UNIVERSIDAD DE A CORUÑA
Departamento de Electrónica y Sistemas

Ph.D. Thesis

DESIGN OF SERIALLY-CONCATENATED LDGM CODES

Autor: FRANCISCO JAVIER VÁZQUEZ ARAÚJO
Directores: LUIS CASTEDO RIBAS, MIGUEL GONZÁLEZ LÓPEZ
Octubre, 2008

UNIVERSIDAD DE A CORUÑA
Departamento de Electrónica y Sistemas

Ph.D. Thesis

DESIGN OF SERIALLY-CONCATENATED LDGM CODES

Autor: FRANCISCO JAVIER VÁZQUEZ ARAÚJO
Directores: LUIS CASTEDO RIBAS, MIGUEL GONZÁLEZ LÓPEZ
Octubre, 2008

D. Luis Castedo Ribas y D. Miguel González López

CERTIFICAN:

Que la memoria titulada "Design of Serially-Concatenated LDGM Codes", ha sido realizada por D. Francisco Javier Vázquez Araújo bajo nuestra dirección en el Departamento de Electrónica y Sistemas de la Universidad de A Coruña y concluye la Tesis que presenta para optar al grado de Doctor.

A Coruña, 17 de septiembre de 2008

Fdo: Dr. Luis Castedo Ribas
Director de la Tesis Doctoral
Catedrático de Universidad
Dpto. de Electrónica y Sistemas
Universidad de A Coruña

Fdo: Dr. Miguel González López
Director de la Tesis Doctoral
Profesor Contratado Doctor
Dpto. de Electrónica y Sistemas
Universidad de A Coruña

A mi familia

Agradecimientos

En primer lugar quiero expresar mi más profundo y sincero agradecimiento a mi tutor y director de tesis, el Dr. Luis Castedo Ribas, por confiar en mí desde el principio, proporcionarme los medios para realizar este trabajo y guiarme durante los últimos cinco años. Por haberme convencido de que hacer el doctorado sería una buena experiencia.

No menos importante es mi agradecimiento al Dr. Miguel González López, que además de ser también director de esta tesis ha sido mi principal fuente de inspiración y ayuda. Este trabajo tampoco hubiese sido posible sin él.

También quiero agradecerle al Dr. Javier García-Frías, de la Universidad de Delaware, las ideas y consejos proporcionados durante el desarrollo de los trabajos en los que se basa esta tesis.

A mi compañero de carrera y de doctorado Manuel, que ya son más de 10 años trabajando (y perdiendo el tiempo) juntos. A Paula, Héctor y Jose Antonio, por las discusiones que hemos mantenido acerca de temas relacionados con esta tesis (y de otros temas más mundanos). Y en general a todos los becarios que han ido pasando por el laboratorio y al resto de la gente del departamento, por haber hecho el trabajo más llevadero.

Y por supuesto a mi familia, a la que está dedicada esta tesis, por aguantarme durante todos estos años.

Resumen

Desde que Shannon demostrara en 1948 la posibilidad de alcanzar probabilidades de error arbitrariamente bajas en un sistema de comunicaciones siempre que la velocidad de transmisión se mantuviese por debajo de cierto límite, uno de los mayores retos en el ámbito de las comunicaciones y, más concretamente, en el campo de la codificación de canal, ha sido encontrar códigos que se acercasen lo más posible a dicho límite con una complejidad de codificación y decodificación razonable. Sin embargo, no fue hasta 1993, año en que Berrou *et al.* presentaron los turbo códigos, cuando se encontró un esquema de codificación capaz de aproximarse a fracciones de decibelio del límite de Shannon con probabilidades de error extremadamente bajas. La idea en la que se basan estos códigos es la decodificación iterativa de componentes concatenados que se intercambian información acerca de los bits transmitidos, lo que se conoce como el "principio turbo".

La generalización de la idea de los turbo códigos llevó en 1995 al redescubrimiento de los códigos LDPC (*Low Density Parity Check*), propuestos anteriormente por Gallager en la década de los 60. Los códigos LDPC son códigos bloque lineales con matrices control de paridad dispersas, capaces de superar el rendimiento de los códigos turbo con una complejidad de decodificación menor que la de éstos. Sin embargo, el hecho de que la matriz generadora de los códigos LDPC no sea dispersa hace que su complejidad de codificación pueda ser excesivamente alta. Los códigos LDGM (*Low Density Generator Matrix*), un caso particular de los LDPC, son códigos con una matriz generadora dispersa, gracias a lo cual presentan una menor complejidad de codificación. Sin embargo, excepto para el caso de tasas de codificación altas, los códigos LDGM son códigos "malos", es decir, tienen una probabilidad de error distinta de cero independiente del tamaño de bloque. Más recientemente se han propuesto los códigos IRA (*Irregular Repeat-Accumulate*), que consisten en la concatenación en serie de un código LDGM y un acumulador y que consiguen acercarse al rendimiento de los códigos LDPC con una complejidad de codificación similar a la de los LDGM.

En la presente tesis doctoral exploramos una alternativa a los códigos IRA que consiste en la concatenación en serie de dos códigos LDGM, esquema que vamos a denominar SCLDGM (*Serially-Concatenated Low Density Generator Matrix*). La premisa básica de los códigos SCLDGM es que un código interno de tasa cercana a la deseada corrija la mayor parte de los errores, y un código externo de tasa cercana a uno corrija los pocos errores que resultan de la decodificación del código interno.

Para que cualquiera de estos esquemas consiga acercarse lo máximo posible al límite de la capacidad es necesario determinar los parámetros del código que mejor se ajustan al canal sobre el que se va a realizar la transmisión. Las dos técnicas más utilizadas en la bibliografía para optimizar códigos LDPC son *Density Evolution* (DE) y *EXtrinsic Information Transfer* (EXIT) *charts*, con las que se han obtenidos códigos optimizados que funcionan a pocas décimas de decibelio de la capacidad del canal AWGN. Sin embargo, no se han descrito técnicas de optimización para códigos SCLDGM, que hasta el momento han sido diseñados de forma

heurística, alejándose del rendimiento conseguido por los códigos LDPC e IRA.

Otro de los avances más importantes que se han producido en los últimos años es la utilización de múltiples antenas en transmisión y recepción, en lo que se conoce como sistema MIMO (*Multiple-Input Multiple-Output*). Telatar demostró que la capacidad de este tipo de canales crece linealmente con el mínimo entre el número de antenas transmisoras y receptoras, lo que permite alcanzar eficiencias espectrales muy superiores a las de los sistemas con una sola antena en transmisión y recepción (o sistemas SISO, *Single Input Single Output*). Esta importante ventaja ha atraído una parte importante de la atención de la comunidad investigadora, y ha provocado que gran parte de los nuevos estándares, tales como WiMax 802.16e y WiFi 802.11n, así como los futuros sistemas 4G, estén basados en sistemas MIMO.

El principal problema de los sistemas MIMO es la elevada complejidad de la detección óptima, que crece exponencialmente con el número de antenas transmisoras y el número de niveles de la modulación. Se han propuesto diferentes algoritmos subóptimos para intentar reducir esta complejidad, entre los que destacan el SIC-MMSE (*Soft Interference Cancellation-Minimum Mean Square Error*) y los detectores esféricos. Otro problema importante es la elevada complejidad que conlleva la estimación de canal, debido al gran número de coeficientes que lo determinan. Existen técnicas, tales como *Maximum Likelihood-Expectation-Maximization* (ML-EM), que se han aplicado con éxito para estimar el canal MIMO pero que, al igual que en el caso de la detección, sufren del problema de una elevada complejidad al aumentar el número de antenas transmisoras o el tamaño de la constelación.

El principal objetivo del presente trabajo es el estudio y la optimización de códigos SCLDGM en canales SISO y MIMO. Para este fin, se propone un método de optimización para códigos SCLDGM basado en EXIT *charts* que permite a estos códigos superar el rendimiento de códigos IRA existentes en la literatura y acercarse al de los códigos LDPC, con la ventaja sobre éstos de una menor complejidad de codificación.

Asimismo, se proponen códigos SCLDGM optimizados para detectores subóptimos MIMO tanto esféricos como SIC-MMSE, constituyendo un sistema capaz de acercarse a los límites de la capacidad con una baja complejidad de codificación, detección y decodificación. Se analizan el esquema BICM (Bit-Interleaved Coded Modulation) y la concatenación de códigos SCLDGM con códigos espacio-temporales, en canales MIMO ergódicos y cuasiestáticos.

Se estudia también la combinación de estos códigos con diferentes algoritmos de estimación de canal, en los que se aprovecha la baja complejidad de los detectores subóptimos para reducir la complejidad de la estimación manteniendo una corta distancia al límite de la capacidad.

Finalmente, se proponen esquemas para tasas de codificación bajas consistentes en la concatenación en serie de varios códigos LDGM, reduciendo la complejidad de los esquemas propuestos recientemente basados en códigos Hadamard.

Summary

Since Shannon demonstrated in 1948 the feasibility of achieving an arbitrarily low error probability in a communications system provided that the transmission rate was kept below a certain limit, one of the greatest challenges in the realm of digital communications and, more specifically, in the channel coding field, has been finding codes that are able to approach this limit as much as possible with a reasonable encoding and decoding complexity. However, it was not until 1993, when Berrou et al. presented the turbo codes, that a coding scheme capable of performing at less than 1dB from Shannon's limit with an extremely low error probability was found. The idea on which these codes are based is the iterative decoding of concatenated components that exchange information about the transmitted bits, which is known as the "turbo principle".

The generalization of this idea led in 1995 to the rediscovery of LDPC (Low Density Parity Check) codes, proposed for the first time by Gallager in the 60s. LDPC codes are linear block codes with a sparse parity check matrix that are able to surpass the performance of turbo codes with a smaller decoding complexity. However, due to the fact that the generator matrix of general LDPC codes is not sparse, their encoding complexity can be excessively high. LDGM (Low Density Generator Matrix) codes, a particular case of LDPC codes, are codes with a sparse generator matrix, thanks to which they present a lower encoding complexity. However, except for the case of very high rate codes, LDGM codes are "bad", i.e., they have a non-zero error probability that is independent of the code block length. More recently, IRA (Irregular Repeat-Accumulated) codes, consisting of the serial concatenation of a LDGM code and an accumulator, have been proposed. IRA codes are able to get close to the performance of LDPC codes with an encoding complexity similar to that of LDGM codes.

In this thesis we explore an alternative to IRA codes consisting in the serial concatenation of two LDGM codes, a scheme that we will denote SCLDGM (Serially-Concatenated Low-Density Generator Matrix). The basic premise of SCLDGM codes is that an inner code of rate close to the desired transmission rate fixes most of the errors, and an external code of rate close to one corrects the few errors that result from decoding the inner code.

For any of these schemes to perform as close as possible to the capacity limit it is necessary to determine the code parameters that best fit the channel over which the transmission will be done. The two techniques most commonly used in the literature to optimize LDPC codes are Density Evolution (DE) and EXtrinsic Information Transfer (EXIT) charts, which have been employed to obtain optimized codes that perform at a few tenths of a decibel of the AWGN channel capacity. However, no optimization techniques have been presented for SCLDGM codes, which so far have been designed heuristically and therefore their performance is far from the performance achieved by IRA and LDPC codes.

Other of the most important advances that have occurred in recent years is the utilization of multiple antennas at the trasmitter and the receiver, which is known as MIMO (Multiple-Input Multiple-Output) systems. Telatar showed that the channel capacity in these kind of systems

scales linearly with the minimum number of transmit and receive antennas, which enables us to achieve spectral efficiencies far greater than with systems with a single transmit and receive antenna (or Single Input Single Output (SISO) systems). This important advantage has attracted a lot of attention from the research community, and has caused that many of the new standards, such as WiMax 802.16e or WiFi 802.11n, as well as future 4G systems are based on MIMO systems.

The main problem of MIMO systems is the high complexity of optimum detection, which grows exponentially with the number of transmit antennas and the number of modulation levels. Several suboptimum algorithms have been proposed to reduce this complexity, most notably the SIC-MMSE (Soft-Interference Cancellation Minimum Mean Square Error) and spherical detectors. Another major issue is the high complexity of the channel estimation, due to the large number of coefficients which determine it. There are techniques, such as Maximum-Likelihood-Expectation-Maximization (ML-EM), that have been successfully applied to estimate MIMO channels but, as in the case of detection, they suffer from the problem of a very high complexity when the number of transmit antennas or the size of the constellation increase.

The main objective of this work is the study and optimization of SCLDGM codes in SISO and MIMO channels. To this end, we propose an optimization method for SCLDGM codes based on EXIT charts that allow these codes to exceed the performance of IRA codes existing in the literature and get close to the performance of LDPC codes, with the advantage over the latter of a lower encoding complexity.

We also propose optimized SCLDGM codes for both spherical and SIC-MMSE suboptimal MIMO detectors, constituting a system that is capable of approaching the capacity limits of MIMO channels with a low complexity encoding, detection and decoding. We analyze the BICM (Bit-Interleaved Coded Modulation) scheme and the concatenation of SCLDGM codes with Space-Time Codes (STC) in ergodic and quasi-static MIMO channels.

Furthermore, we explore the combination of these codes with different channel estimation algorithms that will take advantage of the low complexity of the suboptimum detectors to reduce the complexity of the estimation process while keeping a low distance to the capacity limit.

Finally, we propose coding schemes for low rates involving the serial concatenation of several LDGM codes, reducing the complexity of recently proposed schemes based on Hadamard codes.

Index

List of figures

List of tables

Chapter 1

Introduction

Achieving the channel capacity limit with a digital communications system of affordable complexity has been the main goal of Information Theory since its birth, in 1948. That year Shannon published his celebrated paper, "*A mathematical theory of communication*" [1], in which the surprising result of the channel capacity was established. Before Shannon's paper, it was believed that error-free digital communication was only possible by increasing the signal-to-noise ratio to infinity or reducing the transmission rate to zero. Shannon demonstrated that, with the appropriate channel code, it was possible to achieve error-free communication over a noisy channel with a finite signal-to-noise ratio if and only if the transmission rate was lower than a certain value greater than zero. This value was termed the channel capacity.

Since Shannon demonstrated the existence of this limit, much of the effort in the field of digital communications has been devoted to the search of practical channel codes capable of approaching it. Shannon's theorem only proved that infinitely long random codewords could achieve the limit, but using that approach to design a real code was believed to be impossible. Thus the search was initially directed towards small structured codes.

The first codes developed after the publication of Shannon's paper were short binary linear block codes. The main objective when searching for good block codes was to maximize their minimum distance between codewords (that mainly determines their performance) while at the same time keeping the decoding complexity at a manageable level, so the search for good codes was initially treated as an algebraic problem. To perform decoding, early block codes (such as Hamming [2], Golay [3] or Reed-Muller [4, 5]) used different algorithms to search for the codeword that was closer in Hamming distance to the received one, that was constructed by taking hard decisions over the observations from the channel. These algorithms took advantage of the properties of the sub-vector space generated by the code to avoid exhaustive searches over all the possible codewords, so their complexity was relatively small. Nevertheless, the performance of these short codes was several dBs away from the capacity limit predicted by Shannon. One of the reasons for their mediocre performance was that although minimum Hamming distance is the optimum decoding rule for some channels, such as the Binary Simmetric Channel (BSC), when transmitting over an AWGN channel

optimum decoding consists in looking for the codeword that is closer in Euclidean distance to the received observations. Taking hard decisions over the received observations caused a loss of information, that translated in a loss of performance. The first family of block codes that used *soft* information from the channel (i.e., the channel observations directly) was presented in 1954 [6], but their minimum distance was only two and therefore their usefulness was limited.

Although the search for linear block codes with good algebraic properties continued for a couple of decades before stalling, a different approach, inspired by Shannon's idea of probabilistic coding, began to emerge soon after the presentation of his paper. The objective was to simply search for codes by minimizing their error probability as a function of their complexity. The most notable example of this more pragmatic approach was convolutional coding [7]. Convolutional codes began to be widely used to the detriment of linear block codes, especially after the presentation of the Viterbi algorithm [8]. The Viterbi algorithm provided a simple way to find the maximum likelihood estimate of the transmitted sequence with linear complexity. Also, the Viterbi algorithm considered the soft information from the detector naturally, which had been already demonstrated as the best method for decoding in AWGN channels. A few years later, Bahl, Cocke, Jelinek and Raviv [9] further extended the use of soft information to the output of the convolutional codes, with an algorithm (known as BCJR) that was able to calculate the *a posteriori* probabilities of the bits forming the codeword and thus minimize the bit error probability. However, since the BCJR algorithm did not provide an important advantage in performance over the Viterbi algorithm and its complexity was significantly higher, the Viterbi algorithm continued to be the standard decoding method for convolutional codes for the next decades.

By the time the Viterbi algorithm was introduced, another important step towards the channel capacity limit came in the form of concatenated codes [10]. The idea was to use two simple codes, with a simple encoding and decoding algorithm, that when put together led to a more powerful code. The optimum decoding of this concatenated scheme would involve cosidering the global code resulting from the concatenation, but simply decoding each component code still could result in a good performance. It was originally presented as the concatenation of two block codes, but soon was extended to convolutional codes. An scheme concatenating an outer Reed-Solomon block code and an inner convolutional code became the NASA standard in the 70s, with a performance that was less than 3dB away from the AWGN capacity limit [11].

Although two of the key ideas to approach the Shannon limit were already discovered (concatenated coding and a soft-input soft-output decoding algorithm), it was not until almost two decades later that they were combined and put to practice. In 1993, Berrou, Glavieux and Thitimajshima presented what would be the cornerstone of coding theory for the next years: Turbo Coding [12]. Turbo codes took those two concepts and added the original Shannon's idea of long random-like codes. The resulting scheme consisted of two simple convolutional codes that encoded two different permutations of the same long information sequence. At the receiver, the BJCR algorithm was used in each component decoder. These components interchanged

their soft output information in an iterative fashion, each one using the output probabilities from the other as *a priori* information. This simple scheme was able to perform at 0.5dB from the AWGN capacity limit with reasonable complexity, which resulted in an impressive improvement over the previous best codes. Turbo codes also introduced the concept of extrinsic information, defined as the *a posteriori* information after removing the *a priori* information, i.e., the information *added* by each component to the input information. The exchange of extrinsic information between components, known from then on as the turbo principle, constituted the base for the decoding schemes developed after the Turbo codes.

Two years after Turbo codes were introduced, MacKay [13] rediscovered Gallager's Low Density Parity Check (LDPC) codes [14], that had been introduced in 1962 and forgotten due to their (for the time) extremely high encoding and decoding complexity. LDPC codes were the direct application of Shannon's idea of large, random codes to linear block coding, with the added restriction of considering only sparse parity check matrices in order to allow a feasible decoding. MacKay showed that LDPC codes were able to approach capacity with linear decoding complexity thanks to the use of the Belief Propagation algorithm [15], and the overwhelming increase in computation capacity from the early sixties to the mid nineties allowed these codes to be implemented practically. His work motivated an outburst in the study of codes defined by sparse matrices, which are seen today as the best codes for practical applications and, consequently, the chosen for many modern standards, such as WiFi IEEE 802.11n [16, 17], WiMax IEEE 802.16e [18] and DVB-S2 [19, 20].

In his paper, MacKay also described Low Density Generator Matrix (LDGM) codes, a subclass of LDPC codes with sparse generator matrices. Unlike LDPC codes, LDGM codes cannot approach capacity due to their poor minimum distance, which caused them to be disregarded for most practical applications [21]. However, a coding scheme constituted by two serially concatenated LDGM codes was presented a few years ago [22], showing that these codes can indeed aproach capacity with practical block lengths. In this thesis we study the desing and optimization of concatenated LDGM codes in a wide diversity of configurations and channel models. We will show that, when properly optimized, these codes are able to perform very close to the capacity limits and, in some cases, surpass the performance of general LDPC codes with similar decoding complexity and a lower encoding complexity.

A more detailed review of the birth of Information Theory and the history of channel coding can be found in [23, 24]

1.1 Thesis Overview

This thesis is organized as follows. In Chapter 2 we introduce the concept of channel capacity and review three coding techniques that are able to approach the capacity limits: Turbo codes, Low Density Parity Check (LDPC) codes and Repeat-Accumulate (RA) codes. We will pay special attention to LDPC codes, since they constitute the basis of the scheme that will be studied in the rest of the Thesis: Serially-Concatenated Low Density Generator Matrix

(SCLDGM) codes. In this chapter we also present the optimization procedures used to optimize the previous coding schemes, focusing especially on EXtrinsic Information Transfer (EXIT) charts, as they will be also utilized in the optimization of SCLDGM codes.

In Chapter 3 we introduce the SCLDGM scheme, and analyze its encoding, decoding and optimization. We will study the performance of optimized SCLDGM codes over AWGN and Rayleigh fading SISO channels and show that they are able to approach the capacity limits and surpass the performance of IRA codes.

This study will be extended to Multiple Input Multiple Output (MIMO) systems in Chapter 4, in particular to a Bit-Interleaved Coded Modulation (BICM) scheme in which the modulated symbols are simply multiplexed to the multiple transmit antennas. We examine several suboptimal detectors that solve the problem of exponential complexity of MIMO detection, and find optimum SCLDGM codes for each of them.

In Chapter 5 we continue the study of SCLDGM-coded MIMO systems by considering the concatenation of SCLDGM codes with Space-Time Block Codes (STBC). Specifically, we will study three different STBC codes: the Alamouti code, a Linear Dispersion (LD) code and the Golden code. We will analyze their advantages and disadvantages when compared with the BICM scheme studied in the previous chapter.

In Chapter 6 we investigate the performance of the SCLDGM scheme when the channel is estimated at the receiver. We focus on two Pilot Symbol Aided Modulation (PSAM) schemes to estimate block and quasistatic channels: Least Squares (LS) and Maximum-Likelihood with Expectation-Maximization (ML-EM).

In Chapter 7 we find a solution to the degradation in performance that the schemes presented in previous chapters suffer when considering low rates. We present two schemes, Layered LDGM (LLDGM) codes and IRA-LDGM codes, that are able to perform well even with rates as low as 1/50. We also study the performance of LLDGM codes in a practical multiuser Interleaved-Division Multiple Access (IDMA) scheme, where the use of low rates is mandatory to be able to separate each transmitted stream.

Finally, Chapter 8 is devoted to the conclusions and future work.

Chapter 2

Capacity-approaching coding

A noisy channel poses a limit on the rate at which information can be transferred through it without errors. This limit is known as the channel capacity, and was first introduced by Shannon in 1948 [1]. For many channels, their noisiness can be measured by a single parameter (for example, the relation between the strength of the transmitted signal and the noise power, termed as Signal-to-Noise Ratio (SNR)), and the value of this parameter uniquely determines the maximum data transfer rate that can be achieved under the constraint of error-free transmissions. Correspondingly, there is also a minimum value of the channel parameter for achieving a transmission rate (without errors) greater than zero.

Although the channel capacity limit has been known for more than half a century, it was not until the last decade that some coding schemes were able to approach it. In this chapter we will review three of these schemes: Turbo codes [12], Low Density Parity Check (LDPC) codes [14] and Repeat Accumulate (RA) codes [25]. We will mainly focus our attention in LDPC and RA codes, since they render better performance than turbo codes with a lower complexity.

For these schemes to be able to perform close to the capacity limit, it is necessary to carefully choose the parameters that determine their behavior. The optimization techniques that have been proposed for this task are based on the Density Evolution (DE) principle [26, 27], consisting in tracking the evolution of the probability density function of the messages interchanged in the decoding algorithm. The actual density of messages is continuous-valued, so it is not possible to track it in practice. Two main practical approaches have arisen to overcome this limitation: density quantization, which is computationally involved and its associated error is not easy to control; and density characterization through parameters. Towards this aim, the usual assumption is considering the densities as symmetric Gaussian with a know relation between the variance and mean, so one parameter suffices to completely characterize the density. Two main approaches have been proposed: tracking the mean, known as Density Evolution with Gaussian Approximation (DE-GA) [28], and tracking the mutual information between the message density and the *a priori* bit density, known as Information Content (IC) evolution [29]. IC evolution is particularly useful when the decoding process can be structured in two decoding modules, because in this case the two IC values can be simultaneously plotted, graphically

Figure 2.1: Block diagram of a transmission system

showing the decoding process. This representation is known as Extrinsic Information Transfer (EXIT) charts. We will explain how IC evolution and EXIT charts can be applied to the design of good LDPC and RA codes for transmission over the AWGN channel.

2.1 Channel capacity

The ultimate characterization of communication systems is the realm of the discipline of Information Theory. Information Theory models information sources as random variables, where their possible outcomes constitute the messages emitted by them. In this context, the *information* produced by a source S is defined as its entropy [1], given by

$$H(S) \triangleq -\mathbb{E}_S \left\{ \log \left(p_S(s) \right) \right\} \tag{2.1}$$

where $\mathbb{E}_S\{\cdot\}$ denotes the expectation operator with respect to the random variable S. The logarithm base determines the entropy measurement unit. If we choose 2 as the base (the most usual), the entropy is expressed in bits. When using natural logarithms, entropy is measured in nats.

Let us assume a digital communication system in which a sequence of L symbols $\mathbf{s} = [s_1, \ldots, s_L]$ is transmitted through a noisy channel. Each symbol s_i is an outcome of the random variable S, i.e., $s_i \sim S$. The received sequence will be denoted as $\mathbf{x} = [x_1, \ldots, x_L]$, where $x_i \sim X$ (Figure 2.1). The channel input-output relationship is characterized by the conditional distribution function $p(\mathbf{x}|\mathbf{s})$. Assuming the channel is memoryless,

$$p(\mathbf{x}|\mathbf{s}) = p(x_1, x_2, \ldots, x_L | s_1, s_2, \ldots, s_L) = \prod_{i=1}^{L} p_{X|S}(x_i|s_i) \tag{2.2}$$

where $p_{X|S}(x|s)$ is the known, fixed conditional distribution function of X given S.

Shannon defined the capacity of such a channel as the maximum amount of information that can be communicated through it with vanishing error probability. It can be shown [1] that this equals the maximum, over all the possible distributions of the input S, of the mutual information between the input and the output of the channel, i.e.,

$$C = \max_{p_S} I(S; X) \tag{2.3}$$

where the mutual information is defined as

$$I(S; X) \triangleq \mathbb{E}_{X,S} \left\{ \log \left(\frac{p_{S,X}(s, x)}{p_S(s) p_X(x)} \right) \right\} \tag{2.4}$$

If the base of the logarithm is 2, the capacity is expressed in bits per channel use, and measures how many bits of information can be sent in each symbol s. Elaborating Eq. (2.4) we have

$$\begin{aligned} I(S;X) &= \mathbb{E}_{X,S}\left\{\log\left(\frac{p_{X|S}(x|s)}{p_X(x)}\right)\right\} \\ &= \mathbb{E}_{X,S}\left\{\log(p_{X|S}(x|s))\right\} - \mathbb{E}_X\left\{\log(p_X(x))\right\} \\ &= H(X) - H(X|S) \end{aligned} \tag{2.5}$$

where

$$H(X) = -\mathbb{E}_X\left\{\log p_X(x)\right\} \tag{2.6}$$

is the entropy of X and

$$H(X|S) = -\mathbb{E}_{X,S}\left\{\log p_{X|S}(x|s)\right\} \tag{2.7}$$

is the conditional entropy of X given S. Similarly, it can be easily shown that

$$I(S;X) = I(X;S) = H(S) - H(S|X). \tag{2.8}$$

The conditional entropy $H(S|X)$ can be seen as the remaining uncertainty of the transmitted symbols given the observations at the receiver. If the received observations do not provide any information about the transmitted symbols, i.e., if $p_{S|X}(s|x) = p_S(s)$, the conditional entropy equals the entropy of the transmitted symbols, and the mutual information $I(S;X) = H(S) - H(S) = 0$. Conversely, if the channel introduces no distortion, it can be easily seen that $H(S|X) = 0$, so the mutual information equals the entropy of S and no information is lost through the channel. From this we can say that mutual information (capacity) measures how much of the original information in fact traverses the channel.

Let us now assume that we have a binary source of information and that we pack that information in words of K bits. A fixed-length binary channel code of rate $R_c = K/N$ is an injective mapping from the set of 2^K possible binary information sequences to a set of binary words of length N, with $N > K$. The function of the code is to add some redundancy to the information bits, so that if the received word after the transmission through a channel is one of the possible $2^N - 2^K$ sequences that are not associated with any information sequence, it is possible to detect that an error has been produced and proceed to correct it. After coding, and prior to transmission through the channel, the resulting bits are grouped in words of length M_c and mapped to symbols from a set S, with $|S| = 2^{M_c}$. The transmission rate is thus $R_c M_c$ information bits per transmitted symbol, or bits per channel use. Shannon demonstrated that, for any transmission rate below the channel capacity, there exists a code that can achieve an arbitrarily small block error probability, just by increasing the block length N. Conversely, if the tranmission rate is greater than the channel capacity, the block error probability tends to 1 as the block length tends to infinity, regardless of the used code.

For example, let us consider an Additive White Gaussian Noise (AWGN) channel, in which the transmitted real symbols have an average energy $E_s = \mathbb{E}_S\{s^2\}$ and the received symbols are obtained as

$$x_k = s_k + n_k \tag{2.9}$$

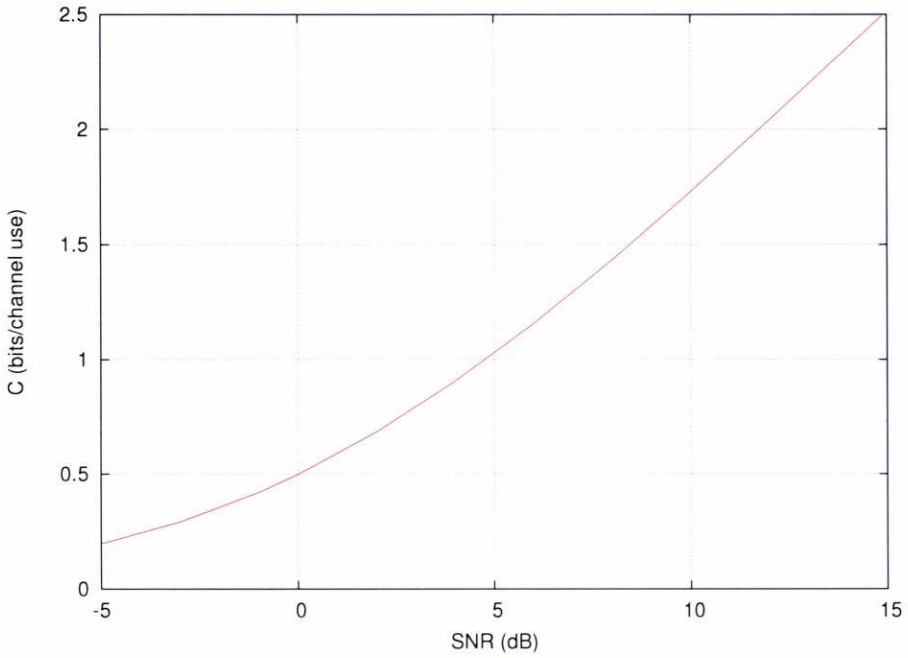


Figure 2.2: Capacity of an AWGN channel

where n_k are independent and identically distributed (i.i.d.) samples of Gaussian noise with zero mean and variance $\sigma_n^2 = N_0/2$. It can be shown that the distribution of the input that maximizes the mutual information for this channel is the Gaussian distribution [1], and that the capacity associated to it is given by

$$C = \frac{1}{2}\log_2(1 + \text{SNR}) \quad \text{bits/channel use} \tag{2.10}$$

where $\text{SNR} = E_s/\sigma_n^2 = 2E_s/N_0$ is the Signal to Noise Ratio. Figure 2.2 shows the capacity versus the SNR for an AWGN channel. For example, if we fix the SNR at the value 0dB, the Shannon theorem affirms that we can find a coding scheme allowing us to communicate without errors through the AWGN channel as long as the transmission rate is below 0.5 bits per channel use. Conversely, if we fix the information rate at 0.5 bits per channel use, we can transmit without errors as long as SNR $>$ 0dB, whereas it is impossible to guarantee error-free transmission for SNR $<$ 0dB, regardless of the employed coding scheme

It is usually more interesting to express the capacity of a channel in terms of the parameter E_b/N_0, where E_b is the average energy per information bit. If we use a code of rate R_c and transmit M_c bits per symbol, the average energy per information bit is given by $E_b = E_s/(R_c M_c)$. This way, we can rewrite Eq. (2.10) as

$$C = \frac{1}{2}\log_2\left(1 + 2R_c M_c \frac{E_b}{N_0}\right) \quad \text{bits/channel use} \tag{2.11}$$

Since we are trying to transmit at a rate of $R_c M_c$ bits per channel use, we have to satisfy

$$R_c M_c < C = \frac{1}{2}\log_2\left(1 + 2R_c M_c \frac{E_b}{N_0}\right) \Rightarrow \frac{E_b}{N_0} > \frac{2^{2R_c M_c} - 1}{2R_c M_c} \tag{2.12}$$

When the information rate $R_c M_c$ tends to 0, which means that the transmission speed also tends to 0, we have

$$\lim_{R_c M_c \to 0} \frac{2^{2R_c M_c} - 1}{2R_c M_c} = \ln(2) \approx -1.59\text{dB} \tag{2.13}$$

It can be easily seen that this is the minimum of all the E_b/N_0 limits for the AWGN channel, i.e., the lowest E_b/N_0 for which it is possible to transmit without errors, and is usually known as the *ultimate Shannon limit*.

2.2 Capacity-approaching codes

Since the introduction of the channel capacity concept, one of the main objectives in Information Theory has been to find practical codes able to approach the E_b/N_0 limit with vanishingly small error probability. In the following sections we will review the most important results obtained in the field of capacity-approaching codes during the last decade.

2.2.1 Turbo codes

Turbo codes [12] were originally presented as two parallel convolutional codes that encode two different permutations $\mathbf{u}$ and $\mathbf{u}'$ of the same information sequence (Figure 2.3). The encoded sequences $\mathbf{c}$ and $\mathbf{c}'$ are merged and sent through the channel. This results in that the produced codewords show a pseudorandom structure.

At the receiver, the detector calculates the log-likelihood ratio (LLR) of each coded bit c_i, defined as

$$L_{\text{ch},i} = \log \frac{p(\mathbf{x}|c_i = 1)}{p(\mathbf{x}|c_i = 0)} \tag{2.14}$$

and passes them to the decoder. In the decoder, a soft-output algorithm that calculates the log-*a posteriori* probability ratios (LAPRs) of the source bits is employed over the trellis defined by each convolutional code [9, 30]. The output LAPRs (L_O) of each decoder component are used by the other as *a priori* information (L_A) to improve its previous decoding output in an iterative fashion. To avoid positive feedback (i.e., counting the same information twice), the *a priori* information is subtracted from the *a posteriori* information before being passed to the other decoder, giving result to what is called the *extrinsic* information (L_E).

The key idea behind the performance of turbo codes is the iterative exchange of extrinsic information between the two component decoders. This is called the "turbo principle". Since the introduction of turbo codes, it has been successfully applied to other problems, such as equalization and detection [31, 32]. This same idea is the basis for the decoding of Low Density Parity Check (LDPC) codes described in the following section.

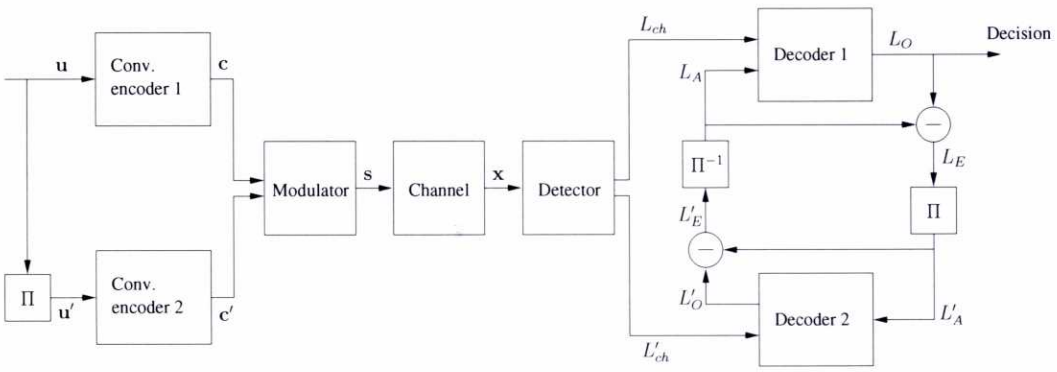


Figure 2.3: Block diagram of a turbo code

2.2.2 LDPC codes

Low Density Parity Check (LDPC) codes [14] are linear block codes with a large parity check matrix $\mathbf{H}$ that is sparse, i.e., with a very low number of elements different from zero. Assuming that the codeword length is N, the code is formed by the sequences $\mathbf{c} = [c_1, c_2, \ldots, c_N]$ that satisfy

$$\mathbf{c}\mathbf{H}^T = \mathbf{0}$$

where the size of the parity check matrix is $(N - K) \times N$, being K the length of the uncoded sequence. Superindex ^{T} represents the matrix transpose operator.

Encoding of a source vector $\mathbf{u} = [u_1, u_2, \ldots, u_K]$ using a rate $R_c = K/N$ LDPC code is performed by multiplying the source vector by a $K \times N$ generator matrix $\mathbf{G}$ which is calculated from the parity check matrix, i.e.,

$$\mathbf{c} = [c_1, c_2, \ldots, c_N] = \mathbf{u}\mathbf{G}$$

To find the generator matrix, the parity check matrix is first put in systematic form $\mathbf{H} = [\mathbf{P}^T|\mathbf{I}_{N-K}]$ using Gaussian elimination. The generator matrix is then simply constructed as $\mathbf{G} = [\mathbf{I}_K|\mathbf{P}]$. The problem with this method is that, in general, although the parity-check matrix is sparse, the generator matrix will not be sparse after the Gaussian elimination, so the encoding complexity can be extremely high. To avoid this issue there exist techniques [33] that transform the parity check matrix without losing most of its sparseness in such a way that it can be used to generate the codewords directly (i.e., without needing a generator matrix), so encoding can be done with reasonable complexity.

Let us assume that the coded bits $\mathbf{c}$ are modulated, transmitted through a noisy channel and received as the sequence $\mathbf{x} = [x_1, x_2, \ldots, x_L]$. Optimum decoding of LDPC codes (and, in general, of any other code) consists in searching for the codeword $\hat{\mathbf{c}}$ that maximizes the *a posteriori* probability (APP), that is,

$$\hat{\mathbf{c}} = \arg\max_{\mathbf{c}\in C} p(\mathbf{c}|\mathbf{x}) \tag{2.15}$$

where C is the set of all the possible codewords. Since the number of codewords in C is 2^K, an exhaustive search is obviously infeasible (the problem is NP-complete). Instead, following the idea of the turbo principle, we can use a suboptimum iterative algorithm that exchanges extrinsic information between different decoding components to calculate the APP of the bits forming the codeword, $p(c_i|\mathbf{x})$. To do this, we must first represent the LDPC code using a *factor graph* [34, 35, 36], a type of bipartite graph that relates a set of variable nodes and a set of factor nodes.

Code representation is a specific application of factor graphs. Generally, factor graphs are used to represent the factorization of a function of several variables

$$g(x_1, x_2, \ldots, x_N) = \prod_{j=1}^{F} f_j(X_j) \tag{2.16}$$

where the sets X_j are different subsets of $X = \{x_1, x_2, \ldots, x_N\}$. The graph is formed by N variable nodes representing the variables in X and F factor nodes representing the functions f_j. There exists a connection between a variable node i and a factor node j if and only if the function f_j has the variable x_i as an argument, i.e., if $x_i \in X_j$.

When the function g is a joint probability mass function, we can calculate the marginal probability functions of each variable as

$$g_i(x_i) = \sum_{\sim\{x_i\}} g(x_1, x_2, \ldots, x_N) \tag{2.17}$$

where the sum index represents all the possible values taken by all the variables except x_i, which is fixed, i.e.,

$$\sum_{\sim\{x_i\}} g(x_1, x_2, \ldots, x_N) \triangleq \sum_{x_1} \sum_{x_2} \cdots \sum_{x_{i-1}} \sum_{x_{i+1}} \cdots \sum_{x_N} g(x_1, x_2, \ldots, x_N) \tag{2.18}$$

If the fractor graph can be put in tree form (i.e., if it has no loops) it can be easily demonstrated by substituting (2.16) in (2.17) and using the distributive property that the marginal probabilities can be calculated by applying a message-passing algorithm on the graph. This algorithm, known as Sum-Product Algorithm (SPA) [14] (and sometimes as Belief Propagation [37]), calculates two kind of messages: the messages passed by a variable node x_i to a factor node f_j

$$\mu_{x_i \to f_j}(x_i) = \prod_{k \neq j} \mu_{f_k \to x_i}(x_i) \tag{2.19}$$

and the messages passed by a factor node f_j to a variable node x_i

$$\mu_{f_j \to x_i}(x_i) = \sum_{\sim\{x_i\}} \left(f_j(X_j) \prod_{k \neq i} \mu_{x_k \to f_j}(x_k) \right) \tag{2.20}$$

where the sum is done over all the values that can be taken by all the variables in X_j except x_i, which is fixed. The SPA starts in the leaves and computes all the messages in the tree. After all of them have been calculated, the marginal probabilities can be calculated as

$$g_i(x_i) = \prod_{j=1}^{F} \mu_{f_j \to x_i}(x_i) \tag{2.21}$$

We can also use the SPA to find the marginal probabilities when the graph contains cycles (as will be the case of LDPC factor graphs), by calculating the messages iteratively following a certain update schedule. However, in this case the SPA leads to a suboptimal solution [15].

When considering LDPC codes, our objective is to calculate the marginal probabilites of the bits forming the codeword. Assuming that all the codewords have the same a priori probability $p(\mathbf{c})$, we can factorize the joint probability function of a codeword given the received symbol sequence as

$$p(\mathbf{c}|\mathbf{x}) \propto p(\mathbf{c})p(\mathbf{x}|\mathbf{c}) = \frac{1}{|C|}\chi_C(c_1, c_2, \ldots, c_N)p(\mathbf{x}|c_1, c_2, \ldots, c_N) \triangleq g(c_1, c_2, \ldots, c_N) \tag{2.22}$$

where $|C| = 2^K$ is the number of codewords and $\chi_C(c_1, c_2, \ldots, c_N)$ is the behavioral model of the code, and is equal to 1 if the codeword $[c_1, c_2, \ldots, c_N]$ belongs to C and 0 otherwise. In a LDPC code the codewords must satisfy the parity conditions imposed by each row in the parity check matrix, so this function itself can be factorized as

$$\chi_C(c_1, c_2, \ldots, c_N) = \prod_{j=1}^{F} f(C_j) \tag{2.23}$$

being F the number of parity bits in the LDPC code and

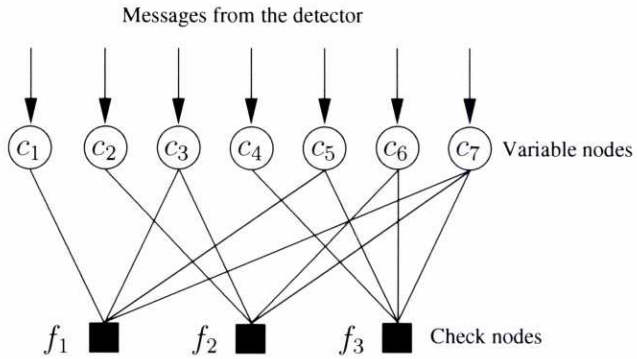
$$f(C_j) = \begin{cases} 1 & \text{if } \sum_{c_i \in C_j} c_i \bmod 2 = 0 \\ 0 & \text{in other case.} \end{cases} \tag{2.24}$$

where C_j contains the bits related to the j-th parity check.

For example, with the parity check matrix $\mathbf{H}$ of Figure 2.4, the behavioral model is given by

$$\chi_C(c_1, c_2, c_3, c_4, c_5, c_6, c_7) = f(c_4, c_5, c_6, c_7)f(c_2, c_3, c_6, c_7)f(c_1, c_3, c_5, c_7) \tag{2.25}$$

With this factorization, we obtain the factor graph represented in that same figure. The variable nodes in the factor graph represent the coded bits and correspond to the columns of the parity check matrix, whereas the factor (or check) nodes represent a parity relation among a certain number of bits and correspond to the rows of the parity check matrix. There exists a connection (edge) between a factor node and a variable node if and only if there is a "1" in the corresponding row and column of the parity matrix. The number of edges connected to a node is called the



$$\mathbf{H} = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix}$$

Figure 2.4: Factor graph of a 3×7 parity check matrix

degree of the node, and the set of different degrees of the variable and check nodes is the *degree profile* of the code. Usually, the parity check matrices of LDPC codes are randomly generated, following the restrictions imposed by the degree profile. If all the nodes of a type (either variable or check) have the same degree, the code is said to be *regular*, and *irregular* in the other case. This definition can also be applied to other codes defined on graphs.

When the SPA is applied to the particular problem of LDPC decoding, the variable to check messages are calculated as

$$\mu_{c_i \to f_j}(c_i) = \prod_{k \neq j} \mu_{f_k \to c_i}(c_i) \tag{2.26}$$

and, by substituting (2.24) in (2.20), the check to variable messages are simplified to

$$\mu_{f_j \to c_i}(c_i) = \sum_{C_j^i} \prod_{k \neq i} \mu_{c_k \to f_j}(c_k) \tag{2.27}$$

where C_j^i is the set of binary vectors in C_j such that their modulo-2 sum is 0 and the bit corresponding to the variable node c_i is fixed.

Since the variables c_i can only take two values, 0 and 1, instead of handling the messages $\mu(0)$ and $\mu(1)$, it is more efficient to use the L-values, defined as

$$L = \log \frac{\mu(1)}{\mu(0)} \tag{2.28}$$

The messages $\mu(1)$ and $\mu(0)$ represent conditional probabilities of the two possible values of a bit, so they must satisfy $\mu(1) + \mu(0) = 1$, and thus we can recover both values with

$$\mu(0) = \frac{1}{1 + e^L} \qquad \mu(1) = \frac{e^L}{1 + e^L} = \frac{1}{1 + e^{-L}} \tag{2.29}$$

If we particularize Eq. (2.19) for the factor graph of an LDPC code and express it as a function

of the L-values, we can compute the variable-to-check messages as

$$\begin{aligned} L_{c_i \to f_j} &= \log \frac{\mu_{\text{ch},i}(1) \prod_{k \neq j} \mu_{f_k \to c_i}(1)}{\mu_{\text{ch},i}(0) \prod_{k \neq j} \mu_{f_k \to c_i}(0)} \\ &= \log \frac{\mu_{\text{ch},i}(1)}{\mu_{\text{ch},i}(0)} + \sum_{k \neq j} \log \frac{\mu_{f_k \to c_i}(1)}{\mu_{f_k \to c_i}(0)} = L_{\text{ch},i} + \sum_{k \neq j} L_{f_k \to c_i} \end{aligned} \tag{2.30}$$

where $\mu_{\text{ch},i}(b) = p(\mathbf{x}|c_i = b)$ is the likelihood of the received symbols given the bit c_i, calculated by the detector, and $L_{\text{ch},i}$ is its corresponding LLR. Similarly, we can compute the check-to-variable messages as

$$L_{f_j \to c_i} = \log \frac{\sum_{C_j^{i,1}} \prod_{k \neq i} \mu_{c_k \to f_j}(c_k)}{\sum_{C_j^{i,0}} \prod_{k \neq i} \mu_{c_k \to f_j}(c_k)} \tag{2.31}$$

being $C_j^{i,b}$ the set of binary vectors in C_j such that their modulo-2 sum is 0 and the bit corresponding to the variable node c_i is equal to b. Substituting the expresions from (2.29) in (2.31), it can be proven by induction [34] that the message can be calculated as

$$\begin{aligned} L_{f_j \to c_i} &= \log \frac{\prod_{k \neq i}(e^{L_{c_k \to f_j}} + 1) + \prod_{k \neq i}(e^{L_{c_k \to f_j}} - 1)}{\prod_{k \neq i}(e^{L_{c_k \to f_j}} + 1) - \prod_{k \neq i}(e^{L_{c_k \to f_j}} - 1)} \\ &= \log \frac{1 + \prod_{k \neq i} \frac{e^{L_{c_k \to f_j}} - 1}{e^{L_{c_k \to f_j}} + 1}}{1 - \prod_{k \neq i} \frac{e^{L_{c_k \to f_j}} - 1}{e^{L_{c_k \to f_j}} + 1}} \\ &= 2 \tanh^{-1} \left(\prod_{k \neq i} \tanh \frac{L_{c_k \to f_j}}{2} \right) \end{aligned} \tag{2.32}$$

where

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} = \frac{e^{2x} - 1}{e^{2x} + 1} \tag{2.33}$$

and

$$\tanh^{-1}(x) = \frac{1}{2} \log \frac{1 + x}{1 - x} \tag{2.34}$$

The messages $L_{c_i \to f_j}$ and $L_{f_j \to c_i}$ are iteratively calculated for every node in the factor graph until the algorithm converges or a maximum number of iterations is performed. Then, a decision is made based on the output L-values of the variable nodes, that represent the log ratio of the bits APP, and that are calculated as

$$L_{O,i} = L_{\text{ch},i} + \sum_j L_{f_j \to c_i} \approx \log \frac{p(c_i = 1|\mathbf{x})}{p(c_i = 0|\mathbf{x})} \tag{2.35}$$

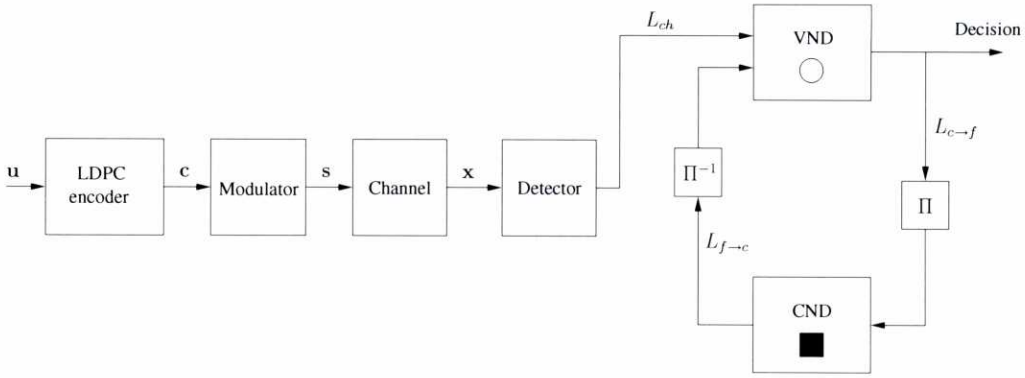


Figure 2.5: LDPC coded system

Thus, if the output L-value is greater than 0, the bit is decided as 1, and 0 in the other case.

Each message $L_{\mathrm{ch},i}$ passed to a variable node is calculated as the LLR of the coded bit corresponding to that variable node (see Eq. (2.14)). The calculation of the LLR depends on the modulation and channel model assumed for the transmission. For example, if the coded bits are BPSK (i.e., $s_i = 2c_i - 1$) and transmitted through an AWGN channel as the one given by Eq. (2.9), $L_{\mathrm{ch},i}$ is calculated as

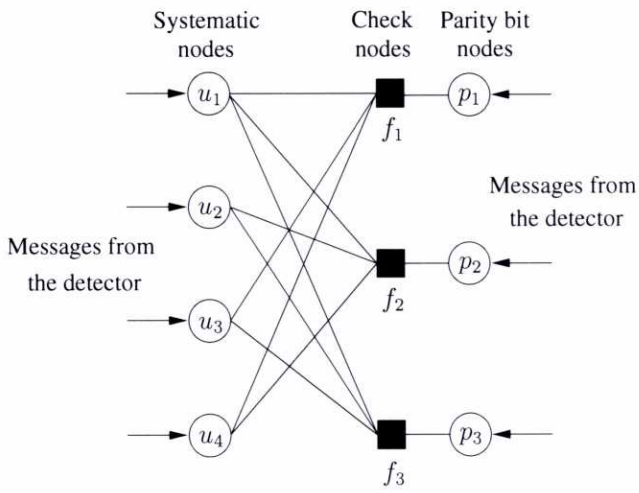
$$L_{\mathrm{ch},i} = \log \frac{p(\mathbf{x}|c_i = 1)}{p(\mathbf{x}|c_i = 0)} = \log \frac{p(x_i|s_i = +1)}{p(x_i|s_i = -1)} = \frac{2x_i}{\sigma_n^2} \tag{2.36}$$

The complexity of the SPA is linear with the block length of the code thanks to the sparseness of the parity check matrix. This ensures that the number of messages to be calculated for every node is a fixed small number. However, its complexity can be further reduced by computing the messages calculated in the check nodes through an approximation rather than in their exact form. In Appendix A we review some of the most important suboptimum implementations of the SPA.

The LDPC decoder can be seen as an example of a turbo decoder with two components: a Variable Node Decoder (VND) and a Check Node Decoder (CND). Each component calculates extrinsic information about the coded bits and passes it to the other component to improve its previous decoding iteration (Figure 2.5). Note that, in the computation of a message, variable and check nodes do not make use of the input probability of the bit for which they are calculating the message (i.e., for the computation of $L_{c_i\to f_j}$, the message $L_{f_j\to c_i}$ is not used, and vice versa), so there is no need for removal of the a priori information: the messages $L_{f\to c}$ and $L_{c\to f}$ are already extrinsic information.

2.2.3 LDGM codes

As explained previously, LDPC codes have, in general, a non-sparse generator matrix. There exists a class of codes with a sparse generator matrix $\mathbf{G}$, known as Low-Density Generator



$$\mathbf{G} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \end{pmatrix}$$

Figure 2.6: Factor graph of a 4×7 systematic generator matrix

Matrix (LDGM) codes [13], that will be a matter of intense study along this work. In particular, we will consider *systematic* LDGM codes, i.e., codes whose gererator matrix have the form $\mathbf{G} = [\mathbf{I}_K|\mathbf{P}]$, where $\mathbf{I}_K$ is the identity matrix of size $K \times K$ and $\mathbf{P}$ is a $K \times L$ sparse matrix, with $L = N - K$. Since the parity check matrix of this generator matrix is given by the sparse matrix $\mathbf{H} = [\mathbf{P}^T|\mathbf{I}_L]$, systematic LDGM codes are in fact a subclass of LDPC codes.

Encoding of LDGM codes is done in the same way as with LDPC codes: the codeword vector $\mathbf{c}$ is obtained by multiplying the vector formed by the uncoded bits $\mathbf{u}$ by the generator matrix $\mathbf{G}$. Since we are considering systematic codes, the codeword can be written as

$$\mathbf{c} = [c_1, c_2, \ldots, c_N] = \mathbf{uG} = [u_1, u_2, ..., u_K, p_1, p_2, ..., p_L]$$

Due to the sparseness of the generator matrix, encoding can be done in linear time.

Decoding of LDGM codes is also carried out by applying the SPA over the factor graph representing the code. In this case the factor graph is constructed based on the generator matrix. In an LDGM factor graph there are two clearly different groups of variable nodes: one for the K uncoded bits and another for the L parity bits. Each parity bit variable node is connected to one (and only one) check node. The connections between the check nodes and the systematic variable nodes are determined by the parity matrix $\mathbf{P}$: there is an edge between a systematic variable node and a check node if there is a "1" in the corresponding row and column of the parity matrix. Figure 2.6 shows the relationship between a low dimension systematic generator matrix and a factor graph.

LDGM codes can also be non-systematic. However, in this case we must impose certain constraints in the degrees of the nodes. The reason for this is that, if the code is non-systematic, the uncoded bit variable nodes in the factor graph do not receive information from the detector (Figure 2.7), and the initial messages passed by them to the check nodes will all be equal to zero. Due to the nature of a check node, an output message calculated by it will be equal to

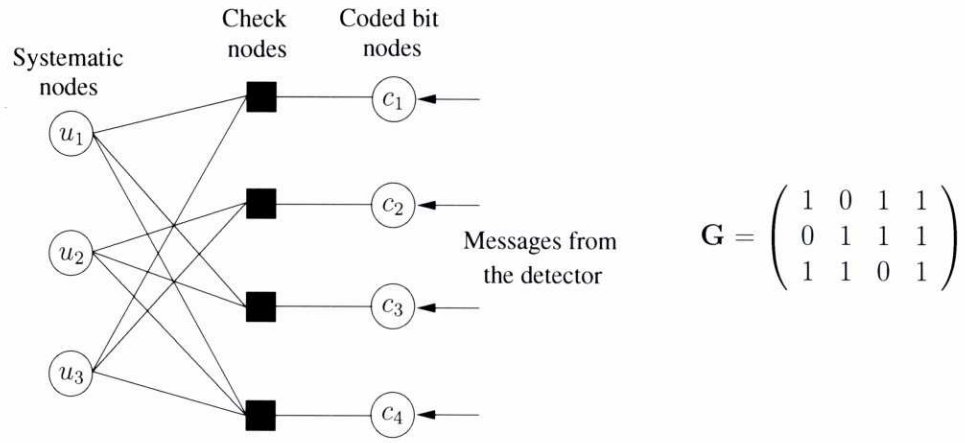


Figure 2.7: Factor graph of a 3×4 non-systematic generator matrix

zero when one of its inputs is zero, because the check is not able to provide any information about a bit if it has no information about all the other bits involved in the modulo-2 sum. If all the check nodes have a degree towards the systematic nodes greater than one, one of their inputs will always be zero, and thus the SPA will not progress from the initial state. This issue can be overcome by *doping* the code [38], which consists on ensuring that some percentage of the checks are only connected to one systematic bit node. These check nodes are now able to provide information to the variable nodes they are connected to and, if the percentage of doped check nodes is high enough, that information will be propagated to all the factor graph.

Although LDGM codes have the advantage of a linear encoding complexity, unlike general LDPC codes, LDGM codes are considered "bad" codes, since they can only attain an arbitrarily low error probability by reducing the rate to zero. This is due to their poor minimum distance [13] which is independent of the block length and causes their performance to suffer a high error floor (i.e., the bit error rate decreases very slowly as we increase the E_b/N_0). This problem has made LDGM codes to be initially disregarded in favor of LDPC codes although, as we will see in the next chapters, they can actually perform extremely well in a serially concatenated scheme.

2.2.4 Repeat-Accumulate codes

Repeat-Accumulate (RA) codes [39] are another subclass of LDPC codes with linear encoding complexity. RA codes consist of the serial concatenation of a repetition code, a bit interleaver, a set of parity checks and an accumulator. RA codes are usually systematic, so the parity-accumulated bits are appended to the uncoded bits to form the final coded sequence (Figure 2.8). Although this is the most usual representation of a RA code, the repeat, interleaving and parity check stages can also be seen as a LDGM encoder with a generator matrix $\mathbf{G} = [\mathbf{I}_K|\mathbf{P}]$, whose sparse parity matrix $\mathbf{P}$ is constructed based on the number of repetitions for each bit, that

determine the number of ones in each row of $\mathbf{P}$; the number of bits added modulo-2 in the parity checks, that determine the number of ones in each column; and the interleaver, that determine the position of the ones in the matrix. Encoding can thus be done simply by multiplying the uncoded bits by the parity matrix to obtain the parity bits

$$\mathbf{uP} = [p_1, p_2, ..., p_L]$$

and then accumulating them modulo-2 to obtain the coded bits

$$\begin{aligned} c_{K+1} &= p_1 \\ c_{K+2} &= p_1 + p_2 = c_{K+1} + p_2 \\ c_{K+3} &= p_1 + p_2 + p_3 = c_{K+2} + p_3 \\ &\vdots \\ c_N &= p_1 + p_2 + \cdots + p_L = c_{N-1} + p_L \end{aligned}$$

which are appended to the K uncoded bits $\mathbf{u}$ to form the final codeword. The objective of the accumulator is to lower the typical error floor of LDGM codes by providing the code with a means of increasing its minimum distance with the block length, making it potentially capacity-achieving.

As occurs with LDPC and LDGM, we can use a factor graph to represent RA codes and apply the SPA over it to decode it. In case the graph is irregular, the code is usually named Irregular Repeat-Accumulate (IRA) code. Since the RA code can be viewed as an LDGM code concatenated with an accumulator, the factor graph is constructed based on the generator matrix (i.e., the repetition pattern, the interleaver and the checks) with a zigzag structure for the accumulator. The zigzag structure accounts for the fact that each parity bit p_i can be calculated as

$$\begin{aligned} p_1 &= c_{K+1} \\ p_2 &= c_{K+2} + c_{K+1} \\ p_3 &= c_{K+3} + c_{K+2} \\ &\vdots \\ p_N &= c_N + c_{N-1} \end{aligned}$$

which means that the parity constraints must be obeyed by the systematic bits and two consecutive coded bits. Figure 2.9 shows the factor graph representation of a RA code.

Also, similarly to LDGM codes, RA codes can be non-systematic, in which case they require code doping in order to progress from the initial state [40].

2.3 Code optimization with EXIT charts

The performance of regular LDPC and RA codes is not very good when compared with that of turbo codes. However, if an irregular degree profile is employed, their performance can

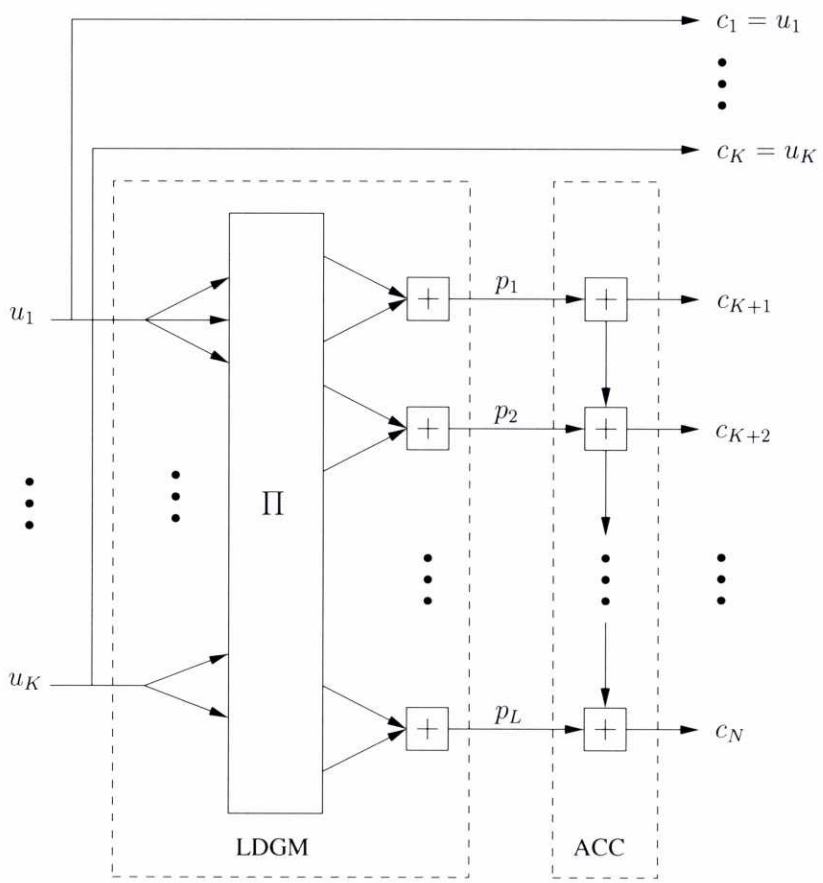


Figure 2.8: Systematic RA encoder

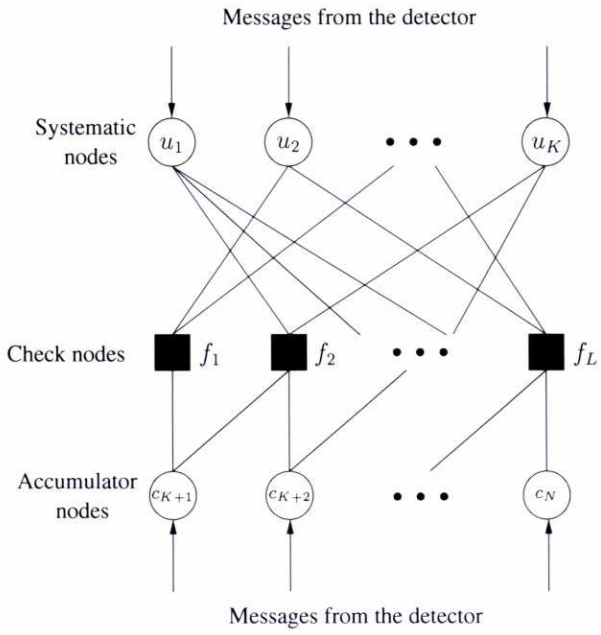


Figure 2.9: Factor graph representation of a RA code

easily surpass that of turbo codes and approach capacity. Optimization for LDPC and RA codes consists in finding the degree profile that achieves the best performance on a given channel. Two basic techniques have been proposed for the design of LDPC and RA codes: Density Evolution (DE) [27] and Extrinsic Information Transfer (EXIT) charts [41].

Density Evolution (DE) is an optimization method based on studying the variation of the probability density function (pdf) of the messages between variable and check nodes in the factor graph representing a code. The code is assumed to have infinite block length and to be randomly generated, although maintaining the restrictions imposed by its degree profile. Although the method in its original form is relatively complex, because it traces the entire variable and check messages pdf, there exist variations with lower complexity. One example is DE with Gaussian Approximation (DE-GA) [28], that approximates the densities of messages as symmetric Gaussian ones, thus tracking only the evolution of the mean. This idea is further exploited by EXIT charts to provide a graphical representation of code convergence.

The idea behind the EXIT charts is to track the extrinsic information exchanged between the components of an iterative decoder. Instead of tracking the pdf of the messages, a representative value is used to simplify the problem and effectively capture the most relevant information about the decoding process: the bitwise mutual information between the messages and the codeword [42].

2.3.1 EXIT charts for LDPC codes

EXIT charts can be applied to LDPC codes by considering that the decoder is formed by two components that exchange extrinsic information about the coded bits: the VND and the CND (see Figure 2.5). Therefore, we must calculate the mutual information $I(M;C)$ between the messages M (either $L_{c\to f}$ or $L_{f\to c}$) and the coded bits represented by them. To compute these mutual informations, we can take into account that the incoming messages to a variable node $L_{f\to c}$ play the same role as the channel LLR L_{ch} in Eq. (2.30). A reasonable assumption is to model the messages M as the output LLR of an independent binary input AWGN (BIAWGN) channel

$$m = \log \frac{p(x|s=+1)}{p(x|s=-1)} \tag{2.37}$$

where $x = s + v$, $s = 2c - 1$, and $v \sim \mathcal{N}(0, \sigma_v^2)$. Note that the noise variance σ_n^2 used in the channel model is independent and (in general) different from the variance σ_v^2. Recalling Eq. (2.36), we know that

$$m = \frac{2x}{\sigma_v^2} = \frac{2}{\sigma_v^2}(s+v) \tag{2.38}$$

so the probability distribution of M given the transmitted symbol is Gaussian with mean $2s/\sigma_v^2$ and variance $4/\sigma_v^2$. If we denote $\sigma^2 \triangleq 4/\sigma_v^2$, then the message given the symbol s follows a distribution $\mathcal{N}(s\sigma^2/2, \sigma^2)$, and the probability density function is given by

$$p(m|s) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(m - s\sigma^2/2)^2}{2\sigma^2}\right) \tag{2.39}$$

Using this model, and taking into account that the relation between C and S is deterministic, we can calculate the mutual information $I(M;C)$ as

$$\begin{aligned} I(M;C) = I(M;S) &= \mathbb{E}_{M,S}\left\{\log_2 \frac{p(m,s)}{p(m)p(s)}\right\} \\ &= \sum_{s=-1,+1} \int_{-\infty}^{+\infty} p(m,s) \log_2 \frac{p(m,s)}{p(m)p(s)}\, dm \\ &= \sum_{s=-1,+1} p(s) \int_{-\infty}^{+\infty} p(m|s) \log_2 \frac{p(m|s)}{p(m)}\, dm \end{aligned} \tag{2.40}$$

Assuming $p(S=+1) = p(S=-1) = 1/2$, we have

$$I(M;S) = \frac{1}{2} \sum_{s=-1,+1} \int_{-\infty}^{+\infty} p(m|s) \log_2 \frac{2p(m|s)}{p(m|S=-1) + p(m|S=+1)}\, dm \tag{2.41}$$

Next, let us define the function

$$\begin{aligned} L_{+1}(m) &\triangleq \log_2 \frac{2p(m|S=+1)}{p(m|S=-1) + p(m|S=+1)} = 1 - \log_2 \frac{p(m|S=-1) + p(m|S=+1)}{p(m|S=+1)} \\ &= 1 - \log_2\left(1 + \frac{p(m|S=-1)}{p(m|S=+1)}\right) = 1 - \log_2\left(1 + e^{-m}\right) \end{aligned} \tag{2.42}$$

and, equivalently

$$L_{-1}(m) \triangleq \log_2 \frac{2p(m|S=-1)}{p(m|S=-1)+p(m|S=+1)} = 1 - \log_2\left(1+e^m\right) = L_{+1}(-m) \quad (2.43)$$

Using these expressions we can rewrite Eq. (2.41) as

$$\begin{aligned} I(M;S) &= \frac{1}{2}\int_{-\infty}^{+\infty} p(m|S=+1)L_{+1}(m)\,dm + \frac{1}{2}\int_{-\infty}^{+\infty} p(m|S=-1)L_{-1}(m)\,dm \\ &= \frac{1}{2}\int_{-\infty}^{+\infty} p(m|S=+1)L_{+1}(m)\,dm + \frac{1}{2}\int_{-\infty}^{+\infty} p(-m|S=-1)L_{-1}(-m)\,dm \end{aligned}$$

Since $p(-m|S=-1) = p(m|S=+1)$ and $L_{-1}(-m) = L_{+1}(m)$, we can calculate the mutual information as

$$\begin{aligned} I(M;S) &= \int_{-\infty}^{+\infty} p(m|S=+1)\left(1-\log_2\left(1+e^{-m}\right)\right)\,dm \\ &= 1 - \int_{-\infty}^{+\infty} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(m-\sigma^2/2)^2}{2\sigma^2}\right)\log_2(1+e^{-m})\,dm \end{aligned} \quad (2.44)$$

It is important to note that this mutual information depends only on the variance σ^2, so we can define a function $J(\sigma^2) \triangleq I(M;C)$. Therefore, $J^{-1}(I)$ gives the variance σ^2 corresponding to an incoming message with mutual information I. Figure 2.10 represents the function $J(\sigma^2)$. The functions $J(\sigma^2)$ and $J^{-1}(I)$ can be numerically approximated for implementation purposes with an error lower than 10^{-3} by [41]

$$I = J(\sigma^2) = \begin{cases} a_1\sigma^3 + b_1\sigma^2 + c_1\sigma & 0 \le \sigma \le 1.6363 \\ 1 - \exp(a_2\sigma^3 + b_2\sigma^2 + c_2\sigma + d_2) & 1.6363 < \sigma \le 10 \\ 1 & \sigma \ge 10 \end{cases} \quad (2.45)$$

and

$$\sigma = \sqrt{J^{-1}(I)} = \begin{cases} a_3 I^2 + b_3 I + c_3\sqrt{I} & 0 \le I \le 0.3646 \\ -a_4\log(b_4(1-I)) - c_4 I & 0.3646 < I \le 1 \end{cases} \quad (2.46)$$

being

$$\begin{array}{llll} a_1 = -0.0421061 & b_1 = 0.209252 & c_1 = -0.00640081 & \\ a_2 = 0.00181491 & b_2 = -0.142675 & c_2 = -0.0822054 & d_2 = 0.0549608 \\ a_3 = 1.09542 & b_3 = 0.214217 & c_3 = 2.33727 & \\ a_4 = 0.706692 & b_4 = 0.386013 & c_4 = -1.75017 & \end{array} \quad (2.47)$$

These functions allow us to efficiently compute the information exchanged between the VND and the CND. Indeed, let us first focus on the VND. From Eq. (2.30) we know that the output message from a variable node i with degree d_v to a check node j is simply the sum of the $d_v - 1$ messages passed from the check nodes different than j plus the message from

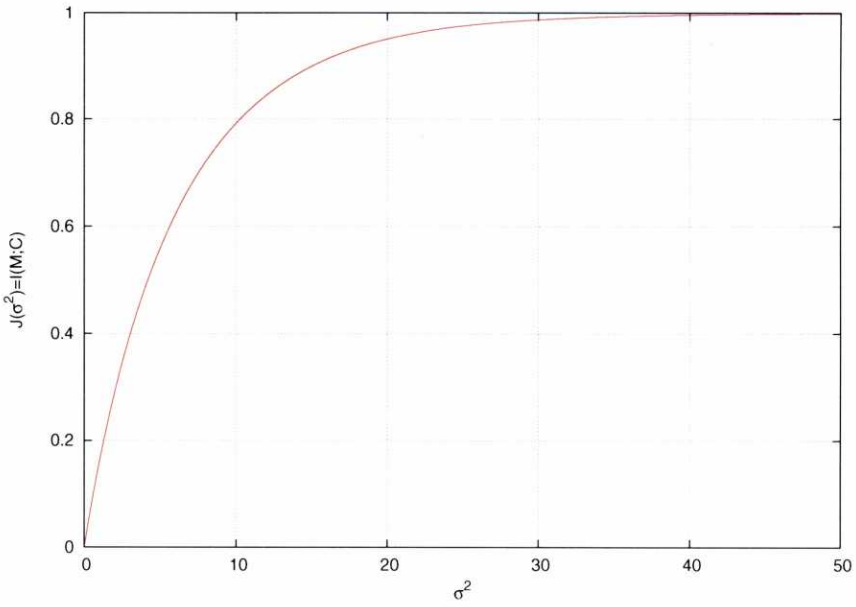


Figure 2.10: Mutual information as a function of the message variance in a BIAWGN channel

the detector. Since we are assuming an independent Gaussian model for the messages, this is equivalent to saying that the variance σ^2 of the output message is the sum of the $d_v - 1$ variances of the input messages from the check nodes plus the variance of the message from the detector. If we assume that all the messages passed from the CND carry an information equal to $I_{A,\mathrm{VND}}$, and we denote the information from the detector as I_{ch}, we can obtain the respective message variances as $J^{-1}(I_{A,\mathrm{VND}})$ and $J^{-1}(I_{\mathrm{ch}})$. Thus, the output information from a variable node of degree d_v (or a set of them) is given by

$$I_{E,\mathrm{VND}}(I_{A,\mathrm{VND}}, I_{\mathrm{ch}}, d_v) = J\left((d_v - 1)J^{-1}(I_{A,\mathrm{VND}}) + J^{-1}(I_{\mathrm{ch}})\right). \tag{2.48}$$

The value $J^{-1}(I_{\mathrm{ch}})$ depends on the E_b/N_0 of the channel. The greater the E_b/N_0 we are considering, the greater the information that comes from the detector, and thus the greater the variance of the associated messages. For example, if we assume a BIAWGN channel such as the one in (2.37), we have $E_s = R_c E_b = 1$. The variance of the messages coming from the detector can thus be easily calculated as

$$J^{-1}(I_{\mathrm{ch}}) = \sigma_{\mathrm{ch}}^2 = \frac{4}{\sigma_n^2} = \frac{8}{N_0} = 8R_c\frac{E_b}{N_0} \tag{2.49}$$

For more complex channels or constellations, the information passed by the detector must be estimated for each E_b/N_0 via Monte Carlo simulations.

In case the code is irregular, and the variable nodes have D_v different degrees $d_{v,1}, \ldots, d_{v,D_v}$, the mutual information must be calculated for every node degree, and then averaged by the

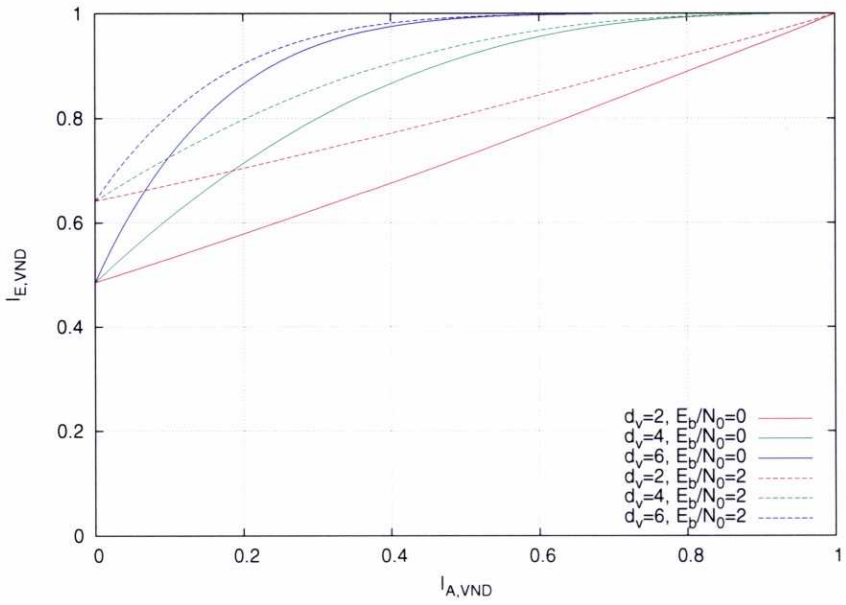


Figure 2.11: EXIT curves for regular Variable Node Decoders (VND) with different degrees in a BIAWGN channel.

number of edges carrying each amount of information. Thus, if we denote $a_{v,i}$ the fraction of nodes having degree $d_{v,i}$, we can calculate the fraction of edges outgoing from the nodes with that degree as

$$b_{v,i} = \frac{a_{v,i} d_{v,i}}{\sum_{j=1}^{D_v} a_{v,j} d_{v,j}} \tag{2.50}$$

and the average mutual information passed to the check nodes is

$$I_{E,\mathrm{VND}}(I_{A,\mathrm{VND}}, I_{\mathrm{ch}}) = \sum_{j=1}^{D_v} b_{v,j} I_{E,\mathrm{VND}}(I_{A,\mathrm{VND}}, I_{\mathrm{ch}}, d_{v,j}) \tag{2.51}$$

Figure 2.11 shows the EXIT curves for regular variable node decoders with different degrees in an BIAWGN channel. We can see that increasing the node degree also increases the amount of extrinsic information provided by the VND.

When computing the mutual information of the check-to-variable messages, it is convenient to resort to a duality property of the Binary Erasure Channel (BEC). A BEC is a channel that, with probabilty p, "erases" the transmitted bit in a way that the receiver cannot recover it, and with probability $1-p$ leaves the bit untouched (Figure 2.12). In this channel, the output mutual information of a check with degree d_c is related to that of a variable node of degree d_c as follows [43]

$$I_{E,\mathrm{CND}}(I_{A,\mathrm{CND}}, d_c) = 1 - I_{E,\mathrm{VND}}(1 - I_{A,\mathrm{CND}}, 0, d_c) \tag{2.52}$$

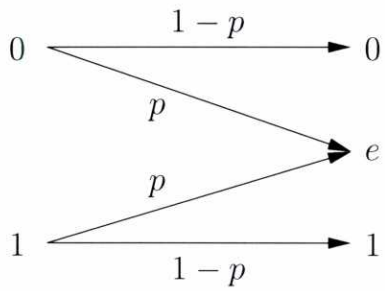


Figure 2.12: Binary erasure channel model with erasure probability p

This property is very accurate (although not exact) for a BIAWGN channel, so we can approximate the extrinsic messages from a CND with degree d_c by substituting (2.48) in (2.52) as

$$I_{E,\mathrm{CND}}(I_{A,\mathrm{CND}}, d_c) \approx 1 - J((d_c - 1)J^{-1}(1 - I_{A,\mathrm{CND}})) \tag{2.53}$$

In case there are different check degrees, the output information must be averaged in the number of edges corresponding to each degree, as in the case of variable nodes. Figure 2.13 shows the EXIT curves for regular check node decoders with different degrees. Contrarily to what occurs at the variable nodes, increasing the check node degree causes a decrease in the extrinsic information provided by the CND.

When, as occurs with LDPC codes, the decoder is constituted by two components, the EXIT curves can show in a graphical way if the code will be able to converge to the correct solution for a given E_b/N_0. To do this, the extrinsic information produced by one of the decoding components is plotted with the horizontal axis as the input information and the vertical as the output, whereas for the other decoding component the axes are swapped (the vertical axis represents the input and the horizontal axis the output). This way we can plot the trajectory followed by the mutual information in the decoding process as a zigzag line, since the output information from one of the components is the input information for the other. If there exists a "tunnel" between the curves corresponding to each component that reaches the point $(1, 1)$, the decoder will be able to converge to the correct solution. If the curves intersect before that point, the exchange of extrinsic information will get "stuck" before achieving the correct solution.

For example, let us consider a regular LDPC code of rate $R_c = 0.5$ and a degree for the variable nodes of $d_v = 4$. Since $N = K/R_c = 2K$, the number of checks is $N - K = N/2$, and thus the degree of the checks in the factor graph is $d_c = 8$ (an edge always connect one variable node to one check node). Figure 2.14 shows the EXIT curves for the VND and CND corresponding to this code in a BIAWGN channel for $E_b/N_0 = 1.5$dB and for $E_b/N_0 = 1.6$dB. Note that the horizontal and vertical axis are swapped for the CND curves. For the case of $E_b/N_0 = 1.6$dB, the mutual information between the coded bits and the messages passed by the variable nodes to the check nodes reaches one, which means that we can recover the transmitted coded bits by observing the variable node messages. For $E_b/N_0 = 1.5$dB, however, the mutual

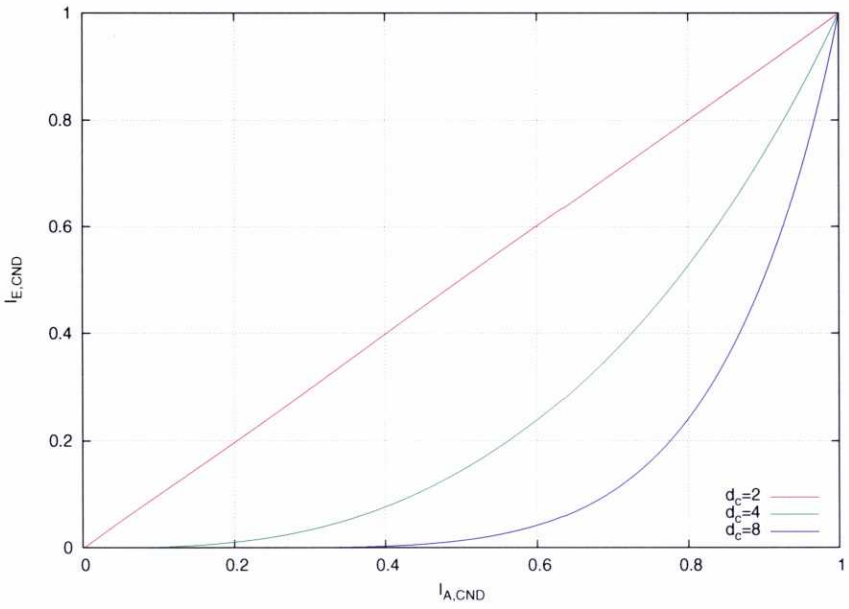


Figure 2.13: EXIT curves for regular Check Node Decoders (CND) with different degrees.

information exchanged between the two components gets stuck at the point where the curves intersect.

EXIT charts allow an easy understanding of the behavior of iterative decoding, and make it easy to find codes that are able to converge close to the capacity limit. In the case of the previous code, its predicted convergence threshold in a BIAWGN channel would be between 1.5dB and 1.6dB. Since the AWGN capacity limit for 0.5 bits per channel use is at $E_b/N_0 = 0$dB, the code would perform at more than 1.5dB away from the capacity limit. By studying the performance of codes with irregular degree profiles it is possible to get much closer to the limits. In fact, it can be shown that the optimization of a LDPC code can be reduced to a curve fitting problem [43]. For example, we can choose a fixed degree for the CND and search for an irregular degree profile for the VND in order that its curve lies as close as possible to the CND without intersecting. Figure 2.15 shows the EXIT chart of an optimized LDPC code with an irregular variable node degree profile of [41]

$$d_v = \{2, 4, 18\} \quad a_v = \{0.508, 0.419, 0.073\} \tag{2.54}$$

and a degree for all the check nodes of $d_c = 8$. As we can see, this code is able to converge at only 0.5dB away from the AWGN capacity limit.

It is important to note that this is the estimated average mutual information trajectory, and not the actual decoding trajectory for an LDPC code with a finite block length. EXIT charts assume independence between the different messages passed in the factor graph, which only happens when the block length tends to infinity. As the block length decreases, EXIT charts

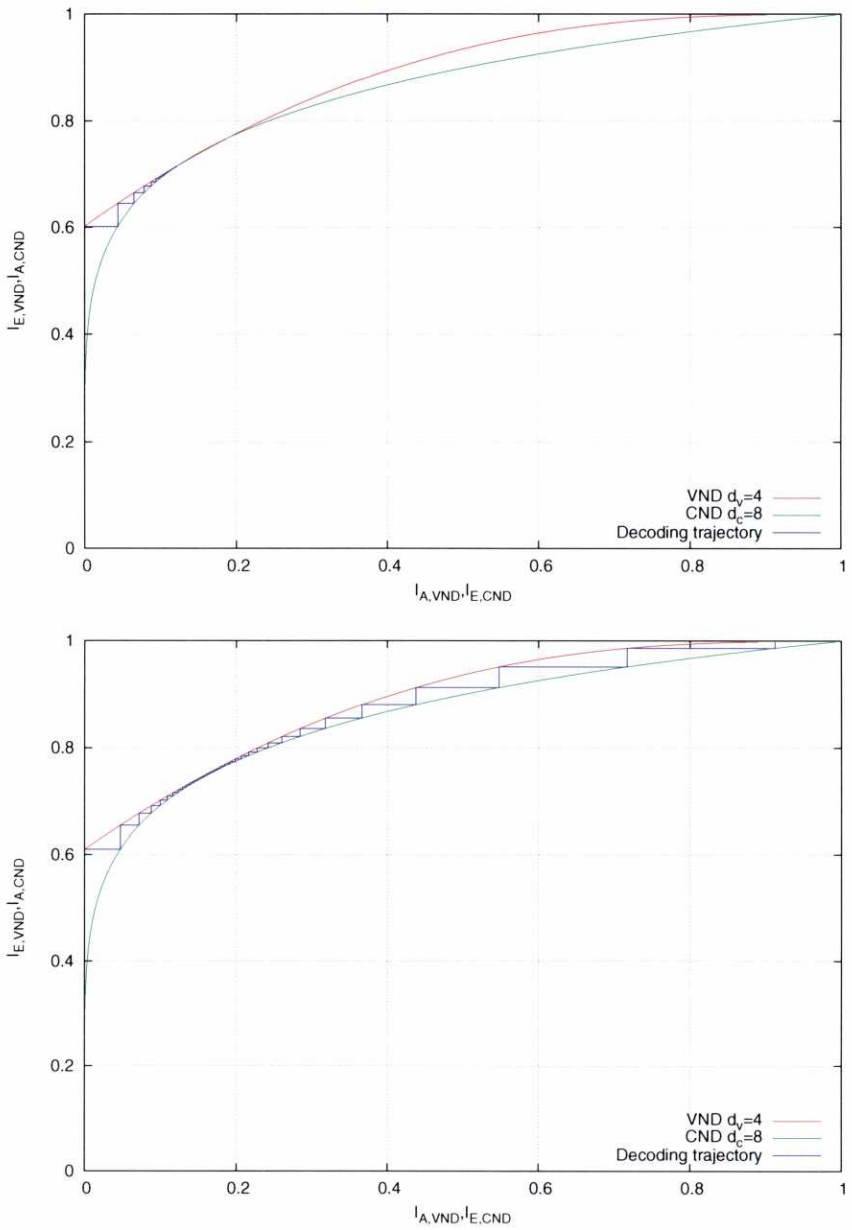


Figure 2.14: EXIT chart of a regular LDPC code with rate $R_c = 0.5$ over a BIAWGN channel, for $E_b/N_0 = 1.5$dB (top) and $E_b/N_0 = 1.6$dB (bottom)

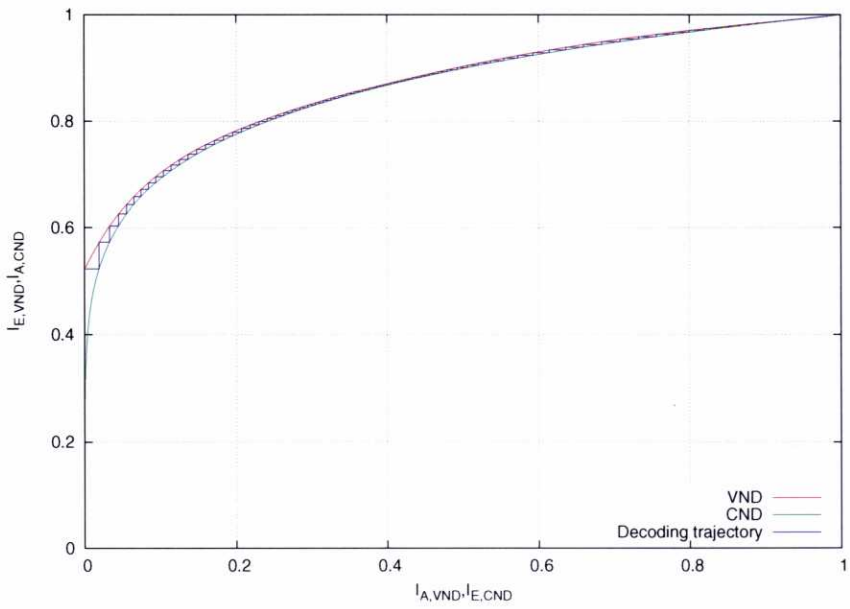


Figure 2.15: EXIT chart of an optimized irregular LDPC code with rate $R_c = 0.5$ over a BIAWGN channel, for $E_b/N_0 = 0.5$dB

become less accurate, and the actual performance of the optimized codes gets worse when compared with the predicted behavior.

2.3.2 EXIT charts for LDGM codes

LDGM codes behavior can also be studied using EXIT charts by considering the exchange of mutual information between the systematic variable nodes, that constitute now the VND, and the check nodes. The function of the parity variable nodes is simply to pass the messages from the channel to the check nodes, so we can calculate the output information from the check nodes with degree d_c as

$$I_{E,\mathrm{CND}}(I_{A,\mathrm{CND}}, I_{\mathrm{ch}}, d_c) \approx 1 - J((d_c - 1)J^{-1}(1 - I_{A,\mathrm{CND}}) - J^{-1}(1 - I_{\mathrm{ch}})) \tag{2.55}$$

whereas the mutual information of the messages passed by the VND to the CND is the same as in the case of LDPC codes, given by (2.48).

Figure 2.16 shows the EXIT function of LDGM CNDs with different degrees and for different values of E_b/N_0. This EXIT function shows the limitation of LDGM codes: the mutual information from the CND does not reach one for an input information of $I_{A,\mathrm{CND}} = 1$, because the maximum information that the check nodes can provide to the variable nodes is limited by the information received by the check nodes from the channel (if the information from one of the bits involved in a modulo-2 sum is innacurate, we cannot obtain accurate information about the

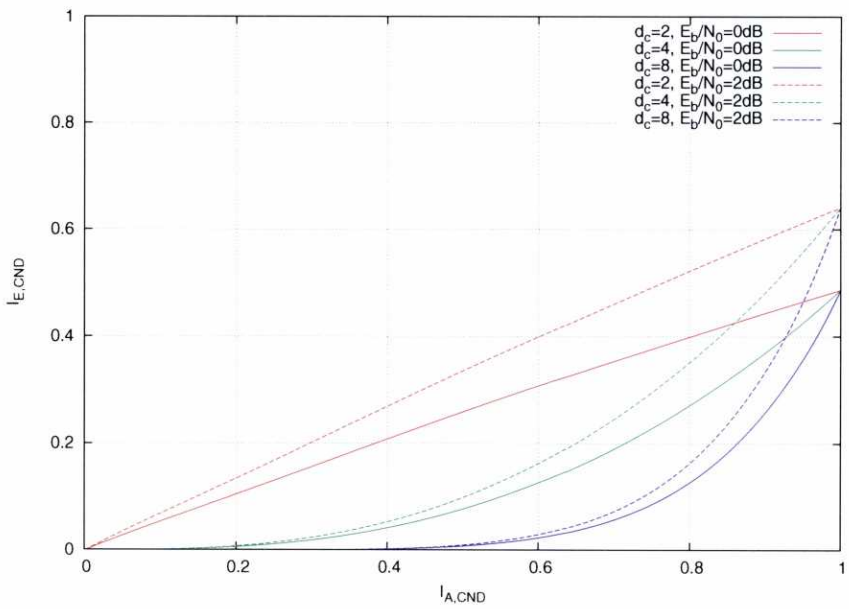


Figure 2.16: EXIT curves for regular LDGM Check Node Decoders (CND) with different degrees in a BIAWGN channel.

rest of the bits). This causes the exchange of the mutual information between the CND and the VND to get stuck for any practical degree profile, because the VND needs an input information of $I_{A,\mathrm{VND}} = I_{E,\mathrm{CND}} = 1$ to be able to produce an output information of $I_{E,\mathrm{VND}} = 1$. Note that LDPC codes do not suffer from this limitation, because their CND does not receive information from the channel directly.

For example, let us consider a regular LDGM code of rate $R_c = 1/2$ and a degree for the systematic variable nodes of $d_v = 4$. Since the number of systematic variable nodes is K, and the number of check nodes is $N - K = K$, the degree of the check nodes is also $d_c = 4$. Figure 2.17 shows the decoding trajectory of this code in a BIAWGN channel at $E_b/N_0 = 2$dB. We can see that the exchange of information stops approximately at the point $(0.58, 0.96)$, which means that a significative percentage of the information bits will be erroneous at the end of the decoding process.

An exception to this behavior is that of very high rate LDGM codes [44]. For high rates, the capacity limit is at higher values of E_b/N_0, and since the information calculated by the detector in a BIAWGN channel is proportional to the rate and the E_b/N_0 (see Eq. (2.49)), the CND is able to get very close to $I_{E,\mathrm{CND}} = 1$, and thus systematic LDGM codes are able to converge to a solution near the capacity limit, even with regular degree profiles. For example, let us consider a rate $R_c = 0.95$ regular LDGM code with a variable node degree of $d_v = 4$. The number of

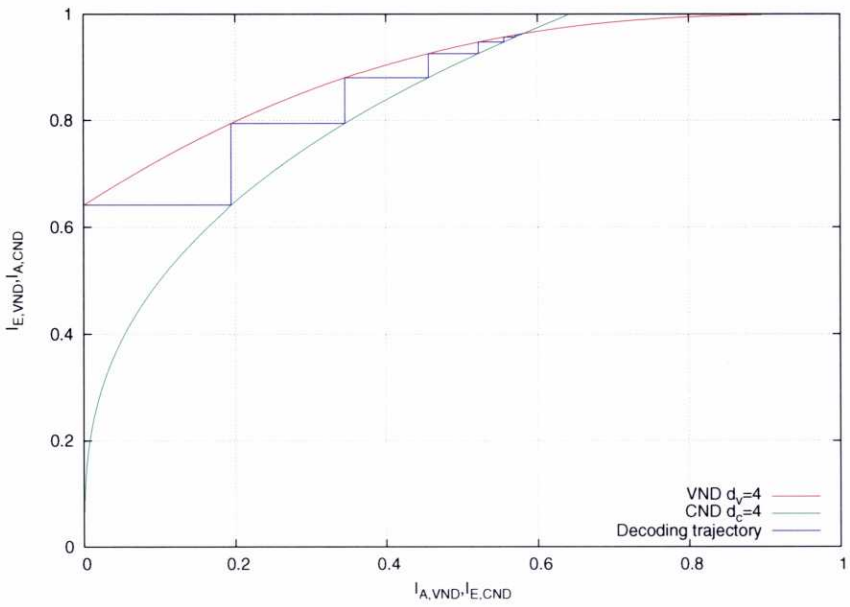


Figure 2.17: EXIT chart of a regular LDGM code with rate $R_c = 0.5$ over a BIAWGN channel, for $E_b/N_0 = 2$dB

check nodes is $N - K = K/R_c - K$, which means that the check node degree is

$$d_c = \frac{d_v K}{K/R_c - K} = \frac{4}{1/0.95 - 1} = 76$$

Figure 2.18 shows the EXIT chart of this code in a BIAWGN channel for $E_b/N_0 = 4.6$dB. As we can see, the mutual information reaches one, which means that the LDGM code will coverge to the solution. In the next chapter we will take advantage of this behavior to design a scheme based of LDGM codes that is able to approach the capacity limits with medium rates.

2.3.3 EXIT charts for RA codes

When analyzing the performance of a RA code with a factor graph as that in Figure 2.9, the two considered components are the variable node decoder (VND) constituted by the systematic variable nodes, and the union of the check node and the accumulator decoders (CNAD). The output information from the VND is given by (2.48). The information passed by the CNAD to the VND is calculated in a similar way as in the LDPC case, but we must add the information passed from the accumulator to the check nodes, that we will denote as $I_{E,\mathrm{ACC}}$. Since this information arrives to the check nodes through two edges in each check, the information from the checks to the systematic variable nodes is given by

$$I_{E,\mathrm{CNAD}}(I_{A,\mathrm{CNAD}}, I_{E,\mathrm{ACC}}, d_c) = 1 - J\left((d_c - 1)J^{-1}(1 - I_{A,\mathrm{CNAD}}) + 2J^{-1}(1 - I_{E,\mathrm{ACC}})\right)$$

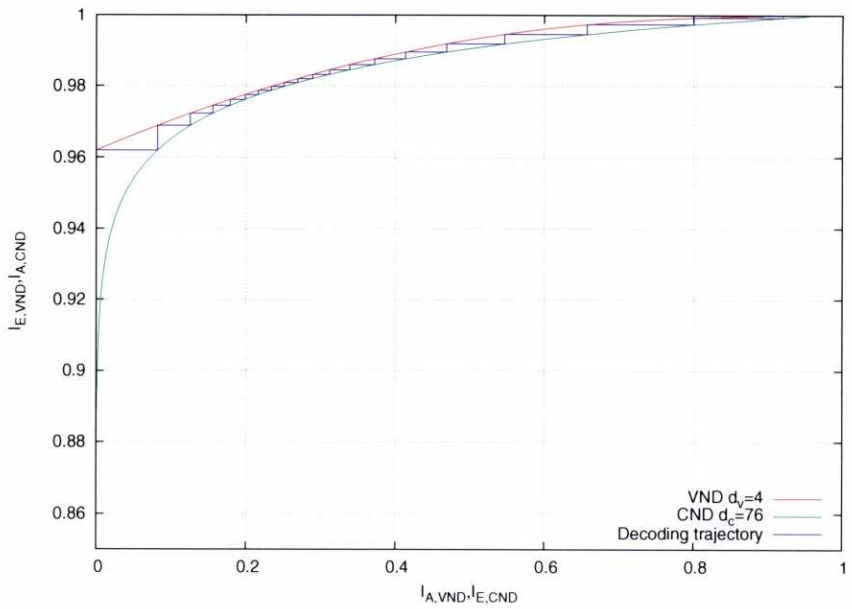


Figure 2.18: EXIT chart of a regular LDGM code with rate $R_c = 0.95$ over a BIAWGN channel, for $E_b/N_0 = 4.6$dB

The information $I_{E,\mathrm{ACC}}$ can be calculated by particularizing (2.48) to the degree-2 variable nodes in the accumulator as

$$I_{E,\mathrm{ACC}}(I_{A,\mathrm{ACC}}, I_{\mathrm{ch}}) = J\left(J^{-1}(I_{A,\mathrm{ACC}}) + J^{-1}(I_{\mathrm{ch}})\right) \tag{2.56}$$

where $I_{A,\mathrm{ACC}}$ is the input information to the accumulator passed from the check nodes, computed as

$$I_{A,\mathrm{ACC}}(I_{A,\mathrm{CNAD}}, I_{E,\mathrm{ACC}}) = 1 - J\left(d_c J^{-1}(1 - I_{A,\mathrm{CNAD}}) + J^{-1}(1 - I_{E,\mathrm{ACC}})\right) \tag{2.57}$$

As occurs with LDGM codes, the two component curves depend on the information passed by the detector (I_{ch}). Since the computation of $I_{E,\mathrm{ACC}}$ is recursive (i.e., it depends on $I_{A,\mathrm{ACC}}$, which depends on $I_{E,\mathrm{ACC}}$), we must perform several inner iterations in the CNAD to obtain the final value for $I_{E,\mathrm{CNAD}}$.

Optimization of IRA codes can be performed in the same way as that of LDPC codes: we fix the degree of the check nodes d_c and find a good irregular degree profile for the systematic variable nodes by fitting the VND curve to the fixed CND curve. Figure 2.19 shows the EXIT charts of an optimized systematic IRA code [40] over an AWGN channel for $E_b/N_0 = 0.5$dB. The degree profile of the IRA code is

$$d_v = \{2, 3, 13\} \quad a_v = \{0.0603, 0.6307, 0.3063\} \tag{2.58}$$

with a degree for all the check nodes of $d_c = 6$. Since the curves do not intersect, the predicted threshold for this code is lower or equal than $E_b/N_0 = 0.5$dB.

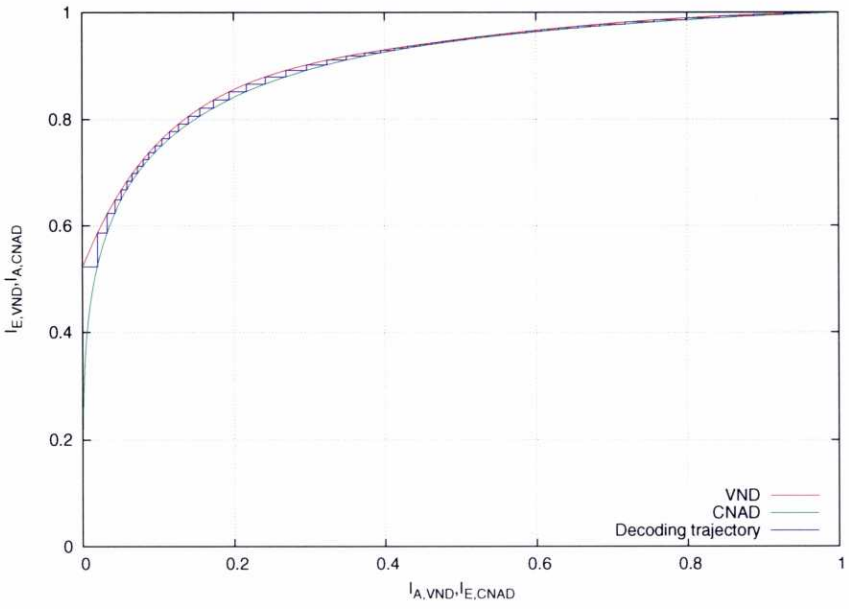


Figure 2.19: EXIT chart of an optimized IRA code with rate $R_c = 0.5$ over a BIAWGN channel, for $E_b/N_0 = 0.5$dB

2.4 Simulation results

In this section we perform computer simulations to assess the accuracy of the EXIT chart analysis for codes based in graphs.

We will start by analyzing the performance of the regular and irregular LDPC codes considered in the previous section. To avoid the high complexity of the encoding process, the simulations are performed by sending the all-zero codeword (since LDPC codes are algebraic block codes, the all-zero sequence is always a valid codeword and has the same error probability as any other codeword). The EXIT charts predict a E_b/N_0 threshold of approximately 1.6dB for the regular LDPC and 0.5dB for the irregular one (the Shannon limit is located at 0dB). Figure 2.20 shows the performance of these codes for a code length of $N = 200000$ bits. The simulated threshold (measured at a BER of 10^{-4}) is in both cases at less than 0.1dB from the predicted one. As mentioned in Section 2.3.1, the difference between the predicted and the real performance can be explained because EXIT chart analysis assume infinite block lengths. As we decrease the block length, we can expect the performance to get further away from the predicted threshold, due to the higher variance of the mutual information of the messages and the correlation between the coded bits. This can be clearly seen in Figure 2.21, that shows the performance of the optimized LDPC code over an AWGN channel for different block lengths. This figure also shows another limitation of the EXIT analysis: the appearance, even with long block lengths, of relatively high error floors (at around 10^{-5}) that cannot be predicted by the

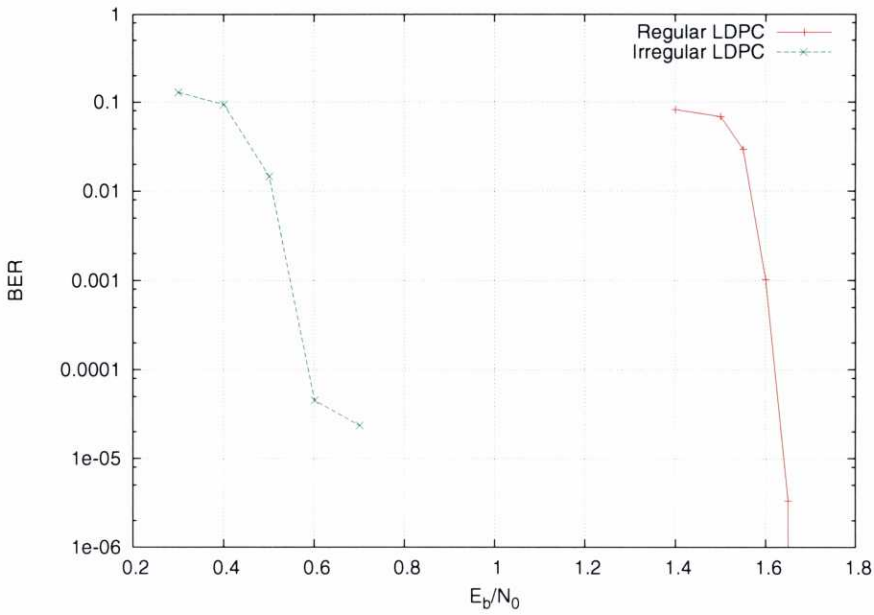


Figure 2.20: Performance of rate $R_c = 0.5$ regular and irregular LDPC codes over a BIAWGN channel, $K = 100000$ information bits.

EXIT chart. These error floors can be caused by the high percentage of degree-2 variable nodes (more than 50%), which makes the code minimum distance poor due to the high probability of short cycles involving these nodes [45, 40].

Figure 2.22 shows the performance of different systematic LDGM codes in the same BIAWGN channel. The high error floors due to the poor minimum distance prevents this class of codes to be used in practical schemes as a standalone channel code. Note that these error floors have a different nature from that of the LDPC code in the previous simulation, since these are much higher, independent of the block length and clearly predicted by the EXIT charts with an intersection between the VND and CND curves.

Finally, we can see in Figure 2.23 the performance of the optimized IRA code considered in Section 2.3.3. As occurs with LDPC codes, the simulated threshold gets closer to the predicted one as we increase the code block length. This optimized IRA code also presents an error floor, but much lower than that of the previous LDPC for practical block lengths.

2.5 Conclusions

In this chapter we have reviewed the concept of channel capacity and presented some of the coding schemes that are able to approach the capacity limits. After introducing turbo codes and the turbo principle, we studied Low Density Parity Check (LDPC) codes, a class of random

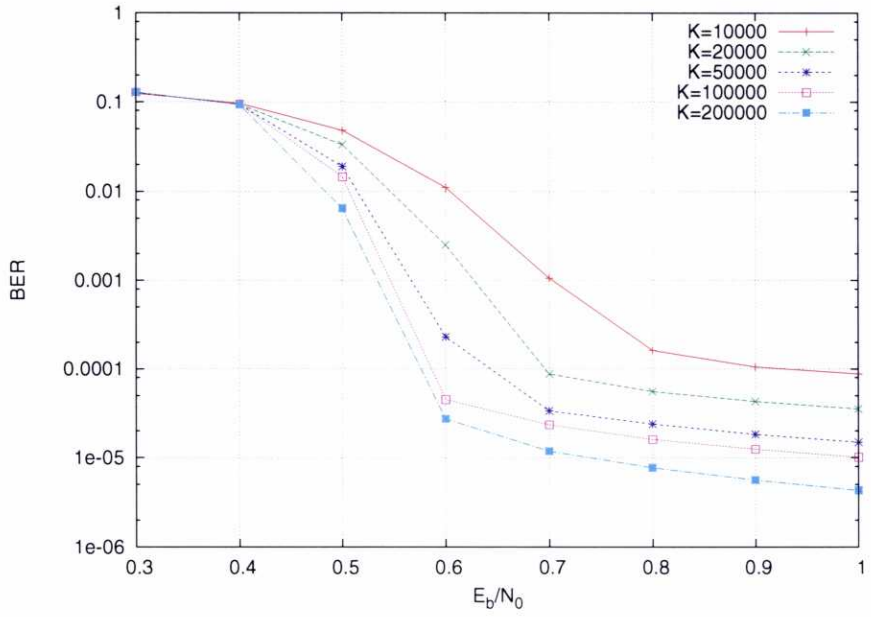


Figure 2.21: Performance of a rate $R_c = 0.5$ optimized LDPC code over a BIAWGN channel, for different block lengths.

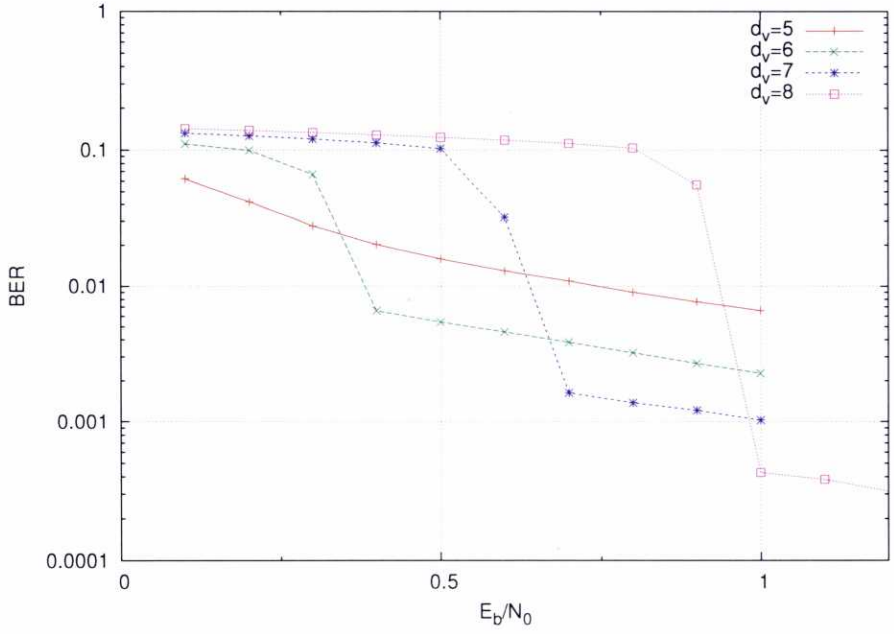


Figure 2.22: Performance of rate $R_c = 0.5$ regular LDGM codes over a BIAWGN channel, for different variable node degrees. $K = 100000$ information bits

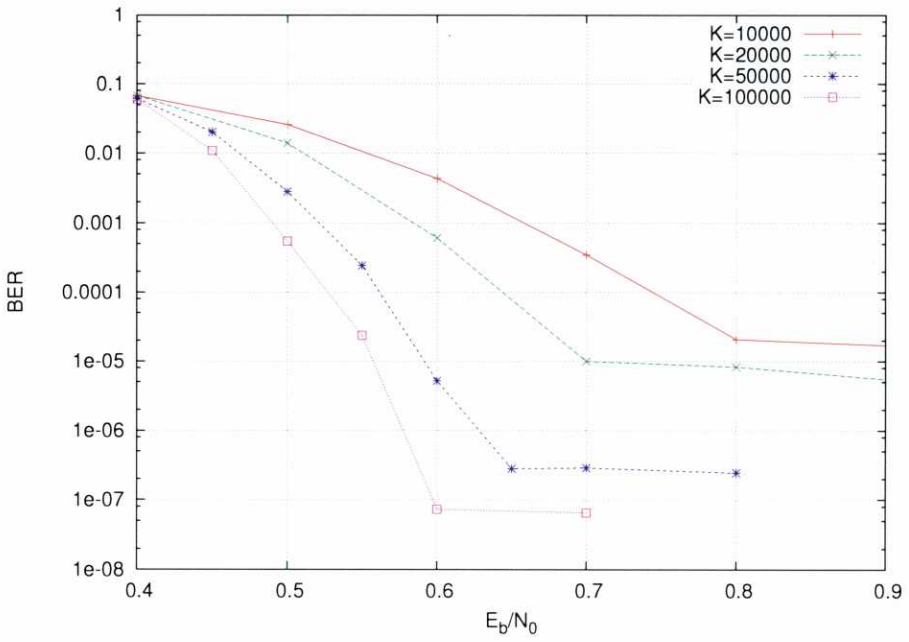


Figure 2.23: Performance of a rate $R_c = 0.5$ optimized IRA code over a BIAWGN channel, for different block lengths.

block codes that are asymptotically "very good". Although optimum decoding of LDPC codes is infeasible, we can use the Sum-Product Algorithm (SPA) over a factor graph determined by the parity-check matrix of the code. The SPA is an iterative message-passing algorithm that exchanges extrinsic information between variable and check nodes, in a similar way as the components of a turbo decoder. In addition to LDPC codes, we also studied systematic Low Density Generator Matrix (LDGM) codes, a subclass of LDPC codes with linear encoding complexity that have been historically disregarded due to their poor minimum distance; and Repeat-Accumulate (RA) codes, another subclass of LDPC codes that also have a lower encoding complexity than general LDPC codes but are still asymptotically "very good". Since both LDGM and IRA codes are subclasses of LDPC codes, they can be decoded by applying the same Sum-Product algorithm over their respective factor graphs.

In order to achieve a good performance with any of these LDPC-based codes, it is necessary to obtain an appropriate degree profile for their factor graphs. We have presented the two optimization techniques that can be used to find codes with a performance close to the capacity limit: Density Evolution (DE) and EXtrinsic Information Transfer (EXIT) charts. We focused on EXIT charts due to their lower complexity and their potential to graphically show the trajectories followed by mutual information during the decoding process. We explained how to easily calculate the extrinsic information of variable and check nodes messages in LDPC, LDGM and IRA codes, which allows us to study their performance without having to resort to

very time-consuming computer simulations. We have shown that both LDPC and IRA codes are easily capable of performing very close to the Binary-Input AWGN (BIAWGN) capacity limit when properly optimized. LDGM codes, on the other hand, cannot approach capacity with an arbitrarily low error probability because of their graph structure, so they are not suitable as standalone codes. However, their good performance in high rates and good threshold make them attractive in concatenated schemes, as we will show in the next chapter.

Finally, to assess the accuracy of the EXIT chart analysis, we performed several computer simulations testing the different classes of codes. We showed that EXIT charts predict the convergence threshold of moderate length LDPC and IRA codes in BIAWGN channels with an error lower than 0.1dB.

Chapter 3

Serially-Concatenated Low-Density Generator Matrix (SCLDGM) codes

Although decoding of LDPC codes can be done at a reasonable complexity using easily parallelizable algorithms, encoding of LDPC codes is a more involved operation. On the contrary, encoding is extremely simple in Low Density Generator Matrix (LDGM) codes. Since LDGM codes are a particular case of LDPC codes, the decoding algorithm is the same. LDGM codes have limited applicability because, in general, they present an unacceptably high error floor, although for high-rates this error-floor may already be sufficiently low [44]. On the other hand, LDGM codes present very good convergence thresholds, i.e., the E_b/N_0 abscissas at which the BER curves start to fall down is rather low. Thus, LDGM codes could be very attractive if we manage to reduce the error floor below a certain level.

One way of reducing the error floor is to concatenate an LDGM code with an outer code [46, 47]. This idea is motivated by the fact that errors in a single (non-concatenated) LDGM code occur in almost every block but the number of errors in each block is rather low. As a consequence, a high-rate outer code is enough to remove an important number of these errors. Since the performance of LDGM codes is particularly very good for high rates, it is natural to consider them as outer codes thus resulting in a very homogeneus coding scheme that we will denote as Serially-Concatenated LDGM (SCLDGM).

SCLDGM codes share some similarities with Repeat Accumulate (RA) codes. RA codes are also constructed from a core LDGM code, but reduce the error-floor by appending an accumulator. In SCLDGM codes the error floor is lowered by concatenating a high-rate LDGM code. An important advantage of SCLDGM codes is that they randomly spread information bits along the output codeword. As a consequence, SCLDGM codes are able to correct burst errors by themselves without resorting to an interleaver prior to transmission. For this reason, SCLDGM codes are very attractive for the transmission over Rayleigh fading channels and for the construction of coded modulation systems. This is not the case for RA codes where the presence of an accumulator introduces correlation in the output codewords and an interleaver may be necessary to eliminate it [40]. Another difference is that the performance of regular

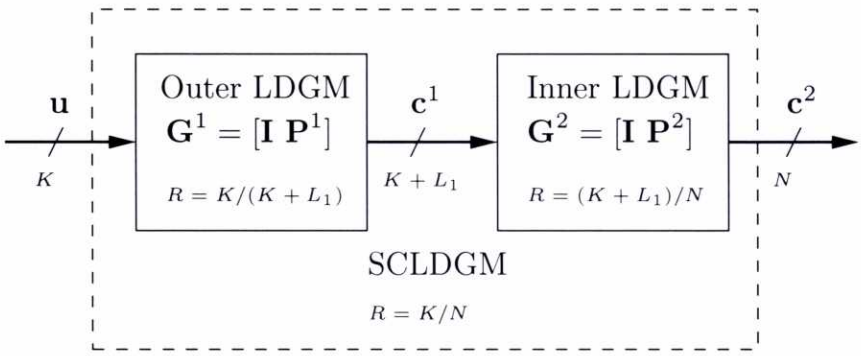


Figure 3.1: Block diagram of a Serially-Concatenated LDGM Code

SCLDGM codes is, unlike that of regular RA codes, rather good and makes unnecessary resorting to irregular degree profiles as in IRA codes.

3.1 Encoding and Decoding of SCLDGM codes

Figure 3.1 shows the block diagram of a SCLDGM code. Let us consider a sequence of K information bits

$$\mathbf{u} = [u_1, u_2, ..., u_K]$$

This sequence is first encoded with an outer systematic LDGM code to produce the coded sequence

$$\mathbf{c}^1 = \mathbf{u}\mathbf{G}^1 = [u_1, u_2, ..., u_K, p_1^1, p_2^1, ..., p_{L_1}^1]$$

where L_1 is the number of parity bits added by the outer code, $\mathbf{G}^1 = [\mathbf{I}_K|\mathbf{P}_{K\times L_1}^1]$ is the outer generator matrix and $\mathbf{P}_{K\times L_1}^1$ is the outer parity matrix. The rate of the outer code is thus $R^1 = K/(K + L_1)$. These bits are then encoded with an inner LDGM code to produce the final codeword, i.e.,

$$\mathbf{c} = [c_1, c_2, ..., c_N] \triangleq \mathbf{c}^2 = \mathbf{c}^1\mathbf{G}^2 = [u_1, u_2, ..., u_K, p_1^1, p_2^1, ..., p_{L_1}^1, p_1^2, p_2^2, ..., p_{L_2}^2]$$

where L_2 is the number of parity bits introduced by the inner code, $N = K + L_1 + L_2$ is the codeword length, $\mathbf{G}^2 = [\mathbf{I}_{K+L_1}|\mathbf{P}_{(K+L_1)\times L_2}^2]$ is the inner generator matrix and $\mathbf{P}_{(K+L_1)\times L_2}^2$ is the inner parity matrix. The rate of the inner code is $R^2 = (K + L_1)/(K + L_1 + L_2)$, and the overall code rate is

$$R_c = R^1R^2 = \frac{K}{K + L_1}\frac{K + L_1}{K + L_1 + L_2} = \frac{K}{K + L_1 + L_2} = \frac{K}{N}.$$

As mentioned before, the inner code rate is close to the overall code rate and the outer code rate is close to one. This is accomplished by choosing $L_1 \ll L_2$, since in this case $R^1 \approx 1$ and $R^2 \approx R_c$.

Fig. 3.2 shows the factor graph of a SCLDGM code. When specifying an SCLDGM code we have to take into account the presence of different types of variable and check nodes: variable nodes can be u_j (systematic bits), p_j^1 (outer parity bits) or p_j^2 (inner parity bits), while check nodes can be either f_j^1 (outer checks) or f_j^2 (inner checks). Degrees have now to be defined for each pair of node types. Let us denote by $d_{v,i}^f$, $i = 1, 2, ..., D_v^f$, the degrees of variable nodes of type v with respect to check nodes of type f. Since there is a one-to-one correspondence between check nodes and their related parity bit nodes, we therefore have only degrees $d_{u,i}^{f^1}$, $d_{u,i}^{f^2}$ and $d_{p^1,i}^{f^2}$. The number of different degrees is, respectively, $D_u^{f^1}$, $D_u^{f^2}$ and $D_{p^1}^{f^2}$. To complete the specification of the degree profile we will denote the fraction of nodes of type v having degree $d_{v,i}^f$ as $a_{v,i}^f$, being v either u or p^1. The number of edges from variable nodes of type v to check nodes of type f is thus given by $e_v^f = \sum_{i=1}^{D_v^f} d_{v,i}^f a_{v,i}^f$. These edges are assigned to the corresponding check nodes uniformly, as in simple LDGM codes, so the degree of check nodes f with respect to variable nodes v is either d_f^v or $d_f^v + 1$. Finally, another code parameter is $p = L_1/(L_1 + L_2)$, i.e., the fraction of outer parity bits with respect to the total number of parity bits. We will say that an SCLDGM code is *regular* when all $D_u^{f^1} = D_u^{f^2} = D_{p^1}^{f^2} = 1$, and irregular otherwise. In the sequel we will drop the subindex i whenever the number of degrees is one.

Decoding of SCLDGM codes is performed by the particularization of Eqs. (2.30) and (2.32) to the corresponding factor graph. At the end of the iterative process, a decision is made using the *a posteriori* L-values of bits $\mathbf{u}$ given by equation (2.35).

It is worth mentioning that, given the two generator matrices $\mathbf{G}^1$ and $\mathbf{G}^2$, we can obtain an equivalent generator matrix $\mathbf{G} = \mathbf{G}^1\mathbf{G}^2$. This new generator matrix is also sparse, so we can consider building a simpler factor graph based on this matrix, with only one layer of variable nodes and one layer of check nodes. However, using this factor graph to decode the received sequences leads to a very poor performance [48]. In fact, the same high error floor can be observed as when decoding a single LDGM code.

3.2 Channel model

In this chapter we will assume that the coded bits $\mathbf{c}$ are mapped to a complex constellation with 2^{M_c} levels (i.e., M_c bits per symbol) and average energy E_s for the transmission through the channel. This is an example of Bit-Interleaved Coded Modulation (BICM) [49, 50], a signalling scheme in which the channel code and bit-to-symbol mapping are designed independently. As an alternative, coding and modulation can be jointly designed, as occurs in Trellis-Coded Modulation (TCM) [51]. However, designing long random codes together with the constellation is a difficult task, and BICM can provide results very close to the capacity limit without the need of a joint design, as we will see in the following sections.

We will consider two channel models: Additive White Gaussian Noise (AWGN) and Rayleigh fading.

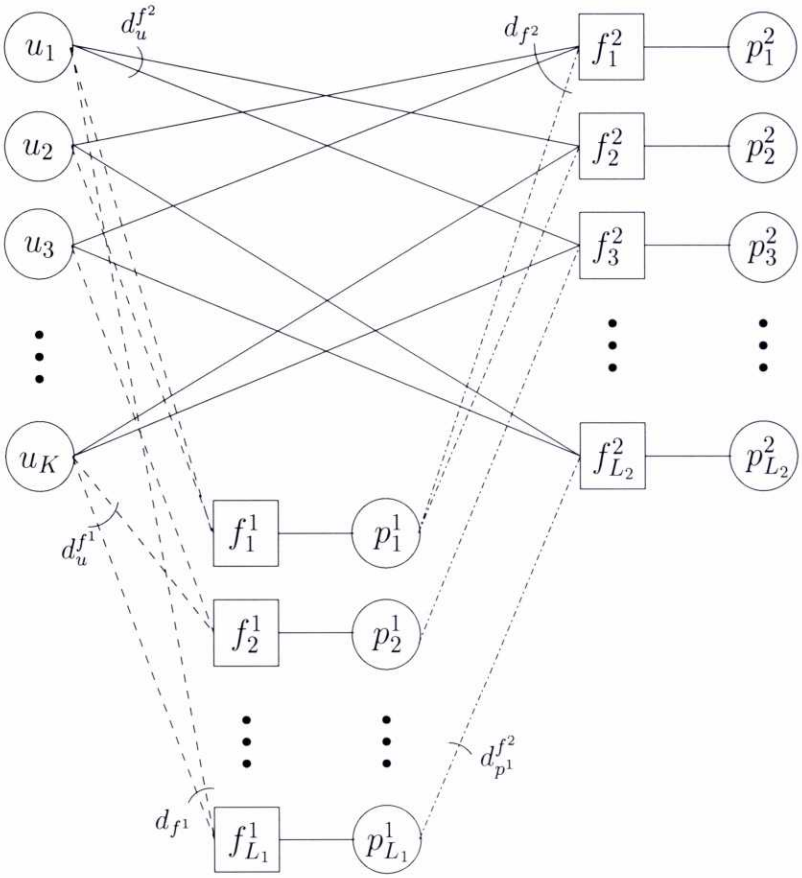


Figure 3.2: Factor graph of a SCLDGM code

3.2.1 AWGN

The complex AWGN channel model assumes that the received observations are obtained as

$$x[k] = s[k] + n[k], \quad k = 1, 2, \ldots, N/M_c \tag{3.1}$$

where $n[k]$ are i.i.d. complex samples of AWGN with variance $\sigma_n^2 = N_0$ (i.e., $N_0/2$ per real dimension). Since the number of information bits per symbol is $R_c M_c$, the E_b/N_0 at reception is given by

$$\frac{E_b}{N_0} = \frac{1}{R_c M_c} \frac{E_s}{N_0} \tag{3.2}$$

Let us assume without loss of generality that the coded bits $c_1, \ldots, c_{M_c}$ are mapped to the constallation symbol s, which is received as the observation x (we drop the index k for the sake of clarity). Since the observations are independent, the LLR values needed by the decoder can be calculated by the optimum detector for each bit c_j as

$$L_{\mathrm{ch},j} = \log \frac{p(x|c_j = 1)}{p(x|c_j = 0)} = \log \frac{p(c_j = 1|x)}{p(c_j = 0|x)} - \underbrace{\log \frac{p(c_j = 1)}{p(c_j = 0)}}_{L_j}, \quad j = 1 \ldots M_c \tag{3.3}$$

Elaborating the *A Posteriori* Probability (APP) of the coded bit, we have

$$p(c_j = b|x) = \sum_{s \in S_j^b} p(s|x) \propto \sum_{s \in S_j^b} p(x|s)p(s) \propto \sum_{s \in S_j^b} \exp\left(-\frac{\|x - s\|^2}{N_0}\right) p(s) \tag{3.4}$$

where S_j^b is the set of constellation symbols in which the j-th bit is equal to b. Using Eq. (2.29) and assuming that the bits mapped to a symbol are independent, we can calculate the *a priori* symbol probability as

$$p(s) = \prod_{i=1}^{M_c} p(c_i = s_i) = \prod_{i=1}^{M_c} \frac{1}{1 + e^{-v_i L_i}} = \prod_{i=1}^{M_c} \frac{e^{v_i L_i/2}}{e^{v_i L_i/2} + e^{-v_i L_i/2}} \tag{3.5}$$

where s_i is the i-th bit of s, $v_i = 2s_i - 1$ and the values

$$L_i = \log \frac{p(c_i = 1)}{p(c_i = 0)} \tag{3.6}$$

are the *a priori* log values of the bits mapped to the constellation symbol s. Since v_i can only take values of -1 and $+1$, the denominator does not depend on v_i, and we can rewrite (3.5) as

$$p(s) = \prod_{i=1}^{M_c} \frac{1}{e^{L_i/2} + e^{-L_i/2}} \prod_{i=1}^{M_c} e^{v_i L_i/2} \propto \exp\left(\sum_{i=1}^{M_c} v_i \frac{L_i}{2}\right) \tag{3.7}$$

Substituting the previous expressions in Eq. (3.3) results in

$$L_{\mathrm{ch},j} = \log \frac{\sum_{s \in S_j^1} \exp\left(-\frac{\|x-s\|^2}{N_0} + \sum_{i=1}^{M_c} v_i \frac{L_i}{2}\right)}{\sum_{s \in S_j^0} \exp\left(-\frac{\|x-s\|^2}{N_0} + \sum_{i=1}^{M_c} v_i \frac{L_i}{2}\right)} - \underbrace{\log \frac{p(c_j=1)}{p(c_j=0)}}_{L_j} \tag{3.8}$$

which can be rewritten as

$$L_{\mathrm{ch},j} = \log \frac{\sum_{s \in S_j^1} \exp\left(-\frac{\|x-s\|^2}{N_0} + \sum_{i=1, i\neq j}^{M_c} v_i \frac{L_i}{2}\right)}{\sum_{s \in S_j^0} \exp\left(-\frac{\|x-s\|^2}{N_0} + \sum_{i=1, i\neq j}^{M_c} v_i \frac{L_i}{2}\right)} \tag{3.9}$$

where S_j^0 and S_j^1 are the set of constellation symbols in which the j-th bit is equal to 0 or 1, respectively.

The fact that the channel LLRs, $L_{\mathrm{ch},j}$, depend on the *a priori* L-values of the bits, L_i, suggests that the detector can be interpreted as as another component of an iterative receiver, that uses the extrinsic information from the decoder as *a priori* information to recalculate the values $L_{\mathrm{ch},j}$. These values are in fact the extrinsic information provided by the detector to the decoder, since we are not using the *a priori* L-value of the bit j to calculate the value $L_{\mathrm{ch},j}$. This means that we can calculate the EXIT characteristic of the detector to study its performance. To do that, we must calculate the information I_{ch} of the messages computed by the detector ($L_{\mathrm{ch},j}$) as a function of the *a priori* information I_A of the messages passed by the decoder (L_i). Since, in general, this function does not have a closed form, we need to perform Monte Carlo simulations to estimate it. The procedure is the following: after fixing the system SNR and the bit-to-symbol mapping, we generate a random sequence of bits and map them to the corresponding constellation symbols, which are sent through the channel. At the same time, we generate a set of L-values with variance $\sigma^2 = J^{-1}(I_A)$ following the model given by (2.37). These *a priori* L-values are passed, together with the channel observations, to the detector, that calculates the $L_{\mathrm{ch},j}$ values using (3.9). Finally, we calculate the mutual information between these $L_{\mathrm{ch},j}$ values and the generated bit sequence.

Figure 3.3 shows the EXIT curves of the optimum detector for an AWGN channel when using Gray mapping and different constellation sizes. We can see that, although as we increase the number of levels in the modulation the slope also increases, the curves remain almost an horizontal line. This means that, when using Gray mapping, the *a priori* information does not take an important role in the detection process, so we can calculate the LLRs of the coded bits once and proceed to the decoding without needing further detector iterations. Thus, the calculation of the $L_{\mathrm{ch},j}$ values for the case of Gray mapping and the channel model given by Eq.

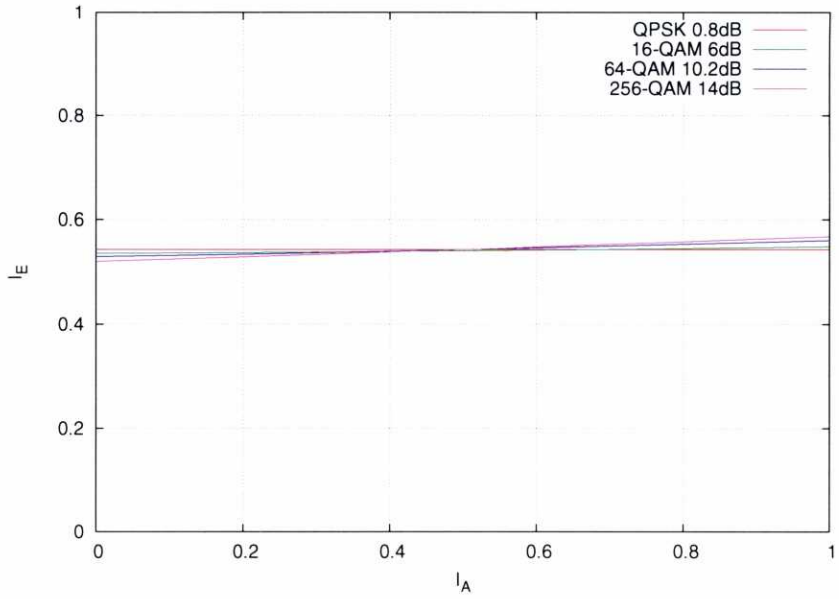


Figure 3.3: EXIT curves for the optimum detector over an AWGN channel for different SNRs.

(3.1) can be simplified to

$$L_{\mathrm{ch},j} = \log \frac{\sum_{s \in S_j^1} \exp\left(-\frac{\|x-s\|^2}{N_0}\right)}{\sum_{s \in S_j^0} \exp\left(-\frac{\|x-s\|^2}{N_0}\right)} \tag{3.10}$$

The detector EXIT curves have an interesting property: when considering a Binary Erasure Channel (BEC), the area below the curve equals the system capacity [42]. This area property can be considered as approximate for the BIAWGN channel assumed for modelling the code and detector messages and gives us an idea of the maximum code rate that can be employed with the detector. For example, all the EXIT curves in Figure 3.3 have approximately the same area below them (slightly above 0.5), so a rate $R_c = 0.5$ optimized code will perform approximately at the SNRs indicated for each modulation in the figure.

EXIT curves can also be calculated for other mappings, but it can be shown that they are always steeper than that of Gray mapping [29]. This means that, when not using Gray mapping, we must perform detector iterations in order to achieve the best results, thus increasing the detection complexity. Also, SCLDGM codes (and, in general, LDPC-based codes) do not adapt well to very steep detector EXIT curves, which makes it harder to approach capacity. Therefore, in the rest of this work we will limit ourselves to Gray-mapped constellations.

AWGN channel capacity

It can be shown that the distribution of s that maximizes the mutual information of the complex AWGN channel model is the Gaussian distribution, and that the capacity associated to it is given by

$$C = \log_2\left(1 + \frac{E_s}{N_0}\right) \tag{3.11}$$

This capacity expression does not make any assumption on the distribution of the transmitted symbols, so the maximization of the mutual information is done over all the possible symbol distributions. This capacity is termed *unconstrained* capacity. However, in practice, we are restricted to a certain constellation of possible symbols, determined by the modulation chosen by the communications system. The *constrained* capacity can be calculated as the mutual information between the input and the output of the channel for a given constellation, as derived in Eq. (2.5)

$$C = I(S;X) = H(X) - H(X|S) \tag{3.12}$$

Assuming the channel model given by (3.1), we can elaborate both entropies as

$$\begin{aligned} H(X|S) &= -\mathbb{E}_{x,s}\left\{\log_2\left(p(x|s)\right)\right\} = -\mathbb{E}_{x,s}\left\{\log_2\left(\frac{1}{\pi N_0}\exp\left[-\frac{\|x-s\|^2}{N_0}\right]\right)\right\} \\ &= \log_2(\pi N_0) + \log_2(e)\mathbb{E}_{x,s}\left\{\frac{\|x-s\|^2}{N_0}\right\} \\ &= \log_2(\pi N_0 e) \end{aligned}$$

and

$$\begin{aligned} H(X) &= -\mathbb{E}_x\left\{\log_2(p(x))\right\} = -\mathbb{E}_x\left\{\log_2\left(\sum_s p(s)p(x|s)\right)\right\} \\ &= -\mathbb{E}_x\left\{\log_2\left(\sum_s \frac{1}{2^{M_c}}\frac{1}{\pi N_0}\exp\left[-\frac{\|x-s\|^2}{N_0}\right]\right)\right\} \end{aligned}$$

And substituting them in Eq. (3.12) we obtain

$$C = -\mathbb{E}_x\left\{\log_2\left(\frac{1}{2^{M_c}}\sum_s \exp\left[-\frac{\|x-s\|^2}{N_0}\right]\right)\right\} - \log_2(e) \tag{3.13}$$

This expectation does not have a closed form solution, so we must use Monte Carlo simulation to find the capacity. Figure 3.4 shows the constrained and unconstrained capacities for an AWGN channel for different constellations. The fact that the uncostrained AWGN capacity is twice the capacity of Figure 2.2 in the previous chapter is because in this chapter we are considering a complex valued model, that effectively doubles the capacity of the real-valued model.

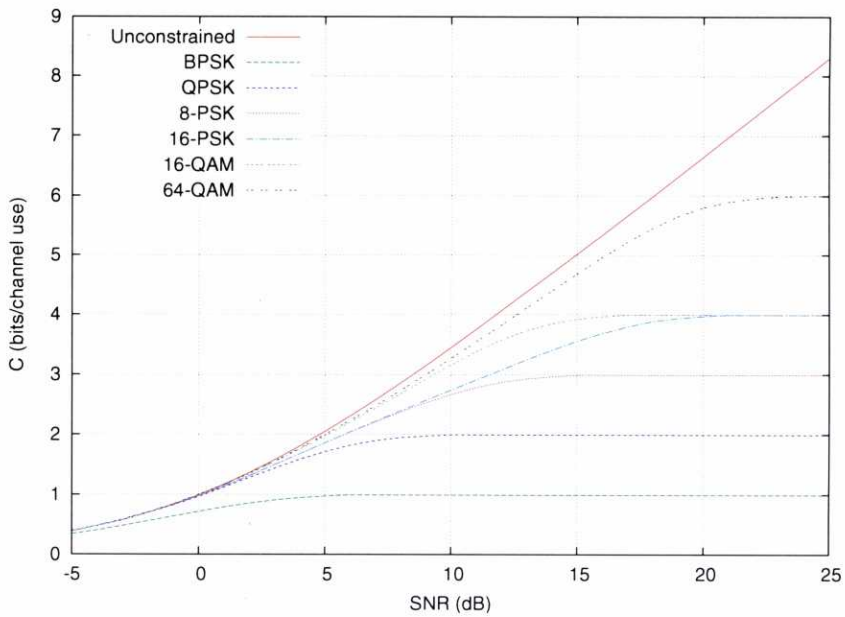


Figure 3.4: Unconstrained and constrained capacities for different constellations in an AWGN channel.

3.2.2 Rayleigh fading

In wireless communications the transmitted signals are usually affected by multipath propagation due to reflections from objects located between the transmitter and the receiver. This causes the effects of destructive and constructive interferences and phase shifting in the transmitted signals, which the AWGN model does not take account of. A more appropriate model for this kind of systems is the Rayleigh fading model, especially when there is no Line Of Sight (LOS) between the transmitter and the receiver. The Rayleigh fading model assumes that the received observation at each time instant is the result of the sum of many independent contributions, each one consisting on the transmitted symbol multiplied by a random complex coefficient that changes the phase and amplitude of the original symbol. If we assume that the number of contributions is large, the coefficients can be modelled as a Gaussian random process, according to the Central Limit Theorem. Thus, considering a memoryless channel, the observations at the receiver can be calculated as

$$x[k] = h[k]s[k] + n[k], \quad k = 1, 2, \ldots, N/M_c \tag{3.14}$$

where $h[k]$ are samples of a $\mathcal{CN}(0,1)$ distribution and $n[k]$ are complex samples of AWGN. We will consider three different fading models: the *quasi-static* fading model, where the channel remains constant during the transmission of a whole codeword (i.e., $h[k] = h$, $k = 1, \ldots, N/M_c$), changing independently between two successive codewords; the *block-fading* model, where the channel remains constant for the transmission of B symbols, $B < N/M_c$;

and the *fast* fading model, where the channel fading coefficient $h[k]$ is assumed to change independently from one use to another. If the code block length is large enough, the channel random process becomes ergodic, and the fast fading model becomes an *ergodic* channel.

Assuming that the receiver has perfect Channel State Information (CSI), i.e., that the receiver knows the values $h[k]$, the LLR values for the decoder are calculated in a similar manner as in the AWGN model as

$$L_{\text{ch},j} = \log \frac{\sum_{s \in S_j^1} \exp\left(-\frac{1}{N_0}\|x - hs\|^2 + \sum_{i=1}^{M_c} v_i \frac{L_i}{2}\right)}{\sum_{s \in S_j^0} \exp\left(-\frac{1}{N_0}\|x - hs\|^2 + \sum_{i=1}^{M_c} v_i \frac{L_i}{2}\right)} - L_j \tag{3.15}$$

As occurs in the AWGN case, when using Gray mapping the EXIT curves for the optimum detector are practically an horizontal line (Figure 3.5), so we can drop the *a priori* information and calculate the extrinsic LLRs as

$$L_{\text{ch},j} = \log \frac{\sum_{s \in S_j^1} \exp\left(-\frac{1}{N_0}\|x - hs\|^2\right)}{\sum_{s \in S_j^0} \exp\left(-\frac{1}{N_0}\|x - hs\|^2\right)} \tag{3.16}$$

Rayleigh fading channel capacity

The distribution that maximizes the mutual information of the Rayleigh fading channel is also the Gaussian distribution, and the capacity associated to it for a fixed channel coefficient h is given by

$$C(h) = \log_2\left(1 + \frac{E_s}{N_0}|h|^2\right) \tag{3.17}$$

When each codeword is transmitted over M different channel realizations, the capacity is calculated as the average of all the realization capacities, i.e.,

$$C(h) = \frac{1}{M}\sum_{k=1}^{M} \log_2\left(1 + \frac{E_s}{N_0}|h[k]|^2\right) \tag{3.18}$$

If the coefficients $h[k]$ are modelled as a random process, this average capacity will be a random variable, that can be characterized by its Cumulative Distribution Function (CDF). The CDF of the average capacity depends on the assumed fading model. If the channel is erdogic the associated capacity will be a fixed value called *ergodic capacity*, which is given by

$$C = \mathbb{E}_h\{C(h)\} = \mathbb{E}_h\left\{\log_2\left(1 + \frac{E_s}{N_0}|h|^2\right)\right\} \tag{3.19}$$

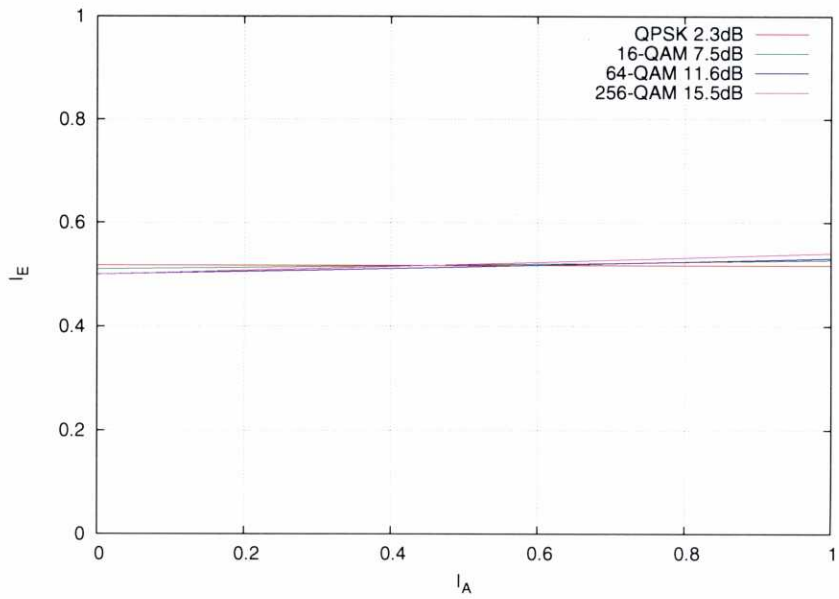


Figure 3.5: EXIT curves for the optimum detector over a Rayleigh fading channel for different SNRs.

On the other hand, quasi-static channels do not have an associated capacity: since we are using a code of a fixed rate to transmit over a single channel realization h, there is always the chance that the channel realization has an associated capacity lower than the information rate, which makes error-free communication impossible. When this happens, the channel is said to be in *outage*. In this case, to get an idea about the channel ability to transmit information, it is necessary to resort to the concept of *outage capacity*: the capacity of the channel provided it is not in outage. This capacity is thus related to the outage probability, i.e., that of the channel being in outage. The capacity CDF of the block fading model depends on the number of different channel realizations during the transmission of a codeword. If the number of realizations is high, the capacity will tend to the ergodic capacity (Figure 3.6).

The constrained capacity of a fixed channel realization h can be calculated in an analogous way as in the AWGN channel

$$C(h) = -\mathbb{E}_x \left\{ \log_2 \left(\frac{1}{2^{M_c}} \sum_s \exp \left[-\frac{1}{N_0} \left\| x - hs \right\|^2 \right] \right) \right\} - \log_2(e) \tag{3.20}$$

and we can define the ergodic constrained capacity as the expected capacity over the random variable h

$$C = -\mathbb{E}_h \mathbb{E}_x \left\{ \log_2 \left(\frac{1}{2^{M_c}} \sum_s \exp \left[-\frac{1}{N_0} \left\| x - hs \right\|^2 \right] \right) \right\} - \log_2(e) \tag{3.21}$$

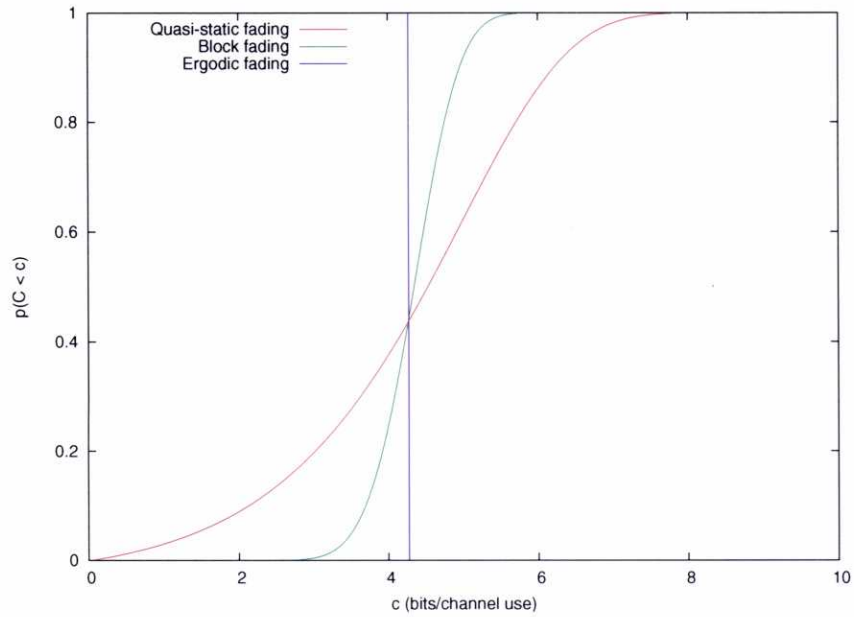


Figure 3.6: Capacity CDF of Rayleigh fading channels for SNR=15

Figure 3.7 shows the constrained and unconstrained capacities of an ergodic Rayleigh fading channel for different constellations, and Figure 3.8 a comparison between AWGN and Rayleigh fading capacities.

Diversity

Diversity is an important concept that appears associated with fading channels. Making use of the diversity provided by the channel consists in improving the reliability of the trasmitted signals by utilizing several communication channels with different characteristics. This scheme can be used to avoid errors due to deep fades, since the probability that all the versions of the same signal are affected by a deep fade is very low. At reception, the multiple received versions of the incoming signal can be combined to obtain the original signal. The performance improvement obtained because of the diversity is called *diversity gain*.

For example, let us consider an n-repetition code that takes one bit as the input and constructs the codeword by repeating that bit n times. This trivial code does not provide any coding gain when used over a BPSK AWGN channel, because the reduction in bit error probability with respect to uncoded transmission for a given SNR is compensated by the extra energy wasted in the coded bits. This means that the performance versus the E_b/N_0 is exactly the same as uncoded transmission for any value of n. However, if we employ a repetition code in a fast fading Rayleigh channel, the performance does improve as we increase the number of repetitions (Figure 3.9). This gain is due to the increase in diversity, since each information

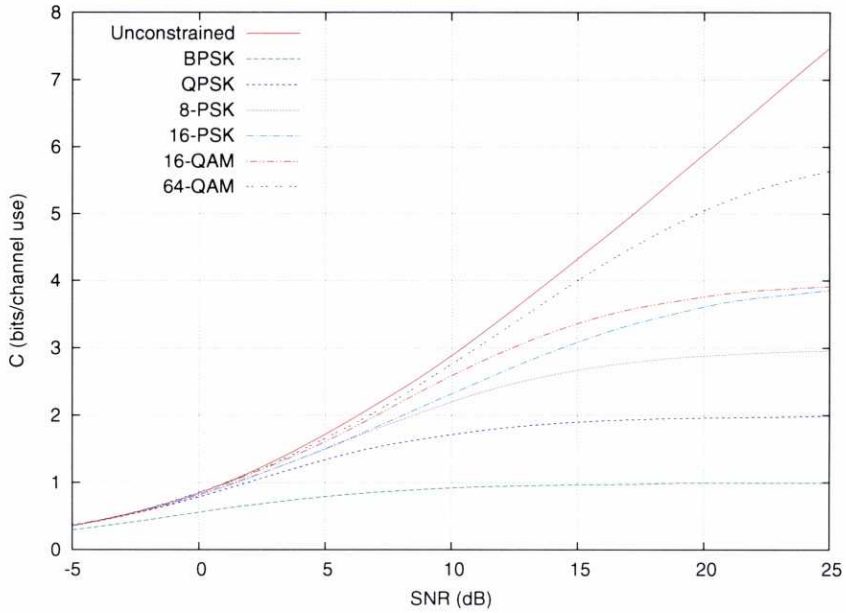


Figure 3.7: Unconstrained and constrained capacities for different constellations in a Rayleigh fading channel.

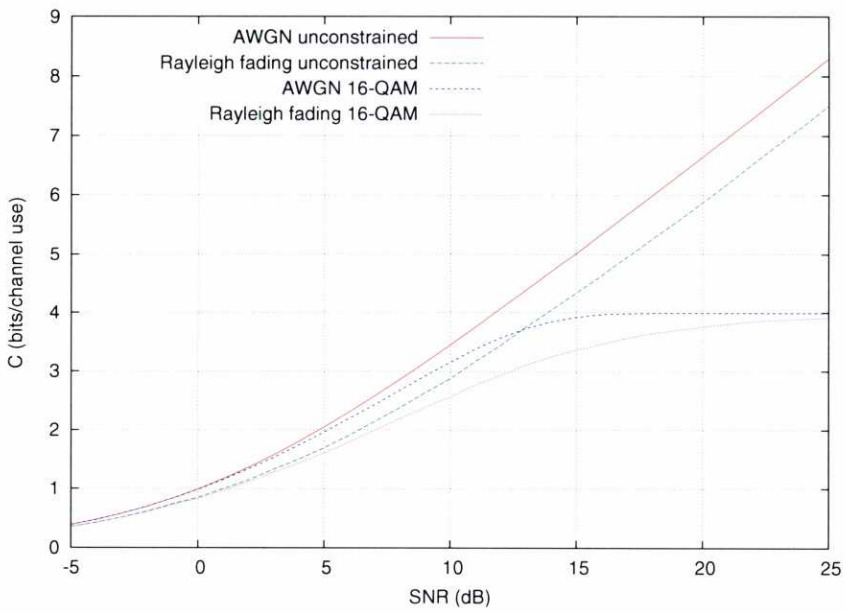


Figure 3.8: Capacities for AWGN and Rayleigh fading channels

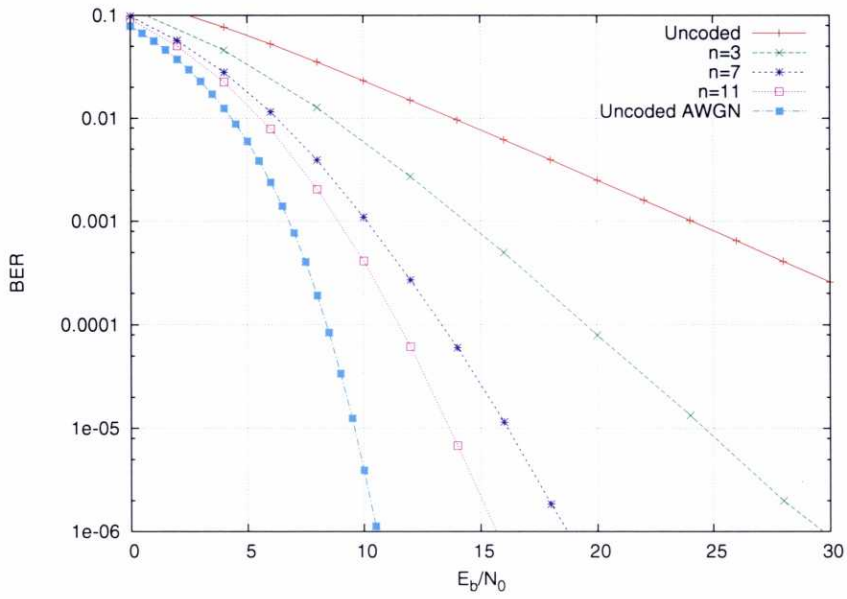


Figure 3.9: Performance of n-repetition codes over a fast fading Rayleigh channel

bit is sent over n different channel realizations (i.e., the diversity is equal to n). This class of diversity is called *temporal diversity*, because the same information is sent in different time intervals. It can be shown that the asymptotic slope of the performance curve is equal to the diversity of the channel [52].

When using large random codes such as LDPC or SCLDGM codes, we are already making use of the diversity provided by the channel. Indeed, if we assume an ergodic channel, each information bit is sent across many different channel realizations, because each information bit is used in the computation of many parity bits that are sent in different time intervals.

3.3 Convergence Threshold

Although, unlike RA and LDPC codes, regular SCLDGM codes are able to perform close to the capacity limit, it is still necessary to find the optimum degrees for the outer and inner codes and the best distribution of rates between both codes. To find the theoretical convergence threshold of SCLDGM codes we can use the EXIT functions defined in the previous chapter. Thus,

particularizing Eqs. (2.51) and (2.52) for the factor graph of a regular SCLDGM code, we have

$$I_u^{f^1} = J\left((d_u^{f^1}-1)J^{-1}(I_{f^1}^u) + d_u^{f^2}J^{-1}(I_{f^2}^u) + \sigma_{ch}^2\right) \tag{3.22}$$

$$I_u^{f^2} = J\left(d_u^{f^1}J^{-1}(I_{f^1}^u) + (d_u^{f^2}-1)J^{-1}(I_{f^2}^u) + \sigma_{ch}^2\right) \tag{3.23}$$

$$I_{p^1}^{f^1} = J\left(d_{p^1}^{f^2}J^{-1}(I_{f^2}^{p^1}) + \sigma_{ch}^2\right) \tag{3.24}$$

$$I_{p^1}^{f^2} = J\left((d_{p^1}^{f^2}-1)J^{-1}(I_{f^2}^{p^1}) + J^{-1}(I_{f^1}^{p^1}) + \sigma_{ch}^2\right) \tag{3.25}$$

$$I_{p^2}^{f^2} = J\left(\sigma_{ch}^2\right) \tag{3.26}$$

$$I_{f^1}^u = 1 - J\left((d_{f^1}-1)J^{-1}(1-I_u^{f^1}) + J^{-1}(1-I_{p^1}^{f^1})\right) \tag{3.27}$$

$$I_{f^1}^{p^1} = 1 - J\left(d_{f^1}J^{-1}(1-I_u^{f^1})\right) \tag{3.28}$$

where $\sigma_{ch}^2 = J^{-1}(I_{ch})$.

For messages from inner checks to outer variable nodes, we have to distinguish between two types of inner check nodes: those that are connected to outer parity bits, $f_{i,a}$, (for simplicity, we assume that an inner check may receive at most one edge from the outer parity bits) and those that are not, $f_{i,b}$. Thus,

$$I_{f^2,a}^u = 1 - J\left(J^{-1}(1-I_{p^1}^{f^2}) + (d_{f^2}-2)J^{-1}(1-I_u^{f^2}) + J^{-1}(1-I_{p^2}^{f^2})\right) \tag{3.29}$$

$$I_{f^2,b}^u = 1 - J\left((d_{f^2}-1)J^{-1}(1-I_u^{f^2}) + J^{-1}(1-I_{p^2}^{f^2})\right) \tag{3.30}$$

The mutual information of messages $f^2 \rightarrow u$ is now given by the sum of $I_{f^2,a}^u$ and $I_{f^2,b}^u$, weighted by the fraction of edges carrying each type of messages. Denoting by $a = L_1 d_{p^1}^{f^2}$ the number of inner checks that are connected to outer parity bits, this results in

$$I_{f^2}^u = \frac{a(d_{f^2}-1)I_{f^2,a}^u + (L_2-a)d_{f^2}I_{f^2,b}^u}{a(d_{f^2}-1) + (L_2-a)d_{f^2}} \tag{3.31}$$

Extension of the previous equations to irregular SCLDGM codes is quite straighforward. It is only necessary to consider that the mutual information associated to a given type of messages is the mean, weighted by the correspondig fraction of edges, of the mutual information of messages originating at nodes having a certain degree.

Code convergence is tested by simulating the evolution of messages mutual information through the iterations of the SPA. For a fixed E_b/N_0 (and thus, a fixed σ_n) we start with all I's equal to zero and, then, we iteratively compute their values. This is exactly what the SPA does but using representatives of messages (their associated I's) instead of their actual values. We say that the code converges when we find a sequence of I's that finally leads to $I_u = 1$, where I_u is the mutual information associated to the *a posteriori* LLRs of bits u (see Eq. (2.35)). When I_u is almost, but strictly less than 1, convergence is achieved but with a significant error-floor.

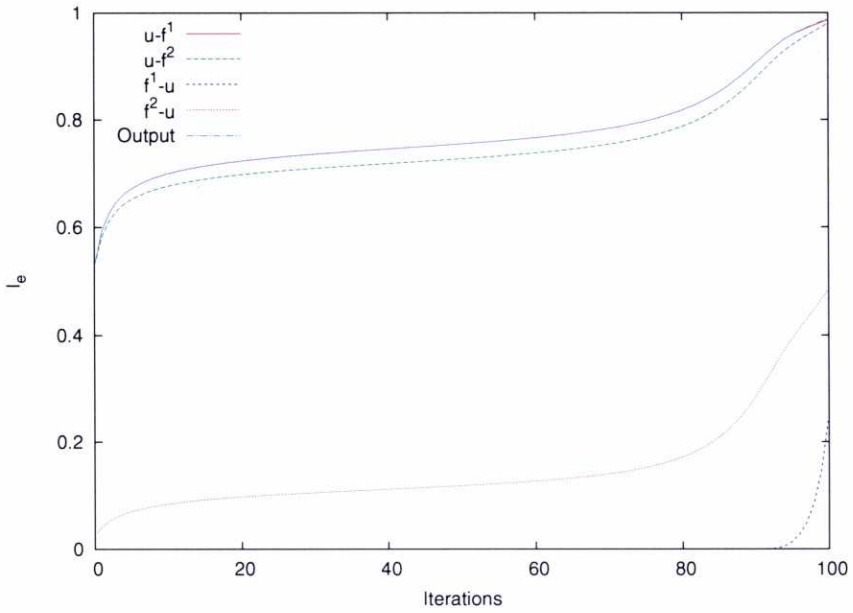


Figure 3.10: Mutual information trajectories for the regular SCLDGM code over a BIAWGN channel when $E_b/N_0 = 0.59$dB.

Mutual information trajectories clearly illustrate decoding convergence. As an example, Fig. 3.10 plots these trajectories for a regular SCLDGM code (#1 in Table 3.1) when $E_b/N_0 = 0.59$dB. Notice that the SPA is not converging because the mutual information of L_u messages, indistinguishable from that of $u \rightarrow f^1$ messages, is close to (but still not equal to) one after 100 iterations. Nevertheless, see how the outer code starts to help from iteration 90: messages from outer checks to outer variables carry mutual information greater than zero. Fig. 3.11 shows the decoding behaviour for $E_b/N_0 = 0.60$dB. Now the outer code starts to help from iteration 65, leading to overall convergence at iteration 75: mutual information of L_u messages is equal to one.

The EXIT functions also explain the need of an outer code in a LDGM code. Figure 3.12 shows the evolution through decoding iterations of mutual information of messages for a single (non-concatenated) regular LDGM code for different values of the E_b/N_0. The code rate is $R_c = 1/2$ and the degree of both variable and check nodes is six. At $E_b/N_0 = 0.25$dB, mutual information of variable-to-check messages stays fixed at about 0.65 from iteration 20. Since 0.65 is far less from one, the BER will be quite high: the code is in the non-convergence region. When $E_b/N_0 = 0.35$dB, mutual information of variable-to-check messages reaches a much higher value (about 0.95) from iteration 65: BER is low but still significant (0.95 is far from 1). When we increase the E_b/N_0 to 0.9dB the error-floor effect clearly appears: a higher mutual information is quickly achieved but it is still not 1, meaning that some errors will still be present, although the E_b/N_0 is far from the convergence threshold observed at $E_b/N_0 = 0.35$dB. Only

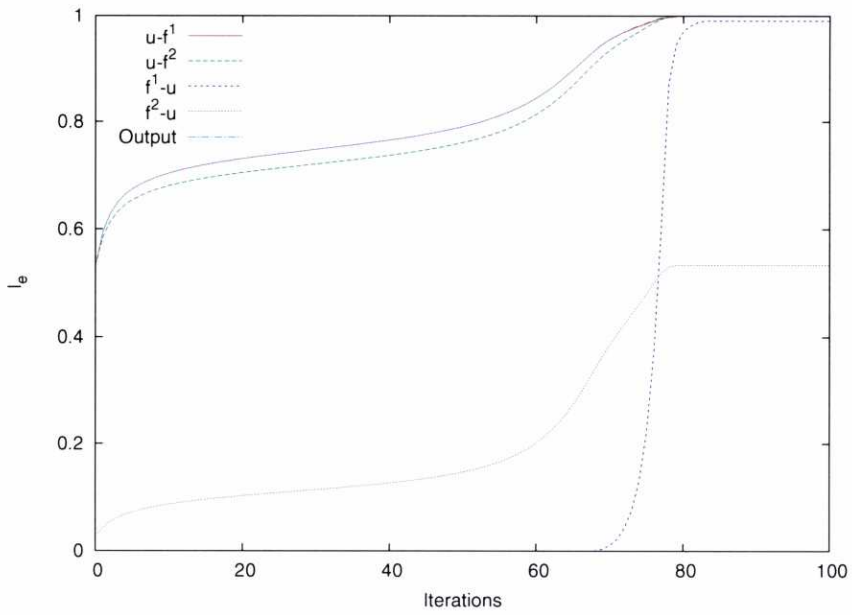


Figure 3.11: Mutual information trajectories for the regular SCLDGM code over a BIAWGN channel when $E_b/N_0 = 0.60$dB.

when we help a LDGM code with an outer code, the error-floor can be lowered.

3.4 Code Optimization

The previous section suggests a straightforward way to optimize SCLDGM codes. We start considering an initial channel E_b/N_0. For this value, we simulate the SPA evolution for all possible code degree profiles and discard those for which the SPA does not converge. Next, if we lower the E_b/N_0 value and repeat the same simulation we will find out that some of the remaining codes still converge (the others that do not converge are now discarded). We proceed in the same way, lowering the E_b/N_0 value until only one convergent code remains. This code is the optimum and the E_b/N_0 value is its convergence threshold.

It is important to note that the codes optimized for a BIAWGN channel are also optimal for modulations with more than two levels over AWGN and Rayleigh fading channels. As we have seen in the previous section, for higher order modulations with Gray mapping, the EXIT function does not depend on the *a priori* information, i.e., the EXIT function is almost a straight line for each particular E_b/N_0 value. Since the EXIT function of a BIAWGN channel is also a straight line, codes optimal for binary modulations are also optimal when using higher order modulations or Rayleigh fading channels.

For regular SCLDGM codes, we examined SPA convergence varying the code parameters between the following limits $0.001 \leq p \leq 0.05$, with increments of 0.0025; $3 \leq d_u^{f^1} \leq 5$;

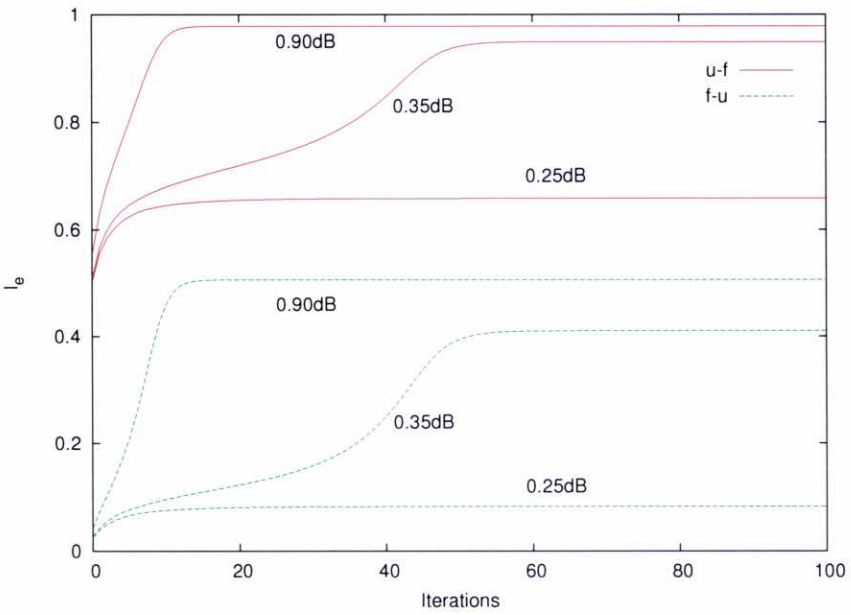


Figure 3.12: Mutual information trajectories for a rate $1/2$ LDGM code code over a BIAWGN channel.

Code	p	$d_u^{f^1}$	$d_u^{f^2}$	$a_u^{f^2}$	$d_{p^1}^{f^2}$	Thres.	Gap
#1 (non-optimized reg.)	0.033	3	6	1	6	0.60	0.41
#2 (optimized reg.)	0.020	3	6	1	40	0.46	0.27
#3 (optimized irreg.)	0.040	3	4, 6, 35	0.25, 0.7, 0.05	40	0.36	0.17

Table 3.1: Degree profiles of rate $1/2$ SCLDGM codes for AWGN channel. Channel capacity is at $E_b/N_0 = 0.19$dB.

$2 \leq d_u^{f^2} \leq 35$; and $2 \leq d_{p^1}^{f^2} \leq 40$. For irregular SCLDGM codes, we considered a degree profile with three different degrees for the systematic bits u with respect to the inner check nodes f^2, i.e., we fixed $D_v^{f^2} = 3$. Parameters were varied between the following limits $0.01 \leq p \leq 0.15$, with increments of 0.005; $3 \leq d_u^{f^1} \leq 5$; $2 \leq d_{u,i}^{f^2} \leq 35$; $0.05 \leq a_{u,i}^{f^2} \leq 1$ with increments of 0.05; and $2 \leq d_{p^1}^{f^2} \leq 25$. We arrived at codes #2 (regular) and #3 (irregular) shown in Table 3.1.

3.5 Simulation Results

Computer simulations were carried out to illustrate the convergence thresholds in practical implementations of the obtained SCLDGM codes with data blocks of finite length. These thresholds are slightly higher than those predicted because EXIT function analysis assumes infinite-length data blocks.

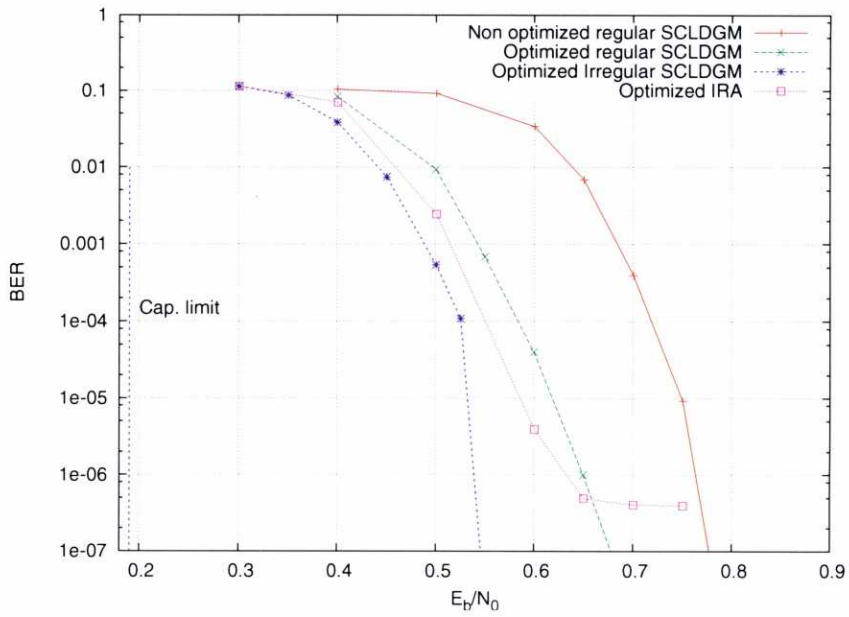


Figure 3.13: BER vs E_b/N_0 for different $R_c = 1/2$ codes over a BIAWGN channel. Block length is $K = 50000$ systematic bits. Channel capacity is at $E_b/N_0 = 0.19$dB

Figure 3.13 compares the performance over an AWGN channel of the obtained regular and irregular SCLDGM codes, the optimized IRA code in [40] and a non-optimized SCLDGM code from [53]. The rate of all codes is $R_c = 1/2$ and the block length $K = 50000$ systematic bits. We performed a maximum of 100 iterations over the factor graph and stop if the same decoded sequence is produced in 3 consecutive iterations. The simulations were carried out until 100 block errors were found or 7000 codewords were transmitted. The non-optimized SCLDGM code is the code #1 in Table 3.1. Examining Figure 3.13, it can be clearly seen the ability of the optimized codes to approach the capacity limit, which is at $E_b/N_0 = 0.19$dB. In addition, the performance of the optimized SCLDGM codes is 0.2dB better than that of the non-optimized SCLDGM code.

Similar considerations can be made for the case of binary-input uncorrelated Rayleigh fading channel (Figure 3.14). As in the BIAWGN case, the channel LLRs only depend on the noise variance, so the codes optimized for the BIAWGN channel also perform very close to the capacity limit. Nevertheless, the gaps with respect to the capacity limit ($E_b/N_0 = 1.8$dB) are slightly higher.

Figure 3.15 shows the performance of the optimized irregular SCLDGM code when transmitting over AWGN and Rayleigh fading channels using higher order modulations. Since the EXIT curves are practically flat for all the modulations, the performance of the optimized code is very good in all cases. However, the increase in the slope for higher order modulations causes a slight increase in the gap to the capacity limit.

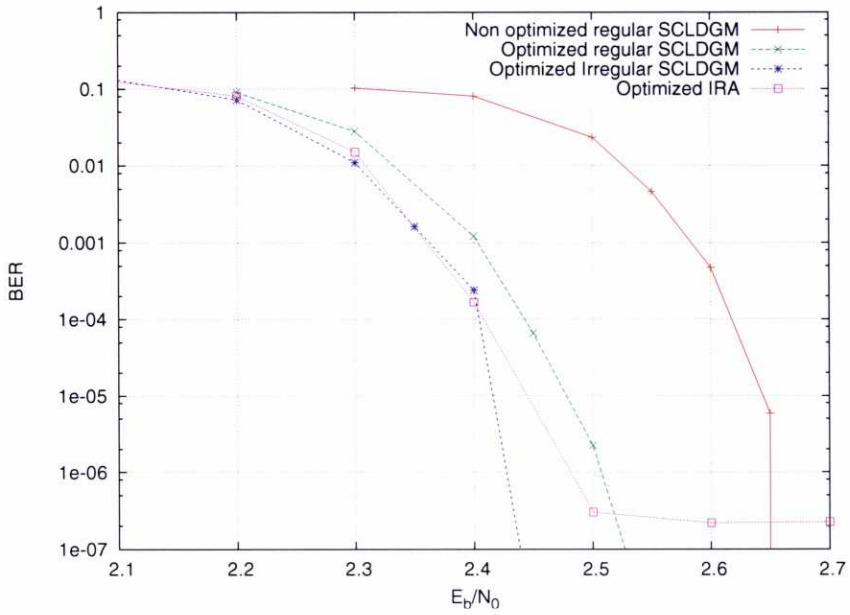


Figure 3.14: BER vs E_b/N_0 for different $R_c = 1/2$ codes over a binary-input Rayleigh fading channel. Block length is $K = 50000$ systematic bits. Channel capacity is at $E_b/N_0 = 1.8$dB

Finally, Figure 3.16 presents the performance of the optimized irregular SCLDGM code for different values of the block length. As expected, increasing the block length improves performance. Also note that the resulting convergence threshold approaches the value predicted by EXIT functions (0.36dB).

3.6 Conclusions

In this chapter we have presented a coding scheme termed Serially-Concatenated Low-Density Generator Matrix (SCLDGM) that is based on the concatenation of two LDGM codes. The basic idea of the scheme is to use the outer high-rate LDGM code as a means to reduce the error floor resulting from the decoding of the inner LDGM code. Thanks to the sparseness of the generator matrices, the encoding complexity is similar to that of a RA code and lower than that of an LDPC code. To decode an SCLDGM code, the SPA is employed over the factor graph resulting from the concatenation of the two factor graphs corresponding to each constituten LDGM code.

We were interested in finding good SCLDGM codes for both AWGN and Rayleigh fading channels with different modulations using optimization techniques based on EXIT charts. Since, except for the case of a BIAWGN channel, there is no closed form solution for the calculation of the mutual information passed by the detector to the decoder, we have to estimate the detector EXIT functions using Monte Carlo simulations. We have showed that, when using

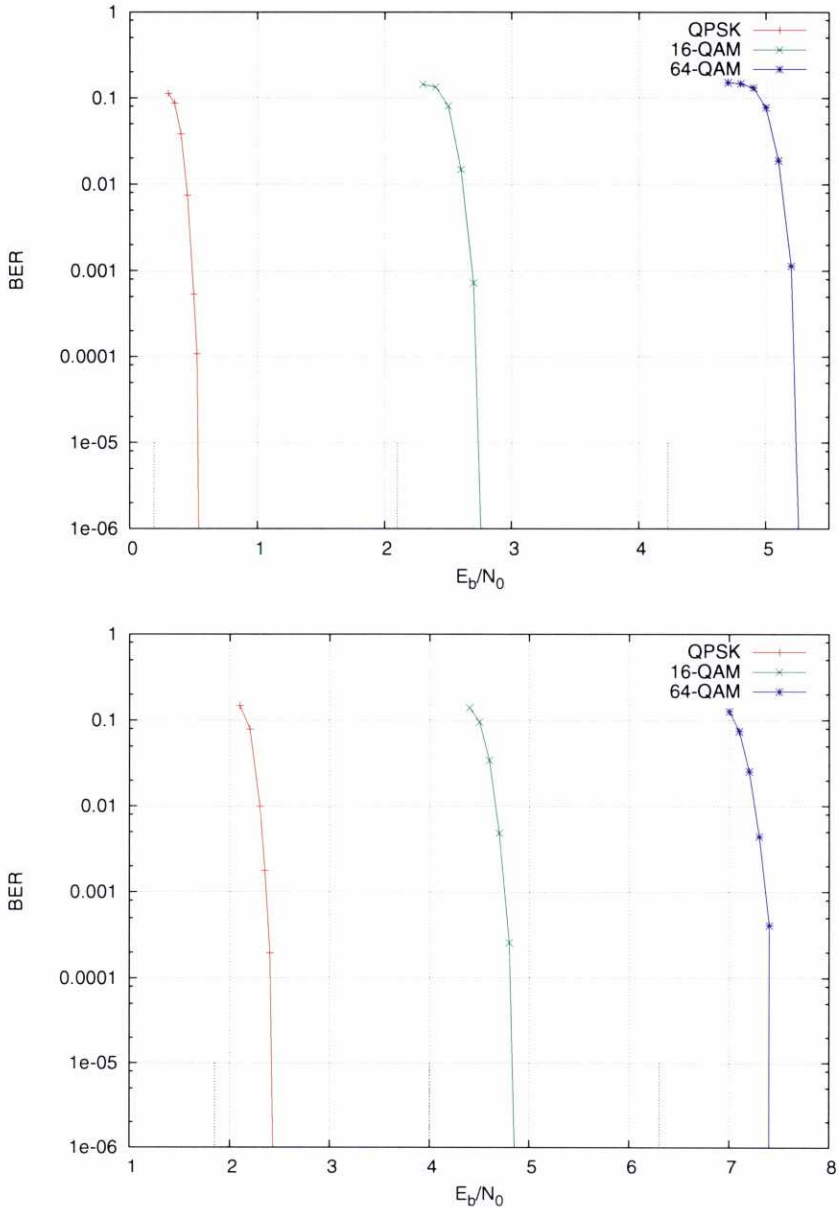


Figure 3.15: BER vs E_b/N_0 of the $R_c = 1/2$ irregular optimized SCLDGM code for different modulations. Block length is $K = 50000$ systematic bits. The dotted lines represent the constrained capacity limits.

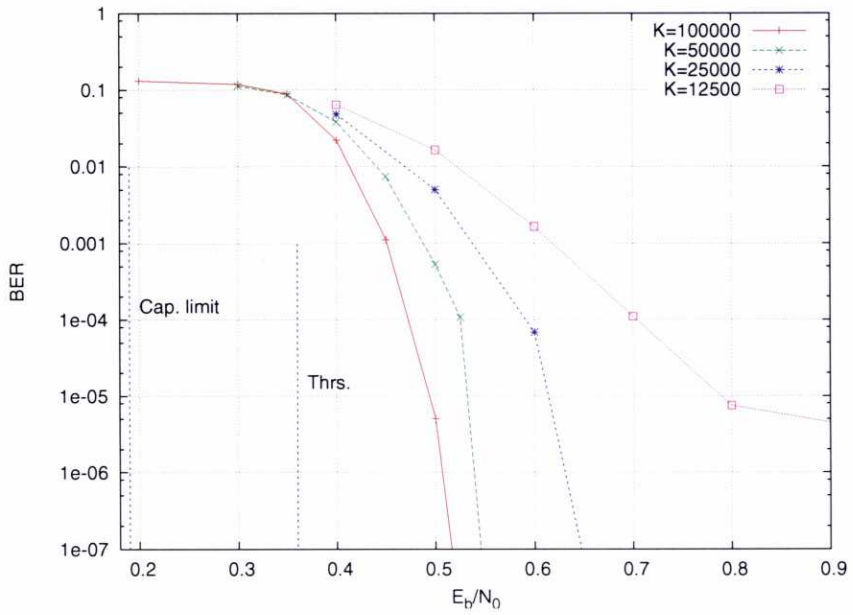


Figure 3.16: BER vs E_b/N_0 for the optimized $R_c = 1/2$ irregular SCLDGM code over a BIAWGN channel for different block lengths, K. Channel capacity is at $E_b/N_0 = 0.19$dB and the convergence threshold is at $E_b/N_0 = 0.36$dB.

Gray mapping, the EXIT curves for all the studied modulations in both channels are practically horizontal lines, so we can expect a code optimized for a BIAWGN channel (which also has an horizontal EXIT function) to exhibit a good performance when using higher order modulations.

Since SCLDGM codes are a particular case of LDPC codes, the same EXIT functions can be applied to them. However, due to the more complex structure of their associated factor graphs, the curve fitting procedure is not appropriate for SCLDGM codes. Instead, we have proposed to use the EXIT functions as substitutes for the SPA, which allow us to predict the convergence threshold of a given code with very little complexity and therefore to perform exhaustive searches to find good codes. We have seen that an SCLDGM code found using this procedure is able to surpass the theoretical and practical performance of an optimized IRA code in AWGN and Rayleigh fading channels with a lower error floor. We have also showed that the SCLDGM code optimized for the BIAWGN channel performs also very close to the constrained capacities when using QPSK, 16 QAM and 64 QAM constellations in AWGN and Rayleigh fading channels.

Chapter 4

Channel Coding for MIMO systems

Information theory predicts that the capacity of a wireless communication link that utilizes multiple antennas at both transmission and reception, known as Multiple Input Multiple Output (MIMO) system, is considerably larger than that of conventional single antenna systems [54, 55]. This fact has attracted a great deal of attention during the last decade, due to the increasing data rate requirements of wireless communication applications. In this chapter we investigate coding, detection and code optimization techniques that can be used with MIMO systems. In particular, we will discuss a Bit-Interleaved Coded Modulation (BICM) scheme in which the coded bits are mapped to constellation symbols and then a simple spatial multiplexer is used to distribute the symbols to the different trasmit antennas [56]. In the receiver, a suboptimum iterative detector will be employed to mitigate the problem of the exponential complexity in the number of transmit antennas suffered by the optimum MIMO detector. We will show that this scheme is able to take advantage of the increased capacity provided by the MIMO channel with an affordable complexity.

4.1 SCLDGM codes for MIMO channels

Figure 4.1 shows an example of a coded MIMO system. At the transmitter, the source bits are encoded, mapped to constellation symbols and passed to a spatial multiplexer, which distributes

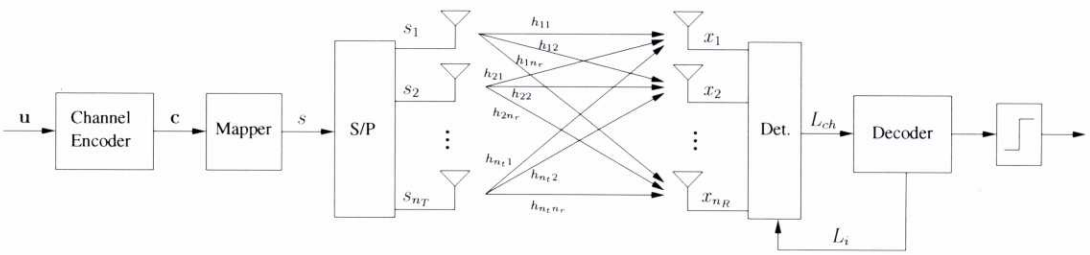


Figure 4.1: Block diagram of a BICM MIMO System

the modulated symbols to the n_T transmit antennas. Each signal, with an average energy E_s, travels to each of the n_R receive antennas through different wireless channels, affected by different fading coefficients. In the receiver, the effect of the n_T transmitted signals is added in each of the receive antennas. If we assume a flat-fading channel, the signal received in the j-th antenna can be written as

$$x_j[k] = \sum_{i=0}^{n_T} h_{ij}[k]s_i[k] + n_j[k] \quad k = 1, 2, \ldots, L \tag{4.1}$$

If we consider the n_R antennas, we can write the previous model in matrix form as

$$\mathbf{x}[k] = \mathbf{H}[k]\mathbf{s}[k] + \mathbf{n}[k] \quad k = 1, 2, \ldots, L \tag{4.2}$$

where[1] $\mathbf{x} = [x_1, x_2, \ldots, x_{n_R}]^T$, $\mathbf{s} = [s_1, s_2, \ldots, s_{n_T}]^T$, $\mathbf{n} = [n_1, n_2, \ldots, n_{n_R}]^T$ and

$$\mathbf{H} = \begin{bmatrix} h_{11} & h_{12} & \cdots & h_{1n_T} \\ h_{21} & h_{22} & \cdots & h_{2n_T} \\ \vdots & \vdots & \ddots & \vdots \\ h_{n_R1} & h_{n_R2} & \cdots & h_{n_Rn_T} \end{bmatrix} \tag{4.3}$$

Each entry h_{ij} is modeled as spatially uncorrelated, circularly-symmetric, complex-valued Gaussian random variables, i.e., $h_{ij} \sim \mathcal{CN}(0, 1)$. The components of the noise vector $\mathbf{n}[k]$ are also both spatial and temporally uncorrelated, and distributed as $\mathcal{CN}(0, N_o)$. Since each of the n_R receive antennas collects an average energy of $n_T E_s$ whent transmitting $n_T M_c R_c$ information bits, we calculate the E_b/N_0 at reception as

$$\frac{E_b}{N_0} = \frac{n_R n_T E_s}{n_T R_c M_c} \frac{1}{N_0} = \frac{n_R}{R_c M_c} \frac{E_s}{N_0} \tag{4.4}$$

As in the SISO case, we can consider three different basic channel models: ergodic, in which the channel matrix changes indepently with every use; quasi-static, in which it remains constant for the transmission of a codeword and changes independently between two successive codewords; and block-fading, where the channel matrix remains constant during B uses, with $B < L$.

4.1.1 Capacity of MIMO channels

With the previous channel model, and assuming that the channel is known at the receiver, it can be shown [54] that the capacity of a given MIMO channel $\mathbf{H}$ is

$$C(\mathbf{H}) = \max_{\mathbf{R_{ss}}: Tr\{\mathbf{R_{ss}}\} = n_T} \log_2 \det \left(\mathbf{I}_{n_R} + \frac{E_s}{N_0} \mathbf{H} \mathbf{R_{ss}} \mathbf{H}^H \right) \tag{4.5}$$

[1] We drop the index k for the sake of clarity

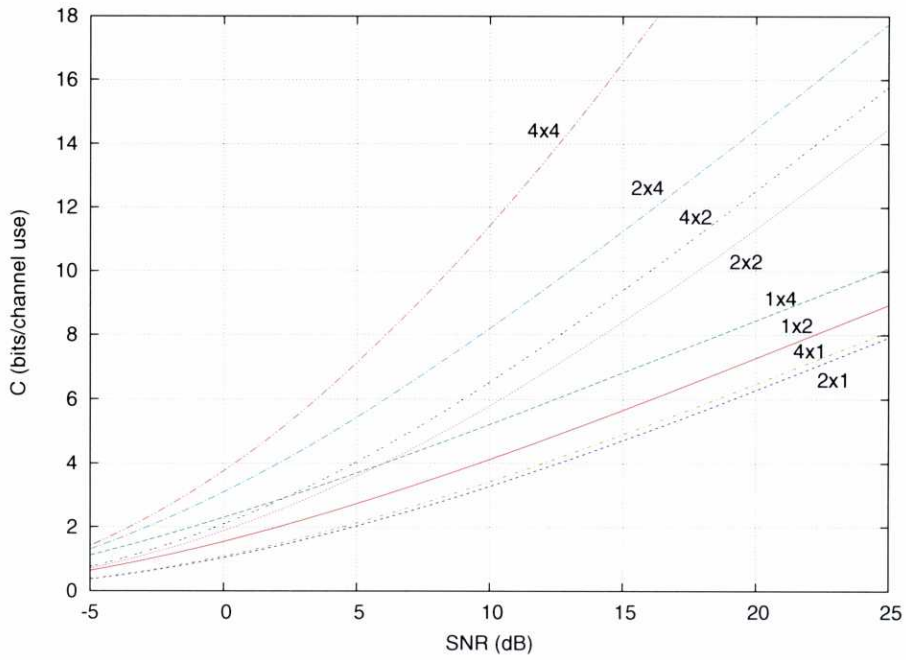


Figure 4.2: Unconstrained capacities of different flat fading Rayleigh MIMO channels.

where $\mathbf{R_{ss}}$ is the autocorrelation matrix of the transmitted symbols. When the channel is unknown at the transmitter, the symbol vector s must be chosen such as $\mathbf{R_{ss}} = \mathbf{I}_{n_T}$, so we have

$$C(\mathbf{H}) = \log_2 \det \left(\mathbf{I}_{n_R} + \frac{E_s}{N_0} \mathbf{H}\mathbf{H}^H \right) \tag{4.6}$$

As in the SISO channel model, if $\mathbf{H}$ is a matrix of random variables and the channel is unkown at the transmitter we can define the ergodic capacity as

$$C = \mathbb{E}_{\mathbf{H}} \left\{ C(\mathbf{H}) \right\} = \mathbb{E}_{\mathbf{H}} \left\{ \log_2 \det \left(\mathbf{I}_{n_R} + \frac{E_s}{N_0} \mathbf{H}\mathbf{H}^H \right) \right\} \tag{4.7}$$

Figure 4.2 shows the ergodic unconstrained capacities of MIMO channels for different antenna configurations. We can observe that the capacity grows with the minimum number of transmit and receive antennas. In the case the minimum is the same, the channel with the largest number of receive antennas has the greatest capacity.

As in the case of SISO channel model, we can calculate the constrained capacity as

$$C = I(\mathbf{s}; \mathbf{x}) = H(\mathbf{x}) - H(\mathbf{x}|\mathbf{s}) \tag{4.8}$$

Substituting the entropies by their expressions

$$H(\mathbf{x}|\mathbf{s}) = \log_2(\pi N_0 e)^{n_R} \tag{4.9}$$

and

$$H(\mathbf{x}) = -\mathbb{E}_{\mathbf{x}}\left\{\log_2\left(\frac{1}{2^{n_T M_c}}\sum_{\mathbf{s}}\frac{1}{(\pi N_0)^{n_R}}\exp\left[-\frac{1}{N_0}\left\|\mathbf{x}-\mathbf{H}\mathbf{s}\right\|^2\right]\right)\right\} \tag{4.10}$$

we obtain an expression for the capacity that depends on the channel matrix $\mathbf{H}$

$$C(\mathbf{H}) = -\mathbb{E}_{\mathbf{x}}\left\{\log_2\left(\frac{1}{2^{n_T M_c}}\sum_{\mathbf{s}}\exp\left[-\frac{1}{N_0}\left\|\mathbf{x}-\mathbf{H}\mathbf{s}\right\|^2\right]\right)\right\} - n_R\log_2(e) \tag{4.11}$$

and thus we can calculate the ergodic constrained MIMO capacity as

$$C = -\mathbb{E}_{\mathbf{H}}\mathbb{E}_{\mathbf{x}}\left\{\log_2\left(\frac{1}{2^{n_T M_c}}\sum_{\mathbf{s}}\exp\left[-\frac{1}{N_0}\left\|\mathbf{x}-\mathbf{H}\mathbf{s}\right\|^2\right]\right)\right\} - n_R\log_2(e) \tag{4.12}$$

4.1.2 Spatial diversity

In the previous chapter we introduced the concept of diversity and studied the role of temporal diversity in the performance of channel codes when transmitting over SISO channels. The MIMO channel introduces another dimension, the space, that can be exploited to increase the diversity by sending the same information through all the channel coefficients h_{ij} that connect the transmit and receive antennas. The diversity gain obtained in this way is called *spatial* diversity gain [57, 58] and is especially important in quasi-static MIMO channels because we are limited to a maximum of $n_T n_R$ different channel realizations during the transmission of a codeword.

Spatial diversity can be subdivided in receive antenna diversity and transmit antenna diversity. In the channel model we are considering, receive antenna diversity is always maximum (i.e., equal to n_R) because the information sent from one transmit antenna travels through the n_R possible channels to the receive antennas. Transmit antenna diversity, however, is not always maximized, since not all MIMO schemes send the same information from all the transmit antennas (which is a requisite for sending all the information through all the possible channels).

There exist several techniques that ensure that the maximum transmit diversity is attained. A trivial technique, analogous to using a repetition code to exploit the temporal diversity in the SISO channel, consists in sending the same symbol s from the different transmit antennas sequentially. This simple code is an example of a *Space-Time Code* (STC) [57, 59, 60], i.e., a code that takes into account the spatial dimension, and can be represented in matrix form as

$$\mathbf{S}_{n_T\times n_T} = \begin{bmatrix} s & 0 & \cdots & 0 \\ 0 & s & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & s \end{bmatrix} \tag{4.13}$$

where each row correspond to a transmit antenna and each column to a time instant. Since we are sending one symbol over n_T time instants, the spatial rate of this repetition code is given

by $R_s = 1/n_T$. It can be shown [61] that a Space-Time Coding scheme achieves maximum transmit diversity if the difference between any two different coded matrices has a rank equal to n_T. In this case the code clearly achieves maximum diversity, since the difference matrix between any two codewords

$$\mathbf{D}_{n_T \times n_T} = \begin{bmatrix} s - s' & 0 & \cdots & 0 \\ 0 & s - s' & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & s - s' \end{bmatrix} \tag{4.14}$$

has a rank equal to n_T for any s' different from s. Note that the rate $R_s = 1$ repetition code consisting in transmitting the same symbol through all the transmit antennas at the same time does not provide any transmit diversity gain even though the same information is sent through all the channel realizations, because the difference matrix between two codewords is

$$\mathbf{D}_{n_T \times 1} = \begin{bmatrix} s - s' \\ s - s' \\ \vdots \\ s - s' \end{bmatrix} \tag{4.15}$$

which has a rank equal to one.

A spatial multiplexer can also be viewed as a trivial STC of maximum rate ($R_s = n_T$) that simply distributes n_T input symbols to the transmit antennas without adding redundancy, which results in the coding matrix

$$\mathbf{S}_{n_T \times 1} = \begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ s_{n_T} \end{bmatrix} \tag{4.16}$$

In this case the minimum rank of the difference matrices between codewords is only one, so a spatial multiplexer does not provide any diversity gain. However, if we consider the whole BICM scheme formed by the channel encoder, the modulator and the spatial multiplexer (Figure 4.1), it is possible to achieve maximum diversity. This is because a coded matrix of the whole BICM scheme (which can also be seen as a Space-Time Code) corresponds to the mapped and multiplexed version of a codeword of length N, i.e., a matrix with dimensions $N/(M_c n_T) \times n_T$. If the difference between any pair of those matrices has a rank equal to n_R, the BICM scheme achieves maximum diversity. The problem is that, due to the random nature of the codes we are considering and the large number of columns in the resulting coded matrices, it is difficult to calculate the diversity achieved by a given code without performing computer simulations [62]. This can lead to a bad performance of optimized codes in quasi-static channels, where the maximum achieved transmit diversity is of great importance in high SNRs. In the next chapter we will address this problem by considering the use of Space-Time Codes specifically designed

to obtain the maximum diversity in the MIMO channel. In this chapter we will focus on the BICM scheme over fast-fading channels, where the temporal diversity is sufficiently high so as to overshadow the small spatial diversity gains.

4.2 MIMO detection

4.2.1 Optimum MIMO detection

Optimum MIMO detection can be carried out in a similar way as optimum SISO detection, but we must take into account the entire received vector in which the bit we are interested in was transmitted. Thus, the LLRs for the decoder are calculated as

$$\begin{aligned} L_{\mathrm{ch},k} &= \log \frac{p(\mathbf{x}|c_k=1)}{p(\mathbf{x}|c_k=0)} = \log \frac{p(c_k=1|\mathbf{x})}{p(c_k=0|\mathbf{x})} - \underbrace{\log \frac{p(c_k=1)}{p(c_k=0)}}_{L_k} \\ &= \log \frac{\sum\limits_{\mathbf{s}\in\mathbf{S}_k^1} \exp\left(-\frac{1}{N_0}\|\mathbf{x}-\mathbf{H}\mathbf{s}\|^2 + \sum\limits_{i=1}^{n_T M_c} v_i \frac{L_i}{2}\right)}{\sum\limits_{\mathbf{s}\in\mathbf{S}_k^0} \exp\left(-\frac{1}{N_0}\|\mathbf{x}-\mathbf{H}\mathbf{s}\|^2 + \sum\limits_{i=1}^{n_T M_c} v_i \frac{L_i}{2}\right)} - L_k \end{aligned} \tag{4.17}$$

where $\mathbf{S}_k^b$ represents the set of all the possible transmitted symbol vectors s where bit $c_k = b$ and $v_i = 2c_i - 1$.

Although the expressions for the optimum MIMO and SISO detectors are very similar, the effect of the *a priori* information in the extrinsic information is more important in the MIMO detector, especially when the number of transmit antennas is greater than the number of receive antennas. Figure 4.3 shows the EXIT curves of optimum MIMO detectors for different antenna configurations when using a QPSK constellation and Gray mapping. The SNRs correspond to a constrained capacity limit of 4 bits per channel use. As we can see, an increment in the *a priori* information provides an important increase in the extrinsic information, which forces us to iterate also in the detector to achieve the best performance.

The main problem of the APP detection in a MIMO system is that the complexity grows exponentially not only with the number of bits per modulation symbol, but also with the number of transmitting antennas, being infeasible in many cases of practical interest. Several suboptimum detectors have been proposed to overcome this limitation, following mainly two ideas: reducing the number of vectors used in the summations in the previous equation, or filtering the interferences from the other antennas to isolate each transmitter stream, thus reducing the dimensionality of the problem. We will study two algorithms that follow each of these approaches: the SIC-MMSE detector and the sphere detector.

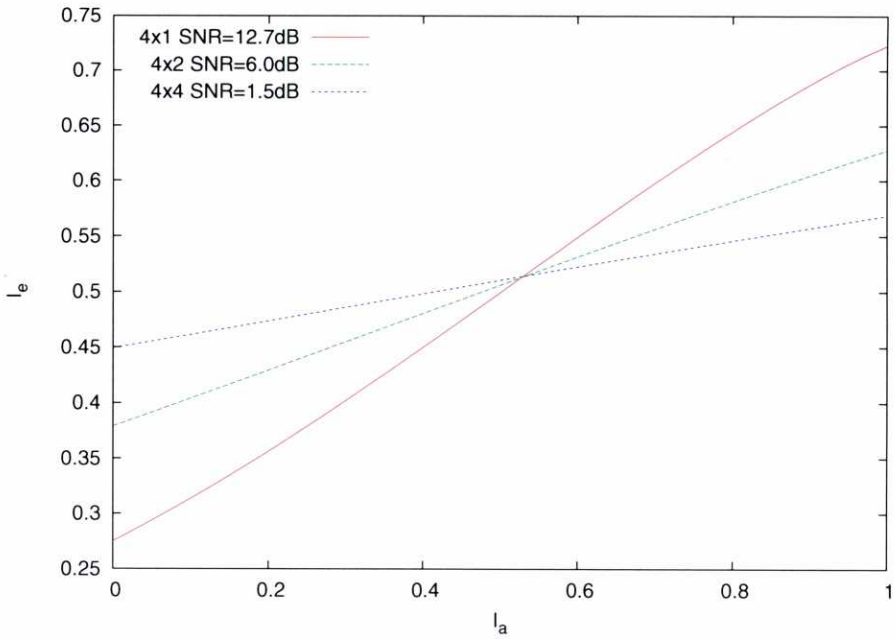


Figure 4.3: EXIT curves for optimum detectors in 4x1, 4x2 and 4x4 MIMO channels.

4.2.2 SIC-MMSE Detector

The SIC-MMSE (Soft Interference Cancellation-Minimum Mean Square Error) detector [63, 64] is a linear detector with $O(n_T^3)$ complexity, similar to the Successive Interference Cancellation with MMSE filtering detector employed in the original Bell Labs Layered Space-Time (BLAST) architecture [55].

The SIC-MMSE detector starts by calculating a soft estimation of each of the transmitted symbols using the extrinsic L-values of the coded bits provided by the decoder

$$\bar{s}_j = \sum_{s \in S} p(s_j = s)s = \sum_{s \in S} s \prod_{i=1}^{M_c} [1 + \exp(-v_{ji} L_{ji})]^{-1}, \quad j = 1, \ldots, n_T \tag{4.18}$$

where S is the set of constellation symbols, v_{ji} is the i-th bit (± 1) of the symbol s_j and L_{ji} the L-value corresponding to that bit. Then, the interferences are subtracted from the received vector for each transmit antenna

$$\mathbf{x}_k = \mathbf{x} - \sum_{j=1, j \neq k}^{n_T} \bar{s}_j \mathbf{h}_j, \quad k = 1, \ldots, n_T \tag{4.19}$$

where $\mathbf{h}_j$ denotes the j-th column of the channel matrix $\mathbf{H}$. After this soft cancellation, an MMSE filter is applied to each vector $\mathbf{x}_k$ to obtain the final estimation of the transmitted symbol

$$\hat{s}_k = \mathbf{w}_k^H \mathbf{x}_k \tag{4.20}$$

In Appendix B we show that the filter that minimizes the mean square error $\sigma_e^2 = \mathbb{E}\{|s_k - \hat{s}_k|^2\}$ can be calculated as

$$\mathbf{w}_k = \left(\mathbf{H}\boldsymbol{\Delta}_k\mathbf{H}^H + \frac{N_0}{E_s}\mathbf{I}_{n_R}\right)^{-1}\mathbf{h}_k \tag{4.21}$$

where

$$\boldsymbol{\Delta}_k = \text{diag}\left[\frac{\sigma_{s_1}^2}{E_s}, \ldots, \frac{\sigma_{s_{k-1}}^2}{E_s}, 1, \frac{\sigma_{s_{k+1}}^2}{E_s}, \ldots, \frac{\sigma_{s_{n_T}}^2}{E_s}\right] \tag{4.22}$$

being

$$\sigma_{s_j}^2 = \sum_{s\in S} \|\bar{s}_j - s\|^2 p(s = s_j) \tag{4.23}$$

It can be shown that the resulting estimations $\hat{s}_k$ are distributed as [63]

$$p(\hat{s}_k|s_k) \sim \mathcal{N}(\mu_k s_k, \eta_k^2) \tag{4.24}$$

where

$$\mu_k = \mathbf{h}_k^H\left(\frac{N_0}{E_s}\mathbf{I}_{n_R} + \mathbf{H}\boldsymbol{\Delta}_k\mathbf{H}^H\right)^{-1}\mathbf{h}_k \tag{4.25}$$

$$\eta_k^2 = \mu_k - \mu_k^2 \tag{4.26}$$

The extrinsic LLRs of the coded bits can then be calculated as

$$L_{\text{ch},jk} = \log\frac{\displaystyle\sum_{s\in S_k^1}\exp\left(-\frac{1}{\eta_k^2}\|\hat{s} - \mu_k s_k\|^2 + \sum_{i=1}^{M_c} v_{ji}\frac{L_{ji}}{2}\right)}{\displaystyle\sum_{s\in S_k^0}\exp\left(-\frac{1}{\eta_k^2}\|\hat{s} - \mu_k s_k\|^2 + \sum_{i=1}^{M_c} v_{ji}\frac{L_{ji}}{2}\right)} - L_{jk} \tag{4.27}$$

where S_k^b is the set of constellation symbols in which the k-th bit is equal to b.

It is interesting to note that the filters $\mathbf{w}_k$ depend on the knowledge of the different transmitted symbols, expressed as their variance. Thus, if a $\sigma_{s_j}^2$ is close to 0, meaning that the soft estimate $\bar{s}_j$ of the symbol transmitted from the antenna j is accurate, the corresponding transmit antenna will not be filtered, since the interference has been completely eliminated in the interference cancellation step. Conversely, if the variance is close to E_s, the interference corresponding to that antenna must be filtered, since the symbol estimation is not accurate. This way the SIC-MMSE algorithm evolves from a pure MMSE filter in the first iteration, when there is no knowledge of the symbols coming from the decoder, to a pure interference cancellation algorithm when the decoder provides the detector with perfect symbol estimations.

Figure 4.4 shows the EXIT characteristic of a SIC-MMSE detector in a 4×4 QPSK MIMO channel. We can obtain the EXIT curves for the different MIMO detectors following the procedure described in Section 3.2. We can see that when the a priori information is accurate

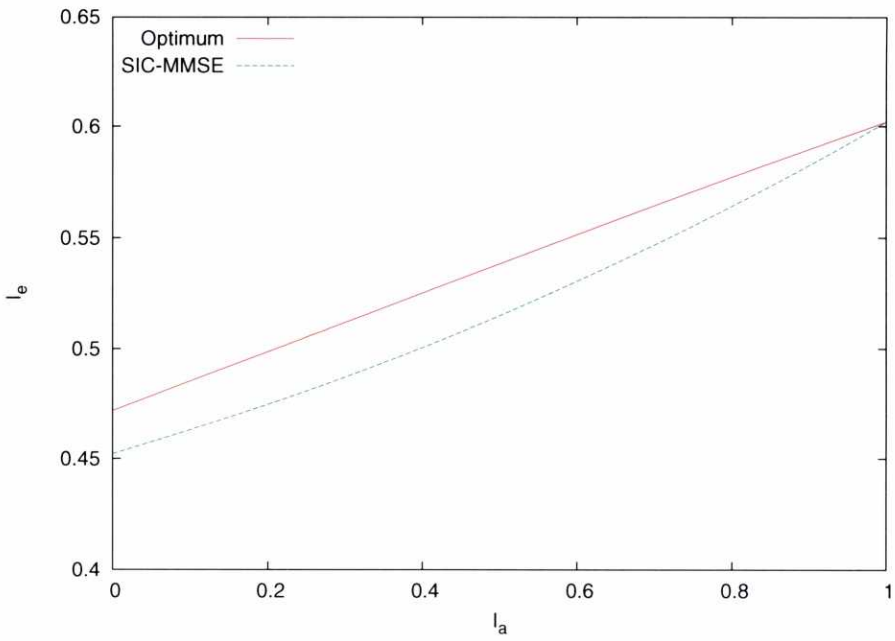


Figure 4.4: EXIT characteristic of the SIC-MMSE detector for a 4×4 QPSK MIMO channel and $E_s/N_0 = 2.0$dB.

(i.e., $I_a = 1$), the performance equals that of the optimum detector, thanks to the perfect cancellations. With values for the *a priori* information lower than one, however, the curve is clearly below the optimum curve, which indicates that the performance will be significantly worse.

4.2.3 ML List Sphere Detector

The idea of List Sphere Detection (LSD) is to approximately calculate Eq. (4.17), whose terms in the summations are in fact, except for an additive constant α, the logarithm of the APP of a transmitted vector s

$$\log p(\mathbf{s}|\mathbf{x}) = -\frac{1}{N_0}\|\mathbf{x} - \mathbf{H}\mathbf{s}\|^2 + \sum_{i=1}^{n_T M_c} s_i \frac{L_i}{2} + \alpha, \tag{4.28}$$

using a list of just N_{cand} candidate vectors s.

The Maximum Likelihood List Sphere Detector (ML LSD) [65] constructs the list using the N_{cand} most likely (ML) vectors. That is, those that maximize just the first term in Eq. (4.28)

$$\log p(\mathbf{x}|\mathbf{s}) = -\frac{1}{N_0}\|\mathbf{x} - \mathbf{H}\mathbf{s}\|^2 + \alpha' \tag{4.29}$$

The list is constructed using a modified version of the Sphere Detector (SD) [66, 67], which efficiently finds the constellation points inside a hypersphere centered on the unconstrained ML estimate of the transmitted vector.

First note that the squared Euclidean distance between vectors $\mathbf{x}$ and $\mathbf{Hs}$ can be expressed as

$$\|\mathbf{x} - \mathbf{Hs}\|^2 = (\mathbf{s} - \hat{\mathbf{s}})^H \mathbf{H}^H \mathbf{H}(\mathbf{s} - \hat{\mathbf{s}}) + \mathbf{x}^H(\mathbf{I} - \mathbf{H}(\mathbf{H}^H\mathbf{H})^{-1}\mathbf{H}^H)\mathbf{x} \tag{4.30}$$

where $\hat{\mathbf{s}} = (\mathbf{H}^H\mathbf{H})^{-1}\mathbf{H}^H\mathbf{x}$ is the unconstrained ML estimate of the transmitted vector and $(\cdot)^H$ has been used for denoting matrix Hermitian (conjugate transpose). Thus, finding the points with highest likelihood is equivalent to finding the points with smallest value of $(\mathbf{s} - \hat{\mathbf{s}})^H \mathbf{H}^H \mathbf{H}(\mathbf{s} - \hat{\mathbf{s}})$.

To avoid an exhaustive search through all the possible transmited vector symbols, the SD finds the vectors that fall inside a sphere of a given radius r

$$(\mathbf{s} - \hat{\mathbf{s}})^H \mathbf{H}^H \mathbf{H}(\mathbf{s} - \hat{\mathbf{s}}) \leq r^2. \tag{4.31}$$

which can be efficiently done through triangularization of matrix $\mathbf{H}^H\mathbf{H}$. For instance, Cholesky decomposition ($\mathbf{H}^H\mathbf{H} = \mathbf{U}^H\mathbf{U}$, with $\mathbf{U}$ upper triangular) is used in [65], although QR decomposition of matrix $\mathbf{H}$ (i.e. $\mathbf{H} = \mathbf{QR}$, with $\mathbf{Q}$ unitary ($\mathbf{Q}^H\mathbf{Q} = \mathbf{I}$) and $\mathbf{R}$ upper triangular) is more efficient [68]. Triangularization allows to rewrite Eq. (4.31) as

$$(\mathbf{s} - \hat{\mathbf{s}})^H \mathbf{U}^H \mathbf{U}(\mathbf{s} - \hat{\mathbf{s}}) = \sum_{i=1}^{n_T} u_{ii}^2 \left| s_i - \hat{s}_i + \sum_{j=i+1}^{n_T} \frac{u_{ij}}{u_{ii}}(s_j - \hat{s}_j) \right|^2 \leq r^2, \tag{4.32}$$

which enables us to find the points with highest likelihood by creating a tree of depth n_T. Starting with the n_T-th antenna, we have

$$|s_{n_T} - \hat{s}_{n_T}| \leq \frac{r}{u_{n_T n_T}}. \tag{4.33}$$

Let us assume a 2^{M_c}-PSK constellation, although the algorithm can be easily extended to other constellations such as QAM. We can rewrite $s_{n_T} = r_c e^{i\theta_{n_T}}$, where r_c is the radius of the PSK constellation and

$$\theta_{n_T} \in \{0, 2\pi/2^{M_c}, \ldots, 2\pi(2^{M_c} - 1)/2^{M_c}\} \tag{4.34}$$

is the phase. We also rewrite $\hat{s}_{n_T} = \hat{r}_{c,n_T} e^{i\hat{\theta}_{n_T}}$. As a consequence, (4.33) becomes

$$|s_{n_T} - \hat{s}_{n_T}|^2 = r_c^2 + \hat{r}_{c,n_T}^2 - 2r_c\hat{r}_{c,n_T}^2 \cos(\theta_{n_T} - \hat{\theta}_{n_T}) \leq \frac{r^2}{u_{n_T n_T}^2}, \tag{4.35}$$

which yields

$$\cos(\theta_{n_T} - \hat{\theta}_{n_T}) \geq \frac{1}{2r_c\hat{r}_{c,n_T}} \left[r_c^2 + \hat{r}_{c,n_T} - \frac{r^2}{u_{n_T n_T}^2} \right] \triangleq \eta_{n_T}. \tag{4.36}$$

If $\eta_{n_T} > 1$, there are no allowable points for this antenna. If $\eta_{n_T} < -1$, all the points of the constellation are candidate solutions. For $-1 \leq \eta_{n_T} \leq 1$ we have

$$|\theta_{n_T} - \hat{\theta}_{n_T}| \leq \cos^{-1} \eta_{n_T}, \tag{4.37}$$

so the range of allowable points is

$$\left\lceil \frac{2^{M_c}}{2\pi}(\hat{\theta}_{n_T} - \cos^{-1} \eta_{n_T}) \right\rceil \leq \frac{2^{M_c}}{2\pi}\theta_{n_T} \leq \left\lfloor \frac{2^{M_c}}{2\pi}(\hat{\theta}_{n_T} + \cos^{-1} \eta_{n_T}) \right\rfloor. \tag{4.38}$$

For each symbol s_{n_T} in antenna n_T obeying condition (4.33) we will have several conditions to be fulfilled by candidates for antenna $n_T - 1$, i.e.

$$\left| s_{n_T-1} - \hat{s}_{n_T-1} + \frac{u_{n_T-1,n_T}}{u_{n_T n_T}}(s_{n_T} - \hat{s}_{n_T}) \right|^2 \leq \frac{r^2 - u_{n_T n_T}^2 |s_{n_T} - \hat{s}_{n_T}|^2}{u_{n_T-1,n_T-1}^2}. \tag{4.39}$$

If we denote

$$\hat{s}'_{n_T-1} \triangleq \hat{s}_{n_T-1} - \frac{u_{n_T-1,n_T}}{u_{n_T n_T}}(s_{n_T} - \hat{s}_{n_T}) = \hat{r}_{c,n_T-1} e^{i\hat{\theta}_{n_T-1}}$$

we have

$$\left| s_{n_T-1} - \hat{s}'_{n_T-1} \right|^2 = r_c^2 + \hat{r}_{c,n_T-1}^2 - 2 r_c \hat{r}_{c,n_T-1} \cos(\theta_{n_T-1} - \hat{\theta}_{n_T-1}) \leq \frac{r^2 - u_{n_T n_T}^2 |s_{n_T} - \hat{s}_{n_T}|^2}{u_{n_T-1,n_T-1}^2}, \tag{4.40}$$

which yields

$$\cos(\theta_{n_T-1} - \hat{\theta}_{n_T-1}) \geq \frac{1}{2 r_c \hat{r}_{c,n_T-1}} \left[r_c^2 + \hat{r}_{c,n_T-1}^2 - \frac{r^2 - u_{n_T n_T}^2 |s_{n_T} - \hat{s}_{n_T}|^2}{u_{n_T-1,n_T-1}^2} \right] \triangleq \eta_{n_T-1}, \tag{4.41}$$

and the process continues until we reach the first antenna. This way we build a tree with the vectors whose distance to the unconstrained ML estimate is smaller than r. (Figure 4.6). Then we have to choose the N_{cand} with highest likelihood by exhaustive search among these vectors and use their likelihoods for computing Eq. (4.17). If the number of vectors in the tree is smaller than the desired number N_{cand}, we must increase the radius and the search process must be started again. On the other hand, if r is chosen too large, building of the tree and searching for the best N_{cand} vectors will take longer. Therefore, the choice of r clearly determines the efficiency of the algorithm. As stated in [65], a possible choice for the radius is

$$r^2 = 2 N_0 K n_R - \mathbf{x}^H (\mathbf{I} - \mathbf{H}(\mathbf{H}^H \mathbf{H})^{-1} \mathbf{H}^H) \mathbf{x}, \tag{4.42}$$

where $K \geq 1$ is chosen so that we are reasonably sure to find at least N_{cand} candidate points in the first search. As we will see in Section 4.4, the performance of the decoding process depends on the size of the candidate list (N_{cand}). When N_{cand} is high we will have to compute more likelihoods; more processing will be needed in the detector but the performance will be better. This can also be clearly seen in the EXIT curve of the detector (Figure 4.5), that gets

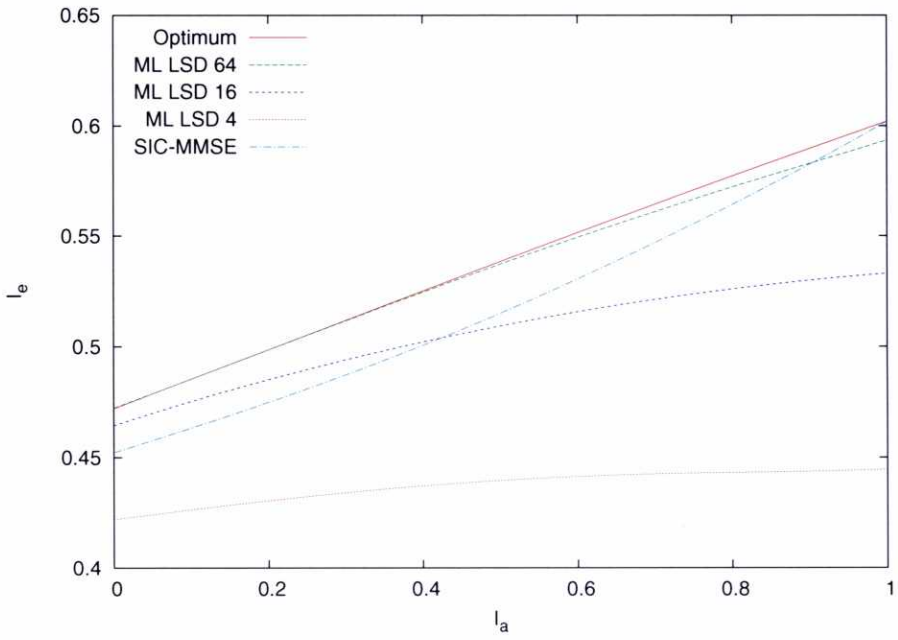


Figure 4.5: EXIT characteristic of SIC-MMSE and ML-LSD for a 4×4 QPSK MIMO channel and $E_s/N_0 = 2.0$dB.

closer to the optimum curve as we increase the number of candidates. When using low values of N_{cand} the performance degrades due to the smaller amount of information processed by the algorithm. Therefore, there is a tradeoff between efficiency and performance in the choice of N_{cand}. Keeping the number of candidates of the sphere detector constant while increasing the number of antennas leads to a linear increase in complexity.

Notice that a candidate in the list is a vector of symbols and no consideration is made about the value of each bit in the candidate, as required in Eq. (4.17). Considering only one candidate list results in the chance of having no candidates for computing one of the sums in Eq. (4.17). As suggested in [65], this can be solved by assigning a minimum value (instead of zero) to the sum and by applying a threshold lower than that used in the SPA decoder to the obtained LLRs, which leads to additional information loss.

4.2.4 MAP List Sphere Detector

An alternative approach to the ML list sphere detector consists in building the list of the N_{cand} vectors with highest APP, according to Eq. (4.28). This approach, proposed by Vikalo *et al.* [69], is termed MAP LSD. With the MAP LSD algorithm we are actually finding the most significant terms for the summations involved in the computation of the channel LLR. However, when used in an iterative receiver this method has to perform the complete tree search each time

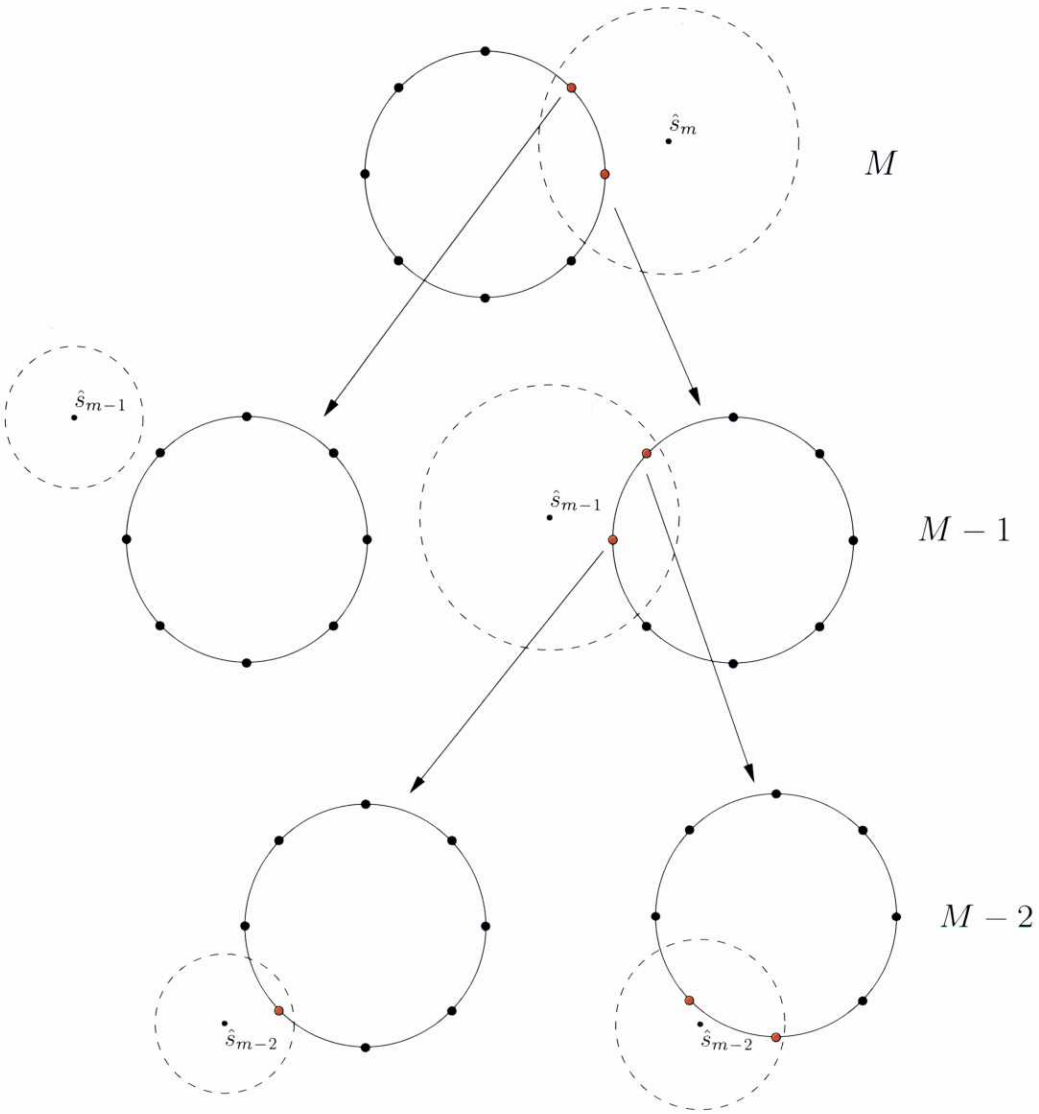


Figure 4.6: Search tree of a sphere detector with a 8-PSK constellation.

new *a priori* bit probabilities are available from other receiver stages. The complexity of ML LSD is smaller, since in this case the search for the most likely vectors is performed only once and the corresponding likelihoods stored for their use in successive iterations.

If we keep fixed the number of operations when comparing both methods, ML LSD allows to employ a larger candidate list, but also note that the size of the candidate list in MAP LSD can be reduced with the number of detection iterations. Rigorous comparison of both methods under the constraint of equal number of operations is difficult [69] but in our simulations we have seen that MAP LSD allows to attain a better performance at a reasonable complexity.

Besides the problem of the possibility of having no candidates in either the numerator or the denominator, building only one candidate list gives rise to another problem in the MAP LSD detector: positive feedback. Although the *a priori* LLR of the bit being processed, L_k, is substracted in Eq. (4.17) to obtain the extrinsic channel LLR, it is implicitly considered in the construction of the candidate list. Note that a high *a priori* LLR of the bit being considered leads to the inclusion in the list of those candidates for which that bit has the value suggested by the *a priori* bit LLR. This effect is more pronounced as the size of the candidate list gets smaller and, clearly, is not compensated by just substracting the *a priori* LLR. This positive feedback is clearly shown by the fact that the mutual information of the output LLRs produced by the MAP LSD (see Fig. 4.7) is higher than that of the optimum APP detector, which indicates that additional information is contained in these output LLRs. Indeed, this mutual information does not constitute a *true* EXIT chart because it does not only measure extrinsic information but also *a priori* information. Since decoding convergence is determined by just the extrinsic information transfer, it would be necessary to cancel the contribution of the *a priori* information from the output LLRs to compute the actual detector EXIT chart. However, this cannot be done because *a priori* information is implicitely introduced by the particular way the candidate list is built.

To overcome this limitation, we consider what we call *Extrinsic* LSD, which consists in rewriting Eq. (4.17) as

$$L_{\text{ch}} = \log \frac{\sum_{\mathbf{s}\in\mathbf{S}_k^1} \exp\left(-\frac{1}{N_0}\|\mathbf{x}-\mathbf{H}\mathbf{s}\|^2 + \sum_{i=1,i\neq k}^{n_T M_c} v_i \frac{L_i}{2}\right)}{\sum_{\mathbf{s}\in\mathbf{S}_k^0} \exp\left(-\frac{1}{N_0}\|\mathbf{x}-\mathbf{H}\mathbf{s}\|^2 + \sum_{i=1,i\neq k}^{n_T M_c} v_i \frac{L_i}{2}\right)} \tag{4.43}$$

Then, separate lists for each bit in a symbol vector are constructed, by searching for the vectors that maximize the expressions in the numerator and the denominator in this equation. This means $2n_T M_c$ candidate lists for computing the bit channel LLRs in a symbol vector, instead of only one as in MAP LSD. The complexity of this approach is $2n_T M_c$ times higher than that of MAP LSD, but its expected performance is better, especially when the number of candidates is low. In addition, since it does not suffer from positive feedback, due to the fact that we are not using the *a priori* probability of a given bit in the construction of the candidate lists for that

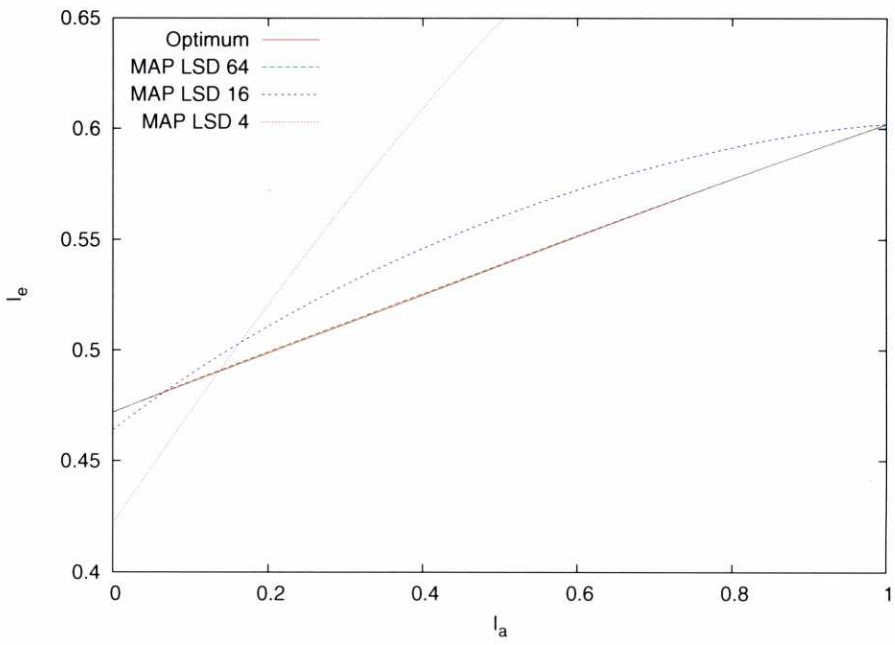


Figure 4.7: EXIT characteristic of MAP-LSD for a 4×4 QPSK MIMO channel and $E_s/N_0 = 2.0$dB.

bit, its *true* EXIT function can be computed, enabling code design (Figure 4.8). However, note that if MAP LSD is used with a high number of candidates, the effect of the *a priori* LLR of the bit under consideration in the obtained list is less severe. This motivates us to use MAP LSD in the final implementation shown in Section 4.4, even though code design is performed using the EXIT function of the Extrinsic LSD.

4.3 Optimization of SCLDGM codes for MIMO channels

In Section 3.4 we described the optimization of SCLDGM codes for SISO channels using EXIT functions. The application of that method to MIMO channels and suboptimum detectors is straightforward: we simply need to substitute the EXIT function of the optimum SISO detector with that of the MIMO detector for which we are optimizing the code. However, in this case it is important to perform several detector iterations during the decoding process. The reason behind this is that in MIMO detectors, the increase in the a priori information provides a significant increment of the extrinsic information that can be used by the decoder. This can be clearly seen in the slope of the EXIT curves in Figures 4.4-4.8. Since the complexity of the calculation of the channel LLRs is much higher than that of the calculation of the messages between variable and check nodes, we will perform one detector iteration for every ten decoder iterations.

Figure 4.9 shows the evolution of the extrinsic information of different messages in the

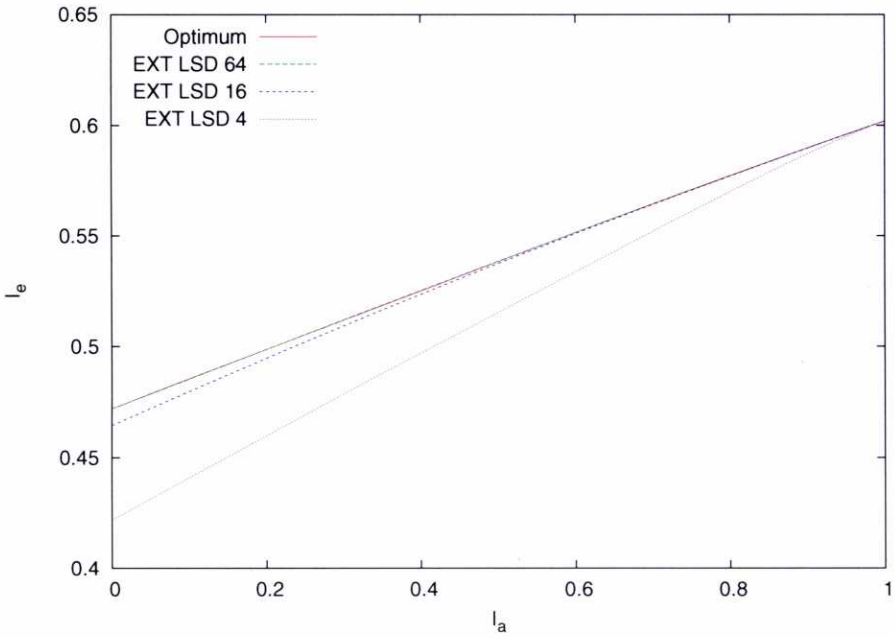


Figure 4.8: EXIT characteristic of Ext-LSD for a 4×4 QPSK MIMO channel and $E_s/N_0 = 2.0$dB.

graph of a SCLDGM code in a 4x4 MIMO channel. The figure also shows the evolution of the extrinsic information passed by the detector to the decoder. The "jumps" observed every ten decoder iterations are due to the increase in the information that comes from the detector.

We applied the optimization procedure for the design of good rate $1/2$ regular SCLDGM codes for different antenna configurations, modulation formats and detectors. We carried out an exhaustive search, varying the code parameters between the following limits: $0.01 \leq p \leq 0.05$, with increments of 0.005; $3 \leq d_u^{f^1} \leq 5$; $2 \leq d_u^{f^2} \leq 30$; and $2 \leq d_{p^1}^{f^2} \leq 40$, fixing also the maximum value of d_{f^2} to 10. The obtained codes are presented in Table 4.1, together with their convergence threshold predicted by EXIT analysis. Note the extraordinary performance of the resulting codes, since the gap to the constrained-input ergodic capacity limit is less than 1dB in many cases. It can also be observed that the distance to the capacity of symmetric antenna configurations is lower than that of systems with more transmit anntenas than receive antennas. This suggests that the SCLDGM scheme is better suited for systems (channels, bit-to-symbol mappings and detectors) whose EXIT curves are not very steep. Similar results can be observed in other graph-based codes [41, 40].

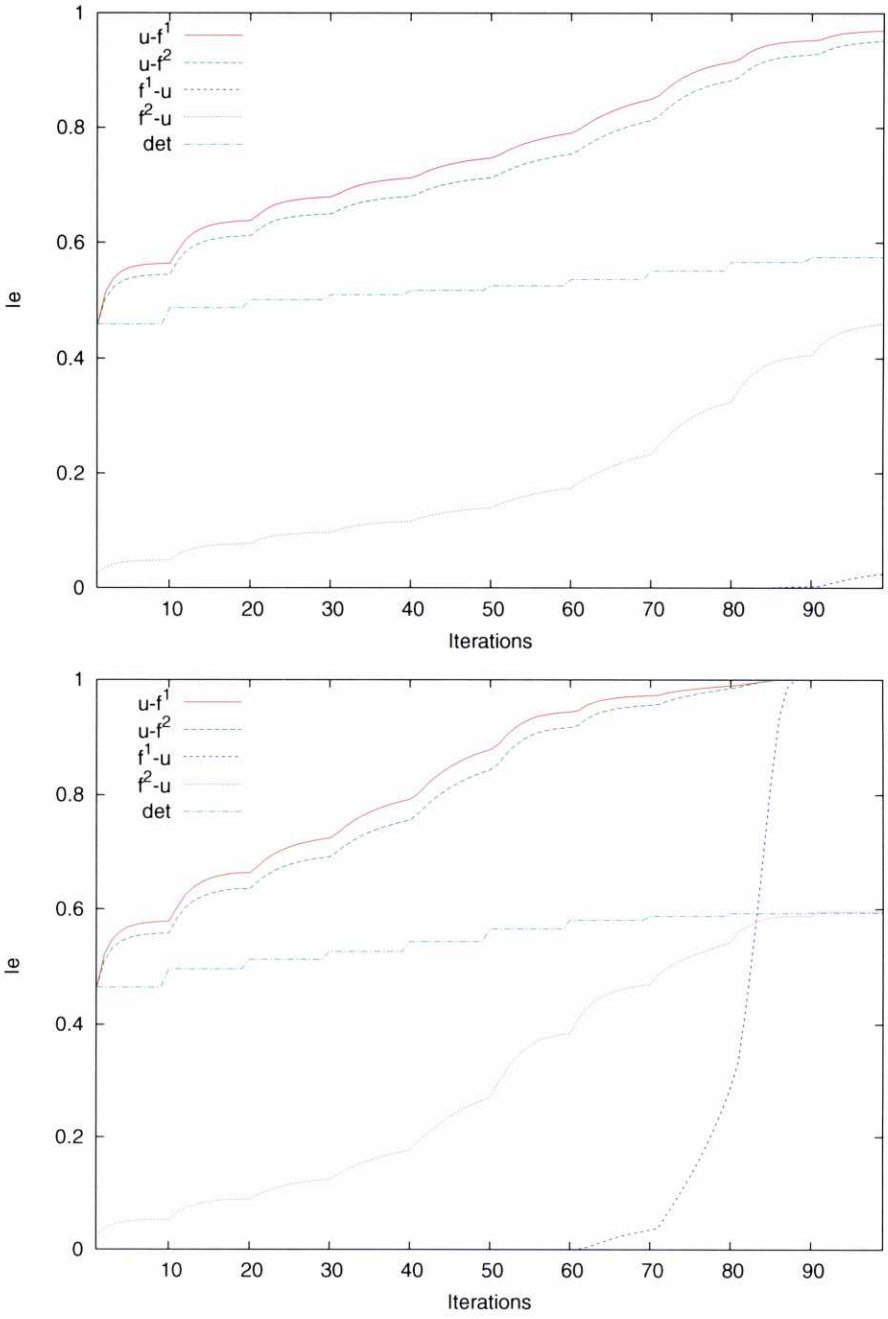


Figure 4.9: Mutual information trajectories for a 4×4 QPSK optimized SCLDGM code. $E_b/N_0 = 1.75$ (top) and $E_b/N_0 = 1.80$dB (bottom).

Antennas	Modul.	Detector	$p(\%)$	$d_u^{f^1}$	$d_u^{f^2}$	$d_{p^1}^{f^2}$	Thresh (dB)	UCL	CCL
1×1	QPSK	Optimum	2%	3	6	6	2.3	1.05	1.8
2×1	QPSK	Optimum	4.5%	3	4	20	3.89	2.55	3.3
2×2	QPSK	Optimum	3%	3	5	28	1.97	1.2	1.55
4×1	QPSK	Optimum	4.6%	3	2	18	8.5	6.25	6.7
4×2	QPSK	Optimum	2%	3	4	20	3.8	2.65	3.0
4×4	QPSK	Optimum	2%	3	5	15	1.8	1.2	1.5
2×2	16QAM	Optimum	4.5%	3	4	20	4.8	3.7	4.1
		ML 128/64	4.5%	3	4	20	4.8/4.9		
		EXT 32/16	4.5%	3	4	20	4.8/4.9		
		MMSE	2%	4	5	40	5.5		
2×2	64QAM	ML 512/256	2%	4	5	23	7.6/7.6	6.1	6.65
		ML 128	2.5%	3	5	20	7.8		
		EXT 128/64/32	4%	3	4	22	7.6/7.6/7.7		
		MMSE	1.5%	4	5	37	8.5		
4×4	16QAM	ML 512/256/128	2.5%	3	5	20	5.1/5.1/5.3	3.8	4.1
		EXT 128/64/32	3.5%	3	4	16	4.8/4.9/5.0		
		MMSE	2.5%	3	4	36	5.7		
4×4	64QAM	ML 256/128/64	2.5%	3	5	20	8.9/9.2/9.7	6.3	-
		EXT 128/64/32	2.5%	3	4	24	7.8/7.9/8.1		
		MMSE	1.5%	5	5	40	10.3		

Table 4.1: Degree profiles of rate $1/2$ optimized SCLDGM codes for different MIMO channels and detectors. "Thresh" stands for the EXIT analysis convergence threshold. "UCL" and "CCL" are, respectively, the Unconstrained-input and Constrained-input Ergodic Capacity Limits.

4.4 Simulation Results

Computer simulations were carried out to evaluate the performance of the obtained SCLDGM codes in practical cases where blocks of finite length are transmitted. We assumed that the MIMO channel changes randomly and independently from one symbol vector to another (fast fading ergodic channel) and is known at the receiver.

Fig. 4.10 compares the performance of regular and irregular SCLDGM codes for 4×4, QPSK and optimum detection. The optimized irregular SCLDGM code is given by

$$p = 3\% \quad d_u^{f^1} = 3 \quad d_u^{f^2} = \{4, 5, 21\} \quad a_u^{f^2} = \{0.5, 0.4, 0.1\} \quad d_{p^1}^{f^2} = 10 \tag{4.44}$$

whereas the regular code is that of Table 4.1. Their theoretical thresholds are 1.75dB and 1.80dB, respectively. Two block sizes are considered: $N = 20000$ and $N = 100000$ coded bits. As explained in the previous chapter, practical convergence thresholds are higher than those predicted by EXIT function analysis, because EXIT analysis assumes infinite-length data blocks. Indeed, notice that the higher the block length is, the closer the convergence threshold is to the theoretical value calculated with the EXIT functions. The figure also shows that the difference in performance between regular and irregular codes is very small (around 0.05dB). For this reason only regular SCLDGM codes are considered in the rest of this section, since this greatly reduces the search space and the code regularity usually allows easier implementations.

Code performance for asymmetric antenna configurations (4×1 and 4×2 QPSK) and $N = 100000$ coded bits is shown in Fig. 4.11. In this case the detector EXIT function has a positive slope, and it is very different from that of a SISO channel (which is an almost horizontal straight line). This means that code design is important to properly match the detector EXIT function. Indeed, notice the difference in performance between an SCLDGM code optimized for the SISO case (see Table 4.1) but applied over the MIMO channel and an SCLDGM code specifically designed for the antenna configuration and modulation format under consideration. For instance, for 4×2 QPSK and a BER$= 10^{-4}$, the performance difference is around 1.3dB. This gap is much larger (around 5dB) for the 4×1 system. This is explained by the fact that, for the same number of transmitting antennas, using less receiving antennas makes the detector EXIT function steeper and, therefore, it is critical that the code matches it (and not the horizontal like characteristic of SISO channels). Notice that the convergence thresholds obtained through simulation practically overlap with the theoretical ones presented in Table 4.1, and they are very close to the constrained capacities (gap of 0.8dB and 1.8dB for the 4×2 and 4×1 systems, respectively).

We next focus on comparing the performance of the SIC-MMSE and list sphere detectors with respect to that of optimum MAP detection when the optimized codes for each respective detector are utilized. Fig. 4.12 shows the performance curves for 2×2 antennas using 16-QAM and a block length of $N = 20000$ coded bits (which will be the length used in the sequel). Optimum detection needs to process $2^{n_T M_c} = 2^8 = 256$ possible symbol vectors. See how using ML LSD with half of the total number of candidates (128) renders near optimum performance, while decreasing the number of candidates to one quarter of the total (64) results

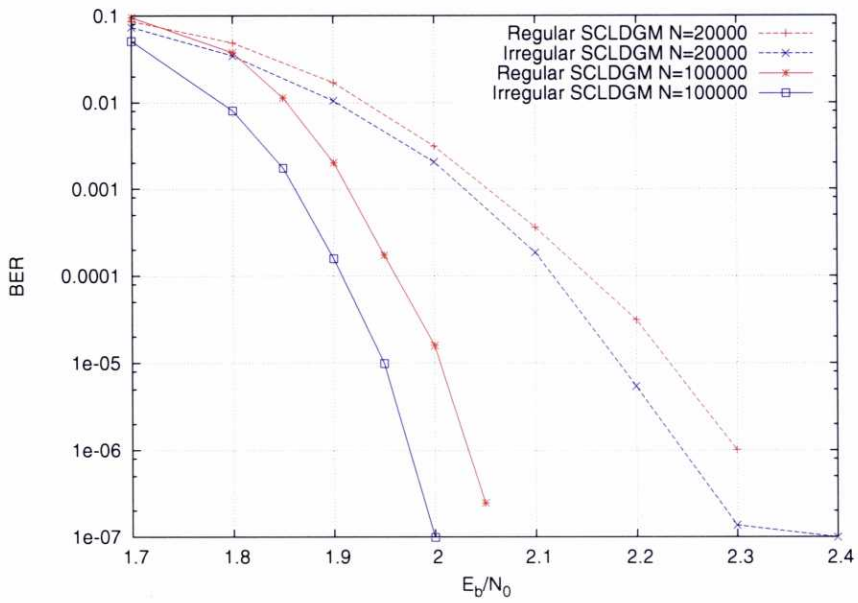


Figure 4.10: Performance of rate $1/2$ regular and irregular SCLDGM codes optimized for a 4×4 QPSK MIMO channel and optimum detection.

in little performance degradation. We can also see that MAP LSD needs only $1/8$ of the total number of candidates (32) to provide near optimum performance (instead of the 128 candidates required by ML LSD). The convergence threshold obtained through simulations is very close to the theoretical one in Table I, and only 1.0dB away from the constrained capacity. The SIC-MMSE detector performs at approximately 0.8dB away from the optimum detector, as predicted by the EXIT analysis.

Optimum MAP detection is not feasible when considering 2×2 64-QAM, since the total number of possible symbol vectors is $2^{n_T M_c} = 2^{12} = 4096$. However, very good performance is obtained with suboptimum LSD, as shown in Fig. 4.13. For a target BER of 10^{-5}, the best performance is achieved by MAP LSD with 128 candidates, which requires an $E_b/N_0 = 7.9$dB, only 0.3dB away from the theoretical threshold of Table I and 1.25dB away from the constrained-input ergodic capacity. Halving the number of candidates (32) results in 0.4dB degradation. Note how eight times the number of candidates (256) is needed to obtain similar performance if ML LSD is used. Again, the SIC-MMSE detector performs the furthest away from the capacity limits.

Fig. 4.14 shows the perfromance of a 4×4 16-QAM MIMO system. The total number of candidates in this case is 65536. The best performance is obtained by MAP LSD with 256 candidates ($1/256$ of the total), requiring an $E_b/N_0 = 5.15$dB for a target BER of 10^{-5}, which is 0.35dB away from the theoretical threshold in Table I and only 1.05dB away from the

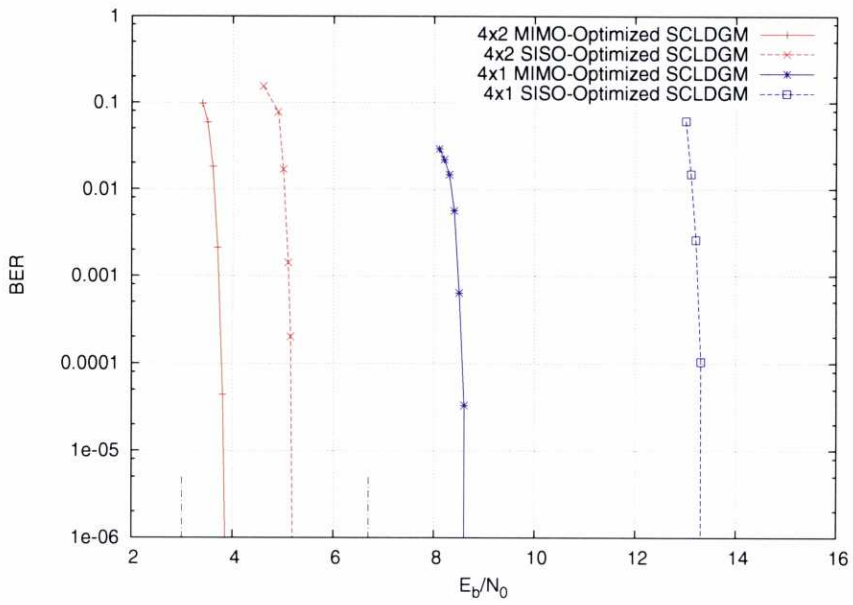


Figure 4.11: Performance comparison between SISO-optimized and MIMO-optimized rate $1/2$ regular SCLDGM codes for 4×2 and 4×1 QPSK MIMO channels and optimum detection.

constrained-input ergodic capacity. Very little performance degradation (0.1dB) results from considering half (128) of such number of candidates. However, further reducing the size of the candidate list to 64 increases the required E_b/N_0 up to 5.7dB. See how 512 candidates are needed when using ML LSD to obtain a performance similar to that of MAP LSD with 64 candidates. Using ML LSD with less candidates results in more performance degradation.

We can also see that for small numbers of candidates (with respect to the total number of possible vectors), the gaps between the theoretical thresholds and those observed in the simulations increase. This can be clearly seen in the performance of the ML LSD with 128 candidates, that is worse than that of the SIC-MMSE detector even though the predicted threshold is lower. This can be explained by realizing that the pdf of the output LLRs of ML/MAP LSD is different from the Gaussian assumption in our optimization procedure. Using a low number of candidates results in many LLRs taking extreme values because the list contains much more candidates for the numerator than for the denominator (or *vice versa*). LLR values greater than 8 (less than -8) are truncated to $+8$ (-8), producing a pdf with peaks in these two extreme values. Since this "clipping" is not needed in the SIC-MMSE detector, its actual threshold remains very close to the predicted one.

Similar conclusions can be drawn from an scenario with a higher number of possible symbol vectors, such as 4×4 64-QAM, where the number of possible candidates is greater than 16 million. Fig. 4.15 shows that in this case differences in performance are larger but maintain

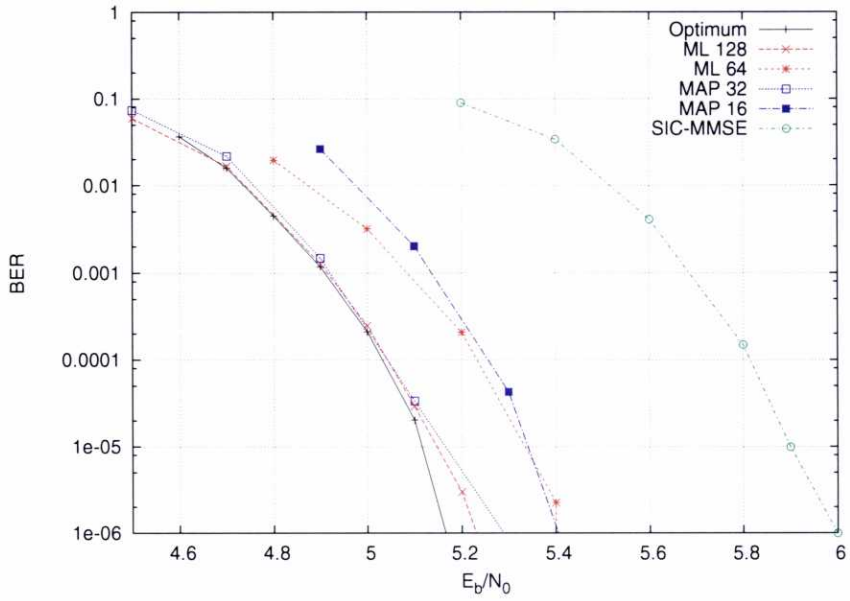


Figure 4.12: Performance of rate $1/2$ regular SCLDGM codes optimized for a 2×2 16-QAM MIMO channel with different types of detectors and number of candidates. The constrained and unconstrained input capacity limits are at $E_b/N_0 = 4.1$dB and $E_b/N_0 = 3.7$dB, respectively.

the same relationships observed for 4×4 16-QAM. Again, the predicted thresholds for the ML and MAP LSDs get less accurate as we decrease the number of candidates, due to the non-Gaussianity of the detector LLRs. In the case of the MAP LSD, this divergence is also caused by the fact that the lower the number of candidates in MAP LSD is, the more different this detector is from the Extrinsic LSD which is used in the code optimization. Nevertheless, the performance of the SCLDGM scheme with the MAP LSD is extremely good.

Finally, let us stress the importance of optimizing the code for the specific LSD that we will be using. Fig. 4.16 shows the effect of code and detector mismatch for a 4×4 16-QAM system. First, notice the performance difference between a code optimized for the used detector and a code optimized for a different detector: for a target BER of 10^{-4}, employing MAP LSD with 32 candidates and its optimized code requires an $E_b/N_0 = 5.9$dB, while using the same detector but with a code optimized for ML LSD with 128 candidates needs an $E_b/N_0 = 6.45$dB. If ML LSD with 128 candidates is considered, using its optimized code requires an $E_b/N_0 = 6.25$dB, while using a code optimized for MAP LSD with 32 candidates needs an $E_b/N_0 = 6.5$dB. In this latter case, the effect of code and detector mismatch is less severe than in the former.

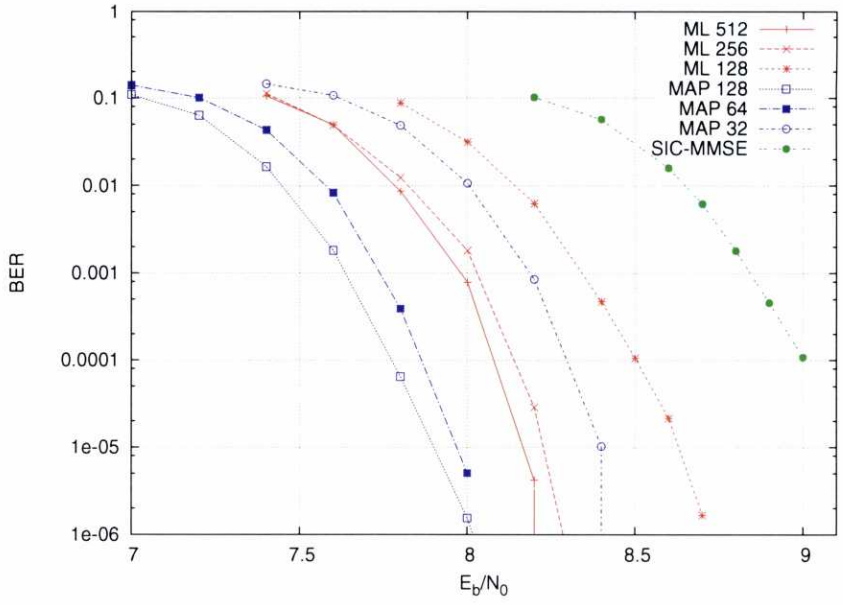


Figure 4.13: Performance of rate $1/2$ regular SCLDGM codes optimized for 2×2 64-QAM MIMO channel with different types of detectors and number of candidates. The constrained and unconstrained input capacity limits are at $E_b/N_0 = 6.65$dB and $E_b/N_0 = 6.1$dB respectively.

4.5 Conclusions

In this chapter we have studied the optimization and performance of SCLDGM codes over Multiple-Input Multiple-Output (MIMO) channels, characterized by the use of several antennas for transmission and reception. We started by establishing the channel model and studied its capacity and spatial diversity characteristics. We showed that the capacity of a MIMO channel grows proportionally with the minimum between the number of transmit and receive antennas, which allows us to achive tranmission rates much higher than with Single-Input Single-Output (SISO) systems.

We analyzed different detection algorithms for the MIMO channel. Since optimum detection complexity is prohibitive when the number of transmit antennas and bits per modulation symbol is large, suboptimum techniques must be employed to calculate the LLR of the coded bits. We have studied two different approaches: the Soft Interference Cancellation-Minimum Mean Square Error (SIC-MMSE) detector, an iterative detector that performs an interference cancellation using the extrinsic information passed by the decoder followed by an MMSE filtering to transform the MIMO to a SISO problem; and three versions of the List Sphere Detector (LSD), that reduce the number of possible transmitted vectors by considering only the most relevant. Specifically, we have studied the Maximum Likelihood LSD (ML LSD), that costructs the list of vectors by choosing those with a higher likelihood; the Maximum A

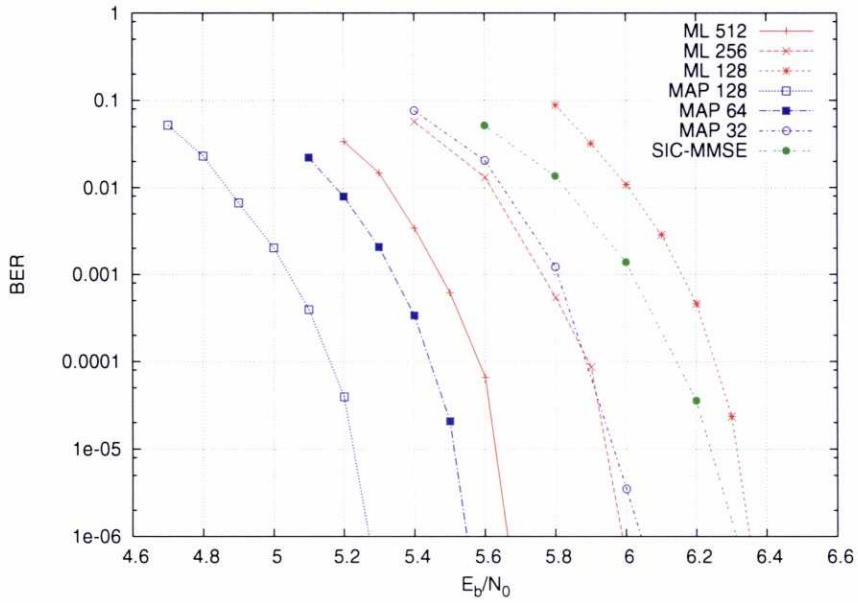


Figure 4.14: Performance of rate $1/2$ regular SCLDGM codes optimized for a 4×4 16-QAM MIMO channel with different types of detectors and number of candidates. The constrained and unconstrained input capacity limits are at $E_b/N_0 = 4.1$dB and $E_b/N_0 = 3.7$dB respectively.

Posteriori LSD (MAP LSD), that chooses those with a higher *a posteriori* probability (i.e., taking into account the bit probabilities from the decoder); and Extrinsic LSD (Ext LSD), that finds the symbol vectors with a higher extrinsic probability (considering the bit probabilities from the decoder except that of the bit for which the detector is calculating the LLR).

To perform the optimization of the SCLDGM codes for the optimum and different suboptimum detectors we have employed EXIT charts. We have showed that the MAP LSD cannot be analyzed using EXIT charts due to a problem of positive feedback, so we have found codes for the Ext LSD instead and used them with the MAP LSD to reduce the complexity, assuming that their performances would be similar. Indeed, we have shown that the combination of SCLDGM codes and MAP LSD is able to approximate the MIMO capacity limits with an affordable complexity, clearly surpassing the performance of the ML LSD and SIC-MMSE-based systems.

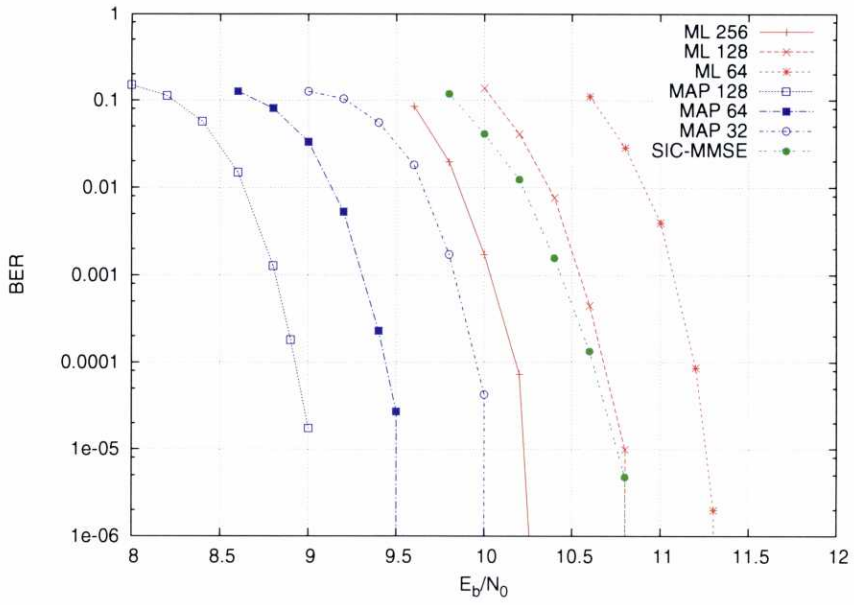


Figure 4.15: Performance of regular SCLDGM codes for 4×4 and 64-QAM. Unconstrained-input Capacity Limit at 6.3dB.

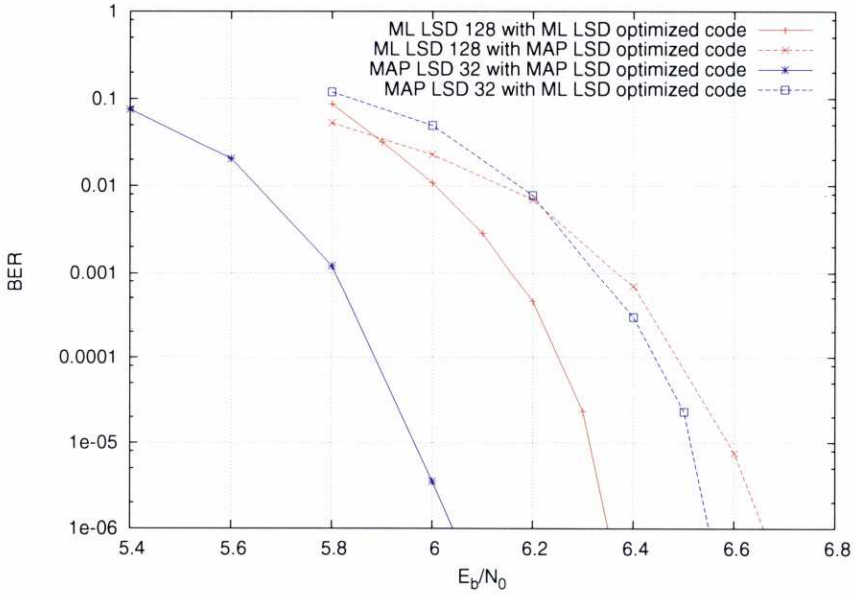


Figure 4.16: Performance degradation due to mismatch between the SCLDGM code and the detector. SCLDGM codes are rate $1/2$ and optimized for a 4×4 16-QAM MIMO channel.

Chapter 5

Concatenation with STBC codes

In the previous chapter we focused on MIMO signaling schemes in which the coded bits are mapped to constellation symbols and multiplexed to the different transmit antennas. Although this scheme has the benefits of a simple design and the ability to approach capacity, there are cases in which the use of a more complex method for the distribution of the symbols to the transmit antennas is advantageous. In this chapter we will consider the concatenation of SCLDGM codes with Space-Time Block Codes (STBC) [59, 60] and compare it with the previous BICM scheme. In particular, we will consider three different STBCs, each of them designed to achive a different goal: allow an easy decoupling of the transmitted streams in the receiver; allow the use of a suboptimum MIMO detector when the number of transmit antenas is higher than that of receive antennas; and increase the overall channel capacity.

5.1 Space-Time Block Codes

Figure 5.1 shows the concatenated scheme we will study in this chapter. As in the previous chapter, the source bits are encoded by an outer code and mapped to a complex constellation. However, instead of simply being multiplexed to the different transmit antennas, the modulated symbols are encoded by an inner Space-Time Block Code (STBC). A STBC is a block code designed specifically to exploit the advantages of MIMO systems by taking into account both

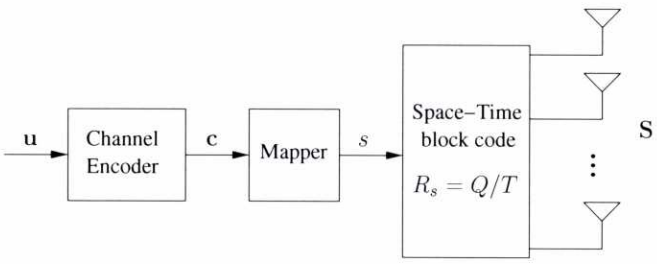


Figure 5.1: Block diagram of a STBC MIMO transmitter

the temporal and the spatial dimensions. Although STBC codes do not provide much coding gain [60], they provide other advantages that can be useful in some MIMO configurations. There exists other kind of Space-Time Codes, named Space Time Trellis Codes (STTC) [57], that provide some coding gain, but they are much more complex to decode. Moreover, since we will be using a capacity approaching code in concatenation with the Space-Time Block Code, the coding gain will already be provided by the outer channel code.

STBCs are defined by a coding matrix

$$\mathbf{S} = \begin{bmatrix} s_{11} & s_{12} & \cdots & s_{1T} \\ s_{21} & s_{22} & \cdots & s_{2T} \\ \vdots & \vdots & \ddots & \vdots \\ s_{n_T 1} & s_{n_T 2} & \cdots & s_{n_T T} \end{bmatrix} \tag{5.1}$$

whose entries s_{ij} represent the complex-valued symbols transmitted from antenna i at time instant j. The symbols are sent using T time intervals.

When an STBC is concatenated after an outer channel code, the sequence of mapped symbols $[s_1, \ldots, s_L]$ is partitioned into blocks of Q symbols. Each block is then encoded into the $n_T \times T$ symbol matrix $\mathbf{S}[k]$, which is transmitted expending T channel uses, resulting in a spatial rate $R_s = Q/T$. From this point of view, and as explained in the previous chapter, the serial-to-parallel converter can be stated as a STBC with $Q = n_T$ and $T = 1$, with a coding matrix

$$\mathbf{S} = \begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ s_{n_T} \end{bmatrix} \tag{5.2}$$

After transmission through the MIMO channel, assuming that the channel remains constant for T time intervals, the matrix of received observations $\mathbf{X}[k]$, is

$$\mathbf{X}[k] = \mathbf{H}[k]\mathbf{S}[k] + \mathbf{N}[k], \quad k = 1, 2, ..., L/R_s, \tag{5.3}$$

where $\mathbf{H}[k]$ is the $n_R \times n_T$ MIMO channel matrix and $\mathbf{N}[k] = [\mathbf{n}_1\ \mathbf{n}_2\ \cdots\ \mathbf{n}_T]$, where each column $\mathbf{n}_i$ contains independent AWGN samples with variance N_0. We assume a spatially uncorrelated Rayleigh fading MIMO channel where the elements in $\mathbf{H}[k]$ are distributed as $\mathcal{CN}(0, 1)$. Under the ergodic assumption, the channel matrix changes each time a new matrix of symbols is transmitted, whereas under the quasi-static assumption, it remains constant during the transmission of a whole codeword. We assume in both cases that the channel changes in an independent fashion from one realization to the next. To calculate the E_b/N_0 at reception we must take into account the rate of the inner STBC. If we assume that the average transmitted energy of the coded symbols is $\mathbb{E}\{|s_{ij}|^2\} = E_s$, we have

$$\frac{E_b}{N_0} = \frac{n_T n_R}{R_s R_c M_c} \frac{E_s}{N_0} \tag{5.4}$$

The mapping performed by an STBC greatly affects the features of the MIMO system: it may change its associated capacity, the attained diversity, and the complexity of the detection process, as well as the applicable detection methods. We will consider three relevant STBCs: the Alamouti scheme [59], which is an Orthogonal STBC for $n_T = 2$ transmitting antennas; the Golden code [70], a 2×2 non-orthogonal STBC that provides a capacity increase; and Linear Dispersion (LD) codes [71], which transform the observation model when $n_T > n_R$ to enable the use of suboptimum methods (e.g., LSD and SIC-MMSE) under the constraint of incurring in minimum capacity loss.

5.1.1 Orthogonal Space-Time Block Codes

Orthogonal Space-Time Block Codes (OSTBC) [72, 59, 60] are codes designed to attain two objectives: obtain full diversity gain without CSI at the transmitter and allow an optimum detection with linear complexity in the number of transmit antennas. The first objective is accomplished by ensuring that the matrix $\mathbf{D} = \mathbf{S} - \mathbf{S}'$, resulting from the differece between two coded matrices, has full rank for every pair of coded matrices [60]. The second can be accomplished by making the rows of the coding matrix orthogonal, so the detector can easily decouple the different transmit streams.

The OSTBC for the case of two transmit antennas is the Alamouti code [59]. Given two input complex symbols, s_1 and s_2, the transmitted symbol matrix for the Alamouti code is

$$\mathbf{S} = \begin{bmatrix} s_1 & -s_2^* \\ s_2 & s_1^* \end{bmatrix} \tag{5.5}$$

According to the channel model defined by Eq. (5.3), if we have only a single receive antenna, the received symbols are

$$\begin{bmatrix} x_1 & x_2 \end{bmatrix} = \begin{bmatrix} h_1 & h_2 \end{bmatrix} \begin{bmatrix} s_1 & -s_2^* \\ s_2 & s_1^* \end{bmatrix} + \begin{bmatrix} n_1 & n_2 \end{bmatrix} \tag{5.6}$$

This can be rewritten as

$$\begin{bmatrix} x_1 \\ x_2^* \end{bmatrix} = \underbrace{\begin{bmatrix} h_1 & h_2 \\ h_2^* & -h_1^* \end{bmatrix}}_{\mathcal{H}} \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} + \begin{bmatrix} n_1 \\ n_2^* \end{bmatrix} \tag{5.7}$$

where the equivalent channel matrix $\mathcal{H}$ is a scaled unitary matrix, i.e.,

$$\mathcal{H}\mathcal{H}^H = \mathcal{H}^H\mathcal{H} = (|h_1|^2 + |h_2|^2)\mathbf{I}_2 = \|\mathbf{H}\|_F^2\mathbf{I}_2 \tag{5.8}$$

where $\|\cdot\|_F$ represents the Frobenius norm of a matrix. This property allows us to decouple the transmitted streams by simply multipliying the hermitian of the equivalent matrix by the received vector. Thus, if we denote $\mathbf{x} \triangleq [x_1, x_2^*]^T$, $\mathbf{s} \triangleq [s_1, s_2]^T$ and $\mathbf{n} \triangleq [n_1, n_2^*]^T$ we can rewrite (5.7) as

$$\mathbf{x} = \mathcal{H}\mathbf{s} + \mathbf{n} \tag{5.9}$$

and we can obtain the vector of scaled estimations $\hat{\mathbf{s}} = [\hat{s}_1, \hat{s}_2]^T$ as

$$\hat{\mathbf{s}} = \mathcal{H}^H \mathbf{x} = \mathcal{H}^H (\mathcal{H}\mathbf{s} + \mathbf{n}) = \|\mathbf{H}\|_F^2 \mathbf{I}_2 \mathbf{s} + \mathcal{H}^H \mathbf{n} \tag{5.10}$$

For the case of two receive antennas, the matrix of received observations is

$$\begin{bmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} \begin{bmatrix} s_1 & -s_2^* \\ s_2 & s_1^* \end{bmatrix} + \begin{bmatrix} n_{11} & n_{12} \\ n_{21} & n_{22} \end{bmatrix} \tag{5.11}$$

which can be rewritten as

$$\begin{bmatrix} x_{11} \\ x_{12}^* \\ x_{21} \\ x_{22}^* \end{bmatrix} = \underbrace{\begin{bmatrix} h_{11} & h_{12} \\ h_{12}^* & -h_{11}^* \\ h_{21} & h_{22} \\ h_{22}^* & -h_{21}^* \end{bmatrix}}_{\mathcal{H}} \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} + \begin{bmatrix} n_{11} \\ n_{12}^* \\ n_{21} \\ n_{22}^* \end{bmatrix} \tag{5.12}$$

and, as in Equation (5.10), we can decouple the transmit streams by multiplying the received vector by the hermitian of the equivalent matrix $\mathcal{H}$.

The covariance matrix of the resulting noise $\mathcal{H}^H \mathbf{n}$ is

$$\mathbb{E}\{\mathcal{H}^H \mathbf{n}\mathbf{n}^H \mathcal{H}\} = \mathcal{H}^H \mathbb{E}\{\mathbf{n}\mathbf{n}^H\}\mathcal{H} = \mathcal{H}^H N_0 \mathbf{I}_2 \mathcal{H} = \|\mathbf{H}\|_F^2 N_0 \mathbf{I}_2 \tag{5.13}$$

so, for both one or two receive antennas, optimum computation of the channel LLRs in each stream $\hat{s}$ can be realized as

$$L_{\mathrm{ch},k} = \log \frac{P(\hat{s}|c_k = 1)}{P(\hat{s}|c_k = 0)} = \log \frac{\displaystyle\sum_{s \in S_k^1} \exp\left(-\frac{\|\hat{s} - \|\mathbf{H}\|_F^2 s\|^2}{\|\mathbf{H}\|_F^2 N_0} + \sum_{i=1}^{M_c} v_i \frac{L_i}{2} \right)}{\displaystyle\sum_{s \in S_k^0} \exp\left(-\frac{\|\hat{s} - \|\mathbf{H}\|_F^2 s\|^2}{\|\mathbf{H}\|_F^2 N_0} + \sum_{i=1}^{M_c} v_i \frac{L_i}{2} \right)} - L_k \tag{5.14}$$

where $v_i = 2c_i - 1$ and S_k^1 and S_k^0 represent the set of all transmitted symbols s where bit $c_k = 1$ and $c_k = 0$, respectively.

We can easily calculate the unconstrained capacity of the equivalent channels $\mathcal{H}$ using Eq. (4.6) and taking into account that we are using two time intervals (and thus we have to divide the resulting capacity by two). For the case of 2×1 we have

$$\begin{aligned} C &= \frac{1}{2}\mathbb{E}\left\{ \log_2 \det\left(I_2 + \frac{E_s}{N_0} \mathcal{H}\mathcal{H}^* \right) \right\} \\ &= \frac{1}{2}\mathbb{E}\left\{ \log_2 \det\left(I_2 + \frac{E_s}{N_0} (|h_1|^2 + |h_2|^2) I_2 \right) \right\} \\ &= \frac{1}{2}\mathbb{E}\left\{ \log_2 \left(1 + \frac{E_s}{N_0} (|h_1|^2 + |h_2|^2) \right)^2 \right\} \\ &= \mathbb{E}\left\{ \log_2 \left(1 + \frac{E_s}{N_0} (|h_1|^2 + |h_2|^2) \right) \right\} \end{aligned}$$

which is the capacity of a 2×1 MIMO channel. This means that by using the Alamouti code, we are not sacrificing unconstrained capacity with respect to a scheme in which we use a simple serial-to-parallel converter as a space-time code. However, for the 2×2 case

$$\begin{aligned} C &= \frac{1}{2}\mathbb{E}\left\{\log_2 \det\left(I_2 + \frac{E_s}{N_0}\mathcal{H}\mathcal{H}^*\right)\right\} \\ &= \frac{1}{2}\mathbb{E}\left\{\log_2 \det\left(I_2 + \frac{E_s}{N_0}(|h_{11}|^2 + |h_{12}|^2 + |h_{21}|^2 + |h_{22}|^2)I_2\right)\right\} \\ &= \frac{1}{2}\mathbb{E}\left\{\log_2\left(1 + \frac{E_s}{N_0}(|h_{11}|^2 + |h_{12}|^2 + |h_{21}|^2 + |h_{22}|^2)\right)^2\right\} \\ &= \mathbb{E}\left\{\log_2\left(1 + \frac{E_s}{N_0}(|h_{11}|^2 + |h_{12}|^2 + |h_{21}|^2 + |h_{22}|^2)\right)\right\} \end{aligned}$$

which can be shown to be lower or equal than the capacity of a 2×2 MIMO channel [71].

Similar OSTBCs can be defined for more than two transmit antennas. However, unlike the Alamouti code, they cannot reach a rate equal to 1 [73, 60]. Also, except for the case of the 2×1 Alamouti code, all OSTBCs incur in an unconstrained capacity loss with respect to the BICM scheme with the same number of transmit and receive antennas [74, 71].

5.1.2 Linear Dispersion Codes

When the number of transmitting antennas is higher than that of the receiving antennas, employing suboptimum methods (i.e., LSD or SIC-MMSE) is more difficult, because the underlying system of equations is underdetermined. Linear Dispersion (LD) codes [71] are linear STBCs that transform the observation model to avoid undetermination, but without sacrificing capacity.

A LD code matrix is defined as

$$\mathbf{S} = \sum_{q=1}^{Q}(s_q C_q + s_q^* D_q) \tag{5.15}$$

where $\{C_q, D_q, q = 1 \ldots Q\}$ are the complex matrices that determine the code. Equivalently

$$\mathbf{S} = \sum_{q=1}^{Q}(\alpha_q A_q + j\beta_q B_q) \tag{5.16}$$

where $s_q = \alpha_q + j\beta_q$, $A_q = C_q + D_q$ and $B_q = C_q - D_q$.

By defining the matrix code $\mathbf{S}$ as a function of the A_q and B_q matrices, a stacked real-valued equivalent observation model can then be easily formulated. First, we calculate the received values as

$$\mathbf{X}_R + j\mathbf{X}_I = \sum_{q=1}^{Q}(\mathbf{H}_R + j\mathbf{H}_I)[\alpha_q(A_{R,q} + jA_{I,q}) + j\beta_q(B_{R,q} + jB_{I,q})] + \mathbf{N}_R + j\mathbf{N}_I \tag{5.17}$$

where the subindexes R and I denote the real and imaginary parts, respectively. By separating the real and imaginary parts of the received values, we can write

$$\begin{aligned}\mathbf{X}_R &= \sum_{q=1}^{Q}[(\mathbf{H}_R A_{R,q} - \mathbf{H}_I A_{I,q})\alpha_q + (-\mathbf{H}_R B_{I,q} - \mathbf{H}_I B_{R,q})\beta_q] + \mathbf{N}_R \\ \mathbf{X}_I &= \sum_{q=1}^{Q}[(\mathbf{H}_R A_{I,q} + \mathbf{H}_I A_{R,q})\alpha_q + (\mathbf{H}_R B_{R,q} - \mathbf{H}_I B_{I,q})\beta_q] + \mathbf{N}_I\end{aligned}$$

If we denote the transposed row n of the matrices $\mathbf{X}_R$, $\mathbf{X}_I$, $\mathbf{H}$, $\mathbf{N}_R$, $\mathbf{N}_I$ as $\mathbf{x}_{R,n}$, $\mathbf{x}_{I,n}$, $\mathbf{h}_n$, $\mathbf{n}_{R,n}$, $\mathbf{n}_{I,n}$, and

$$\mathcal{A}_q = \begin{bmatrix} A_{R,q} & A_{I,q} \\ -A_{I,q} & A_{R,q} \end{bmatrix}^T \quad \mathcal{B}_q = \begin{bmatrix} -B_{I,q} & B_{R,q} \\ -B_{R,q} & -B_{I,q} \end{bmatrix}^T \quad \bar{\mathbf{h}}_n = \begin{bmatrix} \mathbf{h}_{R,n} \\ \mathbf{h}_{I,n} \end{bmatrix} \tag{5.18}$$

we have the equivalent model

$$\begin{bmatrix} \mathbf{x}_{R,1} \\ \mathbf{x}_{I,1} \\ \vdots \\ \mathbf{x}_{R,n_R} \\ \mathbf{x}_{I,n_R} \end{bmatrix} = \mathcal{H} \begin{bmatrix} \alpha_1 \\ \beta_1 \\ \vdots \\ \alpha_Q \\ \beta_Q \end{bmatrix} + \begin{bmatrix} \mathbf{n}_{R,1} \\ \mathbf{n}_{I,1} \\ \vdots \\ \mathbf{n}_{R,n_R} \\ \mathbf{n}_{I,n_R} \end{bmatrix} \tag{5.19}$$

where

$$\mathcal{H} = \begin{bmatrix} \mathcal{A}_1\bar{\mathbf{h}}_1 & \mathcal{B}_1\bar{\mathbf{h}}_1 & \cdots & \mathcal{A}_Q\bar{\mathbf{h}}_1 & \mathcal{B}_Q\bar{\mathbf{h}}_1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathcal{A}_1\bar{\mathbf{h}}_{n_R} & \mathcal{B}_1\bar{\mathbf{h}}_{n_R} & \cdots & \mathcal{A}_Q\bar{\mathbf{h}}_{n_R} & \mathcal{B}_Q\bar{\mathbf{h}}_{n_R} \end{bmatrix} \tag{5.20}$$

is the $2n_RT \times 2Q$ equivalent channel matrix.

Such an equivalent observation model is not underdetermined provided that $n_RT > Q$ or, equivalently, $n_R > R_s$. Optimum LLR computation for LD-coded MIMO systems can be carried out in a completely analogous way to Eq. (4.17) or (5.14), depending on whether the STBC is orthogonal or not, just by considering their resulting equivalent channel model.

As in the case of the Alamouti code, we can calculate the capacity of the resulting equivalent channel model as

$$C = \frac{1}{2T}\mathbb{E}\left\{\log\det\left(I_{2Nn_R} + \frac{E_s}{2N_0}\mathcal{H}\mathcal{H}^*\right)\right\} \tag{5.21}$$

where the $1/2T$ factor is due to the use of T time intervals and a real-valued equivalent channel model.

Linear Dispersion codes are designed by finding the set of matrices A_q and B_q that maximize the capacity of the equivalent channel, i.e.

$$\arg\max_{A_q,B_q,q=1\ldots Q} \frac{1}{2T}\mathbb{E}\left\{\log\det\left(I_{2Nn_R} + \frac{E_s}{2N_0}\mathcal{H}\mathcal{H}^*\right)\right\} \tag{5.22}$$

for the SNR of interest, subject to one of the following constraints

$$1. \quad \sum_{q=1}^{Q} \left(\mathrm{tr}(A_q^* A_q) + \mathrm{tr}(B_q^* B_q)\right) = 2Tn_T$$

$$2. \quad \mathrm{tr}(A_q^* A_q) = \mathrm{tr}(B_q^* B_q) = \frac{Tn_T}{Q}, \quad q = 1 \ldots Q$$

$$3. \quad A_q^* A_q = B_q^* B_q = \frac{T}{Q}\mathbf{I}_{n_T}, \quad q = 1 \ldots Q$$

where $\mathrm{tr}(\cdot)$ denotes the trace of a matrix. Each constraint is stronger than the previous one, so the attainable capacity decreases with them. However, the last constraint usually provides the best coding or diversity gains at the cost of a very small decrease in unconstrained capacity [71].

5.1.3 Golden Code

The Golden code is a STBC designed for a 2×2 MIMO channel [70]. The matrix corresponding to the Golden code is

$$\mathbf{S} = \begin{bmatrix} s_1 + \theta s_2 & s_3 + \theta s_4 \\ j(s_3 + \bar{\theta} s_4) & s_1 + \bar{\theta} s_2 \end{bmatrix} \tag{5.23}$$

where $\theta = (1 + \sqrt{5})/2$ is the golden number and $\bar{\theta} = 1 - \theta$

The main advantage of the Golden code is that it provides a capacity increase at a reasonable complexity cost. This capacity improvement comes from the fact that the resulting constellation at the output of the Golden encoder resembles a Gaussian distribution better than the input constellation (see Fig. 5.2). This effect is usually termed as *shaping* or constellation expansion [75]. Although the size of the resulting constellation is larger, the Golden code does not introduce any redundancy because it expends two channel uses for transmitting four input symbols. Regarding capacity, the use of the Golden code with 2^{M_c}-QAM (e.g., 4-QAM) at its input is in between using BICM with a rate R_c outer code plus 2^{M_c}-QAM (e.g., 4-QAM) and using BICM with a rate $R_c/2$ outer code plus 2^{2M_c}-QAM (e.g., 16-QAM).

Optimum decoding of the Golden code can be carried out by considering it as a particular case of a LD code, since the Golden code matrix can be expressed as in Eq. (5.16) and, thus, the same equivalent observation model as the one in Eq. (5.19) applies.

5.2 BICM vs concatenation with STBCs

Let us first emphasize that STBCs provide little (or no) coding gain. Thus, if channel capacity (either ergodic or outage for the quasi-static case) is to be approached, any STBC has to be used in concatenation with a temporal outer code. Although Trellis-Coded Modulation (TCM) [51] schemes may be considered, when considering Rayleigh fading the usual option is to employ a BICM scheme as an outer code for the STBC. Thus the question is to determine whether or

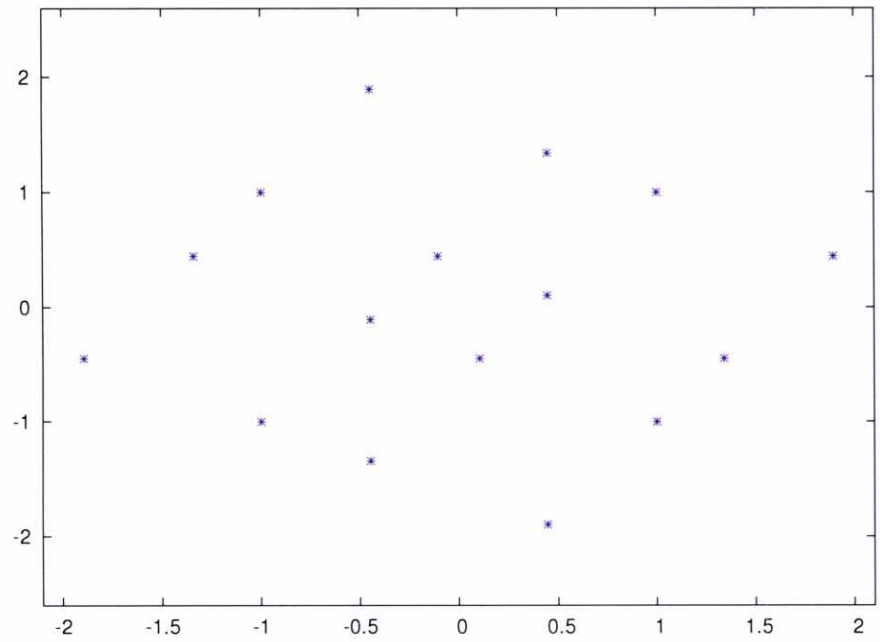


Figure 5.2: Constellation at the output of the 2×2 Golden code when using a 4-QAM constellation at its input.

not concatenating an STBC after a BICM scheme provides any advantage over just spatially multiplexing the output of the BICM scheme. In addition, the spatial rate, R_s, consumed by an STBC establishes a ceiling on the information rate.

We will first focus on the case of ergodic channels and start analyzing the use of STBCs for the case of two transmit antennas. When used as an inner code, the Alamouti code is very attractive because it allows to spatially decouple the ML detection, notably simplifying the overall decoding procedure. The price to be paid is the spatial rate consumed by the Alamouti code, $R_s = 1$, which forces the use of a higher order modulation to compensate for the rate loss with respect to a pure BICM scheme. Moreover, the signal structure that it imposes penalizes the capacity of the equivalent MIMO system. As shown before, the capacity of a $2 \times n_R$ MIMO system with Alamouti coding is less or equal than that of the unconstrained MIMO channel. The equality holds only for the case $n_R = 1$. From this, it is clear that for 2×1 MIMO systems the best choice is Alamouti coding, since it transforms the MIMO system into two parallel, independent SISO systems while maintaining the same overall capacity and ensuring maximum transmit diversity.

Let us now focus on 2×2 MIMO systems. In this case, using the Alamouti code enables low complexity detection, but the system performance is severely affected by the capacity loss when $n_R > 1$. Instead, we can use BICM, since complexity of optimal ML detection for $n_T = 2$ is affordable. Moreover, suboptimal detection schemes such as LSD or SIC-MMSE

may be directly used if needed, since the observation model is fully determined ($n_T = n_R = 2$). An appealing alternative is the use of a Golden-coded system, since it provides a capacity improvement at a little increase in complexity.

BICM without concatenation with STBCs is the best option in the general case $n_T \times n_R$ with $n_T, n_R > 2$. However, the complexity of optimum MAP detection in BICM (Eq. 4.17), exponential in both n_T and M_c, constitutes an important limitation for a high number of transmitting antennas. At a first glance, OSTBCs may seem an appealing alternative in these cases, because their orthogonality allows to spatially decouple the detection of the transmitted streams. However, they incur in a great capacity loss (except for the 2×1 case) and force the use of much higher order constellations to achieve the same information rate as BICM without STBCs. Thus, a more effective solution is to employ suboptimum detection methods (such as LSD or SIC-MMSE) in BICM. However, these methods pose the additional constraint of requiring the observation to be fully determined ($n_T \leq n_R$). This is an important limitation because $n_T > n_R$ occurs frequently in practice (for example, in the downlink of cellular wireless systems).

For the case of $n_T > n_R$, the use of LD codes allows the application of the aforementioned suboptimum detection methods without much capacity penalty. Thus, concatenation of BICM with LD codes constitute a good choice when $n_T > n_R$, $n_T > 2$, and optimum detection is not feasible. Notice also that, similarly to the Golden code, constellation expansion takes place at the output of an LD encoder, but it is controlled because symbols are produced according only to specific sequences.

For quasi-static MIMO channels, STBCs constitute a simple way of extracting full diversity without knowing the MIMO channel at the transmitter, but they do not provide coding gain. If outage capacity is to be approached, STBCs have to be used in concatenation with an outer channel code. Since capacity-achieving codes (e.g. Turbo and LDPC) produce random-like codewords, in practice they usually fulfill the rank criterion [57] that guarantees full diversity. Thus, in these cases, the concatenation with STBCs is not useful, except for the cases when they provide some additional advantage, such as those commented for the 2×1 Alamouti, 2×2 Golden and LD codes.

5.3 SCLDGM Code Optimization for MIMO-STBC systems

Similarly to previous schemes, we can predict the convergence of the concatenated coding scheme by tracking the mutual information between the densities of the different types of messages involved in the SPA and the *a priori* bit density.

For Binary-Input AWGN (BIAWGN) channels, the detector EXIT function does not depend on the *a priori* information, I_A, because each bit is transmitted in an independent channel use. Furthermore, as we have seen in Chapter 3, for Single-Input Single-Output (SISO) channels with perfect CSI at reception and the usual constellations (i.e., PSK and QAM), Gray mapping results in an overall effect of modulation, channel and demodulation with an associated EXIT

Code	$n_T \times n_R$	Modul.	STBC	R_c	p	d_u^{f1}	d_u^{f2}	d_{p1}^{f2}	Thresh	Gap
#1	2×1	4-QAM	None	$1/2$	0.0150	3	5	38	3.87dB	0.57dB
#2	2×1	16-QAM	None	$1/4$	0.0200	3	8	15	3.57dB	0.77dB
#3	$2 \times n_R$	Any	Alamouti	$1/4$	0.0400	3	9	24	-	-
#4	$2 \times n_R$	Any	Alamouti	$1/2$	0.0200	3	6	6	-	-
#5	2×2	16-QAM	None	$1/4$	0.0275	3	9	12	1.90dB	0.60dB
#6	2×2	4-QAM	None	$1/2$	0.0300	3	5	32	2.02dB	0.46dB
#7	2×2	4-QAM	Golden	$1/2$	0.0250	3	5	38	1.87dB	0.45dB
#8	3×1	4-QAM	None	$1/6$	0.0350	3	10	24	1.20dB	0.80dB
#9	3×1	4-QAM	LD	$1/2$	0.0200	3	6	48	1.38dB	0.48dB

Table 5.1: Optimized SCLDGM codes for MIMO channels. "Thresh" stands for the convergence threshold and "Gap" is the gap to the constrained-input capacity limit.

function equivalent to a BIAWGN channel, that is, only dependent on E_b/N_0 (i.e., it is almost an horizontal line for each E_b/N_0 value). Thus, optimum codes for BIAWGN are also optimum for SISO channels and for any modulation format, provided Gray-mapping is used. As a corollary of this result, optimum codes for OSTBC-coded systems (in particular, Alamouti-coded) with Gray mapping are the same as those optimized for BIAWGN channels.

For the case of LD codes (for which, in terms of detection, the Golden code is a particular instance) the detector produces channel LLRs according to an extended channel model. In this case, the detector EXIT function is different from the one corresponding to BICM over the same channel model. Consequently, optimum codes for LD-coded systems differ from those obtained for BICM systems. Fig. 5.3 plots the EXIT characteristic for 2×2 and 3×1 MIMO systems for either BICM or LD coding. They correspond to values of the SNR at each receive antenna (equal to n_T/N_0 assuming each antenna radiates unit energy symbols) close to the convergence threshold of the best code found in each case. Let us first recall that the area property of an EXIT function states that the area below the curve is approximately equal to the BIAWGN channel capacity assumed for modelling the input and output messages of the MIMO detector. For a 2×2 system operating at SNR$= 2.0$dB, it is apparent that the EXIT function corresponding to a Golden-coded system has a larger area below it than that corresponding to a BICM system, which is coherent with the capacity increase associated to the Golden code. Note also that the slope of the two functions is different, which leads to different optimum codes (cf. Table 5.1). For the 3×1 case, the EXIT function corresponding to BICM is located far below the one corresponding to the LD-coded system, because the outer code rates are different ($R_c = 1/6$ and $1/2$, respectively) for the same overall information rate ($1/6$). Besides, their slopes are different. Both reasons justify that the optimum codes for these two schemes are very different from each other.

Table 5.1 summarizes the best regular SCLDGM codes obtained through EXIT evolution for different MIMO systems. The table also shows the convergence thresholds predicted by this analysis, as well as the Constrained-input Capacity Limit (CCL) corresponding to each MIMO

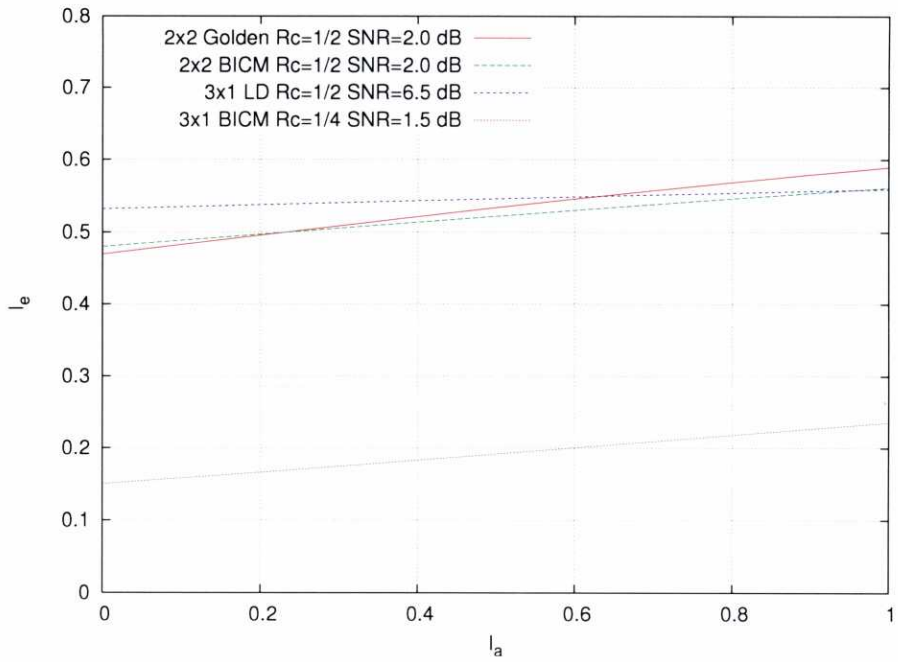


Figure 5.3: EXIT characteristics corresponding to the considered MIMO schemes for a 4-QAM constellation.

system. We have considered the antenna configurations where concatenation with STBCs is more beneficial: 2×1 for the Alamouti code, 2×2 for the Golden code, and 3×1 for the LD code. In particular, the latter case presents the characteristics of being a clearly asymmetric antenna configuration ($n_T > n_R$) as well as having a complexity low enough to appreciate the losses in i) capacity with respect to BICM and ii) performance of suboptimum methods with respect to optimum detection in the LD-coded system. We have chosen these antenna configurations as our testbench to see the gains provided by STBC-concatenated systems. For each antenna configuration, we optimize the outer code to maintain the overall information rate fixed, so we can make a fair comparison between BICM and the concatenated schemes.

As we will see in Section 5.4, SCLDGM codes optimized for Single-Input Single-Output (SISO) channels also exhibit good performance when used in concatenation with the Alamouti code in $2 \times n_R$ MIMO channels. This is not a surprising result since the Alamouti code actually converts a $2 \times n_R$ MIMO channel into two parallel and independent SISO channels. Recall that the detector EXIT function of Gray-mapped SISO systems with all the standard constellations is an horizontal line (i.e. it is constant for any value of I_a), resulting in the same optimum code for any constellation. This explains why we do not include a threshold value for the Alamouti case, since this threshold is different depending on the employed modulation and the channel model.

Although the code design procedure assumes an ergodic channel and an infinite block

length, we will show in Section 5.4 that the obtained codes perform adequately when used in quasi-static channels and with finite block lengths. Indeed, the behaviour of the obtained codes over quasi-static channels is completely analogous to that over ergodic channels.

5.4 Simulation results

Computer simulations were carried out to illustrate the actual performance of the optimized regular SCLDGM codes with data blocks of finite length. We assumed in all the schemes that the transmitter has no CSI and the receiver has perfect CSI. We have considered ergodic and quasi-static channels. For ergodic channels, the observed thresholds are slightly worse than those predicted, since EXIT function analysis assumes infinite-length data blocks. Note that the lower the code rate, R_c, is, i) the higher the gap to the CCL for the best code found, and ii) the higher the gap between the theoretical threshold predicted by EXIT analysis and the threshold observed in simulations. Similar conclusions hold for quasi-static channels when comparing the actual performance with respect to the outage probability limit.

The performance of several systems with a spectral efficiency of two bits per channel use over a 2×1 ergodic MIMO channel is shown in Fig. 5.4. The block length is $K = 10000$ systematic bits (we will also use this block length throughout all the simulations in ergodic channels). The best performance is obtained when using BICM with 16-QAM and code rate $R_c = 1/4$ (code #2). For a BER$= 10^{-4}$ (which we will use as the target BER in the sequel) the required E_b/N_0 is 4.0dB, which is 1.2dB away from the constrained-input capacity limit. Fig. 5.4 also plots the performance obtained when using an Alamouti scheme with 16-QAM and code rate $R_c = 1/2$, with the code optimized for the SISO channel (code #3). The SCLDGM rate was raised up to $R_c = 1/2$ in order to maintain the spectral efficiency equal to two bits per channel use. Not surprisingly, the performance of these two schemes is very similar, because the capacity limit of a 2×1 system is the same irrespectively of whether Alamouti is used or not. Regarding receiver complexity, however, it is apparent that decoding in the concatenated scheme is simpler.

The spectral efficiency of two bits per channel use can also be obtained using BICM with 4-QAM and code rate $R_c = 1/2$. Code #1 has been specifically optimized for this particular situation and its performance is also shown in Fig. 5.4. Lowering the number of bits per symbol in the modulator, M_c, is interesting because it yields to a considerable reduction of the detector complexity. At the target BER, the required E_b/N_0 is 4.55dB (1.25dB away from the CCL for 4-QAM). Thus, this case maintains the same gap to the CCL as the 16-QAM case, but it exhibits a 0.5dB performance degradation due to the capacity loss resulting from changing the modulation format (see Table 5.1). Figure 5.4 also illustrates the importance of designing SCLDGM codes for each specific MIMO configuration in BICM. Indeed, observe the serious degradation in performance when the SISO-optimized codes #3 and #4 are used instead of the MIMO-optimized ones. In these cases, the required E_b/N_0 is 5.25 and 8.0dB, respectively. The loss in performance with respect to the MIMO-optimized codes is 0.7 and

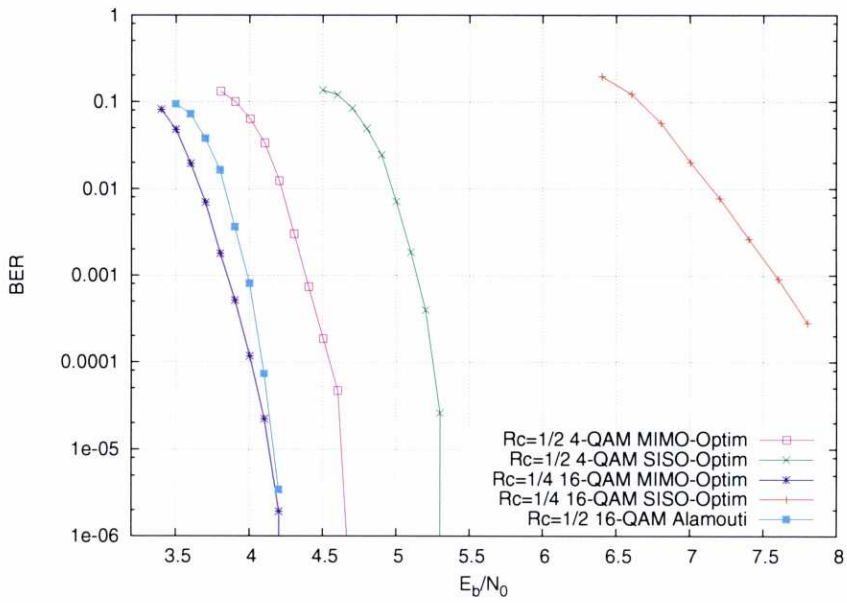


Figure 5.4: Performance of i) SCLDGM + Golden code, and ii) SCLDGM + BICM for a 2×1 MIMO system with a spectral efficiency of 2 information bits per channel use. Ergodic Rayleigh fading.

4.0dB, respectively. From these results, we can conclude that code optimization is more critical when the constellation size increases.

The performance of i) SCLDGM + Golden code, and ii) BICM for a 2×2 MIMO system with a spectral efficiency of two bits per channel use is shown in Fig. 5.5 for an ergodic channel model. Although the $R_c = 1/4$ 16-QAM BICM system has the highest capacity (its corresponding CCL is at $E_b/N_0 = 1.30$dB), the best performance is attained by the Golden-coded system (for which its corresponding CCL is at $E_b/N_0 = 1.43$dB). This is explained because the best $R_c = 1/4$ code found for the 16-QAM BICM system has a threshold at $E_b/N_0 = 1.90$dB (theoretically) and at $E_b/N_0 = 2.20$dB (in practice), which is worse than that of the best $R_c = 1/2$ code found for the Golden-coded system (at $E_b/N_0 = 1.87$dB theoretically and at $E_b/N_0 = 2.15$dB in practice). The $R_c = 1/2$ 4-QAM BICM system shows worse performance, requiring $E_b/N_0 = 2.30$dB at the target BER, which is consistent with its lower system capacity (CCL at $E_b/N_0 = 1.56$dB). We have also included the performance obtained for these systems when the code is the optimum for a SISO model. The gaps in performance with the respect to the MIMO-optimized codes are significant, especially for 16-QAM (0.3 for 4-QAM and 1.10dB for 16-QAM). Finally, observe the poor performance of the Alamouti-coded system (1.25dB worse than the MIMO-optimized code) even when using its optimum code, which is a consequence of its system capacity loss.

Fig. 5.6 shows the results for i) SCLDGM + LD code and ii) BICM for a 3×1 ergodic MIMO

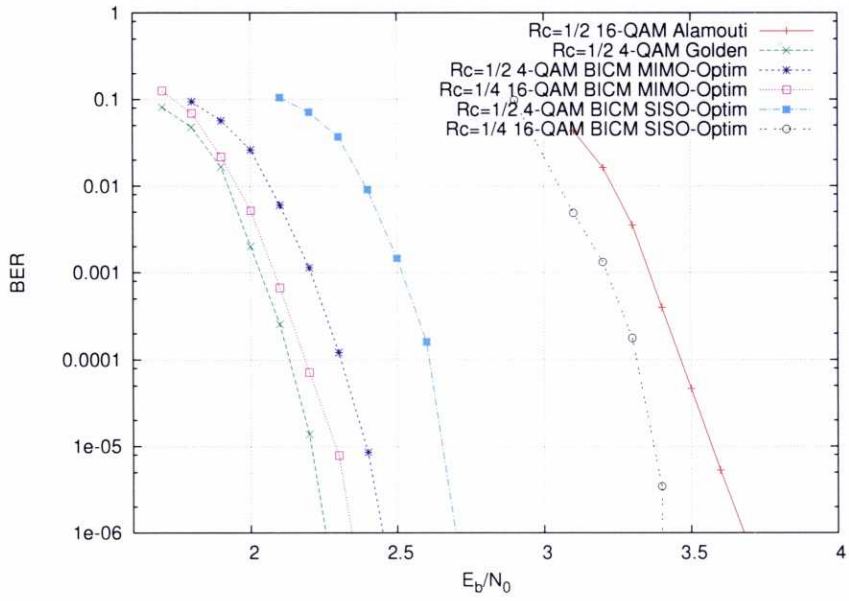


Figure 5.5: Performance of i) SCLDGM + Golden code, and ii) SCLDGM + BICM for a 2×2 MIMO system with a spectral efficiency of 2 information bits per channel use. Ergodic Rayleigh fading.

channel when using 4-QAM with a spectral efficiency of one information bit per channel use. The best performance is exhibited by the BICM system employing an $R_c = 1/6$ SCLDGM code (which requires an E_b/N_0 of 1.25dB for the target BER). Using the LD code given by Eq. (36) in [71] enables the application of suboptimum schemes such as LSD or SIC-MMSE at the cost of sacrificing capacity and, thus, performance. The degradation in actual performance (under optimum detection) with respect to BICM is not severe (0.45dB at the target BER) and is of the same order as the loss in capacity (the CCL for BICM is at $E_b/N_0 = 0.40$dB whereas that corresponding to using the LD code is at $E_b/N_0 = 0.90$dB). This comes from the fact that the lower the rate is, the larger is the gap between the code threshold and the capacity limit. Since not using LD coding requires a lower code rate ($R_c = 1/6$ instead of $R_c = 1/2$), the gap with respect to the CCL for the optimum code is larger (0.80dB instead of 0.48dB). We can also think of an $R = 1/6$ overall coding rate being realized through one code (for the BICM system) with a threshold 0.80dB away from the CCL, or by the concatenation of two codes (for the BICM + LD system) with a threshold $1.38 - 0.40 = 0.98$dB away from the CCL, motivated by two losses: one due to gap between the threshold of the SCLDGM code and the CCL corresponding to using LD coding (0.48dB) and other due to the capacity loss due to employing LD coding. Noting that the CCL corresponding to using LD coding is at $1.38 - 0.48 = 0.90$dB, then the capacity loss of using LD coding is $(0.90 - 0.40 = 0.50$dB). In addition, when the LD code is used, the gap of either MAP LSD or SIC-MMSE with respect to optimum detection is fairly

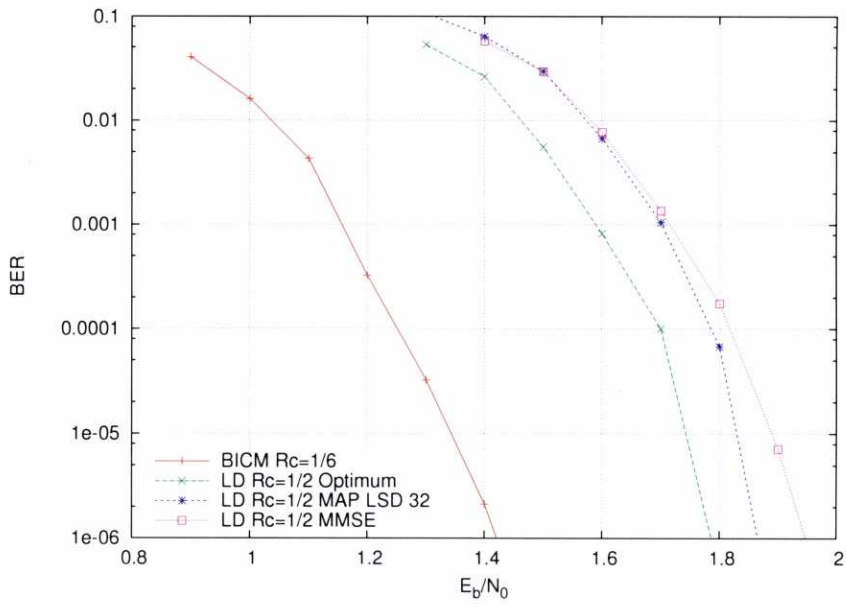


Figure 5.6: Performance of i) SCLDGM + $R_s = 1/3$ STBC (LD code), and ii) SCLDGM + BICM for a 4-QAM 3×1 MIMO system with a spectral efficiency of 1 information bit per channel use. Ergodic Rayleigh fading.

small (around 0.1dB). This corroborates the convenience of employing LD coding as a means of enabling suboptimum detection methods when receiver complexity is a constraint.

We have also studied the performance of these schemes when transmitting over quasi-static channels. Figure 5.7 shows the performance of a SCLDGM BICM scheme, a SCLDGM + Golden code scheme and a SCLDGM + Alamouti scheme over a 2×2 MIMO channel. The channel block length is $B = 500$ symbol vectors. As occurs in the ergodic channel, the best performance is achieved by the SCLDGM + Golden code, although the difference with the BICM schemes is minimal. The figure also shows that all the schemes achieve maximum diversity, since the slope of all the curves is the same as the slope of the outage capacity. This demonstrates that the BICM scheme is also suitable for quasi-static channels, without having to resort to schemes that explicitly maximize the spatial diversity.

Similar results can be observed in Figure 5.8 for a quasi-static 3×1 MIMO channel. Again, the BICM scheme is able to achieve the same diversity as the LD scheme, and both have a very similar performance.

5.5 Conclusions

We have studied MIMO transmission schemes using either BICM by itself or in concatenation with a Space-Time Block Code (STBC). Our study has been supported by the ability of regular

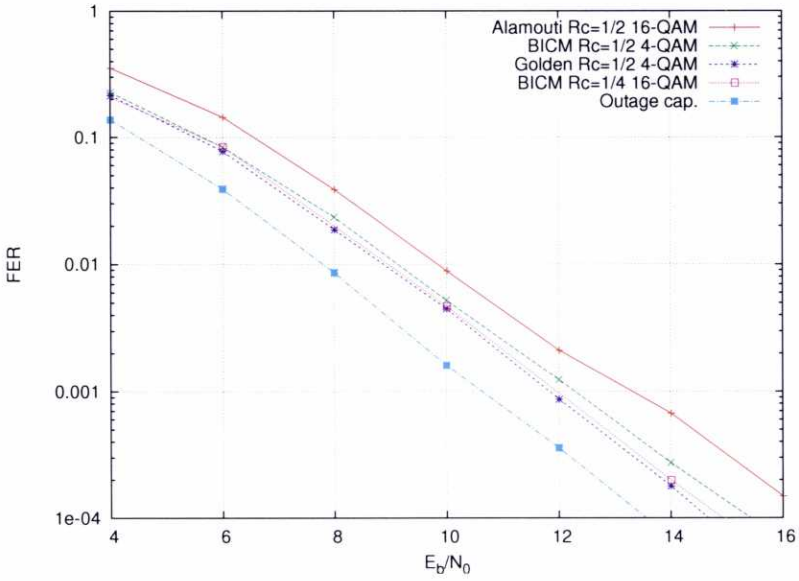


Figure 5.7: Performance of i) SCLDGM + BICM, ii) SCLDGM + Golden code and iii) SCLDGM + Alamouti for a 2×2 MIMO system with a spectral efficiency of 2 information bits per channel use. Quasi-static Rayleigh fading.

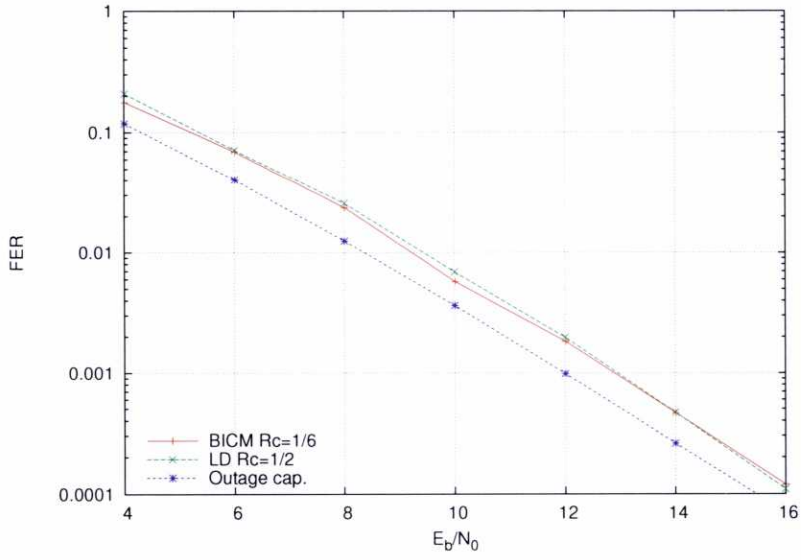


Figure 5.8: Performance of i) SCLDGM + $R_s = 1/3$ STBC (LD code), and ii) SCLDGM + BICM for a 4-QAM 3×1 MIMO system with a spectral efficiency of 1 information bit per channel use. Quasi-static Rayleigh fading.

SCLDGM codes specifically designed for each system to perform close to the theoretical limits. For both ergodic and quasi-static channels, we conclude that BICM without concatenation with STBCs is in general the best option, because it retains maximum system capacity, while practical Turbo-like codes (e.g. SCLDGM codes) can be designed to achieve the provided capacity. Although the complexity of optimum MAP detection might be an issue in BICM systems with a high number of transmitting antennas, near-optimum performance methods with low complexity (e.g. LSD and SIC-MMSE) can be employed.

Exceptions to this rule are i) 2×1 MIMO systems, for which concatenation with the Alamouti code incurs in no capacity loss with respect to BICM without concatenation, presenting the additional advantage of spatially decoupling the optimum detection process, ii) 2×2 MIMO systems, because concatenating with the Golden code provides a capacity increase at little additional detection complexity, and iii) asymmetrical MIMO systems with more transmitting than receiving antennas ($n_T > n_R$) for which optimum MAP detection of BICM without concatenation is computationally unfeasible, while concatenation with Linear Dispersion (LD) codes enables the utilization of suboptimum detection methods (e.g. LSD and SIC-MMSE) at reception that otherwise would be difficult to apply due to the undetermination of the observation model.

Chapter 6

Channel estimation

Up to now, we have assumed that the receiver has perfect Channel State Information (CSI) and uses the known channel matrix to detect the transmitted symbols. However, in practical situations, the channel is unknown at the receiver, and has to be estimated prior to the detection stage. In this chapter we will study the performance of SCLDGM codes in combination with the space time coding schemes studied in the previous chapter when the channel is estimated at the receiver. We will employ two different methods, Least Squares (LS) [76] and Maximum Likelihood with Expectation-Maximization (ML-EM) [77], to find the estimate of the channel matrix, and compare the results with the case of perfect CSI.

6.1 MIMO systems with pilot symbol assisted modulation

Let us assume a quasi-static MIMO system given by the model

$$\mathbf{x}[k] = \mathbf{H}\mathbf{s}[k] + \mathbf{n}[k] \quad k = 1, 2, \ldots, B \tag{6.1}$$

where the channel matrix $\mathbf{H}$ is unknown at both the transmitter and the receiver and B is the number of symbol vectors transmitted through that channel matrix. Our objective is to find a channel estimation $\hat{\mathbf{H}}$ in the receiver using the observations $\mathbf{x}[k]$ and some knowledge about the trasmitted symbols $\mathbf{s}[k]$. Channel estimation methods can be divided in two groups: those that use a sequence of $P < B$ *pilot* symbols that are known at the receiver, which is known as Pilot Symbol Assisted Modulation (PSAM); and those that rely only in the statistical properties of the unknown transmitted signals [78, 79]. Although the latter methods, termed as *blind* estimation methods, have the advantage of not needing to spend time transmitting signals that do not carry any information, they have a higher complexity or exhibit worse performance. Also, as we will see in the following sections, the methods that we will consider need only a small number of pilot symbols to perform close to the case of perfect CSI, so we will concentrate in PSAM estimation methods. In particular, we will study the performance of SCLDGM codes when using two different channel estimation methods: Least Squares and Maximum Likelihood.

6.1.1 Least Squares

Least Squares (LS) [76] is an estimation method that finds the parameter of a model that minimizes the sum of squared residuals between a set of noisy observations and the expected values of the model. For example, when applied to the channel model given by (6.1), and assuming that the first P transmitted symbols are known at the receiver, LS calculates the channel estimation $\hat{\mathbf{H}}$ as

$$\hat{\mathbf{H}} = \arg\min_{\mathbf{H}} \sum_{k=1}^{P} \|\mathbf{x}[k] - \mathbf{H}\mathbf{s}[k]\|^2 \tag{6.2}$$

To solve this minimization problem, let us first rewrite the sum of residuals as

$$\begin{aligned} J(\mathbf{H}) &= \sum_{k=1}^{P} \|\mathbf{x}[k] - \mathbf{H}\mathbf{s}[k]\|^2 = \sum_{k=1}^{P} \mathrm{tr}((\mathbf{x}[k] - \mathbf{H}\mathbf{s}[k])(\mathbf{x}[k] - \mathbf{H}\mathbf{s}[k])^H) \\ &= \sum_{k=1}^{P} \mathrm{tr}(\mathbf{x}[k]\mathbf{x}[k]^H) - \sum_{k=1}^{P} \mathrm{tr}(\mathbf{x}[k]\mathbf{s}[k]^H\mathbf{H}^H) \\ &- \sum_{k=1}^{P} \mathrm{tr}(\mathbf{H}\mathbf{s}[k]\mathbf{x}[k]^H) + \sum_{k=1}^{P} \mathrm{tr}(\mathbf{H}\mathbf{s}[k]\mathbf{s}[k]^H\mathbf{H}^H) \end{aligned} \tag{6.3}$$

Taking the derivative of $J(\mathbf{H})$ with respect to $\mathbf{H}^H$ and setting it equal to zero results in

$$\frac{\partial J(\mathbf{H})}{\partial \mathbf{H}^H} = -\sum_{k=1}^{P} \left(\mathbf{x}[k]\mathbf{s}[k]^H\right)^T + \sum_{k=1}^{P} \left(\mathbf{H}\mathbf{s}[k]\mathbf{s}[k]^H\right)^T = \mathbf{0} \tag{6.4}$$

If we denote $\mathbf{R}_{xs}[k] = \mathbf{x}[k]\mathbf{s}^H[k]$ and $\mathbf{R}_{ss}[k] = \mathbf{s}[k]\mathbf{s}^H[k]$ we can rewrite (6.4) as

$$-\sum_{k=1}^{P} \mathbf{R}_{xs}[k] + \mathbf{H}\sum_{k=1}^{P} \mathbf{R}_{ss}[k] = \mathbf{0} \tag{6.5}$$

which yields

$$\hat{\mathbf{H}} = \left(\sum_{k=1}^{P} \mathbf{R}_{xs}[k]\right) \times \left(\sum_{k=1}^{P} \mathbf{R}_{ss}[k]\right)^{-1} \tag{6.6}$$

The receiver uses the known pilot symbols and the observations to calculate the channel estimation $\hat{\mathbf{H}}$. This estimated channel matrix is employed by the detector to calculate the LLR values that will be passed to the decoder in the same way as the known channel matrix was employed in previous chapters.

6.1.2 Maximum Likelihood with Expectation-Maximization

An alternative to the LS solution to the estimation problem is the Maximum Likelihood (ML) estimate of the channel, given by

$$\hat{\mathbf{H}} = \arg\max_{\mathbf{H}} p(\tilde{\mathbf{x}}|\mathbf{H}), \tag{6.7}$$

where $p(\tilde{\mathbf{x}}|\mathbf{H})$ is the pdf of all the received values $\tilde{\mathbf{x}} = [\mathbf{x}[0]\ \mathbf{x}[1]\ \cdots\ \mathbf{x}[B]]$ given the channel matrix $\mathbf{H}$. This problem does not have a closed form solution, but the Expectation-Maximization (EM) [77] algorithm is able to find the exact solution in an iterative way. The EM algorithm can be employed when the estimated parameter depends on some unobserved (or *hidden*) data that would make the problem easier to resolve if it was known. In the channel estimation case, the hidden data is formed by the sent symbols $\tilde{\mathbf{s}} = [\mathbf{s}[0]\ \mathbf{s}[1]\ \cdots\ \mathbf{s}[B]]$. Together with the *observable* data $\tilde{\mathbf{x}}$ they form what is known as the *complete* data. The EM algorithm uses an initial estimate of the parameter to calculate an improved estimation by maximizing the expectation, with respect to the hidden data given the previous estimation and the observable values, of the log-likelihood of the complete data, i.e.

$$\begin{aligned}\hat{\mathbf{H}}_{i+1} &= \arg\max_{\mathbf{H}} \mathbb{E}_{\tilde{\mathbf{s}}|\tilde{\mathbf{x}};\hat{\mathbf{H}}_i}\left\{\log\left(p(\tilde{\mathbf{x}},\tilde{\mathbf{s}}|\mathbf{H})\right)\right\} \\ &= \arg\max_{\mathbf{H}} \sum_{\tilde{\mathbf{s}}} p(\tilde{\mathbf{s}}|\tilde{\mathbf{x}};\hat{\mathbf{H}}_i)\log\left(p(\tilde{\mathbf{x}},\tilde{\mathbf{s}}|\mathbf{H})\right),\end{aligned} \tag{6.8}$$

Elaborating (6.8) we have

$$\begin{aligned}\hat{\mathbf{H}}_{i+1} &= \arg\max_{\mathbf{H}} \mathbb{E}_{\tilde{\mathbf{s}}|\tilde{\mathbf{x}};\hat{\mathbf{H}}_i}\left\{\log(p(\tilde{\mathbf{x}}|\tilde{\mathbf{s}},\mathbf{H})p(\tilde{\mathbf{s}})\right\} = \arg\max_{\mathbf{H}} \mathbb{E}_{\tilde{\mathbf{s}}|\tilde{\mathbf{x}};\hat{\mathbf{H}}_i}\left\{\log(p(\tilde{\mathbf{x}}|\tilde{\mathbf{s}},\mathbf{H}))\right\} \\ &= \arg\min_{\mathbf{H}} \mathbb{E}_{\tilde{\mathbf{s}}|\tilde{\mathbf{x}};\hat{\mathbf{H}}_i}\left\{\sum_{k=1}^{B}\|\mathbf{x}[k]-\mathbf{H}\mathbf{s}[k]\|^2\right\} \\ &= \arg\min_{\mathbf{H}} \sum_{k=1}^{B} \mathbb{E}_{\mathbf{s}[k]|\tilde{\mathbf{x}},\hat{\mathbf{H}}_i}\left\{\|\mathbf{x}[k]-\mathbf{H}\mathbf{s}[k]\|^2\right\} \\ &= \arg\max_{\mathbf{H}} \sum_{k=1}^{B} \mathbb{E}_{\mathbf{s}[k]|\mathbf{x}[k];\hat{\mathbf{H}}_i}\left\{\|\mathbf{x}[k]-\mathbf{H}\mathbf{s}[k]\|^2\right\}\end{aligned} \tag{6.9}$$

Since the expectation operator is linear, we can perform the same derivation as in (6.6) to obtain a similar solution

$$\begin{aligned}\hat{\mathbf{H}}_{i+1} &= \left(\sum_{k=1}^{B} \mathbb{E}_{\mathbf{s}[k]|\mathbf{x}[k];\hat{\mathbf{H}}_i}\left\{\mathbf{R}_{xs}[k]\right\}\right) \times \left(\sum_{k=1}^{B} \mathbb{E}_{\mathbf{s}[k]|\mathbf{x}[k];\hat{\mathbf{H}}_i}\left\{\mathbf{R}_{ss}[k]\right\}\right)^{-1} \\ &= \left(\sum_{k=1}^{B}\sum_{\mathbf{s}} p(\mathbf{s}[k]|\mathbf{x}[k];\hat{\mathbf{H}}_i)\mathbf{R}_{xs}[k]\right) \times \left(\sum_{k=1}^{B}\sum_{\mathbf{s}} p(\mathbf{s}[k]|\mathbf{x}[k];\hat{\mathbf{H}}_i)\mathbf{R}_{ss}[k]\right)^{-1}\end{aligned} \tag{6.10}$$

The a posteriori probabilities $p(\mathbf{s}[k]|\mathbf{x}[k];\hat{\mathbf{H}}_i)$ are calculated by the detector using the a priori information passed by the decoder as

$$\begin{aligned}p(\mathbf{s}[k]|\mathbf{x}[k];\hat{\mathbf{H}}_i) &\propto p(\mathbf{x}[k]|\mathbf{s}[k];\hat{\mathbf{H}}_i)p(\mathbf{s}[\mathbf{k}]) \\ &\propto \exp\left(-\frac{1}{N_0}\|\mathbf{x}[\mathbf{k}]-\hat{\mathbf{H}}_i\mathbf{s}[\mathbf{k}]\|^2\right)\prod_{i=1}^{n_T M_c}\left[1+\exp(-v_i L_i)\right]^{-1}\end{aligned} \tag{6.11}$$

This equation shows the need of an initial estimate of the channel. To obtain it, we must send some pilot symbols known by the receiver. Thus, if we consider that the first P transmitted symbols correspond to the pilot sequence, the channel estimator simplifies to

$$\begin{aligned}\hat{\mathbf{H}}_{i+1} &= \left(\sum_{k=1}^{P}\mathbf{R}_{xs}[k] + \sum_{k=P+1}^{B}\mathbb{E}_{\mathbf{s}[k]|\mathbf{x}[k],\hat{\mathbf{H}}_i}\{\mathbf{R}_{xs}[k]\}\right) \\ &\times \left(\sum_{k=1}^{P}\mathbf{R}_{ss}[k] + \sum_{k=P+1}^{B}\mathbb{E}_{\mathbf{s}[k]|\mathbf{x}[k],\hat{\mathbf{H}}_i}\{\mathbf{R}_{s}[k]\}\right)^{-1}\end{aligned} \tag{6.12}$$

and the initial estimate of the channel can be calculated as

$$\hat{\mathbf{H}}_0 = \left(\sum_{k=1}^{P}\mathbf{R}_{xs}[k]\right) \times \left(\sum_{k=1}^{P}\mathbf{R}_{ss}[k]\right)^{-1} \tag{6.13}$$

which corresponds to the LS solution to the estimation problem. It is interesting to note that (6.12) can be viewed as a generalized LS channel estimator where for the unknown symbols the correlations $\mathbf{R}_{xs}[k]$ and $\mathbf{R}_{ss}[k]$ are obtained by averaging with respect to the *a posteriori* pdf of those symbols. The lower the variance associated to this pdf, the more information we have about the unknown symbols and, thus, the much more they look like pilot symbols regarding estimation. In this way, the length of the pilot sequence, P, can be reduced with respect to pure LS channel estimation.

The main problem when estimating the channel via the EM algorithm is similar to that in the optimum APP MIMO detector: complexity of the expectation calculation in (6.12) is exponential in the number of antennas and constellation size. The use of a list sphere detector, however, can alleviate this problem: by combining the sphere detection and the EM algorithms, the average in (6.12) is not calculated over all possible transmitted vectors, but only over the candidate list obtained by the LSD, i.e.,

$$\mathbb{E}_{\mathbf{s}[k]|\mathbf{x}[k],\hat{\mathbf{H}}_i}\{\mathbf{R}_s[k]\} = \sum_{\mathbf{s}[k]\in\mathbf{S}} p(\mathbf{s}[k]|\mathbf{x}[k],\hat{\mathbf{H}}_i)\mathbf{R}_s[k] \approx \sum_{\mathbf{s}[k]\in\mathcal{L}} p(\mathbf{s}[k]|\mathbf{x}[k],\hat{\mathbf{H}}_i)\mathbf{R}_s[k] \tag{6.14}$$

where $\mathbf{S}$ is the set of all possible transmitted symbol vectors, with $|\mathbf{S}| = 2^{n_T M_c}$, whereas $\mathcal{L} = \{\mathbf{s}_1, \mathbf{s}_2, ..., \mathbf{s}_{N_{\text{cand}}}\}$ denotes the candidate list obtained by the LSD. The same approximation can be made for computing $\mathbb{E}_{\mathbf{s}[k]|\mathbf{x}[k],\hat{\mathbf{H}}_i}\mathbf{R}_{xs}[k]$. Moreover, the MAP LSD already computes the APPs $p(\mathbf{s}[k]|\mathbf{x}[k])$ needed for the expectation.

EM algorithm with LD codes

When transmitting using an LD code, the a posteriori probabilities $p(\mathbf{s}[k]|\mathbf{x}[k];\hat{\mathbf{H}}_i)$ needed by the EM algorithm cannot be calculated by the detector, since the sent symbols are space-time encoded and the detector uses the equivalent channel model. On the other hand, we can easily calculate the a posteriori probabilities $p(\mathbf{S}[k]|\mathbf{X}[k];\hat{\mathbf{H}}_i)$. Thus, if we denote

$\mathbf{s}'[k] = [\alpha_1, \beta_1, \ldots, \alpha_Q, \beta_Q]^T$ the uncoded version of $\mathbf{S}[k]$, $\mathbf{x}'[k] = [\mathbf{x}_{R,1}^T \mathbf{x}_{I,1}^T \cdots \mathbf{x}_{R,n_R}^T \mathbf{x}_{I,n_R}^T]^T$ the unfolded version of $\mathbf{X}[k]$, and $\mathcal{H}_i$ the equivalent channel matrix constructed using the rows of $\hat{\mathbf{H}}_i$ and Eq. (5.20), we can calculate

$$p(\mathbf{s}'[k]|\mathbf{x}'[k]; \mathcal{H}_i) \propto \exp\left(-\frac{1}{N_0}\|\mathbf{x}'[\mathbf{k}] - \mathcal{H}_i \mathbf{s}'[\mathbf{k}]\|^2\right) \prod_{i=1}^{QM_c} [1 + \exp(-v_i L_i)]^{-1} \tag{6.15}$$

Since there is a one-to-one equivalency between $\mathbf{S}[k]$ and $\mathbf{s}'[k]$, and between $\mathbf{X}[k]$ and $\mathbf{x}'[k]$, we have

$$p(\mathbf{S}[k]|\mathbf{X}[k]; \hat{\mathbf{H}}_i) = p(\mathbf{s}'[k]|\mathbf{x}'[k]; \mathcal{H}_i) \tag{6.16}$$

This way we can reformulate the EM estimation given by Eq. (6.12) as

$$\begin{aligned} \hat{\mathbf{H}}_{i+1} &= \left(\sum_{k=1}^{P} \mathbf{R}_{xs}[k] + \sum_{k=P+1}^{B/Q} \mathbb{E}_{\mathbf{S}[k]|\mathbf{X}[k],\hat{\mathbf{H}}_i}\{\mathbf{R}_{XS}[k]\}\right) \\ &\times \left(\sum_{k=1}^{P} \mathbf{R}_{ss}[k] + \sum_{k=P+1}^{B/Q} \mathbb{E}_{\mathbf{S}[k]|\mathbf{X}[k],\hat{\mathbf{H}}_i}\{\mathbf{R}_{SS}[k]\}\right)^{-1} \end{aligned} \tag{6.17}$$

being $\mathbf{R}_{XS}[k] = \mathbf{X}[k]\mathbf{S}^H[k]$ and $\mathbf{R}_{SS}[k] = \mathbf{S}[k]\mathbf{S}^H[k]$.

6.2 Code optimization

Considering the pilot symbols as additional redundancy, we define the information rate loss due to pilot insertion as $R_p = P/B$. The overall information rate is thus given by $R = R_c R_s R_p M_c$. Accordingly, the E_b/N_0 at reception is calculated as

$$\frac{E_b}{N_0} = \frac{n_T n_R}{R_c R_s R_c M_c} \frac{E_s}{N_0} \tag{6.18}$$

Traditionally, the code rate R_c is assumed as fixed when assessing the performance of a given channel estimation method and, thus, the number of transmitted pilots affects the overall information rate. This is so even when the rate loss due to pilot insertion, R_p, is taken into account in the calculation of the E_b/N_0, as we do in Eq. (6.18). The performance comparison is biased since it benefits the utilization of more pilot symbols, because they allow to obtain better channel estimations at the cost of only an additional price in E_b/N_0, and not in information rate. This way of comparison is not fair and it is completely analogous to compare different coding schemes without taking into account the code rate: the scheme with lowest rate always has the chance of obtaining the best performance, since its corresponding capacity limit is at the lowest E_b/N_0 value.

Contrarily, we are interested in finding the optimum number of pilot symbols for a given information rate, i.e., the optimal combination of code rate, R_c, and pilot insertion rate, R_p.

Keeping constant the information rate when varying the number of inserted pilots is only possible if the rate of the used codes can be changed with high resolution, as occurs with SCLDGM codes. To obtain a fair comparison, we optimize the SCLDGM code for each of the considered code rates.

Table 6.1 shows, for each considered MIMO scheme, the best codes corresponding to the three different numbers of inserted pilots in PSAM transmission (in this case, data frames with length $B = 500$ vectors of symbols are assumed). Notice that since in the optimization we are assuming perfect CSI, using more pilots results in a worse threshold, because the code rate is increased (so the code is less powerful). This forces us to assess in Section 6.3, via Monte Carlo simulations and for realistic conditions (i.e. finite block lengths and imperfect channel estimates), the actual performance of the different numbers of pilots and considered codes.

Code	$n_T \times n_R$	Modul.	STBC	P	R_p	R_c	p	d_u^{f1}	d_u^{f2}	d_{p1}^{f2}	Thresh	Gap
#1	2×1	4-QAM	None	0	1	1/2	0.0150	3	5	38	3.87dB	0.57dB
#2	2×1	16-QAM	None	0	1	1/4	0.0200	3	8	15	3.57dB	0.77dB
			None	10	0.98	0.25510	0.0200	3	8	20	3.60dB	0.80dB
			None	20	0.96	0.26042	0.0250	3	7	18	3.62dB	0.82dB
			None	40	0.92	0.27174	0.0225	3	7	16	3.65dB	0.85dB
#3	$2 \times n_R$	Any	Alamouti	0	1	1/4	0.0400	3	9	24	-	-
#4	$2 \times n_R$	Any	Alamouti	0	1	1/2	0.0200	3	6	6	-	-
#5	2×2	16-QAM	None	0	1	1/4	0.0275	3	9	12	1.90dB	0.60dB
#6	2×2	4-QAM	None	0	1	1/2	0.0300	3	5	32	2.02dB	0.46dB
#7				10	0.98	0.51020	0.0200	3	5	46	2.10dB	0.45dB
#8				20	0.96	0.52083	0.0200	3	5	46	2.21dB	0.47dB
#9				40	0.92	0.54348	0.0150	3	5	42	2.38dB	0.47dB
#10	2×2	4-QAM	Golden	0	1	1/2	0.0250	3	5	38	1.87dB	0.45dB
#11				10	0.98	0.51020	0.0300	3	5	32	1.95dB	0.46dB
#12				20	0.96	0.52083	0.0250	3	5	38	2.05dB	0.49dB
#13				40	0.92	0.54348	0.0200	3	5	38	2.26dB	0.54dB
#14	3×1	4-QAM	None	0	1	1/6	0.0350	3	10	24	1.20dB	0.80dB
#15				12	0.976	0.17077	0.0350	3	10	24	1.26dB	0.81dB
#16				24	0.952	0.17507	0.0300	3	10	32	1.30dB	0.80dB
#17				36	0.928	0.17960	0.0300	3	10	32	1.36dB	0.81dB
#18	3×1	4-QAM	LD	0	1	1/2	0.0200	3	6	48	1.38dB	0.48dB
#19				12	0.976	0.51229	0.0400	3	5	24	1.45dB	0.47dB
#20				24	0.952	0.52521	0.0350	3	5	26	1.53dB	0.47dB
#21				36	0.928	0.53879	0.0300	3	5	32	1.60dB	0.46dB

Table 6.1: Optimized SCLDGM codes for MIMO channels. "Thresh" stands for the convergence threshold and "Gap" is the gap to the constrained-input capacity limit. P and R_p values correspond to a block length of $B = 500$ vectors of symbols.

6.3 Simulation results

We first carried out a simulation experiment to illustrate the performance when the channel is estimated according to the procedure explained in Section 6.1.2, without adjusting the rates of the codes to the number of pilot symbols. We considered a sequence of $N = 20000$ coded bits transmitted over a 4×4 MIMO channel using 16-QAM. This corresponds to the transmission of

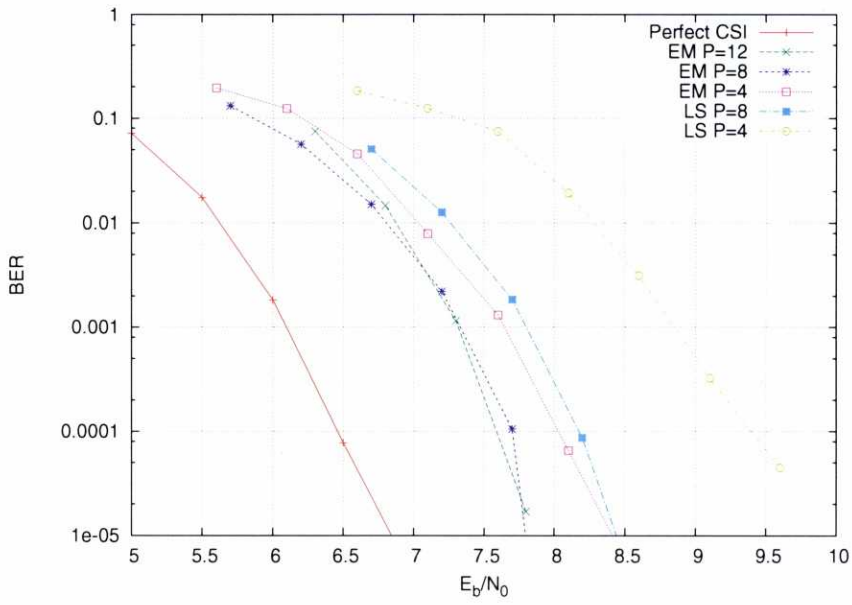


Figure 6.1: Performance of the rate $1/2$ regular SCLDGM code optimized for a 4×4 16-QAM MIMO system and a MAP LSD with 64 candidates when the channel is imperfectly estimated.

$L = 1250$ vectors of symbols. The transmitted bits are splitted into 10 blocks of $B - P = 125$ information vector symbols, and then P pilot vector symbols are appended to each block. Block fading is assumed: the channel remains constant during the transmission of each block of B symbols and randomly changes from one block to another. In the training period we transmit the 16-QAM symbols with highest energy using only one antenna at each time slot, switching the selected transmit antenna between consecutive time slots to guarantee orthogonality of the training sequence. In Fig. 6.1, the E_b/N_0 has been properly normalized to account for the energy "wasted" in pilot symbols. We have also assumed that only one iteration in the EM algorithm is performed for each detection iteration. Fig. 6.1 shows the BER curves for $P = 4$, 8 and 12. It can be seen that with a small number of pilot symbols such as 8 (6.01% of the frame size) and 12 (8.75%) and with the MAP LSD with 64 candidates, the performance degradation due to imperfect channel estimation is less than 1dB. Larger degradation is obtained if a lower number of pilot symbols are used and/or ML LSD (even with 256 candidates) is employed. Notice from Fig. 6.1 that when the channel is perfectly known, there is a diffrerence in performance with respect to the pure ergodic case of about 1dB. This is because in this computer experiment the channel is not purely ergodic but block fading. Obviously, the performance tends to the case of fast fading if the codeword length is increased and B is kept fixed.

As explained in Section 6.2, although this comparison method takes into account the energy lost in the pilot symbols, that is not the case with the information rate lost, thus benefiting the utilization of a larger number of pilot symbols. In the following we will test the codes optimized

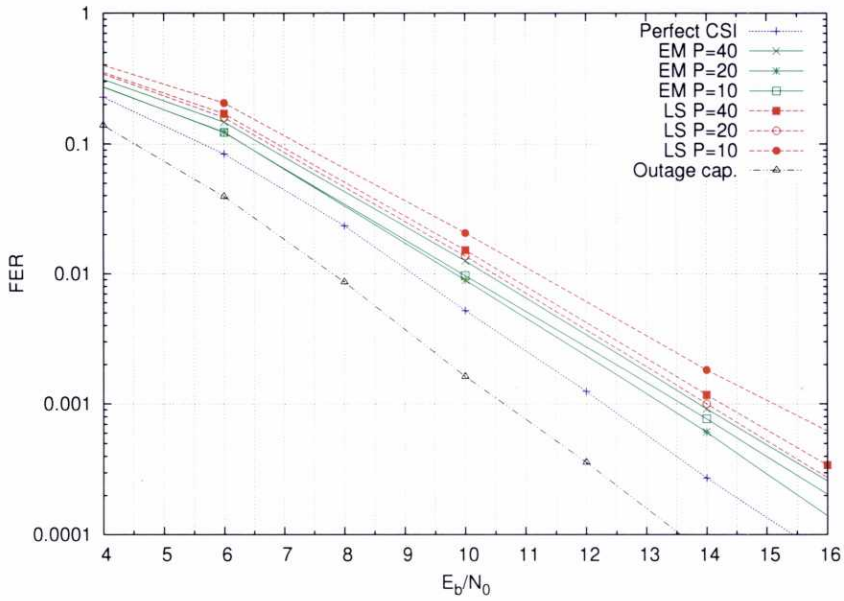


Figure 6.2: Performance of SCLDGM + BICM for a 2×2 MIMO system with a spectral efficiency of 2 information bits per channel use.

in Table 6.1, adjusting the rates to the employed pilot symbols. We will consider quasi-static Rayleigh fading with a block lenght of $B = 500$ vectors of symbols. The obtained results can be used to determine the best pilot configuration, i.e., the optimal values of R_c and R_p for a given information rate, R. Let us pick the 2×2 BICM system to illustrate how to carry out the search for the best pilot configuration for two different channel estimation methods, LS and ML-EM, at a target Frame-Error Rate (FER) of 10^{-3}. Fig. 6.2 plots the performance of a 2×2 4-QAM BICM MIMO system with rate $R = 2$ information bits per channel use. The outage probability limit is also plotted for reference purposes. When perfect CSI is assumed at reception and, thus, no pilots are inserted ($R_p = 1$), the code rate is $R_c = 1/2$. For the target FER, the required $E_b/N_0 = 12.5$dB, which is 1.9dB away from the outage probability limit.

Let us first analyze the resulting performance when classical LS channel estimation is performed at reception. The worst behaviour (at $E_b/N_0 = 15.1$dB, 2.6dB away from that of perfect CSI) is obtained when only $P = 10$ (vectors of symbols) pilots are employed. This corresponds to $R_p = 0.98$ and $R_c = 0.51020$. Increasing the number of pilots to $P = 20$ ($R_p = 0.96$ and $R_c = 0.52083$) provides a gain of 1.1dB, resulting in a required $E_b/N_0 = 14.0$dB, 1.5dB away from that of perfect CSI. Using a higher number of pilots ($P = 40$), results in some performance loss (the required $E_b/N_0 = 14.5$dB, 0.5dB worse than for $P = 20$). Therefore, the optimal number of pilots for the considered block length ($B = 500$ vectors of symbols) is around $P = 20$. Obviously, obtaining the exact optimal number of pilots would require an exhaustive search over all the values of P, or a dichotomy-

based search provided the function to be maximized fulfills some conditions. However, notice that we can obtain a significant performance gain by observing the performance curves for just three different numbers of pilots.

Note that, in contrast with the theoretical thresholds (obtained assuming perfect CSI), using more pilots does not necessarily result in worse performance, because it leads to better channel estimates. Indeed, there is a tradeoff regarding how the redundancy is distributed over pilot symbols and parity bits: increasing the number of pilots improves the quality of channel estimation, but decreases the power of the code. So, the number of pilots has to be increased just up to a point where it is enough to obtain an accurate (although not perfect) channel estimation, but still allows the code rate to be as low as possible.

When channel estimation is performed using the EM algorithm, better performance is obtained. Again, we will try three different numbers of pilot symbols and choose the one resulting in best performance. Using only $P = 10$ pilots results in a required $E_b/N_0 = 13.6$dB, 1.5dB better than that of LS channel estimation with the same number of pilots. Slight performance gain (0.3dB) is observed when $P = 20$, being the required $E_b/N_0 = 13.3$dB, only 1.0dB away from the perfect CSI case. However, increasing the number of pilots to $P = 40$ results in a significant loss in performance (the required $E_b/N_0 = 13.8$dB, 1.5dB away that of perfect CSI), even when compared to using only $P = 10$ pilots. This means that for $P = 10$ or 20, channel estimation is accurate enough, so that increasing the number of pilots does not produce a significant gain in channel estimation accuracy, while it increases the code rate and, thus, reduces its error correction capability. Also observe how the ML-EM channel estimator is more robust than the LS with respect to the number of transmitted pilots: the difference in performance among $P = 10, 20, 40$ is only 0.5dB, while for LS is 1.1dB. With these observations in mind, the optimal number of pilots for the considered blocklength ($B = 500$) is around $P = 20$.

Next, we analyze the performance results for three different antenna configurations (2×1, 2×2 and 3×1), assuming throughout all the figures the best number of pilots obtained in the same way as we have just explained. Fig. 6.3 shows the performance of i) SCLDGM + Alamouti code, and ii) BICM for a 4-QAM 2×1 MIMO system with a spectral efficiency of 2 information bits per channel use. For the target FER$= 10^{-3}$, the outage probability limit is at $E_b/N_0 = 18.2$dB. When perfect CSI is available at reception, the best performance is achieved by the Alamouti-coded scheme, with a required $E_b/N_0 = 19.2$dB, just 1.0dB away from the theoretical limit. The performance of the BICM scheme is quite similar (the required $E_b/N_0 = 19.5$dB, 0.3dB worse). When no CSI is available at reception and LS channel estimation is performed, the Alamouti-coded scheme requires a $E_b/N_0 = 20.0$dB (only 0.7dB more than for perfect CSI) when using $P = 20$ pilots. If ML-EM channel estimation with $P = 10$ pilots is used, the required $E_b/N_0 = 19.5$dB, just 0.3dB more than for perfect CSI. Notice that this performance is even better than that of the BICM scheme with perfect CSI. For the BICM scheme, if no CSI is available at the receiver, using LS channel estimation with $P = 20$ pilots requires an $E_b/N_0 = 20.8$dB, 1.3dB more than for perfect CSI. If ML-EM

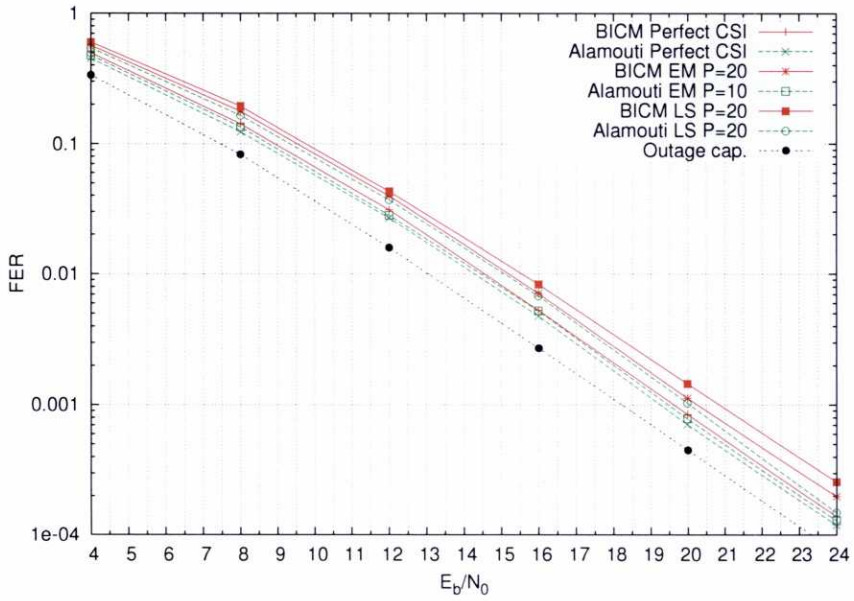


Figure 6.3: Performance of i) SCLDGM + Alamouti code, and ii) SCLDGM + BICM for a 4-QAM 2×1 MIMO system with a spectral efficiency of 2 information bits per channel use. Perfect CSI.

channel estimation is used, the required $E_b/N_0 = 20.2$dB, 0.7dB more than for perfect CSI.

Fig. 6.4 compares the performance of i) SCLDGM + Golden code, and ii) BICM for a 4-QAM 2×2 MIMO system with a spectral efficiency of 2 information bits per channel use. For the target FER$= 10^{-3}$, the outage probability limit is at $E_b/N_0 = 10.6$dB. For perfect CSI at reception, the Golden-coded system exhibits the best performance, with a required $E_b/N_0 = 11.8$dB, which is 1.2dB away from the theoretical limit. The BICM scheme requires $E_b/N_0 = 12.3$dB, 0.5dB more. The 16-QAM Alamouti-coded scheme requires $E_b/N_0 = 13.3$dB, 1dB more than the 4-QAM BICM system. This clearly shows the penalty in performance due to the capacity loss associated to Alamouti coding when $n_R > 1$. On the other hand, the better performance of the Golden-coded system also comes from its higher capacity. For unknown CSI at reception, using LS channel estimation with $P = 20$ pilots requires $E_b/N_0 = 13.0$dB for the Golden-coded scheme whereas $E_b/N_0 = 14.0$dB for the BICM scheme, which are 1.2dB and 1.7dB away from the perfect CSI case. On the other hand, using the ML-EM channel estimator with also $P = 20$ pilots requires $E_b/N_0 = 12.4$dB and 13.2dB for the Golden-coded and BICM schemes, respectively. The gaps with respect to perfect CSI are 0.6 and 0.9dB, respectively.

Finally, Fig. 6.5 plots the performance of a i) 4-QAM 3×1 LD-coded system, and ii) 4-QAM 3×1 MIMO BICM system, both with an information rate $R = 1$ information bit per channel use, and optimum detection. The outage probability limit is at $E_b/N_0 = 12.0$dB for the

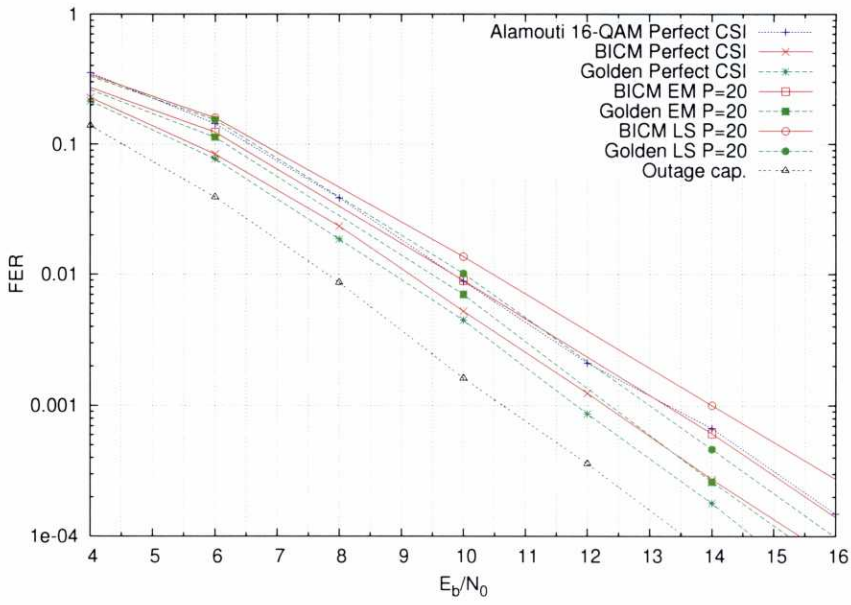


Figure 6.4: Performance of SCLDGM + BICM Vs SCLDGM + Golden code for a 4-QAM 2×2 MIMO system with an average spectral efficiency of 2 information bits per channel use.

target FER$= 10^{-3}$. Both systems exhibit nearly the same performance for the perfect CSI case. The required $E_b/N_0 = 12.8$dB, just 0.8dB away from the outage value. This shows that the capacity loss incurred by LD-coding results in very little performance penalty, while presenting the advantage of enabling the use of suboptimum detection methods. When there is no CSI at the receiver, both schemes have the same performance when LS channel estimation with $P = 36$ pilots is performed, with a required $E_b/N_0 = 14.1$dB, 1.3dB away that of perfect CSI. On the other hand, ML-EM estimation with $P = 24$ pilots requires $E_b/N_0 = 13.5$ and 13.7dB, for the LD-coded and BICM systems, respectively. This probably does not mean that the LD-coded scheme exhibits better performance regarding ML-EM estimation, but that the optimal number of pilot symbols is different for these two systems, and the chosen value $P = 20$ benefits the LD-coded scheme.

Notice also that for the BICM schemes the slope of the different performance curves is nearly the same as that of the outage probability, meaning that these schemes provide full diversity gain without imposing any particular structure on the code (and without trying to force the design to achieve this goal).

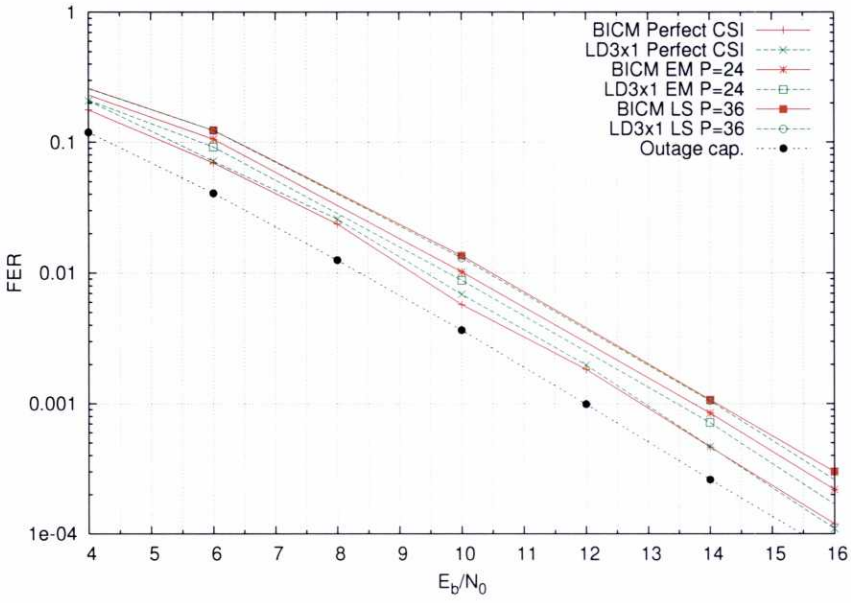


Figure 6.5: Performance of SCLDGM + BICM Vs SCLDGM + LD code for a 4-QAM 3×1 MIMO system with a spectral efficiency of 1 information bit per channel use.

6.4 Conclusions

In this chapter we have considered the more realistic case of an unknown channel at the receiver. We studied two different channel estimation methods, Least Squares (LS) and Maximum Likelihood with Expectation Maximization (ML-EM). Both are examples of Pilot-Assisted Coded Modulation (PSAM) schemes that rely on a training sequence of symbols that is known at the receiver in order to obtain an estimation of the channel. We showed that the ML-EM algorithm can be seen as a generalized LS in which the unknown symbols are estimated by using the probabilities from the decoder and, therefore, its performance and complexity are greater than those of LS. Also, ML-EM suffers from the same problem as the optimum MIMO detectors of an exponential complexity in the number of transmit antennas, so we proposed the combination of ML-EM with the LSD to reduce the complexity. The simulation results show that this combination performs very close to the case of perfect CSI at the receiver.

We have also addressed the problem of choosing the rate of the channel code and the number of pilots while keeping constant the overall information rate. SCLDGM codes enable us to specify the code rate with high resolution, something that is necessary to maintain fixed the overall information rate when varying the number of used pilots. We have a proposed a method to find the best number of pilots under this constraint, illustrating it in the case of only considering three different possible number of pilots. From this, we have been able to show that the number of pilots always remains within a quite small range of values, for both the LS and

ML-EM channel estimators and for all the considered transmission schemes.

Chapter 7

Layered LDGM (LLDGM) codes

The turbo-like codes studied in previous chapters are able to perform near the capacity limits for several types of channels models, but only for medium to high rates (i.e., above $1/4$). The design of low rate coding schemes remains an open research field, in spite of existing many potential application scenarios, such as multilevel coding for Single-Input Single-Output (SISO), Multiple-Input Multiple-Output (MIMO) and multiuser systems. Only few contributions on the design of low rate capacity approaching codes have appeared in the literature (see [80, 81, 82, 83, 84]). Most of them are based on including Hadamard codes as the constraints of a Turbo-like code. The utilization of Hadamard constraints enables to approach the Shannon limits in the low rate regime at a reasonable encoding and decoding complexities, but at the expense of making less flexible the overall coding structure.

As explained in [82], the problem of designing good low rate LDPC codes is that the decoding tunnel through the EXIT chart associated to the code remains very narrow even for E_b/N_0 values far greater than the code theoretical threshold. This theoretical threshold assumes an infinite block length, a cycle free factor graph and an infinite number of decoding iterations. However, the decoding tunnel for actual decoding in practical situations is always narrower than the theoretical one. Thus, if the theoretical tunnel opens very slowly when increasing the E_b/N_0, the actual decoding tunnel needs a significant increase in E_b/N_0 to get opened. This means that the actual performance of a given code will be far away from its predicted threshold. In addition, it can be seen that the lower the rate is, the higher the variable node degrees of optimized LDPC codes are. As a consequence, complexity is increased and it is more difficult to find good codes with a threshold near the capacity limit.

The Generalized LDPC (GLDPC) approach in [82] uses Hadamard constraints to open the decoding tunnel from the constraint node side. Differently, in this work we focus on a coding technique suitable for a wide range of rates that keeps the simplicity and flexibility of LDPC codes, because it only uses single-parity check (SPC) constraints. The decoding tunnel is opened by imposing a layered structure. Our approach is inspired by the intuitive idea that in a concatenated structure the rates of the individual codes remain high enough to provide good performance, and that the required maximum node degrees in a layered structure are lower.

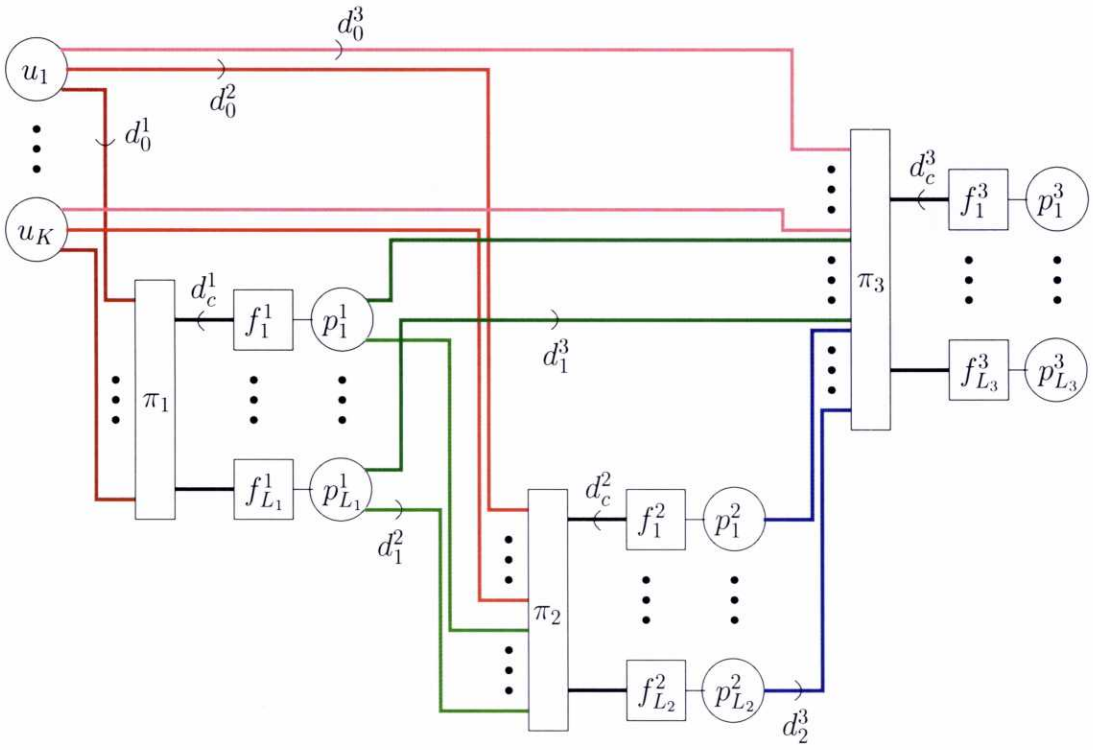


Figure 7.1: Factor graph of a systematic regular Three-layer LDGM code

The proposed structure, that we term Layered LDGM (LLDGM), consists of the serial concatenation of several systematic Low-Density Generator Matrix (LDGM) codes. LLDGM codes resemble Raptor codes [85, 86], but with the fundamental difference that LLDGM codes are finite rate. On the contrary, Raptor codes were conceived under the digital fountain paradigm [87] and are, thus, rateless codes. This difference means that the target channel models in LLDGM codes and, therefore, the optimization criteria are the same as for LDPC codes.

7.1 Layered LDGM codes

SCLDGM codes suffer a degradation in performance similar to that of general LDPC codes for rates below $1/4$. For both general LDPC and SCLDGM codes, there have been proposals aimed at low rates, all of them based on substituting single parity check (SPC) node constraints by Hadamard ones. Although the obtained results approach the capacity limits, using Hadamard codes instead of SPC codes rests flexibility and simplicity to the coding structure. We try to avoid such impairments by restricting ourselves to SPC constraints and resorting to layered structures for achieving near-capacity performance in the low-rate regime.

LLDGM codes extend the basic premise of SCLDGM codes: they are the result of serially

concatenating several systematic LDGM component codes. Fig. 7.1 shows the factor graph corresponding to a Three-layer LDGM code, where thick lines represent groups of several edges. There is a random connection between variable nodes and check nodes within each layer, which is represented as an edge interleaver, π_i, $i = 1, 2, 3$. We use the notation d_{i-1}^j, $i, j = 1, 2, 3$, $i - 1 < j$, to denote the degree of variable nodes in layer i with respect to check nodes in layer j, and d_c^j, $j = 1, 2, 3$, to denote the degree of check nodes in layer j. We will also denote as R_j, $j = 1, 2, 3$ the rate of the LDGM code associated to layer j. This factor graph representation can be easily extended to LLDGM codes with more layers. Note also that in this context, Two-layer LDGM codes are equivalent to SCLDGM codes.

Notice also that a Three-layer LDGM code can be viewed as an SCLDGM followed by another systematic regular LDGM code. Under this point of view, we may think of substituting the SCLDGM code corresponding to the first two layers by another practical LDPC code, such as an IRA code. We will term this structure as IRA-LDGM.

Our subsequent study tries to give a precise answer to the general statement that single LDGM codes are suitable for very high rates ($R > 0.9$), Two-layer LDGM codes (i.e. SCLDGM codes) for medium rates ($0.25 < R < 0.9$), and Three-layer LDGM codes for rates lower than 0.25.

7.2 Code optimization and performance results

Optimization of LLDGM codes using EXIT function analysis parallels the procedure explained in previous chapters. However, there are some differences with respect to the EXIT functions explained for the SCLDGM scheme, due to the utilization of edge interleavers between each layer. Now, all the mutual informations passed to the check nodes from all the groups of variable nodes in the previous layers are "joined" in one mutual information by averaging by the number of edges carrying each amount of information (as in the case of irregular degree profiles). For example, the information received by the second layer of checks in a LLDGM code from the two first layers of variable nodes should be calculated as

$$I_{up^1 \to f^2} = \frac{d_0^2 K I_{u \to f^2} + d_1^2 L_1 I_{p^1 \to f^2}}{d_0^2 K + d_1^2 L_1} \tag{7.1}$$

where $d_0^2 K$ and $d_1^2 L_1$ are the number of edges carrying the informations $I_{u \to f^2}$ and $I_{p^1 \to f^2}$, respectively. As a consequence, we do not make distinctions in the information passed from the check nodes to the different previous layers of variable nodes. For example, the information passed from the second layer of check nodes to both the systematic and outer parity bits would be

$$I_{f^2 \to up^1} = 1 - J\left((d_c^2 - 1) J^{-1}(1 - I_{up^1 \to f^2}) + J^{-1}(1 - I_{p^2 \to f^2})\right) \tag{7.2}$$

Also, no restriction is imposed on the number of edges that a layer of checks can receive from a previous layer of variable nodes. This restriction allowed an easy computation of the information passed from the check nodes to the previous layers of variable nodes in the codes

designed in the previous chapters when no edge interleaver was used (see Section 3.3) but the use of average information makes this restriction unnecessary.

Another difference with the previous optimization is the use of a search algorithm called Differential Evolution (DE) [88] to find the best codes. Although an exhaustive search is still feasible, the increase in the number of parameters for three-layer LDGM codes makes the search with DE much faster. DE is an optimization method for multidimensional functions that finds the global minimum of the desired function with a high probability, even if the function has several local minima. The way DE works is the following: first, it generates a set of random trial parameter vectors. Then it takes two parameter vectors and combines them, using a weighted difference between them, to generate a new parameter vector. If this new vector is better (as defined by the function we are trying to minimize) than any of the vectors in the set, it is added to the set, replacing the worst vector. This is done iteratively until one of the vectors reaches the desired minimum or a maximum number of iterations is done. In our case the parameters are all the degrees of the LLDGM code, as well as the rates of each of the LDGM codes forming it. The function we are trying to minimize is one minus the mutual information reached by the LLDGM code defined by the parameters after 100 iterations.

Tables 7.1 - 7.4 show the best obtained codes for a Binary-Input AWGN (BIAWGN) channel, their corresponding theoretical thresholds, as well as the Constrained-input Capacity Limit (CCL) corresponding to each rate. The code parameters shown in the tables fully determine each code (i.e., the remaining code parameters can be obtained from these ones). Note that the rate $R_c = 0.5$ two-layer LDGM code is very similar to the SCLDGM code optimized in Chapter 3. As explained previously, the main difference is that no restriction is imposed in the edges received by the inner layer of checks.

R_c	d_v	Thresh	CCL	Gap
0.95	4	4.56	4.20	0.36
0.90	5	3.69	3.20	0.49
0.85	6	3.24	2.55	0.69

Table 7.1: Systematic regular LDGM codes

We also assessed the actual performance of the best codes found for BIAWGN channels via Monte Carlo simulations. The results are plotted in Figs. 7.2 - 7.4. In all cases, a maximum of 100 iterations of SPA decoding is performed. Also, the block length is always $K = 10000$ information bits (except for Fig. 7.4). Each simulated point contains 100 bit errors or, when this was not computationally affordable, 20000 transmitted blocks.

In the sequel we analyze the performance of the optimized LLDGM codes focusing on the relationship between the rate regime at wich capacity is approached and the number of code layers.

Let us start considering the difference between the high and medium rate regimes. Table 7.1 shows the optimized single LDGM codes for high rates, while Table 7.2 shows the optimized

R_c	R_1	R_2	$d_0^1\ d_0^2\ d_1^2$	Thresh	CCL	Gap
0.95	0.985585	0.963895	2 3 2	4.66	4.20	0.46
0.90	0.971539	0.926365	2 2 2	3.72	3.20	0.52
0.85	0.944523	0.899925	2 2 2	3.03	2.55	0.48
0.5	0.979427	0.510503	3 6 45	0.43	0.18	0.26
0.2	0.893763	0.223773	3 12 21	-0.65	-0.96	0.31
0.1	0.835640	0.119669	3 20 44	-0.95	-1.27	0.32
0.05	0.793659	0.062999	4 37 68	-1.03	-1.43	0.40

Table 7.2: Optimized systematic regular Two-layer LDGM codes

R	$\{d_u\}$	$\{a_u\}$	Thr.	CCL	Gap
0.5	3 9 24	0.668496 0.195654 0.135850	0.41	0.18	0.23
0.2	4 22 146	0.678028 0.270974 0.050998	-0.60	-0.96	0.36
0.1	5 28 92	0.699420 0.205095 0.095486	-0.52	-1.27	0.75

Table 7.3: Optimized systematic IRA codes

Two-layer LDGM codes. For $R = 0.95$, it is possible to find a single systematic regular LDGM code with a theoretical convergence threshold near the CCL (only 0.36dB away). In this case, it is not worth to employ a Two-layer LDGM code and, indeed, the best code found with such structure has a theoretical threshold further away from the CCL (0.46dB away). The actual performance of these codes (see Fig. 7.2) also shows that the single systematic regular LDGM code has a better threshold than the Two-layer LDGM code. For a target BER=10^{-4} (that we will use always as a benchmark in the sequel) the threshold of the former is 0.42dB away from the CCL while that of the latter is 0.55dB away from the CCL. Note, however, that the Two-layer LDGM code presents a lower error floor.

When considering the lower rate $R = 0.9$, the best single LDGM code still shows a better theoretical threshold than the best Two-layer LDGM code, but the difference between them is much slighter (0.49dB and 0.52dB away from the CCL, respectively). The actual performance of these two codes draws similar conclusions: their actual thresholds are 0.77dB and 0.80dB

R	R_1	R_2	R_3	$d_0^1\ d_0^2\ d_0^3\ d_1^2\ d_1^3\ d_2^3$	Thresh	CCL	Gap
0.5	0.978533	0.978583	0.522152	2 2 6 7 50 50	0.5	0.18	0.32
0.2	0.937629	0.809723	0.263428	3 2 11 45 40 2	-0.68	-0.96	0.28
0.1	0.867084	0.622162	0.185368	3 2 19 10 17 2	-1.03	-1.27	0.24
0.05	0.869119	0.458355	0.125513	3 3 43 27 13 2	-1.31	-1.43	0.12
0.02	0.866780	0.206208	0.111896	3 14 95 63 57 2	-1.35	-1.53	0.18

Table 7.4: Optimized systematic regular Three-layer LDGM codes

R	R_1	R_2	$\{d_0^1\}$	$\{a_0^1\}$	d_0^2	d_1^2	Thresh	CCL	Gap
0.5	0.850285	0.588038	2 3	0.937937 0.062063	3	21	0.39	0.18	0.21
0.2	0.845496	0.236547	3 46	0.987581 0.012419,	13	4	-0.69	-0.96	0.27
0.1	0.799873	0.125019	3 13	0.982002 0.017998	22	5	-1.08	-1.27	0.19
0.05	0.723015	0.069154	3 26	0.971430 0.028570	43	20	-1.33	-1.43	0.10
0.02	0.530587	0.037694	3 7 16	0.890847 0.073981 0.035172	96	27	-1.22	-1.53	0.31

Table 7.5: Optimized systematic IRA-LDGM codes
ls

away from the CCL. Again, the Two-layer LDGM code exhibits a worse threshold but a better error floor. When the rate is decreased down to $R = 0.85$, the situation reverses: now the threshold of the best single LDGM code is 0.69dB away from the capacity limit, while the best Two-layer LDGM code is only at 0.48dB. The actual performance curves corroborate the better threshold of the Two-layer LDGM code (0.57dB away from the CCL whereas that of the single LDGM code is 0.76dB away) showing that it also has a lower error floor.

We now focus on the rate $R = 0.5$. The best Two-layer LDGM code has a better theoretical threshold (0.26dB away from the CCL whereas that of the Three-layer LDGM code is 0.32dB away). Fig. 7.3 also shows that the Two-layer LDGM code presents a better threshold (0.55dB away from the CCL) than that of the Three-layer LDGM code (0.67dB away from the CCL).

For the rate $R = 0.2$, the best codes for these two structures exhibit a very similar behaviour. Their theoretical thresholds are quite similar (0.28 and 0.31dB away from the CCL for the Three-layer LDGM code and the Two-layer LDGM code, respectively) and also are their actual ones (roughly 0.7dB away from the CCL in both cases). However, if we decrease the rate down to $R = 0.1$, Three-layer LDGM codes perform better: the best code found has a theoretical threshold only 0.24dB away from the capacity limit, while that of the best Two-layer LDGM code is 0.32dB away. Note that the capacity limit for this rate is at $E_b/N_0 = -1.32$dB, only 0.27dB away from the ultimate Shannon limit (i.e. the limit when $R \to 0$). The actual thresholds of both the Two-layer LDGM and the Three-layer LDGM codes (0.72 and 0.52dB away from the CCL, respectively) are in agreement with their theoretical ones, and both codes have a comparable error floor.

By further decreasing the rate we reach a point when Two-layer LDGM codes are insufficient to perform near the capacity limits, making necessary to resort to Three-layer LDGM codes. For a rate $R = 0.05$, Two-layer LDGM codes exhibit a severe performance degradation, being completely unable to approach the theoretical limits. The theoretical threshold of the best Two-layer LDGM code found for this rate is 0.40dB away from the CCL (while for the previous rates the gap to the CCL was around 0.3dB). In addition, its actual threshold is far away not only from the CCL, but also from its theoretical threshold (0.78dB, while for the previous rates this gap was at most 0.4dB). Also, the slope of the actual threshold is much smoother than that observed for higher rates. However, Three-layer LDGM codes are perfectly able to perform near the capacity limits: the threshold of the best code found is at

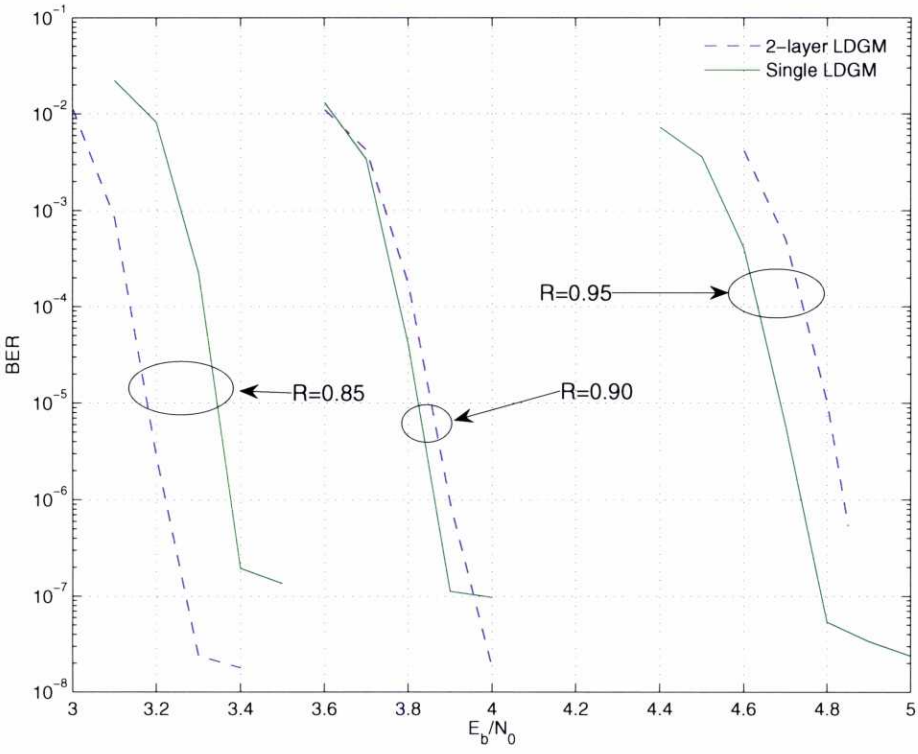


Figure 7.2: LDGM Vs Two-layer LDGM codes for high rates.

$E_b/N_0 = -1.31$dB, only 0.12dB away from the CCL. The actual performance of this code for finite block lengths shows little degradation with respect to the predicted theoretical threshold: for a target BER=10^{-4} the required E_b/N_0 is -0.95dB. Finally, we study the behaviour of Three-layer LDGM codes for $R = 0.02$, where the performance of the best Two-layer LDGM codes are very far away from the capacity limits. The theoretical threshold of the best Three-layer LDGM code is at $E_b/N_0 = -1.35$dB, only at 0.24dB from the ultimate Shannon limit. Its actual performance curve shows a required $E_b/N_0 = -0.97$dB for a target BER=10^{-4}.

Fig. 7.4 plots the actual performance of the best Three-layer LDGM code for $R = 0.05$ when varying the block length. Note the gap of less than 0.2dB at BER=10^{-4} between the curves corresponding to block lengths of $K = 10000$ and $K = 80000$ systematic bits. The latter curve is only 0.18dB away from the code theoretical threshold. Although the performance penalty of using only $K = 10000$ information bits is noticeable, we have used this block length in all the other simulations for complexity reasons: when $R = 0.05$ the codeword length for $K = 10000$ bits is 200000 bits, while for $K = 80000$ bits is 1600000 bits.

Similar conclusions can be extracted about the behaviour of IRA and IRA-LDGM codes. While for rate $R = 0.5$ IRA codes are preferable to IRA-LDGM codes, the situation starts to reverse when the rate is decreased down to $R = 0.2$. In this case, the theoretical threshold

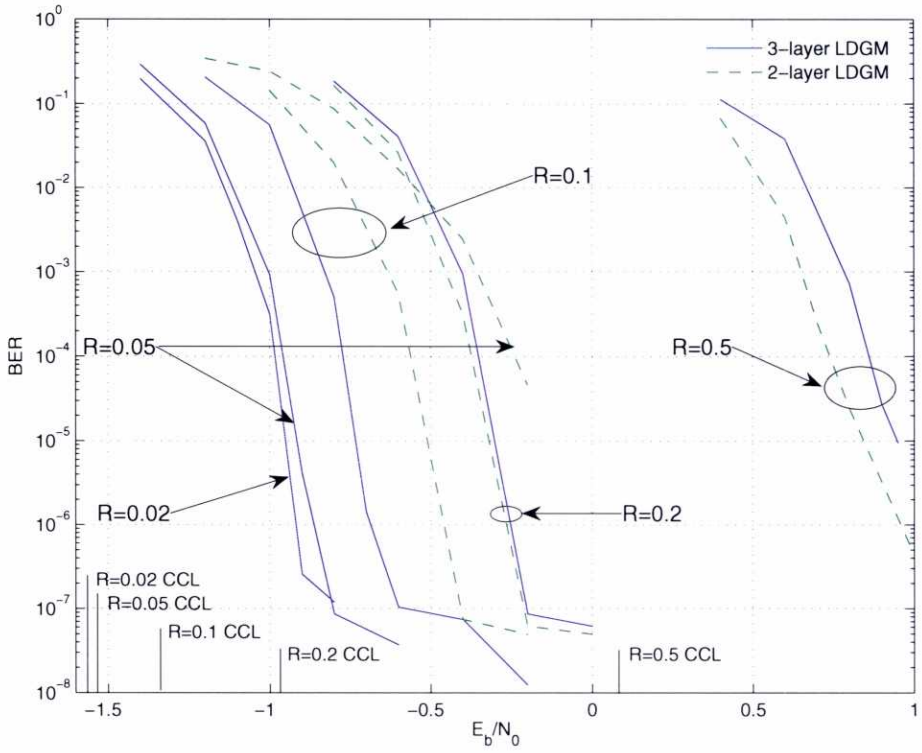


Figure 7.3: Two-layer LDGM codes Vs Three-Layer LDGM codes for low rates.

of the best IRA code (see Table 7.3) is at $E_b/N_0 = -0.60$dB, while that corresponding to the best IRA-LDGM code (see Table 7.5) is at $E_b/N_0 = -0.69$dB. The CCL for this rate is at $E_b/N_0 = -0.96$dB. In addition, the actual performance curves (see Fig. 7.5) show that the gap between the actual thresholds is even greater than that observed between the theoretical ones (0.3dB instead of 0.09dB).

For a rate $R = 0.1$dB, IRA codes are unable to perform near the CCL, as shown by the fact that the threshold of the best IRA code is far away (0.75dB) the CCL. On the other hand, IRA-LDGM codes are able to perform near the CCL: the threshold of the best code is only 0.19dB away from the CCL, and its actual performance curve shows a small degradation in threshold (it requires $E_b/N_0 = -0.75$dB for a BER$= 10^{-4}$) while showing a quite low error floor. Even more impressive results are observed for a rate $R = 0.05$: the threshold of the best IRA-LDGM code found is only 0.10dB away from the CCL (and only 0.2dB away from the ultimate Shannon limit). Its actual threshold for finite block lengths (measured at BER$= 10^4$) is located at $E_b/N_0 = -0.95$dB. Finally, for a rate $R = 0.02$, it can also be seen that IRA-LDGM codes are still able to perform near the CCL.

Finally, we compare the behaviour of Three-layer LDGM codes versus that of IRA-LDGM codes. Tables 7.4 and 7.5 show that the theoretical threholds are very similar. Fig. 7.6 plots

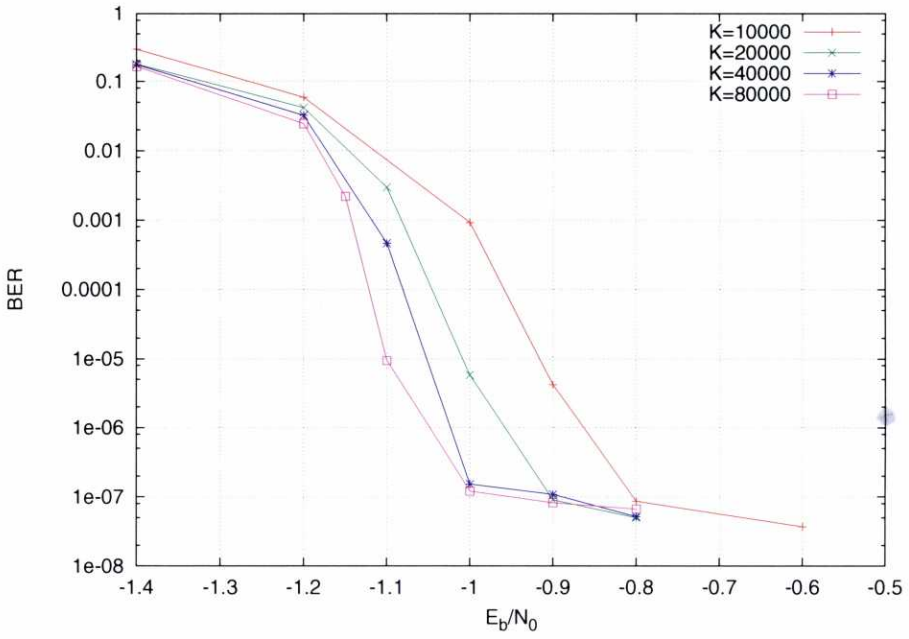


Figure 7.4: Performance of $R = 0.05$ Three-layer LDGM code for different block lengths.

the actual performance curves of these two types of codes for the different considered rates. In general, we can conclude that the performance achieved by the IRA-LDGM and the SCLDGM schemes are approximately the same.

7.3 A practical application: multiuser IDMA system

All the systems that we have studied in previous chapters are single user systems, i.e., there is only one source of information that is encoded and sent through the channel to one destination. In multiuser systems, there are several sources of information that are encoded independently and share the same communications channel. In the receiver, the detector and decoder must be able to separate and obtain the original information of the different users independently.

There exist different channel access methods that allow the separation of the different sources at the receiver [89], such as Time Division Multiple Access (TDMA), that allows the transmitters to use all the bandwith by allocating different time slots for every user; Frequency Division Multiple Access (FDMA), in which all the transmitters use the channel at the same time, but in different bandwith segments; or Code Division Multiple Access (CDMA) [90], a spread spectrum technique that employs orthogonal codes to allow an easy decoupling in the receiver. However, all these systems perform far from the achivable capacity limits.

As an alternative, the separation of the sources at the receiver could rely on the channel

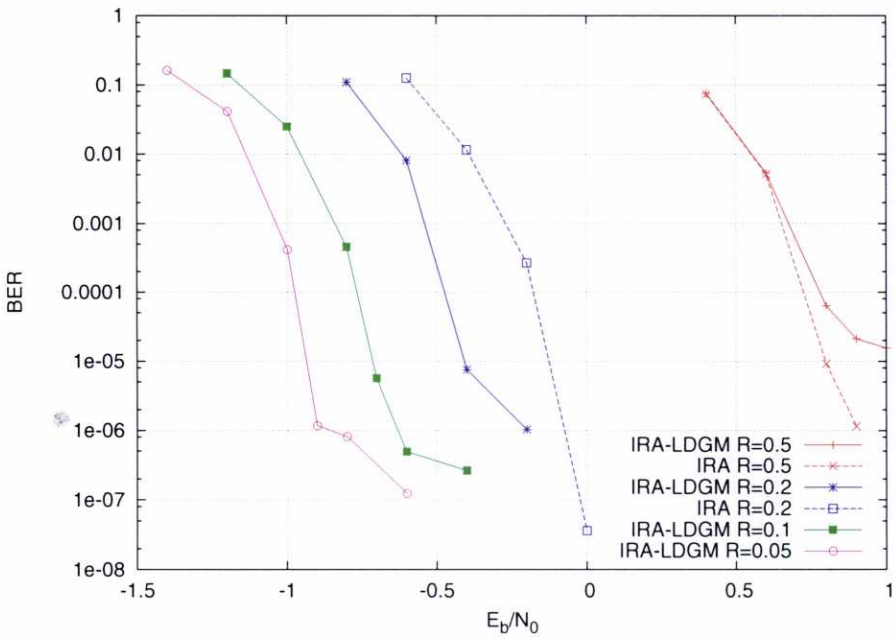


Figure 7.5: IRA Vs IRA-LDGM codes for low rates.

codes themselves. This technique is called IDMA (Interleave Division Multiple Access) [91], since each user employs a different interleaver to avoid the problem of ambiguity. However, for this to work, it is necessary that the rate of the code employed by each user be low, since the SINR (Signal to Interference-plus-Noise Ratio) decreases with the number of users. This is an interesting application for LLDGM codes, since they are able to perform close to the capacity limits with very low rates and with a low complexity.

7.3.1 Detection in multiple access channels

Let us consider a fading Multiple Access Channel (MAC), in which the signal at reception at each time interval can be expressed as

$$x = \sum_{j=1}^{T} h_j s_j + n, \tag{7.3}$$

where T is the number of transmitting users, $s_j \in \mathcal{S}$ the symbol transmitted by user j, h_j its channel coefficient and n is the white Gaussian noise with variance σ^2 (equal to $N_0/2$ per dimension). The AWGN MAC can be seen as a particular case of this model, when all the channel coefficients are equal to the unity. Notice that in this case destructive sum of transmitted symbols may occur, posing a difficulty in reliably recovering the transmitted streams at reception.

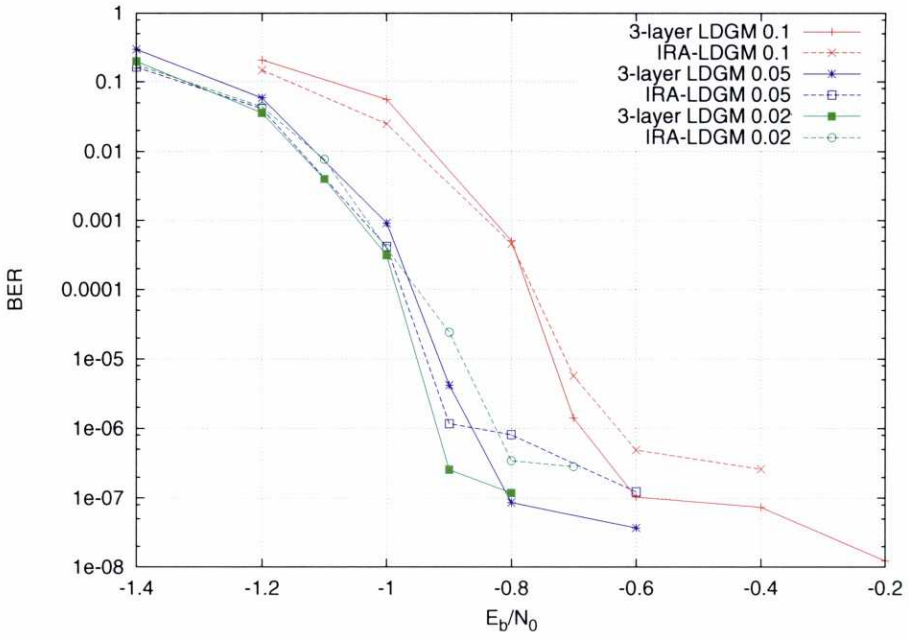


Figure 7.6: Three-layer LDGM Vs IRA-LDGM codes for low rates.

Prior to the decoding of each user channel code, we need to extract from the channel observations the information relevant to each user, that is, the channel bit LLRs. This task is accomplished by a MAC detector. Optimum MAP LLR computation presents a complexity exponential in the number of transmitting users, T, and bits per modulation symbol, $\log_2 |\mathcal{S}|$, so it becomes infeasible for almost any multiuser scenario of practical interest. Since we are considering a single-antenna receiver, the matrix formed by the channel coefficients is rank-defficient and, thus, makes it more difficult to use the suboptimum detection methods studied in Chapter 4, LSD and SIC-MMSE. We can still, however, perform Soft Interference Cancellation (SIC) but without any MMSE filtering.

The SIC detector performs an iterative soft cancellation of the symbols transmitted by the different users, based on the extrinsic information provided by the corresponding channel decoders. According to this, the following observation is obtained at the k-th iteration

$$x_k = x - \sum_{j=1, j\neq k}^{T} \bar{s}_j h_j, \quad k = 1, \ldots, T, \tag{7.4}$$

where $\bar{s}_j$ is the expected symbol for user j, which is estimated as

$$\bar{s}_j = \sum_{s\in\mathcal{S}} p(s_j = s)s \tag{7.5}$$

where $p(s_j = s)$ is computed from the extrinsic information provided by the channel decoder.

The new observation x_k is assumed to contain only the symbol transmitted by the k-th user corrupted by AWGN. However, this AWGN results from the original noise plus the remaining interference after soft cancellation, so we have

$$x_k \sim \mathcal{N}\left(h_k s_k, \sigma^2 + \sum_{i \neq k} \sigma_i^2 |h_i|^2\right), \quad k = 1, \ldots, T, \tag{7.6}$$

where σ_i^2 is the variance of the estimated symbol $\bar{s}_i$, calculated as in (4.23). Clearly, the complexity of this detector grows linearly with the number of transmitting users.

We are interested in the characterization of the SIC detector in terms useful to perform code design and to predict its performance. Such characterization is provided by the EXIT function. For low SNRs and a moderately large number of users, the SIC detector loses no capacity with respect to the optimum detector. We will illustrate this fact in the cases where the optimum detector is feasible to implement. We will see that this is also true for higher number of users and modulation orders since, for the studied examples, the behaviour of the SIC detector gets closer to that of the optimum detector when increasing the number of transmitting users and/or the modulation order. We will also support our claim on the fact that the gap with respect to the unconstrained-input channel capacity of codes designed to match the EXIT function of the SIC detector remains small.

Figure 7.7 plots the EXIT function of the optimum and SIC detectors for four users in an AWGN MAC. It is clear that lowering the SNR makes the EXIT function of the SIC detector closer to that of the optimum detector. Since the target SNRs of our LLDGM IDMA scheme are very low, using SIC detection results in no losses with respect to optimum MAP detection. In addition, it is unnecessary to consider layered detection, i.e. to apply optimum detection for groups of, say 4, users and then SIC among those groups, for a higher number of users, because the effective SNR seen by each layer is very low, and in such situation the SIC detector has the same behaviour as the optimum detector. Similar results can be observed in the Rayleigh fading MAC (see Figure 7.7).

Let us now comment on the EXIT function of the SIC detector for different number of users and modulation formats. Figure 7.8 plots the EXIT function of the SIC detector for 4 to 10 users with BPSK (top) and QPSK with Gray mapping (bottom) in an AWGN MAC. The curves are plotted for the SNR values corresponding to the theoretical thresholds of the best Three-layer LDGM codes found (see Table 7.6). It is clearly seen that the EXIT functions for BPSK and QPSK are practically identical, except for a displacement in the SNR value. Moreover, since the EXIT curve for single-user / single-antenna systems with Gray mapping is an almost straight horizontal line, this means that code design to match the EXIT function of the SIC detector will be very important if channel capacity is to be approached. Also, note that the EXIT curves get steeper as the number of users increases. As we will see in the simulation results, the performance of codes optimized for the single-user AWGN channel but used over a MAC will suffer important degradation, getting more severe as the number of users increases.

Figure 7.9 plots the same EXIT functions in a Rayleigh fading channel. In this case the curves for BPSK and QPSK are different: the QPSK curves are steeper. This means that the

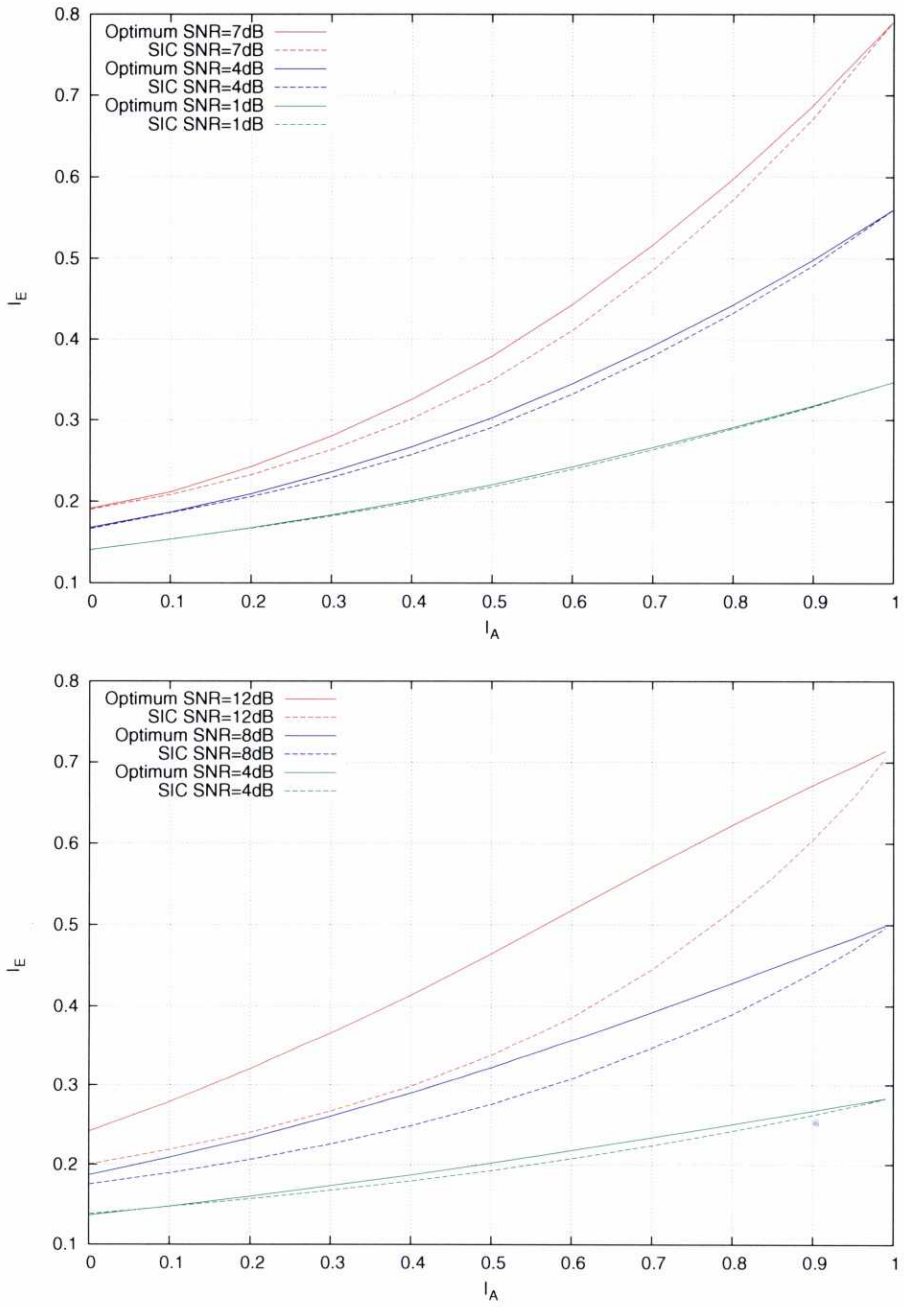


Figure 7.7: EXIT characteristics of the optimum and SIC detectors for an AWGN (top) and Rayleigh fading (bottom) MAC with $T = 4$ users using QPSK.

Users	R_1	R_2	R_3	$d_0^1\ d_0^2\ d_0^3\ d_1^2\ d_1^3\ d_2^3$	UCL	BPSK Thr. (gap)	QPSK Thr. (gap)
4	0.943197	0.381348	0.13901	3 12 35 6 100 2	-0.98	-0.81 (0.17)	-0.80 (0.18)
6	0.964436	0.469823	0.11035	3 12 41 9 66 2	-0.63	-0.43 (0.20)	-0.39 (0.24)
8	0.965925	0.579352	0.089348	3 7 43 18 29 3	-0.33	0.03 (0.36)	0.07 (0.40)
10	0.965685	0.680349	0.076103	3 4 44 11 2 5	0.0	0.41 (0.41)	0.44 (0.44)
20	0.954910	0.572709	0.091427	3 2 32 85 89 2	1.76	2.39 (0.63)	2.45 (0.69)
30	0.948744	0.698884	0.075408	5 2 6 90 75 79	3.68	5.35 (1.67)	

Table 7.6: Optimized systematic regular Three-layer LDGM codes with rate $R = 0.05$ per user for AWGN MAC

Users	R_1	R_2	R_3	$d_0^1\ d_0^2\ d_0^3\ d_1^2\ d_1^3\ d_2^3$	Thresh	UCL	Gap
4	0.963746	0.548778	0.094539	3 7 44 15 8 5	-1.22	-0.81	0.41
6	0.963613	0.630698	0.082271	3 5 42 5 53 8	-1.06	-0.71	0.35
8	0.965849	0.495411	0.10449	3 9 44 15 23 3	-0.9	0.55	0.45
10	0.955075	0.466206	0.11229	4 11 42 11 20 2	-0.75	-0.32	0.43

Table 7.7: Optimized systematic regular Three-layer LDGM codes with rate $R = 0.05$ per user for BPSK ergodic Rayleigh fading MAC.

codes for BPSK and QPSK must be optimized independently. This also means that, since it is easier to optimize codes for flatter curves, we will be able to reach a higher number of users in BPSK than in QPSK for the same code rate, as we will see in the next section.

Using the SIC detector EXIT functions, we optimized Three-layer LDGM codes for various AWGN and Rayleigh fading MACs, assuming that the total MAC capacity is equally partitioned among the users (i.e. each user has the same available data rate, $R = 0.05$). Tables 7.6, 7.7 and 7.8 list the resulting codes, indicating the corresponding gap between the predicted threshold and the unconstrained-input (i.e. Gaussian input) capacity limit, UCL. Notice that, since the detector EXIT function changes with the number of users, so it does the optimum code: there is a different optimum code for each number of users.

7.3.2 Simulation results

In this Section, we present extensive simulation results showing the performance of the proposed MAC scheme in various situations. In the case of AWGN MAC, each user utilizes a different random bit interleaver at the output of the channel encoder, to avoid the ambiguity. In Rayleigh fading channels this is not necessary since the fact that each user transmits through different channel realizations avoids ambiguity. The codes employed in all the simulations are Three-layer LDGM codes with an information block length of 4096 bits.

Let us first consider the case of codes optimized for a single-user AWGN channel when applied in a MAC environment. Figure 7.10 shows the performance, for different numbers of users, when the code rate is $R = 0.05$ and BPSK is used. The performance is relatively good for $T = 4$ users, with a gap to the unconstrained capacity limit of approximately 0.9dB at a BER of 10^{-4}. However, the performance quickly gets very far away from the capacity

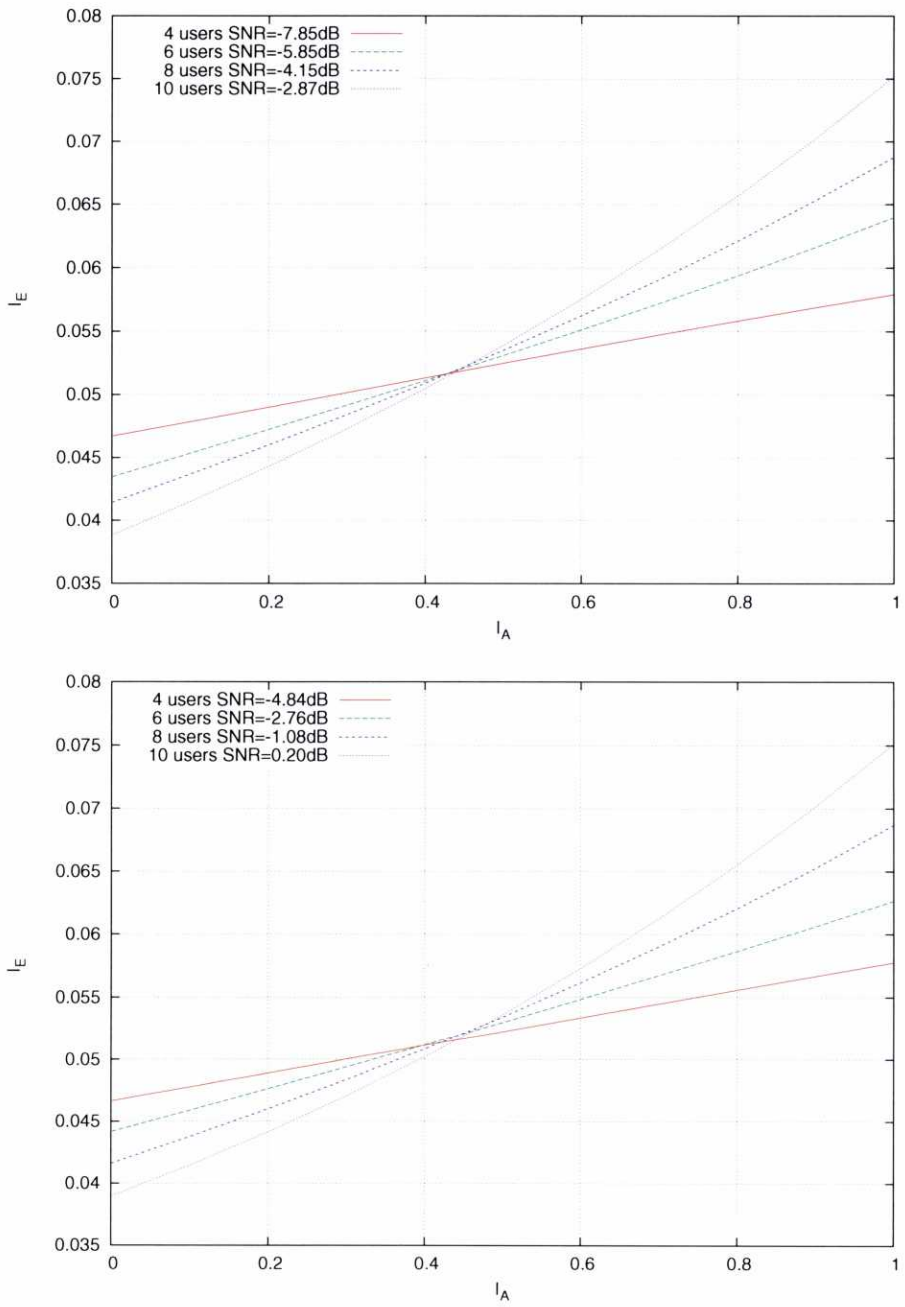


Figure 7.8: EXIT characteristics of the SIC detector for an AWGN MAC with several numbers of users. BPSK (top) and QPSK (bottom).

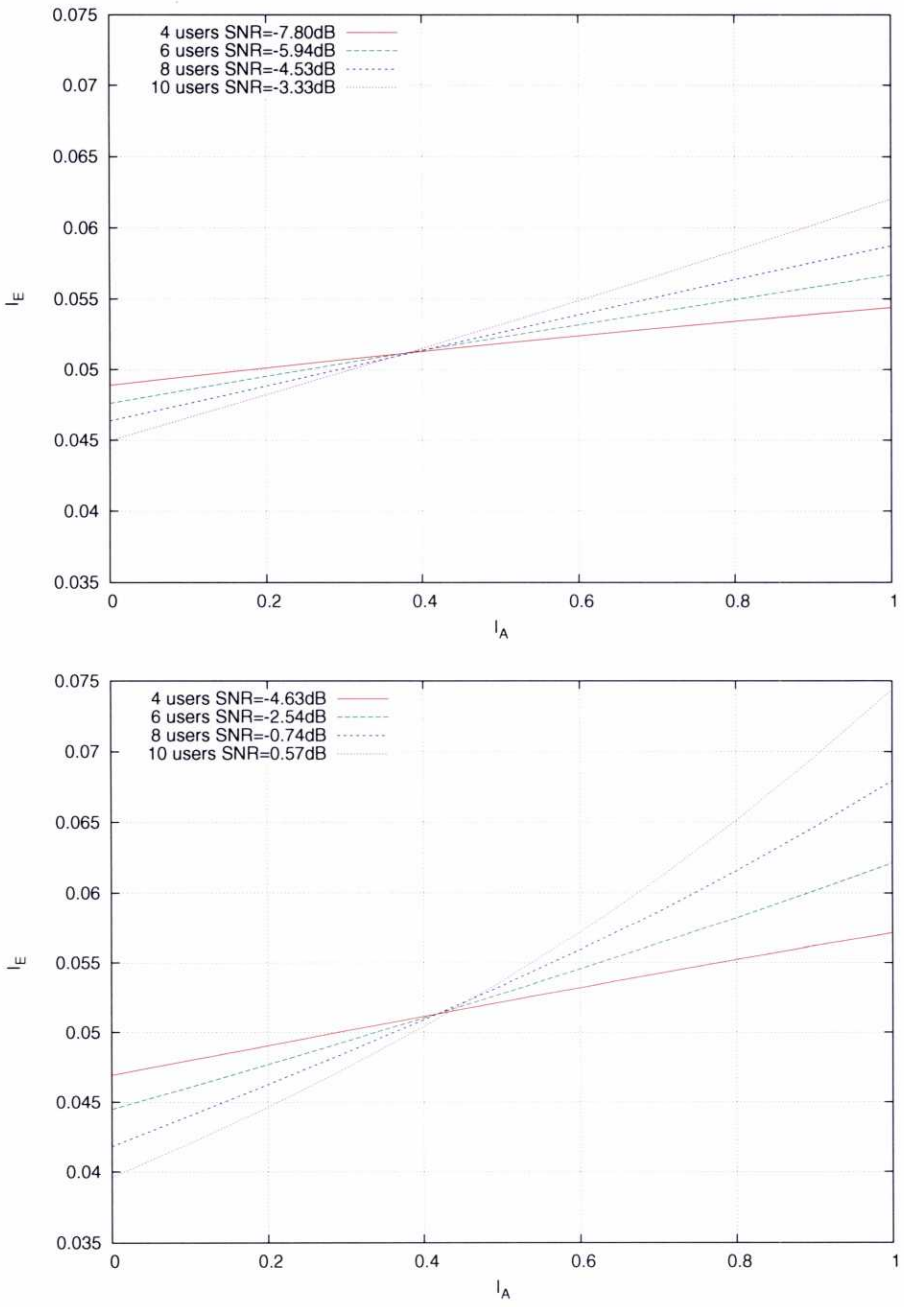


Figure 7.9: EXIT characteristics of the SIC detector for a Rayleigh fading MAC with several numbers of users. BPSK (top) and QPSK (bottom).

Users	R_1	R_2	R_3	$d_0^1\ d_0^2\ d_0^3\ d_1^2\ d_1^3\ d_2^3$	Thresh	UCL	Gap
4	0.965172	0.434980	0.10449	3 14 42 12 64 2	-0.85	-0.65	0.20
6	0.966853	0.460545	0.11229	3 12 44 21 12 2	-0.54	-0.32	0.22
8	0.958066	0.475353	0.10979	3 8 43 23 3 2	-0.21	0.23	0.38
10	0.970263	0.670106	0.076902	3 5 45 13 6 4	0.11	0.57	0.47

Table 7.8: Optimized systematic regular Three-layer LDGM codes with rate $R = 0.05$ per user for QPSK ergodic Rayleigh fading MAC

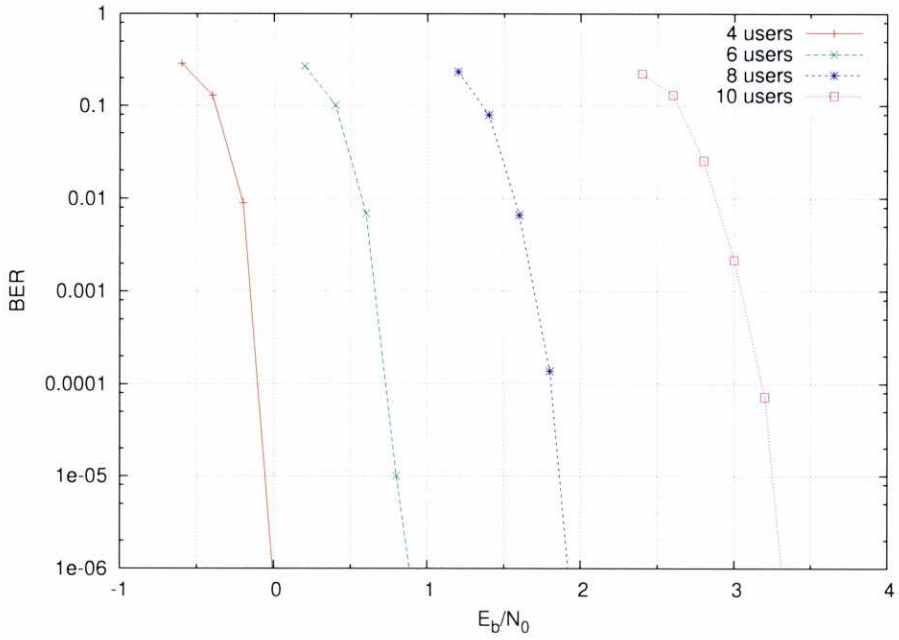


Figure 7.10: Performance of a Three-Layer LDGM code with rate $R = 0.05$ per user optimized for single-user AWGN channels when used in an AWGN MAC for several number of users. BPSK.

limit when we increase the number of users (the gap for 10 users is 3dB). This is in agreement with our previous discussion about the EXIT function of the detector: since it gets steeper when increasing the number of users, a code optimized for a flat EXIT function (corresponding to a single-user AWGN channel) also experiments increasing degradation. To overcome this issue, the codes must be optimized for the particular code rate considered and for the EXIT function of the SIC detector corresponding to the actual number of users. Figure 7.11 shows the performance of the MAC system for BPSK and QPSK using the optimized codes. The gaps to the unconstrained-input capacity limit are lower than 1dB in all cases, showing the prediction accuracy of the EXIT evolution method.

Similar results can be observed in Rayleigh fading channels (Figure 7.12). In this case, however, since the curves for BPSK are less steep than those of QPSK, the AWGN optimized

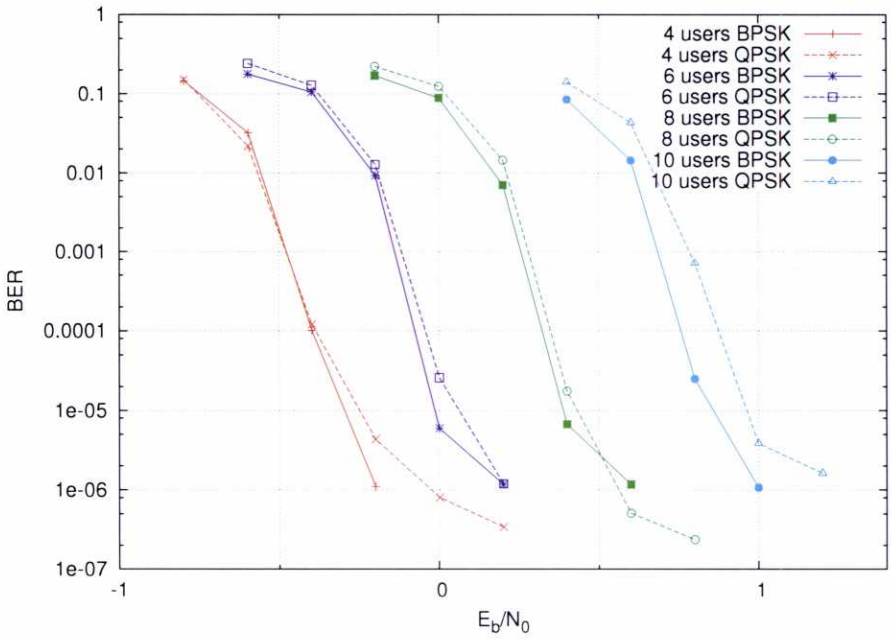


Figure 7.11: Performance of optimized Three-Layer LDGM codes on an AWGN MAC for several number of users. BPSK and QPSK. Code rate per user is $R = 0.05$.

codes perform better in BPSK than in QPSK. In any case, they are far from the curves of the codes optimized for Rayleigh fading MAC in both cases, and get further away as we increase the number of users.

Constell.	Rate	R_1	R_2	R_3	$d_0^1\ d_0^2\ d_0^3\ d_1^2\ d_1^3\ d_2^3$	Thresh.	UCL	Gap
BPSK	0.05	0.944773	0.598414	0.088438	3 2 31 17 91 5	2.18	0.91	1.27
QPSK	0.05	0.965412	0.691712	0.074874	3 2 6 99 94 78	7.0	3.75	3.25
QPSK	0.02	0.954019	0.313275	0.066919	3 11 100 54 14 2	0.98	0.38	0.60

Table 7.9: Optimized systematic regular Three-layer LDGM codes for 30 users Rayleigh fading MAC

We have also studied the performance of Three-layer LDGM codes with a high number of users. Specifically, we have searched for optimized codes for 20, 30 and 40 users using BPSK constellations. In the case of 20 users, it is possible to find a rate $R = 0.05$ code that leads to a performance approximately 0.63dB away from the capacity limit. For a higher number of users, however, the distance to the capacity limit for this per-user code rate increases to more than 1.5dB. This is due to the very high slope of the EXIT chart at the SNR corresponding to the capacity limit for 30 users (1.5 bits per channel use). To be able to approach capacity with that number of users, it is necessary to lower the rate of the channel codes. Therefore, we have searched for codes with rate $R = 0.02$, which allows us to operate at a lower SNR, thus reducing

Users	R_1	R_2	R_3	$d_0^1\ d_0^2\ d_0^3\ d_1^2\ d_1^3\ d_2^3$	UCL	BPSK Thresh. (gap)
30	0.959132	0.306582	0.068015	4 14 100 36 9 2	0.34	0.86 (0.52)
40	0.954019	0.619832	0.033822	4 5 97 22 5 4	1.04	1.73 (0.69)
50	0.960670	0.682137	0.030520	4 2 95 31 3 2	1.76	2.97 (1.21)

Table 7.10: Optimized systematic regular Three-layer LDGM codes with rate $R = 0.02$ per user for AWGN MAC

the slope of the corresponding EXIT function (lower SNR implies lower slope, see Figure 7.9). Table 7.10 show the optimized codes and Figure 7.13 the performance for 20, 30 and 40 users.

7.4 Conclusions

In this chapter we have studied Layered Low-Density Generator Matrix (LLDGM) codes, a LDPC coding structure that consists of the serial concatenation of several systematic LDGM codes. We have shown that a properly optimized LDGM code with only three layers suffices to approach the theoretical limits for rates down to $1/50$. Our study has been carried out both theoretically (i.e. under the cycle free and infinite blocklength assumptions) and practically (for finite block lengths, via Monte Carlo simulations). We have also studied an analogous code structure, termed as IRA-LDGM, consisting on the serial concatenation of an IRA code and a single LDGM code. Again, when properly optimized, IRA-LDGM codes are also able to perform near the theoretical limits in the low rate regime.

In order to test the performance of these low rate codes in a practical system, we have proposed the use of Three-layer LDGM codes in multiple access channels based on Interleave-Division Multiple Access (IDMA). The idea behind IDMA is to use low rate channel codes and a soft interference cancellation detector to separate each transmitted stream. We have designed codes for AWGN and Rayleigh fading Multiple Access Channels (MACs) whose performance is better than previous IDMA schemes based on Turbo-Hadamard codes, and very close to the theoretical capacity limits in several cases of interest.

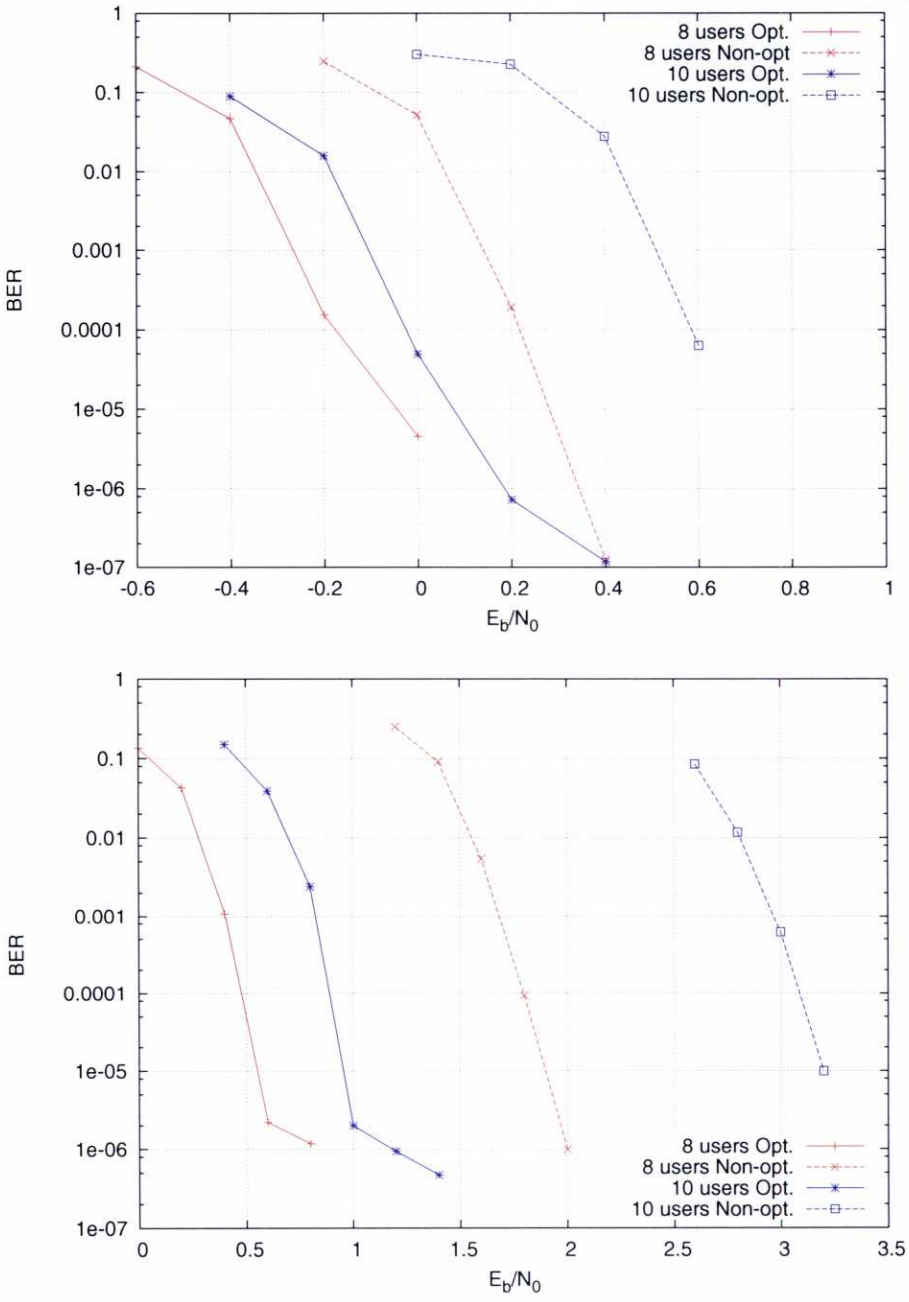


Figure 7.12: Performance of optimized vs non-optimized Three-layer LDGM codes on a Rayleigh fading MAC for BPSK (top) and QPSK (bottom).

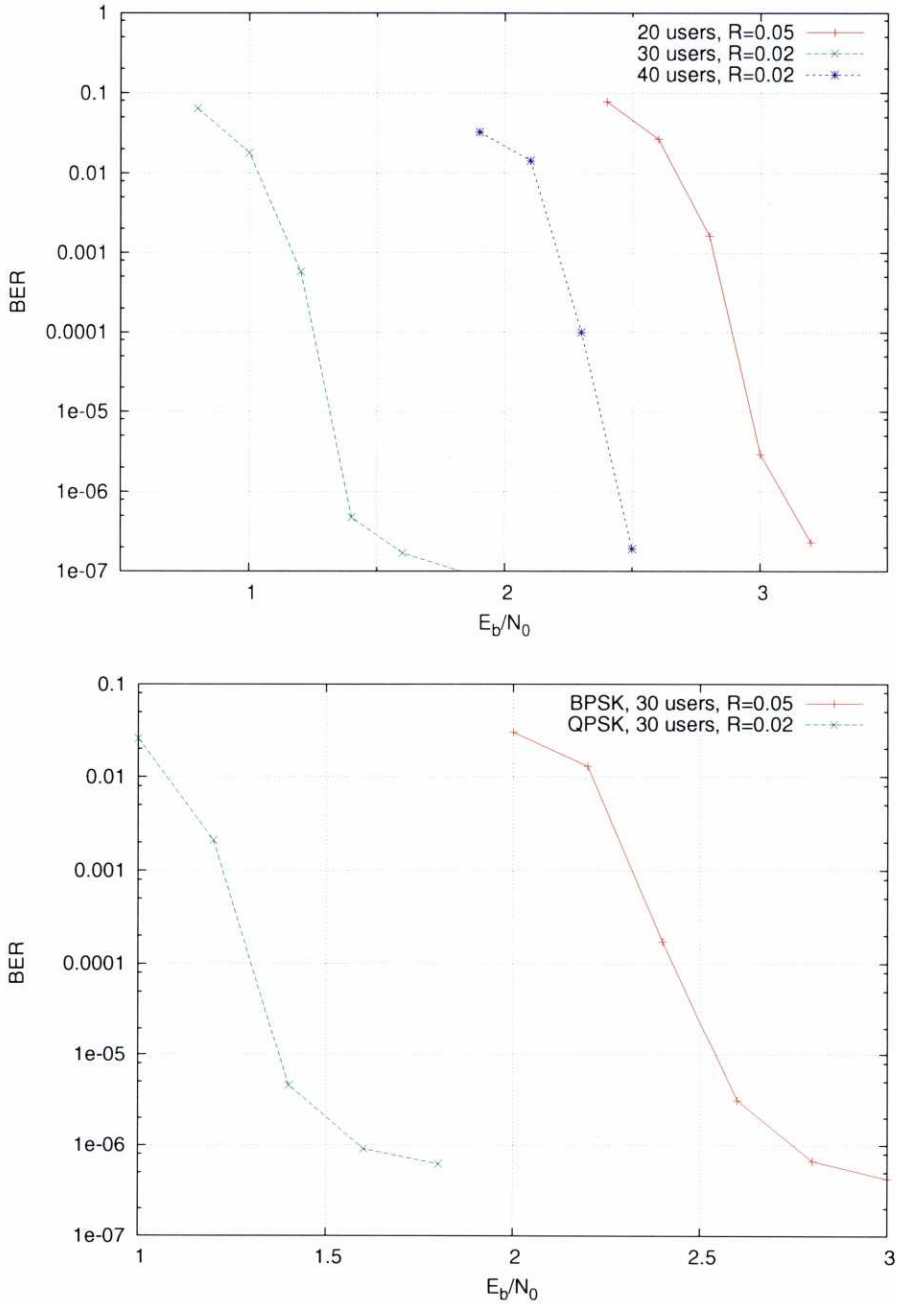


Figure 7.13: Performance of LLDGM codes with a high number of users on an BPSK AWGN (top) and Rayleigh fading (bottom) MAC.

Chapter 8

Conclusions and future work

8.1 Conclusions

The main objective of this work is to investigate the feasibility of Serially-Concatenated Low Density Generator Matrix (SCLDGM) codes as capacity approaching codes in a large number of wireless applications. This class of codes, based on the idea of using an outer high rate systematic LDGM code to correct the error floor of an inner LDGM code, had proven to perform well in SISO systems with a low encoding and decoding complexity, but the lack of adequate optimization put it far from the performance of the best LDPC and IRA codes.

We started by reviewing LDGM, Low Density Parity Check (LDPC), and Repeat-Accumulate (RA) codes in Chapter 2, as well as the algorithm that is usually employed to decode them: the Sum-Product Algorithm (SPA). To understand the iterative behavior of these codes we employed EXtrinsic Information Transfer (EXIT) charts that track the exchange of information between each of the decoder components and showed that LDPC and IRA codes are able to approach the BIAWGN channel capacity. LDGM codes, on the other hand, cannot approach capacity except for very high rates, due to their graph structure. We assesed the accuracy of the EXIT chart analysis for these three classes of codes via computer simulations.

We continued by presenting the SCLDGM signalling scheme in Chapter 3, studying their optimization and analyzing their performance when used in a Bit-Interleaved Coded Modulation (BICM) scheme over Soft-Input Soft-Output (SISO) channels. Since SCLDGM codes are based on LDPC codes, the same decoding algorithm (Sum-Product) and optimization techniques (EXIT charts) can be applied to them. However, due to their more complex structure, we used the EXIT functions in a different way than usual: instead of reducing the optimization procedure to a curve fitting problem, we perform an exhaustive search for the code that converges at the lowest E_b/N_0, using the EXIT functions as a substitute for the real SPA. This simple method allows us to find SCLDGM codes that are able to outperform optimized IRA codes in terms of both threshold and error floor, with a similar encoding and decoding complexity.

In Chapter 4 we extended the study to MIMO systems, characterized by the use of several antennas at both transmission and reception. In these systems, the capacity increase with the

SNR is proportional to the minimum number of transmit and receive antennas, which makes them suitable for applications with high throughput requirements. However, this increased capacity comes at the cost of an exponential complexity in the number of transmit antennas for optimum detection. For this reson, we studied the behavior of SCLDGM codes when combined with different suboptimum detectors. Specifically, we investigated the SIC-MMSE detector, a linear detector based on filtering and soft interference cancellation, and two List Sphere Detectors, ML LSD and MAP LSD, that perform a tree search to reduce the number of possible symbols considered when computing the bit likelihoods. To obtain good SCLDGM codes for each of the detectors, their EXIT function must be calculated first. The MAP LSD detector has the advantage of a better performance than the ML LSD, but due to a problem of positive feedback when considering a low number of candidates, the SCLDGM codes cannot be optimized for it. Instead, we proposed the Extrinsic LSD detector, that avoids using the a-priori information corresponding to the bit for which we are calculating its extrinsic information. We obtained optimized codes for all of them and proved the ability of SCLDGM codes to approach capacity in these systems.

We also considered in Chapter 5 the concatenation of SCLDGM codes with Space Time Block Codes (STBCs). STBCs are short block codes designed to exploit the properties of the MIMO channel. The main objective of these codes is to achieve maximum transmit diversity, while at the same time providing with some other advantages such as increased capacity or a simple optimum detection. The considered STBCs were the Alamouti code, that allows the detector to decouple the transmitted streams in 2×1 MIMO systems without incurring in a capacity loss; the Linear Dispersion codes, that enable us to use general suboptimum detection methods when the number of transmit antennas is lower than the number of receive antennas; and the Golden code, that increases the available capacity. With all of them, the optimized SCLDGM codes are able to approach capacity, in both ergodic and quasi-static channels.

For both the BICM and STBC schemes, we studied the problem of channel estimation in Chapter 6. We considered two different algorithms, Least Squares (LS) and Maximum-Likelihood with Expectation-Maximization (ML-EM). LS is a clasical optimization technique that finds the solution that minimizes the squared sum of residuals of a model, whereas EM finds the maximum-likelihood solution in an iterative manner and can actually be interpreted as a generalized LS. The main problem of ML-EM when applied to the estimation of the MIMO channel is the same as optimum detection: exponential complexity in the number of trasmit anntenas. To overcome this issue we proposed a reduced complexity ML-EM algorithm that, combined with a sphere detector, estimates the channel using the list of candidates calculated by the LSD. Since both the LS and ML-EM techniques rely on a small list of transmitted symbols known at the receiver (i.e., a pilot sequence) to obtain an initial estimation of the channel, we were interested in finding the optimum length of this sequence. To do this, we optimized SCLDGM codes by adapting their rate so that the number of information bits per block remained the same, and performed computer simulations to find out which of the chosen pilot sequence lengths yields the best result.

Finally, we devoted Chapter 7 to the problem of low-rate encoding. Although SCLDGM codes perform better than IRA codes for relatively low rates, below a certain point their performance is severely degraded. We proposed a new scheme, termed Layered LDGM (LLDGM) codes, that can be seen as a generalization of SCLDGM codes for more than two concatenated codes. We showed that a three layer LDGM code is able to perform very close to the capacity limit with a rate of 0.02. We applied this scheme to a multiuser system with Interleave-Division Multiple Access, and proved that it is able to perform at less than 1dB from the multiuser capacity limit in AWGN and Rayleigh fading channels using a very simple detection scheme, surpassing the performance of IDMA systems based on Turbo Hadamard codes.

8.2 Future work

8.2.1 Correlated sources in multiple access channels

As a particular case of multiuser systems, we can consider systems in which the different information sources are correlated. For example, we can assume that we have two users with a probability of sending the same information bit at a given time instant being greater than 0.5. In these kind of systems, when each user transmits through a different channel, it can be shown that the capacity is maximized by separating the source and channel coding stages [92, 93]. This means that the best performance is achieved when the transmitters compress the sources first and then encode them using a capacity approaching code. The receiver decodes the observations from the channel and then decode the source bits according to the encoding scheme employed at transmission. However, in multiple access AWGN channels, decorrelating the sources before encoding them leads to a performance loss due to an increase in the number of destructive interferences. When transmitting over these channels it is interesting to keep the correlation between the senders as high as possible, since this leads to an increase in the effective received SINR. In this sense, systematic LDGM codes seem the ideal channel encoding solution because, unlike LDPC and IRA codes, they keep a high correlation in the parity bits. It has been shown [48, 94] that SCLDGM codes can indeed surpass the separation limit (the limit with separate source and channel coding) but the codes employed until now have been optimized by hand. By optimizing the codes using EXIT techniques it is reasonable to expect a significant increase in performance.

8.2.2 Code Puncturing

All the codes that we have studied along this work transmit all the bits constituting the codeword through the channel. However, it is possible not to transmit part of the codeword (thus increasing the resulting rate) and still be able to decode it. This is known as *code puncturing* and allows the obtention of a code that is closer to the capacity limit than the codes of the same rate without puncturing.

r_{out}	$d_u^{f^1}$	$d_u^{f^2}$	$d_{p^1}^{f^2}$	Thres.	Gap
0.97	3	6	72	0.34	0.15

Table 8.1: Degree profile of a rate $1/2$ punctured regular SCLDGM code for the AWGN channel. Channel capacity is at $E_b/N_0 = 0.19$dB.

The most immediate example of this is the puncturing of the outer coded bits in a 2-layer LDGM code (or a SCLDGM code). The main function of the outer parity bits is to eliminate the error floor of the inner LDGM code. This effect can be observed in the EXIT trajectories of the messages passed from the outer check nodes f_i^1 to the systematic variable nodes u_i. However, we can also observe that until a certain number of iterations has been performed, the information passed from the outer checks to the systematic nodes is zero. This means that the information passed from the outer parity variable nodes p_i^1 to the outer check nodes f_i^1 is not important until the information passed from the inner checks f_i^2 to the outer parity variable nodes p_i^1 is greater than a certain value, approximately 0.5. If we look at the equation that calculates the information passed to outer checks from the outer parity bits

$$I_{p^1}^{f^1} = J\left(d_{p^1}^{f^2} J^{-1}(I_{f^2}^{p^1}) + J^{-1}(I_{ch})\right) \tag{8.1}$$

it is apparent that the information that comes from the channel is negligible, since $d_{p^1}^{f^2}$ is usually quite large for average and high rates, and $J^{-1}(I_{f^2}^{p^1}) \approx J^{-1}(I_{ch})$ when the information $I_{p^1}^{f^1}$ is significant.

As a consequence, if we do not transmit the outer parity bits, the equations that calculate the extrinsic information from the outer parity variable nodes simplify to

$$\begin{aligned} I_{p^1}^{f^1} &= J\left(d_{p^1}^{f^2} J^{-1}(I_{f^2}^{p^1})\right) \\ I_{p^1}^{f^2} &= J\left((d_{p^1}^{f^2} - 1) J^{-1}(I_{f^2}^{p^1}) + J^{-1}(I_{f^1}^{p^1})\right) \end{aligned}$$

This allows us to add some inner parity bits to keep the rate fixed, i.e., $R_c = R_i$. If we optimize this new punctured scheme, we obtain a systematic regular SCLDGM code that converges at 0.34dB, i.e., 0.1dB lower than the unpunctured one, with the parameters indicated in Table 8.1.

This idea can be extended to systematic and inner parity bits by considering the percentage of punctured bits as another optimization parameter, probably increasing the coding gain with respect to unpunctured codes.

8.2.3 Rate-compatible SCLDGM codes

An interesting application for code puncturing is the construction of rate-compatible codes. Rate-compatible codes are low or middle rate codes that are able to perform close to the capacity limits when they are punctured to reach higher rates. This allows a single implementation of

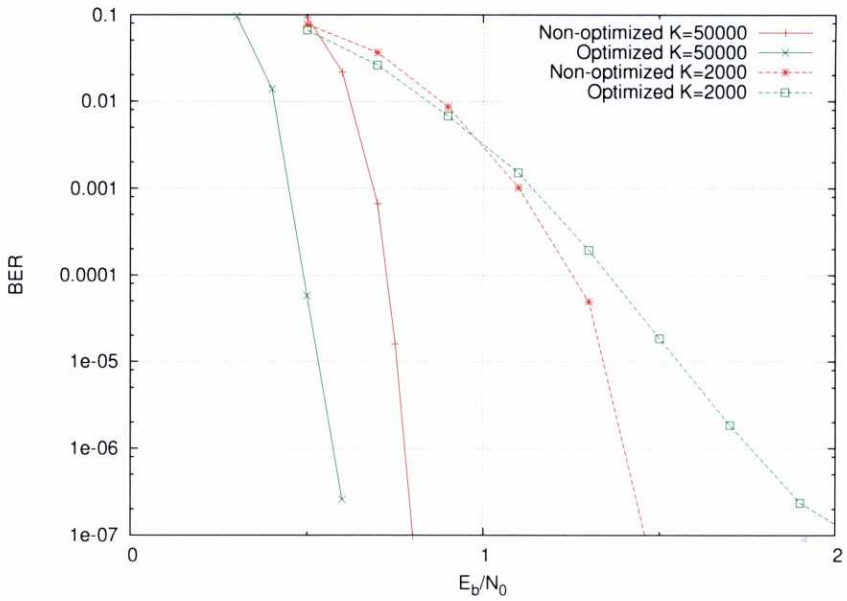


Figure 8.1: Performance of optimized and non-optimized codes for different block lengths.

a code to be used in a system with different rate requirements. The idea was introduced in [95] for BCH codes and further extended to convolutional codes in [96] and to LDPC codes in [97, 98]. A family of rate-compatible SCLDGM codes that are able to perform close to the best rate-compatible LDPCs have been recently presented [99], but the codes were not optimized. It is expected that the performance of this scheme can be improved by employing the optimization techniques presented in this work.

8.2.4 Optimization of short block length SCLDGM codes

The optimization techniques considered along this work, DE and EXIT charts, assume an infinite block length for their calculations. When the block length is short, their predicted threshold is very inaccurate, and the performance of non-optimized codes can even surpass that of optimized codes when reducing the block length (Figure 8.1). There are some procedures, like the elimination of short cycles or avoiding the use of degree-2 nodes [45], that are able to improve the behaviour of IRA and LDPC codes. However, these techniques do not have a great impact on the performance of SCLDGM codes due to the larger minimum degrees of these codes. We have empirically found codes that perform better than optimized codes when using very short block lengths, but the lack of an appropriate optimization method makes this task difficult.

8.2.5 Structured SCLDGM codes

Practical LDPC codes use structured (non-random) graphs to allow a simpler hardware implementation, decrease the encoding complexity and/or achieve lower error floors by avoiding certain short-length cycles [100, 101, 102]. All the SCLDGM codes that we have considered in this work use randomly generated matrices. It would be interesting to design structured SCLDGM codes and compare their performance with those of structured IRA and LDPC codes existing in the literature.

8.2.6 Precoding

Precoding for MIMO channels has been a topic of increasing interest in recent years [103, 104, 105]. Precoding is technique that consists in making use of the knowledge of the channel at the transmitter (that can provided by the receiver via a feedback channel) to maximize the resulting capacity, as well as to provide a method for a simple decoupling of the different transmit streams at reception. This allows for simpler receivers, since one of the main sources of complexity (detection) is moved to the precoder at the transmitter.

There exist several precoding algorithms that show a compromise between complexity and performance and that deal with different problems associated to the feedback channel. However, to the length of our knowledge, no codes have been optimized for most of them. The combination of optimized SCLDGM codes with precoding could result in a scheme that is able to achieve a higher performance with lower complexity than schemes without precoding.

8.2.7 Other types of MIMO channels

In this work we have considered the two most simple MIMO channel models, namely the uncorrelated ergodic fast-fading channel and the quasistatic channel. More realistic MIMO channels include slow time-varying MIMO channels, spatially-correlated MIMO channels or channels with Inter-Symbol Interference (ISI). The optimization techniques presented in this work must take into account the channel properties in order to achieve capacity, and therefore the codes must be properly designed for each of the channels.

Appendix A

Low Complexity implementations of the Sum-Product Algorithm

Since the output messages calculated by the variable nodes in the SPA are simply a sum of the input messages, it seems obvious that the greatest complexity is in the calculation of the check messages

$$L_{c\to v} = 2\tanh^{-1}\left(\prod_{v'\neq v}\tanh\frac{L_{v'\to c}}{2}\right) \tag{A.1}$$

To avoid the computation of the hyperbolic tangents, several suboptimum methods have been proposed. First, let us start by defining the function used to compute the check messages

$$C(x_1, x_2, \ldots, x_n) \triangleq 2\tanh^{-1}\left(\prod_{i=1}^{n}\tanh\frac{x_i}{2}\right) = \log\frac{1+\prod_{i=1}^{n}\frac{e^{x_i}-1}{e^{x_i}+1}}{1-\prod_{i=1}^{n}\frac{e^{x_i}-1}{e^{x_i}+1}} \tag{A.2}$$

If we define the function

$$\theta(x) \triangleq \log\frac{e^{|x|}+1}{e^{|x|}-1} = \log\frac{1+e^{-|x|}}{1-e^{-|x|}} \tag{A.3}$$

we can rewrite the function C as

$$C(x_1, x_2, \ldots, x_n) = -\mathrm{sgn}(x_1, x_2, \ldots, x_n)\,\theta\left(\sum_{i=1}^{n}\theta(x_i)\right) \tag{A.4}$$

where $\mathrm{sgn}(x_1, x_2, \ldots, x_n)$ is equal to -1 if there is an odd number of negative arguments and +1 in the other case. Now the main complexity is in $\theta(x)$, represented in Figure A.1. This function can be approximated by splitting the curve in several fragments and approximating each of them by a linear function. However, since the function is not bounded, the performance of the SPA when using this method will be degraded.

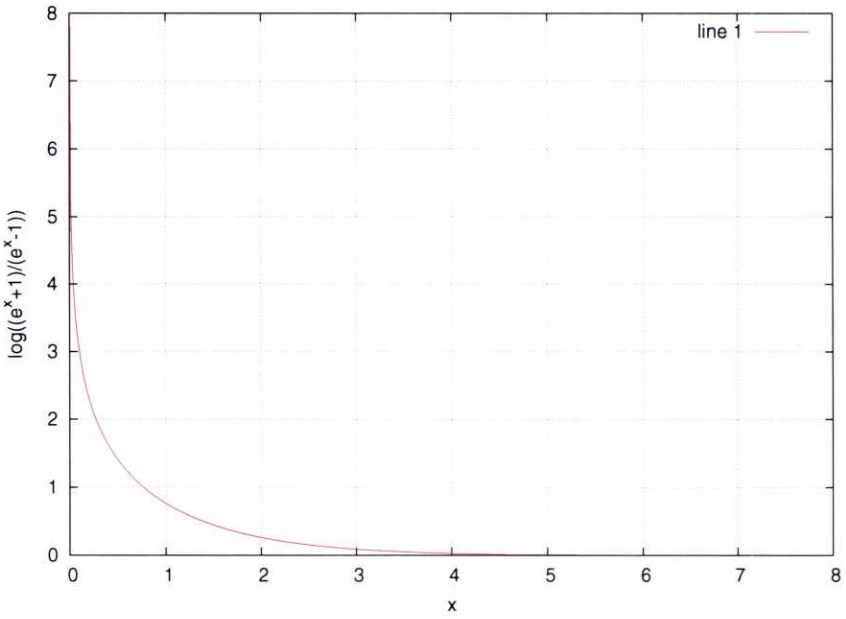


Figure A.1: Function $\theta(x) = \log \frac{e^{|x|}+1}{e^{|x|}-1}$.

To solve this issue, we must employ another approximation. It can be shown that the function C can be recursively calculated as

$$C(x_1, x_2, \ldots, x_n) = C(x_1, C(x_2, C(\ldots, C(x_{n-1}, x_n) \cdots))) \tag{A.5}$$

Elaborating the function C for two variables we have

$$\begin{aligned} C(a,b) &= 2\tanh^{-1}\left(\tanh\frac{a}{2}\tanh\frac{b}{2}\right) = \log\frac{1-\frac{e^a-1}{e^a+1}\frac{e^b-1}{e^b+1}}{1-\frac{e^a-1}{e^a+1}\frac{e^b-1}{e^b+1}} \\ &= \log\frac{1+e^{a+b}}{e^a+e^b} = -\mathrm{sgn}(a,b)\min(|a|,|b|) + \Lambda(|a|,|b|) \end{aligned} \tag{A.6}$$

being

$$\Lambda(a,b) \triangleq \log\left(1+e^{-(a+b)}\right) - \log\left(1+e^{-|a-b|}\right) = \lambda(a+b) - \lambda(|a-b|) \tag{A.7}$$

where

$$\lambda(x) \triangleq \log(1+e^{-x}) \tag{A.8}$$

With this the main source of complexity resides in the function $\lambda(x)$, represented in Figure A.2. This time the function is bounded, and the performance of the SPA using the linear approximation is practically the same as the optimum.

It is important to note that we need to compute the function C for every edge connected to the check node, so we must be careful to avoid repeating calculations. With the first

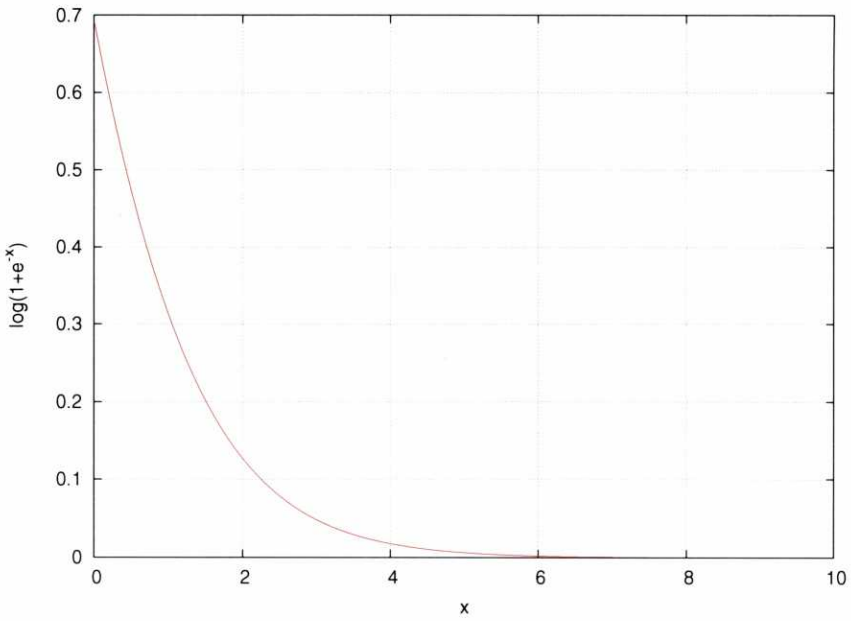


Figure A.2: Function $\lambda(x) = \log(1 + e^{-x})$.

approximation this is easy to accomplish, since we can first calculate the sum of the θ function of all the input messages

$$\Theta = \sum_{v} \theta(L_{v \to c}) \tag{A.9}$$

and then compute each output message as

$$L_{c \to v_i} = \theta\left(\Theta - \theta(L_{v_i \to c})\right) \tag{A.10}$$

However, with the second approximation we must use a forward-backward algorithm, due to the recursive computation of the messages. Thus, in the forward pass we compute the values

$$\alpha_i = C(L_{v_1 \to c}, C(L_{v_2 \to c}, C(\ldots, C(L_{v_{i-2} \to c}, L_{v_{i-1} \to c}) \cdots))), \quad i = 1 \ldots d_c - 1 \tag{A.11}$$

and in the backward pass the values

$$\beta_i = C(L_{v_{i+1} \to c}, C(L_{v_{i+2} \to c}, C(\ldots, C(L_{v_{d_c-1} \to c}, L_{v_{d_c} \to c}) \cdots))), \quad i = 2 \ldots d_c \tag{A.12}$$

in both cases using the linear approximation of the λ function. The output messages can then be calculated as

$$L_{c \to v_i} = C\left(\alpha_i, \beta_i\right) \tag{A.13}$$

The computation of the output L-values can be further simplified by taking into account that the function C is severely determined by the minimum of its arguments, which allows us to approximate the output messages as

$$L_{c \to v_i} \approx \min_{v}(L_{v \to c}) \tag{A.14}$$

for all the edges except for that which carries the minimum input L-value. For this edge we can use the second approximation to calculate the output value. This version of the SPA is known as A-min* [106], and performs approximately 0.1dB worse than the optimum.

Appendix B

MMSE filter

Given a MIMO channel model

$$\mathbf{x} = \mathbf{H}\mathbf{s} + \mathbf{n} \tag{B.1}$$

our objective is to find the linear filter for the k-th transmit antenna $\mathbf{w}_k$ that minimizes the mean square error between the symbol sent from that antenna and the estimated symbol $\hat{s}_k$, given by

$$\hat{s}_k = \mathbf{w}_k^H \mathbf{x} \tag{B.2}$$

Let us first define the error e as

$$e \triangleq s_k - \hat{s}_k = s_k - \mathbf{w}_k^H \mathbf{x} \tag{B.3}$$

The mean square error σ_e^2 is then given by

$$\begin{aligned} \sigma_e^2 &= \mathbb{E}\{|e|^2\} = \mathbb{E}\{|s_k - \mathbf{w}_k^H \mathbf{x}|^2\} = \mathbb{E}\{(s_k - \mathbf{w}_k^H \mathbf{x})(s_k^* - \mathbf{x}^H \mathbf{w}_k)\} \\ &= \mathbb{E}\{|s_k|^2\} - \mathbb{E}\{\mathbf{w}_k^H \mathbf{x} s_k^*\} - \mathbb{E}\{s_k \mathbf{x}^H \mathbf{w}_k\} + \mathbb{E}\{\mathbf{w}_k^H \mathbf{x}\mathbf{x}^H \mathbf{w}_k\} \\ &= E_s - \mathbf{w}_k^H \mathbb{E}\{s_k^* \mathbf{x}\} - \mathbb{E}\{s_k \mathbf{x}^H\}\mathbf{w}_k + \mathbf{w}_k^H \mathbb{E}\{\mathbf{x}\mathbf{x}^H\}\mathbf{w}_k \end{aligned} \tag{B.4}$$

If we denote $\mathbf{p} \triangleq \mathbb{E}\{s_k^* \mathbf{x}\}$ and $\mathbf{R}_{\mathbf{xx}} \triangleq \mathbb{E}\{\mathbf{x}\mathbf{x}^H\}$ we can rewrite (B.4) as

$$\sigma_e^2 = E_s - \mathbf{w}_k^H \mathbf{p} - \mathbf{p}^H \mathbf{w}_k + \mathbf{w}_k^H \mathbf{R}_{\mathbf{xx}} \mathbf{w}_k \tag{B.5}$$

To find the filter that minimizes σ_e^2 we take the derivative of (B.5) with respect to $\mathbf{w}_k^H$ while treating $\mathbf{w}_k$ as a constant, and equal to zero

$$\frac{\partial \sigma_e^2}{\partial \mathbf{w}_k^H} = -\mathbf{p} + \mathbf{R}_{\mathbf{xx}} \mathbf{w}_k = 0 \Rightarrow \mathbf{R}_{\mathbf{xx}} \mathbf{w}_k = \mathbf{p} \Rightarrow \mathbf{w}_k = \mathbf{R}_{\mathbf{xx}}^{-1} \mathbf{p} \tag{B.6}$$

Let us denote the k-th column of $\mathbf{H}$ as $\mathbf{h}_k$. This way we can rewrite the channel model as

$$\mathbf{x} = \mathbf{H}\mathbf{s} + \mathbf{n} = s_k \mathbf{h}_k + \sum_{i \neq k} s_i \mathbf{h}_i + \mathbf{n} \tag{B.7}$$

and thus, assuming that the transmitted symbols are uncorrelated, the vector $\mathbf{p}$ can be calculated as

$$\mathbf{p} = \mathbb{E}\{s_k^* \mathbf{x}\} = \mathbb{E}\{s_k^* s_k\}\mathbf{h}_k + \sum_{i \neq k} \mathbb{E}\{s_k^* s_i\}\mathbf{h}_i + \mathbb{E}\{s_k^* \mathbf{n}\} = \mathbb{E}\{s_k^* s_k\}\mathbf{h}_k = E_s \mathbf{h}_k \tag{B.8}$$

The autocorrelation matrix $\mathbf{R_{xx}}$ is equal to

$$\begin{aligned} \mathbf{R_{xx}} &= \mathbb{E}\{\mathbf{x}\mathbf{x}^H\} = \mathbb{E}\{(\mathbf{H}\mathbf{s} + \mathbf{n})(\mathbf{s}^H\mathbf{H}^H + \mathbf{n}^H)\} \\ &= \mathbb{E}\{\mathbf{H}\mathbf{s}\mathbf{s}^H\mathbf{H}^H\} + \mathbb{E}\{\mathbf{H}\mathbf{s}\mathbf{n}^H\} + \mathbb{E}\{\mathbf{n}\mathbf{s}^H\mathbf{H}\} + \mathbb{E}\{\mathbf{n}\mathbf{n}^H\} \\ &= \mathbf{H}\mathbf{R_{ss}}\mathbf{H}^H + \mathbf{R_{nn}} \end{aligned} \tag{B.9}$$

Since the symbols are uncorrelated, the matrix $\mathbf{R_{ss}}$ is equal to

$$\mathbf{R_{ss}} = \text{diag}\left[\sigma_{s_1}^2, \sigma_{s_2}^2, \ldots, \sigma_{s_{n_T}}^2\right] \tag{B.10}$$

and, assuming that the noise is spatially uncorrelated and has the same variance in all the receive antennas,

$$\mathbf{R_{nn}} = \sigma_n^2 \mathbf{I}_{n_R} = N_0 \mathbf{I}_{n_R} \tag{B.11}$$

Substituting these expressions in (B.6) we obtain

$$\mathbf{w}_k = \left(\mathbf{H}\mathbf{R_{ss}}\mathbf{H}^H + N_0\mathbf{I}_{n_R}\right)^{-1} E_s \mathbf{h}_k = \left(\mathbf{H}\boldsymbol{\Delta}_k\mathbf{H}^H + \frac{N_0}{E_s}\mathbf{I}_{n_R}\right)^{-1} \mathbf{h}_k \tag{B.12}$$

where

$$\boldsymbol{\Delta}_k = \text{diag}\left[\frac{\sigma_{s_1}^2}{E_s}, \frac{\sigma_{s_2}^2}{E_s}, \ldots, \frac{\sigma_{s_{n_T}}^2}{E_s}\right] \tag{B.13}$$

The MMSE filter can be applied after a soft cancellation step that subtracts from the received vector $\mathbf{x}$ the interferences caused by some of the transmit antennas. When this happens, the variances $\sigma_{s_i}^2$ corresponding to the cancelled streams must be estimated using the *a priori* probabilities that were employed in the cancellation step. The variances corresponding to the non-cancelled streams are equal to E_s.

Appendix C

List of acronyms

AWGN Additive White Gaussian Noise

APP *A Posteriori* Probability

BCJR Bahl, Cocke, Jelinek and Raviv

BER Bit Error Rate

BIAWGN Binary Input Additive White Gaussian Noise

BICM Bit-Interleaved Coded Modulation

BSC Binary Simmetric Channel

BEC Binary Erasure Channel

BPSK Binary Phase Shift Keying

CCL Constrained Capacity Limit

CNAD Check Node and Accumulator Decoder

CND Check Node Decoder

DE Differential Evolution, Density Evolution

EM Expectation Maximization

EXIT EXtrinsic Information Transfer

FER Frame Error Rate

IDMA Interleave-Division Multiple Access

IRA Irregular Repeat-Accumulate

LD Linear Dispersion

LDGM Low Density Generator Matrix

LDPC Low Density Parity Check

LLDGM Layered Low Density Generator Matrix

LLR Log-Likelihood Ratio

LS Least Squares

LSD List Sphere Detector

MAC Multiple Access Channel

MAP Maximum *A Posteriori*

MIMO Multiple-Input Multiple-Output

MMSE Minimum Mean Square Error

ML Maximum Likelihood

OSTBC Orthogonal Space-Time Block Coding

PSAM Pilot Symbol Assisted Modulation

QAM Quadrature Amplitude Modulation

QPSK Quadrature Phase Shift Keying

SIC Soft Interference Cancellation

SNR Signal-to-Noise Ratio

SPA Sum-Product Algorithm

STBC Space-Time Block Coding

STC Space-Time Coding

RA Repeat-Accumulate

SCLDGM Serially-Concatenated Low Density Generator Matrix

SISO Single-Input Single-Output

UCL Unconstrained Capacity Limit

VND Variable Node Decoder

References

[1] C. E. Shannon, "A mathematical theory of communication (part I)," *Bell System Technical Journal*, vol. 27, pp. 379–423, 1948.

[2] R. W. Hamming, "Error detecting and error correcting codes," *Bell System Technical J.*, vol. 29, p. 147, Apr. 1950.

[3] M. J. E. Golay, "Notes on digital coding," *Proc. I. R. E.*, vol. 37, p. 657, 1949.

[4] D. E. Muller, "Application of boolean algebra to switching circuit design and to error detection," Department of Computer Science, University of Illinois at Urbana-Champaign, Urbana, Illinois, Tech. Rep. 138, 1954.

[5] I. Reed, "A class of multiple-error-correcting codes and the decoding scheme," *Information Theory, Transactions of the IRE Professional Group on*, vol. 4, no. 4, pp. 38–49, September 1954.

[6] R. Silverman and M. Balser, "Coding for constant-data-rate systems," *Information Theory, Transactions of the IRE Professional Group on*, vol. 4, no. 4, pp. 50–63, September 1954.

[7] P. Elias, "Coding for noisy channels," in *IRE Conv. Rec.*, ser. pt. 4, 1955, pp. 37–46.

[8] A. Viterbi, "Error bounds for convolutional codes and an asymptotically optimal decoding algorithm," *IEEE Trans. Information Theory*, vol. IT-13, pp. 260–269, Apr. 1967.

[9] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Transactions on Information Theory*, vol. 20, no. 2, pp. 284–287, Mar. 1974.

[10] G. D. Forney, Jr., *Concatenated Codes*. Cambridge, MA, USA: M.I.T. Press, 1966.

[11] R. McEliece and L. Swanson, *Reed-Solomon Codes and their Applications*. S. Wicker and V. Bhargava, editors, IEEE Press, New York, 1994.

[12] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near shannon limit error-correcting coding and decoding: Turbo-codes," in *Proc. ICC 93*, May 1993, pp. 1064–1070.

[13] D. J. C. MacKay and R. M. Neal, "Good codes based on very sparse matrices," *C. Boyd Cryptography and Coding: 5th IAM Conference. Lecture Notes in Computer Science*, no. 1025, pp. 100–111, 1995.

[14] R. G. Gallager, *Low Density Parity Check Codes*. Research Monograph 21. MIT Press, Cambridge, Mass., 1963.

[15] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 1988.

[16] *HT PHY Specification*, Enhanced Wireless Consortium Std. 802.11n, Dec. 2005. [Online]. Available: www.enhancedwirelessconsortium.org

[17] E. Perahia, "IEEE 802.11n development: History, process, and technology," *Communications Magazine, IEEE*, vol. 46, no. 7, pp. 48–55, July 2008.

[18] *IEEE Standard for Local and Metropolitan Area Networks. Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems. Amendment 2 and Corrigendum 1*, IEEE Std. 802.16e, Feb. 2006.

[19] A. Morello and V. Mignone, "DVB-S2: The second generation standard for satellite broadband services," *Proc. IEEE*, vol. 94, no. 1, pp. 210–227, Jan. 2006.

[20] F. Kienle, T. Brack, and N. Wehn, "A synthesizable IP core for DVB-S2 LDPC code decoding," Oct. 2007.

[21] D. J. C. MacKay, "Good error-correcting codes based on very sparse matrices," *IEEE Trans. Inform. Theory*, vol. 45, no. 2, pp. 399–431, 1999.

[22] J. Garcia-Frias and W. Zhong, "Approaching Shannon performance by iterative decoding of linear codes with Low-Density Generator Matrix," *IEEE Commun. Lett.*, vol. 7, no. 6, pp. 266–268, June 2003.

[23] Verdu, "Fifty years of shannon theory," *IEEE Transactions on Information Theory*, vol. 44, 1998.

[24] G. D. Forney and D. J. Costello, "Channel coding: The road to channel capacity," *Proceedings of the IEEE*, vol. 95, no. 6, pp. 1150–1177, June 2007.

[25] H. Jin, A. Khandekar, and R. J. McEliece, "Irregular repeat-accumulate codes," in *Proc. 2nd Int. Symp. Turbo Codes & Related Topics*, Brest, France, Sept. 2000, pp. 1–8.

[26] T. J. Richardson and R. L. Urbanke, "The capacity of Low-Density Parity-Check codes under message-passing decoding," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 599–618, 2001.

[27] T. J. Richardson, M. A. Shokrollahi, and R. L. Urbanke, "Design of capacity-approaching Irregular Low-Density parity-check codes," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 619–637, 2001.

[28] S. Y. Chung, T. J. Richardson, and R. Urbanke, "Analysis of sum-product decoding of Low-Density Parity-Check codes using a Gaussian approximation," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 657–670, Feb. 2001.

[29] S. ten Brink, "Convergence of iterative decoding," *IEE Electronics Letters*, vol. 35, no. 13, pp. 1117–1118, June 1999.

[30] S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollara, "A soft-input soft-output APP module for iterative decoding of concatenated codes," *IEEE Commun. Lett.*, vol. 1, no. 1, pp. 22–24, Jan. 1997.

[31] C. Douillard, M. Jézéquel, C. Berrou, A. Picart, P. Didier, and A. Glavieux, "Iterative correction of intersymbol interference: Turbo equalization," *European Transactions on Telecommunications and Related Technologies*, vol. 6, no. 5, pp. 507–511, Sept. – Oct. 1995.

[32] A. Picart, P. Didier, and A. Glavieux, "Turbo-detection: A new approach to combat channel frequency selectivity," in *ICC (3)*, 1997, pp. 1498–1502.

[33] T. J. Richardson and R. L. Urbanke, "Efficient encoding of Low-Density Parity-Check codes," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 638–656, 2001.

[34] F. R. Kschischang, B. J. Frey, and H. A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 498–519, 2001.

[35] N. Wiberg, H.-A. Loeliger, and R. Kötter, "Codes and iterative decoding on general graphs," *European Transactions on Telecommunications and Related Technologies*, vol. 6, no. 5, pp. 513–525, Sept. – Oct. 1995.

[36] R. M. Tanner, "A recursive approach to low complexity codes," *IEEE Transactions on Information Theory*, vol. 27, no. 5, pp. 533–547, 1981.

[37] J. Pearl, "Reverend Bayes on inference engines: A distributed hierarchical approach," in *AAAI*, 1982, pp. 133–136.

[38] S. ten Brink, "Code doping for triggering iterative decoding convergence," *Information Theory, 2001. Proceedings. 2001 IEEE International Symposium on*, pp. 235–, 2001.

[39] D. Divsalar, H. Jin, and R. J. McEliece, "Coding theorems for 'turbo-like' codes," in *Proc. of 36th Allerton Conf. on Communication, Control, and Computing*, Allerton, Illinois, Sept. 1998, pp. 201–210.

[40] S. ten Brink and G. Kramer, "Design of Repeat-Accumulate codes for iterative detection and decoding," *IEEE Trans. Signal Processing*, vol. 51, no. 11, pp. 2764–2772, Nov. 2003.

[41] S. ten Brink, G. Kramer, and A. Ashikhmin, "Design of Low-Density Parity-Check codes for modulation and detection," *IEEE Trans. Commun.*, vol. 52, no. 4, pp. 670–678, Apr. 2004.

[42] S. ten Brink, "Convergence behavior of iteratively decoded parallel concatenated codes," *IEEE Trans. Commun.*, vol. 49, no. 10, pp. 1727–1737, Oct. 2001.

[43] A. E. Ashikhmin, G. Kramer, and S. ten Brink, "Extrinsic information transfer functions: model and erasure channel properties," *IEEE Transactions on Information Theory*, vol. 50, no. 11, pp. 2657–2673, 2004.

[44] J. F. Cheng and R. J. McEliece, "Some high-rate near capacity codecs for the Gaussian channel," in *Proc.* 34^{th} *Allerton Conf. on Communications, Control and Computing*, Oct. 1996.

[45] W.-Y. Weng, A. Ramamoorthy, and R. D. Wesel, "Lowering the error floors of irregular high-rate ldpc codes by graph conditioning," *Vehicular Technology Conference, 2004. VTC2004-Fall. 2004 IEEE 60th*, vol. 4, pp. 2549–2553 Vol. 4, Sept. 2004.

[46] J. Garcia-Frias and W. Zhong, "Approaching Shannon performance by iterative decoding of linear codes with low-density generator matrix," *IEEE Commun. Lett.*, vol. 7, no. 6, pp. 266–268, June 2003.

[47] H. Zhong and T. Zhang, "Block-LDPC: a practical LDPC coding system design approach," *IEEE Trans. Circuits Syst. I*, vol. 52, no. 4, pp. 766–775, Apr. 2005.

[48] W. Zhong, H. Lou, and J. Garcia-Frias, "LDGM codes for joint source-channel coding of correlated sources," in *ICIP (1)*, 2003, pp. 593–596.

[49] E. Zehavi, "8-PSK trellis codes for a Rayleigh channel," *IEEE Trans. Commun.*, vol. 40, pp. 873–884, May 1992.

[50] G. Caire, G. Taricco, and E. Biglieri, "Bit-interleaved coded modulation," in *ICC (3)*, 1997, pp. 1463–1467.

[51] G. Ungerboeck, "Channel coding with multilevel/phase signals," *IEEE Trans. Inform. Theory*, vol. IT-28, pp. 56–67, Jan. 1982.

[52] D. Tse and P. Viswanath, *Fundamentals of wireless communication.* New York, NY, USA: Cambridge University Press, 2005.

[53] M. González-López, L. Castedo, and J. Garcia-Frias, "Bit-interleaved coded modulation using Low-Density Generator-Matrix codes," in *Proc. of Sixth Baiona Workshop on Signal Processing in Communications*, Baiona, Pontevedra, Spain, Sept. 2003.

[54] İ. E. Telatar, "Capacity of multi-antenna Gaussian channels," Bell Labs, Lucent Technologies, Tech. Rep., Oct. 1995, published in *European Transactions on Telecommunications*, Vol. 10, No. 6, pp. 585-595, Nov/Dec 1999.

[55] G. J. Foschini, "Layered space-time arquitechture for wireless communications in a fading environment when using multi-element antennas," *Bell Labs Technical Journal*, vol. 1, no. 2, pp. 41–59, Autumn 1996.

[56] A. M. Tonello, "Space-Time Bit-Interleaved Coded Modulation with an iterative decoding strategy," in *Proc. IEEE VTC Fall 2000*, Boston, Sept. 2000.

[57] V. Tarokh, N. Seshadri, and A. R. Calderbank, "Space-time codes for high data rate wireless communication: performance criterion and code construction," *IEEE Trans. Inform. Theory*, vol. 44, no. 2, pp. 744–765, Mar. 1998.

[58] J. Grimm and M. P. Fitz, "Further results on space-time coding for rayleigh fading," Oct. 18 1998.

[59] S. M. Alamouti, "A simple transmit diversity technique for wireless communications," *IEEE J. Select. Areas Commun.*, vol. 16, pp. 1451–1458, Oct. 1998.

[60] V. Tarokh, H. Jafarkani, and A. R. Calderbank, "Space-time block codes from orthogonal designs," *IEEE Trans. Inform. Theory*, vol. 45, pp. 1456–1467, July 1999.

[61] V. Tarokh, H. Jafarkhani, and A. R. Calderbank, "Space-time block codes from orthogonal designs," *IEEE Transactions on Information Theory*, vol. 45, no. 5, pp. 1456–1467, 1999.

[62] H. Chen, A. Haimovich, and J. Garcia-Frias, "Applications of a linear precoder to multilevel turbo space-time coding," *Signal Processing Advances in Wireless Communications, 2005 IEEE 6th Workshop on*, pp. 991–995, June 2005.

[63] X. Wang and H. V. Poor, "Iterative (turbo) soft interference cancellation and decoding for coded CDMA," *IEEE Trans. Commun.*, vol. 47, no. 7, pp. 1046–1061, July 1999.

[64] B. Lu, G. Yue, and X. Wang, "Performance analysis and design optimization of LDPC-coded MIMO OFDM systems," *IEEE Trans. Signal Processing*, vol. 52, no. 2, pp. 348–361, Feb. 2004.

[65] B. M. Hochwald and S. ten Brink, "Achieving near-capacity on a multiple-antenna channel," *IEEE Trans. Commun.*, vol. 51, no. 3, pp. 389–399, Mar. 2003.

[66] U. Fincke and M. Post, "Improved methods for calculating vectors of short length in a lattice, including a complexity analysis," *Math. Comp.*, vol. 44, pp. 463–471, Apr. 1985.

[67] W. H. Wow, "Universal lattice decoding: principle and recent advances," *Wirel. Commun. and Mob. Comput.*, vol. 3, pp. 553–559, Aug. 2003.

[68] M. O. Damen, H. E. Gamal, S. Member, and S. Member, "On maximum-likelihood detection and the search for the closest lattice point," *IEEE Trans. Inform. Theory*, vol. 49, pp. 2389–2402, 2003.

[69] H. Vikalo, B. Hassibi, and T. Kailath, "Iterative decoding for MIMO channels via modified sphere decoding," *IEEE Trans. Wireless Commun.*, vol. 3, no. 6, pp. 2299–2311, Nov. 2004.

[70] J.-C. Belfiore, G. Reyaka, and E. Viterbo, "The Golden code: A 2×2 full-rate space-time code with nonvanishing determinants," *IEEE Trans. Inform. Theory*, vol. 51, no. 4, pp. 1432–1436, Apr. 2005.

[71] B. Hassibi and B. M. Hochwald, "High-rate codes that are linear in space and time," *IEEE Trans. Inform. Theory*, vol. 48, no. 7, pp. 1804–1824, July 2002.

[72] E. G. Larsson and P. Stoica, *Space-Time Block Coding for Wireless Communications*. Cambridge University Press, 2003.

[73] X.-B. Liang, "Orthogonal designs with maximal rates," *IEEE Transactions on Information Theory*, vol. 49, no. 10, pp. 2468–2503, 2003.

[74] S. Sandhu and A. Paulraj, "Space-time block codes: A capacity perspective," *IEEE Commun. Lett.*, vol. 4, no. 12, pp. 384–386, Dec. 2000.

[75] J. G. D. Forney and L. F. Wei, "Multidimensional constellations - Part I: Introduction, figures of merit, and generalized cross constellations," *IEEE J. Select. Areas Commun.*, vol. 7, no. 6, pp. 877–892, Aug. 1989.

[76] O. Bretscher, *Linear Algebra with Applications (3rd Edition)*. Prentice Hall, July 2004.

[77] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Roy. Stat. Soc.*, no. 39, pp. 1–38, 1977.

[78] Z. Ding and Y. Li, "On channel identification based on second-order cyclic spectra," *Signal Processing, IEEE Transactions on*, vol. 42, no. 5, pp. 1260–1264, May 1994.

[79] G. B. Giannakis, "Filterbanks for blind channel identification and equalization," Jan. 22 1997.

[80] L. Ping, W. K. Leung, and K. Y. Wu, "Low-rate turbo-hadamard codes," *IEEE Transactions on Information Theory*, vol. 49, no. 12, pp. 3213–3224, 2003.

[81] Leung, Yue, Ping, and Wang, "Concatenated zigzag hadamard codes," *IEEE Transactions on Information Theory*, vol. 52, 2006.

[82] G. Yue, L. Ping, and X. Wang, "Generalized Low-Density Parity-Check codes based on Hadamard constraints," *IEEE Trans. Inform. Theory*, vol. 53, no. 3, pp. 1058–1079, Mar. 2007.

[83] N. Souto, J. C. Silva, F. Cercas, A. Correia, and A. Rodrigues, "Low rate convolutional and turbo codes based on non-linear cyclic codes," *Wireless Communications and Mobile Computing*, vol. 7, no. 1, pp. 23–34, 2007.

[84] K. Li, G. Yue, X. Wang, and L. Ping, "Low-rate repeat-zigzag-hadamard codes," *IEEE Transactions on Information Theory*, vol. 54, no. 2, pp. 531–543, 2008.

[85] A. Shokrollahi, "Raptor codes," *IEEE Transactions on Information Theory*, vol. 52, no. 6, pp. 2551–2567, 2006.

[86] R. Palanki, R. Palanki, J. S. Yedidia, and J. S. Yedidia, "Rateless codes on noisy channels," in *in Proc. Int. Symp. Inform. Theory*, 2004, p. 37.

[87] J. Byers, M. Luby, M. Mitzenmacher, and A. Rege, "Digital fountain approach to reliable distribution of bulk data," International Computer Science Institute, Berkeley, CA, Tech. Rep. TR-98-005, Feb. 1998.

[88] K. Price and R. Storn, "Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces," *J. Global Optimiz.*, vol. 11, pp. 341–359, 1997.

[89] T. S. Rappaport, *Wireless Communications Principles and Practice*. Prentice Hall, 1996.

[90] A. J. Viterbi, *CDMA: Principles of Spread Spectrum Communication*. Addison-Wesley, 1995.

[91] K. Y. W. Li Ping, Lihai Liu and W. K. Leung, "Interleave division multiple access (IDMA) communication systems," in *Proc. 3rd International Symposium on Turbo Codes & Related Topics*, Brest, France, 2003, pp. 173–180.

[92] J. Barros and S. D. Servetto, "On the capacity of the reachback channel in wireless sensor networks," in *IEEE Workshop on Multimedia Signal Processing*. IEEE Signal Processing Society, 2002, pp. 408–411.

[93] ——, "Network information flow with correlated sources," *IEEE Trans. Inform. Theory*, vol. 52, pp. 155–170, 2006.

[94] F. Erkip, J. Garcia-Frias, and Z. Xiong, "Signal processing for multiterminal communication systems [from the guest editor]," *Signal Processing Magazine, IEEE*, vol. 24, no. 5, pp. 12–14, Sept. 2007.

[95] S. Lin and P. Yu, "A hybrid ARQ scheme with parity retransmission for error control of satellite channels," *Communications, IEEE Transactions on [legacy, pre - 1988]*, vol. 30, no. 7, pp. 1701–1719, Jul 1982.

[96] J. Hagenauer, "Rate-compatible punctured convolutional codes (RCPC codes) and their applications," *Communications, IEEE Transactions on*, vol. 36, no. 4, pp. 389–400, Apr 1988.

[97] J. Li and K. R. Narayanan, "Rate-Compatible Low Density Parity Check codes for capacity-approaching ARQ schemes in packet data communications," in *IASTED International Conference on Communications, Internet, and Information Technology*, M. H. Hamza, Ed. IASTED/ACTA Press, 2002, pp. 201–206.

[98] J. Ha, J. Kim, and S. W. McLaughlin, "Rate-Compatible puncturing of Low-Density Parity-Check codes," *IEEE Transactions on Information Theory*, vol. 50, no. 11, pp. 2824–2836, 2004.

[99] H. Lou and J. Garcia-Frias, "Rate-compatible Low-Density Generator Matrix codes," *IEEE Transactions on Communications*, vol. 56, no. 3, pp. 321–324, March 2008.

[100] R. Michael, T. D. Sridhara, and T. Fuja, "A class of group-structured LDPC codes," June 16 2001.

[101] M. P. C. Fossorier, "Quasi-Cyclic Low-Density Parity-Check codes from circulant permutation matrices," *IEEE Transactions on Information Theory*, vol. 50, no. 8, pp. 1788–1793, 2004.

[102] R. M. Tanner, D. Sridhara, A. Sridharan, T. E. Fuja, and D. J. C. Jr, "LDPC block and convolutional codes based on circulant matrices," *IEEE Transactions on Information Theory*, vol. 50, no. 12, pp. 2966–2984, 2004.

[103] C. B. Peel, B. M. Hochwald, and A. L. Swindlehurst, "A vector-perturbation technique for near-capacity multiantenna multiuser communication-part I: channel inversion and regularization," *IEEE Transactions on Communications*, vol. 53, no. 1, pp. 195–202, 2005.

[104] R. D. Wesel and J. M. Cioffi, "Achievable rates for tomlinson-harashima precoding," *IEEE Transactions on Information Theory*, vol. 44, no. 2, pp. 824–831, 1998.

[105] A. P. Liavas, "Tomlinson-harashima precoding with partial channel knowledge," *IEEE Transactions on Communications*, vol. 53, no. 1, pp. 5–9, 2005.

[106] C. Jones, E. Vallés, M. Smith, and J. Villasenor, "Approximate-Min* constraint node updating for LDPC code decoding," in *Proc. of IEEE Military Communications Conference*, Oct. 2003, pp. 157–162.