}} \alpha_k(\sigma)\gamma_k(\sigma, \sigma') \tag{2.96}$$

with the initialization $\boldsymbol{\alpha}_0 = [1, 0, \ldots, 0]$, where $\mathcal{S}$ is the set of states that can reach state σ' at depth $k + 1$ (in binary transmission there are 2). Similarly, the backward recursion, initialized with $\boldsymbol{\beta}_N = [1, 0, \ldots, 0]^{\mathrm{T}}$, starts at the end of the trellis and proceeds towards the root, computing at each trellis depth k

$$\beta_k(\sigma) = \sum_{\sigma' \in \mathcal{S}} \beta_{k+1}(\sigma')\gamma_k(\sigma, \sigma'). \tag{2.97}$$

The superscript "T" denotes the transpose operator throughout this thesis. Now $\mathcal{S}$ is the set of states reached from the state σ at depth k. Note that, in the Forney observation model, the backward recursion starts at trellis depth N from the all-zero state, i.e., $\boldsymbol{\beta}_N = [1, 0, \ldots, 0]^{\mathrm{T}}$. In the Ungerboeck model the trellis is not terminated in the all-zero state and consequently the backward recursion must instead be initialized with $\beta_N(\sigma) = 1/|\Omega|^L$, for all σ.

The probabilistic interpretation of the forward and the backward state metrics in the Forney model is

$$\begin{aligned} \alpha_k(\sigma) &= p(\sigma, y_0, y_1, \ldots, y_{k-1}) \\ \beta_{k+1}(\sigma) &= p(y_{k+1}, y_{k+2}, \ldots, y_{N-1}|\sigma) \end{aligned}$$

This interpretation of the state metrics is not valid in the Ungerboeck model. In [34] it is shown that the function $\varphi(x_k, \boldsymbol{a})$ is not a true PDF but since (2.88) holds, Viterbi equalization can be performed. Additionally, a BCJR-type algorithm for ISI (based on (2.88)) is derived and it is shown that its output is equivalent to the output of a standard Forney-based BCJR. However this statement is only true when optimal detection is adopted. In Chapter 4 it is shown that reduced-complexity equalizers, based on the two models, will in general produce different outputs. Note that the BCJR-type algorithm from [34] has the same computational complexity as the Forney-based BCJR algorithm.

2.5 Faster-than-Nyquist Signaling

This section reviews some of the underlying ideas of faster-than-Nyquist (FTN) signaling. This signaling method has existed in some form since 1975 and it is based on the fact that pulse amplitude modulation (PAM) signals of the form

$$\sum a_k h(t - kT) \tag{2.98}$$

where $h(t)$ is a T-orthogonal pulse, can be sent faster than the Nyquist signaling rate $1/T$ without any loss in minimum Euclidean distance. Thus, the asymptotic error rate behavior of an optimal decoder remains unchanged. FTN signaling increases the data transmission rate by reducing the time-spacing between adjacent pulses below the Nyquist rate while keeping a fixed power spectral density (PSD). Note that the PSD shape, in case of IID inputs, in (2.18) only depends on the modulation pulse $h(t)$. FTN provides improved spectral efficiency that cannot be reached by communication systems based on orthogonal (Nyquist) signaling.

The technique was introduced already in 1975 by Mazo [35]. He showed that binary T-orthogonal $\mathrm{sinc}(\cdot)$ pulses in (2.98) could be sent faster (symbol time $\tau T, \tau < 1$) without loss in minimum Euclidean distance. In fact, the symbol time can be reduced to $0.802T$ without any distance loss. In other words, $1/0.802 \approx 25\%$ more bits could be carried in the same bandwidth without affecting the asymptotic error rate. He called this faster-than-Nyquist signaling and the value $0.802T$ is called the *Mazo limit*. Even though the asymptotic error

probability in optimal detection remains unaffected (above the Mazo limit), FTN violates the Nyquist orthogonality criterion and consequently a controlled amount of intentional intersymbol interference (ISI) is introduced. The Nyquist orthogonality criterion states that in order to carry $1/T$ bits/s, a baseband bandwidth of at least $1/2T$ Hz is required.

Due to significant spectral sidelobes, Foschini concluded in [36] that FTN cannot be competitive. However, in his work the ISI support was limited to a small duration. Since then, the concept of FTN has been extended in many ways: The modulation can be coded, it can be nonbinary [37], in a general way it also applies to nonlinear modulation [38], the pulse does not need to be sinc$(\cdot)$ or even orthogonal. The concept can be applied in frequency as well as in time, by placing OFDM-like subcarriers closer than orthogonality allows [39]. Extensions of the FTN idea to multicarrier setups were proposed in [40, 41, 42]. BER results show that for the same bandwidth consumption multicarrier systems are superior to the single carrier system. In [43, 44] FTN receivers and related issues are studied. Additionally, a chapter in [45] is devoted to FTN signaling.

If the sinc$(\cdot)$ pulses arrive faster than $1/T$, the Nyquist orthogonality criterion is violated and ISI introduced. Hence, a more complex maximum-likelihood sequence estimation receiver is required in order to eliminate the effects of the intentional ISI. If the receiver is able to cope with the interference, the spectral efficiency of the system will be improved. This is also true for any other T-orthogonal pulse. In every case there will be a closest packing (a smallest τ and/or a closest subcarrier spacing) at which the minimum Euclidean distance first falls below the isolated pulse value. This is the Mazo limit to signaling with this $h(t)$ and alphabet. In [43], limits for root RC pulses with non-zero excess bandwidth β are derived. Additionally, [43] gives an early study of receivers for FTN signaling. Mazo limits for other pulse shapes, including those which are not orthogonal for any shift T, are derived in [46]. An interesting problem from a mathematical point of view is to find the minimum Euclidean distance of FTN signals. Some early work on this topic appears in [47, 48]. An extension to non-binary signaling over ISI with a non-rational z-transform is given in [49, 50]. In [49, 51, 52] it is shown analytically that FTN capacity is often higher than the capacity of memoryless modulation. Some of the major results from [49, 51, 52] will be stated in this section. In [53] it is shown that binary faster-than-Nyquist signaling can, asymptotically in the signaling rate, achieve the so-called PSD capacity defined as

$$C_{\mathrm{PSD}} = \int_0^\infty \log_2 \left(1 + \frac{2P}{N_0}|H(f)|^2\right) \mathrm{d}f \quad \text{bits/s} \tag{2.99}$$

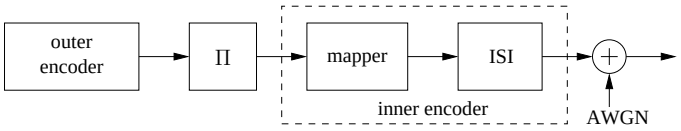

Figure 2.14: System model of a serially concatenated communication system with encoding and intersymbol interference. Π denotes an interleaver.

where P is the average signal power and $P|H(f)|^2$ is the signal PSD. In (2.99), $|H(f)|^2$ is normalized to unit integral. Later in this section, we will put the capacity in (2.99) further into the context. Results on precoding for FTN appear in [54]. Concatenated coding systems based on FTN that operate close to the theoretical capacity bounds were introduced in [55]. An example of a serially concatenated communication system with encoding and intersymbol interference is illustrated in Figure 2.14. In [56] FTN is for the first time considered in a MIMO setup. According to [57], the pulse shape $h(t)$ that results in the most favorable Mazo limit is nearly Gaussian. Note that the Gaussian pulse is not orthogonal for any shift kT. A method that improves the spectral efficiency of a linear modulation system by reducing the spacing between adjacent signals in both time and frequency domains, is, together with some low-complexity detectors, proposed in [58, 59]. The application of time and frequency packing to optical links has recently been considered in [60]. An extension of [59] to a more complex receiver structure appeared recently in [61]. Spectrally efficient FTN-type communication systems together with related hardware implementation issues are considered in [62, 63, 64, 65, 66].

The remainder of Section 2.5 is organized as follows. In Section 2.5.1 the system model for FTN signaling is given. Section 2.5.2 considers the capacity of FTN signals and presents some already existing capacity results.

2.5.1 System Model

Consider ordinary linearly modulated signals whose baseband form is

$$s_{\boldsymbol{a}}(t) = \sum_{k=0}^{\infty} a_k h(t - k\tau T), \quad \tau \leq 1 \tag{2.100}$$

where a_k are real equiprobable independent and identically distributed data symbols drawn from an alphabet Ω and $h(t)$ is a real unit-energy T-orthogonal baseband pulse. This signaling form with $\tau = 1$ underlies many practical

modulations, e.g., TCM and the subcarriers in orthogonal frequency-division multiplexing (OFDM) (in OFDM the data symbols a_k are complex). The signaling rate is $1/\tau T$. By setting $\tau = 1$ we obtain an orthogonal system. This ISI-free signaling will be referred to as *Nyquist signaling* while the case $\tau < 1$ is called *FTN signaling*. Note that in the latter the signaling time is $\mathcal{T} = \tau T < T$, i.e., there exists an integer n where

$$\int h(t)h(t - n\mathcal{T})\,\mathrm{d}t \neq 0. \tag{2.101}$$

Most often in this thesis the modulation pulse $h(t)$ in (2.100) is much narrower band than $1/2\tau T$ Hz and consequently severe ISI is introduced. Decoding of signals at a signaling rate near the Mazo limit is relatively simple. However, for smaller τ, leading to attractive combinations of bandwidth-energy efficiency and, in particular, higher bit densities, the decoder must be more complex. If, additionally, the signals are encoded, one needs to rely on iterative detection schemes. Section 2.7 describes the principles of an iterative receiver structure called turbo equalization.

Instead of transmitting faster, consider now transmission of wider pulses in time, that is, pulses given by

$$h_{\text{wide}}(t) = \sqrt{\tau}h(\tau t), \tag{2.102}$$

where one keeps the transmission rate of $1/T$. Since the widening factor is $1/\tau$, the new pulse is $\mathbb{T}$-orthogonal where $\mathbb{T} = T/\tau \geq T$. Consequently, the same discrete-time model as before is obtained for the new system. In fact, both systems are equivalent in terms of needed SNR versus bandwidth efficiency, measured by the normalized bandwidth.

With IID symbols the PSD from (2.18) for signals of the form (2.100) becomes

$$\Phi_{s_{\boldsymbol{a}}}(f) = \frac{\sigma_a^2}{\tau T}|H(f)|^2 \tag{2.103}$$

where

$$\sigma_a^2 = \mathbb{E}[|a_k|^2]. \tag{2.104}$$

By inserting $f = 0$ into (2.103), the average power P of an FTN transmission equals

$$P = \frac{\sigma_a^2}{\tau T}. \tag{2.105}$$

Furthermore, an AWGN channel follows (2.100). The received signal

$$r(t) = s_{\boldsymbol{a}}(t) + n(t) \tag{2.106}$$

where $n(t)$ is real white noise, is filtered with a filter matched to $h(t)$ and sampled each τT in order to produce the sequence

$$x_k = \int_{-\infty}^{\infty} r(t) h^*(t - k\tau T)\,\mathrm{d}t \tag{2.107}$$

which, according to Section 2.2, forms a set of sufficient statistics for detection. An equivalent discrete-time model of (2.106) is therefore

$$\boldsymbol{x} = \boldsymbol{a} \star \boldsymbol{g} + \boldsymbol{\eta}$$

where, if $g_k \neq \delta[k]$, the sequence $\boldsymbol{g} = [g_{-L}, \ldots, g_0, \ldots, g_L]$ is ISI and $\boldsymbol{\eta}$ is a sequence of colored Gaussian noise. Note that we in general do not encounter finite ISI, i.e., there exists no number L such that $g_k = 0$, $k > L$. Furthermore, we have that

$$\begin{aligned} g_k &= \int_{-\infty}^{\infty} |H(f)|^2 e^{j2\pi k\tau T f}\,\mathrm{d}f \\ &= \sum_{k=-\infty}^{\infty} \int_{-1/2\tau T}^{1/2\tau T} \left| H\left(f + \frac{k}{\tau T}\right)\right|^2 e^{j2\pi k\tau T f}\,\mathrm{d}f \\ &= \int_{-1/2\tau T}^{1/2\tau T} \sum_{k=-\infty}^{\infty} \left| H\left(f + \frac{k}{\tau T}\right)\right|^2 e^{j2\pi k\tau T f}\,\mathrm{d}f \\ &= \int_{-1/2\tau T}^{1/2\tau T} |H_{\mathrm{fo}}(f)|^2 e^{j2\pi k\tau T f}\,\mathrm{d}f, \end{aligned} \tag{2.108}$$

where $|H_{\mathrm{fo}}(f)|^2$ is the *folded* pulse spectrum

$$|H_{\mathrm{fo}}(f)|^2 \triangleq \sum_{k=-\infty}^{\infty} \left| H\left(f + \frac{k}{\tau T}\right)\right|^2, \quad -1/2\tau T \leq f \leq 1/2\tau T. \tag{2.109}$$

By applying a matched filter and sampling at the rate $1/\tau T$ the spectrum of the received ISI sequence is folded around $1/2\tau T$. This is the well-known spectrum folding that occurs from sampling [17]. Note that any transfer function $H(f)$ that gives rise to the same ISI sequence $\boldsymbol{g}$ has statistically equivalent detection properties. Therefore $H(f)$ and $H_{\text{fo}}(f)$ can be interchanged.

Let us conclude this section by formally stating the Mazo limit. We remind the reader that the minimum Euclidean distance of a balanced equispaced M-PAM alphabet and orthogonal transmission is given by the matched filter bound in (2.42).

Definition 1. *The Mazo limit is the smallest value $\tau^{\mathcal{M}}$ that fulfills $d^2_{\text{min}} = d^2_{\text{MF}}$ when $\tau = \tau^{\mathcal{M}}$.*

For 2-PAM and root RC pulses from Section 2.1.5 with excess bandwidths $\beta = \{0,\ 0.1,\ 0.2,\ 0.3\}$ the Mazo limits, rounded off to three digits of precision, are $\tau^{\mathcal{M}} = \{0.802,\ 0.779,\ 0.738,\ 0.703\}$, respectively.

2.5.2 The Capacity of FTN Signaling

Shannon showed in [3] that a signal of bandwidth W Hz spans $\approx 2WT$ independent dimensions during a time interval of T seconds. In other words a bandlimited signal of W Hz is completely specified by a set of $2WT$ numbers during T seconds. These numbers can be viewed as coordinates in a $2WT$ dimensional space. Furthermore, Shannon proved that these numbers can be transmitted by means of time shifted sinc pulses. Consequently, during T seconds, roughly $2WT$ data symbols $\boldsymbol{a} = [a_1, \ldots, a_{2WT}]$ can be sent.

Consider now linearly modulated signals of the form (2.1) where the data symbols $\{a_k\}$ are assumed to be equiprobable and IID. All possible sequences $\boldsymbol{a}$ are allowed unless the sequence $\boldsymbol{a}$ is encoded. Then, the design of the underlying code typically determines the subset of possible data sequences. Let the signals have an average power P and a rectangular PSD in the interval $[-W, W]$ where W is the one-sided width of the signal PSD. The highest transmission rate over the AWGN channel in (2.45) with noise power spectral density $N_0/2$ is given by

$$C = W \log_2 \left(1 + \frac{P}{W N_0} \right) \quad \text{bits/s.} \tag{2.110}$$

This is Shannon's classical capacity result from [3]. If the signals have a smooth PSD $P|H(f)|^2$ it can be approximated with many rectangular pieces, small

channels, which by application of integral calculus extends (2.110) to (2.99). The term *capacity* is in this thesis reserved for signals with a PSD $P|H(f)|$ in AWGN while *constrained capacity* is the maximum information rate under some restriction such as a certain symbol alphabet or FTN signaling. If we let $\mathcal{I}(\boldsymbol{y};\boldsymbol{x}) = \mathfrak{h}(\boldsymbol{y}) - \mathfrak{h}(\boldsymbol{y}|\boldsymbol{x})$ be the mutual information between the sequence $\boldsymbol{y}$ and $\boldsymbol{x}$, where $\mathfrak{h}(\cdot)$ is the differential entropy operator, the information rate is defined as

$$I \triangleq \lim_{N\to\infty} \mathcal{I}(\boldsymbol{y};\boldsymbol{x})/N \quad \text{bits/ch.use} \tag{2.111}$$

where N is the sequence length. The capacity in (2.110) can in principle be achieved by a transmitting $s_{\boldsymbol{a}}(t)$ of the form in (2.1) with $T = 1/2W$ [67], that is

$$s_{\boldsymbol{a}}(t) = \sum_{k=0}^{\infty} a_k \text{sinc}(t - k/2W) \tag{2.112}$$

where $\{a_k\}$ is a sequence of Gaussian data symbols and $\text{sinc}(t)$ is the sinc pulse in (2.38). However, since the sinc pulse is impractical, smoother pulses such as the root RC pulse in Section 2.1.5 are used instead. Despite the extra bandwidth, the optimal detection properties remain the same and so does also the capacity in (2.110). Let us now compare the capacity in (2.110) with that in (2.99) which is repeated here

$$C_{\text{PSD}} = \int_0^{\infty} \log_2\left(1 + \frac{2P}{N_0}|H(f)|^2\right) \mathrm{d}f \quad \text{bits/s.} \tag{2.113}$$

In [49] it is shown that practical non-sinc T-orthogonal pulses $h(t)$, antisymmetric around the point $(1/2T, |H(0)|^2/2)$, can only increase (2.113) compared to signaling with $h_{\text{sinc}}(t)$. Hence the capacity in (2.113) is higher than that in (2.110). However, this capacity increase cannot be achieved by orthogonal signals based on a non-sinc $H(f)$. FTN, on the other hand, utilizes the full potential of a given PSD shape [49]. The antisymmetric property of T-orthogonal pulses was extended by Gibby and Smith into [68]

$$\sum_{k=-\infty}^{\infty} \left|H\left(f + \frac{k}{T}\right)\right|^2 = T, \quad \forall f. \tag{2.114}$$

Let us now derive the capacity of FTN signaling. Henceforth, the only assumption on the data symbols is that $\{a_k\}$ are IID. Consider one of the

discrete-time models ((2.52) or (2.54)) presented in Section 2.2 and let $\Pr(\boldsymbol{a}) = \prod \Pr(a)$ be the probability mass function of the the data sequence $\boldsymbol{a}$. Assume also that N data symbols, $\boldsymbol{a}^N = [a_1, \ldots, a_N]$, are to be transmitted. The constrained capacity of a general ISI channel then equals

$$\begin{aligned} C_{\mathrm{DT}} &\triangleq \sup_{p_{a(a)}} \lim_{N\to\infty} \frac{1}{N} \mathcal{I}(\boldsymbol{x}^N; \boldsymbol{a}^N) \\ &= \sup_{p_{a(a)}} \lim_{N\to\infty} \frac{1}{N} \mathfrak{h}(\boldsymbol{x}^N) - \mathfrak{h}(\boldsymbol{x}^N|\boldsymbol{a}^N) \quad \text{bits/ch.use,} \end{aligned} \tag{2.115}$$

or the same expression with $\boldsymbol{y}$ instead of $\boldsymbol{x}$. The subscript "DT" stands for discrete time. With Gaussian inputs, [67, 69], the capacity in (2.115) is given by

$$C_{\mathrm{DT}} = \frac{1}{2\pi} \int_0^{\pi} \log_2 \left(1 + \frac{\sigma_a^2}{\sigma^2} G(\lambda)\right) \mathrm{d}\lambda \tag{2.116}$$

where

$$G(\lambda) = \sum_k g_k e^{-j\lambda k} = \left| \sum_k f_k e^{-j\lambda k} \right|^2 = |F(\lambda)|^2 \tag{2.117}$$

is the Fourier transform of the ISI sequence $\boldsymbol{g}$, here given in angular frequency. In order to find the constrained capacity of FTN signaling, we need to find $G(\lambda)$ of the corresponding ISI. It can be shown that [49]

$$G(\lambda) = \frac{1}{\tau T} \sum_{k=-\infty}^{\infty} \left| H\left(\frac{\lambda}{2\pi\tau T} + \frac{k}{\tau T}\right)\right|^2 = \frac{1}{\tau T} \left| H_{\mathrm{fo}}\left(\frac{\lambda}{2\pi\tau T}\right)\right|^2. \tag{2.118}$$

Hence, $G(\lambda)$ is proportional to the folded spectrum of $|H(f)|^2$ around the frequency $f = \lambda/2\pi T$. From (2.118) the folded spectrum satisfies

$$|H_{\mathrm{fo}}(f)|^2 = \tau T G(2\pi\tau f T). \tag{2.119}$$

By normalizing the constrained capacity in (2.116) by the signaling rate $1/\tau T$ we get

$$C_{\mathrm{FTN}} \triangleq \frac{1}{\tau T} C_{\mathrm{DT}} \quad \text{bits/s.} \tag{2.120}$$

Finally, inserting (2.118) into (2.116), and performing a variable change, results in

$$\begin{aligned} C_{\mathrm{FTN}} &= \frac{1}{2\pi\tau T}\int_0^{\pi}\log_2\left(1+\frac{P\tau T}{\sigma^2}G(\lambda)\right)\mathrm{d}\lambda \\ &= \frac{1}{2\pi\tau T}\int_0^{\pi}\log_2\left(1+\frac{P}{\sigma^2}\left|H_{\mathrm{fo}}\left(\frac{\lambda}{2\pi\tau T}\right)\right|^2\right)\mathrm{d}\lambda \\ &= \int_0^{1/2\tau T}\log_2\left(1+\frac{2P}{N_0}\left|H_{\mathrm{fo}}(f)\right|^2\right)\mathrm{d}f \quad \text{bits/s} \end{aligned} \tag{2.121}$$

where we have used that $\sigma_a^2 = P\tau T$ and $\sigma^2 = N_0/2$. By setting $\tau = 1$ in (2.121) we obtain the constrained capacity of *orthogonal* or *Nyquist signaling*. If we denote this capacity C_{N} we obtain

$$C_{\mathrm{N}} = \frac{1}{2T}\log_2\left(1+\frac{2PT}{N_0}\right) \quad \text{bits/s} \tag{2.122}$$

where we have used that $|H_{\mathrm{fo}}(f)|^2 = T$ since $G(2\pi\tau Tf)$ in (2.119) equals 1 (no ISI). Under the same assumptions, the following theorem was proved in [49, 52].

Theorem 1. *Unless $h(t)$ is a sinc pulse, there exists τ such that*

$$C_{\mathrm{FTN}} > C_{\mathrm{N}}.$$

For $h(t) = h_{\mathrm{sinc}}(t)$, $C_{\mathrm{FTN}} = C_{\mathrm{N}}$. Hence, by increasing the signaling rate above $1/T$ for non-sinc T-orthogonal pulses, it is possible to achieve a higher constrained capacity than with orthogonal signaling.

Note that by setting $\tau = 1/2WT$ in (2.121), the capacity C_{FTN} is maximized, i.e., it equals the capacity in (2.113). In other words, by signaling at the rate $1/2WT$, no folding of the spectrum occurs in (2.121). For non-sinc $h(t)$ and a signaling rate of $1/T$, the capacity in (2.121) is strictly lower than that in (2.113). Additionally, a smaller τ than $1/2WT$ is meaningless with Gaussian inputs. This is, however, not the case for discrete symbol alphabets. For more FTN capacity results the reader is referred to [49, 70].

2.6 General Linear Channels

In Chapters 4 and 5 of this thesis MIMO channels which constitute a more general class of linear channels, are considered. A discrete-time model of a general linear vector-channel is given by

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{a} + \boldsymbol{w}. \tag{2.123}$$

where $\boldsymbol{y}$ is an $N_r \times 1$ received vector, $\boldsymbol{H}$ is an $N_r \times N_t$, possibly complex-valued, matrix that represents the linear channel and $\boldsymbol{a}$ is an $N_t \times 1$ vector of transmitted data symbols chosen from a constellation Ω. A linear communication channel is characterized by the fact that the output signal, without noise, is a linear mapping of the input signal. Gaussian noise is assumed throughout, i.e., $\boldsymbol{w}$ is an $N_r \times 1$ vector of complex Gaussian noise samples $\boldsymbol{w} \sim \mathcal{CN}(\boldsymbol{0}, \boldsymbol{R_w})$, where $\boldsymbol{R_w}$ is the noise correlation matrix

$$\boldsymbol{R_w} = \mathbb{E}[\boldsymbol{w}\boldsymbol{w}^*]. \tag{2.124}$$

The noise vector $\boldsymbol{w}$ contains colored noise samples if

$$\boldsymbol{R_w} \neq N_0 \boldsymbol{I}_{N_r \times N_r}. \tag{2.125}$$

Many different communication systems can be represented by the discrete-time linear model in (2.123). In this section we will show that they merely differ in the structure of the channel matrix $\boldsymbol{H}$ and the noise correlation matrix $\boldsymbol{R_w}$. Let us begin with ISI channels.

2.6.1 ISI Channels

The single carrier channel in Section 2.2 is a linear channel. The output of its discrete-time representation is a convolution, a linear operation, of the channel impulse response $\boldsymbol{g}$ or $\boldsymbol{f}$ in (2.52) and (2.54) respectively, and the input signal $\boldsymbol{a}$.

Consider now a finite ISI response $\boldsymbol{g}$ (Ungerboeck observation model) of length $2L+1$, i.e. $\boldsymbol{g} = [g_{-L}, \ldots, g_0, \ldots, g_L]$ and assume that there are N symbols in the sequence $\boldsymbol{a} = [a_0, \ldots, a_{N-1}]^{\mathrm{T}}$. For the sampling instances kT, $k = 0, 1, \ldots, N-1$, (2.52) can equivalently be expressed as

$$\boldsymbol{y} = \boldsymbol{G}\boldsymbol{a} + \boldsymbol{\eta} \tag{2.126}$$

where the channel matrix $\boldsymbol{G}$ has the following form

$$\boldsymbol{G} = \begin{bmatrix} g_0 & \cdots & g_L & & \\ g_1^* & g_0 & \cdots & g_L & \\ \vdots & & \ddots & & \\ g_L^* & g_{L-1}^* & & \cdots & g_L \\ & & & \ddots & \\ & & g_L^* & \cdots & g_0 \end{bmatrix} \tag{2.127}$$

and

$$\boldsymbol{a} = \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_{N-1} \end{bmatrix}. \tag{2.128}$$

The dimension of the channel matrix $\boldsymbol{G}$ is $N \times N$ since $N_t = N_r = N$. The noise $\boldsymbol{\eta}$ is a vector containing colored Gaussian noise samples from (2.52). Finally, the noise correlation matrix $\boldsymbol{R_w}$ is a Hermitian matrix where the element at position (i, j) equals $N_0 g_{i-j}$.

Consider now the Forney observation model (2.54) where the noise $\boldsymbol{\eta}$ is whitened. The causal ISI response is given by $\boldsymbol{f} = [f_0, \ldots, f_L]$ so that the discrete-time linear channel model in (2.123) becomes

$$\boldsymbol{y} = \boldsymbol{F}\boldsymbol{a} + \boldsymbol{w} \tag{2.129}$$

where $\boldsymbol{F}$ takes the form

$$\boldsymbol{F} = \begin{bmatrix} f_0 & & & & \\ \vdots & & & & \\ f_L & \cdots & f_0 & & \\ & & \ddots & & \\ & & f_L & \cdots & f_0 \\ & & & & \vdots \\ & & & & f_L \end{bmatrix} \tag{2.130}$$

and $\boldsymbol{w}$ is a vector containing white Gaussian noise (WGN) samples. Since $N_t = N$ and $N_r = N+L$, the dimension of the channel matrix $\boldsymbol{F}$ is $(N+L)\times N$.

In this thesis it is assumed that the ISI sequences $\boldsymbol{g}$ and $\boldsymbol{f}$ are perfectly known at both the transmitter and the receiver. This directly implies that the channel matrix $\boldsymbol{H}$ in (2.123) is perfectly known as well as the the noise correlation matrix $\boldsymbol{R}_{\boldsymbol{w}}$. In the case of deterministic channels where the ISI sequence is completely determined by the modulation pulse $h(t)$, this assumption is reasonable. However, in the case of frequency selective channels $c(t)$, only a good enough estimate of the channel can be expected. In digital subscriber lines (DSL) it is possible to obtain reliable estimates of the communication channel [71], which is not always the case for radio channels.

2.6.2 MIMO Channels

In the previous section, the mathematical model in (2.123) represents a time-sampled sequence, i.e., the elements y_k are sample values of the received signal at different time instances k. In multiple-input multiple-output (MIMO) systems multiple antennas are used at the transmitter (Tx) and the receiver (Rx), providing additional degrees of freedom in one time slot. Now $\boldsymbol{y}$ represents the received signal across the receiving antenna array during a *single* channel use. This idea of using multiple antennas at both the transmitter and the receiver was introduced in [72, 73, 74] and further analyzed in, for example, [75, 76].

The channel matrix element $h_{i,j}$ at position (i,j) in $\boldsymbol{H}$ represents the channel impulse response between receiver antenna i and transmitter antenna j. This is shown schematically in Figure 2.15 for a 2×2 MIMO system. In this thesis only the case with no ISI is considered. A common assumption [77], although not always realistic, is that the elements $\{h_{i,j}\}$ are independent and identically distributed (IID) complex Gaussians, i.e., $h_{i,j} \sim \mathcal{CN}(0,\sigma^2)$. The noise sequence $\boldsymbol{w}$ in (2.123) is then assumed to be a sequence of white Gaussian noise (WGN) samples, i.e., $\boldsymbol{R}_{\boldsymbol{w}} = N_0\boldsymbol{I}_{N_r\times N_r}$. This model is the MIMO counterpart of the Forney ISI-signal (2.54). However, in Chapter 4, we also consider the MIMO counterpart of the Ungerboeck ISI-signal (2.52) given by

$$\boldsymbol{x} = \boldsymbol{H}^{\dagger}\boldsymbol{y} \tag{2.131}$$

where $\dagger$ throughout denotes the Hermitian transpose operator. It is common to impose a constraint on the average energy of the transmitted symbol vector $\boldsymbol{a}$, that is

$$\mathbb{E}[\boldsymbol{a}^*\boldsymbol{a}] \le P_0. \tag{2.132}$$

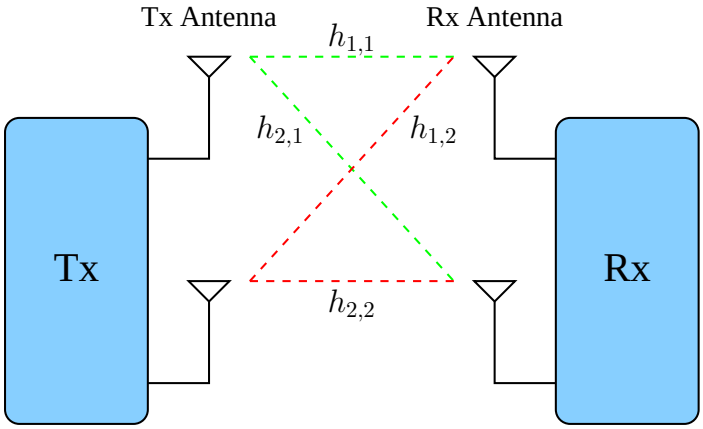

Figure 2.15: System model of a 2×2 MIMO system with transmitter (Tx) and receiver (Rx).

At the receiver, the vector $\boldsymbol{y}$ or $\boldsymbol{x}$ is observed across the Rx antenna array and can now be jointly processed in order to estimate the transmitted symbol vector $\boldsymbol{a}$. However, a MIMO channel depends on the transmission environment which determines how fast the channel changes from one use to another. These channel variations are usually characterized by the so-called coherence time T_C and the coherence bandwidth B_C. If the variations are small, it is possible to obtain reliable estimates of the channel. A popular method is to send known symbols, pilot symbols, from the transmitter to the receiver [78]. After decoding, the receiver obtains an estimate $\hat{\boldsymbol{H}}$ of the true channel $\boldsymbol{H}$. The accuracy of the estimate can be improved by devoting more resources to the training phase but this in general leads to degraded bandwidth efficiency. The two most common techniques for obtaining an estimate of $\boldsymbol{H}$ are *feedback* and *channel reciprocity*. In the feedback technique, $\hat{\boldsymbol{H}}$ is sent from the receiver to the transmitter on a feedback link. If the channel varies rapidly, more frequent feedback of the estimates $\hat{\boldsymbol{H}}$ is required. In channel reciprocity it is assumed that the estimated channel from the transmitter to the receiver, the forward channel, is equivalent to the channel from the receiver to the transmitter, the backward channel. However, in reality the two channels are not necessarily close in time and frequency [77].

Even though perfect channel estimates $\hat{\boldsymbol{H}}$ are difficult to obtain in practice, in this thesis we always assume that $\hat{\boldsymbol{H}} = \boldsymbol{H}$.

2.7 Basic Principles of Turbo Equalization

Figure 2.16 shows a serially concatenated communication system and the corresponding iterative receiver structure for turbo equalization. Turbo equalization was first proposed in [10] for serially concatenated schemes where the mapper together with the ISI channel act as the inner encoder. The method, originally developed for turbo codes (concatenated convolutional codes), is now applied to various communication problems. Note that, even though only serially concatenated schemes are treated in this thesis, the turbo principle [9] can also be applied to parallel concatenations.

The iterative scheme is composed of two constituent blocks, the inner and the outer soft-input soft-output decoders. The inner ISI-decoder is commonly referred to as the equalizer. Additionally, an interleaver (and de-interleaver) rearranges the symbols within a block and in that way decorrelates errors between the nearest symbols. Since the two constituent blocks share the symbol sequence $\boldsymbol{a}$ (input to the inner decoder and a shuffled version of the output from the outer decoder) the idea behind the iterative process is to let the two jointly agree on a final decision on $\hat{\boldsymbol{a}}$, not $\hat{\boldsymbol{u}}$. By exchanging soft information instead of only hard symbol estimates, the BER performance is in general greatly improved. However this usually increases the complexity of the decoding algorithms. The situation is made worse by the need to perform equalization and decoding several times for each data block. In Figure 2.16, and frequently throughout this thesis, convolutional codes are employed for the outer code while the intentional ISI introduced by FTN signaling most often acts as the inner ISI mechanism.

There are other possible detection strategies for the serially concatenated scheme. Optimal MAP/MLSE-based receivers that directly output $\hat{\boldsymbol{u}}$ from $\boldsymbol{r}$ suffer from high computational load since, due to the interleaver, the state space is exponential in the block size. This fact restricts the MAP/MLSE-based approach to rather small block sizes. A non-iterative simple solution would be to first equalize the ISI channel with a hard-output equalizer which produces the estimate $\hat{\boldsymbol{a}}$. Now, in order to obtain a final output $\hat{\boldsymbol{u}}$, the outer decoder uses a de-interleaved version of the estimated sequence, $\Pi^{-1}(\hat{\boldsymbol{a}})$, as input. The main drawback with this method is that the inner decoder generates hard outputs. An obvious improvement is therefore to replace the hard-output inner decoder with a decoder that generates soft values. There are many possible candidates in the literature but an often used method is the MAP-based BCJR algorithm. Note that the outer decoder needs to decode a probabilistic channel in order to produce the final output $\hat{\boldsymbol{u}}$.

In general, optimum and suboptimum MAP-based techniques are used for equalization of the ISI channel. A key objective in this thesis is therefore the

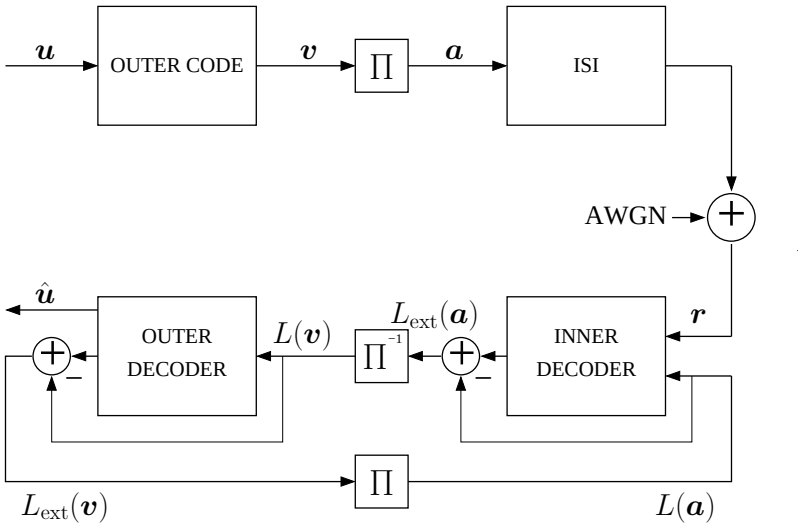

Figure 2.16: Serial concatenated communication system with an iterative receiver performing turbo equalization.

complexity reduction of such algorithms. All approaches use the same iterative structure and vary only in the type of equalizer. The equalizer in Figure 2.16 computes, at each depth k, the APPs $\Pr(a_k = a|\boldsymbol{r})$ where $a \in \Omega$ and $\boldsymbol{r}$ is the received sequence. For simplicity, in this thesis binary PAM (2-PAM) is assumed, i.e., the signal constellation alphabet Ω is $\Omega = \{+1, -1\}$. The extrinsic LLRs, $L_{\text{ext}}(\boldsymbol{a})$, which are fed to the decoder as a priori information, can now be found by subtracting the a priori LLRs, $L(\boldsymbol{a})$, from the a posteriori L-values generated by the equalizer, i.e.,

$$L_{\text{ext}}(a_k) \triangleq \log\left(\frac{\Pr(a_k = +1|\boldsymbol{r})}{\Pr(a_k = -1|\boldsymbol{r})}\right) - \log\left(\frac{\Pr(a_k = +1)}{\Pr(a_k = -1)}\right). \tag{2.133}$$

Note that the a priori LLRs are provided by the decoder but, since there is no a priori information available in the initial iteration, we have that $L(a_k) = 0, \forall k$. The independence assumption (ideal interleaver and large block sizes) together with the concept of treating extrinsic information as a priori are the two main features of any system applying the turbo principle. Extrinsic information is, in the probabilistic domain, generated information about a certain symbol a_k when only accounting for information about the other symbols a_ℓ, $\ell \neq k$.

Now consider the outer decoder. At each depth k, the decoder computes the APPs $\Pr(v_k = a|L(\boldsymbol{v}))$ given only the a priori LLRs $L(\boldsymbol{v}) = \Pi^{-1}(L_{\text{ext}}(\boldsymbol{a}))$. The a priori information is subtracted in order to obtain the extrinsic LLRs

$$L_{\text{ext}}(v_k) \triangleq \log\left(\frac{\Pr(v_k = +1|L(\boldsymbol{v}))}{\Pr(v_k = -1|L(\boldsymbol{v}))}\right) - \log\left(\frac{\Pr(v_k = +1)}{\Pr(v_k = -1)}\right) \tag{2.134}$$

which are then passed to the inner decoder to be used as a priori information. After an initial detection of a received block, the iterative process is repeated a predefined number of iterations (alternatively a suitably chosen termination criterion stops the process). In the final iteration, the outer decoder only computes the data bit estimates

$$\hat{u}_k \triangleq \arg\max_{u_k} \Pr(u_k = u|L(\boldsymbol{v})). \tag{2.135}$$

In this thesis, only the optimal (in terms of BER) MAP symbol detector, realized using the BCJR algorithm, is considered for decoding. Since BCJR equalization for large constellations Ω and/or long ISI responses is too complex to be carried out, various reduced complexity methods based on the same algorithm will be used for the equalizer. Finally it is important to note that a turbo equalization setup with full complexity is reduced complexity compared to the optimal MAP/MLSE detector. More material on turbo equalization can be found in the standard references [79, 80, 81, 82, 83, 84]. For LDPC outer codes, the message-passing algorithm is used.

2.8 EXIT Charts

A popular tool, based on the concept of extrinsic information, for analyzing the behavior and performance of turbo decoding in the fall-off region is the extrinsic information transfer chart, commonly called the EXIT chart. This technique enables the selection of appropriate equalization methods and error correction codes for a given scenario. EXIT charts, developed by Stephan ten Brink [85], plot the mutual information of the component soft-input soft-output decoders in a turbo system where the soft output of one decoder becomes the a priori input of the other. For the next iteration their roles are interchanged. Note that only the extrinsic LLRs are used as output (a priori input value is subtracted from the APP soft output LLR) which avoids propagation of known information. Even though there exist other related methods based on variances and BERs, the powerful EXIT chart technique will be frequently used throughout this thesis.

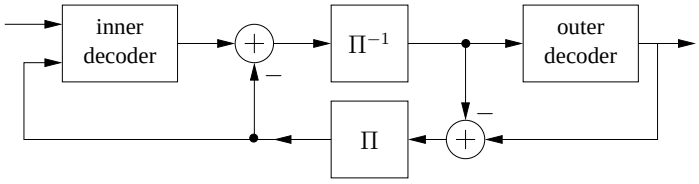

Figure 2.17: Iterative receiver structure.

Now consider serially concatenated systems as depicted in Figure 2.14. The corresponding iterative receiver structure with the component decoders (inner and outer) is shown in Figure 2.17. The EXIT chart technique views the component decoders as non linear LLR transforming elements since the conditional extrinsic LLRs at their input are non linearly transformed into hopefully better quality outputs. Denote now the sequence of extrinsic LLR inputs to each component decoder $L^A_{\text{ext}}(\boldsymbol{a})$ and the corresponding output sequence $L^E_{\text{ext}}(\boldsymbol{a})$. A meaningful iteration implies now that the quality of $L^E_{\text{ext}}(\boldsymbol{a})$ must be better than the quality of $L^A_{\text{ext}}(\boldsymbol{a})$ where quality of $L^{A/E}_{\text{ext}}(\boldsymbol{a})$ is measured by the mutual information $I(L^{A/E}_{\text{ext}}(\boldsymbol{a}); \boldsymbol{a})$. In [85] and based on empirical results, ten Brink suggested that the extrinsic LLRs can be modeled with the following Gaussian distribution

$$L^{A/E}_{\text{ext}}(a[k]) = \mu a[k] + n[k] \tag{2.136}$$

where μ is the mean value and $n[k]$ is an independent Gaussian random variable with variance σ^2 and mean zero. Further a standard assumption is that

$$\mu = \frac{\sigma^2}{2}.$$

By applying the suggested model (2.136), independent (ideal interleaver assumption) a priori extrinsic LLRs $L^A_{\text{ext}}(\boldsymbol{a})$ can be generated. The input-output behavior of each component decoder can now be analyzed. Additionally, the mutual information can be computed as [85]

$$I_A = I(L^A_{\text{ext}}(\boldsymbol{a}); \boldsymbol{a}) = \frac{1}{\sqrt{2\pi}\sigma} \int_{-\infty}^{\infty} e^{-(\lambda-\mu)^2/2\sigma^2} (1 - \log_2(1 + e^{-\lambda}))\, d\lambda.$$

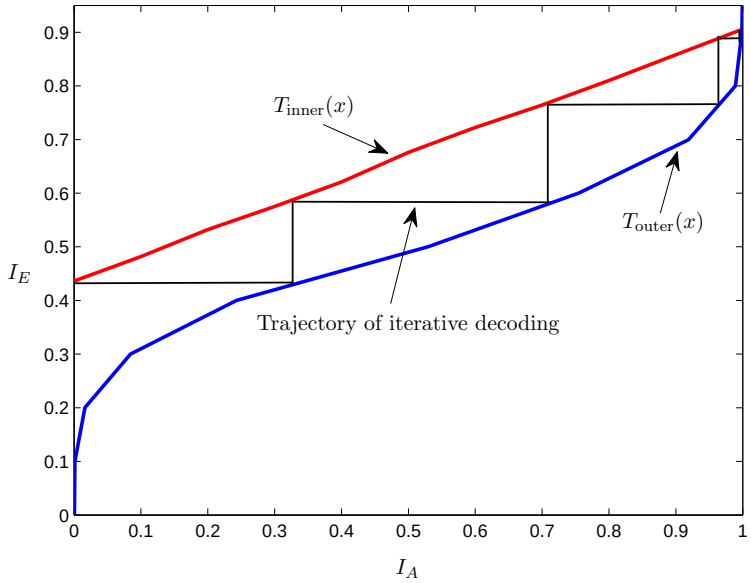

Figure 2.18: EXIT chart example of a serially concatenated system consisting of the outer (7,5) convolutional code and the inner ISI mechanism. The iterative decoding trajectory implies that 4 iterations are performed in the decoder.

Since the output sequence $L^E_{\text{ext}}(\boldsymbol{a})$ is not a sequence of independent Gaussian variables, the computation of $I(L^E_{\text{ext}}(\boldsymbol{a}); \boldsymbol{a})$ is more difficult. There are no analytical formulas for the output extrinsic information at the moment, other than in very simple cases. Consequently, $I(L^E_{\text{ext}}(\boldsymbol{a}); \boldsymbol{a})$ must be found through computer simulations of the component decoders. The computation of I_E proceeds as follows:

1 Generate Gaussian LLRs $L^A_{\text{ext}}(\boldsymbol{a})$ with mutual information $I(L^A_{\text{ext}}(\boldsymbol{a}); \boldsymbol{a}) = I_A$ by following the method proposed in [85].

2 Provide the corresponding received signal to the decoder (only in the case of the inner decoder) and find the output extrinsic LLRs $L^E_{\text{ext}}(\boldsymbol{a})$ using the Gaussian LLRs from Step *1* as input.

3 Estimate the empirical distribution of $L^E_{\text{ext}}(\boldsymbol{a})$ denoted $p_E(\lambda|a)$.

4 Compute the output extrinsic information I_E using the following formula [85]:

$$I_E = \frac{1}{2} \sum_{a=\pm 1} \int_{-\infty}^{\infty} p_E(\lambda|a) \log_2 \left(\frac{2p_E(\lambda|a)}{p_E(\lambda|a=-1) + p_E(\lambda|a=+1)} \right) d\lambda.$$

Note that the output extrinsic information I_E is an empirical function of the component decoder itself, the input extrinsic information I_A and, in the case of the inner decoder, E_b/N_0. This is denoted as

$$I_E = T_{\text{inner}}(x), \quad 0 \leq x \leq 1$$

where the function argument x is the input information I_A. The corresponding notation for the outer decoder is

$$I_E = T_{\text{outer}}(x), \quad 0 \leq x \leq 1.$$

Figure 2.18 shows an EXIT chart example of a serially concatenated system. The two component decoders exchange extrinsic information, which can be seen in the two-dimensional chart. The EXIT function for the inner decoder, $T_{\text{inner}}(x)$, is plotted with its input extrinsic information I_A on the horizontal axis and its output extrinsic information I_E on the vertical axis. The function representing the outer decoder, on the other hand, is plotted with I_A on the vertical axis and I_E on the horizontal axis, i.e., the lower curve is a reflection of $T_{\text{outer}}(x)$. The decoding trajectory can now be followed by stepping between the two curves in the following manner:

$$\begin{array}{lrcccc}
\text{Iteration 1:} & T_{\text{inner}}(0) & = & I_E^{\text{inner}} & = & I_A^{\text{outer}} \\
 & T_{\text{outer}}(I_A^{\text{outer}}) & = & I_E^{\text{outer}} & = & I_A^{\text{inner}} \\
 & & & & & \\
\text{Iteration 2:} & T_{\text{inner}}(I_A^{\text{inner}}) & = & I_E^{\text{inner}} & = & I_A^{\text{outer}} \\
 & T_{\text{outer}}(I_A^{\text{outer}}) & = & I_E^{\text{outer}} & = & I_A^{\text{inner}} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots
\end{array}$$

The iterative process will converge to the intersection point only if there is an open tunnel between the curves and if the number of performed iterations is sufficient. One complete iteration in the chart is represented with the combination of one vertical and one horizontal line. Note that the decoding trajectory predicted by an EXIT chart is only accurate for large block lengths. More complete information on EXIT charts can be found in [86, 87, 88, 89, 90].

Chapter 3

Reduced-Complexity Receivers for Strongly Narrowband ISI Introduced by FTN Signaling

This chapter proposes new M-algorithm BCJR (M-BCJR) algorithms for low-complexity turbo equalization and applies them to severe intersymbol interference (ISI) introduced by faster-than-Nyquist signaling. These reduced-search detectors are evaluated as detectors over the uncoded ISI channel and in iterative decoding of coded FTN transmissions. In the second case, accurate log likelihood ratios are essential and therefore a 3-recursion M-BCJR that provides this is introduced. Focusing signal energy by a minimum phase conversion is also essential; an improvement to this older idea is proposed. The new M-BCJR algorithms are compared to reduced-trellis VA and BCJR benchmarks based on the offset label idea. Additionally, this chapter considers various offset label strategies and chooses the best performing one (in terms of BER) as a benchmark for the proposed M-BCJRs. The FTN signals carry 4–8 bits/Hz-s in a fixed spectrum, with severe ISI models as long as 32 taps. The general conclusion of the chapter will be that the combination of coded FTN and the reduced-complexity BCJR is an attractive narrowband coding method. This chapter is partly based on [91].

3.1 Problem Under Consideration

This chapter investigates the design and complexity of receivers when a convolutionally coded transmission is strongly band limited and the receiver is of the soft-input soft-output type. The modulation method is faster-than-Nyquist (FTN) signaling, i.e., linear modulation with a baseband pulse $h(t)$ according to

$$s(t) = \sqrt{\frac{E_s}{T}} \sum_k a_k h(t - k\tau T), \quad \tau \leq 1 \tag{3.1}$$

where $\{a_k\}$ are binary independent and identically distributed (IID) symbols with zero mean and unit variance, E_s is the average modulation symbol energy, and $h(t)$ is an arbitrary unit energy T-orthogonal pulse. An additive white Gaussian noise (AWGN) channel with noise power spectral density $N_0/2$ follows $s(t)$. Form (3.1), with $\tau = 1$, underlies many practical modulations.

The objective of this chapter is two-fold: To explore iterative receivers for coded narrowband FTN signaling and to find new reduced-complexity BCJR algorithms for use in iterative decoding. The new algorithms follow the well-known M-algorithm idea, meaning that the BCJR recursions are based only on the M dominant terms at each trellis stage. In an iterative scheme log likelihood ratios (LLRs) are passed around between two component decoders based on the BCJR algorithm. The quality of the LLRs strongly affects performance. Unfortunately, the M-algorithm degrades the LLR quality. However, signals can be pre-processed to make better use of the retained M terms in the BCJR recursions. The contributions of this chapter are improved minimum phase modeling, new BCJR and M-BCJR algorithms that produce high quality LLRs, together with test results for coded FTN. The outcome is receivers for narrowband coding that work reasonably close to the optimal receiver with practical complexity.

Since FTN signals are continuous, a receiver contains a matched filter, sampler and possibly a post filter, which together reduce the signaling to a discrete-time convolution of the data $[a_0, a_1, \ldots]$ (binary in this chapter) with the ISI tap set $\boldsymbol{f} = [f_0, f_1, \ldots, f_L]$. This provides a 2^L-state trellis for the channel. Suitable receiver models are derived in Section 3.2, which have the property that zero-mean IID Gaussians with variance $N_0/2$ are added to the discrete convolution values. The FTN signaling is applied in two ways, by itself as an uncoded narrowband communication system and as the inner ISI mechanism in a coded system with iterative decoding. These are shown schematically in Fig. 3.1. The first embraces the elements in the dashed box, and will be referred to as "simple detection" of ISI while the second is turbo equalization [10].

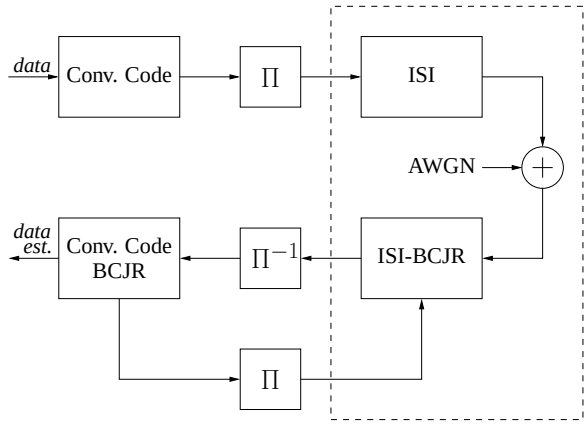

Figure 3.1: Turbo equalization with a simple detection inner coder (dashed box).

Since the intentional ISI introduced by FTN with small τ is severe, it is necessary to reduce the complexity of the ISI-BCJR block in Fig. 3.1. Coded FTN offers an attractive combination of bandwidth reduction and coding gain, but a reasonable receiver is needed. Reduced complexity can be achieved in two basic ways: By reducing the state size of the model $\boldsymbol{f}$, a *reduced-trellis* approach, or by reducing the search of a given trellis, a *reduced-search* approach. This chapter mainly focuses on the second and uses the first as a performance benchmark. In Chapter 5 on the other hand, optimal channel shortening detectors that belong to the first class, are considered. Early work with reduced search decoders primarily treats non-iterative applications where log likelihood ratios are not needed. However, in iterative detection the decoder needs to produce soft information about the symbols. An efficient but impractical (for large L) implementation is the well-know BCJR algorithm [11]. A selection of papers on M- or other reduced complexity BCJRs is [13, 92, 93, 94, 95, 96]. A factor graph based approach has been presented in [97].

It has been known for some time that the receiver front end processing should provide the reduced-search detector with a minimum phase input (see [92, 98] and the more recent [96, 99]). A straightforward solution is to cascade the matched filter/sampler by an all-pass filter that produces a max phase output, and then reverse the output frame. Minimum phase moves energy to the front of the ISI model, which directs the reduced search more efficiently (mini-

mum phase will not improve a full 2^L-state Viterbi algorithm (VA) or BCJR). Energy focusing also aids reduced-trellis decoders, and in order to have a fair benchmark it will be employed there too. Section 3.2 discusses the minimum phase concept further, and proposes a novel extension to it that focuses energy in a more favorable way than simply calculating the mathematically correct minimum phase model.

An important role in this chapter is played by the normalized minimum squared Euclidean distance $d^2_{\min}$ between two signals of form (3.1). The error probability of maximum likelihood simple detection of the $\{a_k\}$ tends in logarithm to $m_{\boldsymbol{e}}Q(\sqrt{d^2_{\min}E_s/N_0})$, asymptotically in the ratio E_s/N_0, where $m_{\boldsymbol{e}}$ is a multiplicity factor that depends on the most likely error events and whether bit error rate (BER) or event error rate (EER) is of interest. A union bound estimate formed from events at several distances near $d^2_{\min}$ is a good estimate for simple detection of ISI at moderate to high E_s/N_0. Some details on bounding and distance finding are given in Section 2.2 but more complete results appear in coding texts, e.g. ref. [20]. As the state size of a reduced-complexity VA or BCJR algorithm gets reduced, its error rate will at some point depart from this ML estimate. Distance-based estimates are thus essential for deciding the minimum required size of an algorithm. For binary $\{a_k\}$, $d^2_{\min} \leq 2$, and the special case $d^2_{\min} = 2$ means that the ISI-affected signal can be detected with the same log error probability as antipodal signaling except for the factor $m_{\boldsymbol{e}}$; that is, asymptotically the effect of the ISI can be removed.

For encoding, this chapter only considers convolutional codes. Above a certain E_s/N_0, called the threshold, it can be shown that the iterations converge to the BER of the convolutional code alone over an antipodal (ISI free) channel with the same E_s/N_0. Below this threshold, convergence is to a much higher BER.

According to Chapter 2 the BCJR algorithm consists of forward and backward linear recursions, instead of the VA's unidirectional add-compare-select. As such, its behavior is rather different from the VA's. There is also a major difference between an algorithm that calculates full LLRs and one that makes decisions about bits, i.e., calculates the LLR *sign*. Accurate LLRs are essential in iterative decoding of FTN signals and producing them is a considerable challenge for the M-algorithm. Earlier work on this subject appears in [96]. However, it is concluded here that the key to high quality LLRs is to add a third, low complexity recursion.

3.1.1 Three Reasons for FTN

The algorithms in this paper apply to general ISI, but the intentional ISI introduced by FTN signaling is interesting for several reasons.

(*i*) It is severe, meaning that it has a combination of large state space, small $d^2_{\min}$, and z-plane zeros on or near the unit circle. It is difficult to assess the effect of complexity reduction unless there is significant complexity to reduce. In Sections 3.4 and 3.5 it is found that many of the ISI models in the literature provide a too easy target for reduction.

(*ii*) FTN is also interesting for theoretical reasons. According to Section 2.5, the signals have a fixed PSD shape, given by the Fourier transform of $h(t)$ in (3.1). Such signals have a Shannon *constrained capacity* for the PSD, given by

$$\int_0^\infty \log_2\left(1+\frac{2P|H(f)|^2}{N_0}\right)\,\mathrm{d}f,$$

where P is the total power and $P|H(f)|^2$ is the signal PSD. In general, this capacity *cannot be reached by codes based on orthogonal pulses* with PSD $P|H(f)|^2$, such as the coded modulations and turbo codes in common use. Recently it has been shown that asymptotically as $\tau \to 0$, this capacity can be achieved with *binary* FTN signals [53]. Studies of best convolutional codes have demonstrated that M-BCJR iterative decoders reach to 1–2 dB from this PSD capacity, with complexity and block length comparable to other iterative decoding [100].

(*iii*) A third reason for FTN signals is that they provide a proper experimental design for narrowband signaling. This chapter explores the behavior of reduced BCJRs as the bandwidth gets reduced and decoding becomes more complex. With narrowband signals receiver error performance is sensitive to the entire shape of the signal PSD, not just to a measure like 3 dB bandwidth. Minimum distance studies show [20] that removal of only a small power from the outer spectrum can change the minimum distance of a signal set significantly; this is the "escaped distance" problem ([20], Chapter 6). These effects grow more pronounced as the bit density carried in bits/Hz-s grows. If a small extra power appears in the stopband—for example, through too-early truncation of the model $\boldsymbol{f}$—receiver error rate can improve, and give a false test result for that model. FTN signals provide a way to increase the transmission bit density, by reducing τ, while maintaining the same PSD shape.

3.1.2 The Choice of FTN Pulse Shape

Although any FTN pulse shape $h(t)$ could be taken, in this chapter $h(t)$ is the unit-energy root raised-cosine (rRC) T-orthogonal pulse with 30% excess bandwidth. Its spectrum is zero outside $\pm 1.3/2T$ Hz. Setting $\tau = 1$ gives the widely used rRC orthogonal pulse. As τ drops below 1, pulses are sent "faster" but the PSD shape remains the same, namely, a raised cosine. The bit density in the uncoded case is $2/\tau$ data bits/Hz-s (taking 3 dB bandwidth as a scale unit). The asymptotic error rate remains $Q(\sqrt{2E_s/N_0})$ for $\tau \geq .703$, the Mazo limit. Thereafter, it is $\approx Q(\sqrt{d^2_{\min} E_s/N_0})$, where $d^2_{\min}$ declines with τ. The Mazo limit itself depends on the pulse excess bandwidth, i.e., $d^2_{\min}$ will fall below 2 at a different τ. Optimizing $h(t)$ within a suitable framework is an interesting future topic but some work on this subject is given in Section 3.7.

The remainder of this chapter is organized as follows. Section 3.2 presents a suitable receiver front end and an improved discrete-time model, that yield white noise and easy control of spectrum and minimum phase. In Section 3.3 benchmark offset VA and BCJR for severe ISI are considered. Sections 3.4 and 3.5 present and evaluate novel M-BCJR algorithms for simple detection and iterative decoding. Other M-BCJR branching strategies are presented in Section 3.6 while Section 3.7 considers the FTN pulse excess bandwidth. Section 3.8 summerizes the chapter.

3.2 Generating Discrete-Time System Models

The conversion of continuous FTN signals to discrete time is considered in this section. Many methods are possible, and by choosing one the discrete-time signal model $\boldsymbol{f}$ seen by the detector/decoder is created. When choosing a method for this chapter, three priorities are considered: Signals with spectral zero regions must be handled in an accurate, straightforward way, noise at the detector input should be white, and the model should be minimum phase.

In this chapter, the following model of the conversion to discrete time (henceforth called "conversion/model") is adopted. This model was introduced in [101]. It assumes linear modulation by $h(t)$ at rate $1/\tau T$ and an AWGN channel, and then processes the signal according to Figure 3.2.

The filter $B(z)$ creates a maximum phase output, which is reversed block-wise to form a minimum phase output. The matched filter is matched to some pulse $\phi(t)$ and sampled at the *faster* rate $1/\tau T$. Let $\{\phi(t - j\tau T)\}$, j an integer, be an orthonormal basis for $h(t)$, such that

$$h(t) = \sum c_j \phi(t - j\tau T) \tag{3.2}$$

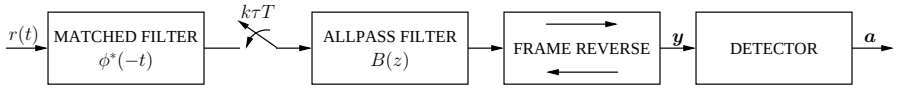

Figure 3.2: Model of the conversion to discrete time.

where

$$c_j = \int h(t)\phi(t - j\tau T)\,\mathrm{d}t. \tag{3.3}$$

The basis pulse $\phi(t)$ is chosen so that $\{c_j\}$ are the energy-normalized samples $h(j\tau T)$ of $h(t)$.[1] The pulse $h(t)$ is infinite-response and time-symmetric, and there is a J such that $\boldsymbol{c} = \{c_j\}$, $j = -J, \ldots, J$ will capture all but δ of the pulse energy, any $\delta > 0$. Since the $\{\phi(t - j\tau T)\}$ are τT-orthonormal, the matched filter samples satisfy two important properties:

- Filtered noise samples are white Gaussian random variables
- Euclidean distance between two noise-free continuous signals from (3.1) can be calculated from their samples.

Two other well-established conversion/models in the literature are the whitened matched filter (WMF) model (also known as the Forney model) and the Ungerboeck model. In the Ungerboeck observation model, the receive filter is matched to $h(t)$ and sampled each τT. There follows no whitening; instead a special detector works with colored noise. A BCJR algorithm for the Ungerboeck model was explored in [34]. The sampler creates a discrete time model of the channel and the FTN and its outputs $\boldsymbol{x}$ are sufficient statistics for estimating $\boldsymbol{a}$. According to Section 2.2, they satisfy $\boldsymbol{x} = \boldsymbol{a} \star \boldsymbol{g} + \boldsymbol{\eta}$; expressed through z-transforms this is

$$X(z) = A(z)G(z) + N(z). \tag{3.4}$$

Here $\boldsymbol{g}$ is the sampled autocorrelation function of $h(t)$,

[1] A condition for this is that the Fourier transform of $\phi(t)$ is constant over the bandwidth of h.

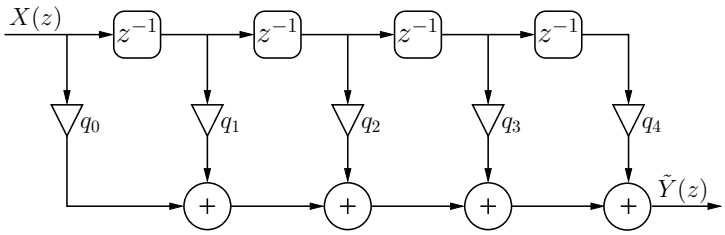

Figure 3.3: An example of a 4th-order FIR filter.

$$g_k = \int h(t)h(t + k\tau T)\,\mathrm{d}t \tag{3.5}$$

and $\boldsymbol{\eta}$ is colored Gaussian noise with correlation sequence $\boldsymbol{g}N_0/2$.

The WMF receiver filter is also matched to $h(t)$ and sampled each τT. However, there follows a whitening filter. The whitening filter decorrelates $\boldsymbol{\eta}$ and is constructed from $\boldsymbol{g}$ by spectral factorization of $G(z)$ into $V(z)V^*(1/z^*)$; for details see Section 2.2.1 and [17, 49]. After whitening by the filter $1/V^*(1/z^*)$, what remains can be expressed as $\tilde{\boldsymbol{y}} = \boldsymbol{a} \star \boldsymbol{v} + \boldsymbol{w}$ or in z-transform

$$\tilde{Y}(z) = A(z)V(z) + W(z) \tag{3.6}$$

where $\boldsymbol{w}$ is white Gaussian noise with variance $N_0/2$. The so-called WMF model of the channel is $V(z)$, and $\boldsymbol{v}$ represents causal ISI with the property $\boldsymbol{v}[n] \star \boldsymbol{v}[-n] = \boldsymbol{g}$.

In fact, many spectral factorizations are possible. Since $\boldsymbol{g}$ is a correlation, the factorization can take place such that $V^*(1/z^*)$ has zeros within the unit circle; the whitener $1/V^*(1/z^*)$, implemented as a finite impulse response (FIR) filter, is thus stable and the channel model becomes $V(z)$ with all zeros *outside* the unit circle. In signal processing, a FIR filter is a filter whose impulse response is of finite duration, i.e., it settles to zero in finite time. FIR filters can be both discrete-time and continuous-time as well as digital or analog. The impulse response of an Nth-order discrete-time FIR filter lasts for $N + 1$ samples before it settles to zero. A 4th-order FIR filter is illustrated in Figure 3.3.

The model $V(z)$ above is in fact the maximum phase model for $\boldsymbol{g}$, which is a strong inconvenience for reduced decoders. However, it can effectively be converted to a minimum phase model by decoding the signal blocks backwards.

We thus can construct a practical whitener and minimum phase discrete model provided that there exists $V(z)$ with all zeros outside the circle. This is, however, often not directly possible with FTN signaling for a fundamental reason. Important practical pulses $h(t)$, such as the root raised cosine (root RC), have spectrum equal to zero outside a certain bandwidth. For example, the root RC pulse with excess bandwidth β is zero outside $(1+\beta)/2T$ Hz. In FTN signaling at the higher rate $1/\tau T$, this value decreases in comparison to the folding frequency $1/2\tau T$, and there will eventually be a null zone in the range $((1+\beta)/2T, 1/2\tau T)$ Hz. This prohibits the Forney observation model; the ISI $\boldsymbol{v}$ can at most synthesize a countable number of frequency nulls. The spectrum $|H(j2\pi f)|^2$ is $|G(e^{j2\pi f})|$, and thus a finite order $G(z)$ can place spectral zeros at only finitely many frequencies.

Many practical FTN cases fall into this difficulty. One solution is to find a finite $G(z)$ approximation with quartets of zeros on the unit circle. The zeros must occur in quartets because $V(z)$ and $V^*(1/z^*)$ each require a conjugate pair. The model may then be refined by splitting the quartet of zeros so that one conjugate pair is slightly inside the circle and one is outside. Note that the adopted conversion/model, illustrated in Figure 3.2, does not fall into this difficulty. The received noise samples are already white, hence there is no need for whitening.

Chapter 4 analyzes the effect of Ungerboeck and Forney metrics on the BER performance for receivers of the M-algorithm type.

3.2.1 Improving the Minimum Phase Model

Turning to the allpass $B(z)$, in the first instance the $B(z)$ that makes $\boldsymbol{f}$ maximum phase is sought. Allpass filters affect neither the statistics of the noise (it is still white) nor the minimum distance of a signal set ([20], Chapter 6). This is true for *any* allpass. Maximum phase is achieved by a particular $B(z)$, the one that reflects outside the unit circle the zeros $\{z_i\}$ of $C(z) = \sum c_j z^{-j}$ that lie inside the circle; that is, the poles of $B(z)$ lie at $\{z_i\}$ and the zeros lie at $\{1/z_i\}$. Zeros of $C(z)$ on the unit circle are not reflected.

With a reduced-complexity detector, there in fact exist $B(z)$ that improve the error rate even more than the mathematically correct $B(z)$. Reduced-complexity algorithms need a steep energy growth in the model taps $\boldsymbol{f}$. Suppose that $B(z)$ produces a more rapid growth, but also a length-K_p low-energy precursor. Since the precursor energy is low, the algorithm can ignore it with almost no effect, i.e., it can work with a $\boldsymbol{f}$ whose first K_p taps are set to zero. Consequently the detector is slightly mismatched to the true channel model. The key issue is: For a given complexity does the better performance exceed the loss from the mismatch. This is a major optimization problem. However,

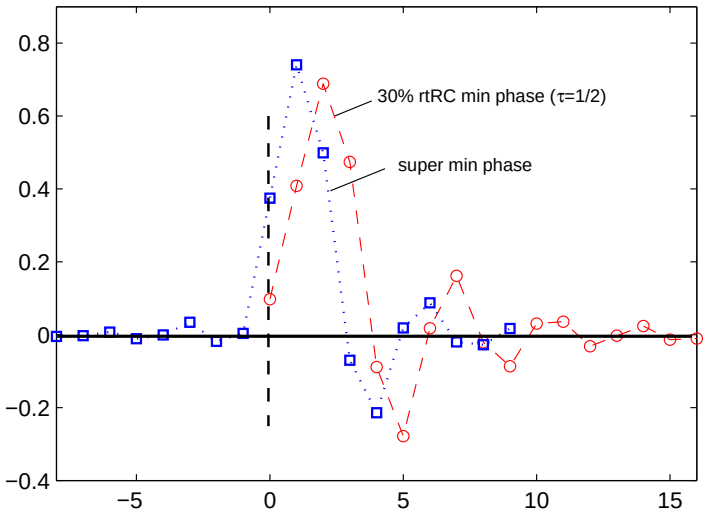

Figure 3.4: Illustration of super minimum phase modeling at FTN $\tau = 1/2$. Mathematically correct minimum phase response $\boldsymbol{f}$ based on 61 pulse samples (circles); super minimum phase response (squares).

this section lists specific $B(z)$ found through extensive searching. An M-BCJR algorithm working with a model $\boldsymbol{f}$ generated by these $B(z)$ can achieve the same bit error rate with 2–4 times smaller M. Such an improved model will be called *super minimum phase*.

The super minimum phase $B(z)$ leads to significant BER improvements. The physical decoder/detector remains the same, except that it runs K_p stages behind the present trellis stage and it computes branch labels ignoring the precursor. This K_p-delayed decoding is an essential element of the super minimum phase method.

An illustration of minimum and super minimum phase models $\boldsymbol{f}$ for the 30% rRC FTN pulse stretched in time by $\tau = .5$ is now given. The central pulse samples are (note that these are not plotted in Figure 3.4)

$$\{c_j\} = \{h(j\tau T)\} = \{\ldots, .040, -.109, -.053, .435, .765, \\ .435, -.053, -.109, .040, \ldots\} \tag{3.7}$$

The maximum phase conversion of 61 of these, reversed, is plotted in Fig. 3.4. Now consider only the center-most 9 samples, the ones between the dots in (3.7). With only these, the reversed max phase conversion is

$$\{.375, .742, .500, -.070, -.216, .014, .077, -.032, .004\} \tag{3.8}$$

and the $B(z)$ that creates it is

$$B(z) = \frac{.107 - .561z^{-1} + z^{-2}}{1 - .561z^{-1} + .107z^{-2}} \ . \tag{3.9}$$

Even though the energy of the new model (3.8) rises faster it lacks the required PSD. If the full model (3.7) is instead filtered by the $B(z)$ from (3.9), the outcome will have the correct spectrum, since $B(z)$ is an allpass. The significant parts of the outcome are plotted (squares) in Fig. 3.4. The values at times $0, \ldots, 8$ are nearly identical to (3.8); what is added is a precursor and the values at $9, 10, \ldots$. The latter points will not affect the M-BCJR complexity. The precursor however increases its complexity but if it can be ignored without damaging the error performance, this new $B(z)$ will be a superior allpass because it leads to a faster rise of the main ISI model energy.

Figure 3.5 plots the improved FTN models $\boldsymbol{f}$ presented to the receiver processor for the main tests in this chapter. The unit-energy models for $\tau = .703, .5, .35, .25$ are respectively

$$\begin{aligned} \boldsymbol{f} \ = \ & [.\mathbf{553}, .\mathbf{793}, -.\mathbf{084}, -.\mathbf{171}, .\mathbf{154}, -.\mathbf{064}, .\mathbf{006}, .\mathbf{010}, -.\mathbf{012}, .\mathbf{015}, \\ & -.\mathbf{016}, .\mathbf{013}, -.\mathbf{008}] \end{aligned} \tag{3.10}$$

$$\begin{aligned} \boldsymbol{f} \ = \ & [-.005, -.003, .007, -.011, -.001, .034, -.019, .003, .\mathbf{375}, .\mathbf{741}, \\ & .\mathbf{499}, -.\mathbf{070}, -.\mathbf{214}, .\mathbf{019}, .\mathbf{087}, -.\mathbf{020}, -.\mathbf{028}, .\mathbf{017}] \end{aligned} \tag{3.11}$$

$$\begin{aligned} \boldsymbol{f} \ = \ & [.025, .012, -.024, .008, .\mathbf{191}, .\mathbf{464}, .\mathbf{623}, .\mathbf{506}, .\mathbf{176}, -.\mathbf{123}, \\ & -.\mathbf{196}, -.\mathbf{075}, .\mathbf{060}, .\mathbf{080}, .\mathbf{013}, -.\mathbf{035}, -.\mathbf{022}] \end{aligned} \tag{3.12}$$

$$\begin{aligned} \boldsymbol{f} \ = \ & [-.010, -.013, -.007, .005, .011, .004, -.008, .001, .\mathbf{060}, \\ & .\mathbf{181}, .\mathbf{339}, .\mathbf{473}, .\mathbf{520}, .\mathbf{443}, .\mathbf{262}, .\mathbf{047}, -.\mathbf{120}, -.\mathbf{182}, -.\mathbf{138}, \\ & -.\mathbf{037}, .\mathbf{055}, .\mathbf{092}, .\mathbf{070}, .\mathbf{018}, -.\mathbf{025}, -.\mathbf{037}, -.\mathbf{021}, .\mathbf{003}, \\ & .\mathbf{016}, .\mathbf{012}, .\mathbf{0004}, -.\mathbf{008}]. \end{aligned} \tag{3.13}$$

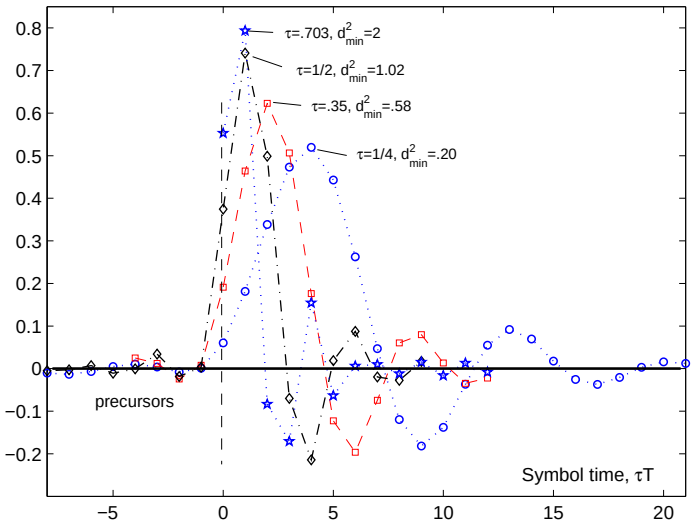

Figure 3.5: Improved unit-energy discrete-time channel models, as seen by the ISI equlizer. FTN $\tau = .703, 1/2, .35, 1/4$.

The precursor values are written in italic in (3.11)–(3.13); all detectors replace these with zeros and work at a delay K_p. The first τ is the Mazo limit for the 30% rRC $h(t)$. The last three models are super minimum phase, with the allpass filter found from a search among $B(z)$ obtained from truncations of $h(t)$. Note that they have taps in the pattern [low energy precursor] + [high energy part] + [long decaying tail]. Insignificant taps before and after have been removed.[2] The test of whether too many taps have been dropped is the model spectrum, and these are plotted for each τ in Figure 3.6. Compared to the ideal rRC spectra, spectral sidelobes must appear, but these are down at least 30 dB. Models with sidelobes down only 15–20 dB can have significantly better minimum distance than the true FTN signals, and the receiver will show an artificially low bit error rate.

The $\tau = .5$ case generates mild ISI and a 50% bandwidth reduction; $\tau = .35$ is severe ISI and a reduction to $\approx 1/3$; the .25 case is extreme ISI and a reduction to 1/4. The signal sets created by these ISIs have square minimum distances of 1.02, .56 and .20, which are energy losses of 2.9, 5.5 and 10.0 dB compared to antipodal signaling.

[2] In tests the transmitted signal generation uses a few extra small taps, as insurance that the PSD is maintained.

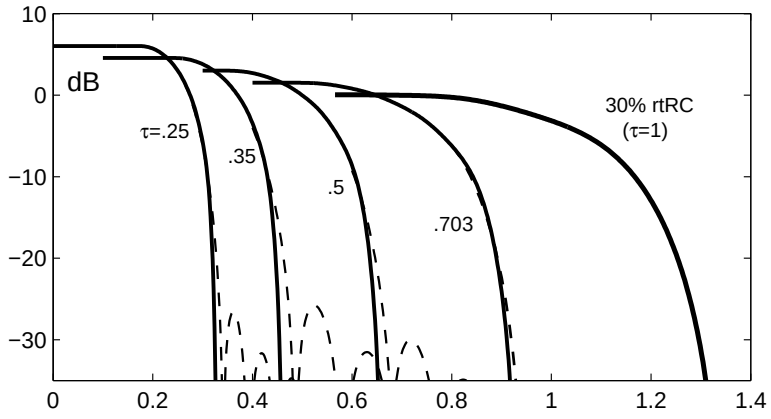

Figure 3.6: Spectra of the channel models (dashed), compared to ideal 30% root RC spectra (solid); FTN $\tau = 1/4, .35, 1/2, .703, 1$. X-axis is $2fT$.

3.2.2 Other ISI Models

The following non-FTN discrete-time models from the literature will be used to compare with earlier work. They are much simpler, and the proposed M-BCJR will need to pursue only 2–3 paths to achieve near-ML performance. The model

$$\boldsymbol{f} = [\sqrt{.45}, \sqrt{.25}, \sqrt{.15}, \sqrt{.1}, \sqrt{.05}] \tag{3.14}$$

features in early turbo equalization papers [10, 93]. It is minimum phase and $d^2_{\min} = 1.12$; the asymptotic VA equalizer EER is $\approx .5Q(\sqrt{1.12E_s/N_0})$. The model

$$\boldsymbol{f} = [.1762, .3163, .4765, .5326, .4765, .3163, .1762] \tag{3.15}$$

appears in several papers [102]; it is minimum phase and has $d^2_{\min} = .2616$. This tap set is said to have the least $d^2_{\min}$ of any $L = 6$ set with binary input. The model $[1, 0, 1, 2, 1, 0, 1]/\sqrt{8}$ was studied in [96]. It has $d^2_{\min} = 2$. In this chapter it will appear in its minimum phase form, which is

$$\boldsymbol{f} = [.670, .366, .178, .443, .379. - .102, .187]. \tag{3.16}$$

Note that the super minimum phase idea is not useful with these short models.

3.3 Reduced-Trellis Benchmarks: The Offset BCJR and Viterbi Algorithms

This section sets up *reduced-trellis* benchmark detectors based on the full VA or BCJR, applied here to simple detection of uncoded ISI. Section 3.5 compares the benchmark error performance to the proposed *reduced-search* M-BCJR methods. Finding a fair benchmark is challenging and in fact, considerable trellis reduction is possible without significant error rate loss. Since algorithms that process reduced trellises are quite simple, the M-BCJR state search needs to be small in order to compete. The goal of this section is to distinguish the two types of complexity reduction, and see how they compare in FTN. Further requirements for a benchmark are that it must work within the constraints of Section 3.2, which are white noise and error performance implied by the full signal set $d^2_{\min}$. The main result in this section is a competitive offset-based benchmark BCJR which associates a single offset state with all its main states. A modified method for retaining backward recursion values makes this reduced-trellis BCJR a fair benchmark for the proposed M-BCJRs in Section 3.4. The concept of main and offset states is defined next.

A key to reducing the state space of the VA or BCJR is to favor high-energy model taps, if it can be done simply, without increasing the error rate. It is assumed that the algorithms are preceded by the Section 3.2 conversion/model, so that the model $\boldsymbol{f}$ is minimum/super minimum phase, with energy focused near the present symbol. The following *offset receiver* will then reduce the complexity induced by the low-energy tail: Instead of generating trellis branch labels ℓ (at trellis stage k) as

$$\ell = \sum_{j=0}^{L} f_j a_{k-j} \tag{3.17}$$

where the total memory L is the sum of the high energy and long tail lengths and symbols $\ldots, a_{k-1}, a_k$ are the symbols following the precursor, form them instead from

$$\ell = \sum_{j=0}^{m} f_j a_{k-j} + \sum_{j=m+1}^{L} f_j a_{k-j}. \tag{3.18}$$

Symbols $a_{k-m}, \ldots, a_{k-1}$ comprise the size-m reduced VA/BCJR *main state*, and stem from high energy symbols, while the $a_{k-L}, \ldots, a_{k-m-1}$ comprise the *offset state*. The second term is an offset to the label ℓ created by early symbol history. A set of offset symbols can be associated with each main state but its symbols do not form part of the algorithm's state variable. However, all $L+1$

taps contribute to a label. In the add-compare-select step of the benchmark offset VA, the offset states of the survivors, together with the oldest main state bit, become the offset states for each new main state. Trellis searching focuses on high energy taps while small taps contribute only to the labels.

This sort of trellis reduction was devised in the 1970s [103] as a way to handle large state spaces of long-response systems, and was applied to ISI problems by several authors in the 1980s. In the best known [30], Duel-Hallen and Heegard calculate $d^2_{\min}$ for the VA receiver as a function of the main state size m. Studies of the VA then and now [99] show that under narrowband ISI a large truncation is possible without significant loss in $d^2_{\min}$. Offset BCJR receivers have been studied since the mid 1990s, although not for narrowband ISI. A major work is Colavolpe *et al.* [93], which gives a full list of references.

A different strategy to reduce trellis size is to add non-allpass prefiltering to the conversion/model. Even though they appear promising [93, 104, 132], these methods color the noise and reduce $d^2_{\min}$, and are not explored further. Ref. [104] presents error rate results, which are used for comparison in the sequel.

3.3.1 The Benchmark VA

The benchmark VA performance is considered next. The offset VA is quite different from the BCJR and its benchmark achieves near-optimal error performance with memory m 1–2 smaller under severe ISI. First considered is the $\tau = .5$ FTN case in Figure 3.7. The offset VA is the standard kind [30], which associates an offset state with each main state. The test setup is: Size $N = 800$ frames of random ± 1 data, with enough frames to give 40–100 error events. Note that errors occur in groups called events which consist of a number of related bit errors. The frames are terminated before and after by L '+1' symbols. The VA output symbol is decided $L+35$ symbols before the present trellis stage. Error events are taken to begin when the receiver output state splits from the transmitter state path and to end after 5 output data are correct. BER is about 3 times EER at higher E_s/N_0 and 4–5 times at lower.

The objective is to find the smallest m that leads to essentially ML performance. The solid curves plot error event rates (EER) for the super minimum phase model (3.11) at main state memories 2,4,6; it is clear that 2^4–2^6 states are needed, and consequently this benchmark state size is about 32. The dotted curves show the offset VA with the mathematically correct minimum phase model derived from (3.7) for the same memories m. Clearly this modeling is worse than the super minimum phase, especially for short memories.

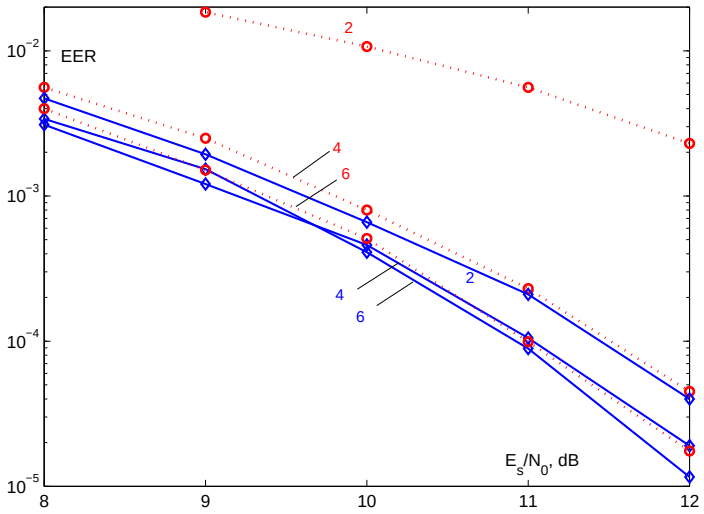

Figure 3.7: Error event rates versus E_s/N_0 for the offset VA with mathematically correct (dots) and super (solid) minimum phase channel models. FTN signals with $\tau = 1/2$. Main state memory 2, 4, 6.

3.3.2 The Benchmark BCJR Algorithm

Consider now the benchmark BCJR algorithm based on the offset-label idea. Recursions (2.96)–(2.97) are applied to the main state in (3.18), computing the 2^{m+1} branch labels while exploiting the second-term label offsets (similarly to [93]). In the case of long narrowband ISIs, stemming from practical FTN signaling, it is observed that certain changes to the offset label computation improve performance. In the interests of a fair benchmark comparison, they are described next.

Offset Label Strategies

The heart of the BCJR is the trellis branch metric γ_k. According to (2.94) the branch metric of the branch connecting the states (σ_i, σ_j) can, in the case of the Forney model, be expressed as

$$\begin{aligned} \gamma_k(\sigma_i, \sigma_j) &= p(\text{ISI in state } \sigma_i \text{ at time } k, y_k \mid \text{ISI in state } \sigma_j \text{ at time } k-1) \\ &= \Pr(a_k) p(y_k|\boldsymbol{a}) \end{aligned} \tag{3.19}$$

where y_k is the kth channel output from (2.54) and $\Pr(a_k)$ is the a priori probability of the symbol a_k. By combining (2.87), (3.18) and (3.19) we obtain the following alternative expression of the branch metric:

$$\begin{aligned}\gamma_k(\sigma_i,\sigma_j) &\propto \Pr(a_k)\exp\Big(-\frac{1}{N_0}\Big|y_k-\ell_{i,j}\Big|^2\Big)\\ &\propto \Pr(a_k)\exp\Big(-\frac{1}{N_0}\Big|y_k-\underbrace{\sum_{j=0}^{m} f_j a_{k-j}}_{\text{“main''}}-\underbrace{\sum_{j=m+1}^{L} f_j a_{k-j}}_{\text{“offset''}}\Big|^2\Big).\end{aligned} \tag{3.20}$$

Its elements contribute whenever a label $\ell_{i,j}$ is close to a received sample y_k. Whereas the VA “picks winners”, continually dropping path segments that fall short, the BCJR counts every contributing region of the trellis. Hopefully, using a reduced main state, correct regions can be pointed out and an accurate LLR computed. In the case of narrowband ISI, the labels $\ell_{i,j}$ in both the VA and BCJR depend strongly on *both* the main and the offset states. In fact, since the ISI model taps corresponding to the offset states are rather small, only a reasonable approximation of the latter is needed. Despite this, our tests indicate that the offset label must be present.

After the frame reversal in Figure 3.2, the application scenario for ISI is as follows. The forward recursion α is taken to be the one proceeding left to right in the direction of time, the direction in which the ISI model phase is minimum. At the extension to trellis stage $k+1$, the alignment of the ISI model taps, state symbols and forward metrics, αs, is

$$\begin{array}{rcl} f_L,\ldots\ldots,f_1, & f_0 & \\ \ldots,a_{k-L},a_{k-L+1},\ldots,a_{k-1}, & a_k & \longrightarrow \quad \text{extension forward} \\ \ldots, \boldsymbol{\alpha}_k, & \boldsymbol{\alpha}_{k+1} & \end{array}$$

with the symbols corresponding to the main state and the focus of ISI model $\boldsymbol{f}$ energy to the right and where the notation $\boldsymbol{\alpha}_k=[\alpha_k(\sigma_1),\alpha_k(\sigma_2),\ldots]$, i.e., the components run over the states. The main state at trellis depth[3] k is defined by the symbols at stages $k-m,\ldots,k-1$, i.e.,

$$\sigma_{k,\text{main}}=[a_{k-m},a_{k-m+1},\ldots,a_{k-1}]. \tag{3.21}$$

[3]Recall that depth k is the trellis stage where the symbol a_k is being exploited for the first time. See Figure 2.12.

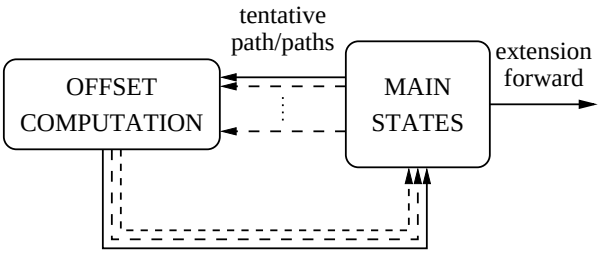

Figure 3.8: Computation of the offset label/labels using tentative paths/paths.

The extension to stage $k+1$ computes $\boldsymbol{\alpha}_{k+1}$; all such $\alpha_{k+1}(\sigma_j)$ are stored. When extending the forward recursion to stage $k+1$ a decision about which symbol/symbols enters/enter the so-called *tentative path/paths* at stage $k-m$ must be made. In the benchmark BCJR this is done using the different offset label strategies, described later in the section. One of them, denoted single offset algorithm, will be chosen for comparison with the proposed M-BCJRs. The tentative path is a decided symbol path used only for the computation of label offsets. This is shown schematically in Figure 3.8. The decision about which symbol enters the tentative path is made using only the information in the forward metrics. Furthermore, the symbols $a_{k-L}, a_{k-L+1}, \ldots, a_k$ determine the label on their respective branch at stage k.

In the backward recursion of the proposed benchmark BCJR algorithm, labels are formed from m main state symbols and $L-m$ offset symbols that lie in the α-decided tentative path. In other words, the offsets in (3.20) are computed using the tentative path from the forward recursion. Since the backward recursion starts from the end of the reversed frame the situation is aligned

$$\begin{array}{rcrll} & & f_L, \ldots\ldots, f_1, & f_0 & \\ \text{extension backward} & \longleftarrow & a_{k-L}, \ldots, a_{k-1}, & a_k & \\ & & \boldsymbol{\beta}_k, & \boldsymbol{\beta}_{k+1} & \boldsymbol{\beta}_{k+2}, \ \ldots \end{array}$$

The $\boldsymbol{\beta}$ vectors do not need to be stored. Instead, $\boldsymbol{\beta}_{k+1}$ can be immediately used together with corresponding $\boldsymbol{\alpha}_k$ to find the LLR at stage k and from it an improved update to the kth symbol in the tentative path. Note that the updated tentative path is much better than the α-decided path; it typically leads to one tenth the bit error rate.

Here are some of the approaches to a reduced state BCJR that have been investigated within this scenario.

(*i*) *True offset algorithm*: In this strategy a different offset state is associated with each main state. A method is needed to choose which offset state survives after the move forward to new main states at stage $k+1$. In analogy to the VA, it is the offset + main path that makes the largest contribution to the new forward metric $\alpha_{k+1}(\sigma_j)$. This strategy performs worse than the other proposed strategies. A simple but effective alternative has proven difficult to find.

(*ii*) *Single offset algorithm*: This method associates the same offset state with all main states. The new offset symbol is taken as the oldest main state symbol that leads to the larger set of α_{k+1} contributions at stage $k+1$ (2^m of these stem from each symbol value). In performance and complexity, this is in fact the best performing strategy found.

(*iii*) *VA aided BCJR*: While the BCJR calculates state probabilities and not symbol decisions, the VA decision path is a good enough tentative path, even when the VA is strongly reduced. A reduced-trellis BCJR which uses the VA decision path as its tentative path needs about one unit smaller m than a BCJR of type (*ii*). However, when the VA complexity is accounted for, there is no overall improvement.

(*iv*) *Iterating*: Multiple passes of any of the preceding can be executed which improves the estimate accuracy of the forward and the backward metrics. Each new forward recursion can use the LLR-determined path from the previous backward recursion for its tentative path. The tentative paths thus constantly improve. However, the accuracy so obtained is not worth the complexity of the additional iterations.

The conclusion is that only a *single* offset state should be associated with all the main states, not a different one for each state, as in the standard offset VA. Furthermore, the symbols used to compute the offset in (3.20) should be soft values, not ± 1. A solution to this comes from the definition of the forward state metric α in the BCJR, which has the form

$$\alpha_k(\sigma_j) \triangleq p(\text{Observe } y_1, \ldots, y_k \cap \text{ISI in state } \sigma_j \text{ at time } k).$$

By summing α_k over the states in $\mathcal{L}_{+1}^{k-m}$, the set of states that have entering symbol $+1$ at stage $k-m$, we obtain

$$\begin{aligned}\pi_{+1} &= \sum_{\sigma_j \in \mathcal{L}_{+1}^{k-m}} \alpha_k(\sigma_j) \\ &= p(\text{Observe } y_1, \ldots, y_k \cap +1 \text{ sent at time } k-m) \qquad (3.22)\end{aligned}$$

and similarly for π_{-1}. The probability of $+1$ at the oldest main state stage can now be estimated as

$$\hat{p}_{+1} = \frac{\pi_{+1}}{(\pi_{+1} + \pi_{-1})}$$

and similarly for $\hat{p}_{-1}$. These enable early decisions about a_{k-m}, i.e., the probabilities are used to decide which symbol enters the tentative path. Although not as reliable as those based on the two-recursion BCJR, they are good enough for calculating a single offset contribution from small ISI model taps. Furthermore a simple and effective soft decision about a_{k-m} is its expected value, that is

$$\hat{a}_{k-m} = \hat{p}_{+1} - \hat{p}_{-1}.$$

A highly likely ± 1 is respectively $\approx \pm 1$ and a completely uncertain symbol is 0, meaning that the corresponding ISI tap is ignored in the offset computation. The single soft offset innovation can improve the BER of an offset BCJR used for simple detection by 10 fold. Further details are given in [101].

Simple Detection Performance

The general behavior of the single offset BCJR in simple detection, as a function of the offset and precursor size, differs little from the offset VA, although the BCJR in our FTN tests requires 1–2 extra units of main state memory to achieve the same error performance. Figure 3.9 compares benchmark VA and BCJR EERs at main state memories $m = 2, 4, 6, 7$, for uncoded FTN with $\tau = .5, .35$ and super minimum phase models (3.11)–(3.12). The same test setup as in Figure 3.7 is adopted and both form tentative symbol estimates at a delay m. The bold lines are the Q-function estimates, based on the full model $\boldsymbol{f}$.[4] The figure shows that the single offset BCJR needs only 32 states ($m = 5$)

[4] A distance study shows that the $d^2_{\min}$-causing error difference sequence is $2, -2, 2$ for $\tau = .5$, with coefficient 1/4 (see [99] for details, and [20] for a general treatment). Thus the full-state VA has EER $\approx .25Q(\sqrt{1.02E_s/N_0})$. The $\tau = .35$ and .25 cases are more complex. The most probable error events at $\tau = .35$ have differences $2, -2$ and $2, -2, 0, 2, -2$; these combine to yield an EER close to $.35Q(\sqrt{.56E_s/N_0})$ for E_s/N_0 =10–15 dB.

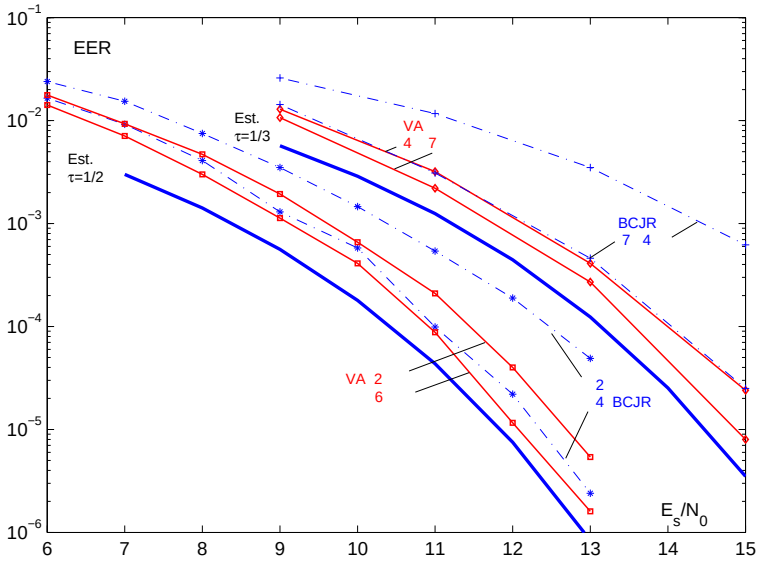

Figure 3.9: Benchmark error event rates vs. E_s/N_0 for simple ISI detection with offset VA (solid) and single offset BCJR (dash dot) at main state memory m. Heavy lines are Q-function estimates.

at $\tau = .5$ and 64–128 ($m = 6$–7) at $\tau = .35$. The offset VA needs somewhat less. This m is also predicted for the VA by the reduced-trellis $d^2_{\min}$ algorithm in [30]. With $\tau = .25$ (not shown) the offset VA needs about 2^9 states, and the BCJR about 2^{13}. These numbers are a benchmark for the M-BCJR results in Section 3.4. Observations for the non-FTN ISI tap sets (3.14)–(3.16) are given at the end of Section 3.4.

In this section an offset-based benchmark BCJR, denoted single offset BCJR, has been proposed. It computes the offset labels using soft symbol values, estimated using the method presented in the section. Additionally, the backward recursion of the single offset BCJR is modified in order to improve the LLR quality in a heavily reduced receiver. Simulation results presented in Section 3.5 show that the single offset BCJR outperforms a reduced-trellis BCJR which completely ignores the contribution from the small ISI taps corresponding to the offset states, and it is therefore a fair benchmark for the proposed M-BCJRs in Section 3.4.

3.4 Proposed M-BCJR Algorithms

This section proposes three new reduced-search M-BCJR algorithms and tests them in simple detection of ISI. The basic M-algorithm for reduced-search of trees and trellises is well known. As a general procedure, the algorithm proceeds breadth-first through a tree structure of metric values, keeping only the dominant M paths at each tree stage. In the M-BCJR, the M-algorithm is applied both in the forward and the backward recursion, finding the dominant M $\alpha_k(\sigma_i)$ and $\beta_k(\sigma_j)$ which, hopefully, are close to the values that a BCJR would find (at the kth time instance). For moderate to severe ISI, the branch metric matrices $\boldsymbol{\gamma}_k$ are very sparse, and most non-zero elements are very small. A useful view is that the M-search implements a sparse matrix calculation in which the vector $\boldsymbol{\alpha}_k$ or $\boldsymbol{\beta}_k$ at each stage is limited to M active components.

The product of the $\{\boldsymbol{\alpha}_k\}$ and $\{\boldsymbol{\beta}_k\}$ produce the set $\{\boldsymbol{\lambda}_k\}$ through $\lambda_k(\sigma_j) = \alpha_k(\sigma_j)\beta_k(\sigma_j)$, σ_j a state at stage k. Log likelihood ratios are obtained from these via

$$\mathrm{LLR}(a_k) \triangleq \log\left(\frac{\Pr(a_k=+1|\boldsymbol{y},\,\mathrm{LLR_{in}})}{\Pr(a_k=-1|\boldsymbol{y},\,\mathrm{LLR_{in}})}\right) = \log\left(\frac{\sum_{\sigma_j\in\mathcal{L}_{+1}}\lambda_k(\sigma_j)}{\sum_{\sigma_j\in\mathcal{L}_{-1}}\lambda_k(\sigma_j)}\right). \quad (3.23)$$

Here $\mathcal{L}_{\pm1}$ are the sets of states reached by $a_k = \pm1$, for which nonzero α and β have both been found. A problem in a heavily reduced search is that one of the sets $\mathcal{L}_{\pm1}$ is often *empty*. In case of an empty set, the numerator or denominator in (3.23) must be replaced by some backup method.

The most straightforward M-BCJR now follows. It works well in simple detection of ISI, and hence it is called simple detection M-BCJR. Recursions start and end in state 0 (all +1 symbols). Inputs to the algorithm are the noisy channel outputs $\boldsymbol{y}$ and a priori probabilities of the symbols. Outputs are the signed LLR values in (3.23). The list of M dominant paths, the M-list, consists of two sublists, one containing α or β values at stage k and one containing the corresponding trellis states. It is straightforward to extend the algorithm to non-binary alphabets and/or MIMO setups.

Forward Recursion. Starting at $k = 0$, perform at stage $1, 2, \ldots, N-1$:

I The forward recursion in (2.96) is computed from the M nonzero values retained in $\boldsymbol{\alpha}_k$. There are M outcomes corresponding to symbol $a_{k+1} = 1$ and M to -1; only the $2M$ corresponding branch metrics γ_k are computed and stored.

II Trellis paths may merge at stage $k+1$. The algorithm detects and removes merges, leaving only one survivor per node whose α value is the sum of the incoming values.

III M largest α values are found and stored for stage $k+1$ and for the β recursion.

Backward Recursion. Starting at $k = N$, perform at stage $N, N-1, \ldots, 2$:

IV The backward recursion in (2.97) is computed from the M nonzero values retained in $\boldsymbol{\beta}_{k+1}$. There are M outcomes corresponding to symbol $+1$ and M to -1; only the $2M$ corresponding branch metrics γ_k are computed.

V Trellis paths may merge at stage k. The algorithm detects and removes merges, leaving only one survivor whose β value is the sum of the incoming values.

VI M largest β values are found, subject to the following condition: *β paths must be kept if their state and stage overlap with that of a stored α.* The M-list is then completed with non-overlapping paths.

Completion. Starting at $k = 0$, perform at stage $0, 1, \ldots, N-1$:

VII Compute the LLR from (3.23). If $\mathcal{L}_{+1}$ or $\mathcal{L}_{-1}$ is empty, the respective λ-sum in (3.23) is set to ϵ, where ϵ is a reserve value set a priori.

The offset state idea is not needed in the M-BCJR; it should simply retain all L state symbols for each of the M paths. It is essential, however, that the M-BCJR ignores the precursor symbols, and it is therefore slightly mismatched to the channel. The merges removed in steps II and V are unlikely, and these steps can be removed without significant performance loss. The idea of a reserve ϵ in step VII and of pursuing β paths that overlap α paths in step VI were both proposed in [96] (the α path list was called the "survivor list"). However, in this thesis the overlapping paths are only given first priority; other β paths are extended for a total of M. The efficiency of this strategy may be seen by observing the search dynamics. During most of the transmission, there are only 1–2 paths in the search overlap, and almost always one of these is correct. Errors occur during rare noise bursts, but it is precisely here that the search is chaotic; the extra β paths are needed in case they merge to α paths a few trellis stages later.

3.4.1 Comments on Complexity

In this section only on the approximate order of the computations is considered. Steps 3 and 6, which find the best M, are equivalent to finding the median

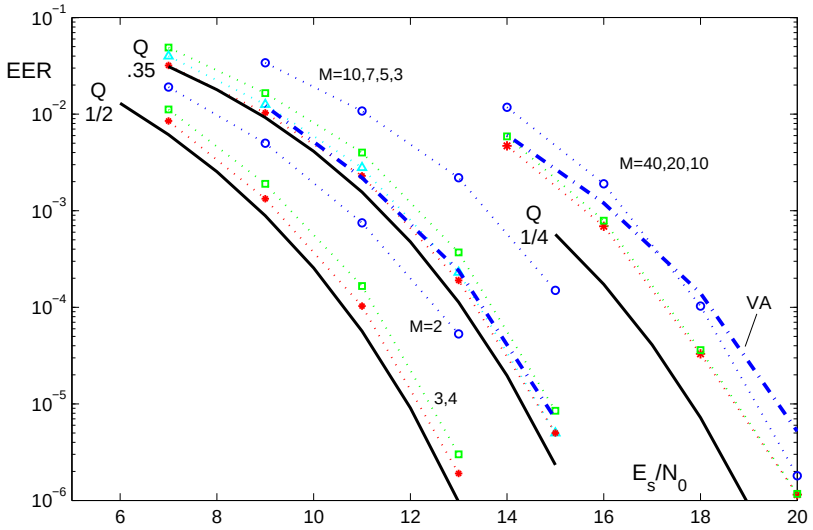

Figure 3.10: Error event rates for simple ISI detection vs. E_s/N_0 in dB for the simple detection M-BCJR (dotted lines); shown for comparison are offset VA (dash-dot) and Q-function estimate (solid).

of a group of items. An important property of median finding is that its computation is linear in M. The M-algorithm does not order a list, which would require order $M \log M$. In keeping with this, a true M-algorithm is the one where *all* computation is of order M. The search for the median is thus implemented in order M, but so is also removal of state merges in steps II and V and finding the overlap of α and β in step VI. The key to the last two is keeping all path lists in state order, which is itself a linear operation. Details of these linear procedures are omitted. Since there are two recursions,[5] computation has the approximate order $2M$, with M the number of trellis states visited at each stage; by this measure the offset BCJR has twice the complexity of the offset VA.

3.4.2 M-BCJR in Simple Detection

Figure 3.10 plots the EER for this M-BCJR algorithm used as a simple detector at the ISI intensities $\tau = .5, .35, .25$. The algorithm decides symbols

[5]The backup recursion in the algorithm that follows in Section 3.4.3 is much smaller than M and its contribution can safely be neglected.

from the LLR sign at its output. Heavy lines show Q-function estimates. For comparison, performances are plotted for the 256- and 4096-state offset VA, for $\tau = .35$ and .25 respectively. The simple detection M-BCJR can perform better than the offset VA, especially at $\tau = .25$, because a practical VA cannot be large enough to deal with every detail of the severe ISI. The M-BCJR needs only $M = 3, 7, 20$ respectively for the three FTN cases at higher E_s/N_0, and somewhat more at lower. The appearance of such an upper limit to M is typical of one-pass M-algorithm searching of code and modulation trellises. Not shown in the figure are results for $\tau = .703$, the Mazo limit; here only $M = 3$ is required.

Comparisons to Earlier Work

Most of the ISI examples in the literature are mild, and we now compare to some of these. Magarini et al. [104] investigate alternatives to the benchmark offset VA, using ISI model (3.15). Same EER and BER results are obtained for the offset VA (Figures 6 and 7 of [104]), which here needs $m = 4$ (16 states). However, they are able to improve this benchmark performance somewhat with a non-allpass prefilter receiver. The simple detection M-BCJR with only $M = 7$ improves upon their prefilter result by 1.5 dB and in fact lies on the full ML bound for BER given in [104].[6]

Fertonani et al. [96] consider ISI (3.16) with $d^2_{\min} = 2$ in a turbo equalization configuration. The M-BCJR with $M = 6$ applied to simple detection of ISI (3.16) achieves BER close to $3Q(\sqrt{2E_s/N_0})$, which is the asymptotic estimate from distance analysis.

3.4.3 Backup and Smoothed Backup M-BCJR

When an accurate LLR is needed, as in iterative decoding in the next section, the simple detection M-BCJR is not sufficient. A serious problem is that an empty $\mathcal{L}_{\pm 1}$ set normally occurs when E_s/N_0 takes practical values. The M-search is then quite sure of the correct symbol, one set is empty and there is no estimate of the LLR magnitude at all, other than the ϵ set a priori. Several solutions exist in the literature. This section proposes a practical one which adds a third, low complexity recursion, whose purpose is to produce a backup LLR magnitude when the two M-BCJR recursions do not. The following algorithm, called the backup M-BCJR is proposed. It replaces step VII with:

[6] The dominant error events lead to a BER estimate $\approx 2Q(\sqrt{.31E_s/N_0})$.

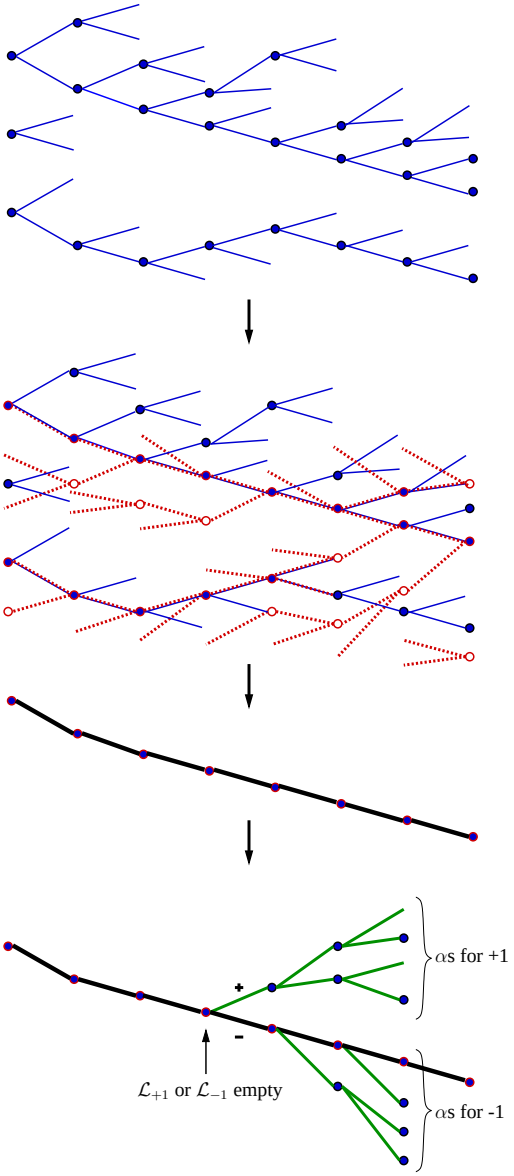

Figure 3.11: Backup M-BCJR example, showing α and β recursions, hard decision path, and backup recursion.

New step VII. Decide the symbols from the sign of (3.23), noting when $\mathcal{L}_{\pm 1}$ is empty. In a third recursion, compute a symbol probability from the αs only, as follows. From each node of the decided symbol path, trace forward through the ISI trellis a certain length of stages; αs that stem from the decided branch form the probability of one symbol outcome and αs in the "incorrect subset" of the node form the probability of the other outcome. The traces are performed with a small M-search of size M_{B} (typically $M_{\mathrm{B}} = 2$ works well).

The necessary search for all the decided nodes at once can be arranged in a simple way. The search gives a backup estimate of $P[a_k = +1]/P[a_k = -1]$, to be used when one (or both) of $\mathcal{L}_{\pm 1}$ is empty; otherwise (3.23) is used. A sketch of the entire backup M-BCJR procedure with $M = 3$ is illustrated in Figure 3.11. First, the α recursion is performed, followed by the β recursion. The β paths, shown dotted, must follow the α paths as a first priority, and in this case they are shown as overlapping. Shown third is the decided path that results from the whole first two recursions. Finally, a backup search with $M_{\mathrm{B}} = 2$ is shown for a branch that was decided to be -1.

The backup LLR values can be noisy in a heavily reduced M-search, especially in the early iterations of a turbo decoder, and therefore a useful technique is to smooth them. A simple moving average filter, such as $.2z + .6 + .2z^{-1}$, can improve the BER of the iterative decoder, if only applied to the backup values in the first iteration. This third scheme is called the smoothed backup M-BCJR.

3.5 Turbo Equalization

This section evaluates the BER performance of turbo equalization when the smoothed backup M-BCJR performs the ISI detection. Whereas only the sign of the LLRs was needed for simple ISI detection, turbo decoding requires reasonably accurate absolute values, especially in the early iterations. This is provided by the smoothed backup M-BCJR. It will be compared with two reduced-trellis benchmark BCJRs, the memory-m single offset BCJR proposed in Section 3.3 and a truncated BCJR that simply calculates its branch labels based on the $m_{\mathrm{tr}} + 1$ dominant taps with no label offsets. EXIT charts [85] are used to monitor convergence behavior.

3.5.1 Turbo Loop Stability

Since a practical implementation of the optimal detector is often prohibitively complex, reduced complexity methods can be used instead. A heavily reduced

detector, compared to the full ISI state space $|\Omega|^L$ where Ω is the symbol alphabet, is often unable to produce exact soft information about some of the detected symbols or bits. In the worst case there could be no realibility information at all. These detectors can in general determine the sign of the log likelihood ratios with reasonably low error probability. However, the absolute values are unknown.

One of the reasons is the irregularity of a reduced trellis. The magnitude of the terms summed in the numerator and in the denominator of (3.23) may differ a lot, resulting in over-estimated LLRs. In an iterative scheme, over-estimated LLRs for correctly detected symbols or bits can speed up the convergence rate while over-estimated LLRs for incorrectly detected data can severly degrade the overall performance. In a heavily reduced search one of the sets $\mathcal{L}_{\pm 1}$ can be empty. This can occur when all paths corresponding to a certain symbol at depth k have been discarded.

A possible solution is to a priori assign a constant value ϵ to either the numerator or the denominator in (3.23), as done in Section 3.4. Alternatively, a constant value can be assigned directly to the LLRs for these symbols. The latter is known as *LLR clipping* [105]. However, if a large number of bits require such assignment, which is often the case for a heavily reduced detector, this method easily fails. An SNR-aware improvement of the LLR clipping method based on error probability estimates in List MIMO detection was proposed in [105]. Additionally, a complete list of references, covering the LLR clipping method is given in [105].

As M decreases, there is an increasing number of bits with undetermined LLRs. The values M in the tests performed in this chapter are chosen with practical receivers and good performance at reasonable SNR in mind. Measurements show that one of the sets $\mathcal{L}_{\pm 1}$ is empty at 50–80% of the trellis stages and consequently some sort of LLR reserve procedure is essential. This result is expected since the M-BCJR algorithm tends to over-estimate the LLRs for small values of M.

Since low-quality and over-estimated LLRs affect the stability and convergence of the iterative detector, some way needs to be found to keep it under control. As a complement to the backup M-BCJR, it is beneficial to scale the likelihoods passed around the turbo loop by a "gain" $g \leq 1$. In the simulations the extrinsic LLRs are therefore scaled by $\sqrt{g}$ before each component decoder. Gains were also suggested in [96].

In Figure 3.12, the effect of the scaling gains g is shown for a serially concatenated setup with a (7,5) convolutional encoder and the $\tau = 0.35$ FTN channel in (3.12). Turbo equalization BER results for the backup M-BCJR with $M = 6$ and $M_{\mathrm{B}} = 2$ are shown. The blocklength $N = 5000$ and 10 iterations are performed. The results indicate that there exists an optimal

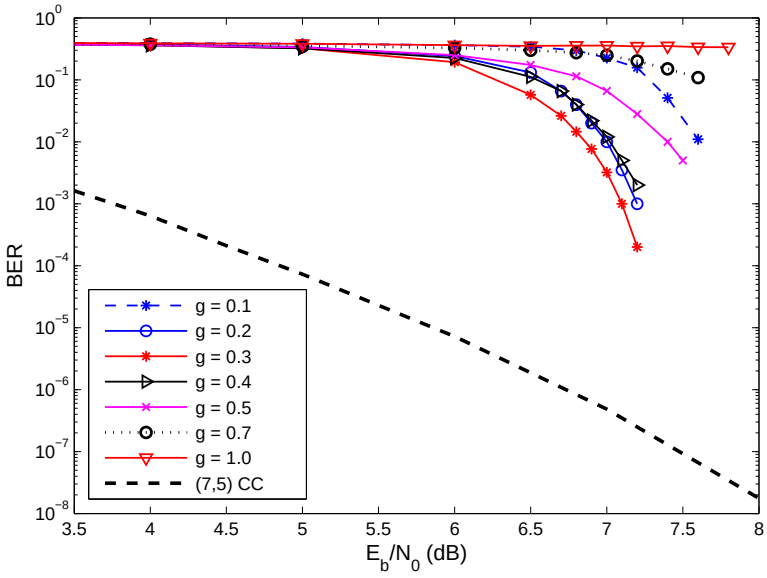

Figure 3.12: Turbo equalizer BER vs. E_b/N_0 for $\tau = 0.35$ and different scaling gains g. The backup M-BCJR with $M = 6$ and $M_{\mathrm{B}} = 2$ is employed and 10 iterations are performed.

g, which is reasonably constant across SNR, and that, without increasing the overall complexity, huge performance improvements are possible. By choosing an appropriate g the convergence to the performance of the underlying code occurs at a considerably lower SNR. In Figure 3.12 the optimal value of g is ≈ 0.3. Note that without scaling of the extrinsic LLRs the BER performance is severely degraded. The value of g depends on the the whole setup, that is, the choice of the component decoders (including the value of M), blocklength N and the SNR. However, once the optimal g is found for a setup, the overall complexity remains unaltered.

3.5.2 Simulation Results

The turbo equalization setup is as follows: A block of N information bits is encoded by the (7,5) rate 1/2 feed-forward convolutional encoder, generating a length $2N$ codeword. The encoded sequence feeds a size $2N$ random interleaver whose output is mapped to binary symbols ($0 \to +1, 1 \to -1$). The signal is terminated so that the transmission begins and ends in the all $+1$ ISI state.

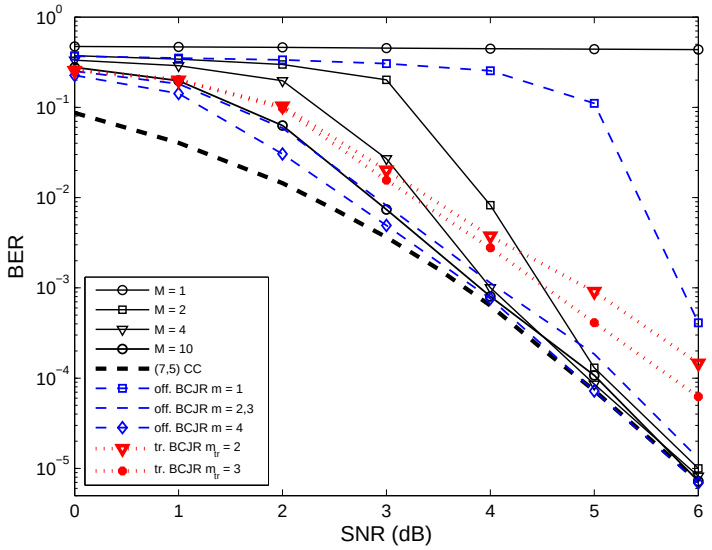

Figure 3.13: Turbo equalizer BER vs. E_b/N_0 for $\tau = 1/2$, comparing single offset BCJR (dashed), smoothed backup M-BCJR (solid) and truncated BCJR (dotted) for different complexities.

Iterative decoding as in Figure 3.1 is performed, applying one of the three BCJRs as the ISI equalizer. All three ignore precursors when forming labels. In the smoothed backup M-BCJR smoothing is applied only at the first iteration with the smoother $[1, 3, 1]/5$. The convolutional decoder is a full-state BCJR (4 states). In this chapter the signal-to-noise ratio (SNR) is defined as E_b/N_0 where $E_b = 2E_s$.

The component decoders exchange soft information in the form of LLRs, hopefully converging to a decision about the data. A complete loop is one "iteration". The block length is $N = 12000$ and 20 iterations are performed (60 for $\tau = .25$). Fewer iterations and shorter block lengths ($N \approx 1000$) are more practical in hardware, and these performed almost as well, but more care is needed to assure loop stability. Decoder tests are run until ≥ 50 blocks with errors occur.

Improved quality of the LLRs and stabilizing loop gains allow a smaller value of M. Simulation results show that the best loop gains in our setups lie near .4 for $\tau = .5$ and .35, and .25 for $\tau = .25$. These values are used and they are much larger than those in [96]. In [82] and [106] it has been shown that recursive precoding leads to additional gains in turbo equalization but such a precoder has not been employed here. The performance of the considered

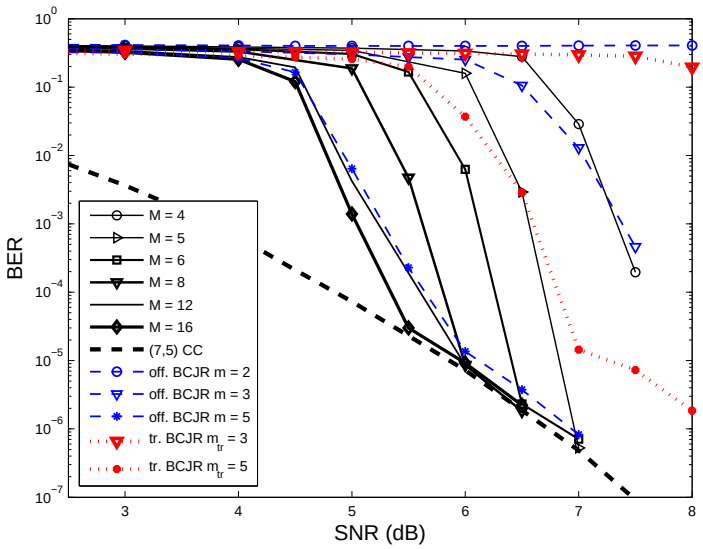

Figure 3.14: Turbo equalizer BER vs. E_b/N_0 for $\tau = 0.35$; single offset BCJR, smoothed backup M-BCJR and truncated BCJR as in Fig. 3.13.

system can therefore never be better than the performance of the underlying convolutional code, shown as a bold dashed 'CC' line in the figures. However, for FTN this performance is obtained at a considerably higher rate in bits/Hz-s.

A constant M over the iterations is employed in this chapter. However, the first few iterations are the most important, and both M and the scaling gain g should vary with the iterations; this should be explored in future work. The first iteration is precisely the simple detection of Section 3.4, so a suggestion for at least the starting Ms are the values found there.

Figure 3.13 shows BER results for the smoothed backup M-BCJR, single offset and truncated BCJRs when $\tau = .5$ with taps (3.11). The relatively mild ISI is not difficult for the first two, but the truncated BCJR suffers from energy loss caused by a too-early truncation and fails to converge to the CC line at high SNR. For $M = 2$ the smoothed backup M-BCJR performs better than the memory-1 single offset BCJR, which has similar complexity, and it achieves very nearly the CC-line BER at SNR $\approx$ 5 dB. For higher complexities the smoothed backup M-BCJR and single offset BCJR are similar but the last is clearly superior to the truncated BCJR, which shows that some of the long-tail taps cannot be ignored.

The situation changes when the FTN signaling rate increases. Figure 3.14

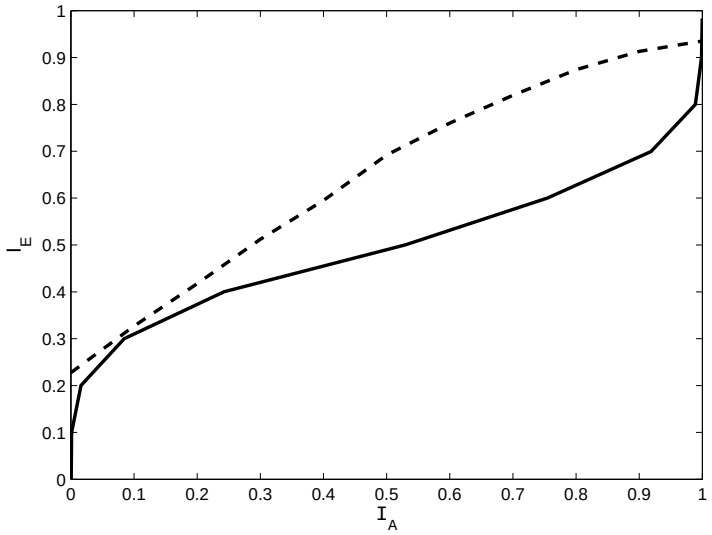

Figure 3.15: An EXIT chart at $E_b/N_0 = 7$ dB, showing extrinsic vs. a priori information for block length 12000. Dashed curve represents the smoothed backup M-BCJR with $M = 5$ and $\tau = 0.35$; solid curve shows the (7,5) outer convolutional code.

plots the $\tau = .35$ case, which is a much more severe ISI. The smoothed backup M-BCJR efficiently removes the ISI even when $M \leq 5$. It is superior to the single offset BCJR for all complexities. The reduced trellis of the truncated BCJR is now much smaller than the effective state space of the ISI, causing severely degraded BER. Even for $m_{\text{tr}} = 5$ the truncated BCJR is unable to eliminate the effects of the intense ISI. Accounting for the long tail taps makes it possible for the single offset BCJR to achieve the CC performance even though the number of main states is equivalent to that of the truncated BCJR. The reduced-search M-BCJR clearly prevails at this higher ISI intensity. Its turbo convergence threshold can be determined through a study of EXIT charts, with the system converging to the CC line when there is an open tunnel between the EXIT curves. Figure 3.15 shows the case $\tau = .35$, $M = 5$ and $E_b/N_0 = 7$ dB, for which there is a narrow tunnel; Figure 3.14 verifies that there indeed is convergence to the CC line with $M = 5$ for the first time at about 7 dB.

Turbo equalization (60 iterations) for the extreme ISI case with $\tau = .25$ and the 32-tap channel model (3.13) is shown in Figure 3.16 for several M. Here the stability of the turbo loop is very sensitive to the scaling gain g and the block

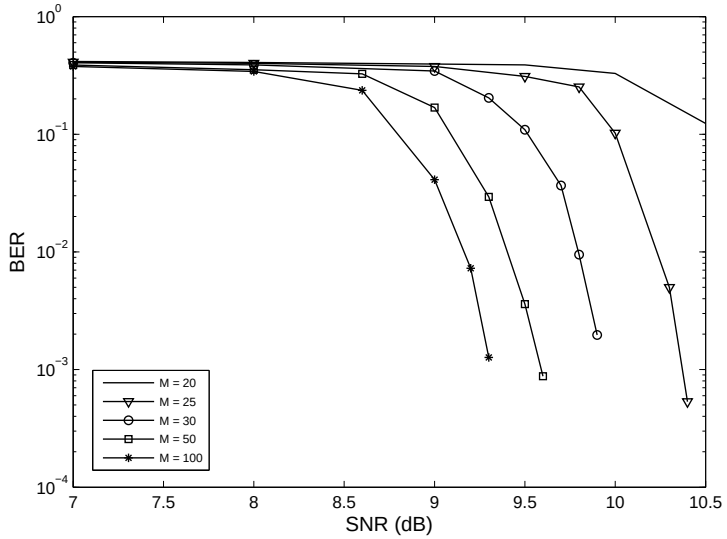

Figure 3.16: Turbo equalizer BER vs. E_b/N_0 for $\tau = 1/4$; smoothed backup M-BCJR with $M = 20$–100. CC reference lies far below the SNR axis.

length. Lengths less than 12000, smaller M and g too large severely degrade the BER. The M in the smoothed backup M-BCJR needs to be in the range 25–100, compared to 20 in simple detection of ISI and many thousands for the two reduced-trellis benchmark BCJRs. Depending on M, sharp convergence thresholds lie in the range 9–10.5 dB; after the last point shown for each curve there is a sudden fall to the CC line, which is located far below the SNR axis.

As further insight into the role of the backup recursion, Figure 3.17 shows the M-BCJR BER for several backup M_{B} when $\tau = .35$. A heavily reduced search with $M = 6$ leads to empty $\mathcal{L}$ sets for 80% of the LLRs. The case $M_{\mathrm{B}} = 0$ corresponds to the simple detection M-BCJR with a small, fixed reserve value ϵ whenever an $\mathcal{L}$ set is empty. Its BER performance is 3 dB worse than the backup M-BCJR with $M_{\mathrm{B}} = 4$. However, most of the performance gain is obtained with only $M_{\mathrm{B}} = 2$. A comparison is also made to published M-BCJRs of which the most important one appears in [96] and is similar to the simple detection M-BCJR without steps IV and VI. Its BER performance is shown with both mathematically correct minimum and super minimum phase models.[7] The figure shows that both the backup recursion and the super min-

[7]The mathematically correct minimum phase model is obtained from 60 or more central

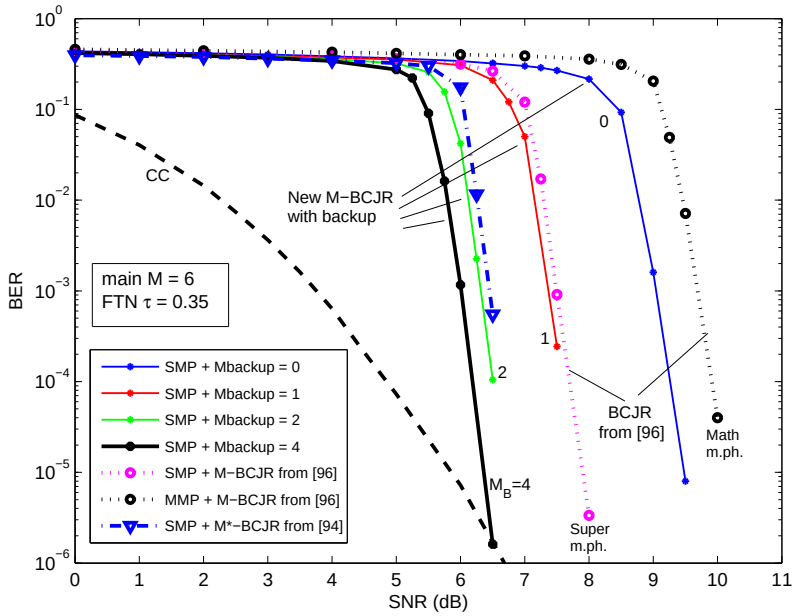

Figure 3.17: Turbo equalizer BER vs. E_b/N_0 for backup M-BCJR with $M_\mathrm{B} = 0, 1, 2, 4$, with comparison to algorithms from [94] and [96]. The second is tested with both mathematically correct and super minimum phase model (3.12). FTN $\tau = .35$ and $M = 6$.

imum phase idea are needed and that the gain from both innovations is about 4 dB. A final comparison in Figure 3.17 is made to the M*-BCJR algorithm of Sikora and Costello [94]. It has similar performance to the backup M-BCJR with $M_\mathrm{B} = 2$, but it is much more complex.

Comparisons to Earlier Work

Colavolpe et al. [93] report results for turbo equalization with tap set (3.14) and the 16-state recursive systematic convolutional code (23,35). They use an offset BCJR, block size 2048 data bits, and scaling gain $g = .15$. Their system needs 6 iterations and a 16-state BCJR to reach the CC line BER at 5 dB, although

samples of $h(t)$ and begins with $\approx .031, .142, .344, \ldots$. To obtain a fair comparison, the receiver in [96] is delayed by $K_p = 1$ (see Section 3.2) and its first tap is set to 0. Its performance will otherwise be worse than shown in Figure 3.17.

8 states performs well (Figure 5 of [93]).[8] With the same setup (except that $g = .44$), the smoothed backup M-BCJR with $M = 3$ and $M_{\mathrm{B}} = 2$ needs only 4 iterations at SNR 5 dB and 3 iterations at 6–7 dB. Even if we account for the complexity of a complete backup recursion, the overall complexity is lower than that in [93].

Fertonani et al. [96] test several M-BCJR decoders with ISI (3.16) and the (7,5) convolutional code, as mentioned in Section 3.4. Their M-BCJRs need 6–8 paths and 20 iterations to reach the CC line (see Figure 5, [96]). The smoothed backup M-BCJR needs $M = 3$ and only 5–11 iterations, depending on the SNR. The poorer performance in [96] probably stems from the lack of a backup recursion and any minimum phase conversion.

3.6 Other M-BCJR Algorithms

In order to reduce the negative effects on the receiver error rate caused by empty $\mathcal{L}_{\pm 1}$ sets, the backup M-BCJR from Section 3.4 adds a third forward recursion. In this section a different approach is considered. Instead of adding a third low-complexity recursion, the M-BCJRs proposed here are constrained to retain a certain number of states (with their α values) corresponding to the less probable input symbol at each trellis depth k. They all construct a reduced trellis in the forward recursion based on the M retained $\alpha_k(\sigma_j)$. As in Section 3.4, state duplicates are not allowed in the M-list. However, in this section there is no side condition that β paths must be kept if their state and stage overlap with that of a stored α.

Furthermore, a genie-aided M-BCJR algorithm, denoted by $\mathcal{G}_1$, which has access to the exact values of $\alpha_k(\sigma_j)$ and $\beta_k(\sigma_j)$ for all depths k (computed with a full-complexity BCJR), will be used as a benchmark. It computes the LLR-values $\mathrm{LLR}(a_k)$ in (3.23) using only the M largest values $\alpha_k(\sigma_j)$. A mathematical formulation of the genie-aided detector $\mathcal{G}_1$ is given in Chapter 4.

Let also $\mathcal{S}_M$ and $\mathcal{S}_{\not M}$ denote the set states with M largest forward metrics α and the set of the remaining states at a certain depth, respectively. The noisy channel outputs $\boldsymbol{y}$ and a priori probabilities of the symbols are inputs to the algorithms while outputs are the LLR values in (3.23).

The following M-BCJR branching strategies have been investigated within this scenario.

Strategy 1: After a forward extension to trellis depth k and before the removal of state duplicates there are M states with most recent input $a_k = +1$ and M states with $a_k = -1$. Let $\mathcal{S}_{+1}$ and $\mathcal{S}_{-1}$ denote the set of states

[8] However, Douillard et al. [10] report that only 3 iterations are needed at 6–7 dB SNR.

with most recent input +1 and -1 at depth k, respectively. By summing the forward metrics of the states in each set, decide which input symbol has a larger contribution at depth k. Take $M/2$ best states, i.e. states with largest forward metrics, from this set and store them in the M-list. Complete the list with $M/2$ best states from the set with the smaller contribution of forward metrics. If M is an odd number, store $\lceil M/2 \rceil$ best states from the set with the larger contribution and complete the list with $(M - \lceil M/2 \rceil)$ best states from the other set. Throughout, $\lceil \cdot \rceil$ denotes the ceiling function. The backward recursion β in (2.97) is computed independently from α. However, if the backward metric of a state σ_j at depth k equals 0, i.e. $\beta_k(\sigma_j) = 0$, it is replaced with $\beta_{k,\min}$ where $\beta_{k,\min}$ is the smallest non-zero backward metric in the M-list at depth k. Note that, in this strategy, the M-list in the forward recursion may contain states from the set $\mathcal{S}_{\not M}$.

Strategy 2: This M-BCJR is constrained to, at each trellis depth k, keep at least one state from the set with the smaller contribution of forward metrics. After sorting and removal of merges, M best states are stored. Assume now that the M-list only contains states from the set $\mathcal{S}_{+1}$. An M-BCJR based on this strategy replaces the state with the smallest metric in the M-list with the best state from the set $\mathcal{S}_{-1}$. The backward recursion is identical to that in the previous strategy.

Strategy 3: In Strategy 1 the number of stored αs from each set $\mathcal{S}_{+1}$ and $\mathcal{S}_{-1}$ is, unless M is an odd number, independent of their metric contributions. In this strategy their metric contributions determine the fraction of states stored in the M-list from the respective set. After sorting and removal of merges, the forward metrics at depth k are normalized as

$$\sum_j \alpha_k(\sigma_j) = 1. \tag{3.24}$$

Furthermore, let

$$A_k^+ = \sum_{\sigma_j \in \mathcal{S}_{+1}} \alpha_k(\sigma_j) \tag{3.25}$$

and similarly for A_k^-. If now $\mathcal{S}_{+1}$ is the set with the largest contribution at depth k, then store the best $\lceil A_k^+ M \rceil$ states from $\mathcal{S}_{+1}$ for stage $k+1$. Complete the M-list with the best states from $\mathcal{S}_{-1}$. The backward recursion is the same as in the previous two strategies.

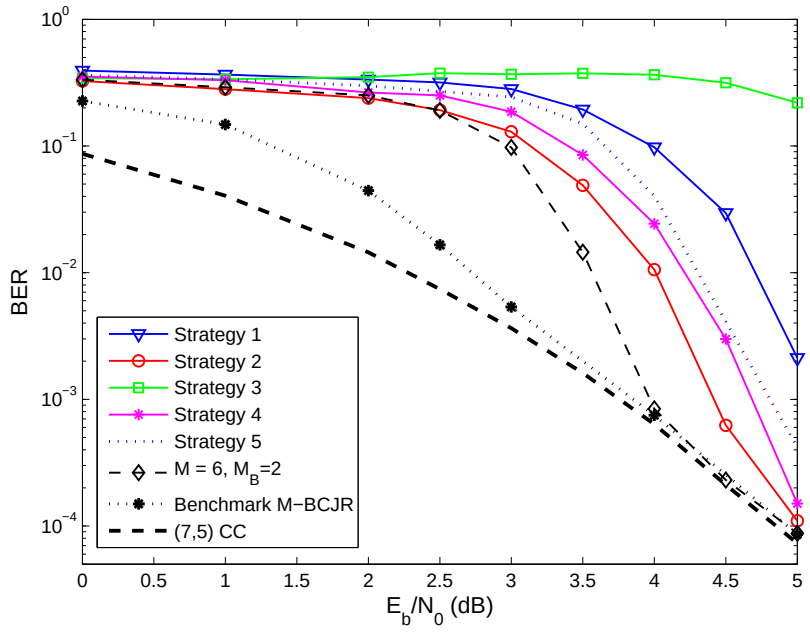

Figure 3.18: Turbo equalizer BER vs. E_b/N_0 for $\tau = 1/2$, comparing the different M-BCJRs for $M = 8$.

Strategy 4: In this M-BCJR no constraint is imposed on the forward recursion α other than that only the M best states at each depth k are retained. In the backward recursion, $\beta_{k,\min}$ replaces all $\beta_k = 0$ as before.

Strategy 5: As Strategy 4 but without the $\beta_{k,\min}$ feature. It is mostly used as a comparison.

The same turbo equalization setup as in Section 3.5 is adopted. However, the blocklength N is here set to $N = 5000$ and 10 iterations are performed. The ISI channel is the $\tau = 1/2$ FTN channel in (3.11). All proposed strategies ignore precursors when forming labels.

The BER results for $M = 8$ are shown in Figure 3.18. A comparison of the BER performance for Strategy 1 and 2 indicates that forcing the M-BCJR to retain many paths which are not among the best M is not a good strategy. However, by letting the M-BCJR only keep the dominant M αs at each depth as in Strategy 4 is also inferior to forcing it to always retain at least one path

from each set $\mathcal{S}_{\pm 1}$ as in Strategy 2. Strategy 2 is in fact the best performing strategy among the proposed 5. A possible explanation is that the negative effects of a constrained forward recursion are smaller than the corresponding gains in the form of improved quality of the soft information exchanged in the turbo loop. The metric contributions of the sets $\mathcal{S}_{+1}$ and $\mathcal{S}_{-1}$ are not appropriate measures to determine the fraction of states stored in the M-list from the respective set. This is clearly indicated in Figure 3.18 by the poor performance of Strategy 3.
Finally, by comparing the BER performance of M-BCJR algorithms based on Strategy 4 and 5, we can conclude that the $\beta_{k,\min}$ feature in the backward recursion can improve the BER performance slightly. However, this might not be the case when the value of M approaches the full complexity value $|\Omega|^L$. All proposed strategies in this section have much inferior performance compared to the performance of the genie-aided M-BCJR, $\mathcal{G}_1$, shown as a benchmark. However, according to Figure 3.18 the backup M-BCJR from Section 3.4 with $M = 6$ and $M_{\mathrm{B}} = 2$ outperforms all 5 M-BCJRs presented in this section. Despite this it is still bounded away from the performance of $\mathcal{G}_1$.

3.7 FTN Pulse Excess Bandwidth Optimization

The objective of this section is to establish the best excess bandwidth β when $h(t)$ is a root raised cosine pulse with excess bandwidth $0 \leq \beta \leq 1$; if $\beta = 0$ a sinc pulse is obtained. The one-sided baseband bandwidth W is given by $W = (1 + \beta)/2T$. Additionally, the Section 3.5 turbo equalization setup (here with $N = 5000$) and the outer (7,5) convolutional code are assumed.

An open convergence tunnel between the EXIT curves is observed for all τ above a certain threshold. Above it the error performance of the concatenated system is virtually identical to the that of the outer convolutional code. The threshold depends on the SNR; this section uses the SNR where the (7,5) code alone achieves BER 10^{-5}, that is, $E_b/N_0 = 5.85$ dB. The EXIT chart in Figure 3.19 shows a case near the threshold τ, where the convergence tunnel is narrow.

In Figure 3.20 turbo equalization receiver tests are shown for β=.1, .2, .3, .4. The component decoder for the FTN signaling is the Section 3.5 truncated BCJR that calculates its labels based on the $m_{\mathrm{tr}} + 1$ dominant taps with no offsets. The ISI response is truncated to memory 6, i.e., $m_{\mathrm{tr}} = 6$ (64 states); 10 iterations in the turbo equalization have been performed. The plot shows BER versus τ. The critical thresholds where the error rate departs from $\approx 10^{-5}$ are clearly seen and lie in the range .30–.43 for the different β.

In order to compare different β the bandwidth consumption must be taken into account. If a system based on $\beta = .4$ can have more compression than one

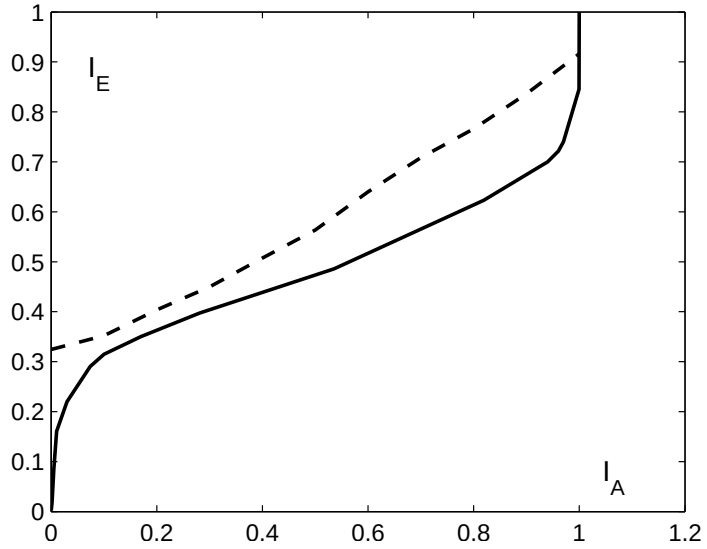

Figure 3.19: An EXIT chart at $E_b/N_0 = 5.85$ dB, showing extrinsic vs. a priori information for block length 5000. Dashed curve is from root RC pulse with $\beta = .3$ and $\tau = .32$; solid curve is from (7,5) outer convolutional code.

based on $\beta = .2$, it cannot necessarily be claimed that $\beta = .4$ is better, since .4 uses more bandwidth. The plot must show BER against the normalized bandwidth W_{norm}, which is W/R, where W is the one-sided baseband bandwidth and R the data bit rate. We have that $W_{\text{norm}} = W/R = ((1+\beta)/2T)/(1/2\tau T) = (1+\beta)\tau$. Figure 3.21 shows the same plot as in Figure 3.20 but now against the normalized bandwidth. According to the figure, the best β are $\beta = .4$ and .3, which are slightly better than $\beta = .2$ and .1. This has significant practical importance since larger β are easier to implement.

3.8 Conclusions

Several BCJR algorithms whose calculation of recursions is limited to M significant terms have been proposed and compared to reduced-trellis VA and BCJR benchmarks based on the offset label idea. Various offset label strategies have been considered; the best performing one (in terms of BER) serves as a benchmark for the proposed M-BCJRs. The application has been to simple

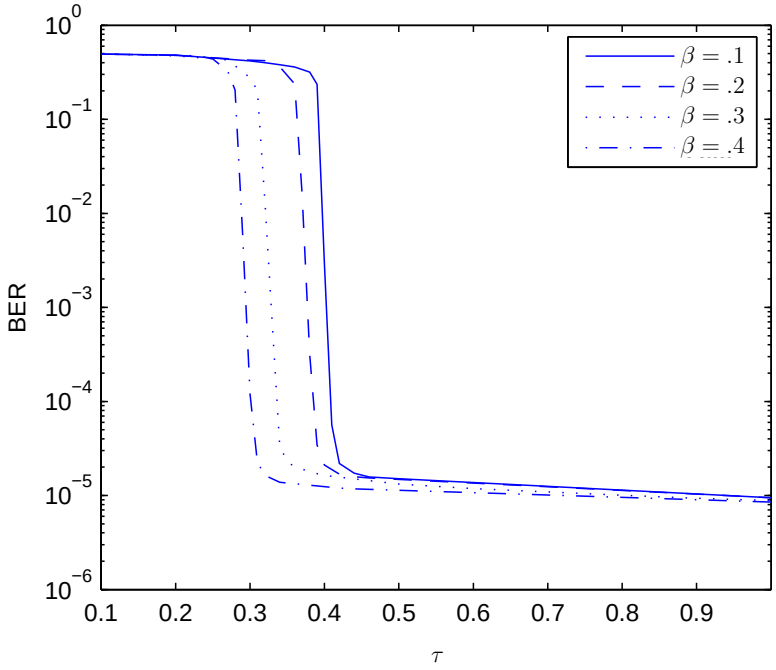

Figure 3.20: Receiver BER tests versus τ for systems based on root RC pulses with excess bandwidth β. All systems operate at $E_b/N_0 = 5.85$ dB.

ISI removal and turbo equalization of channels with spectral zeros and strong narrowband ISI, where M is much smaller than the effective ISI state space. Several important innovations have been proposed. An improvement to the minimum phase allpass filtering sharpens the focus of the ISI model energy. When combined with a delayed and slightly mismatched receiver, the decoding allows a smaller M without significant loss in BER. By adding a third low-complexity M-BCJR recursion, LLR quality is improved for practical values of M, leading to a major BER improvement in turbo equalization. Other innovations are the use of single tentative soft symbol estimates to improve the reduced-trellis benchmark BCJR and a modified method for retaining backward recursion values. All these improvements work together to create a turbo equalizer of reasonable complexity, which in an FTN application can lead simultaneously to an energy saving of 4 dB and a bandwidth reduction of 35% compared to binary orthogonal signaling.

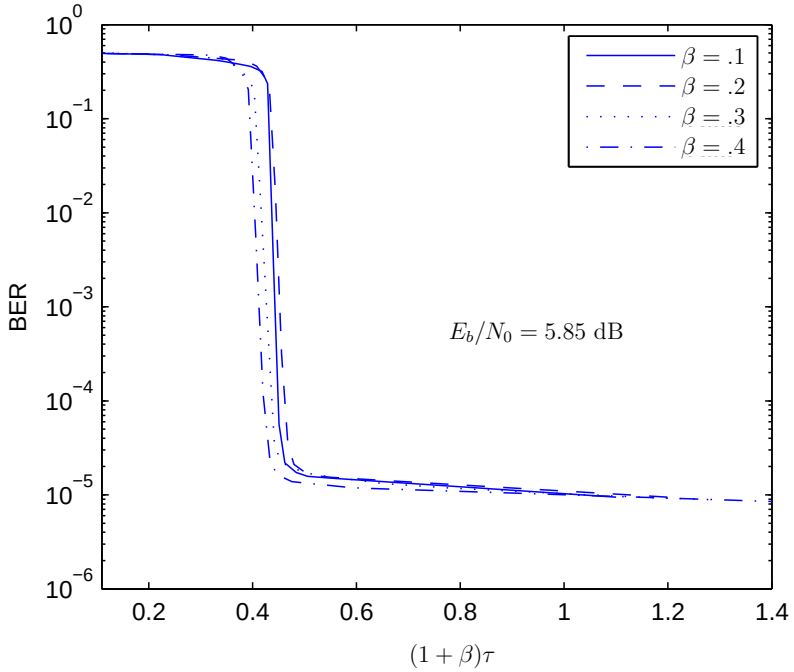

Figure 3.21: Receiver tests for systems based on root RC pulses with excess bandwidth β, plotted against the normalized bandwidth $(1+\beta)\tau$. All systems operate at $E_b/N_0 = 5.85$ dB.

This chapter also considered receivers and arising stability problems when working within the white noise constraint. Chapter 4 on the other hand investigates the effect of the internal metric calculations on the performance of Forney- and Ungerboeck-based reduced-complexity equalizers.

Chapter 4

A Comparison of Ungerboeck and Forney Models for Reduced-Complexity Detection

This chapter investigates the effect of Ungerboeck and Forney metrics on the bit error rate performance for reduced-complexity receivers of the M-algorithm type. As already stated in Chapter 2, it is possible to define a maximum-likelihood receiver for both observation models. In the Forney observation model (2.54) the branch metric for ISI channels is given by (2.87). In the Ungerboeck model (2.52) the noise at the receiver is colored. However, a maximum-likelihood receiver can still be realized using (2.89) instead of (2.87). Even though the final output of a full-complexity detector is identical for both observation models, the internal metric calculations are different. Hence, sub-optimum methods based on the two models need not produce the same final output. Uncoded and serially concatenated systems with ISI and MIMO channels are considered. An example of a serially concatenated system with encoding and ISI is shown in Figure 2.14. Additionally, new models, referred to as *middle models*, working in between the Ungerboeck and Forney models are proposed and evaluated. Based on simulation results, it is demonstrated that practical Forney decoders outperform those operating on the Ungerboeck

model for high signal-to-noise ratios, while the situation is reversed for low SNR levels. A simple method for finding the optimal choice of observation model (in BER sense) is proposed and tested. Mutual information results are given and an analysis of the SNR-asymptotic detector behavior is performed. This chapter was partly presented in [107].

4.1 Introduction

Reduced-complexity detection of intersymbol interference and multiple-input multiple-output (MIMO) channels based on the Forney and Ungerboeck observation models is considered. As shown in Chapter 2, ISI can be introduced by a frequency selective communication channel or by filtering and pulse shaping at the transmitter. In MIMO systems, multiple antennas are used at the transmitter and the receiver to improve communication performance, for example, to increase the data throughput, which is the main reason why they have attracted attention in wireless communications. In both ISI and MIMO channels, equalization is required at the receiver.

The maximum-likelihood receiver, which can efficiently be realized using the Viterbi algorithm, is often prohibitively complex, and alternative detection methods with an acceptable complexity-performance tradeoff are needed. This chapter considers reduced-complexity receivers of the M-algorithm type, and investigates the effect of the Ungerboeck and Forney metrics on the BER performance. Although the performance of a receiver can be measured by several means, this chapter is restricted to BER and mutual information results. Both coded and uncoded transmission over an ISI/MIMO channel, as depicted in Figure 4.1 is considered. In case of multiple transmit and receive antennas, the ISI block is replaced with a MIMO block and data symbols are transmitted block-wise.

As already mentioned in Chapter 3, Forney showed in 1972 [4] that the sampled outputs of a filter, matched to the receive signal pulse, provide sufficient statistics for optimum detection. Since white noise is often preferred, the sampled matched filter (MF) outputs are filtered by a whitening filter which yields the Forney observation model [4]. In [21] Ungerboeck proposed a receiver that works directly on the MF output without whitening. The MF output is commonly referred to as the Ungerboeck model.

It is possible to formulate tree/trellis based detection for both observation models. Even though the final output of the VA or the BCJR decoder is identical for both models, the internal metric calculations are different. This chapter also introduces new models working in between the Ungerboeck and Forney models, which will be referred to as *middle models*. The computational com-

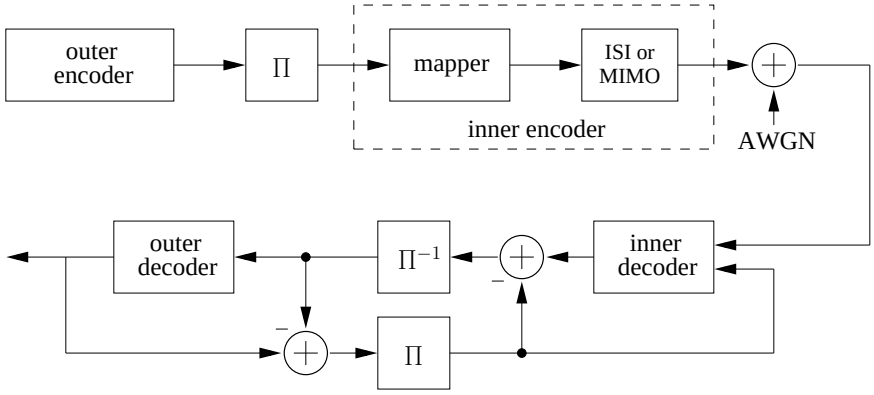

Figure 4.1: Transmitter and iterative receiver structure in a communication system with coding and ISI/MIMO.

plexity of a tree/trellis based algorithm is determined by the number of visited nodes/states. For an ISI channel of memory L and a modulation alphabet Ω, the trellis has $|\Omega|^L$ states. Consequently, detection using a BCJR algorithm for large constellations and/or ISI channels of large memory is too complex. Consider now an $N_r \times N_t$ MIMO system ($N_r \geq N_t$) where N_r is the number of receive antennas and N_t is the number of transmit antennas. It is possible to recast the case $N_r > N_t$ to $N_r = N_t$ via a QR-decomposition of the channel matrix $\boldsymbol{H}$ [108]. More details are given in Chapter 5. Hence, in the rest of the chapter it is assumed that $N_r = N_t = N$. An $N \times N$ MIMO system can now be graphically visualized as a tree of depth N with $|\Omega|$ outgoing branches per node. Since the number of leaf nodes is $|\Omega|^N$, optimum detection can be applied only in rather small setups. When N and/or Ω are large, suitable alternatives are suboptimum algorithms which effectively reduce the tree/trellis state space. For ISI there are various solutions in the literature, e.g., the DFSE [109], the RS-BCJR [93], the T-BCJR [13], the M-BCJR [13] and improved versions [95, 110] including our backup M-BCJR proposed in Chapter 3, the M*-BCJR [94] and the techniques presented in [111] and [112]. All these algorithms, except for [109] and [111], are based on the Forney model. There are many promising tree/trellis based low-complexity MIMO detectors and some examples are the soft-output sphere detector [113] and the soft-output M-algorithm (SOMA) [14], to just mention a few.

This chapter investigates the performance of reduced-complexity detectors that operate on the Ungerboeck model, and compares it with the Forney model.

Simulation-based results are presented together with an analysis of the asymptotic detector behavior. The asymptotic behavior of reduced-complexity ISI detectors in the high SNR regime was first studied in [114]; this chapter extends the analysis to the low SNR regime and to MIMO systems. In a related work by Badri-Hoeher et al. [115], a comparison of Ungerboeck and Forney models was conducted for reduced-complexity multi-user detectors. The conclusions of [115], however, do not translate to the reduced-complexity ISI detection. Motivated by the impressive performance of the M-BCJR algorithms from the previous chapter, the M-BCJR algorithm is chosen as the preferred method in this chapter. Note that the basic M-algorithm is optimal in the sense of minimizing the probability of correct path loss among the constant-complexity breadth-first search algorithms [25, 26].

The rest of the chapter is organized as follows. In Section 4.2 the system models for ISI and MIMO channels are given. Section 4.3 discusses MAP and approximate MAP methods for sequence and symbol detection. An SNR-asymptotic detector behavior analysis is performed in Section 4.4. Finally, Section 4.5 and 4.7 present numerical results for the M- and M*-BCJR algorithm respectively while conclusions are drawn in Section 4.8.

4.2 System Model

4.2.1 ISI channels

Consider a linearly modulated transmit signal whose baseband form is

$$s(t) = \sum_{k=0}^{\infty} a_k h(t - kT), \tag{4.1}$$

where $\{a_k\}$ are possibly encoded, uniformly distributed data symbols with zero mean and unit variance belonging to the alphabet Ω, while $h(t)$ is a continuous pulse which represents the combined effect of the transmit filter and the channel impulse response (CIR), generating finite ISI. Without loss of generality, it is assumed that $h(t)$ is a unit energy pulse, that is, $E_p = \int |h(t)|^2 \mathrm{d}t = 1$. Finally, ideal channel estimation at the receiver side, that is, perfect synchronization and perfect knowledge of the lSI coefficients and the noise variance is assumed.

The received signal is given by $r(t) = s(t) + n(t)$, where $n(t)$ is complex white Gaussian noise with one-sided power spectral density (PSD) N_0. Even though all simulation results for ISI channels in this chapter are based on a baseband model, i.e., 2-PAM with a real $h(t)$, the representation of MIMO channels is complex and therefore, throughout this chapter, all formulas are

given in this more general form. As already mentioned in Chapter 2, Forney showed [4] that the sampled matched filter outputs

$$x_k = \int_{-\infty}^{\infty} r(t)h^*(t - kT)\mathrm{d}t \tag{4.2}$$

form sufficient statistics to estimate the transmitted data from the received signal $r(t)$. Here $h^*(t)$ denotes the complex conjugate of $h(t)$. By inserting the expression for $r(t)$ into (4.2), the sampled matched filter outputs become

$$x_k = \sum_{l=-L}^{L} g_l a_{k-l} + \eta_k \tag{4.3}$$

where

$$\begin{aligned} g_l &= \int_{-\infty}^{\infty} h(t)h^*(t - lT)\mathrm{d}t \\ \eta_k &= \int_{-\infty}^{\infty} n(t)h^*(t - kT)\mathrm{d}t. \end{aligned} \tag{4.4}$$

Note that the noise samples η_k are no longer white and that the unit-energy assumption, $E_p = 1$, implies that $g_0 = 1$. Equation (4.3) will be referred to as the Ungerboeck observation model. Further assumed is that the autocorrelation coefficients satisfy $g_l = 0$ for $|l| > L$; optimal detection then requires $|\Omega|^L$ trellis states. The correlation of the colored noise samples η_k is $\mathbb{E}[\eta_k \eta_{k-l}^*] = g_l N_0$.

The Forney or white noise observation model is often preferred over the Ungerboeck model mostly due to the whiteness of the noise at its output. In this model the sampled MF outputs are filtered with a discrete-time whitening filter producing the sequence $\boldsymbol{y}$ in (2.54) which, for convenience, is repeated here in a slightly different form

$$y_k = \sum_{l=0}^{L} f_l a_{k-l} + w_k. \tag{4.5}$$

In (4.5), $\boldsymbol{f} = [f_0, f_1, \ldots, f_L]$ is a causal $(L+1)$-tap long ISI response sequence and the noise samples $\{w_k\}$ are independent complex Gaussians with variance $\sigma^2 = N_0$. Note that, hereinafter, bold letters are used for vectors (lower case) and matrices (upper case). In agreement with (4.4), $\boldsymbol{g}$ is the autocorrelation sequence of $\boldsymbol{f}$. The causal sequence $\boldsymbol{y}$ also forms a set of sufficient statistics.

Hence, in the case of optimal detection the two models have equivalent detection properties.

There exist many possible whitening filters with the *white-noise-at-the-output* property. However, it is well known that the minimum-phase solution is the one best suited for reduced-search decoders (see [92, 98] and the more recent [99, 112]). This is due to the fact that in a minimum-phase model, the signal energy is concentrated in the front taps, which directs the reduced search more efficiently. Therefore, throughout this chapter, when a whitening filter is used it is assumed that it results in the minimum-phase impulse response $\boldsymbol{f}$. Further improvements of the minimum phase idea are proposed in Chapter 3. However, in the same chapter it is shown that due to spectral zero regions in practical FTN signaling the Forney observation model may be prohibited. Chapter 3 gives a solution to this modeling problem which leads to white noise but without a formal WMF. Therefore, in this chapter, the notation Forney observation model includes all types of signaling that can be represented with the discrete-time model in (4.5).

An interesting fact in the Ungerboeck model is that the observations x_k contain contributions from both the past and the future L symbols, unlike in the Forney model where observations depend only on the current data symbol a_k and the L past symbols $[a_{k-1}, ..., a_{k-L}]$. This is clearly seen if (4.3) is decomposed according to

$$x_k = \underbrace{\sum_{l=-L}^{-1} g_l a_{k-l}}_{\text{“future''}} + \underbrace{a_k}_{\text{“present''}} + \underbrace{\sum_{l=1}^{L} g_l a_{k-l}}_{\text{“past''}} + \eta_k. \tag{4.6}$$

One of the objectives of this chapter is to show that this fundamental difference has a crucial effect on the BER performance of reduced-complexity tree/trellis decoders.

4.2.2 MIMO channels

Next consider transmission of a linearly modulated signal through a MIMO channel affected by additive white Gaussian noise (AWGN). The received signal sample at kth time instance is given by

$$\boldsymbol{y}_k = \boldsymbol{H}\boldsymbol{a}_k + \boldsymbol{w}_k \tag{4.7}$$

where $\boldsymbol{a}_k = [a_{k,1}, a_{k,2}, \ldots, a_{k,N}]^{\mathrm{T}}$ are modulation symbol tuples chosen from an alphabet Ω^N and $\boldsymbol{H}$ is the channel matrix representing the $N \times N$ MIMO system. It is assumed that all coefficients $\{h_{i,j}\}$ are independent and identically distributed (IID) complex Gaussians with unit variance (1/2 in each dimension), denoted as $\mathcal{CN}(0,1)$ and $\boldsymbol{w}_k$ is IID $\mathcal{CN}(0, N_0\boldsymbol{I})$. Note that, as already pointed out in Chapter 2, the ISI model (4.5) can be seen as a special case of the MIMO model (4.7) where the channel matrix $\boldsymbol{H}$ has the following form:

$$\boldsymbol{H} = \begin{bmatrix} f_0 & 0 & & 0 & 0 \\ \vdots & f_0 & \ddots & 0 & 0 \\ f_L & \vdots & \ddots & \vdots & 0 \\ 0 & f_L & \ddots & \vdots & \vdots \\ \vdots & 0 & \ddots & \vdots & \vdots \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ \vdots & \vdots & \ddots & 0 & \vdots \\ \vdots & \vdots & \ddots & f_0 & 0 \\ 0 & \vdots & \ddots & \vdots & f_0 \\ 0 & 0 & \ddots & f_L & \vdots \\ 0 & 0 & & 0 & f_L \end{bmatrix}.$$

Although the results for MIMO channels can be extended to other modulation alphabets, this analysis is restricted to uniformly distributed symbols drawn from the M-QAM alphabet $\Omega = \{\pm A \pm Bj\}$ where

$$A, B \in \sqrt{\frac{3}{2(M-1)}}\{1, 3, ..., (\sqrt{M}-1)\}.$$

Note that the M-QAM alphabet above has been properly normalized such that

$$\mathbb{E}[|a_{k,l}|^2] = 1.$$

As before, the MIMO channel is assumed to be perfectly known at the receiver. After a QL-decomposition, model (4.7) can equivalently be expressed as (henceforth the index k is omitted)

$$\tilde{\boldsymbol{y}} = \boldsymbol{L}\boldsymbol{a} + \tilde{\boldsymbol{w}} \tag{4.8}$$

where $\tilde{\boldsymbol{y}} = \boldsymbol{Q}^\dagger \boldsymbol{y}$, $\tilde{\boldsymbol{w}} = \boldsymbol{Q}^\dagger \boldsymbol{w}$ and $\boldsymbol{H} = \boldsymbol{Q}\boldsymbol{L}$. The matrix $\boldsymbol{L}$ is lower triangular and "†" denotes the Hermitian transpose operator. Since the nth entry of $\tilde{\boldsymbol{y}}$ equals

$$\tilde{y}_n = \sum_{m=0}^{n} \ell_{nm} a_m + \tilde{w}_n, \tag{4.9}$$

$\tilde{\boldsymbol{y}}$ can be represented with a tree of depth N with in total $|\Omega|^N$ leaf nodes.

While (4.9) is the MIMO counterpart of the Forney ISI-signal in (4.5) it is also possible to define the MIMO counterpart of the Ungerboeck ISI-signal in (4.3) by a matrix multiplication with $\boldsymbol{L}^\dagger$, that is

$$\boldsymbol{x} = \boldsymbol{L}^\dagger \tilde{\boldsymbol{y}}. \tag{4.10}$$

In fact, a further strength of the Ungerboeck representation of the MIMO channel in (4.7) is that there is no need to perform any QL- or QR-decomposition [116]. Instead the vector $\boldsymbol{x}$ in (4.10) can be obtained directly by a matrix multiplication of (4.7) with $\boldsymbol{H}^\dagger$, that is

$$\boldsymbol{x} = \boldsymbol{H}^\dagger \boldsymbol{y} = \boldsymbol{G}\boldsymbol{a} + \boldsymbol{\eta} \tag{4.11}$$

where we have defined $\boldsymbol{G} \triangleq \boldsymbol{H}^\dagger \boldsymbol{H}$ and where $\boldsymbol{H}^\dagger \boldsymbol{w} = \boldsymbol{\eta}$. Note that $\boldsymbol{G} = \boldsymbol{H}^\dagger \boldsymbol{H} = \boldsymbol{L}^\dagger \boldsymbol{L}$. In the remainder of this chapter it is assumed that a QL-decomposition of the channel matrix $\boldsymbol{H}$ is always performed and hence the notation $\boldsymbol{y}$ is always used instead of $\tilde{\boldsymbol{y}}$.

4.3 Optimum and Suboptimum Detection

This section discusses MAP and approximate MAP methods for sequence and symbol detection. In Section 4.3.1 the path metric for MAP sequence detection is given for both ISI and MIMO channels while Section 4.3.2 introduces the so-called middle models. Finally, in Section 4.3.3, optimum and suboptimum symbol detection is considered.

4.3.1 Path Metrics for MAP Sequence Detection

ISI Channels

According to Section 2.4, a MAP *sequence* detector finds the most probable data sequence $\hat{\boldsymbol{a}}$ such that

$$\hat{\boldsymbol{a}} = \arg\max_{\boldsymbol{a}} \Pr(\boldsymbol{a}|\boldsymbol{x}) = \arg\max_{\boldsymbol{a}} \Pr(\boldsymbol{a}|\boldsymbol{y}) \tag{4.12}$$

where $\boldsymbol{x}$ and $\boldsymbol{y}$ are received vectors containing Ungerboeck and Forney observations (see (4.3) and (4.5)) while $\Pr(\cdot|\cdot)$ denotes a conditional probability mass function. The whiteness of the noise samples at the receiver in the Forney observation model allows the factorization (2.86) where each term $p(y_k|\boldsymbol{a})$ is given by (2.87). Combined with the data independence assumption, the branch metric for ISI channels at kth trellis stage is proportional to $\Pr(a_k)p(y_k|\boldsymbol{a})$. This chapter works in the log-domain so that (2.87) becomes

$$\log p(y_k|\boldsymbol{a}) \propto -\frac{1}{N_0}\left|y_k - \sum_{l=0}^{L} f_l a_{k-l}\right|^2. \tag{4.13}$$

Note that, for the optimization problem in (4.12), the term $1/N_0$ can be removed if the inputs are equiprobable. However, since this need not be the case in general, $1/N_0$ will be kept throughout the chapter.

The colored noise in the Ungerboeck model prohibits the factorization in the form of (2.86). Nevertheless, the log-likelihood $\log p(\boldsymbol{x}|\boldsymbol{a})$ can still be factorized [21] as

$$\log p(\boldsymbol{x}|\boldsymbol{a}) \propto \sum_{k=0}^{\mathcal{B}-1} \tilde{\varphi}(x_k|\boldsymbol{a}) \tag{4.14}$$

where $\mathcal{B}$ is the length of the received sequence and where, for ISI channels, $\tilde{\varphi}(x_k|\boldsymbol{a})$ is given by

$$\tilde{\varphi}(x_k|\boldsymbol{a}) \triangleq \frac{2}{N_0} a_k^* \left(x_k - \frac{a_k}{2} - \sum_{l=1}^{L} g_l a_{k-l}\right). \tag{4.15}$$

MIMO Channels

Consider now MIMO channels represented by (4.9). The corresponding log-likelihood $\log p(\boldsymbol{y}|\boldsymbol{a})$ is given by

$$\log p(\boldsymbol{y}|\boldsymbol{a}) \propto -\frac{\|\boldsymbol{y}-\boldsymbol{L}\boldsymbol{a}\|^2}{N_0}. \tag{4.16}$$

From (4.8) we have the following commonly used factorization

$$\log p(\boldsymbol{y}|\boldsymbol{a}) \propto \sum_{k=1}^{N} -\frac{|y_k-(\boldsymbol{L}\boldsymbol{a})_k|^2}{N_0} \tag{4.17}$$

which allows a straightforward implementation of Forney-based MAP detection. Note that eq. (4.17) is the MIMO counterpart (in log-domain) of the Forney metric in ((2.86) and (2.87)) for ISI channels. By using the definition $\boldsymbol{G}=\boldsymbol{L}^{\dagger}\boldsymbol{L}$ and reminding the reader that $\boldsymbol{x}$ is the Ungerboeck MIMO signal in (4.10), we obtain the following alternative expression of (4.16):

$$\begin{aligned}
\log p(\boldsymbol{y}|\boldsymbol{a}) \propto & -\frac{\|\boldsymbol{y}-\boldsymbol{L}\boldsymbol{a}\|^2}{N_0} \\
= & -\frac{\boldsymbol{y}^{\dagger}\boldsymbol{y}-2\mathcal{R}\{\boldsymbol{y}^{\dagger}\boldsymbol{L}\boldsymbol{a}\}+\boldsymbol{a}^{\dagger}\boldsymbol{G}\boldsymbol{a}}{N_0} \\
= & \frac{1}{N_0}\sum_{k=1}^{N}\Big[-|y_k|^2+g_{kk}|a_k|^2-2\mathcal{R}\{x_k a_k^*\} \\
& +2\mathcal{R}\{a_k^*\sum_{m=1}^{k-1}g_{mk}a_m\}\Big] \\
= & \frac{1}{N_0}\sum_{k=1}^{N}\left[-|y_k|^2+\left(\sum_{n=k}^{N}|\ell_{nk}|^2\right)|a_k|^2\right. \\
& \left.+2\mathcal{R}\left\{a_k^*\left[\sum_{m=1}^{k-1}\left(\sum_{n=k}^{N}\ell_{nk}^*\ell_{nm}\right)a_m-x_k\right]\right\}\right],
\end{aligned} \tag{4.18}$$

where $\mathcal{R}\{\cdot\}$ denotes the real part of a complex number. Similarly to ISI channels, we now move from the notation $\log p(\boldsymbol{y}|\boldsymbol{a})$ to $\log p(\boldsymbol{x}|\boldsymbol{a})$ where the latter can be expressed as

$$\log \Pr(\boldsymbol{x}|\boldsymbol{a}) \propto \sum_{k=1}^{N} \tilde{\varphi}_k(x_k|a_k, ..., a_1) \tag{4.19}$$

where

$$\tilde{\varphi}_k(x_k|a_k, ..., a_1) = -\frac{1}{N_0}\left[g_{kk}|a_k|^2 + 2\mathcal{R}\left\{a_k^*\left[\sum_{m=1}^{k-1} g_{mk}a_m - x_k\right]\right\}\right]. \tag{4.20}$$

This type of recursive factorization is the MIMO counterpart of the Ungerboeck metric for ISI channels (4.14). It was first applied to MIMO setups in [116]. Since $\tilde{\varphi}_k(x_k|a_k, ..., a_1)$ is only a function of current and past symbols, the same tree as in Forney-based detection can be used to calculate $\log p(\boldsymbol{x}|\boldsymbol{a})$.

$$
\begin{aligned}
k &= 1: \quad -\frac{1}{N_0}|y_1-\ell_{11}a_1|^2 = -\frac{1}{N_0}\Big(|y_1|^2+|a_1|^2|\ell_{11}|^2-2\mathcal{R}\{y_1^*\ell_{11}a_1\}\Big)\\
k &= 2: \quad -\frac{1}{N_0}|y_2-\ell_{21}a_1-\ell_{22}a_2|^2 = -\frac{1}{N_0}\Big(|y_2|^2+|a_2|^2|\ell_{22}|^2\\
&\quad +\underbrace{|a_1|^2|\ell_{21}|^2-2\mathcal{R}\{y_2^*\ell_{21}a_1\}}_{\text{UBM1}}-2\mathcal{R}\{y_2^*\ell_{22}a_2\}+2\mathcal{R}\{a_1^*\ell_{21}^*\ell_{22}a_2\}\Big)\\
k &= 3: \quad -\frac{1}{N_0}|y_3-\ell_{31}a_1-\ell_{32}a_2-\ell_{33}a_3|^2 = -\frac{1}{N_0}\Big(|y_3|^2+|a_3|^2|\ell_{33}|^2\\
&\quad +\underbrace{|a_1|^2|\ell_{31}|^2-2\mathcal{R}\{y_3^*\ell_{31}a_1\}}_{\text{UBM1}}-2\mathcal{R}\{y_3^*\ell_{33}a_3\}+2\mathcal{R}\{a_1^*\ell_{31}^*\ell_{33}a_3\}\\
&\quad +\underbrace{|a_2|^2|\ell_{32}|^2-2\mathcal{R}\{y_3^*\ell_{32}a_2\}+2\mathcal{R}\{a_1^*\ell_{31}^*\ell_{32}a_2\}}_{\text{UBM2}}\\
&\quad +2\mathcal{R}\{a_2^*\ell_{32}^*\ell_{33}a_3\}\Big).
\end{aligned}
$$

4.3.2 Middle Models

Next, it is shown that there exist other models, middle models, where the tree/trellis metrics are calculated in a different manner than in the Forney and Ungerboeck observation models. For simplicity this section begins with a 3×3 MIMO setup example for which the Forney and Ungerboeck metrics at different tree depths are calculated. The input-output channel model (4.17) is assumed. Now the Forney branch metric at kth tree depth is given by the expressions at the top of this page.

If the terms marked UBMk are instead calculated at the kth tree depth the Ungerboeck model is obtained, see (4.20). Note that, in the Forney observation model, the computation of the tree/trellis metrics which depend on a specific symbol a_k is spread over the time instances $k, k+1, ..., N$ while in the Ungerboeck model the computation is performed as early as possible. The middle models are models working in between these extremes. More specifically, in a middle model with an offset p, the computation of the marked terms is performed p steps earlier than in a Forney model. The following metric is

computed in a general p offset middle model at tree/trellis depth m:

$$-\frac{1}{N_0}\left(\sum_{n=1}^{m-1}\left(|a_n|^2|\ell_{m+p,n}|^2-2\mathcal{R}\{y_m^*\ell_{m+p,n}a_n\}\right)\right.$$
$$+\sum_{q=0}^{p}\left(|a_m|^2|\ell_{m+q,m}|^2-2\mathcal{R}\{y_{m+q}^*\ell_{m+q,m}a_m\}\right)$$
$$+\sum_{n=1}^{m-1}\sum_{t=1}^{m-n-1}2\mathcal{R}\{a_n^*\ell_{m+p,n}^*\ell_{m+p,n+t}a_{n+t}\}$$
$$\left.+\sum_{n=1}^{m-1}\sum_{q=0}^{p}2\mathcal{R}\{a_n^*\ell_{m+q,n}^*\ell_{m+q,m}a_m\}\right).$$

Observe that the term $-|y_m|^2/N_0$ has been left out since it has no influence on the detection outcome, i.e., it is constant with respect to $\boldsymbol{a}$. The middle model above is equivalent to the Forney model if $p=0$ while the Ungerboeck model is obtained when $p=N$ (assuming a $N\times N$ MIMO system). In Section 4.5 middle models are evaluated and compared to the two already established models when suboptimum reduced-complexity algorithms are used.

4.3.3 MAP Symbol Detection

This section considers optimum and suboptimum MAP symbol detection. According to Section 2.4, a MAP symbol detector minimizes the symbol error probability by finding, at each time instant k, the most probable symbol $\hat{a}_k$ according to

$$\hat{a}_k=\arg\max_{a_k}\Pr(a_k|\boldsymbol{x})=\arg\max_{a_k}\Pr(a_k|\boldsymbol{y})$$

where $\boldsymbol{x}$ and $\boldsymbol{y}$ are vectors of received observations from (4.3) and (4.5). In iterative detection, soft information in the form of logarithmic a posteriori probability (APP) ratios, is exchanged between the component blocks. If, for notational simplicity, binary 2-PAM signaling is assumed, i.e. $\Omega=\{\pm1\}$, the logarithmic APP ratio, provided by a MAP symbol detector, can be expressed as

$$L(a_k)\triangleq\log\left(\frac{\Pr(a_k=+1|\boldsymbol{x})}{\Pr(a_k=-1|\boldsymbol{x})}\right)=\log\left(\frac{\sum_{\boldsymbol{a}:a_k=+1}\Pr(\boldsymbol{a}|\boldsymbol{x})}{\sum_{\boldsymbol{a}:a_k=-1}\Pr(\boldsymbol{a}|\boldsymbol{x})}\right).\qquad(4.23)$$

For ISI and MIMO channels, the MAP symbol detector can be efficiently realized using the BCJR algorithm [11] based on the factorization (2.86). A BCJR-type algorithm, based on (2.88), was derived in [34] and it was shown that its output is equivalent to the output of a Forney-model-based BCJR.

Since the BCJR algorithm is often prohibitively complex, alternative detection methods with an acceptable complexity-performance tradeoff are needed. This section describes the M-BCJR algorithm, which is the reduced-search method used in this chapter. It is similar to the Chapter 3 simple detection M-BCJR with minor differences in steps IV and VI. For reading convenience the adopted M-BCJR is again given in steps. However, in this chapter it is described without the binary symbol alphabet assumption made in Chapter 3.

A state σ_k is, in case of ISI channels, defined by the L (N for MIMO channels) most recent symbols, i.e. $\sigma_k = [a_{k-L}, \ldots, a_{k-1}]$. Same notations as in Chapters 2 and 3 are used; the recursively calculated forward and backward tree/trellis metrics of the state σ at kth tree/trellis depth are denoted $\alpha_k(\sigma)$ and $\beta_k(\sigma)$ respectively and the metric at time k of the branch connecting the states (σ, σ') is denoted $\gamma_k(\sigma, \sigma')$. The M-BCJR algorithm finds the largest M $\alpha_k(\sigma)$ and $\beta_k(\sigma)$ and based on these, it computes the logarithmic APP ratios (4.23) in the same manner as the BCJR algorithm. For simplicity, the Forney observation model (4.5) and ISI channels are assumed but it is straightforward to extend all steps to the Ungerboeck model and/or MIMO setups. Recursions start and end in the all-zero state. The adopted M-BCJR proceeds as follows:

Forward Recursion. Starting at $k = 0$, perform at stage $1, 2, \ldots, (\mathcal{B} + L - 1)$:

I The forward recursion is computed from the M nonzero values retained in $\boldsymbol{\alpha}_k$. There are M outcomes corresponding to each symbol in Ω; only the $|\Omega|M$ corresponding branch metrics γ are computed and stored.

II Trellis paths may merge at stage $k+1$. The algorithm detects and removes merges, leaving only one survivor per node whose α value is the sum of the incoming values.

III M largest α values are found and stored for stage $k + 1$ and for the β recursion.

Backward Recursion. Starting at $k = (\mathcal{B} + L - 1)$, perform at stage $(\mathcal{B} + L - 2), (\mathcal{B} + L - 3), \ldots, 2$:

IV The backward recursion is computed from the M nonzero values retained in $\boldsymbol{\beta}_{k+1}$ over the branches from step *I*. There are in total $|\Omega|M$ outcomes at each depth.

V Trellis paths may merge at stage k. The algorithm detects and removes merges, leaving only one survivor whose β value is the sum of the incoming values.

VI M largest β values are found and stored for stage $k-1$ and for the completion stage. If the backward metric of a state σ at depth k equals 0, it is replaced with $\beta_{k,\min}$ where $\beta_{k,\min}$ is the smallest non-zero backward metric in the M-list at depth k. Unlike the simple detection M-BCJR from Chapter 2, the M-BCJR used in this chapter does not need to retain β paths if their state and stage overlap with that of a stored α.

Completion. Starting at $k = 0$, perform at stage $0, 1, \ldots, (\mathcal{B} + L - 1)$:

VII Compute the approximate logarithmic APP from (4.23). If the sum in either the numerator or the denominator equals 0, i.e., there is no overlap between $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$, the respective sum is set to ϵ, where ϵ is a backup constant set a priori. In the simulations $\epsilon = 10^{-12}$.

A different algorithm, the M*-BCJR algorithm, is considered in Section 4.6. This reduced complexity MAP detector was proposed in [94] and shows very good performance on ISI channels. The algorithm retains M states at each trellis depth, but rather than eliminating the remaining states they are merged into the M survivor states. Section 4.6 presents and evaluates different merging strategies while Section 4.7 treats the effect of Ungerboeck and Forney metrics on the M*-BCJR performance.

4.4 The Asymptotic SNR Regime

The asymptotic behavior of the two observation models in the high and low SNR regimes is analyzed in this section. For simplicity a real and unit energy ISI pattern with $g_0 = 1$ is assumed together with 2-PAM data sequences $\boldsymbol{a}$ and an M-algorithm with $M = 1$. Note that in the $M = 1$ case there is no trellis pruning, i.e., only a single path is explored which is equivalent to detection with a standard decision feedback (DF) equalizer. It is further assumed that the current tree/trellis state is correct, i.e., it is assumed that the detector holds the correct path $\boldsymbol{a}_{[0,k-1]}$ at tree/trellis depth $k-1$. The high SNR asymptotic regime was considered in [114]. For completeness the main results are summarized here, before proceeding with the study of the low SNR regime.

The high SNR case ($N_0 \to 0$) and the Forney observation model (4.5) are studied first. The received signal sample at kth trellis depth, can be expressed as

$$y_k = f_0 a_k + \sum_{l=1}^{L} f_l a_{k-l} + w_k.$$

Since correct tree/trellis state at stage $k-1$ is assumed, there will be no influence from the past symbols, i.e., the term $\sum_{l=1}^{L} f_l a_{k-l}$ in y_k can be removed. In the high SNR regime, the noise w_k vanishes and consequently the kth received signal sample equals

$$y_k = f_0 a_k.$$

The SNR for determining a_k based on y_k is hence infinite. The $M = 1$ case in the M-algorithm is thus sufficient for correct sequence detection when the Forney observation model is used. As we will show next, this may not be the case when the Ungerboeck observation model is adopted.

Now consider the Ungerboeck observation model with the same assumptions. The influence from the past symbols ($\sum_{l=1}^{L} g_l a_{k-l}$) can be removed from (4.6) and with vanishing noise η_k, the kth received signal sample x_k reduces to

$$x_k = a_k + \sum_{l=-L}^{-1} g_l a_{k-l}.$$

A correct path loss in the M-algorithm with $M = 1$ at trellis stage k can thus occur if

$$\left| \sum_{l=-L}^{-1} g_l a_{k-l} \right| > |a_k|.$$

Since the maximum of the left-hand side is $\sum_{l=-L}^{-1} |g_l a_{k-l}|$, a sufficient[1] and necessary condition for a correct path loss with $M = 1$ is:

[1]Note that (4.24) is a sufficient condition that a correct path loss can possibly occur. With long input blocks, the probability goes to one that the L most recent symbols are such that a correct path loss occurs somewhere in the block.

$$\sum_{l=-L}^{-1} |g_l| - 1 > 0. \tag{4.24}$$

Condition (4.24) is fulfilled by ISI responses of *closed-eye* type. Consequently, for a long input block in the Ungerboeck observation model and in the high SNR case, the sequence $\hat{\boldsymbol{a}}$, estimated by the M-algorithm with $M = 1$, is equivalent to the transmitted sequence $\hat{\boldsymbol{a}} = \boldsymbol{a}$ if and only if the considered ISI is of *open-eye* type, i.e.,

$$\sum_{l=L}^{-1} |g_l| < 1. \tag{4.25}$$

Hence, in the high SNR regime, a significant performance difference between Forney and Ungerboeck detectors (in favor of Forney) is expected.

By a straightforward extension of (4.24) into a MIMO setup, the following condition for a correct path loss at any depth with $M = 1$ is obtained:

$$\sum_{\ell=k+1}^{N} |g_{k\ell}| - g_{kk} > 0, \quad \forall k, \tag{4.26}$$

where $g_{k\ell}$ are the elements of the matrix $\boldsymbol{G}$.

Now consider the low SNR case. The signal to interference plus noise ratio (SINR) has a crucial impact on the branching procedure in a signal tree/trellis. If the Forney observation model (4.5) and $M = 1$ are assumed, the SINR at any trellis stage is given by

$$\mathrm{SINR_F} = \frac{|f_0|^2}{N_0}.$$

In the Ungerboeck observation model (4.3), on the other hand, the SINR becomes

$$\mathrm{SINR_U} = \frac{|g_0|^2}{g_0 N_0 + I_\mathrm{f}}$$

where I_f is the interference energy from the "future" symbols in (4.6), i.e.,

$$I_{\mathrm{f}} = \sum_{l=-L}^{-1} |g_l|^2. \tag{4.27}$$

If now N_0 approaches infinity, it can be observed that $\mathrm{SINR_U} \geq \mathrm{SINR_F}$ since $g_0 = \sum_{l=0}^{L} |f_l|^2 \geq |f_0|^2$. Consequently, in the low SNR regime, the probability of correct path loss is smaller in the Ungerboeck observation model. This will be confirmed by simulation results in Section 4.5.

At depth k in a MIMO setup, the SINR expressions translate into

$$\mathrm{SINR_F} = \frac{|\ell_{kk}|^2}{N_0}.$$

and

$$\mathrm{SINR_U} = \frac{|g_{kk}|^2}{g_{kk}N_0 + I_f}$$

and consequently the same conclusions as in the ISI case hold.

4.5 M-BCJR Receiver Tests

This section presents receiver test results which provide insights into the differences between the observation models when reduced complexity M-BCJR detection is used. The following unit-energy ISI channel models have been used in the tests:

$$\boldsymbol{f} = [\mathbf{.2448}, \mathbf{.4774}, \mathbf{.6868}, \mathbf{.4428}, \mathbf{.2106}] \tag{4.28}$$

$$\boldsymbol{f} = [\mathit{-.0049}, \mathit{-.0028}, \mathit{.0069}, \mathit{-.0109}, \mathit{-.0007}, \mathit{.0341}, \mathit{-.0185}, \mathit{.0034}, \mathbf{.3746}, \mathbf{.7408}, \mathbf{.4989}, \mathbf{-.0700}, \mathbf{-.2143}, \mathbf{.0187}, \mathbf{.0873}, \mathbf{-.0196}, \mathbf{-.0277}, \mathbf{.0168}] \tag{4.29}$$

$$\boldsymbol{f} = [\mathit{.0248}, \mathit{.0122}, \mathit{-.0243}, \mathit{.0076}, \mathbf{.1910}, \mathbf{.4642}, \mathbf{.6230}, \mathbf{.5063}, \mathbf{.1763}, \mathbf{-.1226}, \mathbf{-.1965}, \mathbf{-.0746}, \mathbf{.0604}, \mathbf{.0797}, \mathbf{.0134}, \mathbf{-.0347}, \mathbf{-.0222}] \tag{4.30}$$

$$\boldsymbol{f} = [\mathbf{.5000}, \mathbf{.5000}, \mathbf{-.5000}, \mathbf{-.5000}] \tag{4.31}$$

The length-K_p taps whose values are written in italic in (4.29) and (4.30) are, in the case of Forney observation model, replaced with zeros in the detector which then works at a delay K_p; including these taps will increase the complexity without almost any improvement of the BER performance. The model (4.28) is the minimum phase equivalent of the standard memory $L = 4$ Proakis-C

channel. It has $d^2_{\min} = 0.63$ which is a 5.02 dB loss compared to binary 2-PAM signaling. Models (4.29) and (4.30) are super minimum phase discrete time models of continuous FTN signals which appeared in Chapter 3. They correspond to FTN signaling with the 30% rRC $h(t)$ when $\tau = 1/2$ and 0.35 respectively. As already mentioned earlier in this chapter, in practical FTN signaling the WMF model may be prohibited due to spectral zero regions. However, in this chapter all types of signaling that can be represented using the discrete-time model in (4.5) is included in the Forney observation model. The last model (4.31) is the EPR4 channel which has $d^2_{\min} = 2$.

The corresponding ISI channel models in the Ungerboeck observation model are given in (4.32)-(4.35). Note that only $L+1$ ISI taps are shown $[g_0, \ldots, g_L]$ since the autocorrelation $\boldsymbol{g}$ satisfies $g_{-k} = g_k^*$, for all k.

$$\boldsymbol{g} = [\mathbf{1}, .\mathbf{8421}, .\mathbf{5242}, .\mathbf{2089}, .\mathbf{0516}] \tag{4.32}$$

$$\boldsymbol{g} = [\mathbf{1}, .\mathbf{6236}, .\mathbf{0003}, -.\mathbf{1754}, .\mathbf{0000}, .\mathbf{0731}, -.\mathbf{0009}, -.\mathbf{0282}, .\mathbf{0015}, .\mathbf{0064}, \\ -.\mathbf{0041}, -.\mathbf{0011}, .\mathbf{0024}, -.\mathbf{0002}, -.\mathbf{0007}, .\mathbf{0003}, .\mathbf{0001}, -.\mathbf{0001}] \tag{4.33}$$

$$\boldsymbol{g} = [\mathbf{1.8026}, .\mathbf{3546}, -.\mathbf{0424}, -.\mathbf{1848}, -.\mathbf{1015}, .\mathbf{0332}, .\mathbf{0830}, .\mathbf{0425}, \\ -.\mathbf{0146}, -.\mathbf{0340}, -.\mathbf{0199}, -.\mathbf{0024}, .\mathbf{0028}, .\mathbf{0004}, -.\mathbf{0011}, -.\mathbf{0006}] \tag{4.34}$$

$$\boldsymbol{g} = [\mathbf{1}, .\mathbf{2500}, -.\mathbf{5000}, -.\mathbf{2500}] \tag{4.35}$$

The section now begins with uncoded transmission over ISI channels corresponding to the inner encoder part of Figure 4.1.

4.5.1 Uncoded Transmission over ISI/MIMO Channels

M-BCJR Results for Uncoded ISI

Consider uncoded 2-PAM transmission over the channel (4.28). The information block length used in the simulations is $\mathcal{B} = 5000$ and the algorithm decides on the symbols based on the sign of its LLR output. Figure 4.2 shows BER performance (versus E_s/N_0) of the M-BCJR detectors, based on the two observation models, for different values of M. Note that the average symbol energy E_s is normalized to 1 in all simulation setups. Solid lines correspond to Forney model while dotted lines correspond to Ungerboeck model. Clearly, there is a crossover point between the two; at low SNR levels the Ungerboeck model prevails while the situation to the right of the crossover points (higher SNR) is reversed. In order to visually clarify these points for two values of M, a small part of the figure is enlarged; the crossover points for $M = 4$ and 6 are at $E_s/N_0 \approx 2$ and 4 dB respectively. For larger M these move to a higher SNR and a lower BER. Note that at $E_s/N_0 = 14$ dB, the M-BCJR based on the

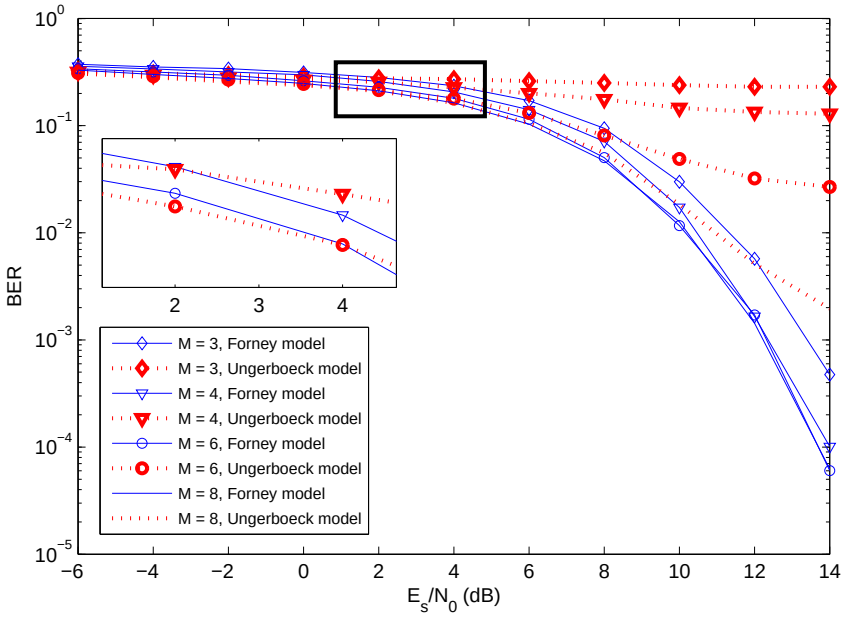

Figure 4.2: Uncoded BER performance of the Forney- and Ungerboeck-based M-BCJR detectors for different values of M, 2-PAM inputs and the 5-tap Proakis-C ISI channel.

Forney observation model with $M = 8$ is only ≈ 0.2 dB away from the BER estimate $Q(\sqrt{.63E_s/N_0})$ while the M-BCJR based on the Ungerboeck model has a much higher BER.

Figure 4.3 shows results for more severe ISI of the closed-eye type, i.e., the $\tau = 0.35$ FTN ISI model. It is observed that the Ungerboeck-based detector completely fails at medium and high SNR levels, suffering from a high error floor. This error floor, which confirms the conclusions from Section 4.4, is eliminated only when the number of preserved states M approaches the full-complexity value. Even though both observation models generate equivalent outputs with optimal detection, these results confirm that it is important to choose an appropriate model when reduced-complexity detection is performed. The M-BCJR based on the Forney model with $M \geq 8$ is, at $E_s/N_0 = 14$ dB, again very close to the estimate $Q(\sqrt{.56E_s/N_0}) \approx 3 \times 10^{-5}$.

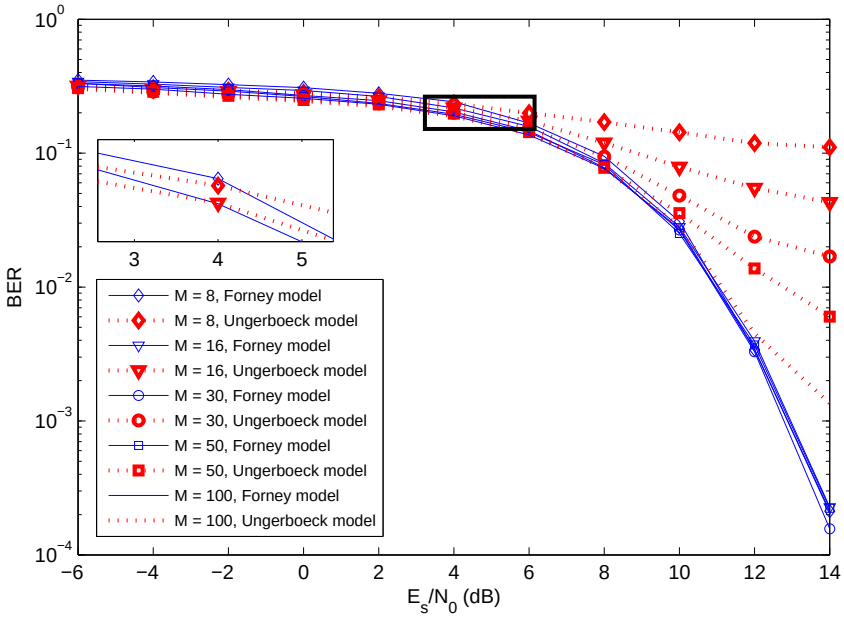

Figure 4.3: Uncoded BER performance of the Forney- and Ungerboeck-based M-BCJR detectors with different values of M for the $\tau = 0.35$ FTN channel and 2-PAM inputs.

M-BCJR Results for Uncoded MIMO

In the case of ISI channels it was assumed that the channel state information was perfectly known. The ISI channel is time invariant and hence the crossover points between the two observation models could be predicted precisely enough using the average SNR value. In MIMO channels on the other hand the average SNR is fixed but the instantaneous SNR can vary significantly from one channel realization to another. Therefore a method for choosing the best observation model for each realization of $\boldsymbol{H}$ is needed in order to improve the overall system performance. In addition to the instantaneous SNR we have looked at other channel properties that might aid us in the choice of model.

Consider now a 4×4 MIMO setup with 4-QAM inputs. Two relevant channel properties of the MIMO channel (4.7) are the condition number

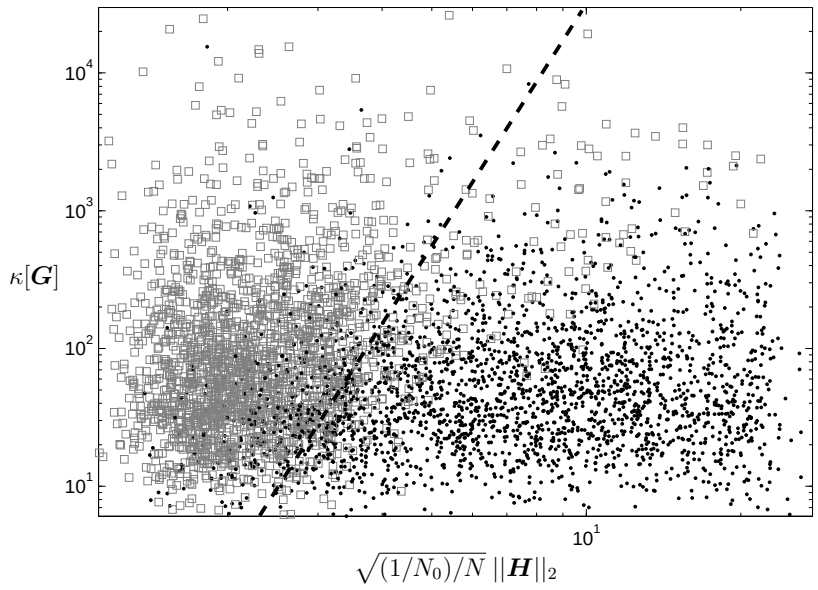

Figure 4.4: Channel realization statistics in loglog scale, generated with an M-BCJR detector with $M = 3$, for optimal (in BER sense) choice of observation model in a 4×4 MIMO setup with 4-QAM inputs. Black dots correspond to Forney model while grey squares correspond to Ungerboeck model. The dashed line illustrates the border where the BER performance of both observation models is almost identical (on average); to the left of this line the optimal choice is the Ungerboeck model, while to the right it is the Forney model.

$$\kappa[\boldsymbol{G}] = \frac{\lambda_{MAX}[\boldsymbol{G}]}{\lambda_{MIN}[\boldsymbol{G}]}$$

where $\lambda_{MAX}[\boldsymbol{G}]$ and $\lambda_{MIN}[\boldsymbol{G}]$ are the maximal and minimal eigenvalues of $\boldsymbol{G}$, and the channel realization energy defined as $\sqrt{(1/N_0)/N}\ ||\boldsymbol{H}||_2$. Note that 2-norm has been applied in the calculation of the condition number.

Figure 4.4 plots the optimal choice of model versus channel realization properties in loglog scale, generated with an M-BCJR detector with $M = 3$. The optimal choice (in the BER sense) of observation model is in the case of Forney model represented with black dots while grey squares correspond to Ungerboeck model. For every channel matrix $\boldsymbol{H}$ realization, the error rate results

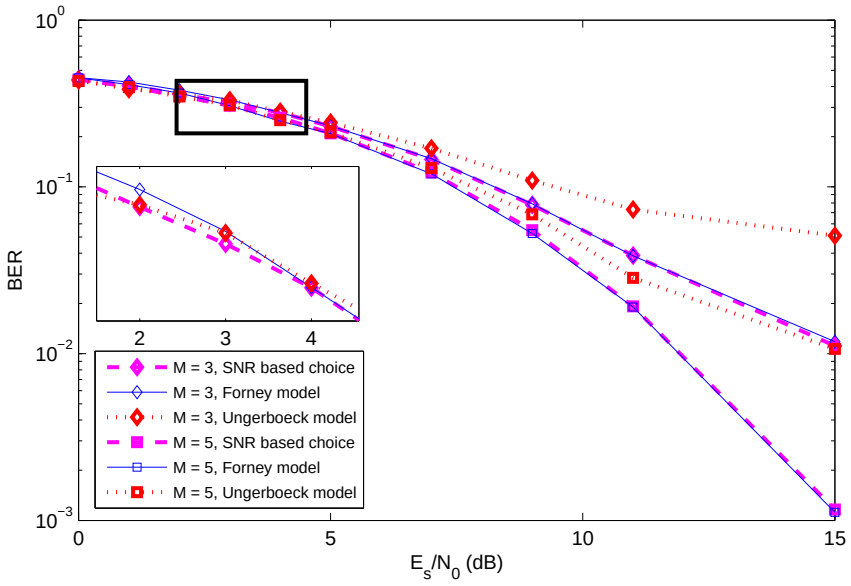

Figure 4.5: Uncoded BER performance of the Forney- and Ungerboeck-based M-BCJR detectors with different values of M in a 4×4 MIMO setup with 4-QAM inputs.

have been averaged over at least 500 noise realizations. In order to distinguish the choice of model, we have divided the plot into disjoint regions using straight lines (linear functions with different slope) in the log-domain. The dashed line illustrates the border where the BER performance of both observation models is almost identical (on average); to the left of this line the optimal choice is the Ungerboeck model, while to the right it is the Forney model.

Our tests show that there are almost no gains in choosing the best model using the condition number compared to the case where only channel realization energy is considered (a straight vertical line). Hence we conjecture that the best choice of observation model (Forney or Ungerboeck model) is only very weakly dependent on the structure of the channel matrix $\boldsymbol{H}$ but is strongly dependent on the instantaneous SNR. Dividing the plot based on only the channel realization energy will be referred to as the *energy-based method*. Other channel properties could possibly improve upon the energy-based method but they are not explored further. Figure 4.5 shows BER performance of M-BCJR

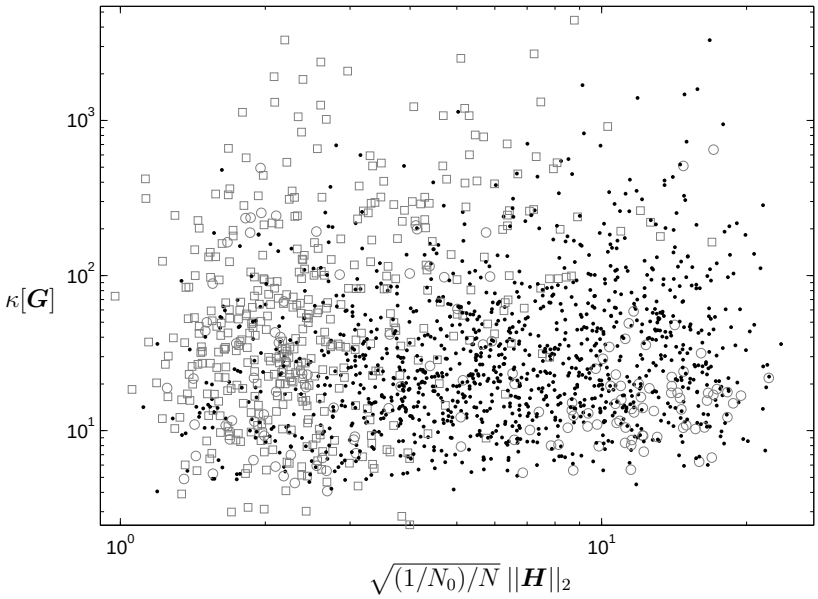

Figure 4.6: Channel realization statistics in loglog scale, generated with an M-BCJR detector with $M = 3$, for optimal (in BER sense) choice of observation model in a 4×4 MIMO setup with 4-QAM inputs. Black dots correspond to Forney model, grey squares correspond to Ungerboeck model while grey circles correspond to the middle model with offset $p = 1$.

detectors (for different M) based on the two observation models in a 4×4 MIMO setup with 4-QAM inputs. Also shown are the results of the proposed energy-based method (dashed lines) which performs well in both the high and low SNR regimes. In a small region (SNR $\approx 2 - 4$ dB), it outperforms detectors based on both the Forney and Ungerboeck observation models. Note that, although the energy-based method relying on the instantaneous SNR would outperform a possible average SNR method, the average SNR can be used with only a small BER penalty. Additionally, Figure 4.5 confirms the existence of a crossover for MIMO setups.

By allowing middle models represented with grey circles, a similar plot to Figure 4.4 has been made (see Figure 4.6. It shows that, for specific channel realizations, the middle model with offset $p = 1$ is the optimal choice of model. However, its statistics are scattered over large areas of the plot, and no simple method to distinguish it when choosing the optimal model has been found.

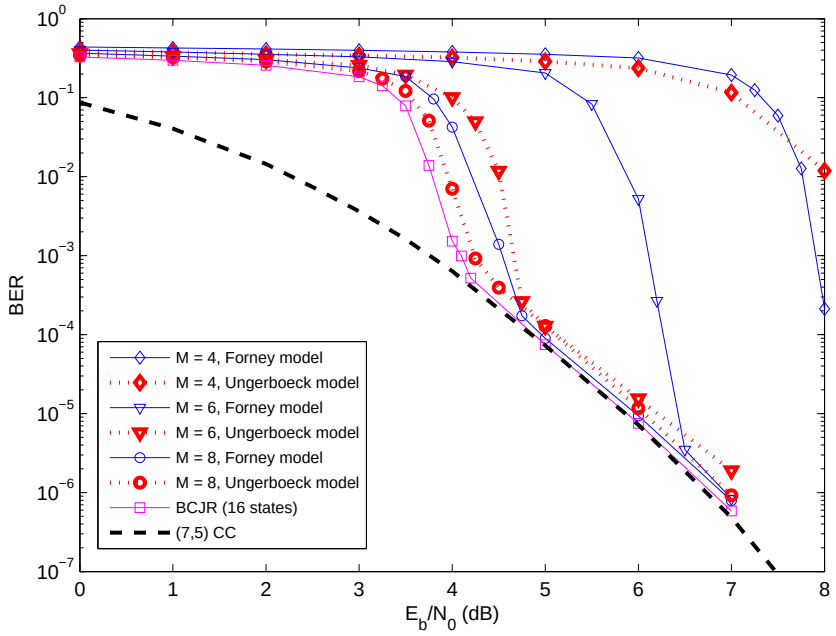

Figure 4.7: Coded BER performance of the Forney- and Ungerboeck-based M-BCJR detectors in a turbo scheme (10 iterations) with different values of M for the 5-tap Proakis-C ISI channel, a memory 2, rate 1/2 outer convolutional code and 2-PAM inputs.

4.5.2 Coded Transmission over ISI/MIMO Channels

Turbo Equalization for Coded ISI

Next consider coded transmission over ISI channels, as shown in Figure 4.1. The transmitter setup is as follows: A block of 5000 information bits, encoded by the outer (7,5) rate 1/2 feed-forward convolutional encoder, feeds a size 10000 random interleaver whose output is mapped to a symbol from the modulation alphabet Ω before being transmitted over an AWGN channel. Signals are terminated so that the transmission begins and ends in a pre-defined ISI state, for example $\sigma = [+1, +1, \ldots, +1]$.

This scheme can be viewed as serially concatenated coding, where the mapper together with the ISI channel act as an inner encoder. The iterative prin-

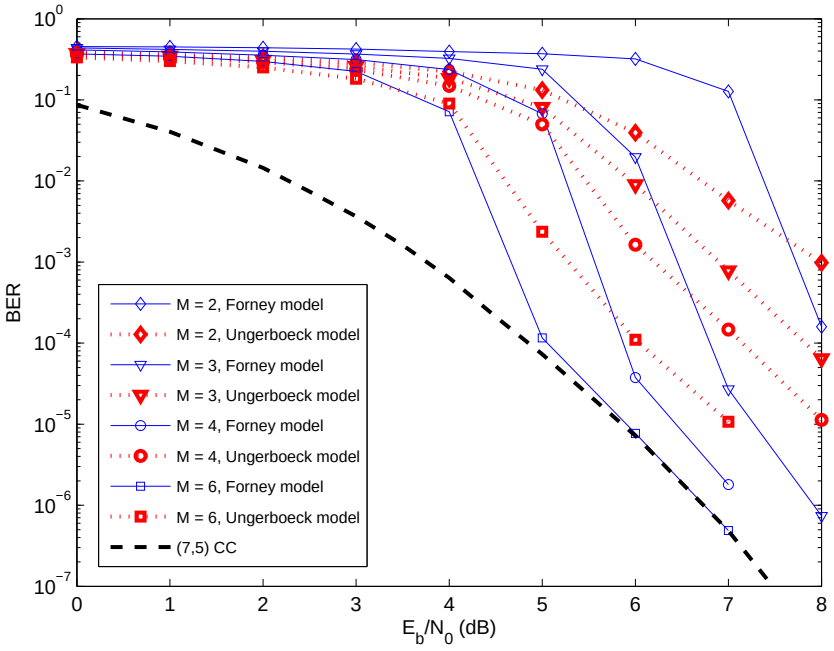

Figure 4.8: Coded BER performance of the Forney- and Ungerboeck-based M-BCJR detectors in a turbo scheme (10 iterations) with different values of M for the $\tau = 1/2$ FTN channel, a memory 2, rate 1/2 outer convolutional code and 2-PAM inputs.

ciple for equalization and decoding can thus be applied at the receiver, as first proposed in [10]. The outer decoder in the turbo scheme is a full-state BCJR (4 states) while the inner decoders are the M-BCJR detectors based on the two observation models. Soft information is passed around the loop 10 times before a final decision is made. Note that turbo equalization requires reasonably accurate absolute values while, in the uncoded case, only the LLR sign was needed.

Turbo equalization BER results for channels (4.28) and (4.29) and 2-PAM inputs are shown in Figure 4.7 and 4.8 respectively for several choices of M. The benchmark performance, since no precoding is employed (see [82] and [106]), is the ISI-free performance of the underlying outer code in AWGN, which is shown in the plots as a bold dashed 'CC' line. Additionally, for comparison in Figure 4.7, the performance of a 16-state BCJR detector is plotted. A

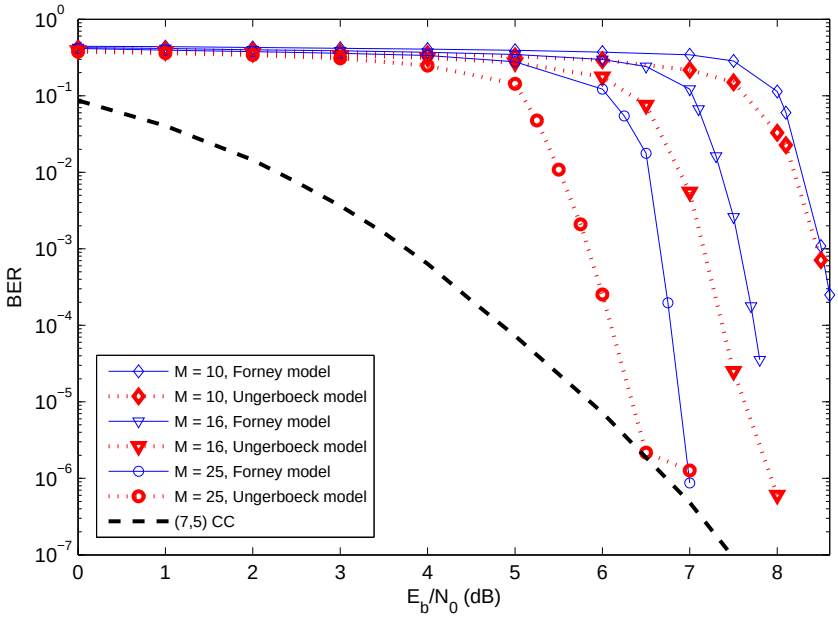

Figure 4.9: Coded BER performance of the Forney- and Ungerboeck-based M-BCJR detectors in a turbo scheme (10 iterations) with different values of M for the $\tau = 0.35$ FTN channel, a memory 2, rate 1/2 outer convolutional code and 2-PAM inputs.

closer inspection of the figures reveals that the crossover point between the two models is still present. Forney-based detectors perform slightly better at higher SNR, while the situation is reversed at low SNR. Note that, for $M = 6$ and 8 in Figure 4.7, the ultimate performance is first reached by the Ungerboeck-based detectors. The SNR range here corresponds to the left-hand side of the detector's operating range considered in Figure 4.2, where the performance difference between the two models is not as drastic as in the right-hand side. With $M = 8$ states, the performance of the Ungerboeck-based detector is very close to that of the 16-state BCJR.

Figure 4.9 shows coded BER performance of the Forney- and Ungerboeck-based M-BCJR detectors for the more severe FTN ISI channel (4.30) and 2-PAM inputs. The results verify that the Ungerboeck-based detectors indeed outperform the Forney-based detectors at practical SNR values. Since the crossover point now appears at a much lower error rate, the Ungerboeck model

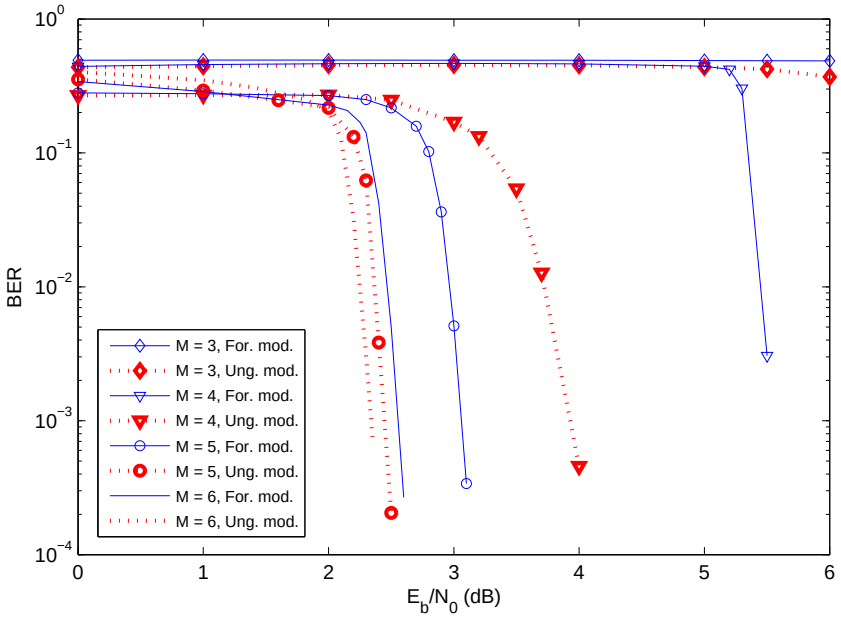

Figure 4.10: Receiver tests after 5 iterations of LDPC encoded transmissions over the EPR4 channel with 2-PAM inputs. The LDPC code is the irregular (32400,64800) standardized code in DVB-S.2.

is clearly the appropriate choice of model.

A different setup is considered in Figure 4.10. A block of 32400 information bits is, after encoding by the irregular rate 1/2 (32400,64800) LDPC code, standardized in DVB-S.2 [117], and mapping to 2-PAM symbols, transmitted over the EPR4 channel (4.31). The signal is corrupted by AWGN before being processed by an iterative scheme. The impressive results after 5 global iterations (20 internal in the LDPC decoder) show that the Ungerboeck-based M-BCJR detectors perform better than those based on the Forney model. For small values of M the differences in BER performance are significant. However, as M approaches the full complexity value $M = 8$, the differences are rather small. This is an expected outcome since optimal detectors based on the two observation models have equivalent detection properties.

Now consider again the same serially concatenated setup but for LDPC codes with other code rates: the rate 1/3 irregular (21600,64800) LDPC code, the rate 3/4 irregular (48600,64800) LDPC code and finally the rate 9/10 ir-

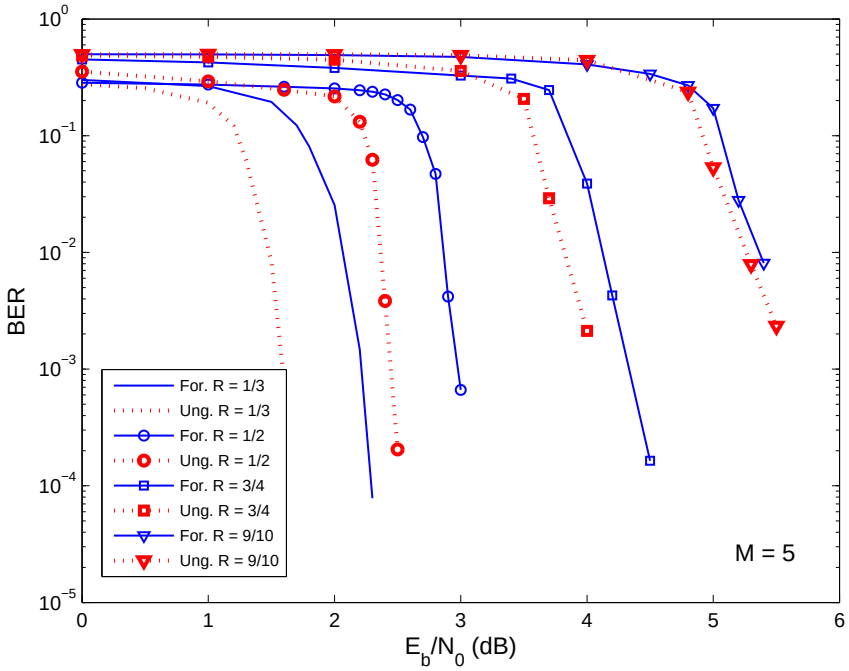

Figure 4.11: Receiver tests after 5 global iterations of LDPC encoded transmissions with different code rates over the EPR4 channel with 2-PAM inputs and for $M = 5$.

regular (58320,64800) LDPC code all chosen from the DVB-S.2 standard [117]. The inputs are again symbols from a 2-PAM alphabet and the same number of global and internal iterations is performed. Figure 4.11 shows the BER performance of Forney- and Ungerboeck-based M-BCJR detectors for $M = 5$ and different code rates. Also shown for comparison are the BER results for the rate 1/2 irregular (32400,64800) LDPC code. According to Figure 4.11, the best performing M-BCJRs for all tested code rates are those based on the Ungerboeck model. For a lower code rate, i.e. when more redundancy is added, the performance gains of using an Ungerboeck-based M-BCJR are larger. For the highest rate, that is rate 9/10, the BER curves are almost identical. This is an expected outcome since a higher code rate implies a higher required SNR in order to reach a target BER. Recall that Ungerboeck-based M-BCJRs perform better than those based on the Forney observation model for low SNRs.

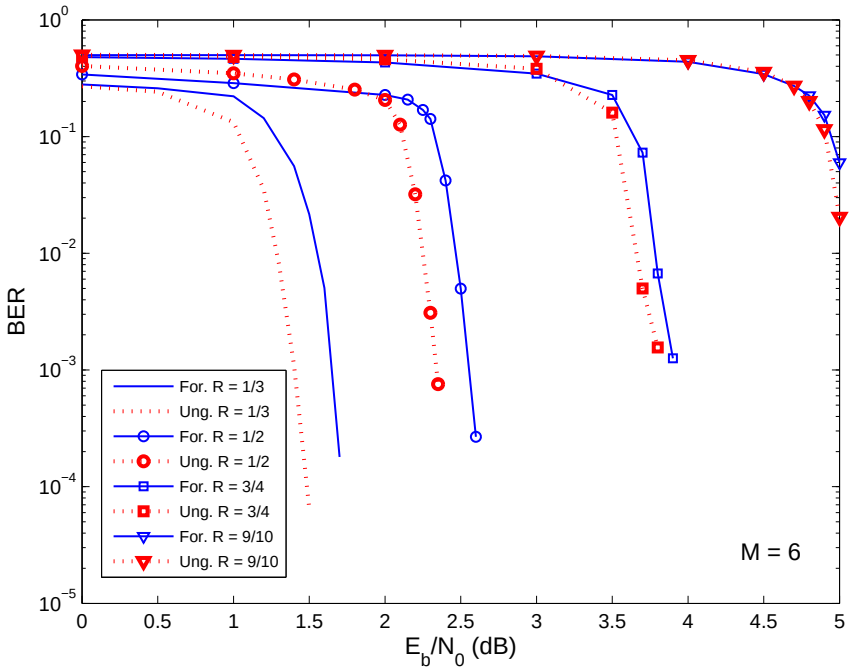

Figure 4.12: Receiver tests after 5 global iterations of LDPC encoded transmissions with different code rates over the EPR4 channel with 2-PAM inputs and for $M = 6$.

In Figure 4.12 simulation results for the $M = 6$ case are shown. As expected, the performance differences for all tested code rates are now smaller. Despite this, the M-BCJR detectors that result in the lowest error rates are those based on the Ungerboeck observation model.

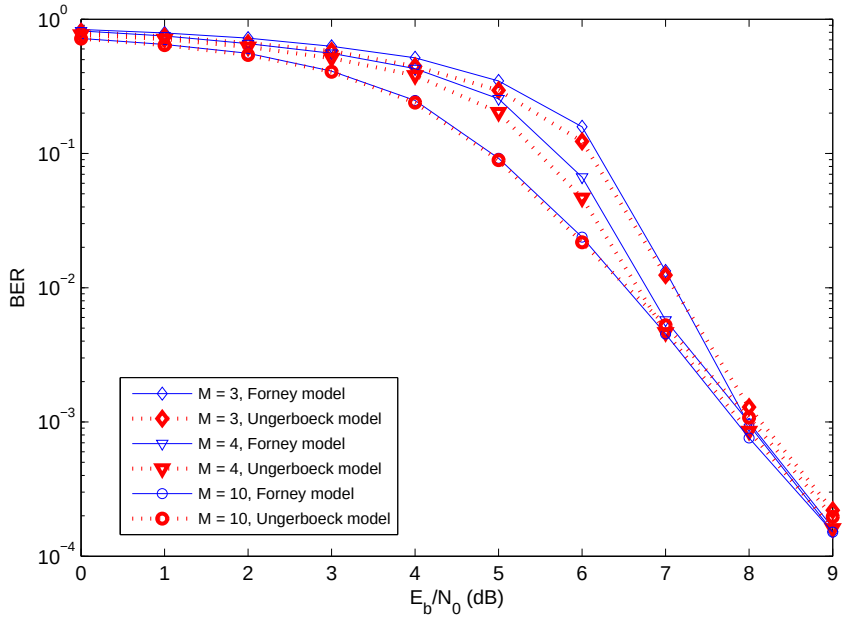

Figure 4.13: Coded BER performance of the Forney- and Ungerboeck-based M-BCJR detectors in a turbo scheme (10 iterations) with different values of M for a 4×4 MIMO setup, a memory 2, rate 1/2 outer convolutional code and 4-QAM inputs.

Turbo Equalization for Coded MIMO

Figures 4.13 and 4.14 show turbo equalization results after 10 iterations for 4×4 and 8×8 MIMO setups with 4-QAM inputs. The results are averaged over > 1000 channel matrix $\boldsymbol{H}$ realizations where the elements $h_{i,j}$ are independent and identically distributed complex Gaussians with unit variance, i.e., $h_{i,j} \sim \mathcal{CN}(0, 1)$. According to Section 4.2, the Forney observation model is obtained by performing a QL-decomposition of the channel matrix H in (4.7) which results in

$$\boldsymbol{y} = \boldsymbol{L}\boldsymbol{a} + \boldsymbol{w} \tag{4.36}$$

where $\boldsymbol{L}$ is a lower triangular matrix.

The Ungerboeck model in MIMO is obtained by a matrix multiplication

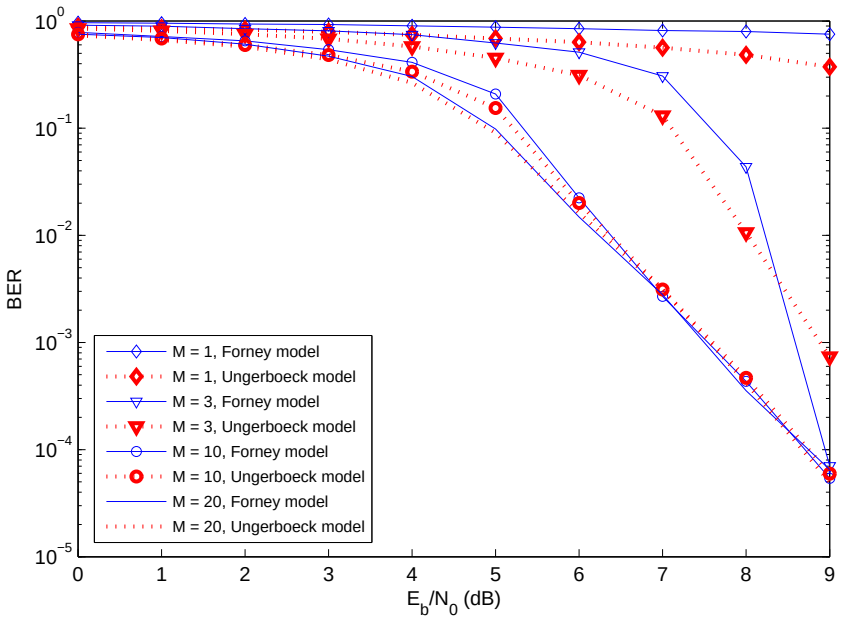

Figure 4.14: Coded BER performance of the Forney- and Ungerboeck-based M-BCJR detectors in a turbo scheme (10 iterations) with different values of M for a 8×8 MIMO setup, a memory 2, rate 1/2 outer convolutional code and 4-QAM inputs.

with $\boldsymbol{L}^\dagger$ (alternatively a matrix multiplication of (4.7) with $\boldsymbol{H}^\dagger$), that is

$$\boldsymbol{x} = \boldsymbol{L}^\dagger \boldsymbol{y} = \boldsymbol{G}\boldsymbol{a} + \boldsymbol{\eta} \tag{4.37}$$

where $\boldsymbol{G} = \boldsymbol{L}^\dagger \boldsymbol{L} = \boldsymbol{H}^\dagger \boldsymbol{H}$. Encoding is performed with the outer (7,5) convolutional code. Again, the figures confirm the predictions from Section 4.4 even though the performance differences between the two models here are rather small. Note that they grow (more obvious in Figure 4.14) with the decreasing size of M. When the value of M approaches the full complexity value, the performance of M-BCJR detectors based on the two models is close to optimal where they are expected to perform equivalently.

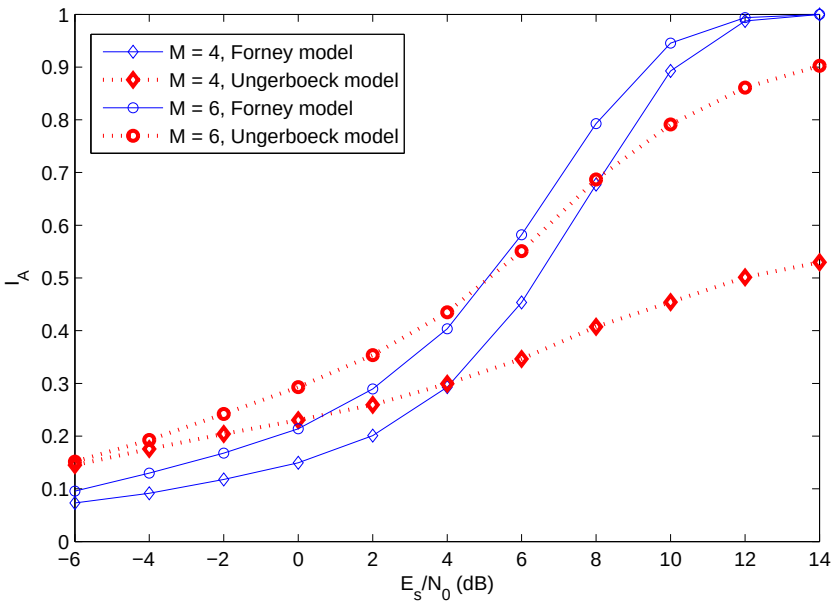

Figure 4.15: Mutual information between the output of the M-BCJR algorithm and the transmitted 2-PAM symbol sequence for the 5-tap Proakis-C ISI channel. Solid curves correspond to the Forney model while the dotted ones correspond to the Ungerboeck model.

4.5.3 Performance Evaluation via Mutual Information

In this section mutual information is used as an detector performance measure in an uncoded system. The *BCJR-once* bound is considered, as defined by Kavčić in [118]. It is the ultimate limit for separate non-iterative equalization/decoding of the channel and the outer code. Analytical computation of the BCJR-once bound

$$I_A = I(\boldsymbol{a}; L(\boldsymbol{a})) \tag{4.38}$$

between the transmitted sequence $\boldsymbol{a}$ from (4.1) and the sequence of L-values $L(\boldsymbol{a})$ generated by an detector is prohibitive in practice. However, with the independence assumption of a_k, one can consider the marginal PDF of the detector output, $\tilde{f}(l|\alpha) \triangleq \tilde{f}(L(a_k) = l|a_k = \alpha)$, and use it to calculate $I(\boldsymbol{a}; L(\boldsymbol{a}))$.

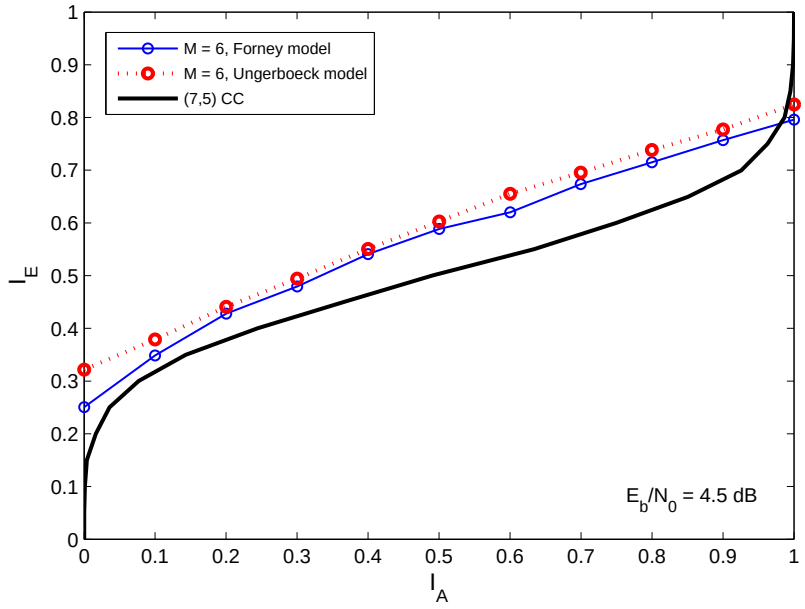

Figure 4.16: An EXIT chart at $E_b/N_0 = 4.5$ dB, showing extrinsic I_E vs. a priori I_A information for M-BCJR detectors based on the Forney and the Ungerboeck observation models with $M = 6$ and for 2-PAM inputs. The channel is the 5-tap Proakis-C ISI channel and the information sequence is encoded using the (7,5) outer convolutional code.

Calculation of mutual information was introduced in Section 2.5. Since the marginal PDF stemming from an ISI channel with binary equiprobable inputs satisfies $\tilde{f}(l|1) = \tilde{f}(-l|-1)$, I_A can be obtained by evaluating the integral

$$I_A = 1 - \int_{-\infty}^{\infty} \tilde{f}(l|1) \log_2(1 + e^{-l}) \mathrm{d}l. \tag{4.39}$$

By using an empirical estimate of the marginal density $\tilde{f}(l|1)$, (4.39) is evaluated numerically. The observation sequences $\boldsymbol{y}$ and $\boldsymbol{x}$ are formed from 10^7 information bits. M-BCJR detectors, based on the two observation models and with no a priori information, are applied to these sequences and a histogram of all $L(a_k)$ where $a_k = 1$ is used to estimate $\tilde{f}(l|1)$.

Figure 4.15 shows I_A for the setup considered in Figures 4.2 and 4.7, that is an uncoded system with 2-PAM inputs transmitted over the 5-tap Proakis-C

ISI channel in (4.28) followed by AWGN. Clearly, in the high SNR region, it can be observed that the mutual information obtained with the Ungerboeck-based detector is below that obtained with the Forney-based detector. The latter shows good performance (close to a full BCJR) even for $M = 4$ while the Ungerboeck-based detector performs poorly. In the low SNR region, on the other hand, the Ungerboeck model yields higher I_A than the Forney model. The mutual information crossover between the two observation models corresponds rather well to the BER crossover in Figure 4.2.

For the coded setup in Figure 4.7 it is not sufficient to only consider the mutual information I_A. Instead, according to Section 2.8, EXIT charts can be used to monitor the iterative convergence behavior of the detectors based on the two observation models. An EXIT chart for the coded setup at $E_b/N_0 = 4.5$ dB and for $M = 6$ is shown in Figure 4.16. The Ungerboeck-based M-BCJR is clearly the best performing detector. This is in fact confirmed by the turbo equalization results in Figure 4.7. For low a priori mutual information I_A the tunnel is wider for the Ungerboeck M-BCJR, allowing it to converge earlier (at a lower SNR) to the performance of the outer (7,5) convolutional code. Non-ideal interleavers and short blocklengths make it impossible for the Forney-based M-BCJR to pass where the tunnel is narrowest. A comparison with Figure 4.15 is possible for $E_s/N_0 \approx 1.5$ dB (corresponding to $E_b/N_0 = 4.5$ dB). The mutual information results are in fact the values in the first turbo iteration without any a priori information, and according to Figures 4.15 and 4.16 the mutual information values at $E_s/N_0 \approx 1.5$ dB correspond rather well to the left-most values in the EXIT chart.

4.6 The M*-BCJR Algorithm

The M*-BCJR algorithm [94], computes the L-values (4.23) in the same manner as the BCJR algorithm; however, similarly as in the M-BCJR [13], at each trellis stage in the forward recursion only M states with the highest forward metric are retained. Unlike in the M-BJCR, the remaining states are not deleted, but rather merged with the surviving states. Merging of two states implies that their forward metrics are summed up and the branches of the inferior state are redirected into the surviving state. An illustrative example of the merging process is shown in Figure 4.17. Such a modified trellis is subsequently used in the backward recursion. Although merging the states slightly increases the complexity, it preserves the balance of the branches that carry opposite symbols at each trellis depth, and thus avoids problems when computing the L-values in (4.23).

Since a state is an L-tuple of the most recent L symbols, then two states

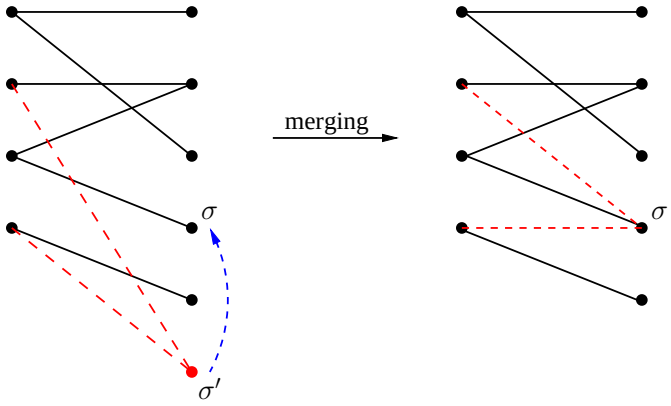

Figure 4.17: An example of a trellis section before and after merging an excess state σ' (shown in red color) into the surviving state σ.

that differ in $t \leq L$ ending positions (oldest symbols) merge in the trellis after t steps. If t is small, the metric difference of the paths leading to the common state is supposed not to be large [94]. If $\mathcal{S}_M$ and $\mathcal{S}_{\not M}$ again denote the set of the M best states and the set of the remaining states at a certain depth, respectively, then a rule proposed in [94] is that a state $\sigma' \in \mathcal{S}_{\not M}$ is merged with such a state $\sigma \in \mathcal{S}_M$ that differs in the least number t of the ending positions. The next subsection discusses the realization of this merging rule in more detail and also proposes alternative merging strategies. Hereinafter, binary representation of the states is assumed, i.e., $L \log_2(|\Omega|)$ bits uniquely define a state. The all-zero state σ_0 for a memory $L = 3$ ISI channel with 4-PAM inputs is represented with $3 \log_2(4) = 6$ bits, that is $\sigma_0 = [0, 0, 0, 0, 0, 0]$.

4.6.1 State Merging Strategies

The following state merging strategies are considered:

1) If $\oplus$ denotes the bitwise x-or operator, then the zero bits in $\sigma \oplus \sigma'$ indicate the positions where the states σ and σ' coincide. The state merging can be efficiently realized in the following way: for each state $\sigma' \in \mathcal{S}_{\not M}$ compute the values $\sigma \oplus \sigma'$ for all $\sigma \in \mathcal{S}_M$; find the state σ which yields the smallest value of $\sigma \oplus \sigma'$ (interpreted as a decimal number), and merge σ' with σ. This merging rule is denoted $\mathcal{R}_1$. It ensures that $\sigma' \in \mathcal{S}_{\not M}$ will be merged with the

state $\sigma \in \mathcal{S}_M$ that coincides with σ' in the largest number of leading positions. In case of a tie, a state with the smaller value of $\sigma \oplus \sigma'$ is preferred.

2) A modified approach, denoted $\mathcal{R}_2$, resolves the above mentioned cases of a tie, in a different way: if there is more than one state $\sigma \in \mathcal{S}_M$ that coincides with $\sigma' \in \mathcal{S}_{\not M}$ in the largest number of leading positions, then merge σ' with the one that has the *smallest forward metric*. Good results obtained with this strategy indicate that the metric values should be taken into account when merging the remaining states.

3) Motivated by the previous observation, strategy $\mathcal{R}_3$ is proposed, which simply merges *all* the states from $\mathcal{S}_{\not M}$ with the state $\sigma \in \mathcal{S}_M$ that has the smallest forward metric. Note that this strategy is the simplest to implement, since it does not require any additional computations or sorting procedures during the merging process, unlike the previous two.

Also tested is to replace the "smallest-metric" choice in $\mathcal{R}_2$ and $\mathcal{R}_3$ by the "largest metric"; however, this variant of the algorithm fails completely. This suggests that, among the chosen M states at each stage, the "good" states with large metric should be left intact, while the "weak" states should be used to "collect" the discarded states from $\mathcal{S}_{\not M}$.

Approaches $\mathcal{R}_1$, $\mathcal{R}_2$, and $\mathcal{R}_3$ have been tested with various ISI patterns. The rules $\mathcal{R}_2$ and $\mathcal{R}_3$ outperform $\mathcal{R}_1$, allowing largest complexity reduction, that is, the smallest M, to reach the specified bit error rate. For a given value of M, $\mathcal{R}_2$ yields the lowest BER, and it will therefore be used hereinafter.

4.7 M*-BCJR Receiver Tests

4.7.1 M*-BCJR Results for Uncoded ISI

Consider first uncoded 2-PAM transmission over an ISI channel of memory L. The complexity of the BCJR detector is of the order 2^L. In the tests, two standard ISI channel models have been used, both causing severe ISI, and both of memory $L = 4$: the minimum-phase equivalent of the Proakis-C channel (4.28) and the channel (3.14) used in [10] and [94]. However, all the results presented here are given for the Proakis-C channel only, with the note that all the observations hold for channel (3.14) as well.

The BER performance of the M*-BCJR detectors, based on the two established observation models, with $M = 4$ states, employing the merging rule $\mathcal{R}_2$, are shown in Figure 4.18. As a reference, the performance of the BCJR

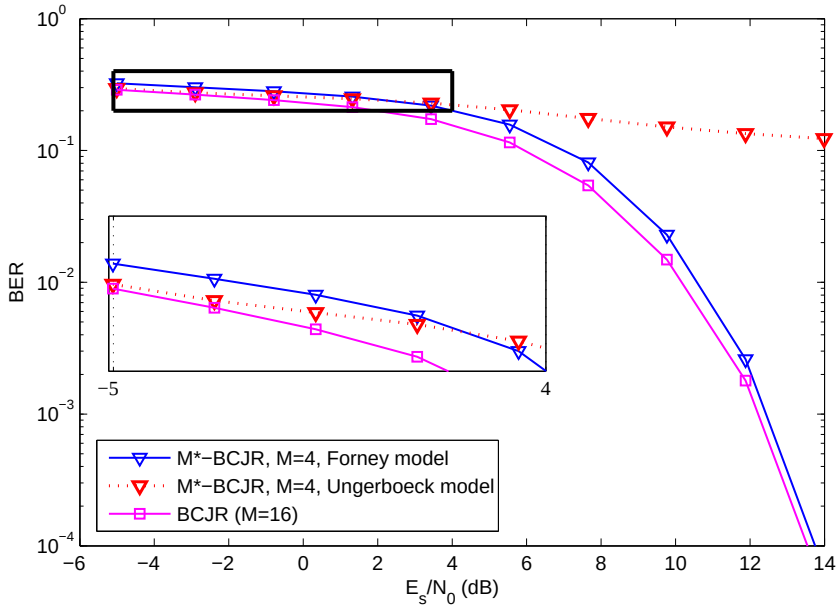

Figure 4.18: Uncoded BER performance of the Forney- and Ungerboeck-based M*-BCJR detectors with $M = 4$, for Proakis-C 5-tap ISI channel.

detector (with $M = 16$ states) is also shown. The Forney-based M*-BCJR follows the BCJR performance with a small loss, while the Ungerboeck-based detector completely fails for the medium and high SNR levels, suffering from a high error floor. Again, the error floor is eliminated only when M approaches the full-complexity value ($M = 16$). In the low SNR region, however, left from the crossover point at $E_s/N_0 \approx 2.5$ dB, the behavior is reversed and the Ungerboeck model yields lower BER than the Forney model.

4.7.2 Turbo Equalization for Coded ISI

This section assumes the same transmitter setup as in Section 4.5.2. The M*-BCJR algorithm is used as inner decoder in the turbo scheme, with the channel parameters from the previous subsection. A systematic memory 1 convolutional code with the generator matrix $(1 + D, 1) = (3, 1)_8$ was used as the outer code, and the block length was 1000 information bits. The BER performance of the scheme is shown in Figure 4.19, for two choices of M, with the benchmark given by the turbo BCJR detector and the underlying outer code. The M*-

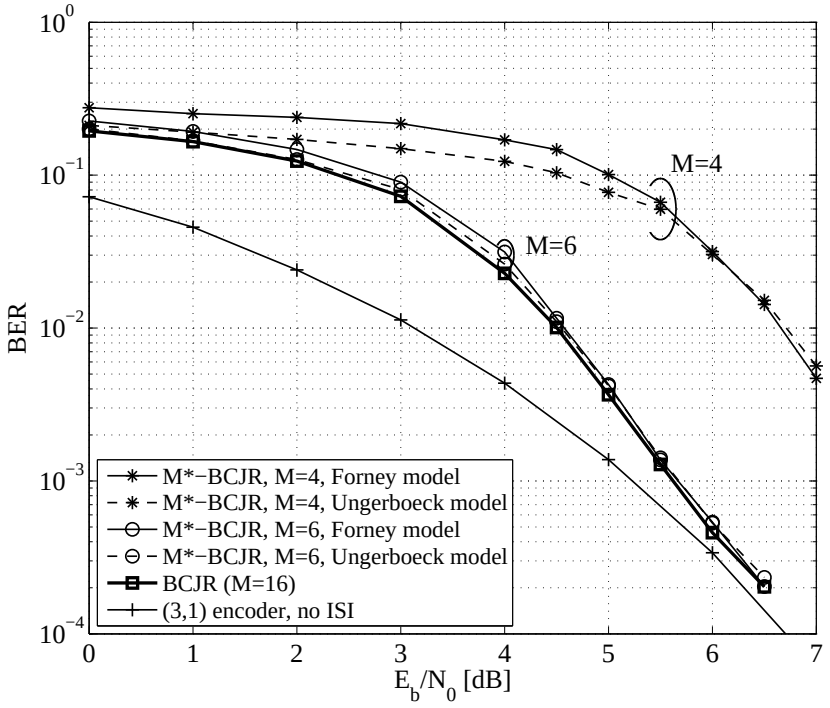

Figure 4.19: Performance of the Forney- and Ungerboeck-based M*-BCJR detectors in a turbo scheme, after 8 iterations, for Proakis-C 5-tap ISI channel, and a systematic memory 1, rate 1/2 outer convolutional code with generator matrix $(1 + D, 1)$.

BCJR detector fails to converge to the BCJR performance if it preserves only $M = 4$ states at each depth in the trellis. With $M = 6$ states, however, the performance is very close to that of the BCJR detector. It can be observed that the M*-BCJR detector in the iterative setup performs (almost) equally well with both Forney and Ungerboeck models. However, Figure 4.19 shows that there is a crossover point between the two models: Forney-based detection performs better for higher SNR while Ungerboeck-based detection is the best choice for low SNR. These observations confirm the previous results based on the M-BCJR algorithm. The weaker outer code in this section was chosen deliberately in order to obtain crossover points in Figure 4.19 at moderate BER.

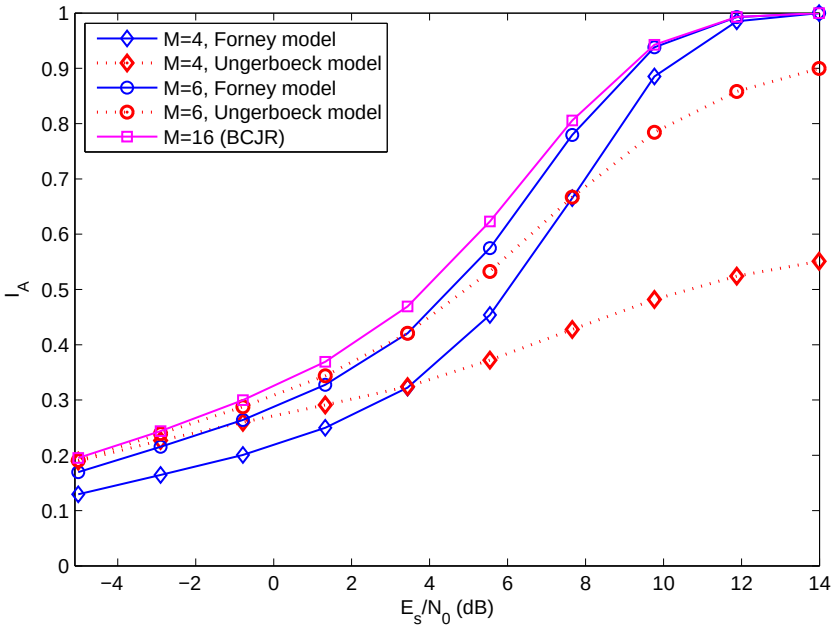

Figure 4.20: Mutual information between the output of the M*-BCJR algorithm and the transmitted symbol sequence.

4.7.3 M*-BCJR Mutual Information Results

Figure 4.20 illustrates the mutual information I_A (4.39) for the M*-BCJR detector and the 5-tap Proakis-C channel. Dotted lines with diamonds and circles represent the Ungerboeck model for two different values of M while the corresponding solid lines represent the Forney model. Additionally, the mutual information of a full-complexity BCJR (16 states) is shown for comparison (solid line with squares). In the high SNR region the mutual information obtained with the Ungerboeck-based M*-BCJR is lower than that obtained with the Forney-based detector. However, at low SNR the Ungerboeck model prevails and consequently there is a crossover between the models. The BER in Figure 4.18 equals BER $= \int_{-\infty}^{0} \tilde{f}(l|1)\mathrm{d}l$, while the mutual information is given by (4.39), thus there cannot be an exact agreement between the mutual information and the BER crossover points. For $M = 4$, the crossover in the iterative receiver test occurs at $E_b/N_0 = 6.1$ dB, i.e., $E_s/N_0 = 3.1$ dB, while the mutual information chart suggests $E_s/N_0 = 3.5$ dB.

As already pointed out, in the analysis of the iterative equalization process, it is not sufficient to consider only I_A. The mutual information I_A is only involved in the first iteration; in subsequent iterations, influence of a priori information must be considered – this is the well known EXIT chart technique [85]. However, I_A predicts the BER performance of the turbo equalizer quite well, which will be explained in the following. If $T_{\mathrm{ISI}}(x)$ denotes the EXIT curve for the ISI channel and the detector under investigation, then there is the following analogy between I_A and $T_{\mathrm{ISI}}(x)$: if a certain detector and ISI model is better than another one, then $T_{\mathrm{ISI}}(x) > T'_{\mathrm{ISI}}(x),\ 0 \leq x \leq 1$, instead of $I_A > I'_A$ for the uncoded case. The starting point of $T_{\mathrm{ISI}}(x)$ is $T_{\mathrm{ISI}}(0) = I_A$, while the ending point is $T_{\mathrm{ISI}}(1) = T_{\mathrm{MLC}}(0)$, where 'MLC' denotes 'memoryless channel' (in fact, $T_{\mathrm{MLC}}(0) = T_{\mathrm{MLC}}(x), 0 < x \leq 1$). Thus, the endpoints of all EXIT curves are the same, and their starting points are determined by I_A. Therefore, when $I_A > I'_A$, it is plausible that $T_{\mathrm{ISI}}(x) > T'_{\mathrm{ISI}}(x),\ 0 \leq x \leq 1$ as well. This explains the good match between I_A and the BER performance of the turbo equalization.

Although I_A is much larger at higher SNR for the M- and M*-BCJR detectors based on the Forney model than for their Ungerboeck-based counterparts, it is not possible to conclude that in general the Forney-based detection is superior to the Ungerboeck-based one. The difference in I_A may be a consequence of the algorithm itself, which approximates L-values $L(\boldsymbol{a})$ with reduced complexity. There are two approximations involved: (i) the L-values are computed with only M nonzero values $\alpha_k(\sigma)$ at every depth k, and (ii) these M nonzero $\alpha_k(\sigma)$ are not computed with full complexity, but they are themselves only approximations.

4.7.4 Genie-Aided Detectors

In this section genie-aided detectors are considered. The objective is to eliminate one of the involved approximations in a reduced-complexity detector and to optimize the other separately. Already at this point, we would like to inform the reader that even if one of the genie-aided Ungerboeck-based detectors shows excellent performance when approximation (ii) is eliminated, the corresponding branching strategy, when incorporated into a real detector, results in poor overall performance. In fact, its mutual information I_A is lower than that of the original M*-BCJR.

Detector $\mathcal{G}_1$

In order to eliminate approximations (ii) from the discussion above, a genie-aided detector, denoted by $\mathcal{G}_1$, is considered next. A genie provides the exact

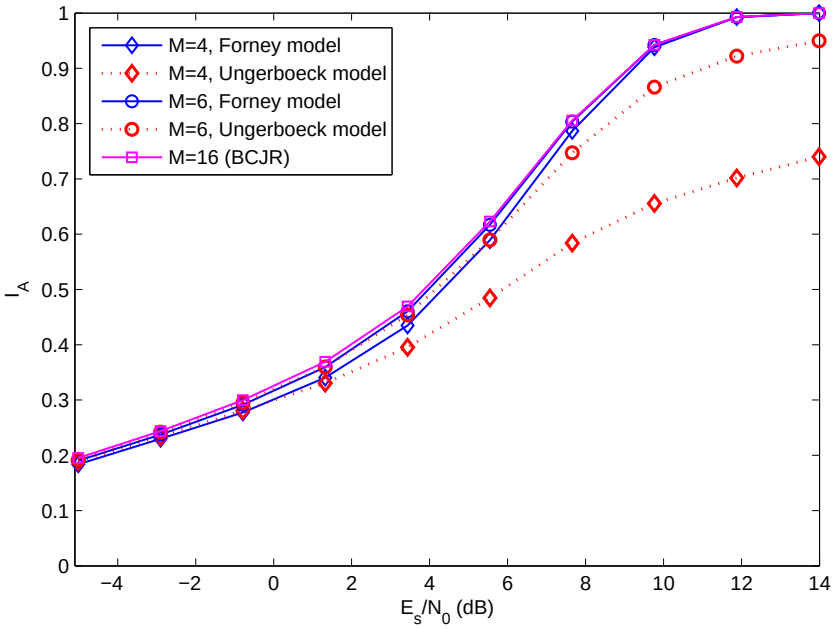

Figure 4.21: Outcome of the genie-aided detector $\mathcal{G}_1$. Dotted curves correspond to the Ungerboeck model and the solid ones to the Forney model.

values of $\alpha_k(\sigma)$ and $\beta_k(\sigma)$ for all depths k. The L-values $L(a_k)$ in (4.23) are computed using the M largest values $\alpha_k(\sigma)$ only. This method serves as a benchmark for detectors that construct a reduced trellis in the forward recursion based on the largest $\alpha_k(\sigma)$. The mathematical formulation of $\mathcal{G}_1$ is as follows: Define δ_k as the Mth largest metric $\alpha_k(\sigma)$ at depth k, and

$$\hat{\alpha}_k(\sigma) \triangleq \begin{cases} \alpha_k(\sigma), & \alpha_k(\sigma) \geq \delta_k \\ 0, & \alpha_k(\sigma) < \delta_k. \end{cases} \tag{4.40}$$

The values $L(a_k)$ are obtained as in (2.92) but with

$$p(\sigma_k = \sigma, \sigma_{k+1} = \sigma', \boldsymbol{x}) = \hat{\alpha}_k(\sigma)\gamma_k(\sigma, \sigma')\beta_{k+1}(\sigma'). \tag{4.41}$$

Figure 4.21 shows the mutual information obtained with the genie-aided detector $\mathcal{G}_1$, for the same parameters as in Figure 4.20. It is readily seen that, even

with $\mathcal{G}_1$, the Ungerboeck model still performs poorly in the high SNR region. Moreover, for a given M, the Forney curve lies strictly above the Ungerboeck curve, which implies the conclusion that detectors which construct reduced trellis in the forward recursion (based on the largest $\alpha_k(\sigma)$), should operate on the Forney model.

Detector $\mathcal{G}_2$

Since the detector $\mathcal{G}_1$ does not perform well with the Ungerboeck model, a more general class of detectors is considered next. These detectors build two independent reduced trellises: one in the forward recursion, based on the largest α-metric, and one in the backward recursion, based on the largest β-metric. The L-values in (4.23) are obtained from the union of the two trellises (explained formally below). Such detectors have been investigated in [110] and [112]. A genie-aided detector $\mathcal{G}_2$, which is a benchmark for this class, is considered. The genie provides all exact $\alpha_k(\sigma)$ and $\beta_k(\sigma)$ values (computed with full complexity). For each trellis stage k, define $\hat{\alpha}_k(\sigma)$ according to (4.40) and $\hat{\beta}_k(\sigma)$ similarly. The branches that are involved in the computation of $L(a_k)$ are those that have at least one endpoint with nonzero metric $\hat{\alpha}_k(\sigma)$ or $\hat{\beta}_{k+1}(\sigma')$. If a certain branch has both endpoints with nonzero $\hat{\alpha}_k(\sigma)$ and $\hat{\beta}_{k+1}(\sigma')$, its contribution to $L(a_k)$ becomes $\hat{\alpha}_k(\sigma)\gamma_k(\sigma,\sigma')\hat{\beta}_{k+1}(\sigma')$. If, however, a branch has only one nonzero endpoint, the genie provides the necessary ("missing") $\alpha_k(\sigma)$ or $\beta_{k+1}(\sigma')$ value, and the contribution becomes $\hat{\alpha}_k(\sigma)\gamma_k(\sigma,\sigma')\beta_{k+1}(\sigma')$ or $\alpha_k(\sigma)\gamma_k(\sigma,\sigma')\hat{\beta}_{k+1}(\sigma')$. For practical detectors of this type, where genie knowledge is not available, the contribution of such branches is not clearly defined. In [110] it was proposed how to handle these cases in practice and compensate for the "missing" endpoint metrics.

The tests show that the outcome of the detector $\mathcal{G}_2$ is, in terms of the mutual information I_A, virtually identical to that of $\mathcal{G}_1$, cf. Figure 4.21. Thus, this approach does not seem to benefit from the Ungerboeck model either.

Detector $\mathcal{G}_3$

In order to understand and solve the weakness of detection strategies based on the Ungerboeck model (for more detailed treatment of this problem, see also [109]), consider again the function $\varphi(x_k, \boldsymbol{a})$, given by (4.15) in log-domain, which defines the BCJR branch metric. By assuming, as throughout this chapter, a 2-PAM symbol alphabet and a unit energy ISI response, the Ungerboeck branch metric can be written as

$$\varphi(x_k, \boldsymbol{a}) = \exp\left(\frac{2}{N_0} a_k^* \left(x_k - \frac{a_k}{2} - \sum_{l=1}^{L} g_l a_{k-l}\right)\right). \tag{4.42}$$

For an arbitrary state at depth k, the φ values associated with the outgoing branches for $a_k = 1$ and $a_k = -1$ are $e^{(2\mu-1)/N_0}$ and $e^{(-2\mu-1)/N_0}$, respectively, where $\mu = x_k - \sum_{l=1}^{L} g_l a_{k-l}$. The received signal at time instant k can, according to (4.6), be written as

$$x_k = \tilde{a}_k + \Sigma_\mathrm{p} + \Sigma_\mathrm{f} + \eta_k, \tag{4.43}$$

where $\tilde{\boldsymbol{a}}$ is the actual transmitted symbol sequence and Σ_p and Σ_f are the contributions to x_k from past and future symbols respectively. Note that $\Sigma_\mathrm{f} \neq I_\mathrm{f}$ in (4.27). The correct path in the trellis (corresponding to the transmitted sequence $\tilde{\boldsymbol{a}}$) passes through the state $\sigma = [\tilde{a}_{k-L} \ldots \tilde{a}_{k-1}]$ at time point k. This implies that the sum $\sum_{l=1}^{L} g_l a_{k-l}$ in (4.42) is equal to the term Σ_p in (4.43). Thus, $\varphi(x_k, \boldsymbol{a})$ equals

$$\varphi(x_k, \boldsymbol{a}) = \exp\left(\frac{2}{N_0}\left[a_k^*\big(\tilde{a}_k + \Sigma_\mathrm{f} + \eta_k\big) - \frac{1}{2}\right]\right). \tag{4.44}$$

We know that in the high SNR region, where the Ungerboeck model shows poor performance, the approximation $\eta_k \approx 0$ holds. If the term Σ_f was not present, the two outgoing branches, corresponding to $a_k = \tilde{a}_k$ and $a_k = -\tilde{a}_k$ would have the metric $\varphi \propto e^{1/N_0}$ and $\varphi \propto e^{-1/N_0}$, respectively, and thus the correct path gets a much larger metric value. But from (4.24) we know that, when $\sum_{l=1}^{L} |g_l| > 1$, corresponding to the closed eye diagram, it is possible that $|\Sigma_\mathrm{f}| > 1$, which implies that $\tilde{a}_k + \Sigma_\mathrm{f}$ can have the sign opposite from $\tilde{a}_k$. This leads to the incorrect path (with $a_k = -\tilde{a}_k$) having a larger metric at depth $k+1$ than the correct path (with $a_k = \tilde{a}_k$). Note that as before this happens without any noise and that there is a non-zero probability for this to occur. The correct state at time $k+1$, corresponding to $a_k = \tilde{a}_k$, would then have a small metric $\alpha_{k+1}(\sigma)$ and would likely be eliminated from the list. Thus, the correct path in the trellis would be lost, and cannot be recovered. Therefore, it is proposed to always include both states (corresponding to $a_k = \pm 1$) into the set of M surviving states at time $k+1$.

The method described above is formally expressed next. Partition the state space $\mathcal{S}$ as $\mathcal{S} = \{\mathcal{P}_1, ..., \mathcal{P}_{2^{L-1}}\}$, where each set $\mathcal{P}_l$ holds a pair of states (σ, σ') such that if $(\tilde{\sigma}, \sigma) \in \mathcal{S}^+$ for some state $\tilde{\sigma} \in \mathcal{S}$, then $(\tilde{\sigma}, \sigma') \in \mathcal{S}^-$. Define

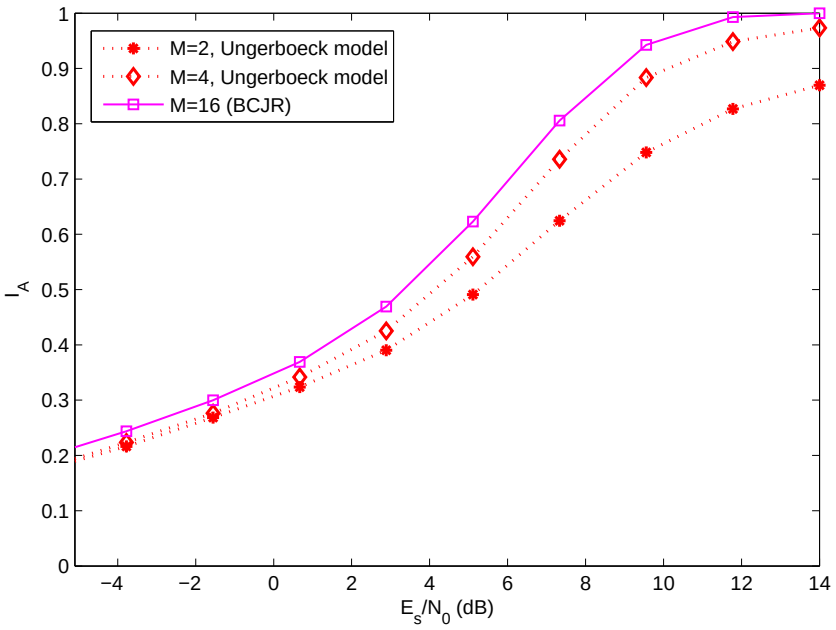

Figure 4.22: Outcome of the genie-aided Ungerboeck-based detector $\mathcal{G}_3$.

$\alpha^l_{\max,k} \triangleq \max\{\alpha_k(\sigma), \alpha_k(\sigma')\}$, $(\sigma, \sigma') \in \mathcal{P}_l$, and define δ_k as the $(M/2)$th largest metric $\alpha^l_{\max,k}$ at each depth k; M is assumed to be an even integer. Then the survivor states are all states that have nonzero $\hat{\alpha}_k(\sigma)$, where

$$\hat{\alpha}_k(\sigma) \triangleq \begin{cases} \alpha_k(\sigma), & \alpha^l_{\max,k} \geq \delta_k,\ \sigma \in \mathcal{P}_l \\ 0, & \text{otherwise.} \end{cases} \tag{4.45}$$

The outcome of this approach, denoted by $\mathcal{G}_3$, based on the Ungerboeck model is shown in Figure 4.22. The performance of $\mathcal{G}_3$ is much better than that of $\mathcal{G}_1$ and $\mathcal{G}_2$. The method works very well even with only two survivor states per depth.

The branching strategy of $\mathcal{G}_3$ can be incorporated into the M*-BCJR algorithm in order to obtain a new practical detector. However, the performance of this detector is not as good as one would expect from Figure 4.22. In fact, its mutual information I_A is below that of the original M*-BCJR algorithm shown in Figure 4.20. This suggests that eliminating one approximation in the

reduced-complexity detector and optimizing the other separately is not a good approach. The approximations are not independent and should be treated that way. How to exploit the gain promised by $\mathcal{G}_3$ is a topic for future research.

4.8 Summary and Conclusions

This chapter considered the performance of reduced-complexity detectors, based on the Ungerboeck and the Forney observation models, for coded and uncoded ISI and MIMO channels. Unlike the Forney model whose channel observations only depend on the current and past data symbols, observations in the Ungerboeck model contain contributions from both past and future symbols. In a MIMO system, the future symbols correspond to the symbols on the transmit antennas that have not yet been reached by the detection process. Even though the final output of a full VA or BCJR is identical for both models the metric calculations are in general different. It is demonstrated that this fundamental difference has a crucial effect on the performance of reduced-complexity detectors of the M-algorithm type. It is also concluded that the Ungerboeck-based detectors perform in general better in the low SNR region but in some cases even for practical SNR values (see Fig. 4.9). However, as the SNR increases, detection based on the Forney model performs in general better.

Additionally, middle models working in between the two extremes were presented and evaluated. A simple scheme for finding the optimal choice of observation model (in BER sense) is proposed. The chapter also reflects on the asymptotic behavior of the two observation models; conclusions drawn there are confirmed by practical receiver test and mutual information results.

In order to investigate the ultimate performance of standard detectors and to better understand the poor performance of Ungerboeck-based detectors in the high SNR region, genie-aided reduced-trellis detectors were considered. One of the genie-aided detectors, constructed for the Ungerboeck model, succeeds in reaching the performance of the Forney-based detector. So far we have found no method which can practically exploit the gains promised by this genie-aided detector.

Finally, it should be highlighted that the choice of observation model *does not impact* the detection complexity as the underlying algorithm is unaltered for a given M. Hence, the gains reported in this chapter come with no additional cost.

In the next chapter a different complexity reducing approach is considered. Channel shortening detectors are optimized from an information theoretical perspective.

Chapter 5

Optimal Channel Shortening for MIMO and ISI Channels

This chapter considers the construction of optimal channel shortening, also known as combined linear Viterbi detection, algorithms for ISI and MIMO channels. In the case of MIMO channels, the concept of channel shortening means a spatial memory reduction among the antennas so that the tree structure which represents MIMO signals is replaced by a trellis. The optimization is performed from an information theoretical perspective and the achievable information rates of the shortened models are derived and optimized. Closed form expressions for all components of the optimal detector of the class are derived. Furthermore, it is shown that previously published channel shortening algorithms can be seen as special cases of the derived model. Some parts of this chapter have appeared in [119].

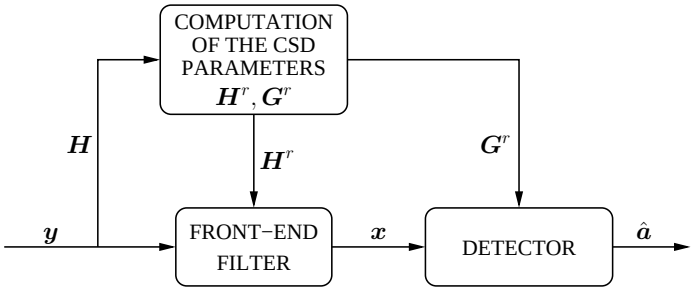

Figure 5.1: An simplified illustration of the detection process when employing a channel shortening detector (CSD).

5.1 Introduction

This chapter considers the construction and optimization of reduced complexity trellis detection methods for ISI and MIMO channels. As already pointed out in Chapter 3, within trellis detection there are two main directions:

- To process the original trellis, but with reduced complexity so that only a fraction of the trellis is explored, a *reduced-search* approach, or
- To construct a reduced trellis which is then processed with full complexity, a *reduced-trellis* approach.

Examples from the first class include the sphere detector[1] [120], the fixed-complexity sphere detector [121], the M-algorithm [6], the soft-output M-algorithm [14], and soft-output sequential detection [15]. This chapter optimizes a general framework, first proposed in [122], for detectors from the second class. The investigated detectors filter the received signal with a channel shortening filter, and then apply full-complexity trellis processing on the shortened model. In the case of MIMO transmission, the front-end filter is replaced with a matrix multiplication that aims at converting the MIMO tree structure into a much smaller trellis. Figure 5.1 illustrates this process using the same notations as throughout this chapter. The front-end filter in matrix form is denoted $\boldsymbol{H}^{\mathrm{r}}$ while the shortened model of the channel is given by $\boldsymbol{G}^{\mathrm{r}}$.

[1]The sphere detector is optimal in the case of hard output detection, but not when used for soft output detection [135]

The history of channel shortening dates back to the early 70s, sometimes under the name *combined linear Viterbi equalization*. Forney showed in 1972 that the Viterbi algorithm implements maximum-likelihood detection of finite-memory ISI-channels [4]. Shortly after Forney's discovery, researchers realized that in many practical scenarios, the duration of the channel response is far too long for practical implementation of the Viterbi algorithm. This generated massive research efforts in order to reduce the computational complexity of the Viterbi algorithm. One approach that appeared promising was channel shortening. Falconer and Magee in 1973 conducted the first investigation of channel shortening [123]. Since Falconer and Magee's work, research on channel shortening has been continuously published [124]–[132]. So far, all channel shortening detectors (CSDs) have been optimized from a minimum mean-square-error perspective, to be made more precise later in the chapter. The capacity is however the ultimate limit of a communication system, i.e., the highest possible transmission rate of a system employing optimal detection. A receiver that operates on the basis of a mismatched channel model can on the other hand not achieve capacity. Instead the so-called generalized mutual information is now the ultimate limit [133, 134]. Our proposed channel shortening approach maximizes the achievable information rate, corresponding to the generalized mutual information without the optimization over the input signal constellation and the distribution, and in that way operates closer to the ultimate limit. The mean-square-error (MSE) is a suboptimum cost function since it does not directly correspond to the highest transmission rate (in terms of generalized mutual information) that can be supported by a shortening detector. The Shannon limit of mismatched detectors, the generalized mutual information, was derived in [133, 134]. Since a channel shortening detector approximates the true channel model with a shorter model it falls under the framework of mismatched detection. Thus, in the early days of channel shortening, the tools in [133, 134] for optimizing the shortening detector were not available. Furthermore, another difference between the approach presented in this chapter and [123]–[132] is that our approach uses a more general framework for channel shortening. Hence, the detectors derived in this chapter are out of reach in [123]–[132].

The framework in this chapter is based on [122], but is extended in several important directions:

- This chapter considers general linear channels, while [122] only treated ISI channels.
- The framework from [122] is in this chapter optimized for Gaussian inputs, and closed form expressions for the filters and the resulting generalized mutual informations are obtained.

- In this chapter it is discovered that the optimal channel shortening filter is intimately connected to the conventional MMSE filter. The difference is that the optimal channel shortening filter is modified to incorporate the trellis processing. The derived filter differs from the filters used in [123]–[132].

- For practical coded modulation systems, the detector is slightly modified so that its error performance is improved. The reason why this is needed is that the detector is optimized with a mutual information cost function. Practical coded modulation systems operate at energies somewhat above that at capacity, and the mutual information optimality does not translate perfectly to BER-optimality of practical systems.

- This chapter provides the optimized branch labels of the reduced trellis in closed form.

5.1.1 System Model

Linearly-modulated transmissions over linear vector-channels affected by additive white Gaussian noise (AWGN) are considered. The received signal can, according to Chapter 2, be described by the input-output discrete-time model

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{a} + \boldsymbol{w} \tag{5.1}$$

where $\boldsymbol{y} = [y_1, \ldots, y_{N_r}]^{\mathrm{T}}$ denotes the received samples, $\boldsymbol{a} = [a_1, \ldots, a_{N_t}]^{\mathrm{T}}$ denotes the input symbols, and $\boldsymbol{w} = [w_1, \ldots, w_{N_r}]^{\mathrm{T}}$ are independent and identically distributed zero mean circularly symmetric complex white Gaussian random variables with variance N_0, i.e., $\boldsymbol{w} \sim \mathcal{CN}(\boldsymbol{0}, N_0\boldsymbol{I}_{N_r \times N_r})$. The $N_r \times N_t$ complex-valued matrix $\boldsymbol{H}$ describes the linear channel which is assumed to be perfectly known at the receiver. The input symbols $\{a_k\}$ which are to be transmitted over the channel belong to a symbol alphabet Ω.

As already stated in Chapter 4, in the case of $N_r \neq N_t$ where $N_t = N$ it is possible to convert the channel into an $N \times N$ channel as follows. If $N_r > N_t$, the channel model can be QR-decomposed into $\boldsymbol{y} = \boldsymbol{Q}\boldsymbol{R}\boldsymbol{a} + \boldsymbol{w}$. The matrix $\boldsymbol{R}$ can be written as

$$\boldsymbol{R} = \begin{bmatrix} \tilde{\boldsymbol{R}} \\ \boldsymbol{0}_{N_r - N_t, N_t} \end{bmatrix}$$

where $\mathbf{0}_{N_r-N_t,N_t}$ is the all-zero matrix of size $(N_r - N_t) \times N_t$ and $\tilde{\boldsymbol{R}}$ is an $N_t \times N_t$ upper triangular matrix. This implies that optimal detection of $\boldsymbol{a}$ can be performed by only considering the first N_t components of $\boldsymbol{y}$. If we denote these by $\tilde{\boldsymbol{y}}$, we can instead work with

$$\tilde{\boldsymbol{y}} = \tilde{\boldsymbol{R}}\boldsymbol{a} + \tilde{\boldsymbol{w}}.$$

In the case $N_r < N_t$, zeros are appended to the channel matrix. Hence the channel model

$$\bar{\boldsymbol{y}} = \begin{bmatrix} \boldsymbol{H} \\ \mathbf{0}_{N_t-N_r,N_t} \end{bmatrix} \boldsymbol{a} + \bar{\boldsymbol{n}}$$

is considered where $\bar{\boldsymbol{y}}$ is an $N_t \times 1$ column vector of received samples and $\bar{\boldsymbol{n}}$ is an $N_t \times 1$ noise vector. Later in the chapter, no restrictions on the structure of the channel matrix shall be made, so that appending zeros is "allowed". In this way, it can safely be assumed that $N_r = N_t = N$ in the reminder of the chapter.

The highest rate I_{R} that can be transmitted over the channel (5.1) per input vector, subject to the fixed symbol alphabet Ω and a certain input symbol distribution, is referred to as the information rate of the system (capacity requires an optimization over the input distribution and constellation). According to Section 2.5, the information rate equals

$$\begin{aligned} I_{\mathrm{R}} &= I(\boldsymbol{Y};\boldsymbol{A}) \\ &= \mathfrak{h}(\boldsymbol{Y}) - \mathfrak{h}(\boldsymbol{Y}|\boldsymbol{A}) \end{aligned} \tag{5.2}$$

where $I(\boldsymbol{Y};\boldsymbol{A})$ is the mutual information operator and $\mathfrak{h}(\cdot)$ is the N-dimensional differential entropy operator defined as

$$\mathfrak{h}(\boldsymbol{Y}) = -\int p_{\boldsymbol{Y}}(\boldsymbol{y}) \log(p_{\boldsymbol{Y}}(\boldsymbol{y}))\, \mathrm{d}\boldsymbol{y}. \tag{5.3}$$

Note that in this chapter a bold capital letter denotes a random vector while a bold lower case letter denotes its realization; deterministic matrices are also written with bold capital letters. Unless stated otherwise, the natural logarithm is used which means that mutual informations are expressed in nats per channel use. Observe that if (5.1) is to be used to represent an ISI channel, a scaling of $I(\boldsymbol{Y};\boldsymbol{A})$ in (5.2) by $1/N$ is needed.

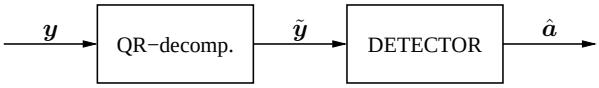

Figure 5.2: QR-decomposition of the channel matrix $\boldsymbol{H}$ prior to detection.

In order to reach the ultimate limit I_{R}, a maximum-a-posteriori detector described in Section 2.4 can be used in order to evaluate the posterior probabilities

$$p_{\boldsymbol{A}|\boldsymbol{Y}}\left(\boldsymbol{A}=\boldsymbol{a}|\boldsymbol{Y}=\boldsymbol{y}\right), \quad \forall \boldsymbol{a} \in \Omega^N.$$

As an alternative, the ML rule can be used. The symbolwise MAP detector is implemented by first performing a QR-decomposition of the channel matrix, and then running the BCJR algorithm on the remaining tree structure. This is shown schematically in Figure 5.2. Unless the channel matrix $\boldsymbol{H}$ possesses some special structure, the complexity of the BCJR algorithm is given by $|\Omega|^N$, which easily gets prohibitive as N and/or $|\Omega|$ become large. In next section the problem of optimally "shortening" the memory of the channel for signals described by (5.1) is addressed. The efficiency of the proposed detector which operates on a shortened model of the channel is measured by the highest communication rate that can be supported when the detector is used. Since the shortening detector is of reduced complexity, this rate must be strictly less than I_{R}. The advantage of this approach with respect to more common approaches, such as measuring the error rate performance of a coded system, is that it gives an ultimate performance limit characterizing the detector, and does not depend on the specific outer code adopted.

5.1.2 Reduced Complexity Trellis Based Detectors

According to Chapter 4, the input-output relation of the channel (5.1) is completely described through

$$\begin{aligned} p_{\boldsymbol{Y}|\boldsymbol{A}}(\boldsymbol{y}|\boldsymbol{a}) &= \frac{1}{(\pi N_0)^N} \exp\left(-\frac{\|\boldsymbol{y}-\boldsymbol{H}\boldsymbol{a}\|^2}{N_0}\right) \\ &= \frac{1}{(\pi N_0)^N} \exp\left(-\frac{\boldsymbol{y}^\dagger\boldsymbol{y} - 2\mathcal{R}\{\boldsymbol{a}^\dagger(\boldsymbol{H}^{\mathrm{r}})^\dagger\boldsymbol{y}\} + \boldsymbol{a}^\dagger\boldsymbol{G}\boldsymbol{a}}{N_0}\right) \end{aligned} \quad (5.4)$$

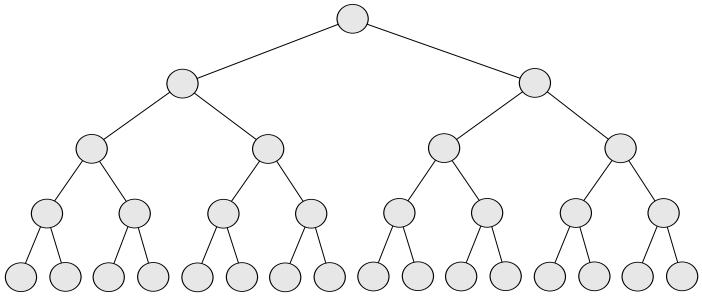

Figure 5.3: Search tree associated with ML detection of MIMO signals with $N = 4$ and BPSK inputs.

where as before $\boldsymbol{G} \triangleq \boldsymbol{H}^\dagger \boldsymbol{H}$, $\mathcal{R}\{x\}$ denotes the real part of x and "†" denotes Hermitian transpose. In [122] a reduced-complexity receiver based on (5.4) is introduced. It replaces (5.4) with

$$\tilde{p}(\boldsymbol{y}|\boldsymbol{a}) = \frac{1}{(\pi N^{\mathrm{r}})^N} \exp\left(-\frac{\boldsymbol{y}^\dagger \boldsymbol{y} - 2\mathcal{R}\{\boldsymbol{a}^\dagger (\boldsymbol{H}^{\mathrm{r}})^\dagger \boldsymbol{y}\} + \boldsymbol{a}^\dagger \boldsymbol{G}^{\mathrm{r}} \boldsymbol{a}}{N^{\mathrm{r}}}\right) \tag{5.5}$$

where the mismatched noise density[2] N^{r} and the matrices $\boldsymbol{H}^{\mathrm{r}}$ and $\boldsymbol{G}^{\mathrm{r}}$ are subject to optimization. Note that $\tilde{p}(\boldsymbol{y}|\boldsymbol{a})$ may not be a valid conditional probability density function, but that will be unimportant later.

Since the term $\exp(-\|\boldsymbol{y}\|^2/N^{\mathrm{r}})$ is constant with respect to the input $\boldsymbol{a}$, it is irrelevant for the detection process and can be removed in the optimization. It follows that it is possible, without loss of generality, to absorb N^{r} into $\boldsymbol{H}^{\mathrm{r}}$ and $\boldsymbol{G}^{\mathrm{r}}$. The mismatched noise density can therefore be set $N^{\mathrm{r}} = 1$. Furthermore, the constant π^{-N} is also irrelevant for detection purposes and can be removed. Consequently, instead of working with (5.5), the likelihood $\tilde{p}(\boldsymbol{y}|\boldsymbol{a})$ is redefined as

$$\tilde{p}(\boldsymbol{y}|\boldsymbol{a}) \triangleq \exp\left(2\mathcal{R}\{\boldsymbol{a}^\dagger (\boldsymbol{H}^{\mathrm{r}})^\dagger \boldsymbol{y}\} - \boldsymbol{a}^\dagger \boldsymbol{G}^{\mathrm{r}} \boldsymbol{a}\right). \tag{5.6}$$

[2]Note that $N^{\mathrm{r}} \neq N_r$. The number of receive antennas in a MIMO system is denoted by N_r, that is with the character r in the subscript while the mismatched noise density uses r in the superscript. It was also assumed earlier in this chapter that $N_r = N_t = N$ and hence, N_r will not be used in the reminder of the chapter.

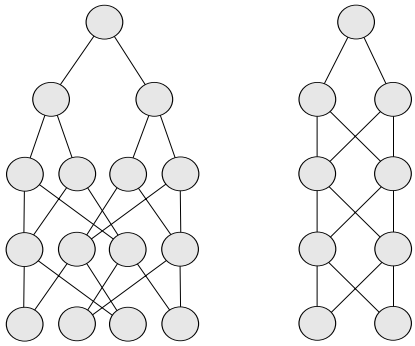

Figure 5.4: Trellises associated with the channel shortening detectors proposed in this chapter with $\nu = 1$ (right) and $\nu = 2$ (left). As in Figure 5.3, MIMO signals with $N = 4$ and BPSK inputs are assumed.

Observe that the need for trellis processing of (5.5) lies solely in the matrix $\boldsymbol{G}^{\mathrm{r}}$. In order to satisfy the reduced memory constraint, i.e., to "shorten the matrix", $\boldsymbol{G}^{\mathrm{r}}$ is constrained to satisfy the following property

$$(\boldsymbol{G}^{\mathrm{r}})_{mn} = 0 \quad \text{if } |m-n| > \nu \tag{5.7}$$

where $(\boldsymbol{G}^{\mathrm{r}})_{mn}$ denotes the element of the matrix $(\boldsymbol{G}^{\mathrm{r}})$ at row m and column n while ν denotes memory of the reduced trellis. Hence, symbolwise MAP detection based on (5.6), as proposed in this chapter, requires $|\Omega|^{\nu}$ states. The branch labels of the underlying trellis are uniquely given by the matrix $\boldsymbol{G}^{\mathrm{r}}$ and the symbol alphabet Ω.

Some examples of reduced trellises are shown in Figures 5.3 and 5.4. In Figure 5.3, the full search tree associated with ML detection of a MIMO signal with N = 4 and BPSK inputs, i.e., $\Omega = \{+1, -1\}$, is illustrated. In total there are $2^4 = 16$ leaf nodes, which is a measure of the ML complexity. Figure 5.4 shows examples of reduced trellises corresponding to $\nu = 1$ and 2, respectively. The search tree in Figure 5.3 has been reduced from 16 leaf nodes into trellises with only 2 and 4 states, respectively. Hence, in the case of MIMO transmission, a complexity reduction by a factor $|\Omega|^{N-\nu}$ is achieved in general.

Conventional channel shortening, [123]–[131], can be seen as the special case of (5.5) when the matrix $\boldsymbol{H}^{\mathrm{r}}$ factorizes as $\boldsymbol{H}^{\mathrm{r}} = \boldsymbol{W}^{\dagger}\boldsymbol{F}$ and $\boldsymbol{G}^{\mathrm{r}} = \boldsymbol{F}^{\dagger}\boldsymbol{F}$. Implicit in such factorization is that $\boldsymbol{F}$ is regarded as the shortened channel, while $\boldsymbol{W}$ is

the "channel shortener", i.e., the task of $\boldsymbol{W}$ is to force $\boldsymbol{WH}$ close to $\boldsymbol{F}$. Since the term $\boldsymbol{y}^\dagger\boldsymbol{y}$ is irrelevant for the detection process, the conventional channel shortening method implies that (5.4) is replaced by

$$\hat{p}(\boldsymbol{y}|\boldsymbol{a}) = \frac{1}{(\pi N_0)^N} \exp\left(-\frac{\|\boldsymbol{Wy} - \boldsymbol{Fa}\|^2}{N_0}\right) \tag{5.8}$$

where, in order to satisfy the memory-ν constraint, the shortened channel $\boldsymbol{F}$ should only contain $\nu + 1$ non-zero diagonals.

In this chapter we will compare the proposed channel shortening detector with the MMSE optimized detector from [123] which minimizes the following cost function:

$$\min_{\boldsymbol{W},\boldsymbol{F}} \lim_{N\to\infty} \frac{1}{N}\mathbb{E}\left[||\boldsymbol{Wy} - \boldsymbol{F}\tilde{\boldsymbol{a}}||^2\right] \tag{5.9}$$

where $\tilde{\boldsymbol{a}}$ are the actual transmitted symbols, i.e., $\boldsymbol{y} = \boldsymbol{H}\tilde{\boldsymbol{a}} + \boldsymbol{w}$. Additionally, in this chapter it is shown that detectors limited to the form (5.8) are not optimal from a mutual information perspective. The reason is that the matrix $\boldsymbol{G}^{\mathrm{r}}$ in (5.6) that maximizes the mutual information may not be positive semi-definite, so that no factorization $\boldsymbol{G}^{\mathrm{r}} = \boldsymbol{F}^\dagger\boldsymbol{F}$ exists. Consequently, conventional channel shortening algorithms are not optimal from a mutual information perspective since they are restricted to input-output relations of the form (5.8). Finally, it is remarked that the form (5.8) is *not* more general than (5.6) in the case of equal power input symbols, i.e.,

$$|a_n|^2 = P, \quad \forall a_n \in \Omega$$

for some constant P. This is easiest seen by considering the last term in (5.6), i.e. $\exp(-\boldsymbol{a}^\dagger\boldsymbol{G}^{\mathrm{r}}\boldsymbol{a})$. If the mutual information optimal $\boldsymbol{G}^{\mathrm{r}}$ is not positive semi-definite there is a constant σ which will allow the factorization

$$\boldsymbol{F}^\dagger\boldsymbol{F} = (\boldsymbol{G}^{\mathrm{r}} + \sigma\boldsymbol{I}) = \tilde{\boldsymbol{G}}^{\mathrm{r}}.$$

By simple manipulations of (5.6) we obtain

$$\begin{aligned}
\tilde{p}(\boldsymbol{y}|\boldsymbol{a}) &= \exp\left(2\mathcal{R}\{\boldsymbol{a}^\dagger(\boldsymbol{H}^{\mathrm{r}})^\dagger\boldsymbol{y}\} - \boldsymbol{a}^\dagger\boldsymbol{G}^{\mathrm{r}}\boldsymbol{a}\right) \\
&\propto \exp\left(2\mathcal{R}\{\boldsymbol{a}^\dagger(\boldsymbol{H}^{\mathrm{r}})^\dagger\boldsymbol{y}\} - \boldsymbol{a}^\dagger\boldsymbol{G}^{\mathrm{r}}\boldsymbol{a} - P\sigma\mathrm{Tr}(I)\right) \\
&= \exp\left(2\mathcal{R}\{\boldsymbol{a}^\dagger(\boldsymbol{H}^{\mathrm{r}})^\dagger\boldsymbol{y}\} - \boldsymbol{a}^\dagger\boldsymbol{G}^{\mathrm{r}}\boldsymbol{a} - \sigma\boldsymbol{a}^\dagger\boldsymbol{a}\right) \\
&= \exp\left(2\mathcal{R}\{\boldsymbol{a}^\dagger(\boldsymbol{H}^{\mathrm{r}})^\dagger\boldsymbol{y}\} - \boldsymbol{a}^\dagger\tilde{\boldsymbol{G}}^{\mathrm{r}}\boldsymbol{a}\right) \\
&\propto \exp\left(-\|(\boldsymbol{F}^\dagger)^{-1}(\boldsymbol{H}^{\mathrm{r}})^\dagger\boldsymbol{y} - \boldsymbol{F}\boldsymbol{a}\|^2\right) \\
&= \exp\left(-\|\boldsymbol{W}\boldsymbol{y} - \boldsymbol{F}\boldsymbol{a}\|^2\right)
\end{aligned}$$

where we in the second and fifth steps have used that any constants in (5.6) are irrelevant for the detection process.

5.1.3 Achievable Information Rates of the Reduced Complexity Detector

A detector that operates on the basis of $\tilde{p}(\boldsymbol{y}|\boldsymbol{a})$ given in (5.5), instead of the true conditional density $p_{\boldsymbol{Y}|\boldsymbol{A}}(\boldsymbol{y}|\boldsymbol{a})$, can support an arbitrarily small error probability if the communication rate is smaller than I_{AIR} where I_{AIR} is referred to as the *achievable information rate*.[3] Further, it is known that for any strictly positive $\tilde{p}(\boldsymbol{y}|\boldsymbol{a})$ [136]

$$\begin{aligned}
I_{\mathrm{AIR}} &\geq I_{\mathrm{LB}} \\
&\triangleq -\mathbb{E}_{\boldsymbol{Y}}\left[\log_2\left(\tilde{p}(\boldsymbol{y})\right)\right] + \mathbb{E}_{\boldsymbol{Y},\boldsymbol{A}}\left[\log_2\left(\tilde{p}(\boldsymbol{y}|\boldsymbol{a})\right)\right] \qquad (5.10)
\end{aligned}$$

where $\mathbb{E}_{\boldsymbol{Y}}$ denotes the expectation operator with respect to the random variable $\boldsymbol{Y}$ and

$$\tilde{p}(\boldsymbol{y}) \triangleq \sum_{\boldsymbol{s}\in\Omega^N} \tilde{p}(\boldsymbol{y}|\boldsymbol{s})\mathrm{Pr}(\boldsymbol{s}). \qquad (5.11)$$

Note that the lower bound I_{LB} directly depends on the choices of $\boldsymbol{G}^{\mathrm{r}}$ and $\boldsymbol{H}^{\mathrm{r}}$. In this chapter the objective is to maximize the lower bound over the choices of $\boldsymbol{G}^{\mathrm{r}}$ and $\boldsymbol{H}^{\mathrm{r}}$ in order to maximize the achievable information rate I_{AIR}. This optimization equals

[3] Observe that this is not the generalized mutual information, which requires an optimization over the input constellation and distribution. The achievable information rate corresponds to the generalized mutual information without this optimization.

$$\max_{\boldsymbol{G}^{\mathrm{r}},\boldsymbol{H}^{\mathrm{r}}} I_{\mathrm{LB}}$$

and is treated next.

5.2 Optimization of I_{LB} for Gaussian Inputs

The goal of this section is to maximize I_{LB} which is a complicated task for a discrete alphabet Ω. However, for Gaussian inputs, a closed form expression can be obtained. One may ask what the value of an optimized detector for Gaussian inputs when used for, say, M-QAM inputs really is? But when the optimized detectors for Gaussian inputs are used for discrete alphabets, Monte Carlo evaluations [137] will verify that the ensuing I_{LB} is excellent.

Under the assumption of Gaussian inputs, the following can be proved

Proposition 1. *With zero-mean, unit-variance, circularly symmetric complex Gaussian inputs, and a given Hermitian matrix $\boldsymbol{G}^{\mathrm{r}}$ with smallest eigenvalue larger than -1, the optimal receiver filter is*

$$\boldsymbol{H}^{\mathrm{r}} = \left[\boldsymbol{H}\boldsymbol{H}^{\dagger} + N_0\boldsymbol{I}\right]^{-1}\boldsymbol{H}\left[\boldsymbol{G}^{\mathrm{r}} + \boldsymbol{I}\right].$$

For this $\boldsymbol{H}^{\mathrm{r}}$, I_{LB} equals

$$I_{\mathrm{LB}} = \log\left(\det\left(\boldsymbol{I} + \boldsymbol{G}^{\mathrm{r}}\right)\right) + \mathrm{Tr}\left(\left[\boldsymbol{G}^{\mathrm{r}} + \boldsymbol{I}\right]\boldsymbol{H}^{\dagger}\left[\boldsymbol{H}\boldsymbol{H}^{\dagger} + N_0\boldsymbol{I}\right]^{-1}\boldsymbol{H}\right) - \mathrm{Tr}\left(\boldsymbol{G}^{\mathrm{r}}\right).$$

□

Proof. The objective is to compute the two terms in (5.10), i.e., $-\mathbb{E}_{\boldsymbol{Y}}\left[\log_2\left(\tilde{p}(\boldsymbol{y})\right)\right]$ and $\mathbb{E}_{\boldsymbol{Y},\boldsymbol{A}}\left[\log_2\left(\tilde{p}(\boldsymbol{y}|\boldsymbol{a})\right)\right]$. Let $\boldsymbol{G}^{\mathrm{r}} = \boldsymbol{Q}\boldsymbol{\Lambda}^{\mathbf{g}}\boldsymbol{Q}^{\dagger}$ denote the eigenvalue decomposition of $\boldsymbol{G}^{\mathrm{r}}$ and set $\boldsymbol{z} = \boldsymbol{Q}^{\dagger}\boldsymbol{a}$. Using these identities in (5.6) gives,

$$\begin{aligned}
\tilde{p}(\boldsymbol{y}) &= \int \tilde{p}(\boldsymbol{y}|\boldsymbol{a}) p_{\boldsymbol{A}}(\boldsymbol{a}) \mathrm{d}\boldsymbol{a} \\
&= \frac{1}{\pi^N} \int \exp(-\|\boldsymbol{z}\|^2) \exp\left(2\mathcal{R}\{\boldsymbol{z}^\dagger \boldsymbol{Q}^\dagger (\boldsymbol{H}^{\mathrm{r}})^\dagger \boldsymbol{y}\} - \boldsymbol{z}^\dagger \boldsymbol{\Lambda}^{\mathbf{g}} \boldsymbol{z}\right) \mathrm{d}\boldsymbol{z} \\
&= \frac{1}{\pi^N} \int \prod_{n=1}^{N} \exp\left(2\mathcal{R}\{z_n^\dagger \, d_n\} - |z_n|^2 \left[\lambda_n^{\mathrm{g}} + 1\right]\right) \mathrm{d}z_n \\
&= \prod_{n=1}^{N} \frac{1}{\lambda_n^{\mathrm{g}} + 1} \exp\left(\frac{|d_n|^2}{\lambda_n^{\mathrm{g}} + 1}\right) \qquad (5.12)
\end{aligned}$$

where λ_n^{g} is the nth element of the diagonal matrix $\boldsymbol{\Lambda}^{\mathbf{g}}$, i.e., $\lambda_n^{\mathrm{g}} = (\boldsymbol{\Lambda}^{\mathbf{g}})_{nn}$. In (5.12) the $N \times 1$ column vector $\boldsymbol{d}$ is defined as $\boldsymbol{d} \triangleq \boldsymbol{Q}^\dagger (\boldsymbol{H}^{\mathrm{r}})^\dagger \boldsymbol{y}$. The first term of (5.10), i.e. the quantity $-\mathbb{E}_{\boldsymbol{Y}} \log(\tilde{p}(\boldsymbol{y}))$, can now be computed as

$$\begin{aligned}
-\mathbb{E}_{\boldsymbol{Y}} \log(\tilde{p}(\boldsymbol{y})) &= -\mathbb{E}_{\boldsymbol{Y}} \left[\sum_{n=1}^{N} \left[\log\left(\frac{1}{\lambda_n^{\mathrm{g}} + 1}\right) + \frac{|d_n|^2}{\lambda_n^{\mathrm{g}} + 1}\right]\right] \\
&= \sum_{n=1}^{N} \left[\log(\lambda_n^{\mathrm{g}} + 1) - \frac{\mathbb{E}_{\boldsymbol{Y}}[|d_n|^2]}{\lambda_n^{\mathrm{g}} + 1}\right]. \qquad (5.13)
\end{aligned}$$

Define $\boldsymbol{R}$ as the expectation

$$\begin{aligned}
\boldsymbol{R} &\triangleq \mathbb{E}\left[\boldsymbol{d}\boldsymbol{d}^\dagger\right] = \mathbb{E}\left[\boldsymbol{Q}^\dagger (\boldsymbol{H}^{\mathrm{r}})^\dagger (\boldsymbol{H}\boldsymbol{a} + \boldsymbol{w})(\boldsymbol{H}\boldsymbol{a} + \boldsymbol{w})^\dagger \boldsymbol{H}^{\mathrm{r}} \boldsymbol{Q}\right] \qquad (5.14) \\
&= \boldsymbol{Q}^\dagger (\boldsymbol{H}^{\mathrm{r}})^\dagger \mathbb{E}\left[(\boldsymbol{H}\boldsymbol{a} + \boldsymbol{w})(\boldsymbol{H}\boldsymbol{a} + \boldsymbol{w})^\dagger\right] \boldsymbol{H}^{\mathrm{r}} \boldsymbol{Q} \\
&= \boldsymbol{Q}^\dagger (\boldsymbol{H}^{\mathrm{r}})^\dagger \left(\boldsymbol{H}\mathbb{E}\left[\boldsymbol{a}\boldsymbol{a}^\dagger\right] (\boldsymbol{H}^{\mathrm{r}})^\dagger + \mathbb{E}\left[\boldsymbol{w}\boldsymbol{w}^\dagger\right]\right) \boldsymbol{H}^{\mathrm{r}} \boldsymbol{Q} \\
&= \boldsymbol{Q}^\dagger (\boldsymbol{H}^{\mathrm{r}})^\dagger \boldsymbol{H}\boldsymbol{H}^\dagger \boldsymbol{H}^{\mathrm{r}} \boldsymbol{Q} + N_0 \boldsymbol{Q}^\dagger (\boldsymbol{H}^{\mathrm{r}})^\dagger \boldsymbol{H}^{\mathrm{r}} \boldsymbol{Q}
\end{aligned}$$

where we have used that $\mathbb{E}\left[\boldsymbol{a}\boldsymbol{a}^\dagger\right] = \boldsymbol{I}$, $\mathbb{E}\left[\boldsymbol{w}\boldsymbol{w}^\dagger\right] = N_0 \boldsymbol{I}$ and $\mathbb{E}\left[\boldsymbol{a}\boldsymbol{w}^\dagger\right] = \mathbb{E}\left[\boldsymbol{w}\boldsymbol{a}^\dagger\right] = \boldsymbol{0}$. As before, the matrix $\boldsymbol{0}$ is an all-zero matrix. We then arrive at

$$-\mathbb{E}_{\boldsymbol{Y}} \log(\tilde{p}(\boldsymbol{Y})) = \sum_{n=1}^{N} \left[\log(\lambda_n^{\mathrm{g}} + 1) - \frac{R_{nn}}{\lambda_n^{\mathrm{g}} + 1}\right] \qquad (5.15)$$

where $R_{nn} = (\boldsymbol{R})_{nn}$, i.e., the element at row n and column n of the matrix $\boldsymbol{R}$.

Now consider the computation of the second term in (5.10). We have that

$$\begin{aligned} -\mathbb{E}_{\boldsymbol{Y},\boldsymbol{A}}\left[\log\left(\tilde{p}(\boldsymbol{y}|\boldsymbol{a})\right)\right] &= \mathbb{E}_{\boldsymbol{Y},\boldsymbol{A}}\left[\boldsymbol{a}^\dagger \boldsymbol{G}^{\mathrm{r}}\boldsymbol{a} - 2\mathcal{R}\{\boldsymbol{a}^\dagger \boldsymbol{H}^{\mathrm{r}}\boldsymbol{y}\}\right] \\ &= \mathrm{Tr}(\boldsymbol{G}^{\mathrm{r}}) - 2\mathcal{R}\{\mathrm{Tr}((\boldsymbol{H}^{\mathrm{r}})^\dagger \boldsymbol{H})\}. \end{aligned} \tag{5.16}$$

By combining the two terms from (5.15) and (5.16), we obtain

$$I_{\mathrm{LB}} = \sum_{n=1}^{N}\left[\log\left(\lambda_n^{\mathrm{g}}+1\right) - \frac{R_{nn}}{\lambda_n^{\mathrm{g}}+1} - \lambda_n^{\mathrm{g}}\right] + 2\mathcal{R}\{\mathrm{Tr}((\boldsymbol{H}^{\mathrm{r}})^\dagger \boldsymbol{H})\}. \tag{5.17}$$

Next consider the optimization of (5.17) over $\boldsymbol{H}^{\mathrm{r}}$. Since $\boldsymbol{\Lambda}^{\mathrm{g}}$ is a diagonal matrix, we have that

$$\begin{aligned} \sum_{n}^{N}\frac{R_{nn}}{\lambda^{\mathrm{g}}+1} &= \mathrm{Tr}(\boldsymbol{R}\left[\boldsymbol{\Lambda}^{\mathrm{g}}+\boldsymbol{I}\right]^{-1}) \\ &= \mathrm{Tr}\left(\boldsymbol{Q}^\dagger(\boldsymbol{H}^{\mathrm{r}})^\dagger\left[\boldsymbol{H}\boldsymbol{H}^\dagger + N_0\boldsymbol{I}\right]\boldsymbol{H}^{\mathrm{r}}\boldsymbol{Q}\left[\boldsymbol{\Lambda}^{\mathrm{g}}+\boldsymbol{I}\right]^{-1}\right) \\ &= \mathrm{Tr}\left((\boldsymbol{H}^{\mathrm{r}})^\dagger\left[\boldsymbol{H}\boldsymbol{H}^\dagger + N_0\boldsymbol{I}\right]\boldsymbol{H}^{\mathrm{r}}\left[\boldsymbol{G}^{\mathrm{r}}+\boldsymbol{I}\right]^{-1}\right). \end{aligned} \tag{5.18}$$

In order to optimize I_{LB} with respect to $\boldsymbol{H}^{\mathrm{r}}$ we should solve

$$\boldsymbol{H}^{\mathrm{r}}_{\mathrm{opt}} = \arg\max_{\boldsymbol{X}} f(\boldsymbol{X}) \tag{5.19}$$

with

$$f(\boldsymbol{X}) \triangleq 2\mathcal{R}\{\mathrm{Tr}(\boldsymbol{X}^\dagger\boldsymbol{H})\} - \mathrm{Tr}\left(\boldsymbol{X}^\dagger\left[\boldsymbol{H}\boldsymbol{H}^\dagger + N_0\boldsymbol{I}\right]\boldsymbol{X}\left[\boldsymbol{G}^{\mathrm{r}}+\boldsymbol{I}\right]^{-1}\right).$$

Since $f(\boldsymbol{X})$ is a real-valued function of the complex-valued matrix $\boldsymbol{X}$, we have that

$$\begin{aligned} \nabla_{\boldsymbol{X}} f(\boldsymbol{X}) &= \frac{\partial f(\boldsymbol{X})}{\partial \mathcal{R}\{\boldsymbol{X}\}} + i\frac{\partial f(\boldsymbol{X})}{\partial \mathcal{I}\{\boldsymbol{X}\}} \\ &= 2\boldsymbol{H} - 2\left[\boldsymbol{H}\boldsymbol{H}^\dagger + N_0\boldsymbol{I}\right]\boldsymbol{X}\left[\boldsymbol{G}^{\mathrm{r}}+\boldsymbol{I}\right]^{-1}. \end{aligned} \tag{5.20}$$

By setting $\nabla_{\boldsymbol{X}} f(\boldsymbol{X}) = \boldsymbol{0}$ a solution to (5.19) is obtained. The front-end filter that maximizes I_{LB} in (5.17) is given by

$$\boldsymbol{H}^{\mathrm{r}}_{\mathrm{opt}} = \left[\boldsymbol{H}\boldsymbol{H}^{\dagger} + N_0\boldsymbol{I}\right]^{-1} \boldsymbol{H}\left[\boldsymbol{G}^{\mathrm{r}} + \boldsymbol{I}\right]. \tag{5.21}$$

Inserting (5.21) into (5.17) gives after some manipulations

$$I_{\mathrm{LB}} = \log\left(\det\left(\boldsymbol{I} + \boldsymbol{G}^{\mathrm{r}}\right)\right) + \mathrm{Tr}\left(\left[\boldsymbol{G}^{\mathrm{r}} + \boldsymbol{I}\right]\boldsymbol{H}^{\dagger}\left[\boldsymbol{H}\boldsymbol{H}^{\dagger} + N_0\boldsymbol{I}\right]^{-1}\boldsymbol{H}\right) - \mathrm{Tr}\left(\boldsymbol{G}^{\mathrm{r}}\right)$$

which proves the proposition.

□

Interestingly, the optimal front-end filter $\boldsymbol{H}^{\mathrm{r}}$ equals the standard MMSE/Wiener filter, compensated by the receiver trellis processing, that is

$$\boldsymbol{H}^{\mathrm{r}}_{\mathrm{opt}} = \boldsymbol{H}_{\mathrm{MMSE}}[\boldsymbol{G}^{\mathrm{r}} + \boldsymbol{I}] \tag{5.22}$$

where $\boldsymbol{H}_{\mathrm{MMSE}} = [\boldsymbol{H}\boldsymbol{H}^{\dagger} + N_0\boldsymbol{I}]^{-1}\boldsymbol{H}$. The trellis processing is represented through $\boldsymbol{G}^{\mathrm{r}} + \boldsymbol{I}$ rather than only $\boldsymbol{G}^{\mathrm{r}}$. This is a surprising fact since in [138], the optimal front-end filter of the proposed MMSE based channel shortening detector equals

$$\tilde{\boldsymbol{H}}^{\mathrm{r}}_{\mathrm{opt}} = \boldsymbol{H}_{\mathrm{MMSE}}\,\boldsymbol{G}^{\mathrm{r}}. \tag{5.23}$$

It is interesting to observe that the first term of the achievable information rate, i.e. $\log\left(\det\left(\boldsymbol{I} + \boldsymbol{G}^{\mathrm{r}}\right)\right)$, equals the conventional mutual information for a vector channel with associated Gram matrix $\boldsymbol{G}^{\mathrm{r}}$. The penalty terms for having a mismatched channel model are linear in $\boldsymbol{G}^{\mathrm{r}}$.

Consider now the optimization of I_{LB}. By the eigenvalue assumption in Proposition 1, it follows that $\boldsymbol{I} + \boldsymbol{G}^{\mathrm{r}}$ is positive definite, hence it has a Cholesky Factorization $\boldsymbol{I} + \boldsymbol{G}^{\mathrm{r}} = \boldsymbol{U}^{\dagger}\boldsymbol{U}$. Due to the memory constraint (5.7), it follows that the upper triangular matrix $\boldsymbol{U}$ only contains $\nu + 1$ nonzero diagonals. The results obtained from the maximization of the achievable information rate over $\boldsymbol{G}^{\mathrm{r}}$ are summarized in

Proposition 2. *Define*

$$\boldsymbol{B} \triangleq -\boldsymbol{H}^\dagger \left[\boldsymbol{H}\boldsymbol{H}^\dagger + N_0\boldsymbol{I}\right]^{-1} \boldsymbol{H} + \boldsymbol{I}. \tag{5.24}$$

Let $\tilde{\boldsymbol{B}}_n^\nu$ *denote the submatrix*

$$\tilde{\boldsymbol{B}}_n^\nu = \begin{bmatrix} B_{n+1\,n+1} & \cdots & B_{n+1\ \min(N,n+\nu)} \\ \vdots & \ddots & \vdots \\ B_{\min(N,n+\nu)\,n+1} & \cdots & B_{\min(N,n+\nu)\ \min(N,n+\nu)} \end{bmatrix}$$

of $\boldsymbol{B}$*, and let* $\boldsymbol{b}_n^\nu$ *be the row vector* $\boldsymbol{b}_n^\nu = [B_{n\,n+1}, \ldots B_{N\ \min(M,n+\nu)}]$*. For* $n = N$*,* $\tilde{\boldsymbol{B}}_n^\nu = 0$ *and* $\boldsymbol{b}_n^\nu = 0$*. Let further* $\boldsymbol{u}_n^\nu$ *denote the row vector* $\boldsymbol{u}_n^\nu = [u_{n\,n+1}, \ldots u_{N\ \min(M,n+\nu)}]$*, where* $\{u_{nm}\}$ *are the elements of* $\boldsymbol{U}$*. Then*

$$\max_{\boldsymbol{G}^{\mathrm{r}}} I_{\mathrm{LB}} = \sum_{n=1}^{N} \log\left(\frac{1}{c_n}\right), \tag{5.25}$$

where the constants c_n *are given by*

$$c_n = B_{nn} - \boldsymbol{b}_n^\nu (\tilde{\boldsymbol{B}}_n^\nu)^{-1} (\boldsymbol{b}_n^\nu)^\dagger.$$

The optimal $\boldsymbol{G}^{\mathrm{r}} = \boldsymbol{U}\boldsymbol{U}^\dagger - \boldsymbol{I}$ *is constructed from*

$$u_{nn} = \frac{1}{\sqrt{c_n}}$$

and

$$\boldsymbol{u}_n^\nu = -u_{nn}\boldsymbol{b}_n^\nu (\tilde{\boldsymbol{B}}_n^\nu)^{-1}.$$

□

Proof. We can manipulate I_{LB} into

$$\begin{aligned} I_{\mathrm{LB}} &= \log(\det(\boldsymbol{U}^\dagger\boldsymbol{U})) + \mathrm{Tr}\left(\boldsymbol{U}\left[\boldsymbol{H}^\dagger\left[\boldsymbol{H}\boldsymbol{H}^\dagger + N_0\boldsymbol{I}\right]^{-1}\boldsymbol{H} - \boldsymbol{I}\right]\boldsymbol{U}^\dagger\right) + \mathrm{Tr}(\boldsymbol{I}) \\ &= 2\sum_{n=1}^{N}\log(u_{nn}) - \mathrm{Tr}\left(\boldsymbol{U}\boldsymbol{B}\boldsymbol{U}^\dagger\right) + N \end{aligned} \tag{5.26}$$

where the upper triangular matrix $\boldsymbol{U}$ has elements $\{u_{nm}\}_{m\geq n}$. Let $\boldsymbol{U_H}\boldsymbol{\Sigma}\boldsymbol{V}^\dagger$ denote the singular value decomposition of $\boldsymbol{H}$. Then the matrix $\boldsymbol{B}$ can be expressed as

$$\boldsymbol{B} = N_0\boldsymbol{V}\left[\boldsymbol{\Sigma}^2 + N_0\boldsymbol{I}\right]^{-1}\boldsymbol{V}^\dagger$$

which is always positive definite for $N_0 > 0$. Since no off-diagonal elements in $\boldsymbol{U}$ appear in the logarithm, (5.26) can be optimized over the diagonal and off-diagonal elements separately as

$$\max_{\boldsymbol{U}} I_{\mathrm{LB}} = \max_{\{u_{nn}\}}\left[2\sum_{n=1}^{N}\log(u_{nn}) + N - \left[\min_{\{u_{nm}\}_{n+1\leq m\leq \min(n+\nu,N)}} \mathrm{Tr}\left(\boldsymbol{U}\boldsymbol{B}\boldsymbol{U}^\dagger\right)\right]\right]. \tag{5.27}$$

With the definitions made in the statement of the Proposition, we have

$$\mathrm{Tr}(\boldsymbol{U}\boldsymbol{B}\boldsymbol{U}^\dagger) = \sum_{n=1}^{N}[u_{nn}\,\boldsymbol{u}_n^\nu]\begin{bmatrix} B_{nn} & \boldsymbol{b}_n^\nu \\ (\boldsymbol{b}_n^\nu)^\dagger & \tilde{\boldsymbol{B}}_n^\nu \end{bmatrix}\begin{bmatrix} u_{nn} \\ (\boldsymbol{u}_n^\nu)^\dagger \end{bmatrix}.$$

The derivative with respect to $\boldsymbol{u}_n^\nu$ equals

$$\frac{\partial}{\partial \boldsymbol{u}_n^\nu}\mathrm{Tr}(\boldsymbol{U}\boldsymbol{B}\boldsymbol{U}^\dagger) = 2u_{nn}\boldsymbol{b}_n^\nu + 2\boldsymbol{u}_n^\nu\tilde{\boldsymbol{B}}_n^\nu.$$

Setting the derivative equal to zero yields the $\boldsymbol{u}_n^\nu$ that minimizes $\mathrm{Tr}(\boldsymbol{U}\boldsymbol{B}\boldsymbol{U}^\dagger)$, and it is

$$(\boldsymbol{u}_n^\nu)^{\mathrm{opt}} = -u_{nn}\boldsymbol{b}_n^\nu(\tilde{\boldsymbol{B}}_n^\nu)^{-1}.$$

By inserting this expression for $\boldsymbol{u}_n^\nu$ back into (5.27) gives

$$\max_{\boldsymbol{U}} I_{\mathrm{LB}} = \max_{\{u_{nn}\}} 2\sum_{n=1}^{N}\log(u_{nn}) + N - \sum_{n=1}^{N}u_{nn}^2 c_n. \tag{5.28}$$

Now we need to maximize I_{LB} over the diagonal elements of the matrix $\boldsymbol{U}$. By taking the derivative of (5.28) with respect to u_{nn} and setting it equal to zero, we obtain

$$u_{nn}^{\text{opt}} = \frac{1}{\sqrt{c_n}}.$$

Inserting this into (5.28) maximizes I_{LB}, which is

$$\max_{\boldsymbol{U}} I_{\text{LB}} = \sum_{n=1}^{N} \log\left(\frac{1}{c_n}\right). \tag{5.29}$$

This concludes the proof. □

By making use of the matrix inversion lemma [139], the optimal achievable rate can be expressed as

$$\begin{aligned} \max_{\boldsymbol{G}^{\text{r}}} I_{\text{LB}} &= \sum_{n=1}^{N} \log\left(\frac{1}{c_n}\right) \\ &= \sum_{n=1}^{N} \log\Big(\left((\tilde{\boldsymbol{B}}_{n-1}^{\nu+1})^{-1}\right)_{11},\Big). \end{aligned} \tag{5.30}$$

However, no additional insights have been found from this form. Note that with $\nu = N-1$, i.e., a full complexity detector, $I_{\text{LB}} = \log(\det(\boldsymbol{I} + \boldsymbol{H}\boldsymbol{H}^{\dagger}/N_0))$. With $\nu = 0$, the performance of an MMSE detector is obtained. Hence, the proposed scheme trades detection complexity against achievable information rate and is general enough to include optimal schemes at full and minimum complexity.

The special case of ISI channels is treated next.

5.2.1 ISI Receivers

According to Section 2.6, the special case of ISI channels can also be represented by the discrete-time linear model in (5.1). In this case, the channel matrix $\boldsymbol{H}$ represents circular convolution with a L-tap discrete-time response $\boldsymbol{h}$. As N grows large, the circular convolution represents normal convolution to any given precision, see [67] for an extensive information-theoretical treatment.

Propositions 1 and 2 can still be applied to derive the corresponding mutual information optimized detector, but since the block length N is large for ISI channels, typically 1000 or more, simplifications are possible. The matrices $\boldsymbol{H}^{\text{r}}$ and $\boldsymbol{G}^{\text{r}}$ are uniquely characterized by the discrete sequences $\boldsymbol{h}^{\text{r}}$ and $\boldsymbol{g}^{\text{r}}$. Let $H^{\text{r}}(\omega)$ and $G^{\text{r}}(\omega)$ denote their respective Fourier transform defined as

$$H^{\mathrm{r}}(\omega) = \sum_{k=-\infty}^{\infty} h_k^{\mathrm{r}}\, e^{-j\omega k} \tag{5.31}$$

and similarly for $G^{\mathrm{r}}(\omega)$. In (5.31) h_k^{r} denotes the kth element of $\boldsymbol{h}^{\mathrm{r}}$.

For ISI channels, the quantity of interest is

$$I_{\mathrm{LB}} = \lim_{N\to\infty} \frac{1}{N} \left[-\mathbb{E}_{\boldsymbol{Y}} \left[\log_2 \left(\tilde{p}(\boldsymbol{y}) \right) \right] + \mathbb{E}_{\boldsymbol{Y},\boldsymbol{A}} \left[\log_2 \left(\tilde{p}(\boldsymbol{y}|\boldsymbol{a}) \right) \right] \right]. \tag{5.32}$$

In order to get better understanding of I_{LB}, Proposition 1 is translated into an ISI formulation,

Proposition 3. *For ISI channels with transfer function $H(\omega)$ and a particular receiver trellis represented through $G^{\mathrm{r}}(\omega)$, where $\min_\omega G^{\mathrm{r}}(\omega) > -1$, the optimal receiver filter is given by*

$$H^{\mathrm{r}}(\omega) = \frac{H^{\dagger}(\omega)}{|H(\omega)|^2 + N_0} (G^{\mathrm{r}}(\omega) + 1).$$

Furthermore, I_{LB} becomes

$$I_{\mathrm{LB}} = \frac{1}{2\pi} \int_{-\pi}^{\pi} \log(G^{\mathrm{r}}(\omega) + 1) + \frac{|H(\omega)|^2 - N_0\, G^{\mathrm{r}}(\omega)}{|H(\omega)|^2 + N_0}\, \mathrm{d}\omega. \tag{5.33}$$

□

Proof. Denote the channel matrix by $\boldsymbol{H} = \boldsymbol{Q}\boldsymbol{\Lambda}\boldsymbol{Q}^{\dagger}$. For circular ISI channels the matrix $\boldsymbol{Q}$ equals the discrete Fourier transform matrix. Represent also $\boldsymbol{H}^{\mathrm{r}} = \boldsymbol{Q}\boldsymbol{\Lambda}^{\mathrm{r}}\boldsymbol{Q}^{\dagger}$ and $\boldsymbol{G}^{\mathrm{r}} = \boldsymbol{Q}\boldsymbol{\Lambda}^{\mathrm{g}}\boldsymbol{Q}^{\dagger}$. With these eigenvalue factorizations, the matrix $\boldsymbol{R}$ simplifies into

$$\boldsymbol{R} = (\boldsymbol{\Lambda}^{\mathrm{r}})^{\dagger}\boldsymbol{\Lambda}^{\mathrm{r}} \left[\boldsymbol{\Lambda}^{\dagger}\boldsymbol{\Lambda} + N_0 \boldsymbol{I} \right].$$

Furthermore,

$$\mathrm{Tr}((\boldsymbol{H}^{\mathrm{r}})^{\dagger}\boldsymbol{H}) = \mathrm{Tr}((\boldsymbol{\Lambda}^{\mathrm{r}})^{\dagger}\boldsymbol{\Lambda}).$$

Together, this leaves us with

$$I_{\mathrm{LB}} = \lim_{N\to\infty} \frac{1}{N} \sum_n \left[\log(\lambda_n^{\mathrm{g}} + 1) - \lambda_n^{\mathrm{g}} - \frac{R_{nn}}{\lambda_n^{\mathrm{g}} + 1} + 2\mathcal{R}\{(\lambda_n^{\mathrm{r}})^\dagger \lambda_n\} \right], \quad (5.34)$$

where $R_{nn} = |\lambda_n^{\mathrm{r}}|^2(|\lambda_n|^2 + N_0)$. By expressing $\lambda_n^{\mathrm{r}} = |\lambda_n^{\mathrm{r}}| \exp(i\gamma_n^{\mathrm{r}})$ and $\lambda_n = |\lambda_n| \exp(i\gamma_n)$, it is clear that I_{LB} is maximized by taking $\gamma_n^{\mathrm{r}} = -\gamma_n$. This yields,

$$I_{\mathrm{LB}} = \lim_{N\to\infty} \frac{1}{N} \sum_n \log(\lambda_n^{\mathrm{g}} + 1) - \lambda_n^{\mathrm{g}} - \frac{R_{nn}}{\lambda_n^{\mathrm{g}} + 1} + 2|\lambda_n^{\mathrm{r}}||\lambda_n|. \quad (5.35)$$

Setting the partial derivative of I_{LB} with respect to λ_n^{r} to zero gives

$$\frac{\partial I_{\mathrm{LB}}}{\partial \lambda_n^{\mathrm{g}}} = 2|\lambda_n| - \frac{2|\lambda_n^{\mathrm{r}}|(|\lambda|^2 + N_0)}{\lambda_n^{\mathrm{g}} + 1} = 0. \quad (5.36)$$

The solution to (5.36) that maximizes (5.35) is obtained for

$$|\lambda_n^{\mathrm{r}}| = \frac{|\lambda_n|(\lambda_n^{\mathrm{g}} + 1)}{|\lambda_n|^2 + N_0}, \quad (5.37)$$

which is the standard MMSE filter, compensated by the receiver trellis processing represented by $\{\lambda_n^{\mathrm{g}}\}$. Inserting (5.37) back into (5.35) gives

$$I_{\mathrm{LB}} = \lim_{N\to\infty} \frac{1}{N} \sum_n \log(\lambda_n^{\mathrm{g}} + 1) - \lambda_n^{\mathrm{g}} + |\lambda_n|^2 \frac{\lambda_n^{\mathrm{g}} + 1}{|\lambda_n|^2 + N_0}. \quad (5.38)$$

Asymptotically as $N \to \infty$, Szegö's Theorem [140] guarantees that I_{LB} converges to

$$\begin{aligned} I_{\mathrm{LB}} &= \frac{1}{2\pi} \int_{-\pi}^{\pi} \log(G^{\mathrm{r}}(\omega) + 1) - G^{\mathrm{r}}(\omega) + |H(\omega)|^2 \frac{G^{\mathrm{r}}(\omega) + 1}{|H(\omega)|^2 + N_0} \, \mathrm{d}\omega \\ &= \frac{1}{2\pi} \int_{-\pi}^{\pi} \log(G^{\mathrm{r}}(\omega) + 1) + \frac{|H(\omega)|^2 - N_0 \, G^{\mathrm{r}}(\omega)}{|H(\omega)|^2 + N_0} \, \mathrm{d}\omega. \end{aligned} \quad (5.39)$$

which concludes the proof. □

In order to derive the optimal $G^{\mathrm{r}}(\omega)$ we can make use of Proposition 2 directly. The matrix $\boldsymbol{B}$ is a Toeplitz matrix that is characterized through the transform (easiest seen from the expression in the proof of Proposition 2)

$$B(\omega) = \frac{N_0}{|H(\omega)|^2 + N_0}.$$

As $N \to \infty$, the matrix $\tilde{\boldsymbol{B}}_n^{\nu}$ is the same for all subindices n and the dimension is always $\nu \times \nu$. The vector $\boldsymbol{b}_n^{\nu}$ is always a $1 \times \nu$ vector and is the same for all n. The elements of $\tilde{\boldsymbol{B}}^{\nu}$ and $\boldsymbol{b}^{\nu}$, where the subindex n has been left out since it is irrelevant for ISI channels, are formed from

$$\int B(\omega) \exp(i\omega k) \mathrm{d}\omega, \quad |k| \leq \nu.$$

The achievable information rate becomes

$$I_{\mathrm{LB}} = \log\left(\frac{1}{c}\right),$$

where

$$c = \int B(\omega)\mathrm{d}\omega - \boldsymbol{b}^{\nu}(\tilde{\boldsymbol{B}}^{\nu})^{-1}(\boldsymbol{b}^{\nu})^{\dagger}.$$

5.3 Numerical Results on Achievable Information Rates

5.3.1 ISI Channels

In this section numerical results for the achievable information rates of the proposed channel shortening detector are presented. First consider the EPR4 channel from Chapter 4

$$\boldsymbol{h} = [1,\ 1,\ -1,\ -1]\,/2 \tag{5.40}$$

with 2-PAM inputs in Figure 5.5. According to Chapter 4, the EPR4 channel has $d^2_{\mathrm{min}} = 2$, i.e., it corresponds to a relatively mild ISI. The information rates I_{LB}, in bits per channel use, for the mutual information optimized detector

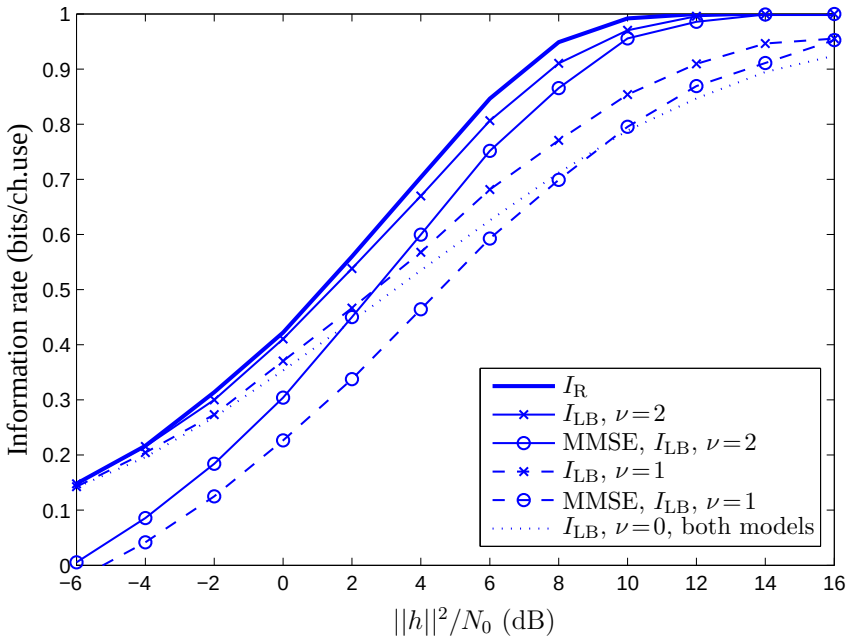

Figure 5.5: Achievable rates of the EPR4 ISI channel with $\nu = 0, \ldots, 3$ for the mutual information and the MMSE optimized detectors. The legend shows the curves from top to bottom at the right hand side of the figure.

as well as the MMSE optimized detector from [123] are plotted. The legend shows the curves from top to bottom at the right hand side of the figure. The top bold line shows I_{R}, the information rate corresponding to a full complexity detector for $\boldsymbol{h}$, i.e., a detector with $\nu = 3$. The two solid lines marked with x-es show I_{LB} for the mutual information optimized detector with $\nu = 1$ and 2 respectively. The two solid lines show the same curves but for an MMSE optimized detector according to [123]; the mismatched noise density was in this case set to the MMSE value. The dotted line corresponds to $\nu = 0$ for both the mutual information rate and the MMSE optimized detectors. Note that with the MMSE cost function in (5.9), $\nu = 0$ yields higher I_{LB} than $\nu = 2$ and $\nu = 3$ in the low SNR regime. The reason is that the target ISI responses for $\nu = 1$ and 2 are very weak in terms of mutual information. The MMSE values are monotonically decreasing with increasing ν since the domain of the optimization is larger. Further, with $\nu = 3$, the MMSE optimized detector does

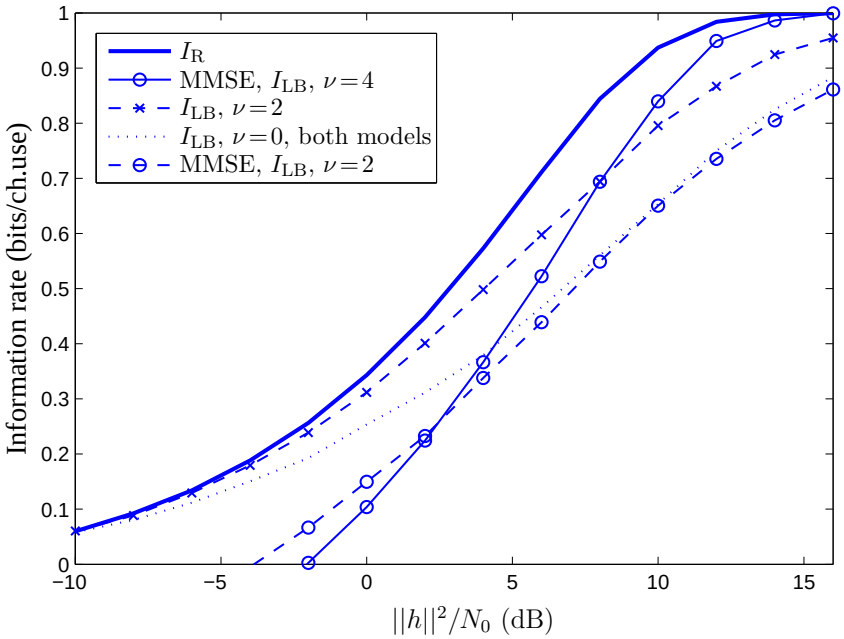

Figure 5.6: Achievable rates of the 5-tap uniform power ISI channel in (5.41) with $\nu = 0, 2$ and 4 for the mutual information and the MMSE optimized detectors. The legend shows the curves from top to bottom at the right hand side of the figure.

not converge to the full complexity detector, thus there will be a gap to the full complexity curve even for $\nu = 3$. The gaps between the MMSE optimized detectors and the mutual information optimized detectors are largest in the low SNR regime.

Next we study the 5-tap uniform power ISI channel

$$\boldsymbol{h} = [1,\ 1,\ 1,\ 1,\ 1]/\sqrt{5} \tag{5.41}$$

with 2-PAM inputs in Figure 5.6. The ISI channel in (5.41) has $d^2_{\mathrm{min}} = 0.8$ which is a 3.98 dB loss compared to orthogonal 2-PAM signaling. The information rates I_{LB}, in bits per channel use, for the mutual information optimized detector as well as the MMSE optimized detector from [123] are plotted. The legend shows the curves from top to bottom at the right hand side of the figure. The top bold line shows the information rate I_{R} corresponding to a full com-

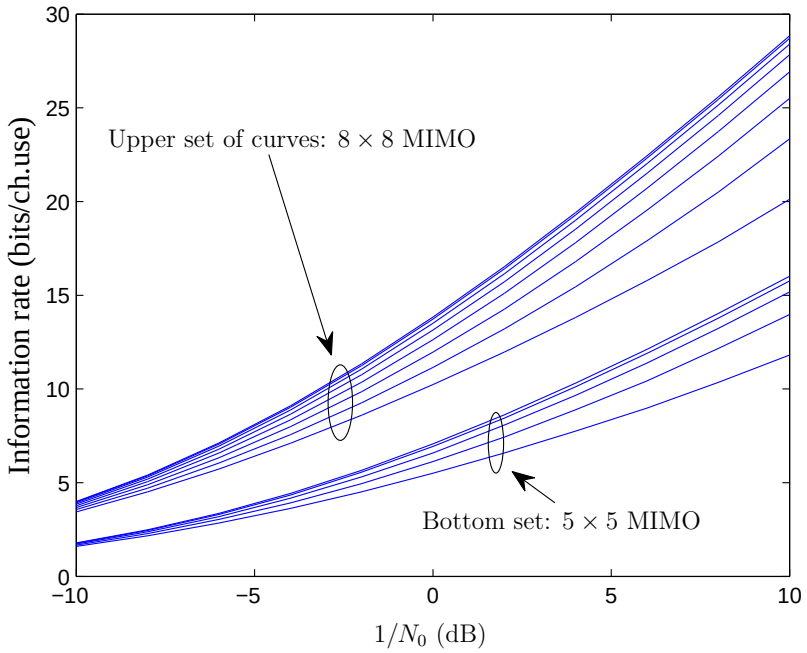

Figure 5.7: Information rates with Gaussian inputs for 5×5 and 8×8 MIMO. Within each set of curves, the upper curve shows the information rate achieved by a full complexity detector, the bottom curve shows the ensuing information rate from an MMSE detector, and the intermediate curves show achievable information rates for the reduced detector with $\nu = 1, 2, 3 \ldots, N-2$. (Note that $\nu = N-1$ corresponds to full complexity.)

plexity detector, i.e., $\nu = 4$. In order to illuminate the suboptimal performance in terms of mutual information of conventional channel shortening based on MMSE optimizations, I_{LB} for $\nu = 4$ with the method from [123] is plotted; this curve is the uppermost thin solid line. According to the figure, there is a 10 dB gap to I_{R} at low SNR. Clearly, such detector is of no practical value, but it highlights the fact that MMSE cost functions do not yield good mutual information performance. The solid line marked with x-es shows I_{LB} for the mutual information optimized detector with $\nu = 2$ while the corresponding curve for the MMSE method from [123] is the bottom solid line (right hand side of the figure). With $\nu = 0$, the achievable information rate is the same for

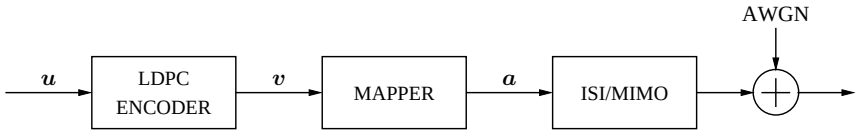

Figure 5.8: A system model of the transmitter. After LDPC encoding and mapping, the information carrying signal is formed by (5.1) and transmitted over the AWGN channel.

both methods, which is shown by the dotted curve. Again, this curve outperforms both $\nu = 2$ and 4 for MMSE optimizations, since weak ISI responses are obtained from a mutual information point of view.

5.3.2 MIMO Channels

Consider now 5×5 and 8×8 MIMO channels with independent and identically distributed complex Gaussian entries $\{h_{i,j}\}$. Figure 5.7 plots the achievable information rates, in *bits* per channel use, with Gaussian inputs and against a measure of the SNR which we take as $1/N_0$. The bottom curve within each set of curves is the information rate corresponding to an MMSE detector while the upper curve is the information rate I_{R} corresponding to a full complexity detector. The intermediate curves show information rates for memory $\nu = 1, 2, 3, \ldots$. Importantly, it can be seen that there is a significant gain when going from $\nu = 0$ (MMSE detector) to $\nu = 1$. In fact, $\nu = 1$ achieves a considerable share of the full complexity information rate. In the MIMO case, the channel matrix $\boldsymbol{H}$ has been permuted prior to optimization of $\boldsymbol{H}^{\mathrm{r}}$ and $\boldsymbol{G}^{\mathrm{r}}$. The permutation has been made by simply rearranging the columns in an increasing order with respect to the energy of the columns. This will in general benefit the channel shortening detector since the elements around the main diagonal of $\boldsymbol{G}^{\mathrm{r}}$ will have larger absolute values. Other, more advanced permutations have also been tested, but virtually no improvements over the energy-permutation were observed.

5.4 Practical Coded Modulation Systems

In this section receiver tests of LDPC encoded transmission systems over ISI and MIMO channels are performed. The system model of the transmitter is shown in Figure 5.8. A sequence $\boldsymbol{u}$ of uncoded information bits is encoded with an LDPC code generating $\boldsymbol{v}$. A mapper takes the encoded sequence as input

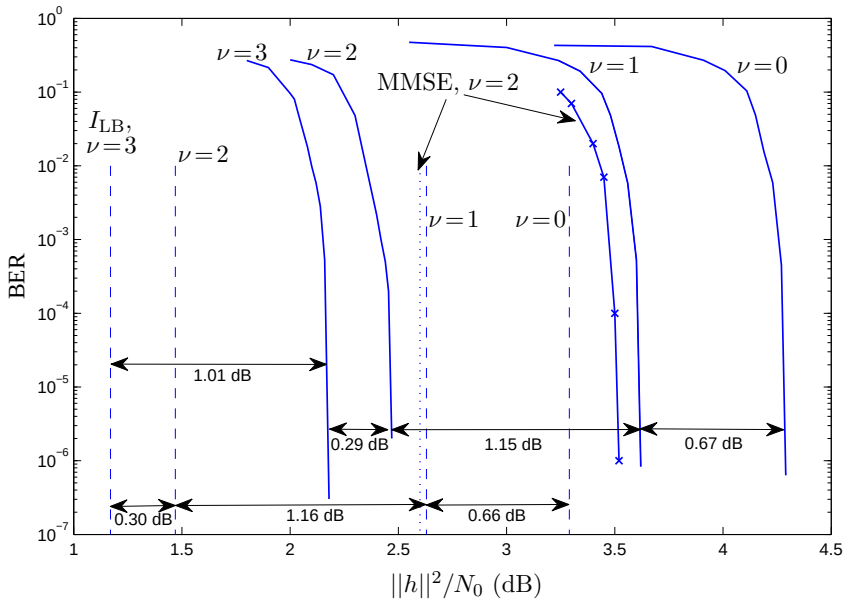

Figure 5.9: Receiver tests of LDPC encoded transmissions over the EPR4 channel with 2-PAM inputs. The LDPC code is the irregular rate 1/2 (32400,64800) standardized code in DVB-S.2. The vertical dashed lines mark the achievable information rates I_{LB} for different values of receiver complexity ν while the solid lines show the actual BERs. The dotted vertical line and the line marked with x-es show the performance of a detector optimized according to [123].

and outputs a sequence of symbols from Ω which are then transmitted over the linear channel in (5.1). The AWGN channel follows before the transmitted signal is detected at the receiver.

An iterative scheme, employing the channel shortening detector for soft-input soft-output detection of the channel and belief propagation for the LDPC code, is adopted. The particular LDPC code used is the irregular rate 1/2 (32400,64800) code from the Digital Video Broadcasting standard (DVB-S.2) [117].[4] In all setups, 50 internal iterations were performed within the LDPC decoder and 4 global iterations within the iterative loop. The ISI channel used in the tests is the EPR4 channel in (5.40). The entries $\{h_{i,j}\}$ of the

[4]This is the default code in Matlab's LDPC package.

channel matrix $\boldsymbol{H}$ representing a MIMO channel are assumed to be IID complex Gaussian random variables.

Tests of the detector with $\nu = 0, 1, 2$ and 3 are performed. Note that $\nu = 0$ corresponds to an MMSE detector while $\nu = 3$ is full complexity. The results are shown by the solid curves in Figure 5.9. The four vertical dashed lines mark the ultimate limit for rate 1/2 encoded systems with 2-PAM inputs for the different values of ν. This limit is the needed $\|\boldsymbol{h}\|^2/N_0$ to obtain $I_{\mathrm{LB}} = 1/2$. As a benchmark comparison, also plotted are the BER and information rate performance of the conventional channel shortening technique from [123] with $\nu = 2$; the BER performance is marked with x-es while the information rate is shown by a dotted line. According to the figure, all BER curves are about 1 dB away from their ultimate limits. Further, the rate I_{LB} is closely related to the BER performance since the gap in I_{LB} between two different values of ν corresponds very well to the gap between the corresponding BER curves. As an example, the gap in I_{LB} between $\nu = 2$ and 3 is .3 dB while the gap between the corresponding BER curves is .29 dB.

According to Figure 5.9 the method from [123], optimal with respect to a certain MMSE criteria, performs more than 1 dB worse than the proposed method in this chapter. As a conclusion, by optimizing the achievable information rate of the detector, modern transmission systems which employ powerful codes can operate closer to the ultimate Shannon limit of the underlying channel when only limited trellis processing can be afforded.

Consider next 4×4 MIMO channels with QPSK inputs and the same LDPC code. The average energy of the QPSK symbols is 2. One codeword $\boldsymbol{v}$ corresponds to 64800/8=8100 MIMO input vectors $\boldsymbol{a}$. A rapid fading case (different channel realization for each channel use) is assumed where each of these 8100 channel matrices is independently drawn and comprises independent and identically distributed circularly symmetric complex Gaussian random variables with zero mean and unit variance. The BER performance is shown in Figure 5.10. In all cases, 50 internal iterations within the LDPC decoder and 4 global ones were carried out. The receiver is tested with $\nu = 0$ (MMSE) in which case a single global iteration is sufficient, $\nu = 1, 2$ and $\nu = 3$ (full complexity). The vertical dashed lines mark the ergodic[5] ultimate limit for systems with QPSK inputs and a rate 1/2 outer code and it correspond to $\mathbb{E}[I_{\mathrm{LB}}] = 1$. Depending on the memory ν, the BER curves lie $0.85 - 1.15$ dB away from the ultimate limits.

[5]The ergodic information rate is the mean rate averaged over the channel realizations.

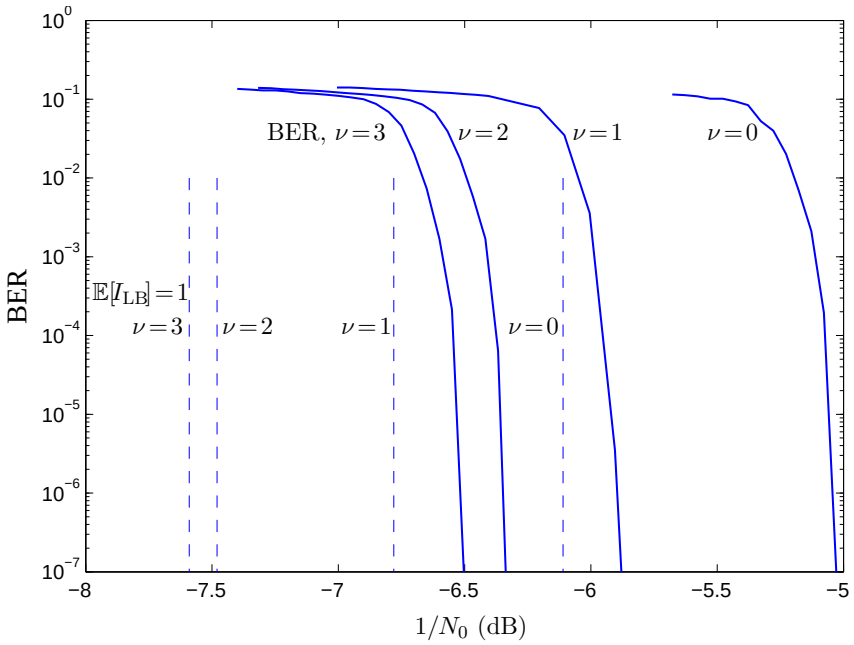

Figure 5.10: Receiver tests of LDPC encoded transmissions over a rapid fading 4×4 MIMO channel with QPSK inputs. The LDPC code is the irregular (32400,64800) standardized code in DVB-S.2. The vertical dashed lines mark the ergodic achievable information rates $\mathbb{E}[I_{\mathrm{LB}}]$ for different values of receiver complexity ν while the solid lines show the actual BERs. Each BER curve lies .85 - 1.15 dB away from its corresponding information rate threshold.

5.5 Results on FTN

This section presents a few examples of practical coded FTN systems that employ the mutual information optimized channel shortening detector. At the transmitter, a sequence of 4000 information bits is encoded using the rate 1/2 (7,5) convolutional code. The encoded sequence feeds a size 8000 random interleaver whose output is mapped to symbols from the 2-PAM alphabet. The transmitted signal is formed using (5.1) where $\boldsymbol{H}$ takes the form in (2.130). This transmitter setup and the corresponding iterative receiver structure are illustrated in Figure 5.11.

For the ISI channel both strong and extreme ISI are investigated. This

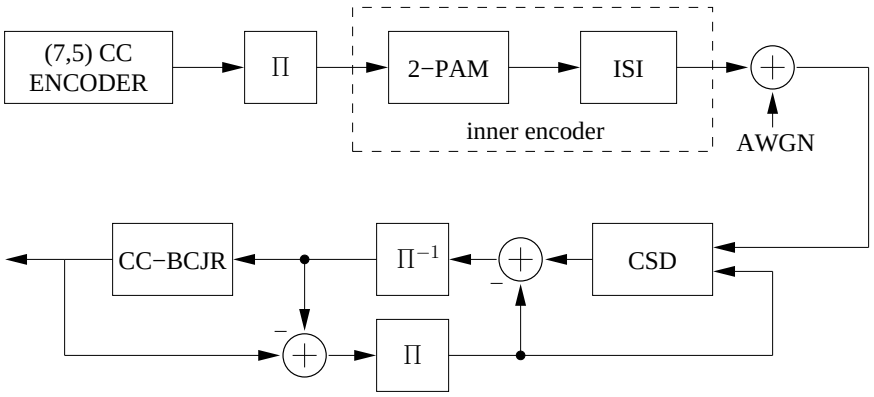

Figure 5.11: A system model of the transmitter and the corresponding iterative receiver structure. In the tests, the (7,5) convolutional code is used for encoding. The symbols are drawn from a 2-PAM alphabet before being transmitted over the ISI channel.

is achieved by using the FTN framework which provides a structured way of measuring the "severeness" of the ISI channel; $\tau = .9$ is not as difficult to detect as $\tau = .5$ etc. In the tests $h(t)$ is taken as a 30% root raised cosine pulse while $\tau = 0.35$ and $\tau = 1/4$. The framework of super minimum phase has been used, resulting in the 17 tap long ISI sequence from Chapter 3, which is

$$\begin{aligned} \boldsymbol{h} &= [.025,.012,-.024,.008,.\mathbf{191},.\mathbf{464},.\mathbf{623},.\mathbf{506},.\mathbf{176},-.\mathbf{123}, \\ &\quad -.\mathbf{196},-.\mathbf{075},.\mathbf{060},.\mathbf{080},.\mathbf{013},-.\mathbf{035},-.\mathbf{022}] \end{aligned} \tag{5.42}$$

for $\tau = 0.35$ while the case $\tau = 1/4$ yields

$$\begin{aligned} \boldsymbol{h} &= [-.010,-.013,-.007,.005,.011,.004,-.008,.001,.\mathbf{060}, \\ &\quad .\mathbf{181},.\mathbf{339},.\mathbf{473},.\mathbf{520},.\mathbf{443},.\mathbf{262},.\mathbf{047},-.\mathbf{120},-.\mathbf{182},-.\mathbf{138}, \\ &\quad -.\mathbf{037},.\mathbf{055},.\mathbf{092},.\mathbf{070},.\mathbf{018},-.\mathbf{025},-.\mathbf{037},-.\mathbf{021},.\mathbf{003}, \\ &\quad .\mathbf{016},.\mathbf{012},.\mathbf{0004},-.\mathbf{008}]. \end{aligned}$$

Let us begin with an example that shows how the proposed channel shortening detector processes the received noisy signal $\boldsymbol{y}$. The assumptions are the $\tau = 0.35$ FTN channel in (5.42), $N_0/2$ is set to 1 and the memory constraint in (5.7) is $\nu = 2$. Figure 5.12 plots the true autocorrelation sequence $\boldsymbol{g}$ (obtained

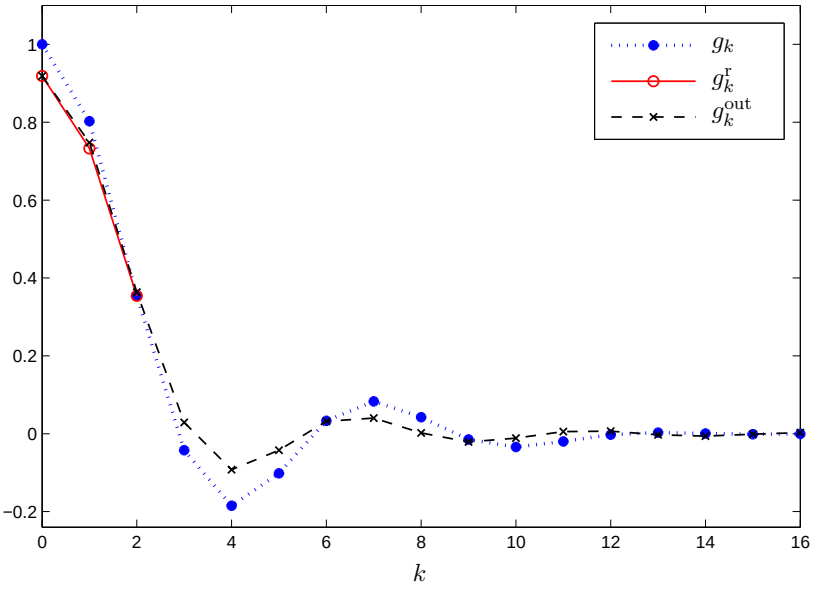

Figure 5.12: Examples of Ungerboeck ISI model taps when employing the mutual information optimized front-end filter $\boldsymbol{h}^{\mathrm{r}}$ and a standard matched filter. The figure shows ISI responses for the $\tau = 0.35$ FTN channel with $N_0/2 = 1$ and $\nu = 2$.

when the standard matched filter is employed as the receiver front-end filter) as a dotted line, the filtered response $\boldsymbol{g}^{\mathrm{out}} = \boldsymbol{h}^{\mathrm{r}} \star \boldsymbol{h}$ as a dashed line and the shortened discrete-time ISI channel model $\boldsymbol{g}^{\mathrm{r}}$ as a solid line. Note that Figure 5.12 only shows half of the ISI taps, i.e., $[g_0, g_1, \ldots]$ and similarly for $\boldsymbol{g}^{\mathrm{out}}$ and $\boldsymbol{g}^{\mathrm{r}}$. By comparing the filtered response $\boldsymbol{g}_{\mathrm{out}}$ with the sequence $\boldsymbol{g}$ we observe that the outer ISI taps in $\boldsymbol{g}_{\mathrm{out}}$ which are not accounted for by the channel shortening detector are much smaller than the corresponding taps in $\boldsymbol{g}$.

Figure 5.13 shows the error probability of the underlying (7,5) convolutional code along with turbo equalization results for the (7,5)-encoded $\tau = 0.35$ ISI channel. Throughout this section, we have used 20 iterations in the iterative loop. Convergence is usually reached much earlier but we do not report any numerical results on the mean number of iterations needed to reach convergence. The outer decoder (see Figure 5.11) in all the performed tests is implemented as a BCJR (4 states). The inner decoder of the turbo equalization is

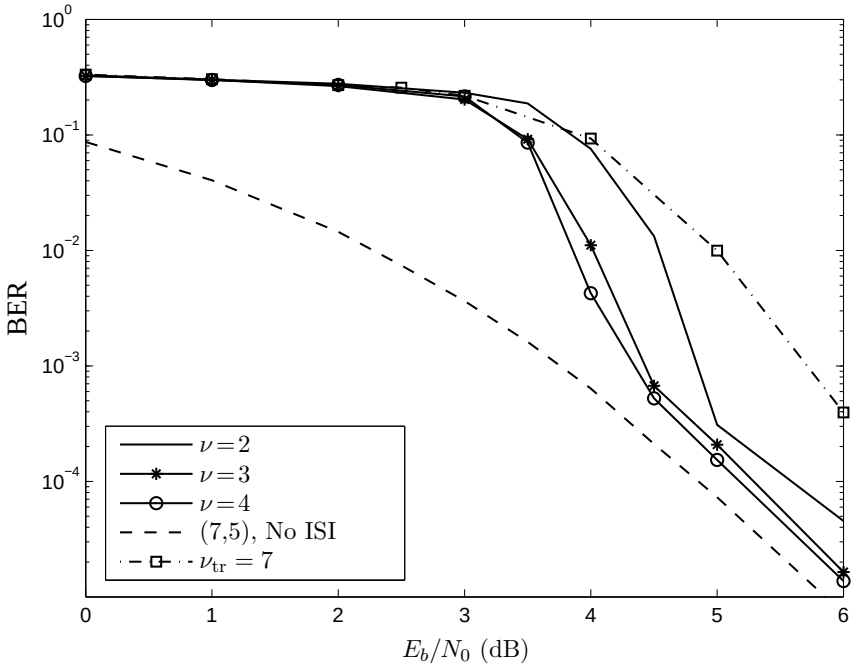

Figure 5.13: Turbo equalization results for $\tau = 0.35$ when using the the proposed channel shortening detector as inner decoder. Results are shown for memories $\nu = 2, 3, 4$. The left-most curve shows the performance of the (7,5) code on an ISI-free channel.

the proposed channel shortening detector, also implemented as a BCJR, with memories $\nu = 2, 3, 4$, i.e. 4, 8, and 16 trellis states (as opposed to 65536 states of a BCJR that operates on the full channel model). The curve marked with $\nu_{\mathrm{tr}} = 7$ corresponds to a receiver in which the BCJR-based inner decoder approximates $\boldsymbol{h}$ with the ν_{tr} strongest taps (ignoring precursors) and regards the neglected ISI taps as Gaussian noise. This inner decoder is commonly referred to as a *truncated BCJR*. Hence, the curve corresponds to $2^7 = 128$ states.

As can be seen, the proposed detector converges to the ISI-free channel performance rapidly while the truncated detector is strongly inferior to the proposed channel shortening detector. However, at high SNRs there are gaps to the (7,5) code. With $\nu = 2$, i.e. 4 states, the gap is around 1 dB at high SNR. These gaps are the results of the capacity-optimization instead of a BER-optimization. The problem is that the receive filter $\boldsymbol{h}^{\mathrm{r}}$ does not maximize the

SNR. This SNR-loss can be clearly seen in Figure 5.12; it is the gap between g_0 and g_0^{out} and it is around 0.45 dB for the presented case.

In order to maximize the SNR, a matched filter should be used as the receiver front-end filter, i.e., $\boldsymbol{h}^{\text{r}} = \boldsymbol{h}[-k]$ where $\boldsymbol{h}[-k]$ denotes the time reversed version of $\boldsymbol{h}$. However, a matched filter is far from capacity optimal for small values of ν, which implies that convergence is not obtained in the early iterations of the turbo equalization. This problem can be resolved by shifting from the capacity optimal front-end filter into a matched filter in the final iteration. Note that the length of $\boldsymbol{g}$, the autocorrelation of $\boldsymbol{h}$, is larger than allowed by the memory constraint in (5.7). Therefore, the Ungerboeck based inner decoder uses only the middle $2\nu+1$ taps (memory ν) of $\boldsymbol{g}$, denoted $\boldsymbol{g}_{\text{trunc}}$, when calculating the branch labels in the last iteration. In addition to the SNR increase in the final iteration, soft-interference cancellation is also employed. The residual ISI $\boldsymbol{g}_{\text{res}}$ is obtained as

$$\boldsymbol{g}_{\text{res}} = \boldsymbol{g} - \boldsymbol{g}_{\text{trunc}}. \tag{5.43}$$

Soft estimates of the 2-PAM symbols are used in the cancellation process. Since the probability of $a_k = +1$ at trellis depth k is given by

$$\Pr(a_k = +1) = \frac{e^{L(a_k)}}{1 - e^{L(a_k)}} \tag{5.44}$$

where $L(a_k)$ is the log likelihood ratio in (2.133), the soft symbol estimates are taken as the expected values

$$\hat{a}_k = \mathbb{E}[a_k] = 2\Pr(a_k = +1) - 1. \tag{5.45}$$

The input to the component ISI decoder in the last iteration are the apriori extrinsic L-values from previous iteration and the sequence

$$\tilde{\boldsymbol{x}} = \boldsymbol{x} - \hat{\boldsymbol{a}} \star \boldsymbol{g}_{\text{res}} \tag{5.46}$$

where $\boldsymbol{x} = \boldsymbol{h}^{\text{r}} \star \boldsymbol{y} = \boldsymbol{h}[-k] \star \boldsymbol{y}$ is the filtered channel output in Figure 5.1. This modification of the detector yields the results shown in Figure 5.14. According to the figure, even the 4-state ($\nu = 2$) trellis decoder is now powerful enough so that it provides the (7,5) code performance already at $E_b/N_0 = 5$ dB. A comparison to the performance of the backup M-BCJR from Chapter 3 can be

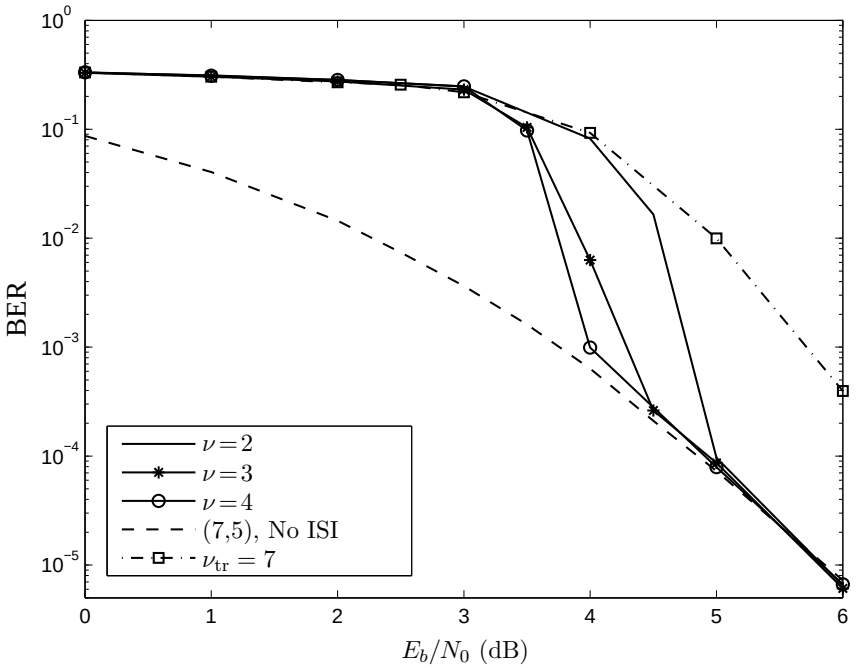

Figure 5.14: Turbo equalization results for $\tau = 0.35$ when using the modified detector as inner decoder and with a matched filter in the last turbo iteration. Results are shown for memories $\nu = 2, 3, 4$. The left-most curve shows the performance of the (7,5) code on an ISI-free channel.

made. Figure 3.14 shows that the backup M-BCJR reaches the (7,5) code performance at $E_b/N_0 \approx 6$ dB with $M = 8$ for the same FTN ISI channel. These results indicate that, if properly modified, the proposed channel shortening detector, optimized for Gaussian inputs, can have impressive performance for discrete symbol alphabets.

The mutual information optimized $\boldsymbol{g}^{\mathrm{r}}$ in the simulations are usually not valid autocorrelation sequences, i.e., the corresponding matrices $\boldsymbol{G}^{\mathrm{r}}$ are not positive semi-definite. This implies that the conventional shortening algorithms [123]–[131], which are all constrained to operate with a positive definite $\boldsymbol{G}^{\mathrm{r}}$ in (5.6), are in fact bounded away from the optimal solution by definition. As an example, for $\tau = 0.35$, $\nu = 1$, and $N_0/2 = 1$, the optimal vector $\boldsymbol{g}^{\mathrm{r}}$ is

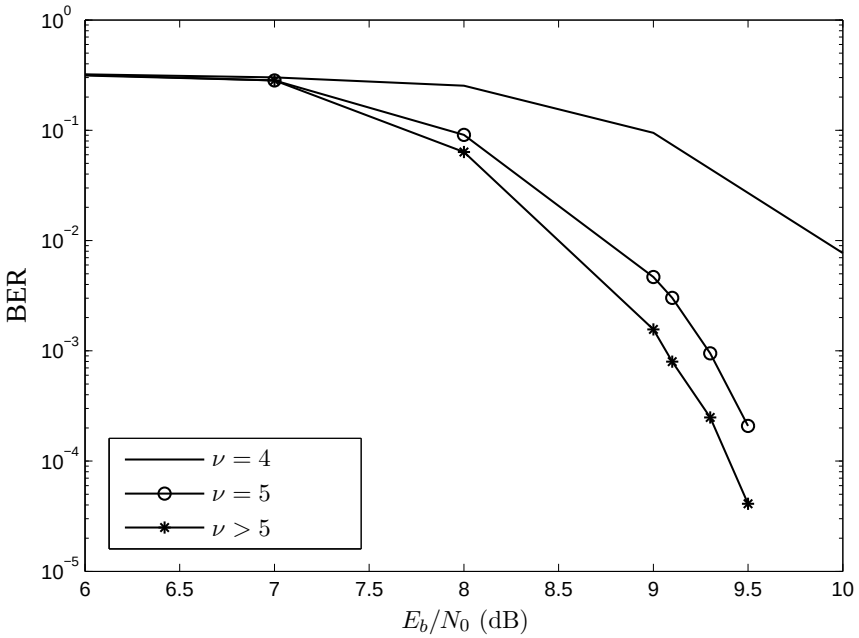

Figure 5.15: Turbo equalization results for $\tau = 1/4$ when using the modified low-complexity decoder as inner decoder and with a matched filter in the last iteration. Results are shown for memories $\nu = 4, 5$ and $\nu = 6, 7, 8, 9, 10$.

$$\boldsymbol{g}^{\mathrm{r}} = [0.4691, \quad 0.68320, \quad 0.4691]$$

which does not have a strictly positive Fourier transform. Note that $\boldsymbol{g}^{\mathrm{r}}$ is the shortened discrete-time ISI channel model used by the Ungerboeck-based BCJR to calculate the branch labels in (4.15). The channel observations are obtained by filtering the received signal with the mutual information optimal front-end filter $\boldsymbol{h}^{\mathrm{r}}$, provided by the channel shortening detector.

In Figure 5.15 turbo equalization results for the case $\tau = 1/4$ are shown. In the last turbo iteration, the matched filter has been used as the receiver front-end filter. Its sampled outputs are fed to the inner decoder as channel observations. In order to satisfy the memory constraint, the inner decoder uses only the middle $2\nu + 1$ taps of the true autocorrelation $\boldsymbol{g}$ when calculating the branch labels in the last iteration. The performance of the channel shortening

detector is virtually the same for $\nu = 6, 7, 8, 9, 10$. Hence the curve marked by asterixes appears to be the ultimate limit of an iterative detector (i.e., the curve that would result from a MAP component decoder). According to the figure, $\nu = 5$ (32 states) results in about .2 dB loss, while $\nu = 4$ (16 states) shows a significant loss. The corresponding turbo equalization results for the backup M-BCJR algorithm and the $\tau = 1/4$ FTN ISI channel are shown in Figure 3.16. The figure shows that for this extreme ISI there are no major differences in BER performance between the two methods. However for small ν and M with comparable complexity, the mutual information optimized detector performs slightly better.

5.6 Conclusions

In this chapter channel shortening detectors for linear channels are optimized from an information theoretical perspective. Gaussian inputs are assumed, and the optimal front-end filter and branch labels of the trellis processing can be given in closed form. The framework used in this chapter is more general than what has been previously used within the area. Practical coded modulation systems based on LDPC codes were tested and it was shown that the resulting BER performance is connected to the achievable information rate of the detector. If the detector is to be used within an iterative detection loop (for FTN), it must be slightly modified so that the SNR loss of the front-end filter is eliminated.

Chapter 6

Summary and Future Work

In this thesis a somewhat unconventional signaling method is considered. Intersymbol interference is intentionally introduced by using a signaling rate which is faster than that allowed by the Nyquist orthogonality criterion. Although Shannon showed in 1949 that the capacity can be achieved using orthogonal ISI-free sinc pulses and long symbol sequences, this memoryless assumption in the modulator can in practical applications lead to significant capacity losses. Faster-than-Nyquist signaling exploits the excess bandwidth of practical T-orthogonal pulses and in that way it can theoretically achieve capacity with discrete symbol alphabets. In FTN the power spectral density is fixed but the bit density in bits/Hz-s is considerably higher than for ordinary orthogonal signaling. FTN methods are important in future satellite and mobile systems, since they are one of the best ways to pack more bits into a given radio bandwidth without increasing the transmission energy.

There is always a tradeoff and in the case of FTN signaling the tradeoff is between increased spectral efficiency and receiver complexity. A main issue in this thesis is how to reduce this complexity while at the same time not degrading the error performance noticeably. Since FTN-induced intersymbol interference can be well-approximated with a finite state machine it thereby admits trellis representation. We have considered several approaches for complexity reduction. The proposed receivers can with practical complexity levels achieve near-optimal performance under severe ISI generated by the higher transmission rate in FTN. These contributions will make this bandwidth efficient signaling method more useful.

In order to better visualize the systems investigated in this thesis, Figure 6.1 plots their transmission rates against their energy efficiency. The basic modulation pulse is the 30% rRC T-orthogonal pulse, and all systems are operating

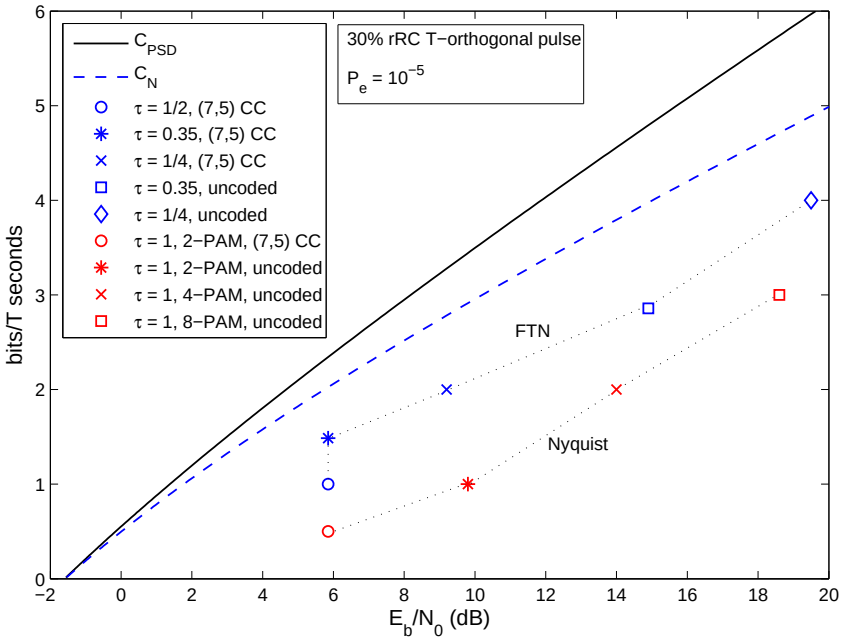

Figure 6.1: Transmission rates in bits/T seconds versus E_b/N_0 in dB of communication systems investigated in this thesis.

at a BER of 10^{-5}. Note that they all have the same PSD shape, namely the 30% RC shape. The plotted transmission rates in Figure 6.1 are given in bits/T seconds. The uppermost curve shows the ultimate PSD limit C_{FTN} as given in (2.121). The second curve from above is the corresponding limit for ordinary orthogonal transmission, i.e., C_{N} from (2.122). We plot the operating points of five different FTN systems: encoded systems using the (7,5) convolutional code with $\tau = 1/2$, 0.35 and 1/4, as well as two uncoded FTN system with $\tau = 0.35$ and 1/4. These operating points can be read off from the receiver tests in Figures 3.13, 3.14, 5.15 and 3.10 in Chapters 3 and 5. For comparison, the figure also shows the operation points of uncoded 2-PAM, 4-PAM, and 8-PAM as well as an encoded 2-PAM system using the (7,5) convolutional code. By comparing the uncoded 2-PAM system with the encoded $\tau = 0.35$ FTN system, we observe that FTN simultaneously offers about 4 dB coding gain and 43% rate increase.

After the introduction, the thesis consists of three main chapters. In the first

main chapter, Chapter 3, several new M-BCJR reduced-search algorithms are proposed and compared to reduced-trellis VA and BCJR benchmarks based on the offset label and other trellis truncation ideas. The proposed M-BCJRs have been applied to simple ISI detection as well as to turbo equalization of coded FTN signals. In a heavily reduced search, there is often no overlap between the decided paths in the forward recursion and the decided paths in the backward recursion for one of the symbols. The decoder is then unable to produce reliable soft information about some of the detected symbols. Additionally, the magnitude of the numerator and the denominator in the log likelihood ratio may erroneously differ too much compared with a full-complexity receiver, resulting in over-estimated LLRs. The sign of the LLR can be determined easily but if iterative detection is to be employed, reliable absolute values are essential. The backup M-BCJR, proposed in Chapter 3, adds a third low-complexity recursion and uses a modified method for retaining backward recursion values, which together considerably improve the LLR quality for practical values of the search size M.

In addition to this, an improvement of the minimum phase idea which concentrates the ISI model energy better than the mathematically correct minimum phase model is proposed. Due to spectral zero regions in practical FTN signaling the standard WMF receiver may be prohibited. However, Chapter 3 gives a solution to this modeling problem which also leads to white noise at the receiver. The proposed modeling, denoted super minimum phase modeling, introduces small precursors. Using a delayed and slightly mismatched receiver which ignores the precursors leads to major BER improvements in turbo equalization at a given complexity. For the offset label based benchmark BCJR algorithm various offset label strategies have been considered. At each forward extension in the benchmark BCJR a decision about which symbol becomes the so-called tentative path must be made. This decision is made using the different offset label strategies and the best performing one, denoted single soft offset BCJR algorithm serves as a benchmark to the proposed M-BCJRs. Since low-quality LLRs affect the stability and convergence of the iterative detector it is also beneficial to scale the extrinsic LLRs exchanged in the turbo loop by a scaling gain $g \leq 1$ before each component decoder. By choosing an appropriate g the convergence to the performance of the underlying code occurs at a considerably lower SNR while the overall complexity remains unaltered.

Future work should consider a number of improvements. A more sophisticated method is needed for scaling the extrinsic LLRs in the turbo loop. Early tests indicate that the value of M should vary with the iterations in order to reduce the overall computation effort. The scaling gains should also vary since a larger M will in general produce high-quality LLRs which do not have a negative effect on the convergence of the iterative scheme. All the results from

Chapter 3 may be extended to other outer codes and larger symbol alphabets.

Chapter 4 investigates the effect of the internal metric calculations on performance of detectors based on the Ungerboeck and the Forney observation models. Coded and uncoded transmission over both ISI and MIMO channels are considered. Optimum detectors based on the two observation models generate the same final output but the internal metric calculations differ. In the Forney observation model, the channel observations only depend on the current and past data symbols while observations in the Ungerboeck model contain contributions from both past and future symbols. Chapter 4 demonstrates that this fundamental difference has a significant effect on suboptimum reduced-complexity techniques based on the M-algorithm. Forney based detection is in general preferable in the high SNR region while the situation is reversed for low SNR values. A simple SNR-aware scheme for choosing the best (in terms of BER) observation model among the two is proposed and evaluated. The gains reported in Chapter 4 come with no additional cost since the detection complexity of the underlying algorithm remains unaltered for a given value of M.

A future direction is to find a more sophisticated, possibly adaptive, method which combines the strength of Forney based detection in the high SNR regime with the strength of Ungerboeck based detection at low SNRs. In Chapter 4 it is also shown that there exist other models, denoted middle models, which operate in between the two standard models. Although it is demonstrated that middle models are the optimal choice for some channel realizations, future work needs to focus on how to efficiently exploit their full potential in suboptimum detection.

The chapter also reflects on the asymptotic behavior of the two standard observation models. Practical receiver test and mutual information results confirm the conclusion predicted by the asymptotic analysis. Genie-aided reduced-trellis detectors were considered in order to better understand the poor performance of Ungerboeck detection in the high SNR regime. Even though one of the genie-aided detectors succeeds in reaching the performance of the Forney-based detector, no method which can exploit the promised gains in practice have been found. Hence future work should focus on this subject. A final area for research is the extension to non-binary alphabets for all setups.

In Chapter 5 channel shortening detectors for general linear channels are considered. All conventional channel shortening detectors are optimized from a minimum mean-square-error perspective which is suboptimum since it does not directly optimize the transmission rate. In contrast to conventional channel shortening detectors, the ones proposed in this thesis are optimized from an information theoretical perspective. By assuming Gaussian inputs, the optimal front-end filter and the corresponding branch labels of Ungerboeck based trellis

processing are derived and given in closed form. Practical tests show that even though the proposed detector is optimized for Gaussian inputs, when employed with simple discrete symbol alphabets it shows excellent performance. Moreover, the framework used in this thesis is more general than what has previously been used within channel shortening. Practical LDPC encoded modulation systems employing the proposed detector are tested and evaluated. In the case of coded FTN signaling, the mutual information optimized detector is unable to reach the performance of the underlying convolutional code. In order to eliminate the SNR loss of the front-end filter a modification of the detector is proposed. The modified detector shows impressive performance over narrowband ISI introduced by FTN signaling.

Future work should consider efficient suboptimal solutions which further reduce the computational effort. A possible direction is the use of the backup M-BCJR from Chapter 3 on the shortened channel model. The mutual information optimal front-end filter could be applied to the noisy channel output and followed by a reduced search using the backup M-BCJR. Other modifications of the shortening detector should be considered, which adaptively (rather than in the final iteration) eliminate the SNR loss of the front-end filter.

References

[1] H. Nyquist, "Certain topics in telegraph transmission theory," *AIEE Transactions*, 617 – 644, 1928.

[2] C. E. Shannon, "A mathematical theory of communications," *Bell System Technical Journal*, vol. 27, pp. 379–429 and 623–656, Jul. and Oct. 1948.

[3] C. E. Shannon, "Communication in the presence of noise," in *Proc. IRE*, vol. 23, pp. 10–21, 1949.

[4] G. D. Forney, Jr.,"Maximum-likelihood sequence estimation of digital sequences in the presence of intersymbol interference," *IEEE Trans. Inform. Theory*, vol. 18, no. 2, pp. 363–378, May 1972.

[5] A. J. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Trans. Inf. Theory*, vol. 13, no. 2, pp. 260–269, Apr. 1967.

[6] J. B. Anderson, *Instrumentable tree encoding of information sources*, M.Sc. Thesis, School of Electrical Engineering, Cornell University, Ithaca, N.Y., Sep, 1969.

[7] C. Berrou, A. Glavieux and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes (1)," in *Proc. IEEE Int. Conf. Commun. (ICC)*, vol. 2, pp. 1064–1070, May 1993.

[8] C. Berrou and A. Glavieux, "Near optimum error correcting coding and decoding: Turbo-codes," *IEEE Trans. Commun.*, vol. 44, no. 10, pp. 1261–1271, Oct. 1996.

[9] J. Hagenauer, "The turbo principle: Tutorial introduction and state of the art," in *Proc. Int. Symp. Turbo Codes*, pp. 1–11, ENST de Bretagne, France, Sep. 1997.

[10] C. Douillard, A. Picart, P. Didier, M. Jezequel, C. Berrou and A. Glavieux, "Iterative correction of intersymbol interference: Turbo equalization," *Eur. Trans. Telecomm.*, vol. 6, no. 5, pp. 507–512, Sept./Oct. 1995.

[11] R. Bahl, J. Cocke, F. Jelinek and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. Inform. Theory*, vol. 20, no. 2, pp. 284–287, March 1974.

[12] J. Hagenauer and P. Hoeher, "A Viterbi algorithm with soft-decision outputs and its applications," in *Proc. IEEE Global Telecomm. Conf. (GLOBECOM)*, vol. 3, pp. 1680–1686, Dallas, Nov. 1989.

[13] V. Franz and J.B. Anderson, "Concatenated decoding with a reduced-search BCJR algorithm," *IEEE J. Sel. Areas Commun.*, vol. 16, pp. 186–195, Feb. 1998.

[14] K. K. V. Wong, *The soft-output M-algorithm and its applications*, Ph.D. thesis, Dept. Electrical and Computer Eng., Queens University, Canada, 2006.

[15] J. Hagenauer and C. Kuhn, "Turbo equalization for channels with high memory using a list-sequential equalizer," in *Proc. Int. Symp. Turbo Codes*, ENST de Bretagne, France, Sept. 2003.

[16] B. M. Hochwald and S. ten Brink, "Achieving near-capacity on a multiple antenna channel," *IEEE Trans. Commun.*, vol. 53, pp. 389–399, Mar. 2003.

[17] J. G. Proakis and M. Salehi, *Digital communications*, 5th ed., McGraw-Hill, NY, 2008.

[18] A. J. Viterbi and J. K. Omura, *Priciples of digital communication and coding*, McGraw-Hill, NY, 1979.

[19] J. B. Anderson, *Digital transmission engineering*, IEEE Press, Piscataway, NJ, 2nd ed., 2005.

[20] J. B. Anderson and A. Svensson, *Coded modulation systems*, Kluwer-Plenum, New York, 2003.

[21] G. Ungerboeck, "Adaptive maximum-likelihood receiver for carrier-modulated data-transmission systems," *IEEE Trans. Commun.*, vol. 22, no. 5, pp. 624–636, May 1974.

[22] A. V. Oppenheim and R. W. Schafer, *Discrete-time signal processing*, Prentice-Hall, Englewood Cliffs, NJ, 1989.

[23] G. J. Foschini, "Performance bound for maximum-likelihood reception of digital data," *IEEE Trans. Inf. Theory*, vol. 21, no. 1, pp. 47–50, Jan. 1975.

[24] D. E. Knuth, *The art of computer programming, vol. 3: Sorting and searching*, Addison-Wesley, Reading, Mass., 1973.

[25] J. B. Anderson and E. Offer, "Reduced-state sequence detection with convolutional codes," *IEEE Trans. Inform. Theory*, vol. 40, no. 5, pp. 965–972, May 1994.

[26] T. Aulin, "Breadth-first maximum likelihood sequence detection: Basics," *IEEE Trans. Commun.*, vol. 47, no. 2, pp. 208–216, Feb. 1999.

[27] F. L. Vermeulen and M. E. Hellman, "Reduced state Viterbi decoding for channels with intersymbol interference," in *Proc. IEEE Int. Conf. Commun. (ICC)*, pp. 37B.1–37B.9, Minneapolis, June 1974.

[28] G. J. Foschini, "A reduced state variant of maximum likelihood sequence detection attaining optimum performance for high signal-to-noise ratios," *IEEE Trans. Information Theory*, vol. 23, no. 5, pp. 605–609, Sept. 1977.

[29] S. J. Simmons, "Breadth-first trellis decoding with adaptive effort," *IEEE Trans. Communs.*, vol. 38, no. 1,, pp. 3–12, Jan. 1990.

[30] A. Duel-Hallen and C. Heegard, "Delayed decision-feedback sequence estimation," *IEEE Trans. Communs.*, vol. 37, pp. 428–436, May 1989.

[31] M. V. Eyuboglu and S. U. Qureshi, "Reduced-state sequence estimation with set partitioning and decision feedback," *IEEE Trans. Communs.*, vol. 36, pp. 13–20, Jan. 1988.

[32] R. Johannesson and K. Sh. Zigangirov, *Fundamentals of convolutional coding*, IEEE Press, Piscataway, NJ, 1999.

[33] J. B. Anderson and S. Mohan, *Source and channel coding*, Kluwer, Boston, MA., 1991.

[34] G. Colavolpe and A. Barbieri, "On MAP symbol detection for ISI channels using the Ungerboeck observation model," *IEEE Commun. Letters*, vol. 9, pp. 720–722, Aug. 2005.

[35] J. E. Mazo, "Faster-than-Nyquist signaling," *Bell Syst. Tech. J.*, vol. 54, pp. 1451–1462, Oct. 1975.

[36] G. J. Foschini, "Contrasting performance of faster binary signaling with QAM," *Bell Laboratories Technical Journal*, vol. 63, pp. 1419–1445, Oct. 1984.

[37] C. K. Wang and L. S. Lee, "Practically realizable digital transmission significantly below the Nyquist bandwidth," in *Proc. IEEE Global Telecomm. Conf. (GLOBECOM)*, pp. 1187–1191, Dec. 1991.

[38] N. Seshadri, *Error performance of trellis modulation codes on channels with severe intersymbol interference*, Ph.D. thesis, Dept. Elec., Comp. and System Eng., Rensselaer Poly. Inst., Troy, NY, Sept. 1986.

[39] F. Rusek and J.B. Anderson, "Multi-stream faster-than-Nyquist signaling," *IEEE Trans. Communs.*, vol. 57, pp. 1329–1340, May 2009.

[40] F. Rusek and J. B. Anderson, "The two dimensional Mazo limit," in *Proc. IEEE Int. Symp. Information Theory*, pp. 970–974, Adelaide, 2005.

[41] F. Rusek and J. B. Anderson, "Successive interference cancellation in multistream faster-than-Nyquist signaling," in *Proc. Intl. Wireless Comm. and Mobile Computing Conf. (IWCMC'06)*, Vancouver, Canada, July 2006.

[42] F. Rusek and J. B. Anderson, "Improving OFDM: Multistream faster-than-Nyquist signaling," in *Proc. 4th Int. Symp. Turbo Codes & Related Topics*, Munich, Germany, 2006.

[43] A. D. Liveris and C. N. Georghiades, "Exploiting faster-than-Nyquist signaling," *IEEE Trans. Communs.*, vol.51, pp. 1502–1511, Sept. 2003.

[44] J. H. Lee and Y. H. Lee, "Design of multiple MMSE subequalizers for faster-than-Nyquist-rate transmission," *IEEE Trans. Communs.*, vol. 52, pp. 1257–1264, Aug. 2004.

[45] A. D. Liveris, *On distributed coding, quantization of channel measurements and faster-than-Nyquist signaling*, Ph.D. thesis, Dept. Elec. Eng., Texas AT&M Univ., April 2006.

[46] F. Rusek and J. B. Anderson, "M-ary coded modulation by Butterworth filtering," in *Proc. IEEE Int. Symp. Information Theory*, pp. 184, Yokohama, Japan, 2003.

[47] J. E. Mazo and H. J. Landau, "On the minimum distance problem for faster-than-Nyquist signaling," *IEEE Trans. Information Theory*, vol. 34, pp. 1420–1427, 1988.

[48] D. Hajela, "On computing the minimum distance for faster-than-Nyquist signaling," *IEEE Trans. Information Theory*, vol. 36, pp. 289–295, 1990.

[49] F. Rusek, *Partial response and faster-than-Nyquist signaling*, Ph.D. thesis, Elec. and Information Tech. Dept., Lund Univ., Lund, Sweden, Sept. 2007.

[50] F. Rusek and J. B. Anderson, "Non-binary and precoded faster-than-Nyquist signaling," *IEEE Trans. Communs.*, vol. 56, pp. 808–817, May 2008.

[51] F. Rusek and J. B. Anderson, "On information rates of faster-than-Nyquist signaling," in *Proc. IEEE Global Telecomm. Conf. (GLOBECOM)*, San Francisco, Ca., 2006.

[52] F. Rusek and J. B. Anderson, "Constrained capacities for faster-than-Nyquist signaling," *IEEE Trans. Information Theory*, vol. 55, pp. 764–775, Feb. 2009.

[53] Y. G. Yoo and J. H. Cho, "Asymptotic optimality of binary faster-than-Nyquist signaling," *IEEE Commun. Letters*, vol. 14, no. 9, pp. 788–790, Sep. 2010.

[54] K. T. Wu and K. Feher, "Multilevel PRS/QPRS above the Nyquist rate," *IEEE Trans. Communs.*, vol. 33, pp. 735–739, July 1985.

[55] F. Rusek and J. B. Anderson, "Serial and parallel concatenations based on faster-than-Nyquist signaling," in *Proc. IEEE Int. Symp. Information Theory*, pp. 970–974, Seattle, WA., July 2006.

[56] F. Rusek, "A first encounter with faster-than-Nyquist signaling over the MIMO channel," in *Proc. IEEE Wireless Comm. Networking Conf. (WCNC)*, Hong Kong, March 2007.

[57] F. Rusek and J. B. Anderson, "Optimal sidelobes under linear and faster-than-Nyquist modulation," in *Proc. IEEE Int. Symp. Information Theory*, pp. 2301–2304, Nice, June 2007.

[58] A. Barbieri, D. Fertonani and G. Colavolpe, "Improving the spectral efficiency of linear modulations through time-frequency packing," in *Proc. IEEE Int. Symp. Information Theory*, pp. 2742–2746, Toronto, Canada, July 2008.

[59] A. Barbieri, D. Fertonani and G. Colavolpe, "Time-frequency packing for linear modulations: Spectral efficiency and practical detection schemes," *IEEE Trans. Communs.*, vol. 57, no. 10,, pp. 2951–2959, Oct. 2009.

[60] G. Colavolpe, T. Foggi, A. Modenini and A. Piemontese,"Faster-than-Nyquist and beyond: How to improve spectral efficiency by accepting interference," *Optics Express*, vol. 19, no. 27, Dec. 2011.

[61] A. Modenini, G. Colavolpe and N. Alagha, "How to significantly improve the spectral efficiency of linear modulations through time-frequency packing and advanced processing," in *Proc. IEEE Int. Conf. Commun. (ICC)*, pp. 3260–3264, June 2012.

[62] I. Kanaras, A. Chorti, M. Rodrigues and I. Darwazeh, "Spectrally efficient FDM signals: Bandwidth gain at the expense of receiver complexity," in *Proc. Intl. Conf. on Commun.*, pp. 1–6, 2009.

[63] M. R. D. Rodrigues and I. Darwazeh, "A spectrally efficient frequency division multiplexing based communication channels," in *Proc. 8th Intl. OFDM Workshop*, Hamburg, 2003.

[64] P. N. Whatmough, M. R. Perrett, S. Isam and I. Darwazeh, "VLSI architecture for a reconfigurable spectrally efficient FDM baseband transmitter," in *Proc. IEEE Int. Symp. Circuits and Syst. (ISCAS)*, pp. 1688–1691, May 2011.

[65] R. G. Clegg, S. Isam, I. Kanaras and I. Darwazeh, "A practical system for improved efficiency in frequency division multiplexed wireless networks," *IET Commun.*, vol. 6, no. 4, pp. 449–457, March 2012.

[66] D. Dasalukunte, *Multicarrier faster-than-Nyquist signaling transceivers*, Ph.D. thesis, Elec. and Information Tech. Dept., Lund Univ., Lund, Sweden.

[67] W. Hirt, *Capacity and information rates of discrete-time channels with memory*, Ph.D thesis, no. ETH 8671, Inst. Signal and Information Processing, Swiss Federal Inst. Technol., Zurich, 1988.

[68] R. A. Gibby and J. W. Smith, "Some extensions of Nyquist's telegraph transmission theory", *Bell System Technical Journal*, vol. 44, no. 2, pp. 1487–1510, Sep. 1965.

[69] S. Shamai, L. H. Ozarow and A. D. Wyner, "Information rates for a discrete-time Gaussian channel with intersymbol interference and stationary inputs," *IEEE Trans. Inf. Theory*, vol. 37, no. 6, pp. 1527–1539, Nov. 1991.

[70] D. Kapetanović, *On linear transmission systems*, Ph.D. thesis, Elec. and Information Tech. Dept., Lund Univ., Lund, Sweden, June 2012.

[71] T. Starr, J. M. Cioffi and P. J. Silverman, *Understanding digital subscriber line technology*, Prentice Hall, Upper Saddle River, NJ 1999.

[72] G. G. Raleigh and J. M. Cioffi, "Spatio-temporal coding for wireless communication," *IEEE Trans. Commun.*, vol. 46, no. 3, pp. 357–366, Mar. 1998.

[73] J. Salz, "Digital transmission over cross-coupled linear channels," *AT&T Technical Journal*, vol. 64, no. 6, pp. 1147–1159, Jul.-Aug. 1985.

[74] G. J. Foschini, "Layered space-time architecture for wireless communication in a fading environment when using multiple antennas," *Bell System Technical Journal*, pp. 41–59, Autumn 1996.

[75] G. Foschini and M. Gans, "On limits of wireless communications in a fading environment when using multiple antennas," *Wireless Personal Communications*, Kluwer Academic Publishers, vol. 6, no. 3, pp. 311–335, 1998.

[76] I. Telatar, "Capacity of multi-antenna Gaussian channels," *European Trans. Tel.*, vol. 10, no. 6, pp. 585–595, Nov.-Dec. 1999.

[77] A. Paulraj, R. Nabar and D. Gore, *Introduction to space-time wireless communications*, Cambridge University Press, Cambridge, UK, 2003.

[78] B. Hassibi and B. M. Hochwald, "How much training is needed in multiple-antenna wireless links?," *IEEE Trans. Inf. Theory*, vol. 49, no. 4, pp. 951–963, Apr. 2003.

[79] 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.

[80] M. Tüchler, A. C. Singer and R. Koetter, "Minimum mean squared error equalization using a priori information," *IEEE Trans. Signal Process.*, vol. 50, no. 3, pp. 673–683, March 2002.

[81] R. Koetter, A. C. Singer and M. Tüchler, "Turbo equalization," *IEEE Signal Processing Magasine*, vol. 21, no. 1, pp. 67–80, 2004.

[82] K. R. Narayanan, "Effect of precoding on the convergence of turbo equalization for partial response channels," *IEEE J. Select. Areas Commun.*, vol. 19, pp. 686–698, April 2001.

[83] M. Tüchler, C. Weis, E. Eleftheriou, A. Dholakia and J. Hagenauer, "Application of high-rate tail-biting codes to generalized pertial response channels," in *Proc. IEEE Global Telecomm. Conf. (GLOBECOM)*, vol. 5, pp. 2965–2971, San Antonio , Nov. 2001.

[84] D. Doan and K. R. Narayanan, "Some new results on the design of codes for inter-symbol interference channels based on convergence of turbo equalization," in *Proc. IEEE Int. Conf. Commun. (ICC)*, pp. 1873–1877, New York, April 2002.

[85] S. ten Brink, "Convergence of iterative decoding," *IEE Electronics Letters*, vol. 35, pp. 806–808, May 1999.

[86] S. ten Brink, "Designing iterative decoding schemes with the extrinsic information transfer chart," *AEÜ Int. Journ. Electron. and Commun.*, vol. 54, pp. 389–398, Nov. 2000.

[87] S. ten Brink, "Convergence behavior of iteratively decoded parallel concatenated codes," *IEEE Trans. Commun.*, vol. 49, pp. 1727–1737, Oct. 2001.

[88] A. Ashikhmin, G. Kramer and S. ten Brink, "Extrinsic information transfer functions: Model and erasure properties," *IEEE Trans. Information Theory*, vol. 50, pp. 2657–2673, Nov. 2004.

[89] J. Hagenauer, "The EXIT chart - introduction to extrinsic information transfer in iterative processing," *EUSIPCO*, vol. 9, pp. 1541–1548, Sep. 2004.

[90] G. Colavolpe, G. Ferrari and R. Raheli, "Extrinsic information in iterative decoding: A unified view," *IEEE Trans. Commun.*, vol. 49, no. 12, pp. 2088–2094, Dec. 2001.

[91] A. Prlja and J. B. Anderson, "Reduced-complexity receivers for strongly narrowband intersymbol interference introduced by faster-than-Nyquist signaling," *IEEE Trans. Commun.*, vol. 60, no. 9, pp. 2591–2601, Sep. 2012.

[92] C. Fragouli, N. Seshadri and W. Turin, "Reduced-trellis equalization using the BCJR algorithm," *Wireless Commun. & Mobile Computing*, vol. 1, pp. 397–406, 2001.

[93] G. Colavolpe, G. Ferrari and R. Raheli, "Reduced-state BCJR-type algorithms," *IEEE J. Sel. Areas Communs.*, vol. 19, pp. 848–859, May 2001.

[94] M. Sikora and D.J. Costello, Jr., "A new SISO algorithm with application to turbo equalization," in *Proc. IEEE Int. Symp. Information Theory*, pp. 2031–2035, Adelaide, Australia, Sept. 2005.

[95] C. M. Vithanage, C. Andrieu and R. J. Piechocki, "Novel reduced-state BCJR algorithms," *IEEE Trans. Communs.*, vol. 55, pp. 1144–1152, June 2007.

[96] D. Fertonani, A. Barbieri and G. Colavolpe, "Reduced-complexity BCJR algorithm for turbo equalization," *IEEE Trans. Communs.*, vol. 55, pp. 2279–2287, Dec. 2007.

[97] D. Fertonani, A. Barbieri and G. Colavolpe, "Novel graph-based algorithms for soft-output detection over dispersive channels," in *Proc. IEEE Global Communs. Conf.*, New Orleans, Dec. 2008.

[98] K. Balachandran and J. B. Anderson, "Reduced complexity sequence detection for nonminimum phase intersymbol interference channels," *IEEE Trans. Inform. Theory*, vol. 43, pp. 275–280, Jan. 1997.

[99] A. Prlja, J. B. Anderson and F. Rusek, "Receivers for faster-than-Nyquist signaling with and without turbo equalization," in *Proc. IEEE Int. Symp. Information Theory*, Toronto, Canada, July 2008.

[100] J. B. Anderson and M. Zeinali, "Best rate 1/2 convolutional codes for turbo equalization with severe ISI," in *Proc. IEEE Int. Symp. Information Theory*, Boston, July 2012.

[101] J. B. Anderson, A. Prlja and F. Rusek, "New reduced state space BCJR algorithms for the ISI channel," in *Proc. IEEE Int. Symp. Information Theory*, Seoul, June 2009.

[102] R. R. Anderson and G. J. Foschini, "The minimum distance for MLSE digital data systems of limited complexity," *IEEE Trans. Information Theory*, vol. 21, pp. 544–551, Sept. 1975.

[103] J. B. Anderson, "Tree encoding of speech," *IEEE Trans. Information Theory*, vol. 21, pp. 379–387, July 1975.

[104] M. Magarini, A. Spalvieri and G. Tartara, "The mean-square delayed decision feedback sequence detector," *IEEE Trans. Communications*, vol. 50, pp. 1462–1470, Sept. 2002.

[105] D. L. Milliner, E. Zimmermann, J. R. Barry and G. Fettweis, "Channel state information based LLR clipping in list MIMO detection," in *Proc. IEEE Int. Symp. Pers. Indoor Mob. Radio Commun. (PIMRC)*, pp. 1–5, 2008.

[106] I. Lee, "The effect of a precoder on serially concatenated coding systems with an ISI channel," *IEEE Trans. Commun.*, vol. 49, pp. 1168–1175, July 2001.

[107] F. Rusek, M. Lončar and A. Prlja, "A comparison of Ungerboeck and Forney models for reduced-complexity ISI equalization," in *Proc. IEEE Global Communications Conference (GLOBECOM)*, Washington DC, Dec. 2007.

[108] E. G. Larsson, "MIMO detection methods: How they work," *IEEE Signal Processing Magazine*, vol. 26, no. 3, pp. 91–95, May 2009.

[109] A. Hafeez and W. E. Stark, "Decision feedback sequence estimation for unwhitened ISI channels with applications to multiuser detection," *IEEE J. Select. Areas Commun.*, vol. 16, no. 9, pp. 1785–1795, Dec. 1998.

[110] C. Fragouli, N. Seshadri and W. Turin, "On the reduced trellis equalization using the M-BCJR algorithm," in *Proc. Conf. Inform. Sciences and Systems*, Princeton University, USA, Mar. 2000.

[111] G. Colavolpe, D. Fertonani and A. Piemontese, "SISO detection over linear channels with linear complexity in the number of interferers," *IEEE J. Select. Topics Sign. Process.*, vol. 5, no. 8, pp. 1475–1485, Dec. 2011.

[112] D. Fertonani, A. Barbieri and G. Colavolpe, "Reduced-complexity BCJR algorithm for turbo equalization," in *Proc. IEEE Int. Conf. Commun. (ICC'06)*, Istanbul, Turkey, June 2006.

[113] C. Studer and H. Bolcskei, "Soft-input soft-output sphere decoding," in *Proc. IEEE Int. Symp. Inform. Theory*, Toronto, Canada, pp. 2007-2011, 2008.

[114] M. Loncar and F. Rusek, "On reduced-complexity equalization based on Ungerboeck and Forney observation models," *IEEE Trans. Sign. Process.*, vol. 56, no. 8, pp. 3784–3789, Aug. 2008.

[115] S. Badri-Hoeher, P. A. Hoeher, H. Chen, C. Krakowski and W. Xu, "Ungerboeck metric versus Forney metric in reduced-state multi-user detectors," in *Proc. 4th Int. Symp. Turbo Codes & Related Topics*, Munich, Germany, 2006.

[116] C. Kuhn and N. Goertz, "A low complexity path metric for tree-based multiple-antenna detectors," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Glasgow, June, 2007.

[117] ETSI EN 302 307 V1.1.2, *Digital Video Broadcasting (DVB); Second generation framing structure, channel coding and modulation systems for broadcasting, interactive services, news gathering and other broadband satellite applications*, June 2006.

[118] A. Kavčić, X. Ma and M. Mitzenmacher, "Binary intersymbol interference channels: Gallager codes, density evolution, and code performance bounds," *IEEE Trans. Inform. Theory*, vol. 49, no. 7, pp. 1636–1652, July 2003.

[119] F. Rusek and A. Prlja, "Optimal channel shortening for MIMO and ISI channels," *IEEE Trans. Wireless Commun.*, vol. 11, no. 2, pp. 810–818, Feb. 2012.

[120] U. Fincke and M. Pohst, "Improved methods for calculating vectors of short length in lattice, including a complexity analysis," *Math. Comput.*, vol. 44, no. 170, pp 463–471, Apr. 1985.

[121] L. G. Barbero and J. S. Thompson, "Fixing the complexity of the sphere decoder for MIMO detection," *IEEE Transactions on Wireless Communications*, vol. 7, no. 6, pp. 2131–2142, June 2008.

[122] F. Rusek and D. Fertonani, "Bounds on the information rate of intersymbol interference channels based on mismatched receivers," *IEEE Trans. Inform. Theory*, vol. 58, no. 3, pp. 1470–1482, March 2012.

[123] D. D. Falconer and F. R. Magee, "Adaptive channel memory truncation for maximum likelihood sequence estimation," *The Bell System Technical Journal*, vol. 52, no. 9, pp. 1541–1562, Nov. 1973.

[124] S. A. Fredricsson, "Joint optimization of transmitter and receiver filter in digital PAM systems with a Viterbi detector," *IEEE Trans. Inform. Theory*, vol. IT-22, no. 2, pp. 200–210, March 1976.

[125] C. T. Beare, "The choice of the desired impulse response in combined linear-Viterbi algorithm equalizers," *IEEE Trans. Commun.*, vol. 26, pp. 1301–1307, 1978.

[126] N. Sundstrom, O. Edfors, P. Ödling, H. Eriksson, T. Koski and P. O. Börjesson, "Combined linear-Viterbi equalizers - a comparative study and a minimax design," in *Proc. IEEE Vehicular Technology Conference (VTC)*, pp. 1263–1267 vol. 2, Stockholm, Sweden, June 1994.

[127] N. Al-Dhahir and J. M. Cioffi, "Efficiently computed reduced-parameter input-aided MMSE equalizers for ML detection: A unified approach," *IEEE Trans. Inform. Theory*, vol. 42, pp. 903–915, April 1996.

[128] M. A. Lagunas, A. I. Perez-Neia and J. Vidal, "Joint beamforming and Viterbi equalizer in wireless communications," in *Proc. Thirty-First Asilomar Conference on Signals, Systems & Computers*, pp. 915–919, vol. 1, Pacific Grove, Ca., Nov. 1997.

[129] S. A. Aldosari, S. A. Alshebeili and A. M. Al-Sanie, "A new MSE approach for combined linear-Viterbi equalizers," in *Proc. IEEE Vehicular Technology Conference (VTC)*, pp. 1707–1711, vol. 3, Tokyo, Japan, May 2000.

[130] R. Venkataramani and M. F. Erden, "A posteriori equivalence: A new perspective for design of optimal channel shortening equalizers," *arXiv:0710.3802v1*.

[131] A. Shaheem, *Iterative detection for wireless communications*, Ph.D. thesis, School of Electrical, Electronic and Computer Engineering, University of Western Australia, 2008.

[132] U. L. Dang, W. H. Gerstacker and S. T. M. Slock, "Maximum SINR prefiltering for reduced-state trellis-based equalization," *IEEE Int. Conf. Commun. (ICC)*, Kyoto, June 2011.

[133] N. Merhav, G. Kaplan, A. Lapidoth and S. Shamai, "On information rates for mismatched decoders," *IEEE Trans. Inform. Theory*, Nov. 1994.

[134] A. Ganti, A. Lapidoth and I. E. Telatar, "Mismatched decoding revisited: General alphabets, channels with memory, and the wide-band limit," *IEEE Trans. Inform. Theory*, vol. 46, no. 7, pp. 2315–2328, Nov. 2000.

[135] J. Boutros, N. Gressety, L. Brunel and M. Fossorier, "Soft-input soft-output lattice sphere decoder for linear channels," in *Proc. IEEE Global Telecomm. Conf. (GLOBECOM)*, San Francisco, Dec. 2003.

[136] D. M. Arnold, H. A. Loeliger, P. O. Vontobel, A. Kavcic and W. Zeng, "Simulation-based computation of information rates for channels with memory," *IEEE Trans. Inform. Theory*, vol. 52, no. 8, pp. 3498–3508, Aug., 2006.

[137] H. D. Pfister, J. B. Soriaga and P. H. Siegel, "On the achievable information rates of finite state ISI channels," in *Proc. IEEE Global Communications Conference (GLOBECOM)*, pp. 2992–2996, Washington DC, Dec. 2007.

[138] N. Al-Dhahir, "FIR channel-shortening equalizers for MIMO ISI channels," *IEEE Trans. Commun.*, vol. 49, no. 2, pp. 213–218, Feb. 2001.

[139] K. B. Petersen and M. S. Pedersen, *The matrix cookbook*, Technical University of Denmark, Nov. 2008.

[140] R. M. Gray, *Toeplitz and circulant matrices: A review*, Foundations and Trends in Communication and Information Theory, NOW Publishers, vol. 2, no. 3.