Performance optimization
for
capacity-approaching
channel coding schemes

Author

Hadi SAWAYA

Contents

Introduction **1**
0.1 Historical background 1
0.2 Objectives and methodology 2
0.3 Major contributions 2
0.4 Summary 3

1 Performance of optimal codes **5**
1.1 Introduction 5
1.2 Channel models and channel capacity 6
1.2.1 Channel models 6
a.) Discrete memoryless channels 6
b.) Discrete-input, continuous-output channel 8
c.) Continuous-input, continuous-output channel 8
1.2.2 Channel capacity 8
1.3 General random coding theorem 10
1.3.1 System model and notations 10
1.3.2 Maximum likelihood decoding 10
1.3.3 Upper bounds on error probability 11
a.) Union bound and Bhattacharyya bound 11
b.) Gallager's bound 13
1.3.4 The general random coding theorem 14
1.3.5 Properties of the reliability function $E(R)$ 18
1.4 Performance of optimal codes over the BSC channel 19
1.5 Channel capacity computation 22
1.5.1 Calculation of the capacity of the continuous-input AWGN channel 22
1.5.2 Calculation of the capacity of the binary symmetric channel 23
1.5.3 Calculation of the capacity of the discrete-input AWGN channel 24
1.6 Performance of optimal spherical codes over the AWGN channel 26
1.6.1 System model and notations 26
1.6.2 Calculation of the lower bound 27
1.6.3 Calculation of the upper bound 29
1.6.4 Performance of optimal spherical codes 29
1.6.5 Comparison with the performance of optimal codes transmitted over a Binary Symmetric Channel 31
1.7 Performance of lattice codes 33

1.7.1 Lower bound . . . 34
1.7.2 Upper bound . . . 35
1.7.3 Performance of optimal lattice codes as a function of SNR . . 36
1.7.4 Numerical results . . . 37
1.8 Conclusion . . . 39

2 Turbo codes and irregular turbo codes **41**
2.1 Introduction . . . 41
2.2 Principles and basic results related to turbo codes . . . 42
2.2.1 Turbo encoder . . . 42
2.2.2 Iterative decoding of turbo codes . . . 44
a) SISO decoding of a convolutional constituent code . . . 44
b) Iterative turbo decoding . . . 45
2.2.3 Basic results and design criteria . . . 46
2.2.4 Numerical results . . . 47
2.3 Symbol-based iterative decoding . . . 50
2.3.1 Construction . . . 50
2.3.2 APP evaluation . . . 50
2.3.3 Numerical results . . . 51
2.4 Irregular turbo codes . . . 53
2.4.1 Irregular turbo encoding . . . 53
Example . . . 55
Remark . . . 55
2.4.2 Decoding irregular turbo codes . . . 55
2.4.3 Selecting the degree profile . . . 57
2.4.4 Performance . . . 58
2.5 Conclusion . . . 61

3 Asymptotic performance of turbo codes **63**
3.1 Introduction . . . 63
3.2 System model . . . 64
3.3 Isotropy property . . . 65
3.4 Partial APP and log-ratios evaluation . . . 67
3.5 Log-ration density propagation . . . 68
a) Subcode node . . . 69
b) Symbol node . . . 70
c) Density propagation in the case of an irregular turbo code . 72
3.6 The transfer Function Method . . . 73
3.6.1 Joint convergence of total and partial LLR . . . 74
3.6.2 Gaussian approximation . . . 75
a) Transfer function of a subcode node . . . 76
b) Transfer function of a symbol node . . . 77
3.6.3 Transfer function of a regular turbo code . . . 77
3.6.4 Transfer function of an irregular turbo code . . . 79
3.7 Numerical results . . . 81
3.8 Conclusions . . . 83

4 **Multilevel coded modulations** **85**
4.1 Introduction . . . 85
4.2 System model . . . 86
4.3 Capacity of multilevel coding . . . 90
4.4 Capacity regions . . . 92
4.5 Rate design rules . . . 93
4.5.1 Balanced distance rule . . . 93
4.5.2 Capacity rule . . . 94
4.5.3 Coding exponent rule . . . 95
4.6 Labeling strategies . . . 96
4.6.1 Block partitioning . . . 97
4.6.2 Gray labeling . . . 99
4.6.3 Parallel independent decoding . . . 99
4.6.4 Performance comparison for finite block length . . . 101
4.7 Dimensionality of the constituent signal constellation . . . 103
4.8 MLC based on asymmetric 2^L-ASK modulation . . . 104
4.9 MLC based on asymmetric 8-PSK modulation . . . 108
4.10 Simulation results . . . 111
4.11 Conclusion . . . 112

5 **MLC in a multiple-antenna system** **115**
5.1 Introduction . . . 115
5.2 System model . . . 116
5.3 Capacity of an ergodic Rayleigh channel . . . 116
Remarks . . . 118
5.4 Outage probability of a non-ergodic block fading channel . . . 119
5.5 Multiple antennas examined using MLC . . . 120
5.5.1 Capacity of the equivalent channels . . . 121
5.5.2 Capacity regions and rate design . . . 122
5.5.3 Outage probability . . . 123
5.6 Orthogonal designs . . . 124
5.6.1 Maximal-ratio receiver combining . . . 124
5.6.2 Alamouti scheme . . . 124
a) Alamouti scheme with $n_r = 1$ receive antenna . . . 125
b) Alamouti scheme with $n_r = 2$ or more receive antennas . . . 126
5.6.3 Other orthogonal designs . . . 128
5.7 Bit-interleaved coded modulations for MIMO systems . . . 128
5.7.1 Capacity of a MIMO system with BICM . . . 129
5.7.2 Iterative detection and APP decoding . . . 131
5.7.3 Numerical results . . . 132
5.8 MLC based on a two stage BICM . . . 133
5.9 Conclusion . . . 135

A **Joint convergence of total and partial APP** **141**

List of Figures

1.1 Channel model. 6
1.2 Binary symmetric channel. 7
1.3 Discrete input, AWGN channel. 8
1.4 Block diagram of a basic digital communication system. 10
1.5 Upper bound on the performance of optimal codes transmitted over the BSC channel, $R = \frac{1}{2}$ bit/dimension. 21
1.6 Upper bound on the performance of optimal codes transmitted over the BSC channel, $R = \frac{2}{3}$ bit/dimension. 21
1.7 Continuous input, AWGN channel. 22
1.8 Capacity of the continuous AWGN channel as a function of the average signal-to-noise ratio per bit $\frac{E_b}{N_0}$ in dB. 23
1.9 Capacity of the BSC as a function of the error probability p. 23
1.10 Capacity of a BSC channel as a function of the average signal-to-noise ratio per bit $\frac{E_b}{N_0}$ in dB. 24
1.11 Capacity of the discrete, ASK input, AWGN channel as a function of the average signal-to-noise ratio per bit $\frac{E_b}{N_0}$ in dB. 25
1.12 Geometrical representation in the N-dimensional space. 27
1.13 Lower and upper bound on the performance of optimal spherical codes, $R = 2$ bits/dimension. 30
1.14 Lower and upper bound on the performance of optimal spherical codes, $R = 3$ bits/dimension. 31
1.15 Performance of optimal codes transmitted over the continuous AWGN channel and the BSC channel, $R = \frac{1}{2}$ bit/dimension. 32
1.16 Performance of optimal codes transmitted over the continuous AWGN channel and the BSC channel, $R = \frac{2}{3}$ bit/dimension. 32
1.17 Sphere T, T', spherical shell T^{Δ}. 35
1.18 Lower and upper bounds on the performance of optimal lattice codes, $R = 1$ bit/dimension. 38
1.19 Lower and upper bounds on the performance of optimal lattice codes, $R = 1.5$ bits/dimension. 38

2.1 Encoder of a rate-1/3 parallel turbo code. 43
2.2 RSC constituent code. Octal generators $g = (7, 5)$ *i.e.* $g_1 = [g_{11}\ g_{12}\ g_{13}] = [1\ 1\ 1]$ and $g_2 = [g_{21}\ g_{22}\ g_{23}] = [1\ 0\ 1]$. $L_c = 3$. 43
2.3 SISO decoder. 45
2.4 Interative decoder of a turbo code. 46

2.5 Bit error probability of turbo codes and optimal codes. $R = 1/2$. $N = 2^{17}$. 49
2.6 Bit error probability of turbo codes and optimal codes. $R = 1/2$. $N = 2^8$ and 2^{11}. 49
2.7 Bit error probability of TC with symbol-based iterative decoding. $R = 1/2$. $N = 2^{11}$. 52
2.8 Bit error probability of TC with symbol-based iterative decoding. $R = 1/2$. $N = 2^{17}$. 52
2.9 A similar encoding scheme for turbo codes. 53
2.10 Encoding scheme of an irregular turbo. 54
2.11 Iterative decoder of an Irregular turbo code. 56
2.12 Fraction f_e of degree $d_e = 10$. $E_b/N_0 = 0.4$ dB. 58
2.13 Degree of "elite" bits for a fraction $f_e = 0.1$. $E_b/N_0 = 0.4$ dB. 58
2.14 Bit error probability of regular and irregular turbo codes. $R = 1/2$. $N = 2^{17}$. 59
2.15 Bit error probability of irregular turbo codes with symbol-based iterative decoding compared to LDPC codes. $R = 1/2$. $N = 10^4$ and 10^5. 60
2.16 Bit error probability of the rate-1/3, $K = 8920$, $N = 26760$, CCSDS turbo code compared to the irregular turbo code with the same parameters. 60

3.1 Propagation tree of concatenated codes. 69
3.2 Probability density of $extLR_m$. $Q = 2$. $m = 5,\ 10,\ 20,\ 30,\ 40,\ 50$ iterations. 71
3.3 Probability density of LR_m. $Q = 2$. $m = 5,\ 10,\ 20,\ 30,\ 40,\ 50$ iterations. 72
3.4 Number of iterations needed to achieve $P_{e_m} < 10^{-5}$. Rate-1/2 turbo code constructed using two rate-2/3 RSC codes with octal generators $g = (37, 21, 27)$ 73
3.5 Transfer function Method 74
3.6 Transfer functions H and G^{-1}. Quaternary turbo code constructed using two rate-2/3 RSC codes with octal generators $g = (31, 27, 37)$. $Eb/N0 = 0.65$ dB. 75
3.7 Transfer functions H and G^{-1}. Quaternary turbo code. Octal generators $g = (37, 21, 27)$. 78
3.8 Transfer functions H and G^{-1}. Quaternary irregular turbo code. Octal generators $g = (37, 21, 27)$. 80

4.1 Multilevel encoding scheme. 87
4.2 Ungerboeck's partitioning of an 8-ASK signal constellation. 87
4.3 8-ASK, MLC encoder. 88
4.4 Equivalent channels of an 8-ASK modulation (Ungerboeck's partitioning). 89
4.5 Multistage decoding of an MLC scheme. 89

4.6 Capacity C of the 8-ASK modulation (AWGN channel) and capacities C_1, C_2 and C_3 of the equivalent channels as a function of $\frac{E_s}{N_0}$ in dB. Ungerboeck's partitioning is applied. 91
4.7 Multiple access using MLC. 92
4.8 Asymmetric 8-PSK constellation 93
4.9 Total rate R of the 8-ASK MLC scheme (AWGN channel) and rates R_1, R_2 and R_3 of the three levels, derived from the coding exponent rule, as a function of $\frac{E_s}{N_0}$ in dB. $P_e = 10^{-3}$. Ungerboeck's partitioning is applied. 96
4.10 Distribution of rates R_1, R_2 and R_3 of the three levels of an 8-ASK MLC scheme (AWGN channel), derived from the coding exponent rule, as a function of the block length N. $P_e = 10^{-3}$. Ungerboeck's partitioning is applied. 97
4.11 Block partitioning of an 8-ASK signal constellation. 98
4.12 Multistage decoding of an MLC scheme when block partitioning is applied. 98
4.13 Capacity C of the 8-ASK modulation and capacities C_i, $i = 1, 2, 3$ of the equivalent channels as a function of $\frac{E_s}{N_0}$ in dB. Block partitioning is applied. 99
4.14 Partitioning of an 8-ASK signal constellation: Gray labeling. 100
4.15 Capacity C of the 8-ASK modulation and capacities C_i, $i = 1, 2, 3$ of the equivalent channels as a function of $\frac{E_s}{N_0}$ in dB. Gray labeling is applied. 100
4.16 Parallel independent decoding of an MLC scheme. 101
4.17 Capacity C of the 8-ASK modulation. Total capacity C_{PDL} and capacities C_{PDL_i}, $i = 1, 2, 3$ of the equivalent channels as a function of $\frac{E_s}{N_0}$ in dB, when Gray labeling with parallel independent decoding is applied. 102
4.18 Required $\frac{E_b}{N_0}$ in dB to achieve a word error probability $P_e = 10^{-3}$, as a function of the block length N. 8-ASK modulation. Labeling strategies: Ungerboeck's partitioning (UP), block partitioning (BP), Gray labeling (GR) and Gray labeling + PDL (GR/PDL). 103
4.19 Parameters α_1 and α_2 of the asymmetric 8-ASK modulation as a function of $\frac{E_s}{N_0}$ in dB. 105
4.20 Capacity C of the 8-ASK modulation, capacity C_a of the asymmetric 8-ASK modulation and capacities $C_{a1} = C_{a2} = C_{a3} = C_a/3$ of the equivalent channels as a function of $\frac{E_s}{N_0}$ in dB. 106
4.21 Required $\frac{E_b}{N_0}$ in dB to achieve a word error probability $P_e = 10^{-3}$, as a function of the block length N. 8-ASK modulation. 107
4.22 Asymmetric 8-PSK constellation 108
4.23 Parameter θ in degrees as a function of $\frac{E_s}{N_0}$ in dB. Asymmetric 8-PSK constellation. Block partitioning is applied. 109
4.24 Capacities C and C_a of the symmetric and asymmetric, 8-PSK, capacities C_i and C_{ai}, $i = 1, 2, 3$ of their equivalent channels as a function of $\frac{E_s}{N_0}$ in dB. Block partitioning is applied. 109

4.25 Required $\frac{E_b}{N_0}$ in dB to achieve a word error probability $P_e = 10^{-3}$, as a function of the block length N. 8-PSK modulation. 110
4.26 Bit error probability of an MLC with different decoding strategies. Block partitioned asymmetric 8-ASK. Total rate $R = 1.5$ bits/dimension. $N = 2^{11}$. AWGN channel. 113
4.27 Bit error probability of a 4-ASK MLC scheme. $R = 1.5$ bits/ dimension. AWGN channel. 113
4.28 Bit error probability of an 8-PSK MLC scheme. $R = 1.5$ bits/ dimension. AWGN channel. 114
4.29 Simulation results for the AWGN channel, presented in the bandwidth efficiency plane. 114

5.1 MIMO channel with n_t transmit and n_r receive antennas. 117
5.2 Capacity of a MIMO channel with continuous input as a function of the total signal-to-noise ratio at each receive antenna $\rho = \frac{E_s}{N_0}$ in dB. . 118
5.3 Outage probability of a continuous MIMO channel as a function of the average signal-to-noise ratio at each receive antenna $\rho = \frac{E_s}{N_0}$ in dB. $R = 2$ and $R = 4$ bits per channel use. $n_t = n_r = 2$. 120
5.4 Example of receive constellation with QPSK and $n_t = 2$ transmit antennas. 121
5.5 Total capacity C and capacities C_1 and C_2 of the equivalent channels of the MIMO system with a QPSK modulation and an ergodic Rayleigh fading channel. $n_t = n_r = 2$. 122
5.6 Outage probability of a QPSK. $n_t = n_r = 2$. 123
5.7 Transmitter in a space-time block code based on orthogonal design. . 125
5.8 Capacity of the Alamouti scheme compared to the total capacity of a MIMO channel. $n_t = n_r = 2$. 127
5.9 BICM transmitter. 128
5.10 16-QAM constellation. a $\rightarrow$ better protected bits. b $\rightarrow$ less protected bits. 130
5.11 Capacity of a MIMO channel with BICM. $n_t = n_r = 2$. 131
5.12 BICM iterative decoder. 132
5.13 Two stage BICM transmitter. 133
5.14 Capacity of a MIMO channel with a two stage BICM. 16-QAM constellation. $n_t = n_r = 2$. 134
5.15 Two stage BICM iterative decoder. 135
5.16 Frame error rate of a BICM over a non-ergodic MIMO channel. $n_t = n_r = 2$. 136
5.17 Bit error rate of a BICM over a MIMO ergodic Rayleigh fading channel. $n_t = n_r = 2$. 136

List of Tables

2.1 Gilbert-Varshamov bound on $\frac{d_{min}}{N}$ as a function of the code rate R. . 48

3.1 Minimal E_b/N_0 in dB, achievable by an iterative decoder. Rate-1/2 turbo code. 81

3.2 Minimal E_b/N_0 in dB, achievable by a symbol-based iterative decoder. Rate-1/2 turbo code. Rate-2/3 RSC code with octal generators $g = (37, 21, 27)$. 82

3.3 Minimal E_b/N_0 in dB, achievable by a bit-based iterative decoder. Rate-1/2 turbo code. Rate-1/2 RSC code with octal generators $g = (37, 21)$. 82

Introduction

0.1 Historical background

For more than 50 years, channel coding has been widely studied. The purpose was to find digital communications systems that have a capacity and a performance close to the limits found by Shannon for the transmission over the *additive white Gaussian noise* (AWGN) channel [78] [79]. The researchers were mainly interested by binary codes designed for a binary phase shift keying (BPSK) modulation.

In 1993, a coding scheme, known as *turbo codes* was presented by Berrou *et al* [6]. For a bit rate equal to 1/2, a performance within 0.5 dB of the capacity limit was achieved.

Turbo codes can be seen as a sub-class of a more general class of codes known as *compound codes*. A compound code is constructed using simple constituent codes that can be easily decoded using an *iterative* decoder instead of the largely complex *maximum likelihood* decoder.

Turbo codes are not the first compound codes that were proposed in the literature. *Product codes* based on block codes, were introduced by Elias [23] in 1954. In 1966, Forney [30] introduced *concatenated codes* in order to construct large block codes by serially concatenating two or more small codes.

In addition, Gallager [35] introduced *low density parity check* (LDPC) codes in 1963, and presented an iterative probabilistic decoding algorithm. However, for complexity reasons, the work of Gallager was neglected. LDPC codes regained significant attention [56] [64] [55] after the apparition of turbo codes. The performance limit of infinite length LDPC codes under iterative decoding was determined by estimating the probability density function of the decoder output from its input density [64]. This method is known as *density evolution*. Other sub-optimal methods like the *Gaussian approximation* method were established in different manners [86] [24] [16] [74].

Based on these concepts, Chung, Forney, Richardson and Urbanke finally showed that, under iterative decoding, LDPC codes can achieve the capacity of the AWGN channel with a BPSK input [15]. Moreover, for block length $N \simeq 2^{17}$ and for a bit error probability greater than 10^{-5}, LDPC codes that can perform better than the original turbo code were proposed by Richardson, Shokrollahi, and Urbanke [65]. This has been achieved by introducing an irregularity in the parity check matrix of the LDPC code.

The main drawback of LDPC codes is in the relatively high complexity of the encoder. Hence, the implementation of the encoder seems difficult in practice.

0.2 Objectives and methodology

The main objective in the thesis is the construction of capacity-approaching coding schemes for binary modulations with codes rates $R < 1$ bit/dimension and for M-ary modulations with high code rates ($R > 1$ bits/dimension).

For a BPSK modulation transmitted over the AWGN channel, a coding scheme based on *parallel concatenated convolutional codes* (PCCC), known as turbo codes, is used. The performance of turbo codes is improved in order to achieve similar and even better performance than LDPC codes. This is done, first, by implementing *symbol-based* iterative decoding and, secondly by introducing an *irregularity* [32] as for LDPC codes. The combination of these two methods will be of a great interest.

For M-ary modulations with high code rates, *lattice codes*, that can achieve the capacity of the AWGN channel [93] are first considered. The performance of optimal lattice codes are shown to be equal to the performance of optimal spherical codes found by Shannon in 1959 [79]. However, due to complexity, methods for decoding lattice codes with a large code length are still unknown. Therefore, another class of coded modulations, known as *multilevel coded modulations* (MLC), was considered. As for lattices, MLC can achieve capacity. It can be used to represent a wide range of coded modulations and also the constructions B though E of dense lattices [17]. The main objective is to approach the capacity of the AWGN channel using a multilevel coding scheme that has equal capacity at each level. The complexity of the system is largely reduced since only one architecture of encoding and decoding is sufficient. This concept may be interesting for the construction of a multilevel unequal error protection (UEP) with multistage decoding.

0.3 Major contributions

The major contributions of this work appear in the following topics :

- The error probability of optimal lattice codes is derived for the additive white Gaussian noise channel. It is shown that lattice codes can, not only achieve capacity on the AWGN channel, but also achieve the performance of finite length optimal spherical codes. A corresponding paper was published in SITA 99 conference [71].

- Multilevel coded modulations based on *asymmetric* amplitude shift-keying (ASK) and phase shift-keying (PSK) constellations are analyzed. It is shown that, When *block partitioning* is applied to an asymmetric constellation, the capacity of the modulation transmitted over an AWGN channel can be attained using constituent codes with equal signaling rates at each level. Moreover, the asymmetric ASK may achieve a gain in performance and capacity. This can be viewed as a shaping gain introduced by these constellations. The same procedure may be applied to quadrature amplitude modulations (QAM) viewed as a concatenation of two one-dimensional ASK. For finite code length N, performance is derived analytically by evaluating Gallager's coding exponent. Two papers on asymmetric constellations were published in RTST 2000 [72] and ISIT 2001 [73] conferences.

- The problem of reaching the capacity of the AWGN channel with BPSK input is solved using regular and irregular turbo codes. Both bit-based and symbol-based iterative decoding are considered. For finite code length, the performance of the rate-1/2 irregular turbo code are improved using symbol-based iterative decoding. A paper on the asymptotic performance of regular turbo codes with symbol-based iterative decoding was published in WCC 2001 conference [74]. The results on the performance of irregular turbo codes with bit-based and symbol-based iterative decoding are to be published later.

- Finally, a multilevel coded modulation scheme, with a *bit-interleaved coded modulation* at each level, is proposed for the multiple-input multiple-output (MIMO) block fading channel. In the case of a 16-QAM, with 2 transmit and 2 receive antennas, this scheme achieves a significant gain over the Alamouti scheme [1].

0.4 Summary

This thesis is divided into five chapters that are summarized as follows :

The basics of information theory are reviewed in chapter 1. The channel capacity and the *general random coding theorem* are first analyzed. Gallager's calculations of the upper bound on the performance of optimal codes, found from the general random coding theorem, are then described. This leads to a tight upper bound on the performance of finite length optimal codes over the *binary symmetric channel* (BSC). For the continuous-input continuous-output AWGN channel, the geometrical approach, developed by Shannon in 1959 in order to determine a lower bound on the performance of all spherical codes [79], is analyzed. For a code length N greater than 100, this bound is too close to the upper one derived by Shannon in the same paper using random coding techniques. This gives us the performance of optimal spherical codes, *i.e.* best spherical codes, over the continuous AWGN channel. Finally, the main results regarding the capacity of lattice codes are reviewed and the performance of optimal non spherical codes, in particular the performance of optimal lattice codes, is established. This performance is compared to the one obtained in the case of optimal spherical codes.

In chapter 2, the principles and the basic results related to parallel turbo codes and irregular turbo codes are described. Symbol-based iterative decoding is then analyzed. This decoding technique is applied to a parallel turbo code constructed using binary recursive systematic convolutional (RSC) code of rate 2/3. Both regular and irregular turbo codes [32] are considered. For finite code length, the performance of the irregular turbo code on a Gaussian channel with BPSK input is compared to the one obtained for the best irregular LDPC codes found by Urbanke [65].

In chapter 3, the asymptotic performance of turbo and irregular turbo codes on a Gaussian channel with BPSK modulation is studied. Both *binary* and *quaternary* symbol-based iterative decoding are considered. The graphical representation of the code is based on Tanner's bipartite graphs [81]. The search for good code parameters is first done using the *Gaussian-approximation* with *error probability matching* (EPM-GA) [74], then exact thresholds are determined using density evolution [64].

In chapter 4, the main results found in the literature, especially by Huber *et al* [98][45][26][99][27], regarding multilevel coded modulations are reviewed. We mainly focus on the capacity, the rate design rules and the labeling strategies of MLC. Then, asymmetric one-dimensional ASK and two-dimensional PSK constellations, with equal capacity at each level, are developed. For finite code length, the performance of the asymmetric constellations is evaluated using Gallager's coding exponents.

In chapter 5, the principles of multilevel coding are applied to a MIMO flat fading channel. We consider both *ergodic* flat fading Rayleigh channels and *non-ergodic* block fading channels. The channel coefficients between each pair of transmit and receive antenna are assumed to be statistically independent and known by the receiver. Two coding schemes based on *bit-interleaved coded modulations* [12] and multilevel coding are proposed. The performances of these two systems are compared with those obtained for an *orthogonal design* [84] like the Alamouti scheme [1].

Chapter 1

Performance of optimal codes

1.1 Introduction

The *coding theorem*, established by Shannon [78] in 1948, states that for signaling rates R below the channel capacity C, information can be transmitted with an arbitrary low error probability.

In fact, the *word error probability* P_e of the best block code of length N is upper bounded by

$$P_e \leq e^{-NE(R)} \tag{1.1}$$

where $E(R)$ is the Gallager coding exponent [36] [37]. $E(R)$ is positive if the code rate R is less than the capacity C.

Gallager calculated $E(R)$ for different channels like the *Discrete Memoryless Channel* (DMC), the *Binary Symmetric Channel* (BSC) and the *continuous additive white Gaussian noise* (AWGN) channel.The results found by Gallager, using *random coding techniques*, can be used as an average performance estimation for the ensemble of random codes when N is finite and R has a fixed valid value. They allow us to say that there exists a code of length N and signaling rate R which has an error probability that is as small as the ensemble-average error probability. The performance of the best code, *i.e. optimal code*, cannot be determined since no valid lower bound has been specified.

Instead of using random coding techniques, Shannon used a geometrical approach to determine a lower bound [79] on the performance of all spherical codes transmitted over a continuous AWGN channel. Some modifications and improvements of his work have been done recently [22] [51]. Nevertheless, the expression derived by Shannon for the lower bound $Q(\theta)$ is still the simplest one. This function gives the probability of moving outside a cone of half-angle θ. Its asymptotic expression is very accurate: for a code length N greater than 100, it is too close to the exact optimal performance. This is validated numerically by comparing $Q(\theta)$ to the upper bound derived by Shannon in the same paper [79], using random coding techniques.

The general random coding theorem and the upper bounds on the performance of optimal codes cannot be applied to structured codes such as *lattice codes*, because these results are based on random coding techniques. Therefore, several papers in the literature treated lattice codes. The main purpose of these studies is to show that lattice codes can achieve the capacity of the additive white Gaussian noise channel [19] [20] [53] [59] [62] [54]. The proof was recently refined by Urbanke & Rimoldi [93]. These works also included a performance analysis of lattice codes. Furthermore, Tarokh *et al* [82] gave a new lower bound on the error probability of lattice codes. They stated that the asymptotic behavior of the new bound, as the dimension N of the code goes to infinity, coincides with Shannon limit for the Gaussian channel.

In this chapter, we review the results previously mentioned and we specially show that lattice codes can perform as good as optimal spherical codes. We begin in section 1.2 by defining channel models and their capacity. In section 1.3, we study the general random coding theorem. The results of this section are applied in section 1.4 to calculate an upper bound on the performance of optimal codes transmitted over a Binary Symmetric Channel. A computation of the capacity of some usual channels is then derived in section 1.5. In section 1.6, we determine the performance of optimal spherical codes transmitted over the continuous AWGN channel. A comparison with the performance over the BSC channel is made also. We finally show in section 1.7 that optimal lattice codes can perform as good as optimal spherical codes when N is sufficiently large.

1.2 Channel models and channel capacity

A communication channel is a system in which the outputs depend probabilistically on its inputs. It is characterized by a probability transition matrix that determines the conditional distribution of the output alphabet Y given the input alphabet X. For $y \in Y$ and $x \in X$ the conditional probability is denoted by $P(y|x)$ (see figure 1.1).

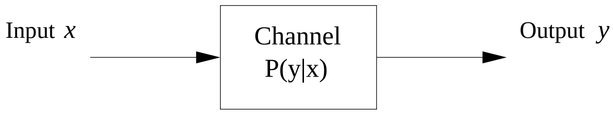


Figure 1.1: Channel model.

Several channel models can be defined, depending on whether the input and/or the output are discrete and/or continuous. We first define these models and then introduce the concept of channel capacity.

1.2.1 Channel models

a.) Discrete memoryless channels

A *discrete memoryless channel* (DMC) is characterized by a discrete input alphabet $X = \{a_1, \ldots, a_k, \ldots, a_Q\}$ where Q is the size of the input alphabet, a discrete output alphabet $Y = \{b_1, \ldots, b_j, \ldots, b_J\}$, and a set of conditional probabilities. We denote the given conditional probabilities by $P_{jk} = P(b_j|a_k)$ $j = 1, \ldots, J; k = 1, \ldots, Q$. For $y \in Y$ and $x \in X$ the conditional probability is denoted by $P(y|x)$. Each output letter of the channel depends only on the corresponding input so that for an input sequence of length N, denoted $\mathbf{x} = (x_1, x_2, \ldots, x_N)$, the conditional probability of a corresponding output sequence, denoted $\mathbf{y} = (y_1, y_2, \ldots, y_N)$, may be expressed as[1]

$$P_N(\mathbf{y}|\mathbf{x}) = \prod_{n=1}^{N} P(y_n|x_n) \tag{1.2}$$

This is the memoryless condition of the definition.

In general, the conditional probabilities $P_{jk} = P(b_j|a_k)$ that characterize a DMC can be arranged in the matrix form $\mathbf{P} = [P_{jk}]$. The matrix $\mathbf{P}$ is called the *probability transition matrix* of the channel.

Definition 1.1 *Symmetric channel*
A channel is said to be symmetric *if the rows (respectively the columns) of the channel transition matrix* $\mathbf{P}$ *are the permutations of the other rows (respectively the other columns). A channel is said to be* weakly symmetric *if only every row of the transition matrix* $P(y|x)$ *is a permutation of the other rows, and all the column sums* $\sum_x p(y|x)$ *are equal.*

The most common type of the *symmetric* DMC channel is the *binary symmetric channel* (BSC) where $X = Y = \{0, 1\}$ and the conditional probabilities are of the form (see figure 1.2)

$$P(1|0) = P(0|1) = p$$

$$P(0|0) = P(1|1) = 1 - p$$

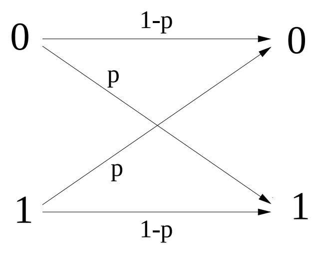


Figure 1.2: Binary symmetric channel.

We can generalize our definition of DMC to channels with alphabets that are not discrete. A common example is the additive white Gaussian noise channel which we define next.

[1] An input sequence of length N is denoted in bold face : $\mathbf{x} = (x_1, x_2, \ldots, x_N)$.

b.) Discrete-input, continuous-output channel

Now, suppose that the input is discrete, *i.e.*, $x \in X = \{a_1, \ldots, a_k, \ldots, a_Q\}$ and the output is continuous, *i.e.*, $y \in Y =]-\infty, +\infty[$. This leads us to define a memoryless channel that is caracterized by the discrete input x and the continuous output y, and the set of conditional probability density functions

$$p(y|a_k), \qquad k = 1, 2, \ldots, Q \tag{1.3}$$

The most important channel of this type is the *additive white Gaussian noise channel* (AWGN), for which $y = x + n$ where n is a zero-mean Gaussian random variable with variance σ_n^2.

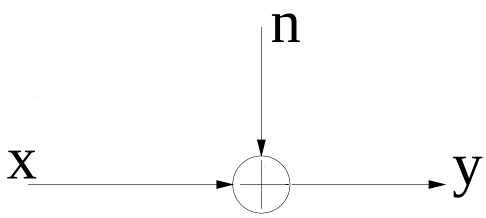


Figure 1.3: Discrete input, AWGN channel.

For a given $x = a_k$, it follows that y is Gaussian with mean a_k and variance σ_n^2. That is,

$$p(y|a_k) = \frac{1}{\sqrt{2\pi}\sigma_n} e^{-\frac{(y-a_k)^2}{2\sigma_n^2}} \tag{1.4}$$

Memoryless again means that for any input sequence $\mathbf{x}$ of length N and any corresponding output sequence $\mathbf{y}$ we have

$$p_N(\mathbf{y}|\mathbf{x}) = \prod_{n=1}^{N} p(y_n|x_n) \tag{1.5}$$

c.) Continuous-input, continuous-output channel

Both the input and the output are continuous, *i.e.*, $x \in X =]-\infty, +\infty[$ and $y \in Y =]-\infty, +\infty[$. The channel is defined by the probability density function $p(y|x)$. A special case of this channel is the *continuous additive white Gaussian noise channel* for which $y = x + n$ where n is a zero-mean Gaussian random variable with variance σ_n^2.

1.2.2 Channel capacity

The *channel capacity* C is defined as the maximum of the average mutual information $I(X;Y)$ where the maximization is over all possible input probability distributions. That is,

$$C = \max_{P(x)} I(X;Y) \tag{1.6}$$

The average mutual information $I(X;Y)$ is the difference between the entropy $H(X)$ and the conditional entropy $H(X|Y)$

$$I(X;Y) = H(X) - H(X|Y) \tag{1.7}$$

Depending on the channel model, we have

- **for a discrete memoryless channel:**

$$C = \max_{P(x)} \sum_x \sum_y P(x)P(y|x) \log \left(\frac{P(y|x)}{\sum_{x'} P(x')P(y|x')} \right) \tag{1.8}$$

- **for a discrete-input, continuous-output channel:**

$$C = \max_{P(x)} \sum_x \int_y P(x)p(y|x) \log \left(\frac{p(y|x)}{\sum_{x'} P(x')p(y|x')} \right) dy \tag{1.9}$$

- **for a continuous-input, continuous-output channel:**

$$C = \max_{p(x)} \int_x \int_y p(x)p(y|x) \log \left(\frac{p(y|x)}{\int_{x'} p(x')p(y|x')dx'} \right) dxdy \tag{1.10}$$

Properties 1.1 *:*

1. $C \geq 0$
2. $C \leq log(Q)$
3. $C \leq log(J)$
4. $I(X;Y)$ *is a continuous function of* $p(x)$*.*
5. $I(X;Y)$ *is a concave function of* $p(x)$

Since $I(X;Y)$ is a concave function over a closed convex set, a local maximum is a global maximum. The maximum can be found using Lagrange multipliers or using nonlinear optimization techniques like gradient search [18].The maximization is done under the constraints that

$P(x) \geq 0$ and $\sum_{x \in X} P(x) = 1$, *i.e.*, $p(x) \geq 0$ and $\int_{-\infty}^{+\infty} p(x)dx = 1$.

The unit of the channel capacity depends on the base of the logarithm. When the logarithm is base 2 the unit is bits per channel use, also known as bits per dimension. For the natural logarithm (base e) the unit is nats per dimension.

The major significance of the channel capacity formulas given above is that they serve as upper limits on the transmission rate for reliable communication over a noisy channel. The fundamental role that the channel capacity plays is given by the *channel coding theorem* due to Shannon [78].

Theorem 1.1 *Channel coding theorem*
There exist channel codes that make it possible to achieve reliable communication, with as small an error probability as desired, if the transmission rate $R \leq C$, where C is the channel capacity. If $R > C$, it is not possible to make the error probability approaches zero.

Before calculating the capacities of the channels defined previously, we are going to study and evalate the reliability function known as the *Gallager exponent*. These calculations prove the channel coding theorem, determine the capacity graphically and determine a tied upper bound on the performance of optimal codes transmitted over a BSC channel.

1.3 General random coding theorem

1.3.1 System model and notations

Figure 1.4 illustrates the block diagram of a basic digital communication system.

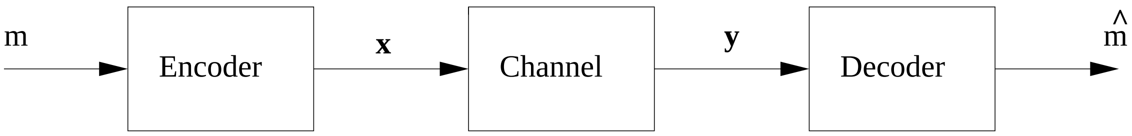


Figure 1.4: Block diagram of a basic digital communication system.

The channel encoder takes a message m of K Q-ary information symbols and maps it into a unique N symbols sequence $\mathbf{x} = (x_1, x_2, \ldots, x_N)$ called a codeword. For a specific value of m, $\mathbf{x}$ is denoted $\mathbf{x_m}$. The total number of codewords is $M = Q^K$, the signaling rate[2] is $R = \frac{\log M}{N}$. The channel is assumed to be a discrete memoryless one. Let X_N be the set of all sequences of length N that can be transmitted over the given channel and Y_N be the set of all sequences of length N that can be received. Both X_N and Y_N are finite sets. Let $P_N(\mathbf{y}|\mathbf{x})$ (for $\mathbf{y} = (y_1, y_2, \ldots, y_N) \in Y_N$ and $\mathbf{x} \in X_N$) be the conditional probability of receiving sequence $\mathbf{y}$ given that $\mathbf{x}$ was transmitted. The probability input distribution is denoted $Q_N(\mathbf{x})$.

1.3.2 Maximum likelihood decoding

The goal of the decoder is to form a mapping from the vector $\mathbf{y}$ to a decision $\hat{m}$ on the message transmitted. The most reasonable, as well as the most convenient,

[2] In this chapter, the unit of R is the same as for the capacity C, iti s in bits per dimension when the logarithm is base 2, it is in nats per dimension for the natural logarithm.

criterion for this decision is to minimize the error probability. This leads to a criterion based on selecting the message corresponding to the maximum of the set of probabilities $\{P(m|\mathbf{y})\}$. This decision criterion is called the *maximum a posteriori probability* (MAP) criterion:

$$\hat{m} = m \quad if \quad P(m|\mathbf{y}) \geq P(m'|\mathbf{y}) \quad \forall\ m' \neq m \tag{1.11}$$

If m satisfies inequality (1.11) but equality holds for one or more values of m', we may achieve the same error probability by choosing any of these m' as the decision. .

Using Bayes rule, the posterior probabilities may be expressed as

$$P(m|\mathbf{y}) = \frac{P(m).P(\mathbf{y}|m)}{P(\mathbf{y})} \tag{1.12}$$

where $P(\mathbf{y}|m)$ is the conditional probability distribution of the observed vector given m, and $P(m)$ is the *a priori probability* of the message m being transmitted.

Some simplification occurs in the MAP criterion when the M messages have equal a priori probabilities, *i.e.*, $P(m) = 1/M$ for all M. Furthermore, we note that the denominator in (1.12) is independent from the transmitted message. Consequently, the decision rule based on finding the message that maximizes $P(m|\mathbf{y})$ is equivalent to finding the signal that maximizes $P(\mathbf{y}|m)$. The decision criterion based on the maximum of $P(\mathbf{y}|m)$ over the M messages is called the *maximum likelihood* (ML) *criterion*. We observe that a detector based on the MAP criterion and one that is based on the ML criterion make the same decisions as long as the a priori probabilities $P(m)$ are all equal.

In the following we assume equally probable messages and thus maximum likelihood decoding is optimum.

1.3.3 Upper bounds on error probability

Having established the optimum decoder to minimize error probability, we now wish to determine its performance as a function of the signal set.

a.) Union bound and Bhattacharyya bound

Let Y_m be the set of all sequences $\mathbf{y}$ decoded as m and Y_m^c be the complement of Y_m. The probability of decoding error when $\mathbf{x_m}$ is transmitted is equal to

$$P_{em} = P(\mathbf{y} \in Y_m^c|\mathbf{x_m}) = \sum_{\mathbf{y}\in Y_m^c} P_N(\mathbf{y}|\mathbf{x_m}) \tag{1.13}$$

The complement of the decision region can be written as

$$\begin{aligned} Y_m^c &= \Big\{\mathbf{y} : \ln P_N(\mathbf{y}|\mathbf{x_{m'}}) \geq \ln P_N(\mathbf{y}|\mathbf{x_m}) \quad for\ some\ m' \neq m\Big\} \\ &= \bigcup_{m' \neq m} \Big\{\mathbf{y} : \ln P_N(\mathbf{y}|\mathbf{x_{m'}}) \geq \ln P_N(\mathbf{y}|\mathbf{x_m})\Big\} \end{aligned}$$

$$= \bigcup_{m' \neq m} Y_{mm'} \tag{1.14}$$

where $Y_{mm'}$ is the decision region for $\mathbf{x_{m'}}$ when only two codewords $\mathbf{x_m}$ and $\mathbf{x_{m'}}$ are considered. Using (1.14) in (1.13), we find

$$\begin{aligned} P_{em} &= P\Big\{\mathbf{y} \in \bigcup_{m' \neq m} Y_{mm'} | \mathbf{x_m}\Big\} \\ &\leq \sum_{m' \neq m} P\Big\{\mathbf{y} \in Y_{mm'} | \mathbf{x_m}\Big\} \\ &= \sum_{m' \neq m} P_e(m \to m') \end{aligned} \tag{1.15}$$

where $P_e(m \to m')$ denotes the pairwise error probability when $\mathbf{x_m}$ is sent and $\mathbf{x_{m'}}$ is the only alternative. The bound of (1.15) is called a *union bound*. The inequality becomes an equality only in the case where $M = 2$.

Further calculations can be made by upper bounding $P_e(m \to m')$. In fact,

$$\begin{aligned} P_e(m \to m') &= Prob\Big\{\mathbf{y} \in Y_{mm'} | \mathbf{x_m}\Big\} \\ &= \sum_{\mathbf{y} \in Y_{mm'}} P_N(\mathbf{y}|\mathbf{x_m}) \\ &= \sum_{\mathbf{y}} f(\mathbf{y}).P_N(\mathbf{y}|\mathbf{x_m}) \end{aligned} \tag{1.16}$$

where we define the function $f(\mathbf{y})$ as

$$f(\mathbf{y}) = \begin{cases} 1 & if\ \mathbf{y} \in Y_{mm'} \\ 0 & otherwise \end{cases} \tag{1.17}$$

We can now upper bound $P_e(m \to m')$ by upper bounding the function $f(\mathbf{y})$:

$$f(\mathbf{y}) \leq \sqrt{\frac{P_N(\mathbf{y}|\mathbf{x_{m'}})}{P_N(\mathbf{y}|\mathbf{x_m})}} \tag{1.18}$$

We can establish the validity of this bound by noting that the right-hand side of (1.18) is always non-negative, thereby satisfying the inequality when $f(\mathbf{y}) = 0$. When $f(\mathbf{y}) = 1$ the inequality is also satisfied since $\mathbf{y}$ is inside $Y_{mm'}$ and therefore the numerator is greater than the denominator.

Using equation (1.16) and replacing $f(\mathbf{y})$ by its bound, we obtain

$$P_e(m \to m') \leq \sum_{\mathbf{y}} \sqrt{P_N(\mathbf{y}|\mathbf{x_{m'}}).P_N(\mathbf{y}|\mathbf{x_m})} \tag{1.19}$$

This expression is called the *Bhattacharyya bound*. Combining the union bound (1.15) with the Bhattacharyya bound we obtain a new expression for the union bound

$$P_{em} \leq \sum_{\mathbf{y}} \sum_{m' \neq m} \sqrt{P_N(\mathbf{y}|\mathbf{x_{m'}}).P_N(\mathbf{y}|\mathbf{x_m})} \tag{1.20}$$

The union bound diverges when the signal-to-noise ratio is less than the limit fixed by the *cut-off rate* denoted by R_0, *i.e.*, when $R_0 < R < C$ [97] [63]. In this case, the only valid upper bound is Gallager's bound.

b.) Gallager's bound

When the union bound fails to give useful results, a more refined technique give an improved bound which is tight over a significantly wider range. Returning to the original general expression (1.13), we begin by defining the set $\tilde{\Lambda}_m$

$$\tilde{\Lambda}_m = \left\{ \mathbf{y} : \sum_{m' \neq m} \left[\frac{P_N(\mathbf{y}|\mathbf{x_{m'}})}{P_N(\mathbf{y}|\mathbf{x_m})} \right]^{\lambda} \geq 1 \right\} \qquad for\ \lambda > 0. \tag{1.21}$$

which contains the set Y_m^c. In fact, if $\mathbf{y} \in Y_m^c$ then

$$\frac{P_N(\mathbf{y}|\mathbf{x_{m'}})}{P_N(\mathbf{y}|\mathbf{x_m})} \geq 1 \tag{1.22}$$

since maximum likelihood decoding was applied. Therefore, since λ is positive, raising the fraction to the λ power keeps it greater than or equal to 1. Finally, summing over all $m' \neq m$ will include the m' for which (1.22) holds, in addition to other non negative terms. Hence $\mathbf{y} \in \tilde{\Lambda}_m$ and $Y_m^c \subset \tilde{\Lambda}_m$.

Now, since the terms in (1.13) are always non-negative, by enlarging the domain of summation (replacing Y_m^c by $\tilde{\Lambda}_m$) we obtain the bound

$$P_{em} \leq \sum_{\mathbf{y} \in \tilde{\Lambda}_m} P_N(\mathbf{y}|\mathbf{x_m}) = \sum_{\mathbf{y}} \phi(\mathbf{y}).P_N(\mathbf{y}|\mathbf{x_m}) \tag{1.23}$$

where

$$\phi(\mathbf{y}) = \begin{cases} 1 & if\ \mathbf{y} \in \tilde{\Lambda}_m \\ 0 & otherwise \end{cases} \tag{1.24}$$

Furthermore, we have

$$\phi(\mathbf{y}) \leq \left[\sum_{m' \neq m} \left[\frac{P_N(\mathbf{y}|\mathbf{x_{m'}})}{P_N(\mathbf{y}|\mathbf{x_m})} \right]^{\lambda} \right]^{\rho} \qquad \forall\ \mathbf{y} \in Y_N,\ \forall\ \rho > 0\ and\ \forall\ \lambda > 0 \tag{1.25}$$

Substituting this bound in (1.23) yields to

$$P_{em} \leq \sum_{\mathbf{y}} P_N(\mathbf{y}|\mathbf{x_m})^{1-\lambda\rho} \Big[\sum_{m' \neq m} P_N(\mathbf{y}|\mathbf{x_{m'}})^{\lambda} \Big]^{\rho} \qquad \forall\ \rho > 0\ and\ \lambda > 0 \tag{1.26}$$

Since λ and ρ are arbitrary positive numbers, we may choose $\lambda = \frac{1}{1+\rho}$ and thus obtain

$$P_{em} \leq \sum_{\mathbf{y}} P_N(\mathbf{y}|\mathbf{x_m})^{\frac{1}{1+\rho}} \Big[\sum_{m' \neq m} P_N(\mathbf{y}|\mathbf{x_{m'}})^{\frac{1}{1+\rho}} \Big]^{\rho} \qquad \forall\ \rho > 0 \tag{1.27}$$

This bound, known as *Gallager's bound* [97], is more powerful than the union bound (1.20), since the latter is the special case of this bound obtained by setting $\rho = 1$.

1.3.4 The general random coding theorem

Equation (1.27) yields a bound to P_{em} for a particular set of codewords. Aside from certain special cases, this bound is too complicated to be used if the number of codewords is large. The exit from this impasse was clearly indicated by Shannon [78], who first employed the central technique of information theory now referred to as *random coding*. The basis of this technique is very simple: given that the calculation of the error probabilities for a particular set of M codewords of dimension N is not feasible, consider instead the average error probability over the ensemble of all possible sets of M codewords with dimensionality N. A tight upper bound on this average over the entire ensemble turns to be amazingly simple to calculate. Obviously at least on signal set must have an error probability which is not greater than the ensemble average; hence the ensemble average is an upper bound on the error probability for the *optimal code i.e. best code.*

If $P_{em}(\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_M})$ is the error probability for the mth message with a given signal set, the average error probability for the mth message over the ensemble of all possible signal sets is

$$\bar{P}_{em} = \sum_{\mathbf{x_1}} \sum_{\mathbf{x_2}} \cdots \sum_{\mathbf{x_M}} Q_N(\mathbf{x_1}) Q_N(\mathbf{x_2}) \ldots Q_N(\mathbf{x_M}) P_{em}(\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_M}) \qquad \forall\ m \tag{1.28}$$

Replacing $P_{em}(\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_M})$ by its Gallager upper bound (1.27) and changing the order of the summations, we obtain as the upper bound on the ensemble error probability when message 1 is sent

$$\bar{P}_{e1} \leq \sum_{\mathbf{y}} \sum_{\mathbf{x_1}} Q_N(\mathbf{x_1}).P_N(\mathbf{y}|\mathbf{x_1})^{\frac{1}{1+\rho}}.$$

$$\left\{ \sum_{\mathbf{x_2}} \sum_{\mathbf{x_3}} \cdots \sum_{\mathbf{x_M}} Q_N(\mathbf{x_2}) Q_N(\mathbf{x_3}) \ldots Q_N(\mathbf{x_M}) \left[\sum_{m'=2}^{M} P_N(\mathbf{y}|\mathbf{x_{m'}})^{\frac{1}{1+\rho}} \right]^{\rho} \right\} \tag{1.29}$$

Equation (1.29) is valid for all $\rho > 0$. Note that we considered the case $m = 1$, only to simplify notations. To proceed further, we must restrict the arbitrary parameter ρ to lie in the unit interval $0 \leq \rho \leq 1$. Then, limiting attention to the term in braces (1.29) and defining

$$f_N(\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_M}) = \sum_{m'=2}^{M} P_N(\mathbf{y}|\mathbf{x_{m'}})^{\frac{1}{1+\rho}} \qquad \forall\ \rho > 0 \tag{1.30}$$

f_N is a convex $\bigcap$ function when $0 \leq \rho \leq 1$. Thus we can apply the Jensen inequality defined in Lemma 1.1.

Lemma 1.1 *Jensen inequality*
Let $f(x)$ be a convex $\bigcap$ real-valued function defined on the real line. Let x be a random variable with finite expectation. Then, for all probability density functions $p(x)$ de x:

$$E[f(x)] \leq f(E[x])$$

For convex $\bigcup$ functions, the inequality is reversed.

We have then

$$\sum_{\mathbf{x_2}} \cdots \sum_{\mathbf{x_M}} Q_N(\mathbf{x_2}) \ldots Q_N(\mathbf{x_M}).[f_N(\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_M})]^{\rho} \leq \\ \left[\sum_{\mathbf{x_2}} \cdots \sum_{\mathbf{x_M}} Q_N(\mathbf{x_2}) \ldots Q_N(\mathbf{x_M}).f_N(\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_M})\right]^{\rho} \tag{1.31}$$

Next, using the definition of $f_N(\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_M})$, we can evaluate the right side of (1.31) exactly to be

$$\left[\sum_{\mathbf{x_2}} \cdots \sum_{\mathbf{x_M}} Q_N(\mathbf{x_2}) \ldots Q_N(\mathbf{x_M}).f_N(\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_M})\right]^{\rho}$$

$$\begin{aligned} &= \left[\sum_{\mathbf{x_2}} \cdots \sum_{\mathbf{x_M}} Q_N(\mathbf{x_2}) \ldots Q_N(\mathbf{x_M}). \sum_{m'=2}^{M} P_N(\mathbf{y}|\mathbf{x_{m'}})^{\frac{1}{1+\rho}}\right]^{\rho} \\ &= \left[\sum_{m'=2}^{M} \sum_{\mathbf{x_{m'}}} Q_N(\mathbf{x_{m'}}) P_N(\mathbf{y}|\mathbf{x_{m'}})^{\frac{1}{1+\rho}}\right]^{\rho} \\ &= \left[(M-1) \sum_{\mathbf{x}} Q_N(\mathbf{x}) P_N(\mathbf{y}|\mathbf{x})^{\frac{1}{1+\rho}}\right]^{\rho} \end{aligned} \tag{1.32}$$

where the last step follows from the fact that each vector $\mathbf{x_{m'}}$ is summed over the same space X_N. Consequently, $\bar{P}_{e1}$ is upper bounded by

$$\bar{P}_{e1} \leq (M-1)^\rho \sum_{\mathbf{y}} \sum_{\mathbf{x_1}} Q_N(\mathbf{x_1}) P_N(\mathbf{y}|\mathbf{x_1})^{\frac{1}{1+\rho}} \Big[\sum_{\mathbf{x}} Q_N(\mathbf{x}) P_N(\mathbf{y}|\mathbf{x})^{\frac{1}{1+\rho}} \Big]^\rho \tag{1.33}$$

and

$$\bar{P}_{e1} \leq (M-1)^\rho \sum_{\mathbf{y}} \Big[\sum_{\mathbf{x}} Q_N(\mathbf{x}) P_N(\mathbf{y}|\mathbf{x})^{\frac{1}{1+\rho}} \Big]^{1+\rho} \tag{1.34}$$

Now if, for any $m \neq 1$, we were to interchange the indices 1 and m throughout the above derivation from (1.29) on, we would arrive to the same bound which is consequently independent of m. We obtain finally the following upper bound on the ensemble average error probability when any message m is sent

$$\bar{P}_{em} \leq (M-1)^\rho \sum_{\mathbf{y}} \Big[\sum_{\mathbf{x}} Q_N(\mathbf{x}) P_N(\mathbf{y}|\mathbf{x})^{\frac{1}{1+\rho}} \Big]^{1+\rho} \qquad \forall\ 0 \leq \rho \leq 1;\ m = 1, 2, \ldots, M \tag{1.35}$$

This bound is valid for any discrete (Q-ary) input and discrete or continuous output channel, provided in the latter case we replace the summation over Y_N by an N-dimensional integral and take $P_N(\mathbf{y}|\mathbf{x})$ to be a density function.

We shall now assume that the channel is *memoryless* (DMC channel), *i.e.*,

$$P_N(\mathbf{y}|\mathbf{x}) = \prod_{n=1}^{N} P(y_n|x_n)$$

for all $\mathbf{x} \in X_N$ and $\mathbf{y} \in Y_N$ and all N. Now we restrict the class of ensemble of codes under consideration to those in which each letter of each codeword is chosen independently of all other letters

$$Q_N(\mathbf{x}) = \prod_{n=1}^{N} P(x_n)$$

then, we have for a *memoryless channel*

$$\bar{P}_{em} \leq (M-1)^\rho \sum_{y_1} \cdots \sum_{y_N} \Big[\sum_{x_1} \cdots \sum_{x_N} \prod_{n=1}^{N} P(x_n) P(y_n|x_n)^{\frac{1}{1+\rho}} \Big]^{1+\rho} \tag{1.36}$$

$$\bar{P}_{em} \leq (M-1)^\rho \prod_{n=1}^{N} \sum_{y_n} \Big[\sum_{x_n} P(x_n) P(y_n|x_n)^{\frac{1}{1+\rho}} \Big]^{1+\rho} \tag{1.37}$$

Trivially over-bounding $M-1$ by M and since the term in the product is independent of n, we obtain

$$\bar{P}_{em} \leq M^\rho \Big[\sum_{y} \Big[\sum_{x} P(x) P(y|x)^{\frac{1}{1+\rho}} \Big]^{1+\rho} \Big]^N \tag{1.38}$$

We may express this equation alternatively in terms of the signaling rate per dimension $R = \ln(M)/N$ nats per dimension ($M = e^{NR}$)

$$\bar{P}_{em} \leq \exp(\rho NR).\exp\Big(N \ln(\sum_y [\sum_x P(x)P(y|x)^{\frac{1}{1+\rho}}]^{1+\rho})\Big) \tag{1.39}$$

We can simplify the notation in (1.39) somewhat by observing that X is the set of input letters, which is denoted $a_1, \ldots, a_k \ldots, a_Q$, where Q is the size of the channel input alphabet. Also, Y is the set of output letters, denoted $b_1, \ldots, b_i \ldots, b_J$, where J is the size of the output alphabet. Now let P_{ik} denote the channel transition probability $P(b_i|a_k)$ and let $P(a_k) = P_k$ denote the probability with which letter a_k is chosen in the code ensemble. Substituting this notation in (1.39) we get

$$\bar{P}_{em} \leq \exp\Bigg(-N\Big[-\rho R - \ln \sum_{i=1}^{J} (\sum_{k=1}^{Q} P_k.P_{ik}^{\frac{1}{1+\rho}})^{1+\rho}\Big]\Bigg) \tag{1.40}$$

Since the right-hand side of (1.40) is independent of m, it is a bound on the ensemble probability of decoding error and is independent of the probabilities with which the codewords are used. Since at least one code in the ensemble must have an error probability as small as the average, we have proved the following theorem, known as the *general random coding theorem*:

Theorem 1.2 *General random coding theorem*
Consider a discrete memoryless channel with an input alphabet of Q symbols, $a_1, \ldots, a_Q$, an output alphabet of J symbols, $b_1, \ldots, b_J$, and transition probabilities $P_{ik} = P(b_i|a_k)$. For any code length N, any number of codewords $M = e^{NR}$, and any probability distribution on the use of the codewords, there exists a code for which the probability of decoding error is bounded by

$$P_e \leq \exp\left(-N[-\rho R + E_0(\rho, \mathbf{P})]\right) \tag{1.41}$$

$$E_0(\rho, \mathbf{P}) = -\ln \sum_{i=1}^{J} (\sum_{k=1}^{Q} P_k.P_{ik}^{\frac{1}{1+\rho}})^{1+\rho} \tag{1.42}$$

where $0 \leq \rho \leq 1$, and $\mathbf{P} = (P_1, P_2, \ldots, P_Q)$.

We get the tightest bound on P_e by minimizing over ρ and $\mathbf{P}$. This gives us the trivial corollary:

corollary 1.1
Under the same conditions as theorem 2, there exists a code for which

$$P_e \leq e^{-NE(R)} \tag{1.43}$$

$$E(R) = \max_{\rho, \mathbf{P}} [-\rho R + E_0(\rho, \mathbf{P})] \tag{1.44}$$

The function $E(R)$ is the *reliability function*, known as the *Gallager coding exponent.* Except for small values of R, corollary 1.1 provides the tightest known general bound on error probability for the discrete memoryless channel. Note that Gallager [37] calculated $E(R)$ for a continuous AWGN channel and signaling rate R close to capacity. As stated by Gallager, this exponent rate is the same as the exponent rate derived by Shannon [79]. However Shannon's coefficients are considerably tighter. Shannon's calculations are reviewed in section 1.6.3.

We discuss the properties of $E(R)$ in the next section.

1.3.5 Properties of the reliability function $E(R)$

The maximization of (1.44) over ρ and $\mathbf{P}$ depends on the behavior of the function $E_0(\rho, \mathbf{P})$. Theorem 1.3 describes $E_0(\rho, \mathbf{P})$ as a function of ρ, and theorem 1.4 describes $E(R)$ and proves the channel coding theorem.

Theorem 1.3
Consider a channel with Q inputs, J outputs, and transition probabilities

$$P_{ik}, \ 1 \leq k \leq Q; \ 1 \leq i \leq J$$

Let $\mathbf{P} = (P_1, P_2, \ldots, P_Q)$ be a probability vector on the channel inputs, and assume that the average mutual information

$$I(\mathbf{P}) = \sum_{k=1}^{Q} \sum_{i=1}^{J} P_k P_{ik} \left(\ln \frac{P_{ik}}{\sum_{j=1}^{Q} P_j P_{ij}} \right) \tag{1.45}$$

is nonzero. Then

$$E_0(\rho, \mathbf{P}) = 0 \qquad for\ \rho = 0 \tag{1.46}$$

$$E_0(\rho, \mathbf{P}) > 0 \qquad for\ \rho > 0 \tag{1.47}$$

$$\frac{\partial E_0(\rho, \mathbf{P})}{\partial \rho} > 0 \qquad for\ \rho > 0 \tag{1.48}$$

$$\left. \frac{\partial E_0(\rho, \mathbf{P})}{\partial \rho} \right|_{\rho=0} = I(\mathbf{P}) \tag{1.49}$$

$$\frac{\partial^2 E_0(\rho, \mathbf{P})}{\partial^2 \rho} \leq 0 \tag{1.50}$$

Using this theorem, we can easily perform the maximization of (1.44) over ρ for a given $\mathbf{P}$. Define

$$E(R) = \max_{0 \leq \rho \leq 1} [-\rho R + E_0(\rho, \mathbf{P})] \tag{1.51}$$

Setting the partial derivative of the bracketed part of (1.51) equal to 0, we get

$$R = \frac{\partial E_0(\rho, \mathbf{P})}{\partial \rho} \tag{1.52}$$

From (1.50), if some ρ in the range $0 \leq \rho \leq 1$ satisfies (1.52) then that ρ must maximize (1.51). Furthermore, from (1.50) $\frac{\partial E_0(\rho,\mathbf{P})}{\partial \rho}$ is a non-increasing function of ρ, so that a solution to (1.52) exists if R lies in the range

$$\left.\frac{\partial E_0(\rho, \mathbf{P})}{\partial \rho}\right|_{\rho=1} \leq R \leq I(\mathbf{P}) \tag{1.53}$$

In this range, it is most convenient to use (1.52) to relate $E(R, \mathbf{P})$ and R parametrically as a function of ρ. This gives us

$$E(R, \mathbf{P}) = E_0(\rho, \mathbf{P}) - \rho \frac{\partial E_0(\rho, \mathbf{P})}{\partial \rho} \tag{1.54}$$

$$R = \frac{\partial E_0(\rho, \mathbf{P})}{\partial \rho} \qquad 0 \leq \rho \leq 1 \tag{1.55}$$

For $R < \frac{\partial E_0(\rho,\mathbf{P})}{\partial \rho}|_{\rho=1}$, the parametric equations (1.54) and (1.55) are not valid. In this case, the function $-\rho R + E_0(\rho, \mathbf{P})$ increases with ρ in the range $0 \leq \rho \leq 1$, and therefore the maximum occurs at $\rho = 1$. Thus

$$E(R, \mathbf{P}) = E_0(1, \mathbf{P}) - R \quad for \quad R < \left.\frac{\partial E_0(\rho, \mathbf{P})}{\partial \rho}\right|_{\rho=1} \tag{1.56}$$

These results are used, in the next section, to determine the performance of optimal codes transmitted over a binary symmetric channel.

Theorem 1.4
For every discrete memoryless channel, the function $E(R)$ is positive, continuous, and convex $\bigcup$ for all the rates R in the range $0 \leq R < C$. Thus the error probability bound $P_e \leq \exp(-NE(R))$ is an exponentially decreasing function of the code length N for $0 \leq R < C$.

1.4 Performance of optimal codes over the BSC channel

The transition probabilities of a binary symmetric channel are $P_{12} = P_{21} = p$, and $P_{11} = P_{22} = 1-p$. Clearly, the input probability vector that maximizes $E_0(\rho, \mathbf{P})$ is $P_1 = P_2 = \frac{1}{2}$. For this choice of $\mathbf{P}$,

$$\begin{aligned} E_0(\rho, \mathbf{p}) &= -\ln \sum_{i=1}^{2} \Big(\sum_{k=1}^{2} P_k . P_{ik}^{1/(1+\rho)} \Big)^{1+\rho} \\ &= \rho \ln 2 - (1+\rho) \ln \Big[p^{1/(1+\rho)} + (1-p)^{1/(1+\rho)} \Big] \end{aligned} \tag{1.57}$$

We now differentiate (1.57) and evaluate the parametric expressions for exponent and rate, (1.54) and (1.55). We obtain[3]

$$R = \ln 2 - H(p_\rho) \tag{1.58}$$

$$E(R, \mathbf{P}) = p_\rho \ln \frac{p_\rho}{p} + (1 - p_\rho) \ln \frac{1 - p_\rho}{1 - p} \tag{1.59}$$

where

$$H(p_\rho) = -p_\rho \ln p_\rho - (1 - p_\rho) \ln(1 - p_\rho)$$

$$p_\rho = \frac{p^{1/(1+\rho)}}{p^{1/(1+\rho)} + (1 - p)^{1/(1+\rho)}}$$

These equations are valid for $0 \leq \rho \leq 1$, or for

$$\ln 2 - H \left[\frac{\sqrt{p}}{\sqrt{p} + \sqrt{1 - p}} \right] \leq R \leq C \tag{1.60}$$

For rates below this, we have (1.56), which becomes

$$E(R, \mathbf{P}) = \ln 2 - 2 \ln(\sqrt{p} + \sqrt{1 - p}) - R \tag{1.61}$$

For low rates, the exponent $E(R, \mathbf{P})$ does not yield a tight bound on error probability. The exponent is so large at low rates that previously negligible effects such as assigning the same codeword to two messages suddenly become important. Using expurgation techniques, Gallager [36] determined a tighter reliability function $E(R, \mathbf{P})$:

$$R + \frac{\ln 4}{N} = \ln 2 - H(\delta) \tag{1.62}$$

$$E(R, \mathbf{P}) = \frac{\delta}{2} \ln \frac{1}{4q(1 - q)} \tag{1.63}$$

where the parameter δ is related to ρ by $\frac{\delta}{1-\delta} = [4p(1 - p)]^{1/2\rho}$. Equations (1.62) and (1.63) are valid for $\delta \geq \frac{\sqrt{4p(1-p)}}{1+\sqrt{4p(1-p)}}$. Note that equations (1.62) and (1.63) are not of great interest since for practical signal-to-noise ratios, *i.e.* practical error probabilities, the rate R is not too low and thus, (1.62) and (1.63) cannot be used.

For large N, $\frac{\ln 4}{N}$ goes to zero. Thus, using equation (1.62), δ approaches $\frac{d_{min}}{N}$, where d_{min} is the Gilbert-Varshamov bound [35] on minimum distance of a binary code.

Figures 1.5 and 1.6 show the upper bound on the word error probability of optimal codes transmitted on the BSC, as a function of the average signal-to-noise ratio per bit $\frac{E_b}{N_0}$ in dB.

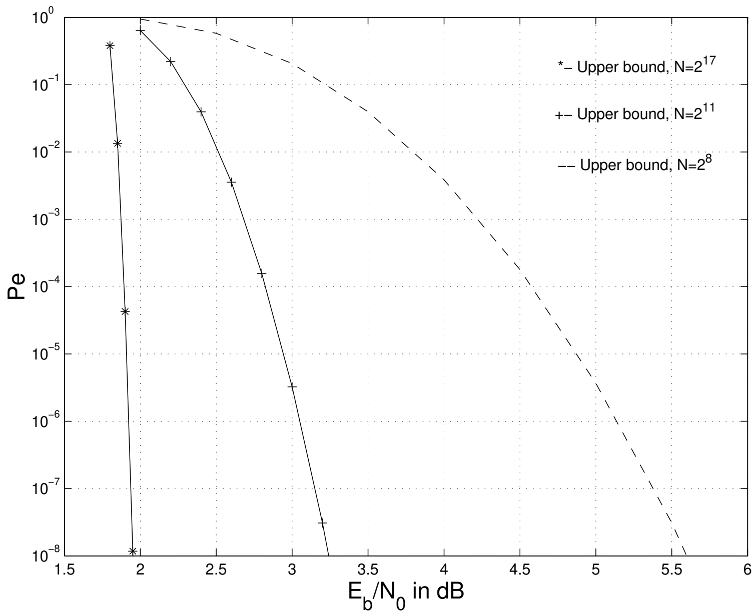


Figure 1.5: Upper bound on the performance of optimal codes transmitted over the BSC channel, $R = \frac{1}{2}$ bit/dimension.

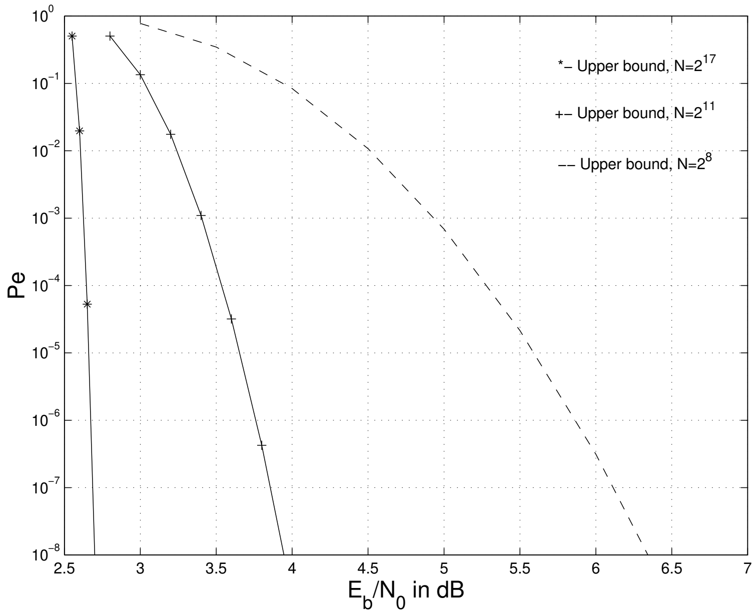


Figure 1.6: Upper bound on the performance of optimal codes transmitted over the BSC channel, $R = \frac{2}{3}$ bit/dimension.

For the practical signal-to-noise ratios of figures 1.5 and 1.6, *i.e.* $P_e > 10^{-8}$, the rate R verifies equation (1.60). Therefore, the upper bound on the word error probability was calculated using equations (1.58) and (1.59). The probability p_ρ is determined using equation (1.58), while the error probability p is equal to $Q(\sqrt{\frac{2RE_b}{N_0}})$ when a BPSK modulation is used.

1.5 Channel capacity computation

In this section, we calculate the capacity of channels like the continuous AWGN channel, the binary symmetric channel and the discrete-input AWGN channel.

1.5.1 Calculation of the capacity of the continuous-input AWGN channel

For a continuous AWGN channel, the output is $y = x + n$ where n is a zero-mean Gaussian random variable with variance σ_n^2.

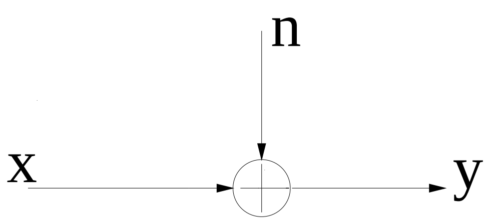


Figure 1.7: Continuous input, AWGN channel.

The maximum of $I(X;Y)$ over the input probability density function $q(x)$ is obtained when the input x is a zero-mean random variable [18]

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma_x} e^{-\frac{x^2}{2\sigma_x^2}}$$

where σ_x^2 is the variance of the input x. Further calculations derived in (A.1) show that the capacity of the continuous AWGN channel is equal to

$$C = \frac{1}{2}\log_2\left(1 + \frac{\sigma_x^2}{\sigma_n^2}\right) \tag{1.64}$$

Now let $P = \sigma_x^2 = 2RE_b$ be the signal power per dimension where R is the signaling rate in bits per dimension and E_b is the average energy per information bit. Let $\sigma_n^2 = N_0$ be the noise variance per dimension. In this case, $y = x + b$ corresponds to the base-band equivalent model of a pass-band transmission $Z(t) = S(t) + N(t)$ where N(t) is a white Gaussian noise of power density $\frac{N_0}{2}$. We obtain

$$C = \frac{1}{2}\log_2\left(1 + \frac{2RE_b}{N_0}\right) \tag{1.65}$$

[3]R is in nats/dimension

To compute C as a function of the average signal-to-noise ratio per bit $\frac{E_b}{N_0}$, we must replace R by C and thus solve

$$C = \frac{1}{2}\log_2\left(1 + \frac{2CE_b}{N_0}\right) \tag{1.66}$$

This is done using the fixed point algorithm. The results are sketched in figure 1.8. The capacity is equal to zero for $\frac{E_b}{N_0}$ equal to -1.6 dB.

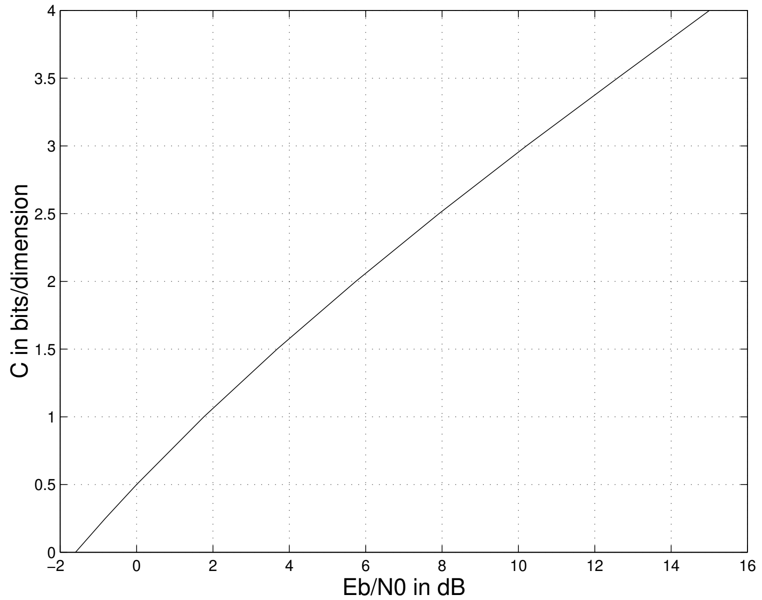


Figure 1.8: Capacity of the continuous AWGN channel as a function of the average signal-to-noise ratio per bit $\frac{E_b}{N_0}$ in dB.

1.5.2 Calculation of the capacity of the binary symmetric channel

For the BSC channel, with transition probabilities $P(1|0) = P(0|1) = p$, the average mutual information is maximized when the input probabilities $P(0) = P(1) = \frac{1}{2}$.

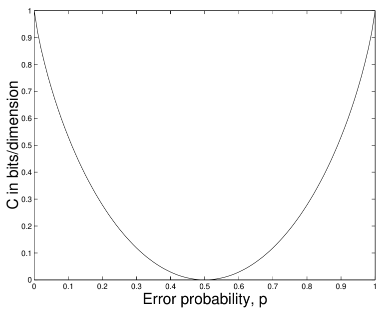


Figure 1.9: Capacity of the BSC as a function of the error probability p.

Thus, the capacity of the BSC is

$$C = p.\log_2 p + (1-p)\log_2(1-p) = 1 - H_2(p) \tag{1.67}$$

where $H_2(p)$ is the binary entropy function.

A plot of C versus p is illustrated in figure 1.9. Note that for $p = 0$, the capacity is 1 bit/dimension. On the other hand, for $p = \frac{1}{2}$, the mutual information between the channel input and output is zero. Hence the channel capacity is zero.

For a BPSK modulation $p = Q(\sqrt{\frac{2RE_b}{N_0}})$, where $\frac{E_b}{N_0}$ is the average signal-to-noise ratio per bit. To calculate C as a function of $\frac{E_b}{N_0}$ we choose $R = C$ and then use the fixed point algorithm. Results are shown in figure 1.10

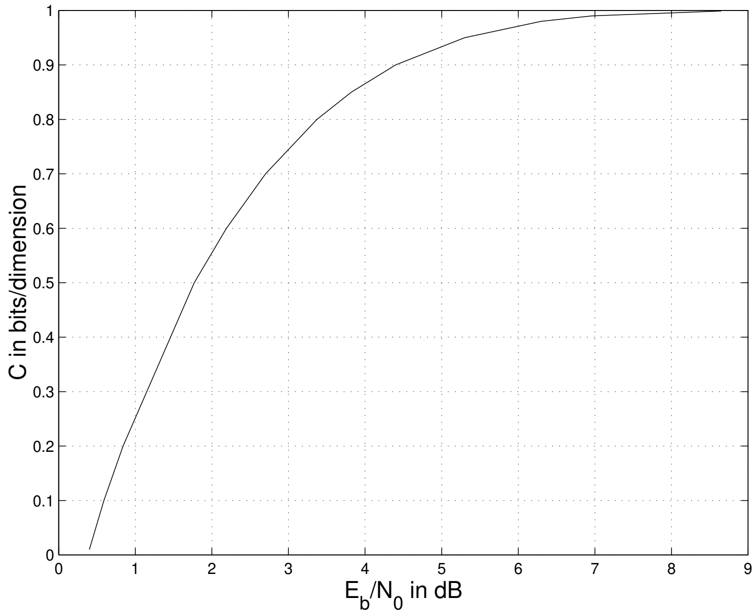


Figure 1.10: Capacity of a BSC channel as a function of the average signal-to-noise ratio per bit $\frac{E_b}{N_0}$ in dB.

1.5.3 Calculation of the capacity of the discrete-input AWGN channel

For a Q-ary discrete-input continuous-output channel, the capacity is equal to (see equation (1.9))

$$C = \max_{P(x)} \sum_x \int_y P(x)p(y|x)\log\left(\frac{p(y|x)}{\sum_{x'} P(x')p(y|x')}\right) dy$$

The maximum is achieved by a uniform distribution on the input [18]

$$P(x) = \frac{1}{Q} \qquad \forall\, x \in X = \{a_1, a_2, \dots, a_Q\}$$

We obtain

$$C = \frac{1}{Q}\sum_{k=1}^{Q}\int_y p(y|a_k)\log\Big(\frac{p(y|a_k)}{\frac{1}{Q}\sum_{k'=1}^{Q} p(y|a_{k'})}\Big)dy \tag{1.68}$$

where

$$p(y|a_k) = \frac{1}{\sqrt{2\pi}\sigma_n} e^{-\frac{(y-a_k)^2}{2\sigma_n^2}} \tag{1.69}$$

We can compute C numerically using equation (1.68) and replacing $p(y|a_k)$ by its expression. For example, for Q-ary ASK modulations, the capacity C calculated as a function of $\frac{E_b}{N_0}$, is shown in figure 1.11. Note that in this case, $X = \Big\{\pm 1; \pm 3; \dots; \pm(Q-1)\Big\}$, $E_b = \frac{Q^2-1}{6}$ and $\sigma_n^2 = N_0$.

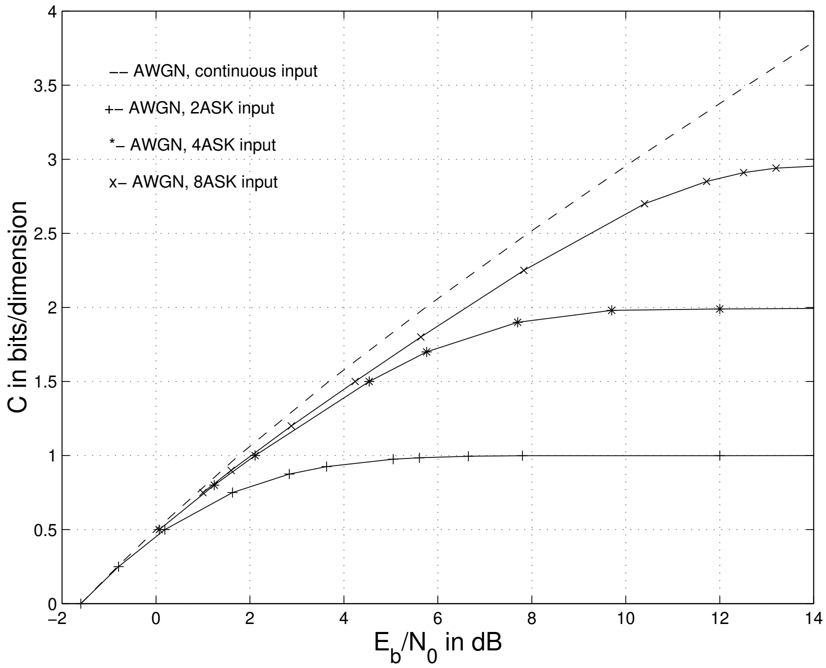


Figure 1.11: Capacity of the discrete, ASK input, AWGN channel as a function of the average signal-to-noise ratio per bit $\frac{E_b}{N_0}$ in dB.

We revien in the next section the calculation of the performance of an optimal spherical code derived bu Shannon [79] for the AWGN channel.

1.6 Performance of optimal spherical codes over the AWGN channel

For continuous Gaussian channels, Shannon developed lower and upper bounds on the error probability of optimal spherical codes [79]. For such codes, each codeword lies on the surface of a sphere of radius $\sqrt{NP}$ where P is the signal power per dimension. The upper bound was found using random coding techniques. On the other hand, Shannon used a geometrical approach to determine the lower bound [79]. Some modifications and improvements of his work have been done recently [22] [51]. Nevertheless, the expression derived by Shannon for the lower bound $Q(\theta)$ is still the simplest one. This function gives the probability of moving outside a cone of half-angle θ. Its asymptotic expression is very accurate for a code length N greater than 100. This has been validated numerically by comparing $Q(\theta)$ to the upper bound derived by Shannon in the same paper [79].

In this section, we review these calculations and we show numerically that lower and upper bounds are close together for signaling rates near channel capacity, *i.e.*, practical signal-to-noise ratios.

1.6.1 System model and notations

Let us make a geometrical representation of the code set in the N-dimensional real space $\mathbb{R}^N$. We assume that the $M = 2^{NR}$ codewords are equally probable, and uniformly distributed on the N-dimensional sphere $Sph(o, N)$ centered at the origin o, with radius $\sqrt{NP}$. The signal power per dimension is $P = 2RE_b$, R is the signaling rate in bits per dimension and E_b is the average energy per information bit.

The codeword $\mathbf{x} = (x_1, \ldots, x_k, \ldots, x_N)$ is transmitted over an additive white Gaussian noise (AWGN) channel

$$y_k = x_k + n_k$$

where n_k is a real Gaussian noise with zero mean and variance N_0.

The optimal performance of this code transmitted over the AWGN channel is bounded as follows [79]

$$Q(\theta_1) \leq P_{eopt}(N, R, P) \leq Q(\theta_1) - \int_0^{\theta_1} \frac{\Omega(\theta)}{\Omega(\theta_1)} dQ(\theta) \tag{1.70}$$

where $Q(\theta)$ is the probability for a point x at distance $\sqrt{NP}$ from origin, being moved outside a circular cone of half-angle θ with vertex at the origin o and axis ox.

The function $\Omega(\theta)$ is the solid angle of a cone of half-angle θ, or equivalently the area of a unit N-sphere cut out by the cone. The angle θ_1 is chosen such that the solid angle $\Omega(\theta_1)$ is equal to $\frac{1}{2^{NR}}\Omega_N(\pi)$ where $\Omega_N(\pi)$ is equal to the surface of the unit N-dimensional sphere.

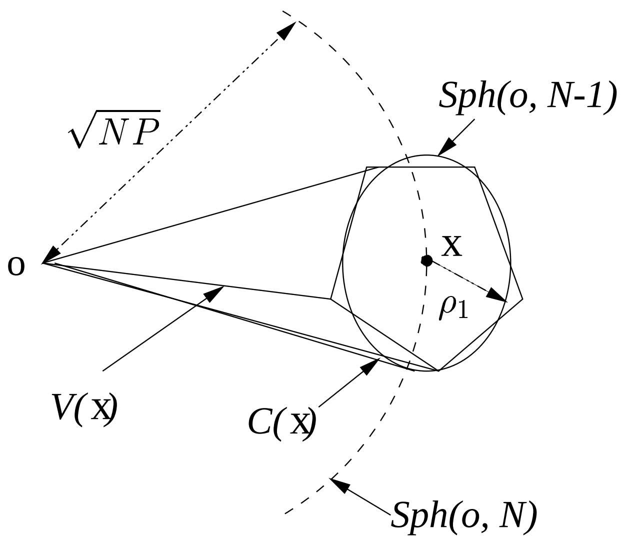


Figure 1.12: Geometrical representation in the N-dimensional space.

1.6.2 Calculation of the lower bound

Figure 1.12 shows the codeword $\mathbf{x}$ surrounded by a circular cone $C(\mathbf{x})$ delimited by an $(N-1)$-dimensional sphere (circle) $Sph(o, N-1)$ and a pyramid $V(\mathbf{x})$ drawn in an hexagonal form. $V(\mathbf{x})$ is the Voronoi region associated to the codeword $\mathbf{x}$, *i.e.* the decision region of $\mathbf{x}$ when maximum likelihood (ML) decoding is applied. The circle $Sph(o, N-1)$ is the intersection of the sphere $Sph(o, N)$ and a hyper-plane denoted by H. The radius $\rho_1 = \rho_1(\theta_1)$ of the circle is chosen such as the solid angle $\Omega(\theta_1)$ of the pyramid $V(\mathbf{x})$ is equal to that of the cone $C(\mathbf{x})$ delimited by $Sph(o, N-1)$. Note that

$$\Omega(\theta_1) = \frac{1}{2^{NR}}\Omega_N(\pi) \tag{1.71}$$

The codewords are supposed to be uniformly distributed and equally probable. By symmetry, the average word error probability P_{eopt} is equal to the conditional error probability $P_e(\mathbf{x})$ when $\mathbf{x}$ is transmitted. $P_e(\mathbf{x})$ is thus the probability for the received signal to be moved by the Gaussian noise outside the pyramid $V(\mathbf{x})$.

The Gaussian noise probability density p_n is a decreasing function of the Euclidean distance d. It can be easily shown that

$$\int_{C(c)} \frac{\partial p_n}{\partial V} dV \geq \int_{V(c)} \frac{\partial p_n}{\partial V} dV \tag{1.72}$$

Let $Q(\theta_1)$ be the probability for a codeword on the axis of a cone of solid angle $\Omega(\theta_1)$ being moved outside this cone. We have

$$Q(\theta_1) = 1 - \int_{C(c)} \frac{\partial p_n}{\partial V} dV \tag{1.73}$$

We thus obtain the following lower bound for the word error probability

$$Q(\theta_1) \le P_{eopt} \tag{1.74}$$

The relation $R = R(\theta_1)$ and the expression of $Q(\theta)$ have been developed by C.E. Shannon [79]. The solid angle $\Omega(\theta)$ is computed by summing elementary ring surfaces $d\Omega(\phi) = \Omega_{N-1}(\pi) sin^{N-2}\phi d\phi$:

$$\Omega(\theta) = \frac{(N-1)\pi^{\frac{N-1}{2}}}{\Gamma(\frac{N+1}{2})} \int_0^\theta \sin^{N-2}\phi d\phi \tag{1.75}$$

where $\Gamma(\frac{N+1}{2})$ is the Gamma Euler function. Using equation (1.71), we get

$$2^{NR} = \frac{\sqrt{\pi}\Gamma(\frac{N-1}{2})}{\Gamma(\frac{N}{2}) \int_0^{\theta_1} \sin^{N-2}\phi d\phi} \tag{1.76}$$

Asymptotically, when $N >> 1$, equation (1.76) becomes

$$2^{NR} \simeq \frac{\sqrt{2\pi N} \sin\theta_1 . \cos\theta_1}{\sin^N \theta_1} \tag{1.77}$$

The lower bound $Q(\theta_1)$ can be computed from its derivative

$$Q(\theta_1) = \int_{\theta_1}^{\frac{\pi}{2}} -dQ(\phi) + Q(\frac{\pi}{2}) \tag{1.78}$$

Notice that $Q(\frac{\pi}{2}) = \frac{1}{2} erfc(\sqrt{\frac{NP}{2N_0}})$, where $erfc$ is the complementary error function. This term can be neglected since it has a minor influence on $Q(\theta_1)$.

Finally, the function $Q(\theta_1)$ can asymptotically be approximated by

$$Q(\theta_1) = \frac{e^{-NE_L(\theta_1)}}{\sqrt{N\pi}.\sqrt{1+G^2}.\sin\theta_1.(\sqrt{\frac{P}{N_0}}\sin^2\theta_1 - \cos\theta_1)} \tag{1.79}$$

where

$$E_L(\theta_1) = \frac{P}{2N_0} - \frac{1}{2}\sqrt{\frac{P}{N_0}} G.\cos\theta_1 - \ln(G.\sin\theta_1) \tag{1.80}$$

$$G(\theta_1) = \frac{1}{2}\left[\sqrt{\frac{P}{N_0}}\cos\theta_1 + \sqrt{\frac{P}{N_0}\cos^2\theta_1 + 4}\right] \tag{1.81}$$

The cone half-angle θ_1 is computed by solving equation (1.77) for a fixed space dimension N and a fixed rate R. Different approximations of $Q(\theta)$ were investigated. Also some exact calculations were derived [22] [51]. We found numerically that exact formulas [22] [51] and the asymptotic expression of equation (1.79) coincide for $N > 100$. Equations (1.77), (1.79), (1.80) and (1.81) are used latter to evaluate $Q(\theta_1)$ numerically.

1.6.3 Calculation of the upper bound

The upper bound of P_{eopt} was found by Shannon using an argument based on random codes.

Consider the ensemble of codes obtained by placing M points randomly on the surface of a sphere of radius $\sqrt{NP}$. More precisely, each point is placed independently of all others with probability measure proportional to surface area or, equivalently, to solid angle.

The average error probability $\bar{P}_e$ of for all codewords, was calculated by Shannon. He found

$$\bar{P}_e = -\int_{\theta=0}^{\pi} \left\{1 - [1 - \frac{\Omega(\theta)}{\Omega(\pi)}]^{M-1}\right\} dQ(\theta) \tag{1.82}$$

Since this is an average of P_e over all codes, there must exist a particular code in the ensemble with verifies $P_{eopt} \leq \bar{P}_e$. Further calculations derived by Shannon showed that

$$P_{eopt}(N, R, P) \leq Q(\theta_1) - \int_0^{\theta_1} \frac{\Omega(\theta)}{\Omega(\theta_1)} dQ(\theta) \tag{1.83}$$

As for the lower bound, we can asymptotically approximate the upper bound by

$$\left(1 - \frac{\cos\theta_1 - \sqrt{\frac{P}{N_0}}G\sin^2\theta_1}{2\cos\theta_1 - \sqrt{\frac{P}{N_0}}G\sin^2\theta_1}\right) \cdot \frac{e^{-NE_L(\theta_1)}}{\sqrt{N\pi}.\sqrt{1+G^2}.\sin\theta_1.\left(\sqrt{\frac{P}{N_0}}\sin^2\theta_1 - \cos\theta_1\right)} \tag{1.84}$$

This result is valid for $\theta_1 < \theta_c$, $0 \leq \theta_c \leq \pi/2$, where θ_c is the unique root of $2\cos\theta - \sqrt{\frac{P}{N_0}}G\sin^2\theta$, for any fixed $\frac{P}{N_0} > 0$. The rate that corresponds to the critical value $\theta = \theta_c$ is denoted by R_c.

For $\theta_1 > \theta_c$, *i.e.* for rates $R < R_c$ the previous upper bound does not hold. The asymptotic expression in this case is

$$\frac{1}{\cos\theta_c \sin^3\theta_c \sqrt{N\pi[1+G(\theta_c)^2]\frac{\partial^2 E_L(\theta_c)}{\partial\theta_c^2}}} e^{-N[E_L(\theta_c)-R]} \tag{1.85}$$

For practical signal-to-noise ratios, the rate R is close to the capacity C ($R_c << R < C$). Thus the asymptotic expression of equation (1.85) is not of great interest.

1.6.4 Performance of optimal spherical codes

For $\theta_1 < \theta_c$, the two asymptotes differ only by the factor

$$\left(1 - \frac{cos\theta_1 - \sqrt{\frac{P}{N_0}}Gsin^2\theta_1}{2cos\theta_1 - \sqrt{\frac{P}{N_0}}Gsin^2\theta_1}\right)$$

which is independent of N. This factor increases as θ_1 increases from the value θ_0, corresponding to channel capacity, to the critical value θ_c, corresponding to the critical rate for which the denominator vanishes. Over this range the factor increases from 1 to ∞. In other words, for large N, P_{eopt} is determined within a factor. Furthermore, the percentage uncertainty due to this factor is smaller at rates closer to channel capacity, approaching zero as the rate approaches capacity.

Slepian [80] checked this result for large $N(> 100)$. We also did the same numerical verification (see figures 1.13 and 1.14) using the approximations of the lower and upper bounds given by Shannon (equations (1.79) and (1.84)).

This implies that for practical average signal-to-noise ratio per bit $\frac{E_b}{N_0}$ and reasonable error probability values $P_e > 10^{-8}$, i.e. rates near channel capacity, we have

$$P_{eopt}(N, R, P) = P_{eopt}\left(N, R, \frac{E_b}{N_0}\right) \simeq Q(\theta_1) \tag{1.86}$$

The result of equation (1.86) is very interesting. The performance of optimal spherical codes are equal to the lower bound $Q(\theta_1)$ which can be evaluated numerically using equations (1.77), (1.79), (1.80) and (1.81).

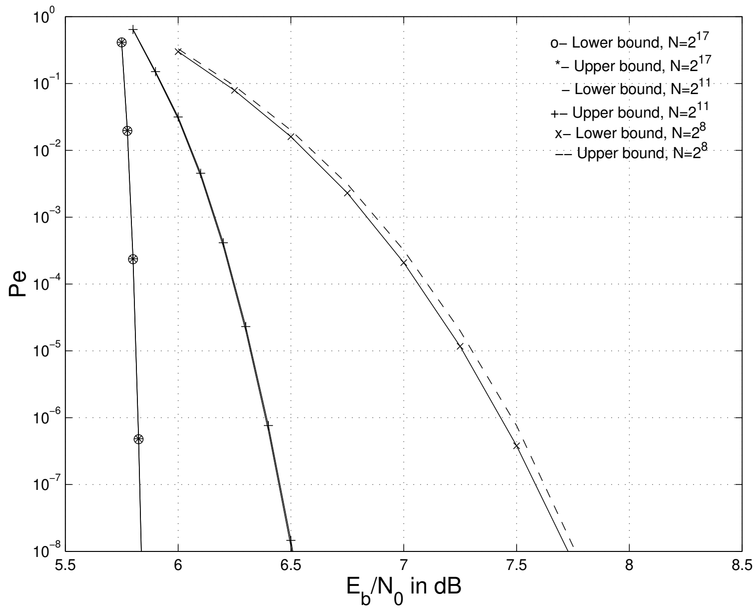


Figure 1.13: Lower and upper bound on the performance of optimal spherical codes, $R = 2$ bits/dimension.

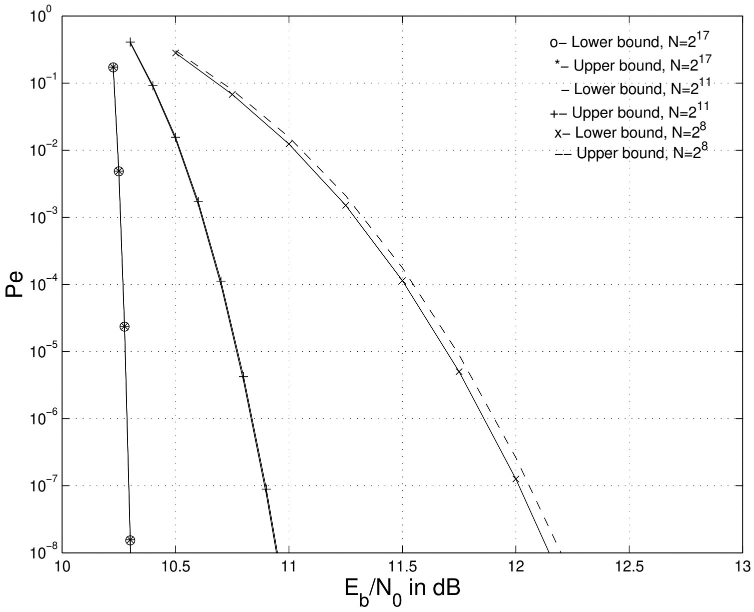


Figure 1.14: Lower and upper bound on the performance of optimal spherical codes, $R = 3$ bits/dimension.

1.6.5 Comparison with the performance of optimal codes transmitted over a Binary Symmetric Channel

Numerical calculations derived in the last section showed that for practical signal-to-noise ratios the upper bound derived by Shannon using random coding techniques is approximately equal to the lower bound and consequently equal to the performance of optimal codes transmitted over the continuous AWGN channel.

The same random coding techniques were also used by Gallager to derive the general random coding theorem and then to calculate an upper bound on the performance of optimal codes transmitted over a Binary Symmetric Channel (see sections 1.3 and 1.4). Therefore, we may also approximate their performance by their upper bound calculated in section 1.4. This approximation is more accurate for large values of code length N.

Figures 1.15 and 1.16 show a comparison between the performance over the AWGN and the BSC channels for code rates equal to 1/2 and 2/3 bits per dimension. For $N = 2^{17}$, and for both code rates, the degradation in the performance of the BSC channel is almost equal to the one we have for the capacity ($\sim 2dB$). This also validates the upper bound as a good approximation of the performance of optimal codes over both AWGN and BSC channels.

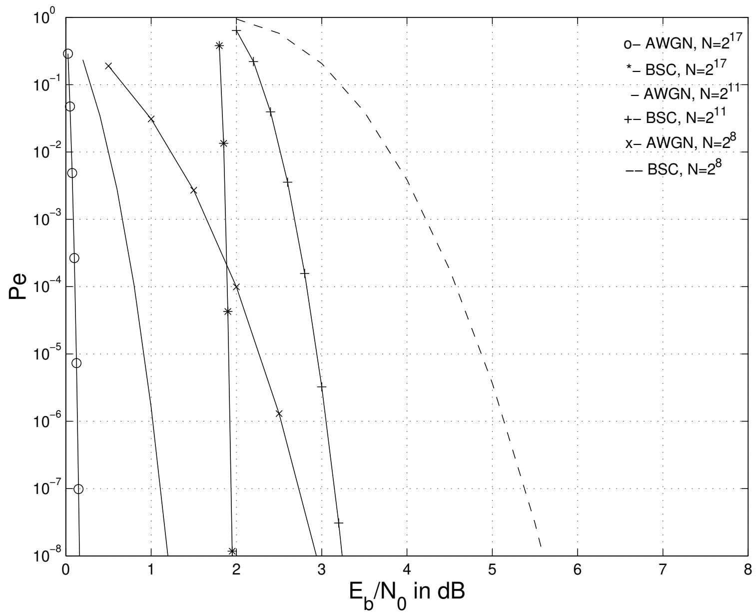


Figure 1.15: Performance of optimal codes transmitted over the continuous AWGN channel and the BSC channel, $R = \frac{1}{2}$ bit/dimension.

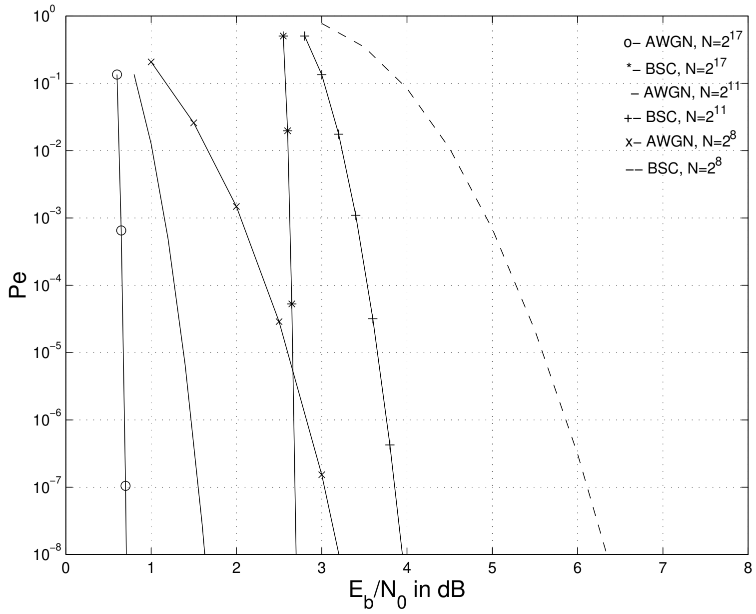


Figure 1.16: Performance of optimal codes transmitted over the continuous AWGN channel and the BSC channel, $R = \frac{2}{3}$ bit/dimension.

1.7 Performance of lattice codes

The proof of the channel coding theorem and the calculations of the upper bound on the performance of optimal codes are based on a *random coding* argument and, hence, codes that achieves capacity and the performance of optimal codes may exhibit little or no structure making them not suitable for practical applications. Thus, it is of great interest to investigate the maximum reliable transmission rates and the best performance achieved by structured codes like *lattice codes.*

Lattice codes consist of the intersection of a lattice Λ (or a translate of a lattice) with a bounded shaping region T, which is typically a ball centered at the origin. To take full advantage of the underlying lattice structure, we would like to neglect the effects of the bounding region T and simply decode to the nearest lattice point, whether or not this point lies in T. Such decoder is referred to as *lattice decoder.* The optimum decoding procedure on the other hand is *minimum distance decoding* which maps the received point into the closest lattice point inside T.

The main prior work on information-theoretic limits of lattice codes are two papers by De Buda [19], [20]. In his first paper [19], De Buda considered spherically shaped lattice codes with lattice decoding and showed that arbitrarily small error probability can be achieved at all rates below $1/2\log_2(P/N)$ bits per dimension, where P and N are the signal power and noise variance, respectively. He also gave an exponential error bound that is only close to Shannon's bound for large signal-to-noise ratios.

In his second paper [20], De Buda seemed to have proved that lattice codes can even achieve the full channel capacity $1/2\log_2(P/N)$, and that there exists lattice codes which satisfy Shannon's upper bound within the factor 4. For technical reasons, he considered thick-shell shaping region T rather than spherical shaping. Moreover, he assumed minimum distance decoding rather than lattice decoding. However, an error in [20] was reported by Linder *et al* [53]. They were able to fix the problem, by replacing de Buda's thick shells with thin shells. Consequently, the corresponding codes lost most of their lattice structure and rather resembled to random spherical codes.

Magalhaẽs and Battail [59] also studied lattice codes and derived error exponents. In fact, they seemed to have proved that, even with lattice decoding, the full capacity $1/2\log_2(1 + P/N)$ is achievable. However, a mistake in the proof (the lattice points inside a $(P + N)$-sphere are tacitly assumed to have average power P) invalidated their argument.

Poltyrev [62] considered unbounded constellations and lattices, for which he gave a Gallager-type exponential random coding bound. He also proved that $1/2\log_2(P/N)$ is achievable.

Finally, Urbanke and Rimoldi [93] were able to prove that lattice codes with a bounding region T equal to a ball achieve the capacity $1/2\log_2(1 + P/N)$ of a continuous AWGN channel when minimum distance decoding is used. (The proof, that $1/2 \log_2(P/N)$ is the capacity of lattice codes under lattice decoding is still missing).

Note that all these authors based their lattice results on the following version of the Minkowski-Hlawka theorem [70]. This theorem states that for any Riemann integrable function $f : \mathbb{R}^N \longrightarrow \mathbb{R}$ of bounded support and any positive ϵ, there exists a lattice Λ in $\mathbb{R}^N$ with fundamental volume 1 such that

$$\sum_{x \in \Lambda} f(x) < \int_{\mathbb{R}^N} f(x)dx + \epsilon$$

De Buda stated that, for lattices, the Minkowski-Hlawka theorem can replace the usual random coding arguments. However, all known proofs of the theorem are obtained from averaging over a large, usually infinite, class of lattices. In this sense, the Minkowski-Hlawka theorem is random coding and may be regarded as a pry-Shannon result in information theory. Loeliger [54] used this theorem to determine averaging bounds for lattice codes.

In the next sections, we show that lattice codes can, not only achieve capacity on the AWGN channel but also perform as good as finite length Shannon's optimal spherical codes when the length is sufficiently large.

1.7.1 Lower bound

Let T be the N-dimensional ball of radius $\sqrt{NP_m}$ and volume $V = V_N(\sqrt{NP_m})^N$, where P_m is the peak power per dimension and V_N is the volume of a unit sphere.

Consider a lattice $\Lambda \subset \mathbb{R}^N$ with a fundamental region P_f and a point s inside P_f. The $M = 2^{NR}$ codewords of a lattice code $C(\Lambda, T)$ are the points of the translated lattice $\Lambda + s$ that belong to T [93]

$$C(\Lambda, T) = (\Lambda + s) \cap T \tag{1.87}$$

Let $P^C_{eopt}(N, R, \frac{P_m}{N_0})$ be the word error probability of this code. To each codeword add a further coordinate such that in the $N+1$-dimensional space, the resulting point lies exactly on the $N+1$-dimensional sphere surface. If the first N coordinates of a point have values $x_1, x_2, \ldots, x_N$ with $\sum_{i=1}^N x_i^2 \leq NP_m$, the added coordinate will have the value

$$x_{N+1} = \sqrt{(N+1)P_m - \sum_{i=1}^{N} x_i^2} \tag{1.88}$$

This gives a new spherical code of the type described in section 2, i.e. $M = 2^{NR}$ words of length $N+1$ at signal-to-noise ratio $\frac{P_m}{N_0}$. The error probability of the initial code is at least as great as that of the new one, since the added coordinate can only improve the decoding process

$$P_e^d \leq P^C_{eopt}(N, R, \frac{P_m}{N_0}) \tag{1.89}$$

where P_e^d is the word error probability of the new spherical code of length $N+1$. By comparing the new code to the optimal spherical code of length $N+1$, we derive the following inequality

$$P_{eopt}\left(N+1, \frac{NR}{N+1}, \frac{P_m}{N_0}\right) \leq P_e^d \leq P_{eopt}^C\left(N, R, \frac{P_m}{N_0}\right) \tag{1.90}$$

In large dimensions, $P_{eopt}(N, R, \frac{P_m}{N_0}) \simeq P_{eopt}(N+1, \frac{NR}{N+1}, \frac{P_m}{N_0})$. Thus, we can write when $N \gg 1$

$$P_{eopt}\left(N, R, \frac{P_m}{N_0}\right) \lesssim P_{eopt}^C\left(N, R, \frac{P_m}{N_0}\right) \tag{1.91}$$

1.7.2 Upper bound

Define $P_{eopt}^{ns}(N, R, \frac{P_m}{N_0})$ as the word error probability of the best non-spherical code bounded by the ball T. This code is not necessarily a lattice code. Clearly, since the points lie inside the sphere of radius $\sqrt{N.P_m}$ and not at its surface like for optimal spherical codes, this will relax the condition on the code. So the performance is improved in the latter case and we have

$$P_{eopt}^{ns}\left(N, R, \frac{P_m}{N_0}\right) \leq P_{eopt}\left(N, R, \frac{P_m}{N_0}\right) \tag{1.92}$$

This result cannot be directly applied to lattice codes because we cannot check that the best non-spherical code is a lattice code. As described below, the Urbanke & Rimoldi upper bound [93] on lattice codes shows that for high dimensions, the performance of a lattice code is bounded by that of an optimal spherical code similar to equation (1.92).

In fact, let T' be the N-dimensional sphere of radius $\sqrt{NP'}$ ($P' < P_m$) and volume $V' = V_N(\sqrt{NP'})^N$, as illustrated in figure 1.17.

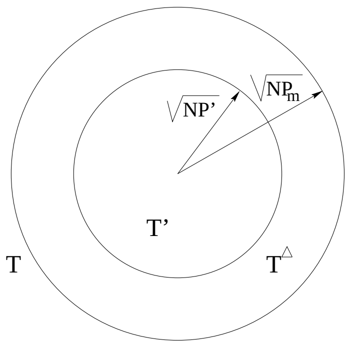


Figure 1.17: Sphere T, T', spherical shell T^{Δ}.

Define $T^{\Delta} = T - T'$ with volume $V^{\Delta} = V - V'$and define the two lattice codes $C'(\Lambda, T') = (\Lambda + s) \cap T'$ and $C^{\Delta}(\Lambda, T^{\Delta}) = (\Lambda + s) \cap T^{\Delta}$, that are sub-codes of $C(\Lambda, T)$. Let $M' = 2^{NR'}$ and $M^{\Delta} = 2^{NR^{\Delta}}$ be the cardinalities of these codes. A suboptimal decoder satisfies the following inequality [93]

$$P_{eopt}^{C}\left(N, R, \frac{P_m}{N_0}\right) \leq \frac{M'}{M} + P_e^{C^\Delta} \tag{1.93}$$

where $P_e^{C^\Delta}$ is the word error probability of the lattice code $C^\Delta(\Lambda, T^\Delta)$. De Buda [20] upper bounded $P_e^{C^\Delta}$ by

$$P_e^{C^\Delta} \leq 4.P_{eopt}\left(N, R^\Delta, \frac{P'}{N_0}\right) \tag{1.94}$$

The above bound has a factor of 4 when compared to the lower bound (1.91) for large values of N. To eliminate this factor, we can apply the upper bound of [93] using our notations:

$$P_{eopt}^{C}\left(N, R, \frac{P_m}{N_0}\right) \leq 4\frac{V'}{V} + P_{eopt}\left(N, R^\Delta, \frac{P'}{N_0}\right) \frac{2.\sqrt{NP_m}(N-1)\pi^{\frac{N-1}{2}}(NP')^{N/2}}{2^{-NR^\Delta}V^\Delta\Gamma(\frac{N+1}{2})} \int_0^{\theta'_1} (\sin x)^{N-2} dx \tag{1.95}$$

It is easy to show that (1.95) simplifies to

$$P_{eopt}^{C}\left(N, R, \frac{P_m}{N_0}\right) \leq P_{eopt}\left(N, R^\Delta, \frac{P'}{N_0}\right) + 4\frac{V'}{V} + 2\sqrt{NP_m}N\frac{V'}{V^\Delta} \tag{1.96}$$

Choose P' too close to P_m. Asymptotically, as $N \longrightarrow +\infty$, $\frac{V'}{V} \longrightarrow 0$, $N^{3/2}\frac{V'}{V^\Delta} \longrightarrow 0$, $R^\Delta \longrightarrow R$. Then

$$P_{eopt}^{C}\left(N, R, \frac{P_m}{N_0}\right) \leq P_{eopt}\left(N, R, \frac{P_m}{N_0}\right) \tag{1.97}$$

This result is the same as the one we had for optimal non spherical codes (equation (1.92)). Combining (1.91) and (1.97), we find

$$P_{eopt}^{C}\left(N, R, \frac{P_m}{N_0}\right) \simeq P_{eopt}\left(N, R, \frac{P_m}{N_0}\right) \qquad (N >> 1) \tag{1.98}$$

Numerical results showed that for finite values of N, the gap between the lower bound (1.90) and the upper bound (1.96) is very small (see figure 1.18). The result expressed in equation (1.98) says that an optimal lattice code performs as good as an optimal spherical code when the space dimension is large enough. As illustrated in figure 1.18, the signal-to-noise ratio difference is less than 0.25 dB for $N = 2048$.

1.7.3 Performance of optimal lattice codes as a function of SNR

To derive the word error probability as a function of the average signal-to-noise ratio per bit $\frac{E_b}{N_0}$, we have to find the relation between E_b and the peak power per dimension P_m.

- First, we calculate the average energy E_b^{cu} per information bit of a lattice code $C^{cu}(\Lambda, T^{cu})$ where T^{cu} has a cubic shape, $T^{cu} = [-A; +A]^N$, and a volume $Vol(C^{cu}) = (2A)^N = V_N(\sqrt{NP_m})^N$. The cube boundary is given by

$$A^2 = \frac{1}{4}NP_m \sqrt[N/2]{V_N} \tag{1.99}$$

Let $E_p^{cu} = NRE_b^{cu}$ be the average energy per point. E_p^{cu} can be calculated using the continuous approximation [82] [28] valid for large values of N and R

$$\begin{aligned} E_p^{cu} &= \frac{1}{2}\int\int\int_{[-A,+A]^N} ||\mathbf{x}||^2 \frac{d\mathbf{x}}{Vol(C^{cu})} \\ &= \frac{1}{2}\int\int\int_{x\in[-A,+A]^N} (x_1^2 + x_2^2 + \ldots + x_N^2)\frac{dx_1 dx_2 \ldots dx_N}{(2A)^N} \end{aligned}$$

By evaluating N simple integrals we find $E_p^{cu} = \frac{NA^2}{6}$. Thus

$$E_b^{cu} = \frac{E_p^{cu}}{NR} = \frac{A^2}{6R} = \frac{1}{24R} N P_m \sqrt[N/2]{V_N} \tag{1.100}$$

- Secondly, since a spherical N-dimensional constellation has a shaping gain [28] [17] equal to

$$\gamma_s(C) = \frac{\pi \times (N+2)}{12 \times \sqrt[N/2]{\Gamma(N/2+1)}} \tag{1.101}$$

and since $V_N = \frac{\pi^{N/2}}{\Gamma(N/2+1)}$, we get

$$E_b = \frac{E_b^{cu}}{\gamma_s(C)} \simeq \frac{N}{N+2}.\frac{1}{2R}.P_m \simeq \frac{P_m}{2R} \tag{1.102}$$

Equation (1.102) is identical to the one we had for codes with equal power points ($P = 2RE_b$).

We conclude finally that the word error probability of an optimal lattice code transmitted over an additive white Gaussian noise channel with average bit to noise ratio $\frac{E_b}{N_0}$, is asymptotically equal to

$$P_{eopt}^{C}\left(N, R, \frac{E_b}{N_0}\right) \simeq P_{eopt}\left(N, R, \frac{E_b}{N_0}\right) \simeq Q(\theta_1) \tag{1.103}$$

1.7.4 Numerical results

Figures 1.18 and 1.19 show the lower and the upper bounds of optimal lattice codes of rate 1 bit/dimension and 1.5 bits/dimension respectively, as a function of the average signal-to-noise ratio per bit $\frac{E_b}{N_0}$ in dB. The lower bound $Q(\theta)$ is calculated using equations (1.77), (1.79), (1.80) and (1.81). Curves are plotted for $N = 2^8$, $N = 2^{11}$ and $N = 2^{17}$. The small gap between the two bounds shows that for $N \geq 2048$, we can approximate the performance of optimal lattice codes by their lower bound like in equation (1.103).

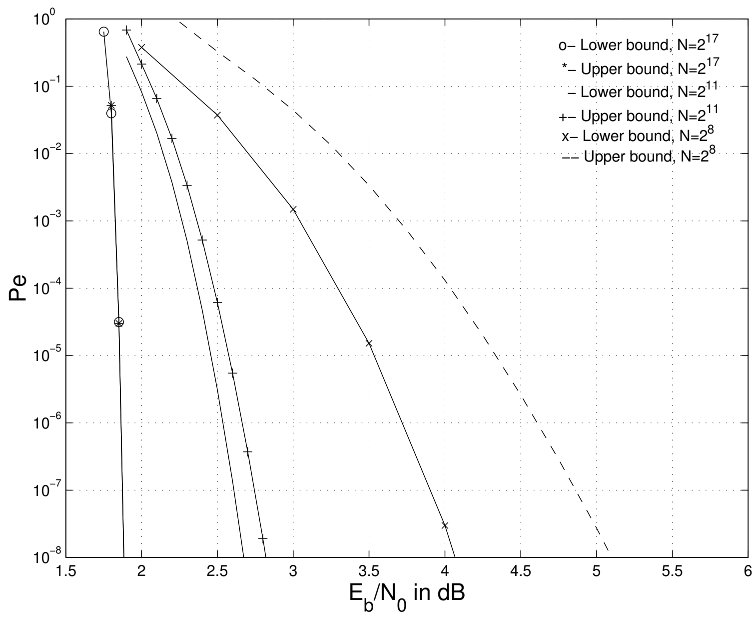


Figure 1.18: Lower and upper bounds on the performance of optimal lattice codes, $R = 1$ bit/dimension.

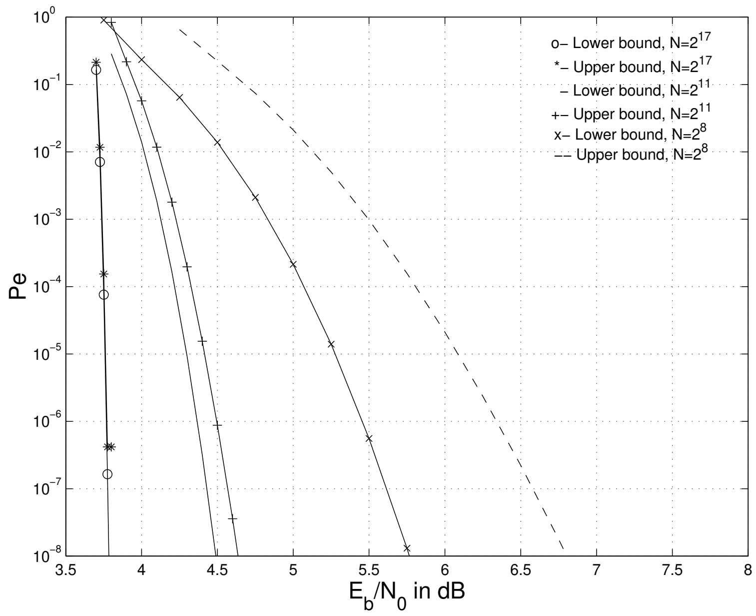


Figure 1.19: Lower and upper bounds on the performance of optimal lattice codes, $R = 1.5$ bits/dimension.

1.8 Conclusion

In this chapter, the main results regarding the capacity and the performance of optimal codes are reviewed for both continuous additive white Gaussian noise channel and binary symmetric channel. These results determine the best achievable performance. Digital communications systems with performance close to these limits must be found. As proved in this chapter, this may be done using structured codes like lattice codes since these codes can not only achieve capacity but also perform as good as optimal spherical codes. The main problem with lattice codes is decoding. The *sphere decoding* technique [17] can be used with lattice codes that have a large code length N. Several work was done to find a sub-optimal decoding technique for large values of N [49] [11]. However, these results are not sufficient to validate lattice codes for the use in a transmission over the AWGN channel.

In the next two chapters, the capacity of the AWGN with BPSK input is approached using turbo codes and irregular turbo codes. For high code rates, chapter 4 describes a class of coded modulations, known as multilevel coded modulations, that can achieve capacity.

Chapter 2

Turbo codes and irregular turbo codes

2.1 Introduction

In 1993, a coding scheme known as *Turbo Codes* (TC) was presented by Berrou, Glavieux and Thitimajshima [6]. Turbo codes can be seen as a sub-class of a more general class of codes known as *compound codes*. A compound code is constructed using simple constituent codes that can be easily decoded. Turbo codes are not the first compound codes that were proposed in the literature.

Product codes based on block codes, were introduced by Elias [23] in 1954. Product codes are based on the concatenation of at least two linear codes. If we consider two codes $C_1(N_1, K_1, d_1)$ and $C_2(N_2, K_2, d_2)$, the product code parameters are the product of the elementary code parameters N_i, K_i, d_i; $i = 1, 2$ that respectively stand for the block length, the information bits length and the minimum Hamming distance of code C_i. The product code is represented by a matrix obtained by encoding K_2 rows of K_1 information bits by code C_1, then encoding the N_1 columns of the matrix by code C_2. Each of the N_2 rows of the coded matrix is a codeword C_1 as each of the N_1 columns is a codeword of C_2. The minimum Hamming distance of the product code is equal to $d = d_1 \times d_2$.

In 1966, Forney [30] introduced *concatenated codes* in order to construct large block codes by serially concatenating two or more small codes.

In 1963, Gallager [35] introduced the *low density parity check* (LDPC) codes and presented an iterative probabilistic decoding algorithm. However, for complexity reasons, the work of Gallager was neglected till the introduction of turbo codes. Since then, LDPC were largely studied [56] [64] [55]. These studies showed that the capacity of the AWGN channel with BPSK input can be reached using LDPC codes and iterative decoding [15]. Moreover, for block length $N \simeq 2^{17}$ and for a bit error probability greater than 10^{-5}, LDPC codes that can perform better than the original turbo code of Berrou *et. al.* [6] were found [65]. This was done by introducing a kind of irregularity in the parity check matrix of the LDPC.

The main disadvantage of LDPC codes is in the complexity of the encoder. Therefore, we focus our study on the *parallel concatenated convolutional codes* (PCCC) known as turbo codes. This scheme consists of two constituent systematic convolutional encoders linked by an interleaver. The information bits at the input of the first encoder are scrambled by the interleaver before entering the second encoder. Performance close to the limits of chapter 1 was found using an iterative decoding technique based on soft-input soft-output (SISO) decoding of the constituent codes.

We improve the performance of turbo codes first by implementing *symbol-based* iterative decoding, and secondly by introducing an *irregularity* [32] as for LDPC codes. The application of symbol-based iterative decoding to an irregular turbo code leads to very interesting results.

The principles and the basic results related to parallel turbo codes are studied in section 2.2. We specifically review the turbo encoder, the iterative turbo decoding and the basic results and design criteria related to parallel turbo codes. Numerical results on the performance of parallel turbo codes are presented and compared to those we have for optimal spherical codes (cf. chapter 1). In section 2.3, we introduce symbol-based iterative decoding. This decoding techniques is applied to a parallel turbo code constructed using binary recursive systematic convolutional (RSC) code of rate 2/3. The performance of this turbo coding scheme is determined numerically by Monte Carlo simulation. In section 2.4, we study irregular turbo codes. Finally, symbol-based iterative decoding of an irregular turbo code is done. The performance of this system is determined and compared to the one we have for the best irregular LDPC codes found by Urbanke [65].

2.2 Principles and basic results related to turbo codes

We focus our study only on *parallel concatenated convolutional codes* known as *turbo codes*. We review in this section the basic principles, results and design criteria found in the literature regarding this concatenation scheme. We begin with the turbo encoder.

2.2.1 Turbo encoder

Figure 2.1 shows the encoder of a rate-1/3 parallel turbo code C obtained from the concatenation of two constituent rate-1/2 recursive systematic convolutional (RSC) codes. The information bits at the input of the first encoder are fed to an *interleaver* before entering the second encoder. The trellis of each constituent code starts and ends at the zero-state. This is done by terminating the trellis using at most $L_c - 1$ transitions, where L_c is the constraint length of the RSC code. Therefore, the turbo code can be viewed as a linear block code of dimension N.

The information bits are transmitted just once. If all the parity bits are transmitted, the total code rate of the turbo code will be equal to 1/3. This is the first turbo coding scheme that was first proposed by Berrou *et. al.*[6].

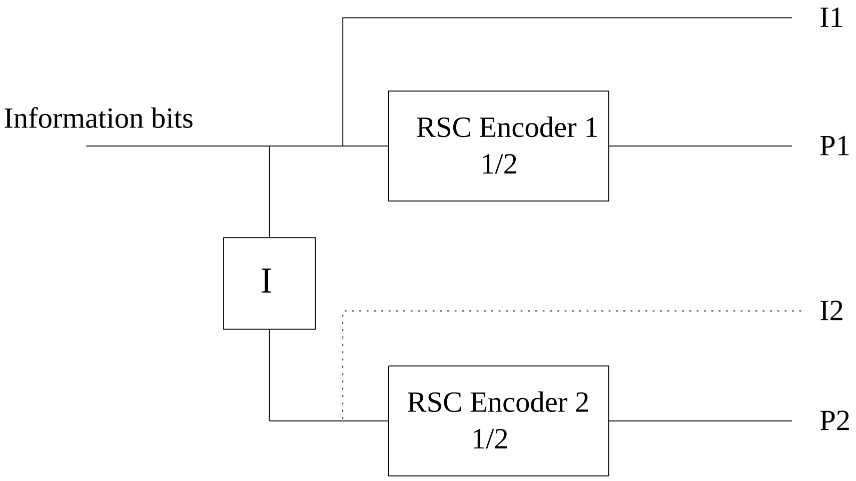


Figure 2.1: Encoder of a rate-1/3 parallel turbo code.

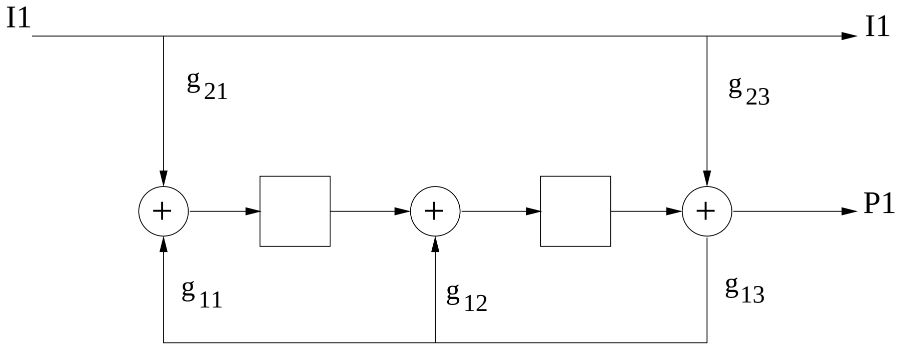


Figure 2.2: RSC constituent code. Octal generators $g = (7,5)$ *i.e.* $g_1 = [g_{11}\ g_{12}\ g_{13}] = [1\ 1\ 1]$ and $g_2 = [g_{21}\ g_{22}\ g_{23}] = [1\ 0\ 1]$. $L_c = 3$.

In general, we can construct turbo codes with more than 2 constituent codes. Now, for two constituents RSC codes $C_i(n_i = k_i + 1, k_i, L_{ci})$, $i = 1, 2$ of rates $R_i = k_i/n_i$ and constraint length L_{ci}, the termination of the trellis needs at most T_i transitions

$$T_i = \lceil \frac{L_{ci} - 1}{k_i} \rceil \tag{2.1}$$

the total rate of the turbo code is equal to

$$R = \frac{R_1 R_2}{R_1 + R_2 - R_1 R_2 + \frac{n_1 T_1 + n_2 T_2}{N}} \tag{2.2}$$

For $R_1 = R_2$ and if we neglect the trellis endings, we get

$$R = \frac{R_1}{2 - R_1} \tag{2.3}$$

We can change the total code rate R by *puncturing* the parity bits available at the outputs of the two constituent codes. For example, to get a code rate $R = 1/2$ we can just puncture the previous code using the following puncturing matrix (at each coding step, one of the two parity bits is deleted)

$$P = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \tag{2.4}$$

A few more parity bits must be punctured if we take in account the trellis endings.

2.2.2 Iterative decoding of turbo codes

A maximum likelihood decoder, as described in chapter 1, cannot be applied to a turbo code. This is due to the complexity of ML decoding. Turbo codes are decoded by an iterative decoding technique known as *turbo decoding.* Its universal version is called the *sum product algorithm.* At each iteration, the constituent codes are decoded using a so called *Soft Input - Soft Output* (SISO) decoder. To understand how turbo decoding works, we first consider the SISO decoding of a convolutional constituent code.

a) SISO decoding of a convolutional constituent code

Let $C(N, K)$ denotes the linear binary constituent code with rate R. If C is a block code then it has length N and dimension K. If C is a finite length binary convolutional code, we neglect the starting and terminating phases in the trellis and we consider C as a (N, K) linear block code.

Let $\mathbf{c} = (c_1, c_2, \ldots, c_N)$ be a codeword and $\mathbf{y} = (y_1, y_2, \ldots, y_N)$ be the vector of received real symbols at the decoder input. The *a posteriori probability* (APP) associated to a coded bit c_j, $j = 1 \ldots N$, is defined as

$$APP(c_j) = P(c_j|\mathbf{y}) = \sum_{\mathbf{c}\in C} P(c_j, \mathbf{c}|\mathbf{y}) = \sum_{\mathbf{c}\in C} \frac{p(\mathbf{y}|c_j, \mathbf{c})P(c_j, \mathbf{c})}{p(\mathbf{y})} \tag{2.5}$$

The joint probability $P(c_j, \mathbf{c})$ is null if the j^{th} bit of $\mathbf{c}$ is not equal to c_j. Assuming that channel observations and *a priori* probabilities are independent, we obtain

$$APP(c_j) \propto \sum_{\mathbf{c}\in C|c_j} p(\mathbf{y}|\mathbf{c})P(\mathbf{c}) = \sum_{\mathbf{c}\in C|c_j} \prod_{\ell=1}^{N} p(y_\ell|c_\ell)\pi(c_\ell) \tag{2.6}$$

where $\pi(c_j)$ is the *a priori* information regarding the bit c_j. The sum in the above equation is performed for a fixed value of c_j. Thus, the APP can be factorized into 3 terms

$$APP(c_j) \propto p(y_j|c_j) \times \pi(c_j) \times Ext(c_j) \tag{2.7}$$

$Ext(c_j)$ is the novelty on the bit computed by SISO decoding, usually called *extrinsic information.* It is defined by

$$Ext(c_j) \propto \sum_{\mathbf{c} \in C | c_j} \prod_{\ell=1, \ell \neq j}^{N} p(y_\ell | c_\ell) \pi(c_\ell) \tag{2.8}$$

Figure 2.3 represents the block diagram of a SISO decoder. The inputs are the likelihoods $p(y_j|c_j), j = 1, \ldots, N$ and the K or N a priori informations $\pi(c_j)$. The outputs of the SISO are the K or N APPs and Extrinsic informations.

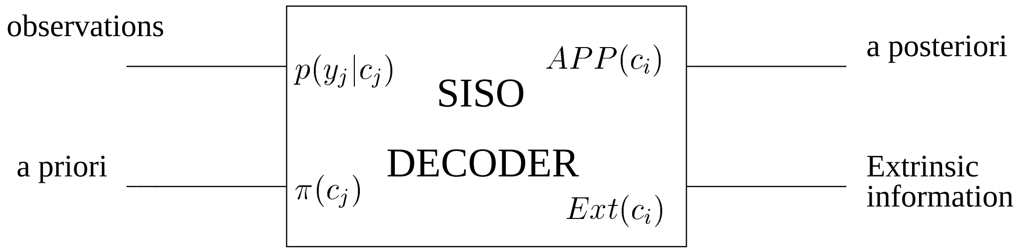


Figure 2.3: SISO decoder.

These formulas are theoretical. In order to evaluate them, we can implement the BCJR algorithm [2] also known by the *Forward-Backward algorithm*. Other sub-optimal techniques can also be implemented [42] [67] [68].

b) Iterative turbo decoding

Figure 2.4 shows the iterative turbo decoder. Two SISO decoders are implemented, one for each constituent code.

We begin by evaluating the likelihoods $p(y_j|c_j), j = 1, \ldots, N$. If c_j is an information bit, $p(y_j|c_j)$ is fed to the first decoder and to the second via an interleaver. This is the same interleaver used in encoding. If c_j is a parity bit, $p(y_j|c_j)$ is fed to one of the two decoders depending on whether c_j is a parity bit of C_1 or C_2. If puncturing was performed on parity bits, the corresponding likelihoods is fixed to 1/2.

At the first decoding iteration, the a priori input of the first decoder is also fixed to 1/2. Then, at each decoding iteration, we begin by decoding the constituent code C_1, the extrinsic output of this decoder is fed through the interleaver to the second decoder. This extrinsic information is considered as an a priori information which will boost the SISO decoder in his evaluation of the APPs of the information bits. The new extrinsic information is fed through an inverse interleaver to the a priori input of the first encoder.

This procedure can be repeated as many times as needed. After a specific number of iterations, the last APP values, of decoder 1 or 2, are used to decide upon the information bits[1]

[1] $APP(c_j)$ represents $APP(c_j = 1)$. $APP(c_j = 0)$ is just equal to $1 - APP(c_j = 1)$.

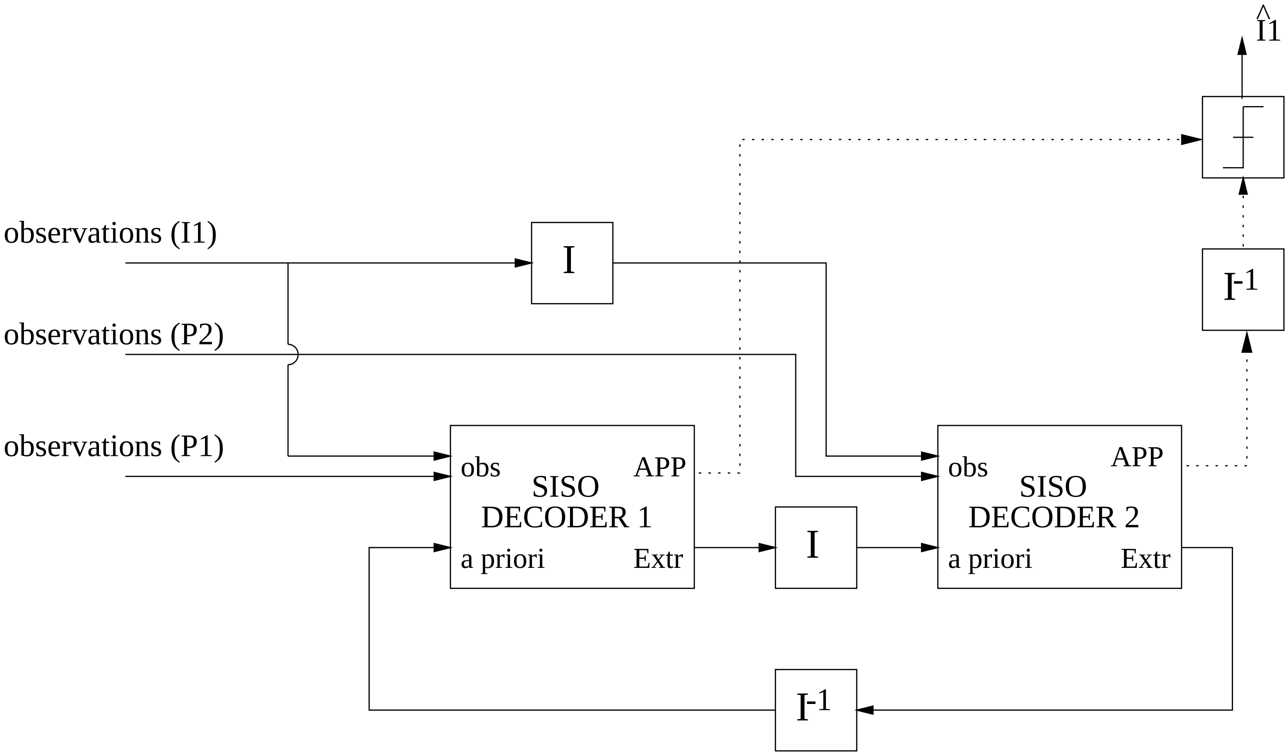


Figure 2.4: Interative decoder of a turbo code.

$$\hat{c}_j = 1 \qquad \Longleftrightarrow \qquad APP(c_j) > 0.5 \tag{2.9}$$

Although iterative turbo decoding is a sub-optimal technique that is used instead of ML decoding, the performance in term of bit error probability goes to ML decoding when the signal-to-noise ratio is large. This is not true for the word error probability, since the SISO decoder only minimizes the bit error probability.

2.2.3 Basic results and design criteria

Turbo codes were widely studied by the researchers in order to understand how they can achieve remarkable coding gains and to find new design criteria. In this section, we summarize the main results and design criteria proposed for the parallel concatenation scheme. Most of these design criteria were found from the analytical performance of turbo codes based on the union bound on the average bit error probability of ML decoding [3] [4].

- Under ML decoding, a parallel turbo code exhibits an interleaving gain of $1/N$ [3] and a serial turbo code has an interleaving gain proportional to $1/N^{(d_1+1)/2}$ [5], where d_1 is the minimum distance of the outer code.

- Iterative decoding approaches the ML performance bound for increasing number of iterations. Therefore, iterative decoding is an efficient and an affordable complexity means to decode most of the compound codes where maximum-likelihood algorithm is not feasible.

- Sub-optimal decoding algorithms can be used instead of the Forward-Backward algorithm in order to reduce the complexity of the system [42] [67] [68].
- The loss in rate due to trellis ending can be avoided by the use of specific RSC codes and tail-biting. This is only significant for small block length N.
- To achieve an interleaving gain equal to $1/N$, the convolution constituent codes of a parallel turbo code must be recursive. In fact, non-recursive convolutional codes don't exhibit any interleaving gain [4].
- The minimum distance d_{free} of the code is less important than the interleaving gain. Nevertheless, the asymptotic performance depends on d_{free}. For small minimum distance, an error floor appears in the curve of the bit error probability.
- To maximize d_{free}, a primitive feedback polynomial must be chosen for the RSC code. d_{free} also depends on the choice of the interleaver.
- The graphical representation of compound codes is a good method in order to check the efficiency of the interleaver [81] [33] [47]. Short cycles in the graph reduce the performance of the compound code.
- The convergence analysis of iterative decoding is difficult for finite block length. For infinite block length, we can assume that the graph is cycle free. The thresholds in terms of signal-to-noise ratio, on the convergence of iterative decoding, can be determined using the density evolution [64] or the Gaussian approximation with error probability matching (EPM-GA) [74]. We study these two methods in the next chapter and show that performance within 0.03 dB from capacity can be achieved using an irregular turbo code decoded with a symbol-based iterative decoder. Symbol-based iterative decoding and irregular turbo codes are defined in the next sections.

2.2.4 Numerical results

Figure 2.5 shows the bit error probability of a rate-1/2 parallel turbo code over the AWGN channel as a function of the average signal-to-noise ratio per bit E_b/N_0 in dB. A BPSK modulation was used. The turbo encoder has two identical constituent codes with octal generator $(37, 21)$ and a randomly chosen interleaver of length $K = 65536$, which means that the code length of the code is equal to $N = 2 * 65536 = 2^{17}$. We executed 20 decoding iterations. We can obviously see the *error floor* due to the the small minimum Hamming distance of the code.

In order to validate the good performance of this code, we can determine the E_b/N_0 that achieves a bit error probability equal to 10^{-5} and compare it to the capacity limit of a binary input AWGN channel, which is equal to $E_b/N_0 = 0.187\ dB$.

Another comparison may be done with the performance of optimal spherical codes derived in the previous chapter. We just need to convert the word error probability of these optimal codes into a bit error probability. For binary optimal codes, we can simply use this equation

$$P_b \simeq \frac{d_{min}}{N} P_e \tag{2.10}$$

where P_e is the word error probability calculated in chapter 1, P_b is the bit error probability, N the code length of the code and d_{min} is the Gilbert-Varshamov bound on minimum distance for a binary code. Gallager (cf. chapter 1, equation (1.60)) showed that for large values of N we have

$$H_2(\frac{d_{min}}{N}) = 1 - R \tag{2.11}$$

where R is the the rate of the binary code in bits/dimension and $H_2(.)$ is the binary entropy function. For $0 < \frac{d_{min}}{N} < \frac{1}{2}$, we have

$$\frac{d_{min}}{N} = H_2^{-1}(1 - R) \tag{2.12}$$

Table 2.1 shows some values of $\frac{d_{min}}{N}$ as a function of the total code rate R.

R	$\frac{d_{min}}{N}$
1/3	0.1739
1/2	0.1100
2/3	0.0615

Table 2.1: Gilbert-Varshamov bound on $\frac{d_{min}}{N}$ as a function of the code rate R.

The bit error probability is sketched in figure 2.5. For $P_b = 10^{-5}$, the turbo code is at 0.55 dB from the performance of a binary optimal code that has the same length $N = 2^{17}$. For $P_b < 10^{-5}$, the degradation is more important due to the error floor. Moreover, a $P_b = 10^{-5}$ can be obtained for the same signal-to-noise ratio by a binary optimal code with a code length almost equal to $N = 3100$. However, it is important to mention that the optimal code is transmitted over a continuous input AWGN channel. A degradation in its performance will occur if BPSK is considered. In this case, the E_b/N_0 that achieves a bit error probability equal to 10^{-5} will be greater than 0.187 dB which is the capacity limit. That is the reason why the real degradation in performance of the turbo code is less than 0.48 dB.

For a code length equal to 2^8, the degradation in performance between the turbo code and the optimal binary code is more important (see figure 2.6). This can be explained by the fact that the gain achieved by turbo code depends on the length of the interleaver.

In the next two sections, we improve the performance of turbo codes first, by applying *symbol-based* iterative decoding, and second, using *irregular* turbo codes.

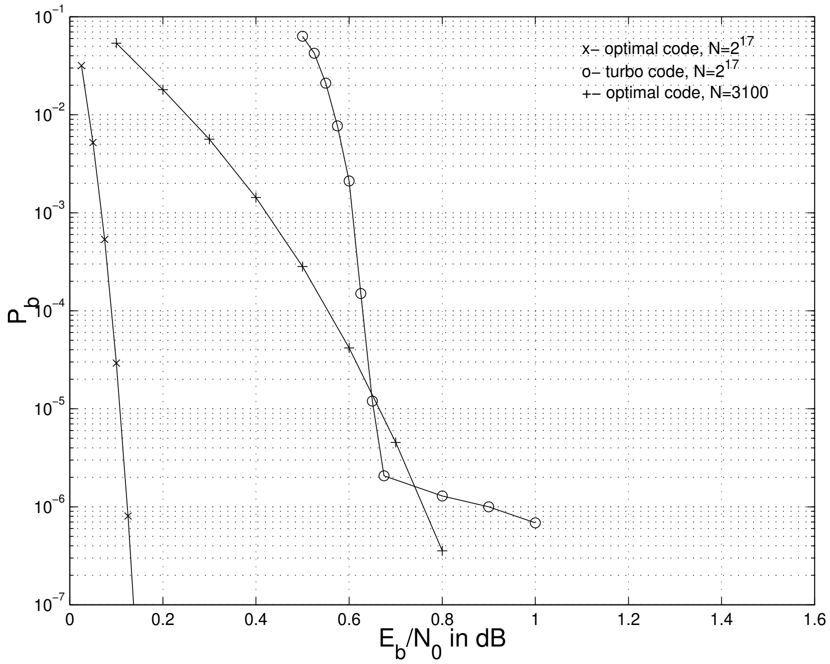


Figure 2.5: Bit error probability of turbo codes and optimal codes. $R = 1/2$. $N = 2^{17}$.

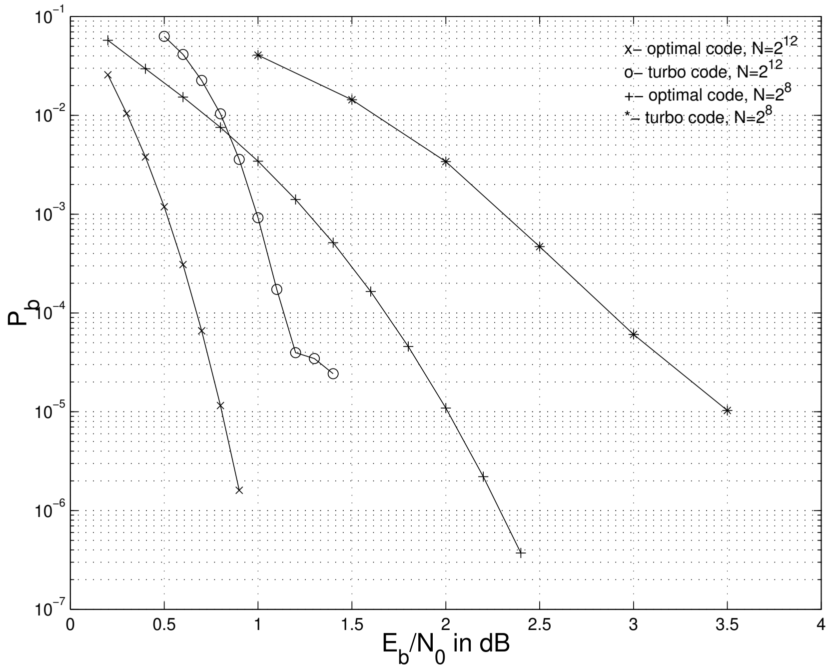


Figure 2.6: Bit error probability of turbo codes and optimal codes. $R = 1/2$. $N = 2^8$ and 2^{11}.

2.3 Symbol-based iterative decoding

Let us consider concatenated codes based on binary convolutional constituent codes of rate $k/k+1$, converted into block codes with finite length and dimension. To improve the performance of iterative decoding, the *graph cycles* are reduced by grouping bit nodes into symbol nodes, *i.e.* m bits of a trellis transition are combined into a $Q = 2^m$-ary symbol.

2.3.1 Construction

Let C denote a linear binary convolutional code with rate $R = k/n$, converted into an (N, K) linear block code. Let $\mathbf{c} = (c_1, c_2, \ldots, c_j \ldots c_N)$ be a codeword and $\mathbf{y} = (y_1, y_2, \ldots, y_N)$ be the real received vector available at the decoder input. When m coded bits c_j are combined into one $Q = 2^m$-ary symbol $S_j = (c_{m.j-m+1}, c_{m.j-m+2}, \ldots, c_{m.j})$, $j = 1, 2, \ldots, n = N/m$, the vector of received values is rewritten $\mathbf{y} = (Y_{s_1}, Y_{s_2}, \ldots Y_{s_j}, \ldots, Y_{s_n})$ where $Y_{s_j} = (y_{m.j-m+1}, y_{m.j-m+2}, \ldots, y_{m.j})$.

For example, we consider a turbo code (parallel concatenation) based on two RSC codes of rate 2/3. The final turbo code rate is $R = 1/2$. In this case, only the information bits are grouped into quaternary symbols. Each couple of information bits belonging to the same trellis transition are combined into one quaternary symbol. There is no need to do the same to parity bits. In fact, the extrinsic information propagates in the graph of the code which contains only bit or symbol information nodes, and subcode nodes. Consequently, each trellis branch includes one quaternary information symbol and one parity bit. At the encoder, the interleaving is done on a *symbol-by-symbol* basis.

An equivalent construction of quaternary turbo codes is possible for rate-$1/n$ constituent codes. This can be done by merging two consecutive transitions in the trellis of the convolutional constituent code [10].

Note that we are not dealing with non-binary codes [7]. The turbo code is binary (we just used a symbol by symbol interleaver), but the decoding is done on quaternary information symbols and parity bits.

2.3.2 APP evaluation

The APP associated to a symbol S_j is defined by

$$APP(S_j) \propto p(Y_{s_j}|S_j) \times \pi(S_j) \times Ext(S_j) \tag{2.13}$$

where the new extrinsic information is

$$Ext(S_j) = \sum_{\mathbf{c} \in C|S_j} \prod_{\ell=1, \ell \neq j}^{n} p(Y_{s_\ell}|S_\ell)\pi(S_\ell) \tag{2.14}$$

The SISO decoder can be implemented using the forward-backward algorithm as for the binary case. The only difference, when *quaternary* symbols are considered, is in the number of inputs and the number of APPs and extrinsic informations that must be evaluated for each symbol.

In the binary case, we have for example two APPs, one for $c_j = 0$ and a second one for $c_j = 1$. These two APPs verify $APP(c_j = 0) + APP(c_j = 1) = 1$. The same is true for $p(y_j|c_j)$, $\pi(c_j)$ and $Ext(c_j)$. Therefore, for each bit, the length of the inputs and the outputs of the SISO decoder is equal to one. For example, we just need the APP for $c_j = 1$ as in equation (2.9).

For quaternary symbols, we have $S_j = (c_{2j-1}, c_{2j})$. Four APPs must be defined ($APP(S_j = (0,0))$, $APP(S_j = (0,1))$, $APP(S_j = (1,0))$ and $APP(S_j = (1,1))$) and the length of the inputs and the outputs of the SISO decoder is equal to three. In order to make a decision on the bits c_j, we determine $APP(c_j = 1)$ using the following equation

$$APP(c_j = 1) = \begin{cases} APP(S_p = (1,0)) + APP(S_p = (1,1) & if\ j = 2p-1 \\ \\ APP(S_p = (0,1)) + APP(S_p = (1,1) & if\ j = 2p \end{cases} \tag{2.15}$$

In other hand, if we use a BPSK modulation to transmit the coded bits, the likelihoods $P(Y_{s_j}|S_j)$ must be calculated using $P(y_{2j-1}|c_{2j-1})$ and $P(y_{2j}|c_{2j})$. For example, we have

$$P(Y_{s_j}|S_j = (0,0)) = \frac{P(y_{2j-1}|c_{2j-1} = 0).P(y_{2j}|c_{2j} = 0)}{\sum_{i=0}^{1}\sum_{l=0}^{1} P(y_{2j-1}|c_{2j-1} = i).P(y_{2j}|c_{2j} = l)} \tag{2.16}$$

where the normalization was done in order to get a sum of likelihoods equal to 1.

2.3.3 Numerical results

We consider a rate-1/2 turbo code constructed using rate-2/3 RSC constituent codes. The octal generators are $g = (37, 21, 27)$ and the code length is equal to $N = 2^{17}$. For code length $N = 2^{11}$, we consider a turbo code that has a better minimum distance in order to get an error floor at a bit error probability smaller than 10^{-5} (octal generators $g = (31, 27, 37)$). Both bit-based and symbol-based iterative decoding are applied. Figures 2.7 and 2.8 show the bit error probability as a function of the average signal-to-noise ratio per bit E_b/N_0.The gain in signal-to-noise ratio achieved by symbol-based iterative decoding over bit-based iterative decoding is almost equal to 0.13 dB for both $N = 2^{17}$ and $N = 2^{11}$. However, when compared to the punctured rate-1/2 turbo code constructed using rate-1/2 constituent codes, the gain is smaller especially for $N = 2^{17}$ where the achieved gain is equal to 0.05 dB. For $N = 2^{17}$, and a rate-2/3 constituent code with octal generators $g = (25, 37, 35)$, a bit error probability equal to 10^{-4} is achieved at $E_b/N_0 = 0.51$ dB, when symbol-based iterative decoding is done; This is at 0.31 dB of the capacity limit. However, the error floor appears at $P_b > 10^{-5}$.

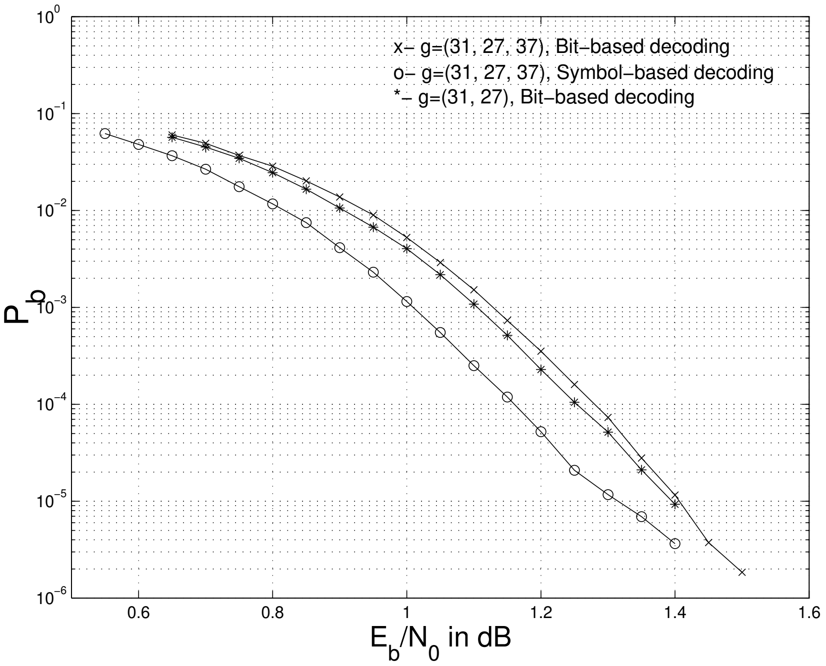


Figure 2.7: Bit error probability of TC with symbol-based iterative decoding. $R = 1/2$. $N = 2^{11}$.

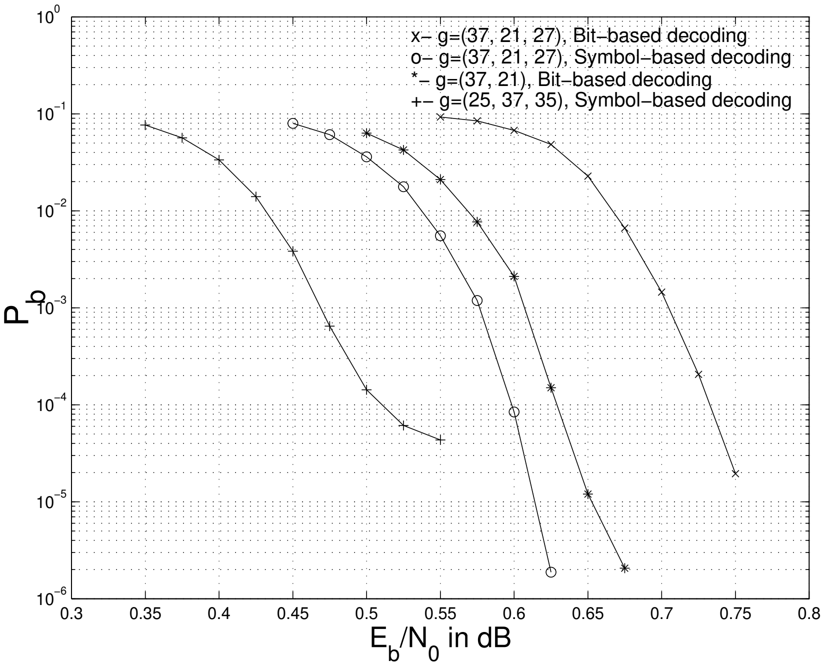


Figure 2.8: Bit error probability of TC with symbol-based iterative decoding. $R = 1/2$. $N = 2^{17}$.

2.4 Irregular turbo codes

Recent work on *irregular* LDPC codes has shown that by making the codeword bits participate in varying numbers of parity check equations, significant coding gain can be reached [55] [65]. For code length equal to $N = 10^5$, Richardson *et al* [65] found irregular LDPC codes that perform better than the original rate-1/2 turbo code which we described in section 2.2.4. For a bit error probability equal to 10^{-5}, the coding gain achieved by irregular LDPC is equal to 0.16 dB. Frey *et al* [32] [34] introduced irregularity to turbo codes in order to achieve better performance. Very interesting results were found for code rates equal to $R = 1/3$ and $R = 1/2$. However, for $R = 1/2$ and a code length equal to $N = 2^{17}$ an error floor appeared at a bit error probability greater than 10^{-4}.

In this section, we combine *irregular* turbo codes with *symbol-based* iterative decoding in order to achieve better performance and to lower the level of the *error floor*. Using this method, we were able to perform as good as, and even better than LDPC codes at a bit error probability equal to 10^{-5} and a code length $N = 10^5$. We begin by reviewing the basics of irregular turbo codes then we design and simulate our new irregular turbo coding scheme that implement symbol-based iterative decoding.

2.4.1 Irregular turbo encoding

Let us consider a regular turbo code $C(N, K)$ as defined in section 2.2.1. K is the information bits length and N is the code length. Each information bit is used twice while encoding; first, by the constituent code C_1 and second by C_2 after interleaving. Now if we add an interleaver at the input of the first encoder we get a similar coding scheme that has the same performance. This new coding scheme can be implemented differently using a $(2, 1)$ binary repetition code, an interleaver "I" of length $2K$ at the output of the repetition code and one RSC constituent code (see figure 2.9). Only the K information bits and the $N - K$ parity bits of the RSC code are transmitted by the encoder.

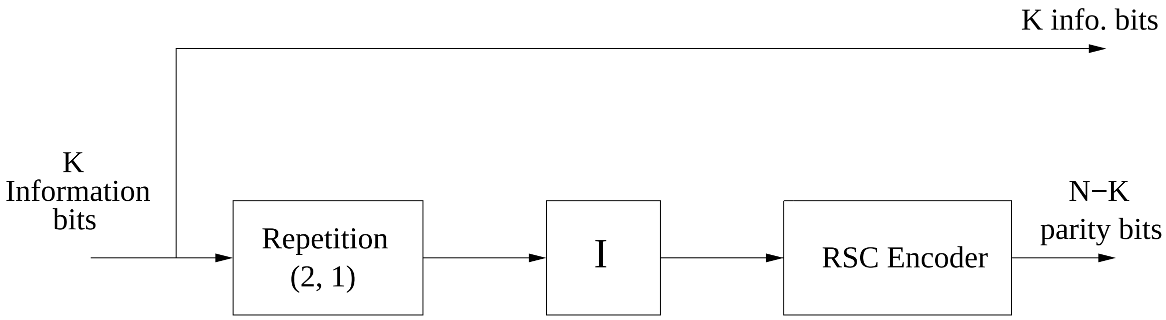


Figure 2.9: A similar encoding scheme for turbo codes.

The *degree* of an information bit is equal to the number of repetitions done to the bit. In this case, all the information bits have the same degree $d = 2$. Notice that the performance of this regular turbo code is identical to the one we get using the standard turbo encoder of figure 2.1 when the interleaver length is sufficiently high.

Now to construct an irregular turbo code, we just need to divide the K information bits into classes where each class has a specific degree d. If $f_d \in [0, 1]$ is the fraction of bits of degree d, the average information bits degree is

$$d_{av} = \sum_{d=d_{min}}^{d_{max}} d.f_d \tag{2.17}$$

The minimum degree d_{min} verifies $d_{min} \geq 2$. The maximum degree is d_{max}. The irregular turbo encoder can be implemented as in figure 2.10.

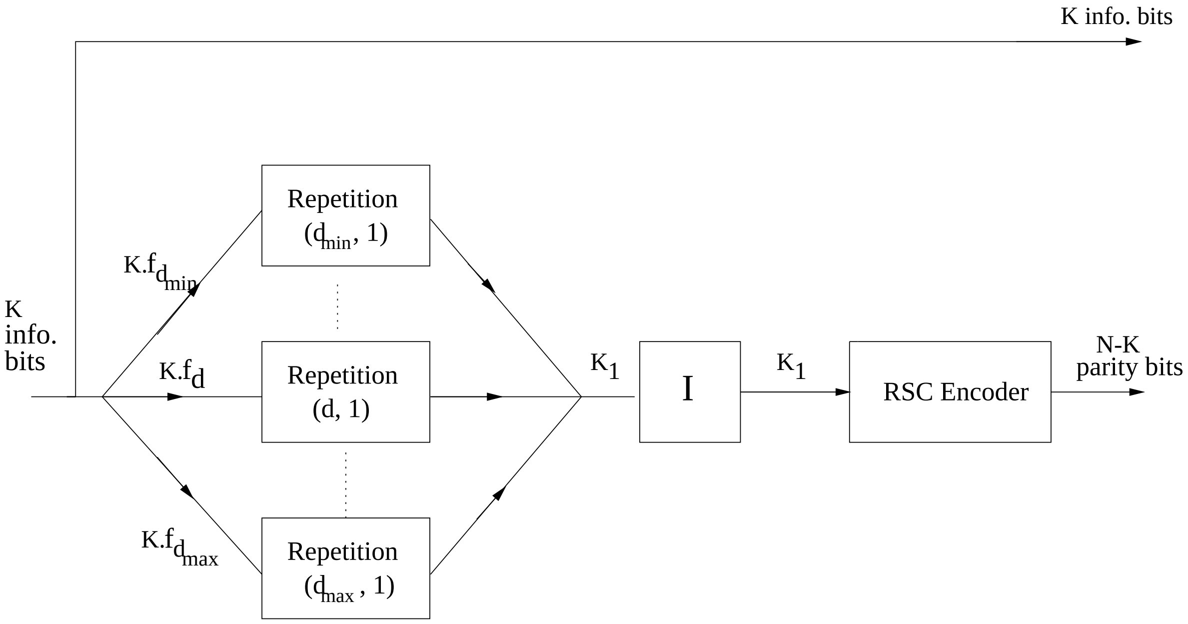


Figure 2.10: Encoding scheme of an irregular turbo.

The RSC constituent code can be considered as a linear block code $C_1(N_1, K_1)$. Let R_1 be the code rate of this code. The number of bits at the input of the constituent code is $K_1 = K.d_{av}$ and the number of parity bits is $N_1 - K_1 = K_1(\frac{1}{R_1} - 1)$. R_1 is related to the total code rate R of the irregular turbo code and the average degree of information bits by

$$d_{av} = \frac{\frac{1}{R} - 1}{\frac{1}{R_1} - 1} \tag{2.18}$$

In fact, the number of parity bits for the irregular code is equal to the one we have for the constituent code

$$N - K = K(\frac{1}{R} - 1) = K_1(\frac{1}{R_1} - 1)$$

.

Example

Let us examine the construction of a rate-1/2 irregular turbo code. We consider a degree profile where only $d = 2$ and $d = 10$ have none zero fractions. The fractions are $f_2 = 0.9$ and $f_{10} = 0.1$. Thus, the average degree of the information bits is equal to $d_{av} = 2.8$. From equation (2.17), we get that the code rate of the constituent code is equal to $R_1 = 0.7368$. R_1 is greater than R since $d_{av} > 1$. Moreover, if the average degree is increased, the rate of the constituent code must also be increased.

For $K = 50000$ and $N = 100000$ we have $K_1 = K.d_{av} = 140000$ and $N1 = 190000$. If, the constituent code is a rate-1/2 RSC code, then 90000 parity bits must be punctured in order to get $R_1 = 0.7368$. This is 64.3% of the total parity bits. The minimum distance of the code significantly decreases and an error floor may appear at relatively high bit error probability. When a rate-2/3 RSC code is used, 20000 parity bits, *i.e.* 18.2% of the total parity bits must be punctured. This may lower the error floor due to small minimum distance.

Remark

To define the irregular turbo code, we specified the degree profile of the information bits and not the degree profile of coded bits as in the papers of Frey *et al* [32] [34] (the degree of parity bits is equal to one). The two definitions are similar. For a degree profile on coded bits, the rate-1/2 irregular turbo code analyzed in the previous example will have non zero fractions for $d = 1$, $d = 2$ and $d = 10$. The fractions will be $f_1 = 0.5$, $f_2 = 0.45$ and $f_3 = 0.05$. the average degree of coded bits is equal to $d_{av} = 1.9$. We get the same code rate for the constituent code ($R_1 = 0.7368$) using the following equation

$$d_{av} = \frac{R - 1}{R_1 - 1} \tag{2.19}$$

2.4.2 Decoding irregular turbo codes

Figure 2.11 shows a block diagram of the decoder. One SISO decoder is needed, since there is only one RSC constituent code. The coded bits of the irregular turbo code are denoted $c_j,\ j = 1, \ldots, N$.

Let $c_{ji},\ i = 0, \ldots, d_j - 1$ be the d_j repetitions of the coded bits, where d_j is the degree of the coded bit c_j (For the parity bits, we have $d_j = 1$). In this case, the N_1 coded bits of the RSC code are the N_1 bits $c_{ji},\ j = 1, \ldots, N;\ i = 0, \ldots, d_j - 1$.

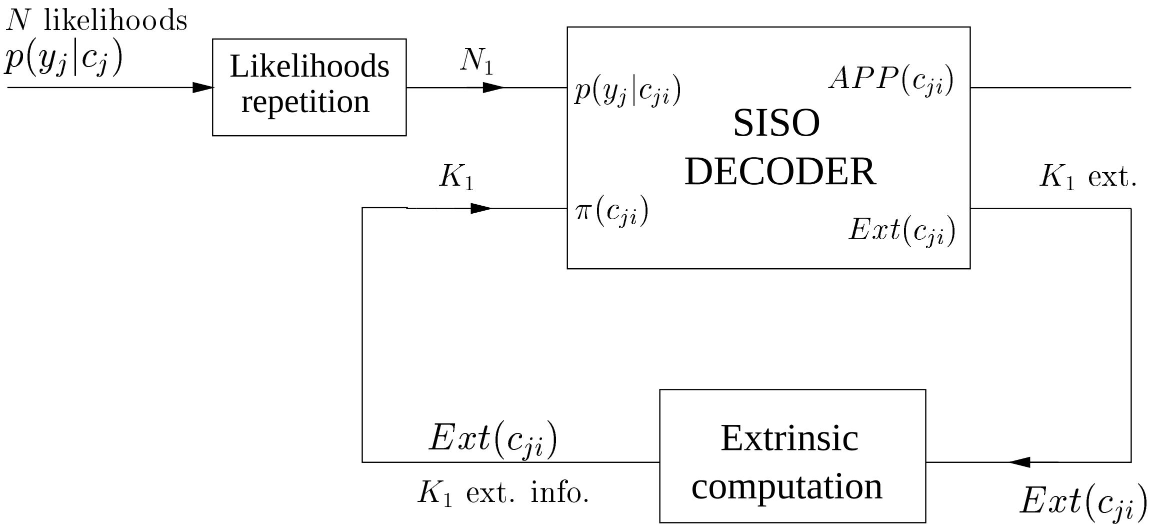


Figure 2.11: Iterative decoder of an Irregular turbo code.

The inputs of the SISO decoder are the N_1 likelihoods $p(y_j|c_{ji})$, $j = 1, \ldots, N$, $i = 0, \ldots, d_j - 1$ and the K_1 a priori informations $\pi(c_{ji})$ of the input bits. The outputs of the SISO decoder are the N_1 or K_1 APPs and the K_1 extrinsic informations on the input bits of the constituent code.

We begin by evaluating the likelihoods $p(y_j|c_j), j = 1, \ldots, N$ at the channel output. A repetition is applied to these likelihoods in order to determine the N_1 likelihoods $p(y_j|c_{ji})$ of the constituent code. This operation is represented by the block "Likelihoods repetition" in figure 2.11. If c_j is one of the K information bits, then a search is made in order to get the degree d_j and the positions of the repetitions c_{ji}, $i = 0, \ldots, d_j - 1$ after interleaving was done. Then the likelihoods of the repetitions are fixed to $p(y_j|c_j)$. No repetition is needed for the $N - K = N_1 - K_1$ likelihoods of the parity bits, since the degree of a parity is equal to one.

At the first decoding iteration, the a priori information on the K_1 input bits of the constituent code are fixed to 1/2. Then, at each decoding iteration, SISO decoding is applied.

The K_1 extrinsic informations at the output of the SISO decoder must be treated by the "Extrinsic computation" block in order to get the new K_1 extrinsic informations regarding the bits c_{ji}. In fact each of these bits has a degree d_j. Therefore, the total Extrinsic information is the product[2] of the $d_j - 1$ extrinsic informations of the bits c_{jl}, $l \neq i$.

$$Ext(c_{ji}) = \prod_{l=0, l \neq i}^{d_j - 1} Ext(c_{jl}) \tag{2.20}$$

[2]The extrinsics on coded bits c_j must be normalized in order to get $Ext(c_j = 0) + Ext(c_j = 1) = 1$. This is note necessary if we use log ratios instead as in chapter 3.

The total APP of an information bit $c_j = c_{j0}$ is equal to

$$APP(c_j) \propto p(y_j|c_j) \times \pi(c_j) \times Ext(c_{j0}) \tag{2.21}$$

After a specific number of iterations, decision is done using the APP values of the information bits obtained from equation (2.21). The number of iterations needed for convergence is significantly greater than the one we have for regular turbo codes. An explanation to this phenomena can be found by analyzing the convergence of iterative decoding of turbo codes and irregular turbo codes. This is discussed in the next chapter.

2.4.3 Selecting the degree profile

The search for good *degree profiles* is very complex in practice, due to the large number of parameters that must be optimized. For example, for $d_{min} = 2$ the number of degrees and fractions of a profile is equal to $2(d_{max} - 1)$. However, we just have 2 equations

$$d_{av} = \sum_{d=2}^{d_{max}} d.f_d \tag{2.22}$$

$$\sum_{d=2}^{d_{max}} f_d = 1 \tag{2.23}$$

Therefore, we do as in [34]. We just choose two non zero fractions. One for degree 2, which is the degree of information bits of a regular turbo code. The second one is $d_e > 2$. The fractions of bits of degree d_e is denoted f_e. This is the fractions of "elite" bits which will increase the performance of the iterative decoder. The convergence for these bits is reached first. For the remaining iterations, they can be considered as "Pilot" bits [89]. Equations (2.22) and (2.23) become

$$2f_2 + d_e f_e = d_{av} = \frac{\frac{1}{R} - 1}{\frac{1}{R_1} - 1} \tag{2.24}$$

$$f_2 + f_e = 1 \tag{2.25}$$

For a fixed rate R, the number of parameters is 4: d_e, f_e, f_2 and the rate R_1 of the constituent. The optimization of these parameters can be done using the density evolution or the Gaussian approximation with error probability matching (EPM-GA) if the interleaver has a very large or infinite length (cf. chapter 3). For finite length, this must be done by Monte Carlo simulation of the bit error probability.

We begin by fixing the degree d_e of "elite" bits and varying the fraction f_e. For a particular value of f_e, R_1 and f_2 are determined using equations (2.24) and (2.25). The fraction that achieves the best performance is denoted f_{eopt}. Now, for a fixed fraction $f_e = f_{eopt}$, we change the degree d_e in order to determine d_{eopt}.

This profile is not the best, since we did not optimize over all the combinations (d_e, f_e). Furthermore, better performance may be attained using a profile with more than two non zero fractions.

As an example, we consider a rate-1/2 irregular turbo code constructed using a rate-2/3 RSC constituent code. The octal generators of the RSC codes are $g = (37, 21, 27)$. Symbol-based iterative decoding is applied, which means that the interleaving is done on a symbol-by-symbol basis. The code length of the code is equal to N^{17}. The average signal-to-noise ratio per bit is fixed to 0.4 dB. Using the method described previously, we find $f_{eopt} = 0.1$ and $d_{eopt} = 10$ (see figures 2.12 and 2.13).

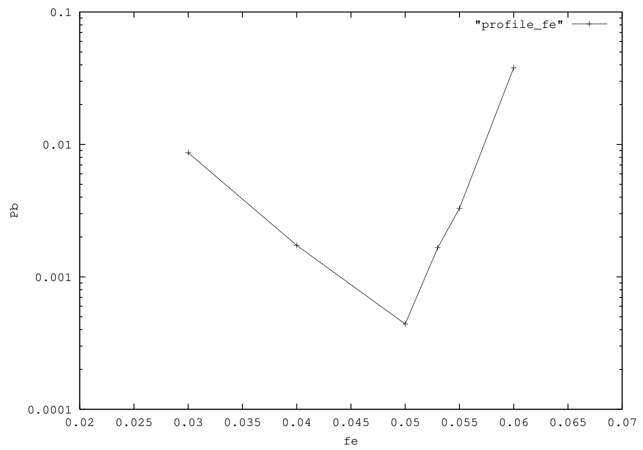


Figure 2.12: Fraction f_e of degree $d_e = 10$. $E_b/N_0 = 0.4$ dB.

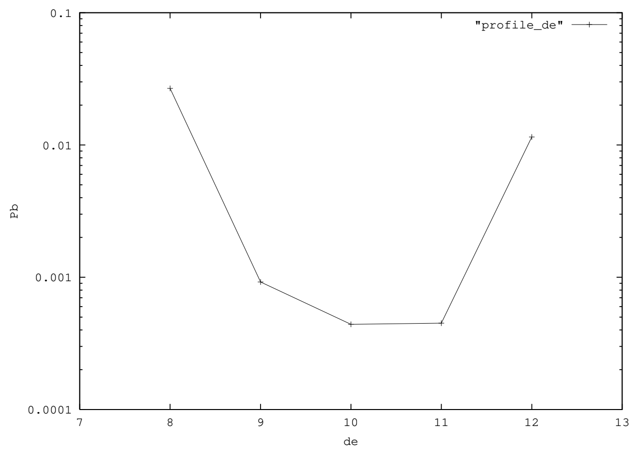


Figure 2.13: Degree of "elite" bits for a fraction $f_e = 0.1$. $E_b/N_0 = 0.4$ dB.

2.4.4 Performance

Figure 2.14 shows the performance of the rate-1/2 irregular turbo code analyzed in the previous section ($R_1 = 2/3$; g= $(37, 21, 27)$; $N = 2^{17}$). We did 100 decoding iterations. The code is at 0.23 dB from capacity at a bit error probability equal to 10^{-5}. The error floor in the performance of the rate-1/2 irregular turbo code proposed by Frey and MacKay [34] is lowered significantly. It appears at a bit error probability equal to 6×10^{-6} instead of 4×10^{-4}. This is mainly due to the use of a rate-2/3 instead of a rate-1/2 RSC constituent code. For a bit error probability greater than 2×10^{-4}, the gain achieved by our system is very small. Now, when compared to the standard turbo code proposed by Berrou *et al* [6], the gain achieved in signal-to noise ratio is equal to 0.24 dB at a bit error probability equal to 10^{-5}.

In figure 2.15, we compare our rate-1/2 irregular code to the best LDPC codes found by Richardson, Shokrollahi and Urbanke [65]. For a code length N equal to 10^4, the gain achieved by the irregular turbo code is equal to 0.1 dB for a bit error probability equal to 10^{-5}. For $N = 10^5$, the two systems perform almost equaly.

For reference, we sketche the performance of the rate-1/3 irregular turbo code proposed by Frey and MacKay [34] (see figure 2.16). Performance is compared with the rate-1/3 standard turbo code of the Consultative Committee for Space Data Systems (CCSDS). The octal generators are $g = (31, 33)$. $K = 8920$ and $N = 26760$. The degree profile of information bits is $d_e = 12$ and $f_e = 0.11$. The gain achieved by the irregular code is equal to 0.2 dB at a bit error probability equal to 10^{-5}.

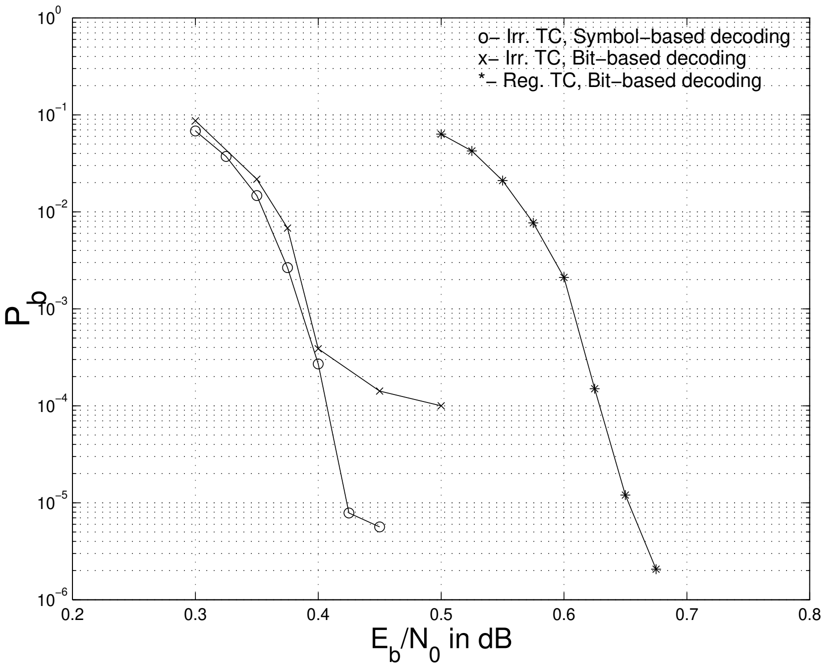


Figure 2.14: Bit error probability of regular and irregular turbo codes. $R = 1/2$. $N = 2^{17}$.

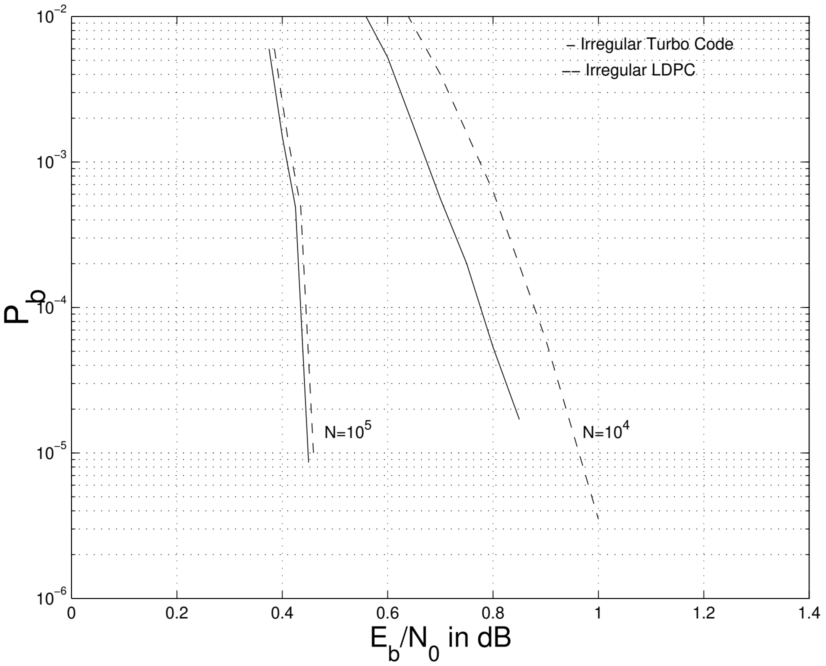


Figure 2.15: Bit error probability of irregular turbo codes with symbol-based iterative decoding compared to LDPC codes. $R = 1/2$. $N = 10^4$ and 10^5.

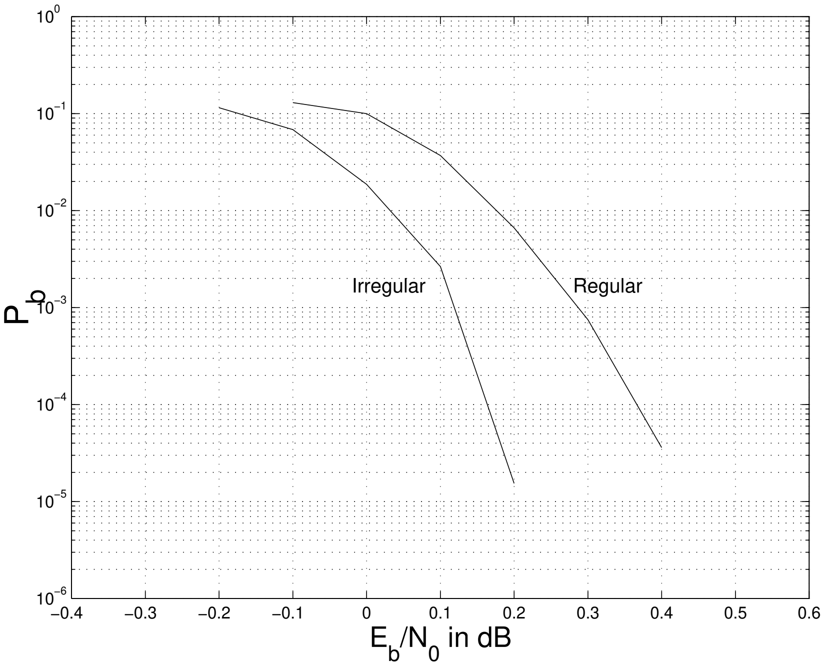


Figure 2.16: Bit error probability of the rate-1/3, $K = 8920$, $N = 26760$, CCSDS turbo code compared to the irregular turbo code with the same parameters.

2.5 Conclusion

In this chapter, we reviewed the basics of regular and irregular turbo codes. We introduced symbol-based iterative decoding in order to achieve better performance. The Rate-1/2 irregular turbo codes bases based on a rate-1/2 RSC constituent code suffered from error floor, which appears at a bit error probability greater than 2×10^{-4} for a code length equal to 2^{17}. This is due to the large number of punctured parity bits which reduces the minimum distance of the irregular turbo code. We lowered the error floor using a rate-2/3 RSC constituent code with octal generators $g = (37, 21, 27)$. Symbol-based iterative decoding was done in order to achieve better performance. A bit error probability equal to 10^{-5} was achieved at a signal-to-noise ratio equal to 0.42 dB. This is at 0.23 dB from the capacity of the binary input AWGN channel. Moreover, for a bit error probability equal to 10^{-5}, the performance was better than the best known LDPC codes even for very large values of N. In any case, for small values of N, a regular turbo code can achieve better performance than LDPC.

Now, we just need to find if regular and irregular turbo codes can achieve capacity as for LDPC codes. Hence, we study their asymptotic performance in the next chapter.

Chapter 3

Asymptotic performance of turbo codes

3.1 Introduction

The graph representation of error-correcting codes [81][47] is a powerful tool to study the behavior of an *iterative decoder* when applied to any compound code with an interleaver of infinite size. *Iterative decoder* is also known as the *message passing algorithm* or *probabilistic decoder* [35] [81] and its universal version is called the *sum-product algorithm* [48].

Recently, Richardson and Urbanke [64] presented a method for determining the performance limits of Gallager low-density parity check codes under iterative decoding. The performance limit of such infinite length code under iterative decoding provides the minimal signal-to-noise ratio for which the bit error probability vanishes when the number of iterations goes to infinity. The key idea is to estimate the probability density function of the decoder output from its input density. This approach is known as *density evolution*. Huge work was done in this field[1] [55] [65] [66]. Boutros *et al* [95] introduced the property called *isotropy* which controls the number of distinct distributions propagating in the graph of the compound code. Other authors proved that large random graphs can be *cycle-free*. This allows the calculation of precise convergence thresholds. The parameters of the code can be optimized in order to reduce these thresholds. Using this approach, Urbanke *et al* [15] found irregular low-density parity check codes with performance at 0.0045 dB from the Shannon limit.

To speed up the search for good code parameters we can use a sub-optimal method that leads to quite accurate results. This is the *Gaussian approximation* method which was defined in very different ways [86] [24] [16] [74]. A Gaussian approximation based on extrinsic information transfer functions was first introduced by Ten Brink [86]. Using this method, he recently designed a rate-1/2 serial concatenated code with a repetition code and an eight-state rate-one code that is within 0.1

[1] refer to the special issue on codes on graphs and iterative algorithms, IEEE transactions on Information theory, Vol. 47, No 2, February 2001

dB of the Shannon limit [88]. Gamal and Hammos introduced a similar Gaussian-approximation based on signal-to-noise ratio matching (SNRM-GA) [24]. Boutros *et al* [74] introduced the Gaussian-approximation with error probability matching (EPM-GA) which is of great interest for analyzing the convergence of the iterative decoder.

In this chapter, we study the asymptotic performance of turbo and irregular turbo codes on a Gaussian channel with BPSK modulation. We consider both binary and quaternary symbol-based iterative decoding. The graphical representation of the code is based on Tanner's bipartite graphs [81]. The search for good code parameters are first done using the Gaussian-approximation with error probability matching. Results are refined and exact thresholds are determined using density evolution. For example, for a rate-1/2 irregular turbo code with a binary recursive convolutional constituent code of rate 2/3 and octal generators $g = (37, 21, 27)$, a threshold within $0.03dB$ of Shannon limit is found. Quaternary symbol-based iterative decoding is done.

The system model is described in section 3.2. In section 3.3, we review the isotropy property introduced by *Boutros et al* [95]. In section 3.4 , we define the partial a posteriori probability and describe the way to evaluate the log-ratios (LR) for a given constituent code C. We consider both binary and Q-ary symbols. Section 3.5 focuses on the graph representation of iterative decoding and the associated APP density propagation. The performance limit is determined by estimating the probability density function of the decoder output via a propagation formula established under isotropy conditions [95]. In section 3.6, we define the transfer function of a subcode node that assumes a Gaussian distributed a priori information. Finally, numerical results on the asymptotic performance of regular and irregular turbo codes are shown in section 3.7.

3.2 System model

The system model is identical to the one defined in the previous chapter. The turbo code is considered as a linear block code $C(N, K)$; a codeword is denoted by $\mathbf{c} = (c_1, c_2, \ldots, c_N)$ and the channel output is represented by $\mathbf{y} = (y_1, y_2, \ldots, y_N)$.

For symbol-based iterative decoding, we have the same notations as in section 2.3 of chapter 2. We also assume that

- the interleaver size is infinite and the graph representation of the turbo code is free from cycles.
- the all zero codeword is transmitted over an additive white Gaussian noise (AWGN) channel using a binary phase shift keying (BPSK) modulation

$$y_j = -1 + n_j; \quad j = 1, \ldots, N \tag{3.1}$$

where n_j is the zero-mean random variable with variance $\sigma^2 = N_0$. The average energy per coded bit is $E_c = 1/2$ and the average energy per information bit is $E_b = E_c/R$.

We begin by a review of the *isotropy* property as defined by Boutros *et al* [95]. This property simplifies the analyze of the density propagation in the graph of compound codes.

3.3 Isotropy property

Definition 3.1 *Conditional Weight Enumerator*
The two conditional weight enumerator polynomials associated to a bit c_j, for $j = 1 \ldots N$, are $A_j^0(x) = \sum_i A_{ji}^0 x^i$ and $A_j^1(x) = \sum_i A_{ji}^1 x^i$, where the integers A_{ji}^0 and A_{ji}^1 are the number of codewords in C of Hamming weight i such that $c_j = 0$ and $c_j = 1$ respectively.

From Definition 1, it is obvious that $A_j^0(x) + A_j^1(x) = A(x)$, where the latter is the complete weight enumerator polynomial [58]. It is sometimes useful to introduce the indeterminate y associated to the number of zero bits, e.g. $A_j^0(x,y) = \sum_i A_{ji}^0 x^i y^{n-i}$.

Equation (2.6) combined to definition 1 allows us to write the total a posteriori probability as in Result 1, where the special notation $\otimes$ represents the sum or the product of two identically distributed random variables X and Y. The sum $X+Y$ is written as $\otimes$ and the product $X.Y$ as $X^{\otimes 2}$.

Result 3.1 *APP and Conditional Weight*
The input random variables $p(y_\ell|c_\ell)\pi(c_\ell)$ are identically distributed. Then,

$$APP(c_j = 1) \propto \sum_{i=0}^{N} A_{ji}^1 \otimes [p(y_1|c_1 = 1)\pi(c_1 = 1)]^{\otimes i} [p(y_1|c_1 = 0)\pi(c_1 = 0)]^{\otimes n-i} \quad (3.2)$$

The complementary probability $APP(c_j = 0)$ has a similar expression. From equation 3.2, we deduce that the probability density function of $APP(c_j = 1)$ and the input distributions of the a priori and the observation are linked via a transformation defined by $A_j^1(x,y)$, or equivalently, the set of coefficients $\{A_{ji}^1\}$. In the special case where $A_j^1(x)$ is independent of the position j, $A_j^1(x) = A^1(x)$ and $A_j^0(x) = A^0(x)$, all the bits of the constituent code hold the same APP distribution, *i.e.*, all bits are equally protected by the information propagation in the graph. This is the isotropic property.

Definition 3.2 *Isotropic Code*
The constituent code C is called isotropic if the probability distribution of the a posteriori information is independent of the bit position. Otherwise, C is said to be anisotropic.

The definition above assumes that the channel observations are independent and identically distributed. The same assumption is made for the a priori probabilities. The degree of anisotropy $d_a(C)$ is the number of distinct APP distributions, $1 \leq d_a(C) \leq N$. Clearly, $d_a(C) = 1$ if C is isotropic. The determination of the anisotropy degree is very important for the performance analysis of an iterative decoder. Of course, this analysis can be considerably simplified if $d_a(C) = 1$, where we need to examine the evolution of only one type of probability distribution.

Example 3.1
The (7,4,3) binary Hamming code is isotropic. Indeed, the conditional weight enumerator expression is independent from the bit position. $A^0(x) = 1 + 4x^3 + 3x^4$ and $A^1(x) = 3x^3 + 4x^4 + x^7$. It is very easy to check that the extended code (8,4,4) is still isotropic with conditional weight enumerators $1 + 7x^4$ and $7x^4 + x^7$.

Example 3.2
The (9,3,3) binary cyclic code generated by the polynomial $x^6 + x^3 + 1$ is isotropic. The conditional weight enumerators are $A^0(x) = 1 + 2x^3 + x^6$ and $A^1(x) = x^3 + 2x^6 + x^9$.
The extended code (10,3,4) is anisotropic with $d_a = 2$. The weight enumerator $A^0(x)$ of the parity bit added by the extension is $1 + 3x^6$, whereas the 9 remaining bits have the conditional weight distribution $1 + 2x^4 + x^6$.

Result 3.2 *cyclic block codes*
A cyclic code C is isotropic.

Proof. Consider two bits c_j and c_ℓ at positions j and $\ell > j$. The coefficient A^0_{ji} is the number of weight i codewords in C satisfying $c_j = 0$. Since C is cyclic, a cyclic shift of $\ell - j$ bits converts any codeword with $c_j = 0$ to another codeword of equal weight with $c_\ell = 0$. We deduce that $A^0_{ji} = A^0_{\ell i}$ for all j and ℓ. *QED.*

Lemma 3.1 *extension of an isotropic code*
The extension of an isotropic code is isotropic iff the conditional weight enumerator of the initial code satisfies $A^1_i = A^0_{i-1}$ for all even i.

Proof. The isotropic $C(N,K)$ code is extended to a $C_e(N+1,K)$ code by adding a zero sum parity check bit c_{N+1}. Let $A(x) = A^0(x) + A^1(x)$ be the complete weight enumerator of C. Let $A_e(x) = A^0_e(x) + A^1_e(x)$ be the complete weight enumerator of C_e. We have to prove that $A^0_e(x)$ is independent from the bit position j. First, consider $1 \leq j \leq N$. Then $A^0_e(x) = \sum_{i\ even} A^0_i x^i + \sum_{i\ odd} A^0_i x^{i+1}$. It is obvious that both terms in $A^0_e(x)$ do not depend on j because the initial code C is isotropic.
Now, consider the parity check bit c_{N+1}. The associated conditional weight enumerator is $A^0_e(x) = \sum_{i\ even} A_i x^i = \sum_{i\ even} A^0_i x^i + \sum_{i\ even} A^1_i x^i$. This polynomial is equal to the previous one iff $A^1_i = A^0_{i-1}$ for all even weights i. *QED.*

Result 3.3 *extended primitive BCH codes*
Any extended $(N+1, K)$ binary primitive BCH code is isotropic.

Proof. Let $C(N, K)$ be a binary BCH code with primitive length. As a linear cyclic code, C is isotropic. Consider a codeword $c(x) = \sum_{\ell=0}^{N-1} c_\ell x^\ell \in C$ of even Hamming weight i. By definition of the BCH construction, the generator polynomial of C contains a primitive factor, *i.e.*, $c(\alpha) = 0$ where α is a primitive element of the finite field $GF(N+1)$. Let us focus on the conditional weight enumerator of the first bit c_0. Take $c(x)$ of even weight i having $c_0 = 1$. Thus, there exist $i-1$ integers $\ell_1, \ell_2, \ldots, \ell_{i-1}$ such that $1 + \alpha^{\ell_1} + \ldots + \alpha^{\ell_{i-1}} = 0$. This codeword is easily converted into $i-1$ distinct codewords of weight $i-1$ with $c_0 = 0$ by writing $1 + \alpha^{\ell_j} = \alpha^{\ell'_j}$, $j = 1 \ldots i-1$. The inverse operation is similar, it toggles c_0 and transforms a weight $i-1$ codeword into $i-1$ distinct codewords of weight i. This implies that $A_i^1 = A_{i-1}^0$. According to lemma 1, the extended BCH code $C_e(n+1, k)$ is isotropic. QED.

Result 3.4 *convolutional codes*
The study of the isotropic property of convolutional codes is less formal. For an infinite length convolutional code of rate $R = K/N$, the coded bit at a fixed position j, $j = 1 \ldots N$, has the same a posteriori distribution on all trellis branches. This is due to the branchwise cyclic structure of the convolutional code. Two bits belonging to the same trellis transition may not have the same APP distribution. We conclude that the anisotropy degree of a convolutional code is upper bounded by N.

Result 3.5 *turbo codes*
In the special case of parallel concatenated convolutional (Turbo) codes, the constituent codes are linked via their information bits only. This guarantees the isotropy of the APP propagation in the whole turbo code. However, an irregular turbo code is anisotropic. The degree of anisotropy $d_a(C)$ is equal to the number of non-zero fractions. For example, if we have non-zero fractions for degrees equal to $d = 2$ and $d = 10$, two APP distributions must be evaluated.

3.4 Partial APP and log-ratios evaluation

The a posteriori probability $APP(c_j)$ of a coded bit c_j is the product of the channel observation $p(y_j|c_j)$, the a priori probability $\pi(c_j)$ and the extrinsic information $Ext(c_j)$. A useful parameter for the study of density propagation is the *partial APP* which is the product of the channel observation and the a priori probability

$$\mathcal{A}(c_j) = p(y_j|c_j) \times \pi(c_j) \tag{3.3}$$

Instead of dealing directly with APP properties, it is convenient to use the following *log-ratios* (LR) in order to avoid the evaluation of the proportionality coefficient:

- Received log-likelihood ratio

$$\mathcal{B}_0(j) = LLR_0(j) = log\frac{p(y_j|c_j=1)}{p(y_j|c_j=0)} = \frac{-2}{N_0}y_j \tag{3.4}$$

- Extrinsic log-likelihood ratio

$$extLLR_m(j) = log\frac{Ext_m(c_j = 1)}{Ext_m(c_j = 0)} \tag{3.5}$$

- Partial a posteriori log-likelihood ratio

$$\mathcal{B}_m(j) = log\frac{\mathcal{A}_m(c_j = 1)}{\mathcal{A}_m(c_j = 0)} \tag{3.6}$$

- Total a posteriori log-likelihood ratio

$$LLR_m(j) = log\frac{APP_m(c_j = 1)}{APP_m(c_j = 0)} = \mathcal{B}_m(j) + extLLR_m(j) \tag{3.7}$$

In the case of symbol-based-iterative decoding, the logarithmic ratios are gathered into vectors of $Q-1$ components:

- Received LR : $\mathbf{LR_0(j)} = (LR_0^1(j), \ldots, LR_0^q(j), \ldots LR_0^{Q-1}(j))$

$$LR_0^q(j) = \log\frac{p(Y_{s_j}|S_j = q)}{p(Y_{s_j}|S_j = 0)}, \quad q = 1, 2, \ldots, Q-1 \tag{3.8}$$

- Extrinsic LR : $\mathbf{extLR(j)} = (extLR^1(j), \ldots, extLR^q(j), \ldots extLR^{Q-1}(j))$

$$extLR^q(j) = \log\frac{Ext(S_j = q)}{Ext(c_j = 0)}, \quad q = 1, 2, \ldots, Q-1 \tag{3.9}$$

- Partial APP LR : $\mathcal{B}(\mathbf{j}) = (\mathcal{B}^1(j), \ldots, \mathcal{B}^q(j), \ldots \mathcal{B}^{Q-1}(j))$

$$\mathcal{B}^q(j) = \log\frac{\mathcal{A}(S_j = q)}{\mathcal{A}(S_j = 0)}, \quad q = 1, 2, \ldots, Q-1 \tag{3.10}$$

- Total a posteriori LR : $\mathbf{LR(j)} = \mathcal{B}(\mathbf{j}) + \mathbf{extLR(j)}$

$$LR^q(j) = \log\frac{APP(S_j = q)}{APP(S_j = 0)} = \mathcal{B}^q(j) + extLR^q(j), \quad q = 1, 2, \ldots, Q-1 \tag{3.11}$$

3.5 Log-ration density propagation

Let us describe the information propagation in the graphical model of the concatenated code. The decoding iteration is denoted by m. The *degree* of a *bit* or *symbol node* is designated by d. The degree of a *subcode node* is equal to n. In the binary case, n is equal to the length N of the constituent code. The latter is assumed to be *isotropic* (all the symbols have the same degree d as for a regular turbo code). Consequently, the a posteriori information is independent of the bit position j.

Figure 3.1 shows the local neighborhood of a bit or a symbol in a cycle-free graph. Two different types of codes, based on a *bipartite Tanner graph*, are illustrated [81][47]. A (3,6) low-density parity check code [35][56] tree structure is first depicted. The bit nodes have degree 3 and the subcode nodes have degree 6. The *local constraint* in the subcode node is defined by the single parity check code (6,5,2). The total rate of the (3,6)-LDPC code is 1/2. The second tree show the graphical structure of a Turbo code based on two identical RSC codes with octal generators $g = (25, 37, 35)$. The local constraint is defined by a window of size W inside the subcode trellis. The turbo code rate is 1/2. Each symbol node represents the two information bits of a branch in the encoder trellis.

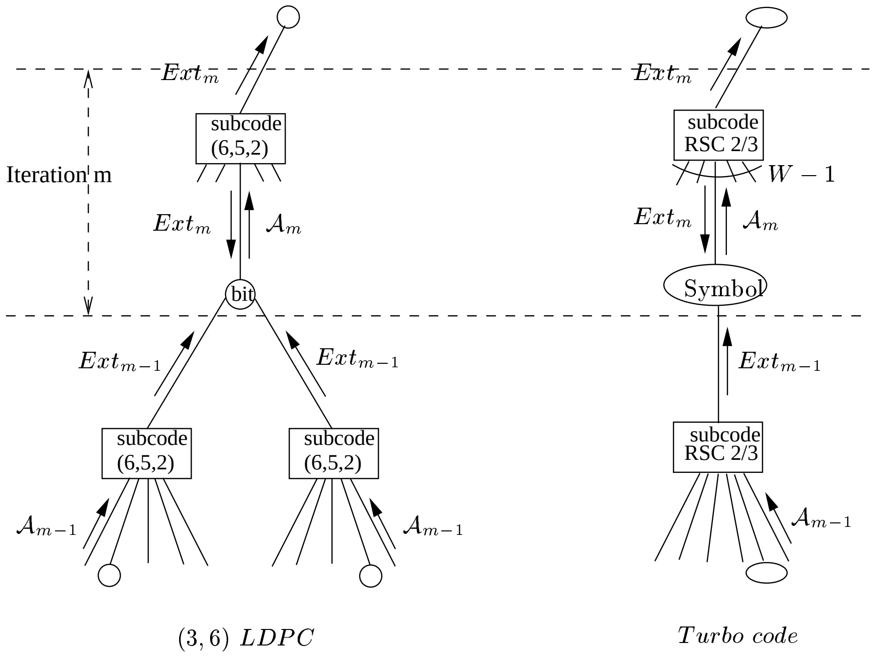


Figure 3.1: Propagation tree of concatenated codes.

Different propagation schedules are possible [47]. In a *flooding schedule*, all subcode nodes in the graph operate independently from each other and the decoding iteration is undefined. In the sequel, we assume that *information propagates upwards* in the graph. One decoding iteration includes two levels in the propagation tree. The first level contains subcode nodes and the second one contains only symbol nodes, *i.e.*, one iteration is equivalent to a round trip in the bipartite graphical representation. Let us now study the information evaluated by subcode and symbol nodes.

a) Subcode node

According to equation (2.14), one subcode node computes an extrinsic information from its $n-1$ inputs (see figure 3.1). If the local constraint of the code is equal to W, then $W-1$ inputs are considered.

$$Ext_{m-1}(S_j) = \sum_{c\in C|S_j} \prod_{\ell=1,\ell\neq j}^{W} p(Y_{s_\ell}|S_\ell)\pi_{m-1}(S_\ell) \tag{3.12}$$

where $\pi_{m-1}(S_\ell)$ is the a priori probability of symbol S_j at iteration $m-1$. We obviously considered that the the channel observations and the a priori informations are independent. The observations are independent since the transmission is done over a discrete memoryless channel. The independence of the a priori probability is due to the infinite size of the interleaver which insures that the graph is cycle free.

b) Symbol node

The partial a posteriori probability evaluated by a symbol node S_j in the graph at iteration m is given by

$$\mathcal{A}_m(S_j) = p(Y_{s_j}|c_j)\pi_m(S_j) \tag{3.13}$$

where the a priori probability $\pi_m(S_j)$ is the product of $d-1$ independent extrinsic informations supplied by the other $d-1$ subcode neighbors at the previous iteration.

$$\mathcal{A}_m(S_j) = p(Y_{s_j}|c_j) \prod_{\ell=1,\ell\neq j}^{d} Ext_{m-1}(S_j)_\ell \tag{3.14}$$

$Ext_{m-1}(S_j)_\ell$ is one of the d extrinsic informations evaluated by the subcode nodes at iteration $m-1$.

The a posteriori probability $APP_m(S_j)$ is the combination of the two opposite information streams on the same edge

$$APP_m(S_j) = \mathcal{A}_m(S_j)Ext_m(S_j) \tag{3.15}$$

Strictly speaking, the total APP is the product of d extrinsic informations and the initial channel observation.

Let $\mathcal{B}_\mathbf{m}(\mathbf{j})$ denotes the vector of partial log-ratios at iteration m, where $\mathcal{B}_m^q(j)$ is equal to

$$\mathcal{B}_m^q(j) = \log \frac{\mathcal{A}_m(S_j = q)}{\mathcal{A}_m(S_j = 0)}, \quad q = 1, 2, \ldots, Q-1 \tag{3.16}$$

For $m = 0$ the partial APP log-ratio is equal to the received log-ratio

$$\mathcal{B}_0^q(j) = LR_0^q(j) = \log \frac{p(Y_{s_j}|S_j = q)}{p(Y_{s_j}|S_j = 0)}, \quad q = 1, 2, \ldots, Q-1 \tag{3.17}$$

The extrinsic log-ratio $\mathbf{extLR_m(j)}$ is computed by a subcode node from the partial a posteriori log-ratios $\mathcal{B}_\mathbf{m}(\ell)$, $\ell \neq j$. From [95], we have

$$extLR_m^q(j) = \log \frac{Ext_m(S_j = q)}{Ext_m(S_j = 0)} = \log \frac{\sum_{i=1}^{n} A_{ij}^q \otimes [\exp(\mathcal{B}_m^q(j))]^{\otimes i-1}}{\sum_{i=0}^{n-1} A_{ij}^0 \otimes [\exp(\mathcal{B}_m^q(j))]^{\otimes i}}, \quad q = 1, 2, \ldots, Q-1 \tag{3.18}$$

A_{ij}^q and A_{ij}^0 are the coefficients of the conditional weight enumerators, and $\otimes$ represents the sum or the product of two id random variables.

By combining equations (3.16), (3.17) and (3.18) and omitting the symbol position j, thanks to the isotropy property, we get the *density propagation formula* that describes the propagation of the partial APP LR

$$\mathcal{B}_m^q = \mathcal{B}_0^q + (d-1) \otimes \left[\log \frac{\sum_{i=1}^{n} A_i^q \otimes [\exp(\mathcal{B}_{m-1}^q)]^{\otimes i-1}}{\sum_{i=0}^{n-1} A_i^0 \otimes [\exp(\mathcal{B}_{m-1}^q)]^{\otimes i}} \right], \quad q = 1, 2, \dots, Q-1 \quad (3.19)$$

This formula explains how the outgoing probability distribution of $\mathcal{B}_\mathbf{m}$ is linked to the incoming distribution of $\mathcal{B}_\mathbf{m-1}$. Indeed, the probability density function of $\mathcal{B}_\mathbf{m}$ is equal to the convolution of the received $\mathbf{LR_0}$ density with $d-1$ identical densities related to the distribution of $\mathcal{B}_\mathbf{m-1}$

$$\mathcal{B}_\mathbf{m} = \mathbf{LR_0} + (d-1) \otimes \mathbf{extLR_{m-1}} \quad (3.20)$$

Furthermore, the total a posteriori distribution is equal to the convolution of the $\mathcal{B}_\mathbf{m}$ density and the new extrinsic $\mathbf{extLR_m}$ density since

$$\mathbf{LR_m} = \mathcal{B}_\mathbf{m} + \mathbf{extLR_m}. \quad (3.21)$$

As an example, figures 3.2 and 3.3 show the density propagation of the extrinsic information and the total APP. For large number of iterations, the probability density function of the extrinsic information can be approximated by a Gaussian. This is also true for the total APP even for small values of m.

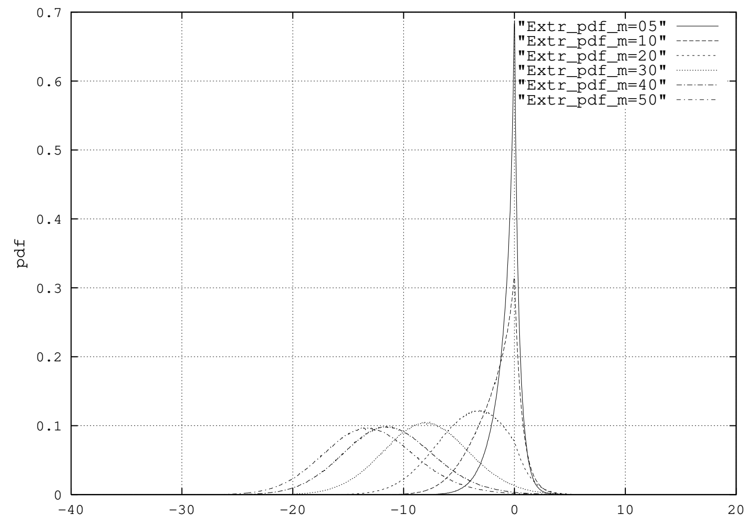


Figure 3.2: Probability density of $extLR_m$. $Q = 2$. $m = 5, 10, 20, 30, 40, 50$ iterations.

The performance limit of the soft-input soft-output iterative decoder when applied to a compound code is given by the minimal achievable SNR, *i.e.*, the minimal

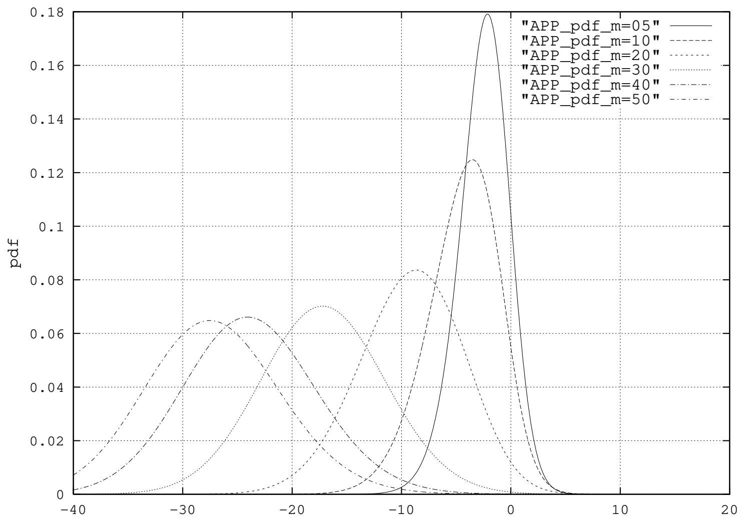


Figure 3.3: Probability density of LR_m. $Q = 2$. $m = 5,\ 10,\ 20,\ 30,\ 40,\ 50$ iterations.

value of the signal-to-noise ratio E_b/N_0 for which the symbol error probability at iteration m, Pe_m, tends to 0 when m goes to infinity. Pe_m can be calculated using the probability density function of $\mathbf{LR_m}$ denoted by $p_m(x_1, \ldots, x_q \ldots x_{Q-1})$. Assuming that the all zero codeword is transmitted, we get

$$Pe_m = \int_0^{+\infty} p_m(x_1, \ldots, x_q \ldots x_{Q-1}).dx_1 \ldots dx_q \ldots dx_{Q-1} \tag{3.22}$$

Hence, the *threshold* value of E_b/N_0 defining the limit of iterative decoding can be determined using equation (3.25), where the distribution $p_m(x_1, \ldots, x_q \ldots x_{Q-1})$ is the convolution of the densities of $\mathcal{B}_\mathbf{m}$ and $\mathbf{extLR_m}$. At each iteration, these two densities can be evaluated by Monte Carlo simulation.

Figure 3.4 show the number of iterations needed to achieve a symbol error probability $P_{e_m} < 10^{-5}$ as a function of the average signal-to-noise ratio per bit E_b/N_0 in dB. A rate-1/2 turbo code, constructed using two rate-2/3 RSC codes with octal generators $g = (37, 21, 27)$ is considered. Symbol-based iterative decoding is done. The threshold on the convergence of the iterative decoder is equal to $0.42dB$. Notice that in practice, a $P_{e_m} < 10^{-5}$ can be regarded as equal to zero.

c) Density propagation in the case of an irregular turbo code

We have previously considered that the code is *isotropic*. All the symbols have the same degree d. For an irregular turbo code with a degree profile $f_d,\ d = 2, \ldots, d_{max}$, different LR densities must be evaluated. From equations (3.20) and (3.21), we get

$$\mathcal{B}_\mathbf{dm} = \mathbf{LR_0} + (d-1) \otimes \mathbf{extLR_{m-1}} \qquad d = 2, \ldots, d_{max} \tag{3.23}$$

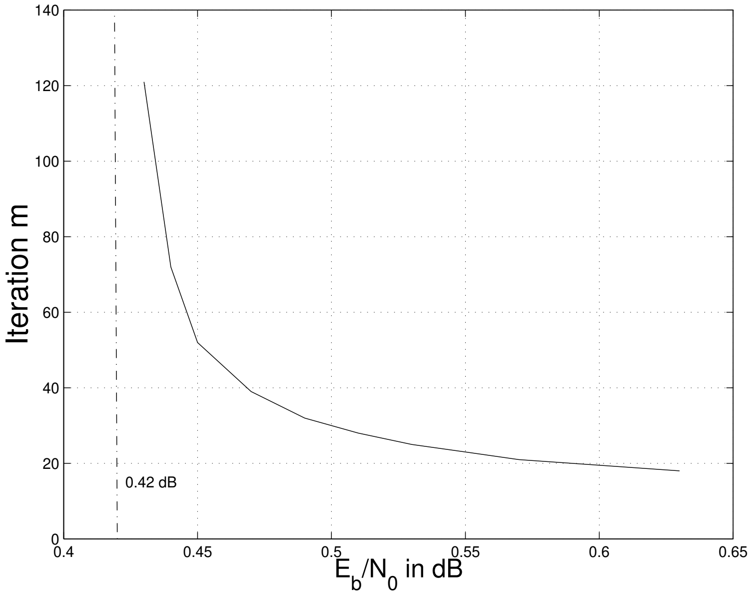


Figure 3.4: Number of iterations needed to achieve $P_{e_m} < 10^{-5}$. Rate-1/2 turbo code constructed using two rate-2/3 RSC codes with octal generators $g = (37, 21, 27)$

$$\mathbf{LR_{dm}} = \mathcal{B}_{\mathbf{dm}} + \mathbf{extLR_m} \qquad d = 2, \ldots, d_{max}. \tag{3.24}$$

where $\mathcal{B}_{\mathbf{dm}}$ and $\mathbf{LR_{dm}}$ are the partial and total APP LRs of the symbols of degree d.

The symbol error probability at iteration m is equal to

$$Pe_m = \sum_{d=2}^{d_{max}} f_d. \int_0^{+\infty} p_{dm}(x_1, \ldots, x_q \ldots x_{Q-1}).dx_1 \ldots dx_q \ldots dx_{Q-1} \tag{3.25}$$

In the next section, the threshold value of E_b/N_0 is determined by calculating the transfer functions of both symbol nodes and subcode nodes.

3.6 The transfer Function Method

The *transfer function* gives a qualitative overview of the information propagation. It consists in predicting the behavior of the iterative decoding by observing the propagation at the symbol nodes and subcode nodes. Different methods were discussed in the literature [86] [24]; however, we just describe the transfer function method based on a *Gaussian approximation* with *error probability matching* (EPM-GA) [74].

As it is shown on the graphical representation in figure 3.1, one decoding iteration is composed of a subcode level processing, and afterwards of a symbol node level processing. Consequently, different error probabilities propagate through the graph, depending whether it is a symbol level or a subcode level. The transfer function of a symbol node (respectively of a subcode node) gives the relation existing between the error probability at the input and the output of a symbol node (respectively of a subcode node).

$$\textit{Subcode node} \; : \; Pe_{out} = H(Pe_{in})$$

$$\textit{Symbol node} \; : \; Pe_{out} = G(Pe_{in})$$

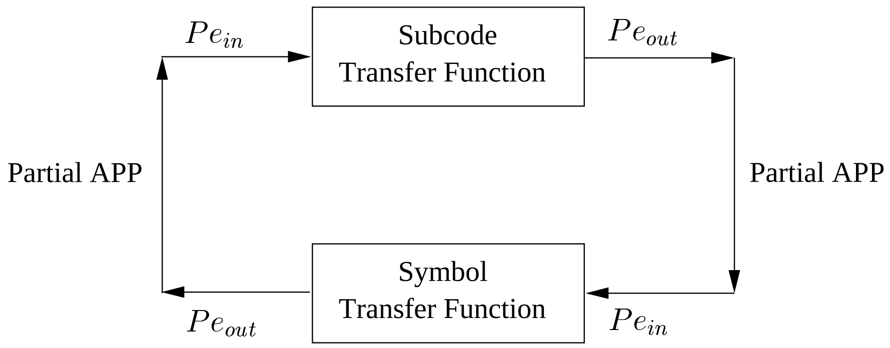


Figure 3.5: Transfer function Method

Notice that the output error probability of a symbol node (respectively of a subcode node) is the input error probability of the subcode node (respectively of a symbol node). To understand the behavior of an iterative decoder, the transfer functions of the symbol and subcode nodes are drawn on the same diagram. The error probability propagates through iterations according to figure 3.6. A steady state is reached at the fixed point defined by

$$H(Pe) = G^{-1}(Pe). \tag{3.26}$$

If Pe is not equal to 0, the iterative decoder does not produce the right codeword. The largest noise variance N_0, such as the fixed point Pe is equal to 0 (as in figure 3.6), defines the performance limit of a turbo code under iterative decoding.

In order to evaluate these transfer functions, we first use the result on the joint convergence of the total and the partial APP LR over the AWGN channel [95]. This result is of a great interest since it is the partial APP LR that propagates through the graph of the code.

3.6.1 Joint convergence of total and partial LLR

The all zero codeword is transmitted. For the AWGN channel, the two following statements are equivalent

- $\forall \epsilon > 0, \exists \; m_0$ such that $\forall m > m_0$, $P(LLR_m > 0) < \epsilon$.
- $\forall \epsilon > 0, \exists \; m_0$ such that $\forall m > m_0$, $P(\mathcal{B}_m > 0) < \epsilon$.

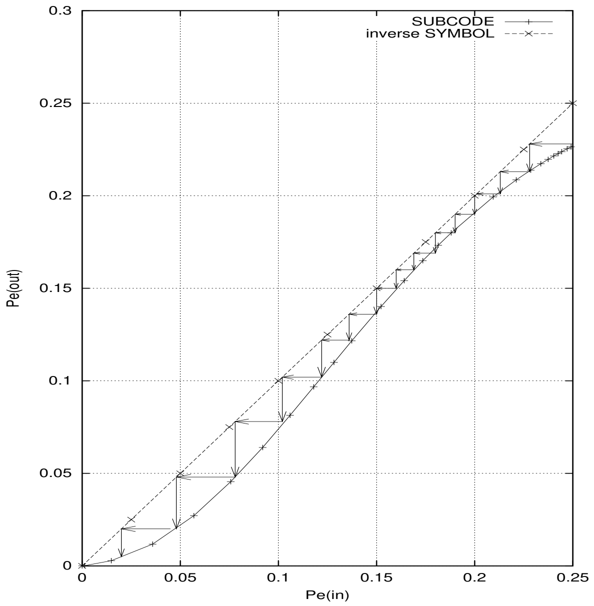


Figure 3.6: Transfer functions H and G^{-1}. Quaternary turbo code constructed using two rate-2/3 RSC codes with octal generators $g = (31, 27, 37)$. $Eb/N0 = 0.65$ dB.

The proof of the above result is done using classical probabilistic arguments [96] (cf. Appendix A). We conclude that the true bit error probability Pe_m tends to zero iff the partial error probability computed from the distribution of $\mathcal{B}_m$ also tends to zero when m increases[2]. Hence, the threshold value of E_b/N_0 defining the limit of iterative decoding can be determined from the transfer functions of a symbol node and a bit node based on the partial error probability.

Furthermore, an approximation must be made in order to evaluate the partial error probabilities at the input and the output of the symbol and the subcode node.

3.6.2 Gaussian approximation

Let us consider a turbo code based on d identical isotropic constituent codes. The codeword $\mathbf{c} = (0, 0, \dots, 0)$ is transmitted over the AWGN channel using a BPSK modulation. The real additive channel noise is Gaussian with zero mean and variance N_0. Hence, the received real values y_j at the channel output are Gaussian random variables with mean -1 and variance N_0. From equation (3.4), we get that the received log-likelihood ratios $LR_0(j)$ are Gaussian random variables with mean $-\frac{2}{N_0}$ and variance $\frac{4}{N_0}$.

Without loss of generality, we consider the case of quaternary symbols $\{00, 01, 10, 11\}$ $(Q = 4)$. Due to the independence of the channel observations $p(y_j|c_j)$, $LR_0^1(j) =$

[2]For symbol-based iterative decoding, $\mathcal{B}_m > 0$ is equivalent to $\mathcal{B}_m^q > 0,\ \forall q = 1, \dots, Q$

$\log \frac{p(Y_{s_j}|S_j=1)}{p(Y_{s_j}|S_j=0)}$ and $LR_0^2(j) = \log \frac{p(Y_{s_j}|S_j=2)}{p(Y_{s_j}|S_j=0)}$ related respectively to the symbols 01 and 10, are independent Gaussian variables with mean $-\frac{2}{N_0}$ and variance $\frac{4}{N_0}$. $LR_0^3(j) = \log \frac{p(Y_{s_j}|S_j=3)}{p(Y_{s_j}|S_j=0)}$ can thus be written as the sum of $LR_0^1(j)$ and $LR_0^2(j)$ and is also a Gaussian variable. We assume in the sequel that a priori informations on symbols have Gaussian distributions at the input and the output of the SISO decoder. The mean and the variance of these variables are respectively equal to $-\frac{2}{N_0(apriori}$ and $\frac{4}{N_0(apriori)}$, where $N_0(apriori)$ is the parameter associated to the a priori distribution. This assumption is all the more satisfied as the number of iterations m increases (see figure 3.2).

a) Transfer function of a subcode node

At the input of the subcode node, the error probability is defined by

$$\begin{aligned} Pe_{in}(\text{subcode node}) &= \text{Prob}\left(\mathcal{B}^1 > 0 \text{ or } \mathcal{B}^2 > 0 \text{ or } \mathcal{B}^3 > 0\right) \\ &= 1 - \text{Prob}\left(\mathcal{B}^1 < 0, \mathcal{B}^2 < 0, \mathcal{B}^3 < 0\right) \end{aligned}$$

The partial APP LRs $\mathcal{B}^i$, $i = 1, 2, 3$, are the sum of the channel observations and $(d-1)$ a priori informations which are assumed to be Gaussian. Therefore, the partial APP LRs are also presumed to be Gaussian random variables, but they are not independent. Their mean is $m = E[\mathcal{B}]$ and their covariance matrix is $\Gamma = [E[\mathcal{B}_c^i \mathcal{B}_c^j]]$, where $E[\ ]$ is the mathematical expectation, and B_c^i is the partial APP LR of the symbol centered around its mean value. From the probability density of $\mathcal{B}$

$$p(\mathcal{B}) = \frac{1}{\sqrt{2\pi}^3 \sqrt{det(\Gamma)}} \exp\left(-\frac{1}{2}(\mathcal{B} - m)\,\Gamma^{-1}(\mathcal{B} - m)^t\right) \tag{3.27}$$

we get the input error probability

$$Pe_{in}(\text{subcode node}) = 1 - \int_{\mathcal{B}^1<0, \mathcal{B}^2<0, \mathcal{B}^3<0} p(\mathcal{B}) d\mathcal{B}^1 d\mathcal{B}^2 d\mathcal{B}^3 \tag{3.28}$$

The integral in equation (3.28) is very difficult to evaluate. It is necessary to introduce some assumptions concerning the LRs independence[3].

The partial APP LR $\mathcal{B}^1$ of the first symbol 01 and the partial APP LR $\mathcal{B}^2$ of the second symbol 10 are assumed to be independent, as for LR_0^1 and LR_0^2. Notice that the mean and the variance of $\mathcal{B}^1$ and $\mathcal{B}^2$ are equal to

$$m_{\mathcal{B}^1} = m_{\mathcal{B}^2} = -\frac{2}{N_0} - \frac{2(d-1)}{N_0(apriori)} \tag{3.29}$$

$$\sigma^2_{\mathcal{B}^1} = \sigma^2_{\mathcal{B}^2} = \frac{4}{N_0} + \frac{4(d-1)}{N_0(apriori)} \tag{3.30}$$

[3]In the binary case, there's only one variable. The exact value of the integral can be determined without introducing any assumption.

The partial APP of the third symbol, 11, can subsequently be written as the sum of two independent Gaussian variables, *i.e.*, $\mathcal{B}^3 = \mathcal{B}^1 + \mathcal{B}^2$. The error probability (3.28) is simplified by

$$\begin{aligned} Pe_{in}(\text{subcode node}) &= 1 - \text{Prob}(\mathcal{B}^1 < 0)\text{Prob}(\mathcal{B}^2 < 0) \\ &= 1 - \left(1 - \frac{1}{2}\text{erfc}\left(\sqrt{\frac{1}{2N_0} + \frac{(d-1)}{2N_0(apriori)}}\right)\right)^2 \\ &\approx \text{erfc}\left(\sqrt{\frac{1}{2N_0} + \frac{(d-1)}{2N_0(apriori)}}\right) \end{aligned} \tag{3.31}$$

where erfc is the error function.

At the subcode output, the error probability is given by

$$Pe_{out}(\text{subcode node}) = \int_{\mathcal{B}^1>0 \text{ or } \mathcal{B}^2>0 \text{ or } \mathcal{B}^3>0} p(\mathcal{B})d\mathcal{B} \tag{3.32}$$

where the distribution of $\mathcal{B}$ is derived by SISO decoding (Monte-Carlo simulation).

b) Transfer function of a symbol node

The $d-1$ partial APP LRs at the input of each symbol node are defined in the same way as the sum of two independent Gaussian variables, *i.e.*, the channel observation with mean $-\frac{2}{N_0}$ and variance $\frac{4}{N_0}$, and an a priori information. As for the subcode transfer function, we assume that the partial APP LRs $\mathcal{B}^1$ and $\mathcal{B}^2$ at the symbol input are independent Gaussian variables with mean $-\frac{2}{N_0} - \frac{2}{N_0(apriori)}$ and variance $\frac{4}{N_0} + \frac{4}{N_0(apriori)}$. The resulting error probability at the input of a symbol node is

$$Pe_{in}(\text{symbol node}) \approx \text{erfc}\left(\sqrt{\frac{1}{2N_0} + \frac{1}{2N_0(apriori)}}\right) \tag{3.33}$$

The symbol node computes the new partial APP LR, by summing the channel observation with the $d-1$ independent a priori informations provided by the other codes. The output partial APP LRs, $\mathcal{B}^1$ and $\mathcal{B}^2$, are Gaussian with mean $-\frac{2}{N_0} - \frac{2(d-1)}{N_0(apriori)}$ and variance $\frac{4}{N_0} + \frac{4(d-1)}{N_0(apriori)}$. The corresponding error probability at the output of the symbol node can be written as

$$Pe_{out}(\text{symbol node}) \approx \text{erfc}\left(\sqrt{\frac{1}{2N_0} + \frac{d-1}{2N_0(apriori)}}\right) \tag{3.34}$$

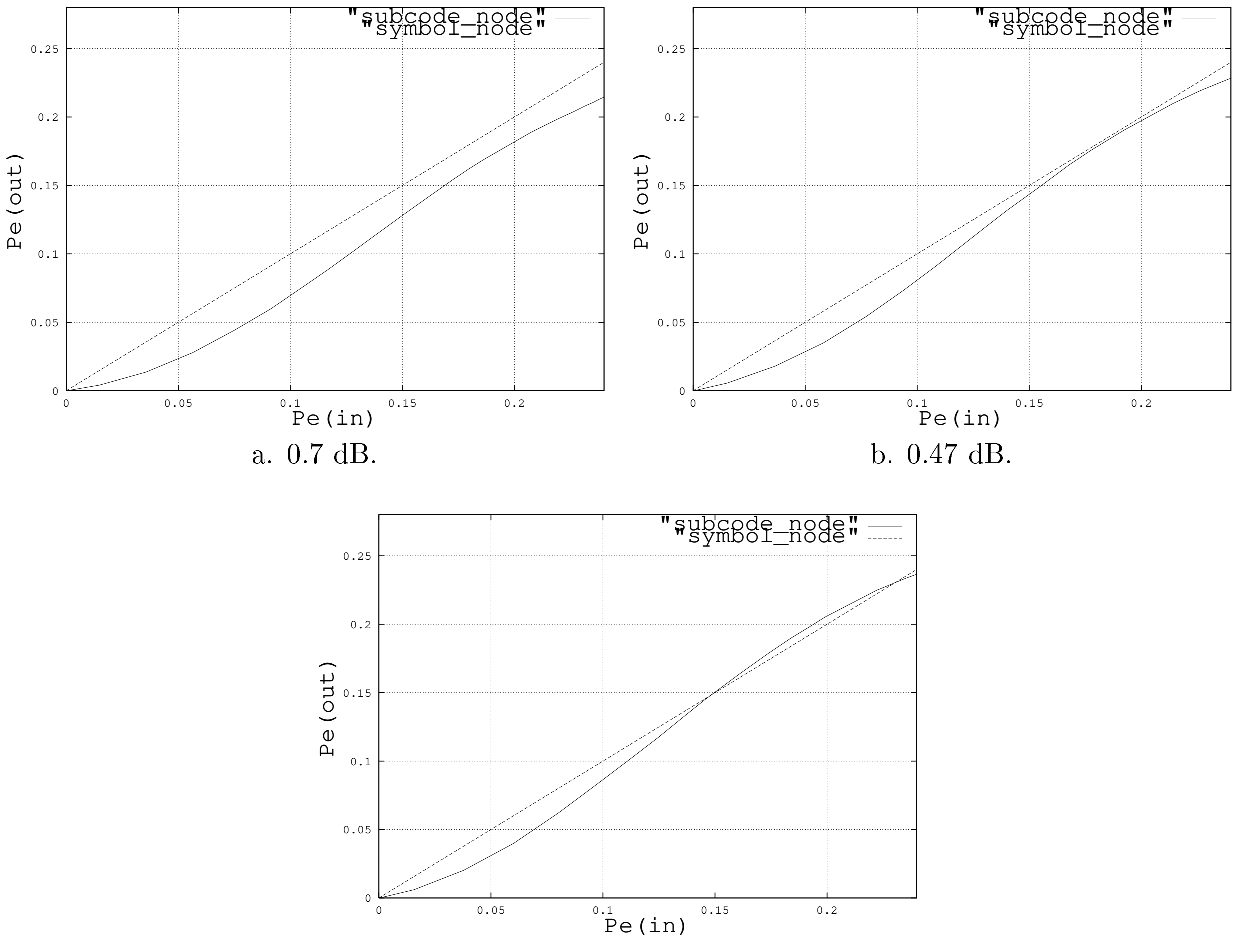
3.6.3 Transfer function of a regular turbo code

Let us consider a regular turbo code obtained from the parallel concatenation of two RSC constituent codes. All the information symbols have the same degree $d = 2$. For quaternary symbol-based iterative decoding, we have

$$Pe_{in}(\text{symbol node}) = Pe_{out}(\text{symbol node}) \approx \text{erfc}\left(\sqrt{\frac{1}{2N_0} + \frac{1}{2N_0(apriori)}}\right) \tag{3.35}$$

$$Pe_{in}(\text{subcode node}) \approx \text{erfc}\left(\sqrt{\frac{1}{2N_0} + \frac{1}{2N_0(apriori)}}\right) \tag{3.36}$$

$$Pe_{out}(\text{subcode node}) = \int_{\mathcal{B}^1>0 \text{ or } \mathcal{B}^2>0 \text{ or } \mathcal{B}^3>0} p(\mathcal{B})d\mathcal{B} \tag{3.37}$$



a. 0.7 dB. b. 0.47 dB.

c. 0.3 dB.

Figure 3.7: Transfer functions H and G^{-1}. Quaternary turbo code. Octal generators $g = (37, 21, 27)$.

In the case of bit based iterative decoding, similar results can be determined. The exact value of the subcode node input probability can be evaluated using equation (3.31). We get

$$Pe_{in}(\text{subcode node}) = \frac{1}{2}\text{erfc}\left(\sqrt{\frac{1}{2N_0} + \frac{1}{2N_0(apriori)}}\right) \tag{3.38}$$

$$Pe_{out}(\text{subcode node}) = \int_{\mathcal{B}>0} p(\mathcal{B})d\mathcal{B} \tag{3.39}$$

$$Pe_{in}(\text{bit node}) = Pe_{out}(\text{bit node}) = \frac{1}{2}\text{erfc}\left(\sqrt{\frac{1}{2N_0} + \frac{1}{2N_0(apriori)}}\right) \tag{3.40}$$

Figure 3.7 shows the transfer functions H and G^{-1} of a rate -1/2 turbo code obtained from the parallel concatenation of two rate-2/3 RSC codes with octal generators $g = (37, 21, 27)$. Symbol-based iterative decoding is applied. For an average signal-to-noise ratio per bit $E_b/N_0 = 0.7$ dB (see figure 3.7.a.), the fixed point is equal to zero. In fact, 0.7 dB is largely greater than the threshold of the code, which is a little smaller than 0.47 dB (see figure 3.7.b.). For $E_b/N_0 = 0.3$ dB, there is an intersection between the two curves. The fixed point is $Pe \neq 0$; therefore no convergence can be reached.

3.6.4 Transfer function of an irregular turbo code

Let us consider an irregular turbo code with degree profile f_d, $d = 2, \ldots, d_{max}$. Symbol-based iterative decoding is done. The input error probability of a symbol node is identical to the one we have for an isotropic code. The input or the output error probability of a subcode node and the output error probability of a symbol node are defined by

$$Pe = \sum_{d=2}^{d_{max}} f_d.Pe^d \tag{3.41}$$

where Pe^d is the corresponding input or output error probability. From equation (3.31), (3.32), (3.33) and (3.34), we get

$$Pe_{in}(\text{subcode node}) \approx \sum_{d=2}^{d_{max}} f_d.\text{erfc}\left(\sqrt{\frac{1}{2N_0} + \frac{(d-1)}{2N_0(apriori)}}\right) \tag{3.42}$$

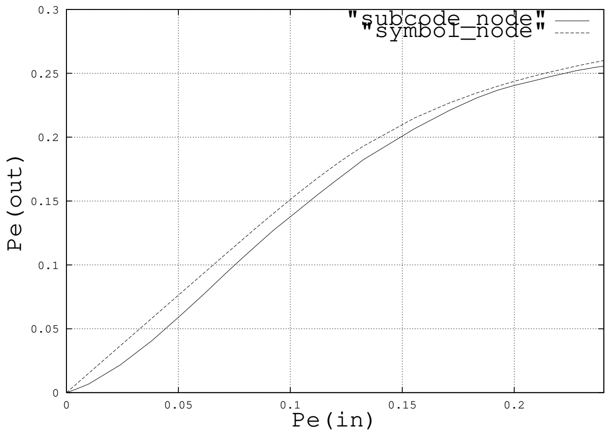
$$Pe_{out}(\text{subcode node}) = \sum_{d=2}^{d_{max}} f_d. \int_{\mathcal{B}_d^1>0 \text{ or } \mathcal{B}_d^2>0 \text{ or } \mathcal{B}_d^3>0} p(\mathcal{B}_\mathbf{d})d\mathcal{B}_\mathbf{d} \tag{3.43}$$

$$Pe_{in}(\text{symbol node}) \approx \text{erfc}\left(\sqrt{\frac{1}{2N_0} + \frac{1}{2N_0(apriori)}}\right) \tag{3.44}$$

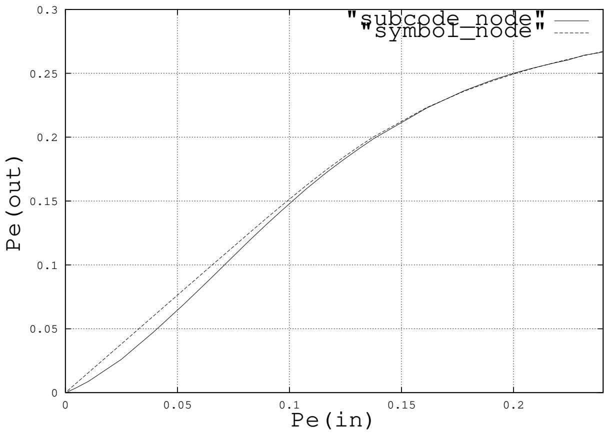
$$Pe_{out}(\text{symbol node}) \approx \sum_{d=2}^{d_{max}} f_d.\text{erfc}\left(\sqrt{\frac{1}{2N_0} + \frac{d-1}{2N_0(apriori)}}\right) \tag{3.45}$$

where $\mathcal{B}_d$ is the vector of the partial LRs of the symbols of degree d. Similar results can be found when bit-based iterative decoding is applied.

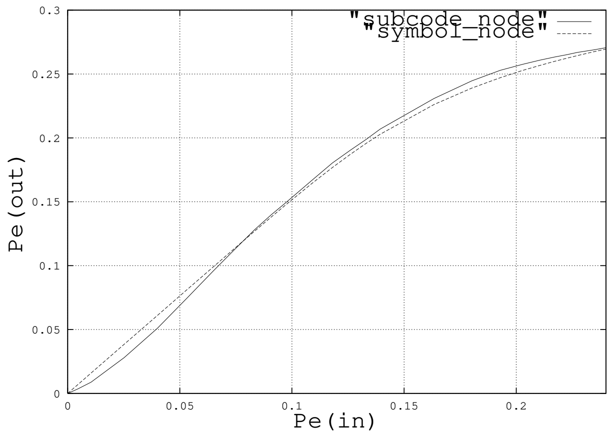
As for turbo codes, in figure 3.8, we represent the transfer functions H and G^{-1} of a rate-1/2 irregular turbo code constructed using two rate-2/3 RSC codes with octal generators $g = (37, 21, 27)$. The degree profile of the quaternary information symbols is $f_2 = 0.905$ and $f_{10} = 0.095$.



a. 0.5 dB.



b. 0.28 dB.



c. 0.19 dB.

Figure 3.8: Transfer functions H and G^{-1}. Quaternary irregular turbo code. Octal generators $g = (37, 21, 27)$.

If we compare the transfer functions of the regular and the irregular turbo codes represented in figures 3.7 and 3.8, we conclude that the number of iterations needed by a turbo code to achieve convergence is significantly smaller than the one we have for an irregular turbo code. In fact, for the regular turbo code, once we get out of the critical zone ($0.15 < Pe(in) < 0.2$), convergence is fast. For the irregular turbo code, the two curves are almost parallel. This explains why we did 100 decoding iterations for the finite length irregular turbo studied in the previous chapter instead of 20 iterations as for the regular turbo code.

Nevertheless, the advantage of irregular turbo codes is that better performance can be reached for both finite and infinite code length.

To validate the transfer function method under the simplified independence assumptions, promising numerical results are shown in the next section. The results are compared to the exact estimates obtained using the APP propagation of section 3.5.

3.7 Numerical results

First, let us consider a turbo code built from the parallel concatenation of two recursive systematic convolutional codes of rate 2/3. The turbo code rate is equal to 1/2. The minimal signal-to-noise ratio that achieves the capacity of the AWGN channel with BPSK input is $\frac{E_b}{N_0} = 0.187$ dB. The degree of symbol nodes is $d = 2$. Computer simulations using the forward-backward algorithm [2] showed that a window size $W = 32 \times \nu + 1$, which is 16 times the code memory around the central symbol, is sufficient to take into account the local constraint of the convolutional code.
Table 3.1 shows the minimal signal-to-noise ratio achievable by an iterative decoder[4] and an infinite length interleaver, determined using first, the density propagation formula and second, the transfer function method. Different octal generators are considered for both bit nodes based turbo code and symbol nodes based turbo code. We also consider the standard rate-1/2 turbo code proposed by Berrou (two RSC codes with octal generators $g = (37, 21)$).

Turbo Code	Binary	Quaternary	
Generators in octal	Exact APP propagation	Exact APP propagation	EPM-Gaussian approximation
(31, 21, 37)	0.64	0.48	0.51
(37, 21, 27)	0.60	0.42	0.46
(25, 37, 35)	0.55	0.31	0.40
(37, 21)	0.5	-	-

Table 3.1: Minimal E_b/N_0 in dB, achievable by an iterative decoder. Rate-1/2 turbo code.

We conclude from these results that the approximate values of the minimal signal-to-noise ratio determined using the transfer function method are close to the exact ones determined using the density propagation formula. This validates the Gaussian approximation introduced in section 3.6. Moreover, symbol-based turbo codes

[4]The bit error probability goes to zero for signal-to-noise ratios higher than the values specified in tables 3.1, 3.2 and 3.3

perform better than bit-based turbo codes. For octal generators $g = (25, 37, 35)$, the minimum signal-to-noise ratio achievable using quaternary symbols and iterative decoding is equal to 0.31 dB which is 0.13 dB from the capacity of the AWGN channel with a BPSK modulation.

Better performance is reached by introducing an irregularity. For example, we consider the rate-1/2 irregular turbo code with octal genarators (37, 21, 27). Symbol-based iterative decoding is applied. Results are presented in table 3.2 for different degree profiles with two non zero fractions. The minimum signal-to-noise ratio is equal to 0.21 dB which is within 0.03 dB of the capacity. Notice, that the optimal degree profile ($f_2 = 0.915,\ f_{11} = 0.085$), obtained using the Gaussian approximation is different from the optimal degree profile found using the exact APP propagation ($f_2 = 0.905,\ f_{10} = 0.095$). However the parameters of the corresponding degree profiles are too close. Therefore, we can first search for a good degree profile using the Gaussian approximation, then find the best one using exact APP propagation.

Degree profile	Exact APP propagation	EPM-Gaussian approximation
$f_2 = 0.901,\ f_9 = 0.099$	0.24	0.34
$f_2 = 0.905,\ f_{10} = 0.095$	0.21	0.3
$f_2 = 0.915,\ f_{11} = 0.085$	0.22	0.27

Table 3.2: Minimal E_b/N_0 in dB, achievable by a symbol-based iterative decoder. Rate-1/2 turbo code. Rate-2/3 RSC code with octal generators $g = (37, 21, 27)$.

Finally, we consider a rate-1/3 irregular turbo code built from the parallel concatenation of two RSC codes of rate 1/2. The octal generators are $g = (37, 21)$. The minimal signal-to-noise ratio that achieves the capacity of the AWGN channel with BPSK input is $\frac{E_b}{N_0} = -0.495$ dB. We first search for the best degree profiles with two non zero fractions. The two best profiles contains "elite" bits with degree equal to 8 and 9 (see table 3.3). Then, we use the two best "elite" bits to construct an irregular turbo code with a degree profile that contains three non zero fractions. The best performance are found for fractions $f_2 = 0.888$, $f_8 = 0.06$ and $f_9 = 0.052$.

Degree profile	Exact APP propagation
$f_2 = 1$	-0.13
$f_2 = 0.861,\ f_7 = 0.139$	-0.35
$f_2 = 0.884,\ f_8 = 0.116$	-0.44
$f_2 = 0.891,\ f_9 = 0.109$	-0.41
$f_2 = 0.905,\ f_{10} = 0.095$	-0.39
$f_2 = 0.888,\ f_8 = 0.060,\ f_9 = 0.052$	-0.47

Table 3.3: Minimal E_b/N_0 in dB, achievable by a bit-based iterative decoder. Rate-1/2 turbo code. Rate-1/2 RSC code with octal generators $g = (37, 21)$.

The thershold is equal to -0.47 dB. This is within 0.03 dB of the capacity.

3.8 Conclusions

We studied in this chapter the asymptotic performance of turbo and irregular turbo codes using the log-density propagation method and the the transfer function method based on the Gaussian approximation with error probability matching (EPM-GA).

The transfer function method leads to accurate results. It can be used to speed up the search for good degree profiles. On the other hand, this method can be used to analyze the convergence of iterative decoding. From the transfer functions of regular and irregular turbo codes, we concluded that the convergence of iterative decoding is slower for an irregular turbo code.

The exact thresholds on the performance of an infinite length turbo or irregular turbo code under iterative decoding were determined using the log-density propagation method. For both, the rate-1/3 irregular turbo code with bit-based iterative decoding and the rate-1/2 irregular turbo code with symbol-based iterative decoding, the threshold is within 0.03 dB of the Shannon limit of the discret input AWGN channel with maximum-likelihood decoding.

Notice that, for the rate-1/2 irregular turbo code with symbol-based iterative decoding, we just considered a degree profile with only two non-zero fractions. Smaller thresholds can probably be reached using a larger degree profile. This is not necessary since that the 0.03 dB value can be considered as almost equal to zero.

Finally, we conclude from these results that, under iterative decoding, irregular turbo codes can achieve capacity of the BPSK-input AWGN channel as for irregular LDPC codes.

In the next chapter, we study multilevel coded modulations which can be used to approach the capacity of a M-ary disctrete input AWGN channel ($M > 2$)at high code rates.

Chapter 4

Multilevel coded modulations

4.1 Introduction

The performance improvement, in the case of binary codes, is achieved by expanding the bandwidth of the transmitted signal by an amount equal to the reciprocal of the code rate.

However, efficient digital communication systems have to be both power and bandwidth efficient, especially in the case of bandwidth-constrained channels. For such channels, the digital communication system is designed using M-ary modulations that have signaling rates $R > 1$ bit/dimension. Coding and modulation are combined. We get the so called *coded modulations*.

The most powerful applicable coded modulations systems are the well known *Trellis Coded Modulations* (TCM) that were first described by Ungerboeck in 1982 [90] (see also [91] and [92]), and *Multilevel Coded Modulations* (MLC) introduced by Imai and Hirakawa [46].

The common idea to these two methods, is to optimize the code in Euclidean space rather than dealing with Hamming distance. This is done by successively binary partitioning the signal set $A = \{a_1, a_2, \ldots, a_M\}$ of a M-ary modulation ($M = 2^L$) using Ungerboeck's set partitioning rule. After L levels of partitioning, each signal point will be mapped by a binary vector $\mathbf{x} = (x_L, x_{L-1}, \ldots, x_1)$. Ungerboeck's approach, that is based on maximizing the minimum intra-subset Euclidean distance, is the most set partitioning strategy used in practice.

The difference between TCM and MLC is in coding. In a TCM system, the least significant bits of $\mathbf{x}$ are encoded using one convolutional code and the most significant bits remain uncoded. On the other hand, in an MLC system each bit x_i of the signal point is individually protected by an individual binary code E_i.

Compared to Ungerboeck's TCM, the MLC approach has the advantage of providing flexible transmission rates since the latter becomes independent of the dimensionality of the signal constellation. Furthermore, any code like block codes, convolutional codes, turbo codes, etc, can be used as a component code.

However, MLC suffers of two major disadvantages that explain why MLC is not of great interest in practice. First, the performance is severely degraded due to high error rates at lower levels. Secondly, an MLC system is more complex than a TCM based-one.

In this chapter, we review the main results found in the literature, especially by Huber *et al* [98][45][26][99][27], regarding multilevel coding. We mainly focus on the capacity, the design rules and the labeling strategies of MLC. Then, asymmetric one-dimensional ASK and two-dimensional PSK constellations are studied. The main purpose of these studies is to determine, for the *additive white Gaussian noise channel* (AWGN), a multilevel coding scheme that has equal capacity at each level i. Therefore, applying the capacity design rule, the individual rates at each level become equal. Complexity of the system is largely reduced since only one architecture of encoding and decoding is sufficient.

The system model of an MLC scheme is introduced in section 4.2. The proof that MLC, combined to *multistage decoding* (MSD), can achieve the capacity is reviewed in section 4.3. In section 4.4 we determine the capacity region, *i.e.* the region corresponding to the rates that can be used to achieve capacity. The different rate design rules like the *balanced distance rule*, the *capacity design rule*, and the *coding exponent design rule*, are investigated in section 4.5. In section 4.6, we study the different labeling strategies of some practical interest like *block partitioning* strategy and the *Gray labeling* strategy. In these sections, We based our study on the case of a $M = 2^L$-ary ASK modulation (specially 8-ASK). To explain why all this study is based on a one-dimensional ASK, the dimensionality of the constituent signal constellation is reviewed in section 4.7. For this one-dimensional modulation, an asymmetric constellation with equal capacity at each level is introduced in section 4.8. The case of MLC systems based on asymmetric PSK constellations, which is of great interest in practice, is studied in section 4.9. Finally, section 4.10 resume the results of some Monte Carlo simulations obtained using MLC schemes based on ASK and PSK modulations.

4.2 System model

Let us consider a D-dimensional modulation with $M = 2^L$ signal points taken from the signal set $A = \{a_1, a_2, \ldots, a_M\}$. The multilevel encoder is shown in figure 4.1. Each bit $x_i \in \{0, 1\}$ of the binary address vector $\mathbf{x} = (x_L, x_{L-1}, \ldots, x_i, \ldots, x_1)$ is independently protected. bfx is then used to select a point from the constellation.

The code rate of each individual encoder is equal to $R_i = \frac{K_i}{N}$, where N is the length of the multilevel code. The total rate of the multilevel scheme is equal to $R = \frac{K}{N}$ where $K = \sum_{i=1}^{L} K_i$. Thus, a block of K information bits is encoded into N address vectors $\mathbf{x}$. Each of those vectors is used to select a signal point a from the constellation. The bijective mapping $a = M(\mathbf{x})$ is done using Ungerboeck's parti-

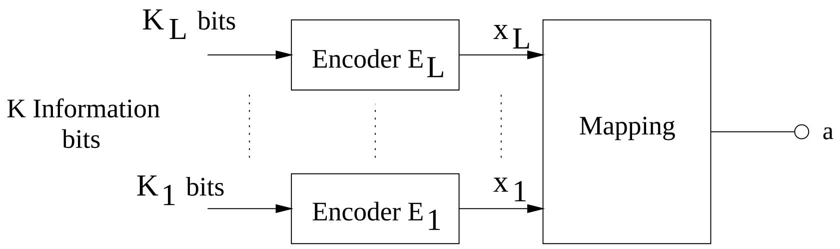


Figure 4.1: Multilevel encoding scheme.

tioning.

For example, Ungerboeck's partitioning of the one-dimensional 8-ASK modulation is illustrated in figure 4.2.

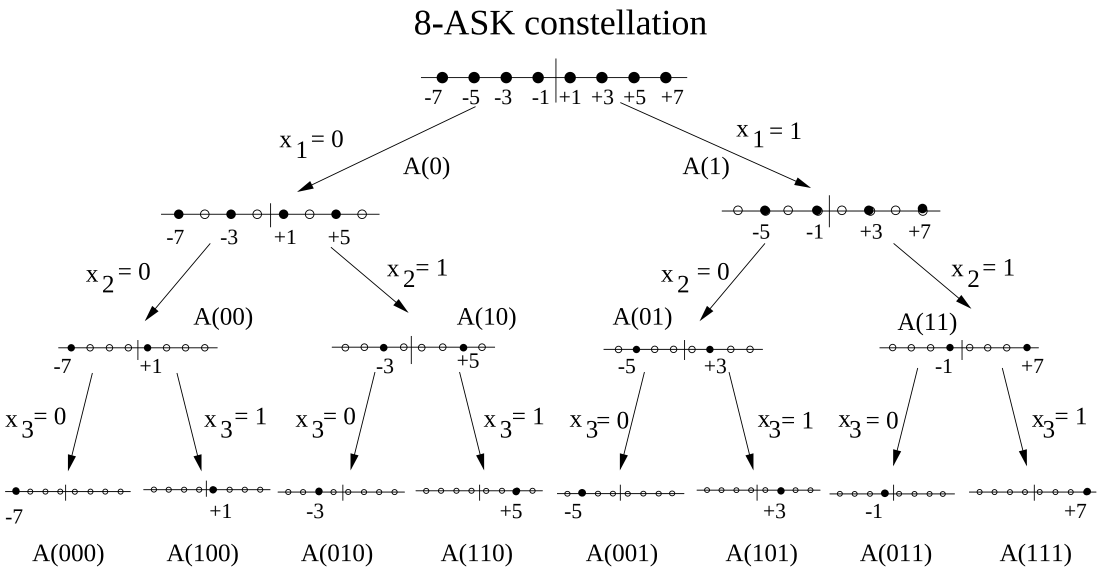


Figure 4.2: Ungerboeck's partitioning of an 8-ASK signal constellation.

At each level of partitioning, points of the constellation are divided into 2 subsets. The partitioning is done in a way to maximize the intra-subset minimum Euclidean distance. Each subset at partioning level i is uniquely labeled by the vector $(x_i, x_{i-1}, \ldots, x_1)$. The corresponding subset is denoted $A(x_i, x_{i-1}, \ldots, x_1)$. After L levels of partitioning, each subset will contain one signal point labelled by the vector $\mathbf{x} = (x_L, x_{L-1}, \ldots, x_i, \ldots, x_1)$.

In the case of a $M = 2^L$-ary ASK modulation ($D = 1$), this partitioning can be represented by the following equation

$$a = M(\mathbf{x}) = 2^{L-1}.[(2x_L - 1) + \sum_{i=1}^{L-1} \alpha_i.(2x_i - 1)] \qquad where \quad \alpha_i = \frac{1}{2^{L-i}} \tag{4.1}$$

Thus, the MLC encoder can be viewed as the sum of L independent binary modulations (see figure 4.3).

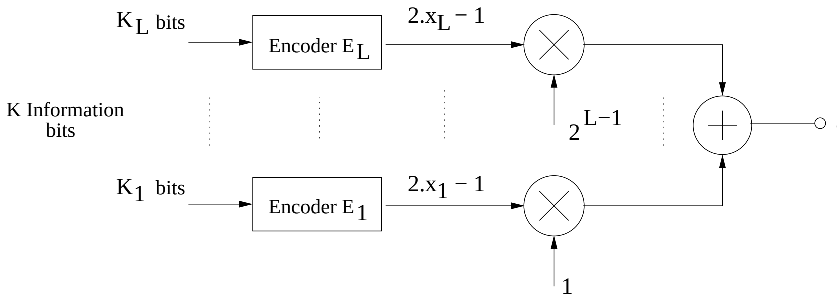


Figure 4.3: 8-ASK, MLC encoder.

Applying the chain rule of mutual information [37], the transmission of the vector $\mathbf{x}$ over the physical channel can be separated into L equivalent channels, one for the transmission of each bit $x_i, i = 1, 2, \ldots, L$ (see figure 4.4).

The input and the output of the physical channel are respectively represented by the random variables X and Y. The inputs of the equivalent channels are represented by the L binary random variables $X_1, X_2|X_1, \ldots, X_L|(X_1, \ldots, X_{L-1})$, while the outputs are represented by the same random variable Y. From the chain rule of mutual information, we get

$$I(Y;X) = I(Y;X_1, X_2, \ldots, X_L) = I(Y;X_1)+I(Y;X_2|X_1)+\ldots+I(Y;X_L|(X_1, X_2, \ldots, X_{L-1})) \tag{4.2}$$

Therefore, a suboptimal decoding technique, known as multistage decoding (see figure 4.5), can be used. The constituent codes E_i, at each level i, are successively decoded. We begin by decoding the level 1. Then, at each level i, decoder D_i uses the decided bits $\hat{x}_j$, $j = 1, \ldots, i-1$, of lower levels in order to compute his metric and determine the decided bits $\hat{x}_i$. As long as error-free decisions $\hat{x}_j = x_j$ are generated by the decoder D_j, MSD can be interpreted as an implementation of the chain rule.

In order to achieve better performance, interleaving between the levels, forwarding of reliability information from lower to higher levels using soft-output decoding algorithms, and especially iterative MSD have been proposed [100]. Simulation results show (cf. section 4.10) that the performance of MSD is almost equal to the one we get using iterative MSD. This can be explained by the fact that MLC combined to MSD achieves the capacity of the modulation transmitted over the physical channel (cf. section 4.3) and that this performance is only slightly inferior from the one we have for an optimum *maximum likelihood decoder*. We should also note that

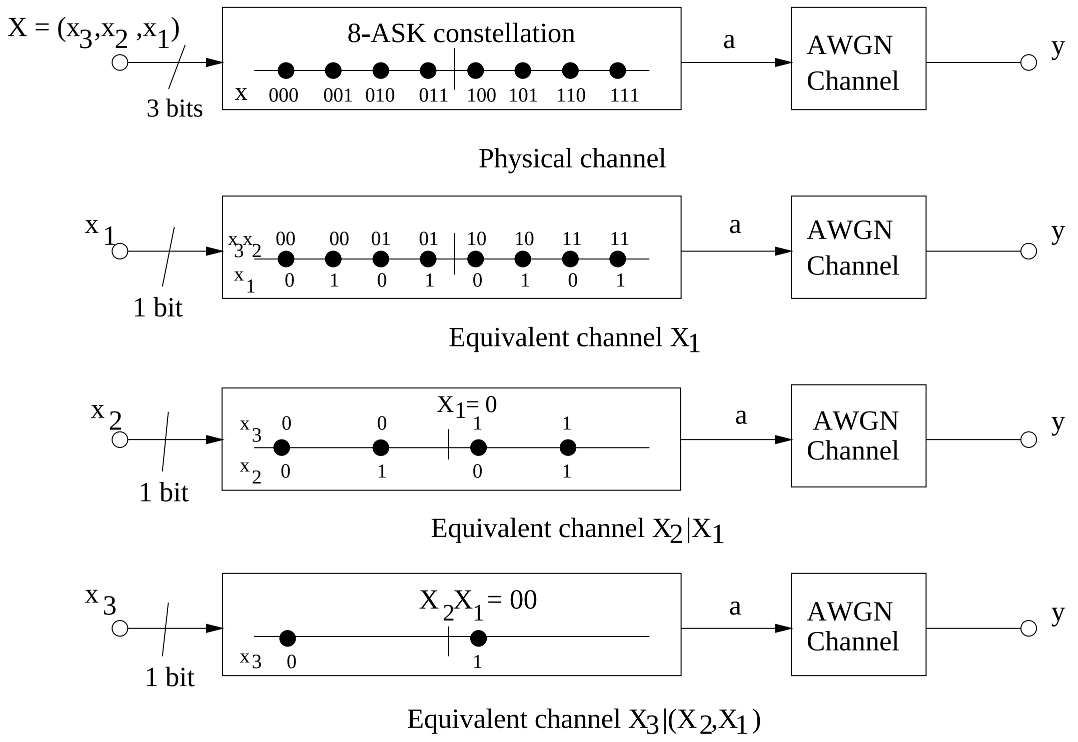


Figure 4.4: Equivalent channels of an 8-ASK modulation (Ungerboeck's partitioning).

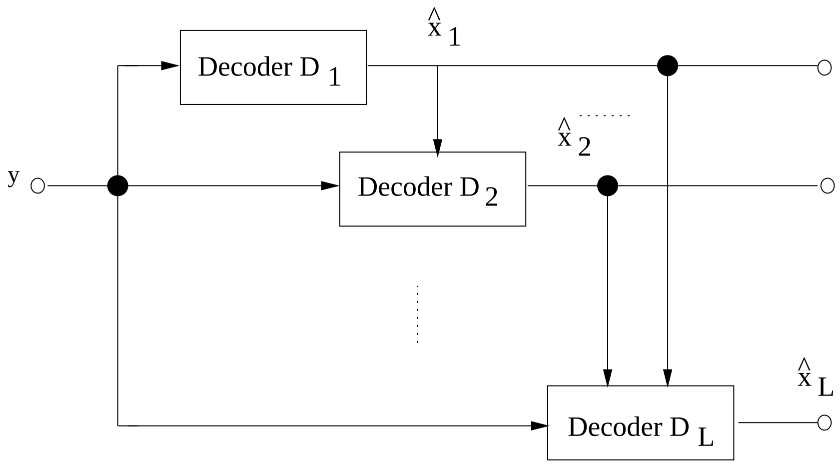


Figure 4.5: Multistage decoding of an MLC scheme.

the use of interleaving between levels increases the delay of data and the codeword length. Using such long component codes in direct way would yield a quite better performance.

4.3 Capacity of multilevel coding

Let $C_i,\ i = 1, 2, \ldots, L$ be the capacities of the L equivalent channels of a $M = 2^L$-ary multilevel coding modulation system. C_i is the maximum of the average mutual information $I(X_i|(X_1, \ldots, X_{i-1}), Y)$ where the maximization is done over all possible input probability distributions

$$C_i = \max_{P(x)} I(X_i|(X_1, \ldots, X_{i-1}); Y) \tag{4.3}$$

As mentioned in chapter 1, in the case of discrete input channels, the maximum is achieved by a uniform distributed input [18]. Furthermore, from the chain rule of mutual information, we have

$$I(X_i|(X_1, \ldots, X_{i-1}); Y) = I(X_i, \ldots, X_L|(X_1, \ldots, X_{i-1}); Y) - I(X_{i+1}, \ldots, X_L|(X_1, \ldots, X_i); Y) \tag{4.4}$$

Since the subsets at one partitioning level may not be congruent, the both mutual informations $I(X_i, \ldots, X_L|(X_1, \ldots, X_{i-1}); Y)$ and $I(X_{i+1}, \ldots, X_L|(X_1, \ldots, X_i); Y)$ are calculated by averaging over all possible combinations of $(x_1, \ldots, x_{i-1})$ and $(x_1, \ldots, x_i)$ respectively :

$$I(X_i, \ldots, X_L|(X_1, \ldots, X_{i-1}); Y) = E_{x_1, \ldots, x_{i-1}}\{I(X_i, \ldots, X_L|(x_1, \ldots, x_{i-1}); Y)\} \tag{4.5}$$

$$I(X_{i+1}, \ldots, X_L|(X_1, \ldots, X_i); Y) = E_{x_1, \ldots, x_i}\{I(X_{i+1}, \ldots, X_L|(x_1, \ldots, x_i); Y)\} \tag{4.6}$$

Replacing equations (4.5) and (4.6) in (4.3) and (4.4), we get

$$C_i = E_{x_1, \ldots, x_{i-1}}\{I(X_i, \ldots, X_L|(x_1, \ldots, x_{i-1}); Y)\} - E_{x_1, \ldots, x_i}\{I(X_{i+1}, \ldots, X_L|(x_1, \ldots, x_i); Y)\} \tag{4.7}$$

Finally, it is easy to show [98] [45] [26] that the capacity C of an $M = 2^L$-ary multilevel coding modulation system is equal to the sum of the capacities C_i of the equivalent channels i, $(i = 1, \ldots, L)$

$$C = \sum_{i=1}^{L} C_i \tag{4.8}$$

The capacity C can be approached via multilevel encoding and multistage decoding, if and only if the individual rates R_i are chosen to be equal to the capacities of the equivalent channels, *i.e.* $R_i = C_i$.

For example, for an 8-ASK scheme ($a \in \{\pm1; \pm3; \pm5; \pm7\}$) transmitted over an AWGN channel, the capacity C as well as the capacities C_1, C_2, C_3 of the 3 equivalent channels, when Ungerboeck's partitioning is applied, are shown in figure 4.6.

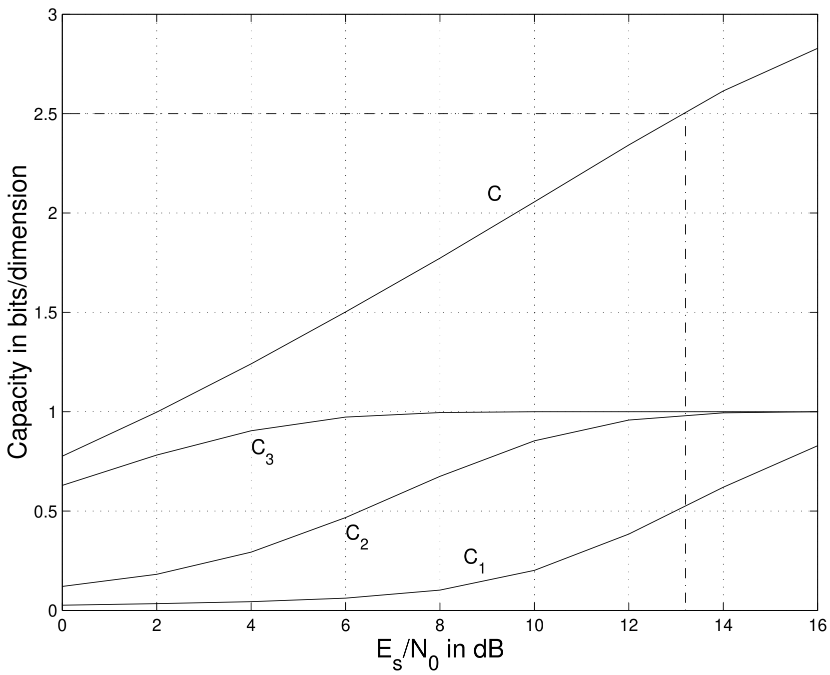


Figure 4.6: Capacity C of the 8-ASK modulation (AWGN channel) and capacities C_1, C_2 and C_3 of the equivalent channels as a function of $\frac{E_s}{N_0}$ in dB. Ungerboeck's partitioning is applied.

From these results we can conclude that:

1. As for lattice codes (see chapter 1), capacity can be approached via structured multilevel codes.

2. Sub-optimum multistage decoding can achieve the capacity if the rates R_i are equal to C_i.

3. For any digital communication system based on an $M = 2^L$-ary modulation, the capacity of the channel can be achieved using MLC that employs binary codes, and MSD. However, for finite block length N, non-binary codes may have some advantages in performance or complexity [69] [101]. In sections 4.8 and 4.9, we reduce the complexity of the MLC system by using asymmetric constellations.

4. There is no restriction on the particular labeling of signal points. Thus, Ungerboeck's partitioning is not essential to approach the capacity. Nevertheless, for finite code length, Ungerboeck's partitioning strategy turns out to lead the highest performance among MLC schemes with different partitioning strategies (cf. section 4.6). On the other hand, alternative partitioning strategies may be favorable for some other practical purposes as discussed also in section 4.6.

4.4 Capacity regions

L-level coded modulations may be interpreted as a special multiple access problem, *i.e.* L binary output sequences accessing a common channel. For example, figure 4.7 shows a 2-level MLC system accessed by 2 users: one at each level.

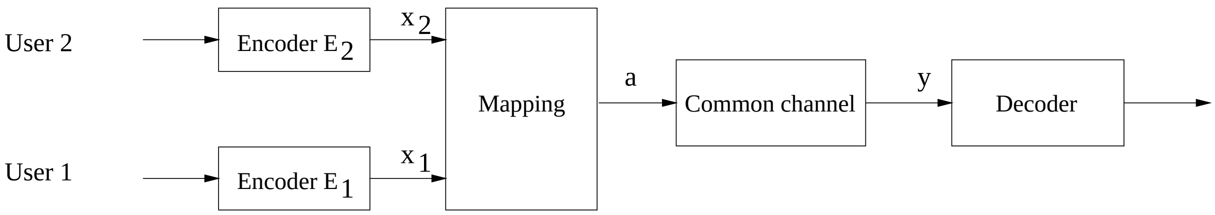


Figure 4.7: Multiple access using MLC.

The rates of the 2 levels are respectively R_1 and R_2. These rates are restricted by the following conditions:

1. Total rate $R = R_1 + R_2$ cannot exceed the capacity of the common channel

$$R_1 + R_2 \leq I((X_1, X_2); Y) \tag{4.9}$$

2. Rate for one user cannot exceed the capacity of the corresponding equivalent channel provided that the message of the other one is known at the receiver side

$$R_1 \leq I(X_1|X_2; Y), \quad R_2 \leq I(X_2|X_1; Y) \tag{4.10}$$

The two conditions of equations (4.9) and (4.10) form a polygon in the rate plane (see figure 4.8). Points "A" and "B" correspond to the chain rule equation (4.2) for both possible expansions and thus, the capacity is achievable even by a sub-optimum MSD. Note that, point "B" simply corresponds to an exchange of inputs X_1 and X_2, *i.e.* when X_2 is first decoded. This is due to a different mapping strategy known as *block partitioning* (see section 4.6).

In [98], Huber *et al* proved that MLC schemes with rate design corresponding to the points on the straight line connecting points "A" and "B" can achieve the capacity of the multilevel coding scheme if *maximum likelihood decoding* (MLD) is applied instead of MSD. As a result, the complexity of the system is largely increased. This problem can be avoided when a proper rate design is applied (as for points "A" and "B"). Moreover, using the same arguments, these results can be extended to the case of L-level MLC schemes.

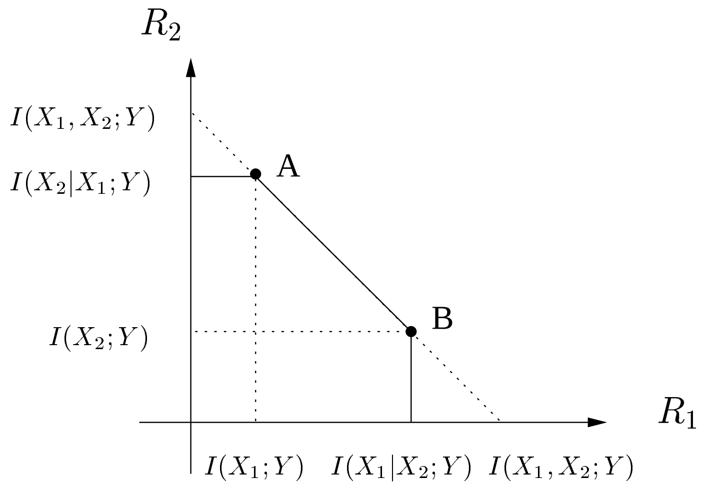


Figure 4.8: Asymmetric 8-PSK constellation

4.5 Rate design rules

A key element in the design of multilevel codes is the assignment of the code rates R_i of the L individual encoders C_i, $i = 1, \ldots, L$. In this section, we review the different rate design rules that can be used to determine these codes rates. We begin by investigating the *balanced distance rule* which was traditionally used to construct coded modulations schemes like multilevel codes [9].

4.5.1 Balanced distance rule

Let (N, K_i, δ_i) denote the parameters of the code C_i, $i = 1, \ldots, L$ where δ_i is the minimum Hamming distance of the code. Let d_i be the minimum intra-subset Euclidean distance over all subsets at partitioning level i. We assume that d_i was maximized by applying Ungerboeck's partitioning.

A lower bound on the minimum Euclidean distance $d_{E_{min}}$ between codewords in signal space is derived by (see e.g. [77])

$$d^2_{E_{min}} \geq \min_L (d_i^2.\delta_i), \qquad i = 1, \ldots, L. \tag{4.11}$$

Therefore, in order to maximize $d_{E_{min}}$, the constituent codes C_i, (*e.g.* the code rates R_i), must be selected in a way that the products $d_i^2.\delta_i$ are equal for all levels.

$$d_i^2.\delta_i = constant, \qquad i = 1, \ldots, L. \tag{4.12}$$

This is the *balanced distance* rule that was initially proposed for the design of multilevel codes [46] and the design of lattice codes (constructions B through E presented in [17]).

For an Ungerboeck partitioned 8-ASK constellation, the intra-subset minimum Euclidean distances are $d_1 = 2$, $d_2 = 4$ and $d_3 = 8$. From (4.12), we get that the normalized minimum Hamming distances of the constituent codes with length N must satisfy

$$2^2.\frac{\delta_1}{N} = 4^2.\frac{\delta_2}{N} = 8^2.\frac{\delta_3}{N} = constant. \tag{4.13}$$

For large code length N and for linear binary block codes with minimum Hamming distances which meet the Gilbert-Varshamov bound, the rates R_i and the minimum Hamming distances δ_i are related by [35]

$$R_i = 1 - H_2(\delta_i/N)\ bits/dimension \tag{4.14}$$

where $H_2(x)$ is the binary entropy function. For $Cte = 0.26$, the total rate R is equal to 2.5 bits/dimension and the rate distribution is $R_1 = 0.66$, $R_2 = 0.88$ and $R_3 = 0.96$.

This method suffers of two major disadvantages :

- First, the balanced distance rule does not take into account the multiple representations of binary symbols which cause an enormous multiplication of possible error events especially at lower levels. The error probability at the first level significantly increases. These errors propagate through higher levels and therefore, a degradation in performance occurs when multistage decoding is applied. More complex decoding techniques that can approach ML decoding must be implemented instead of MSD (cf. section 4.2).

- Secondly, several compound codes, like turbo codes, cannot be characterized by their minimum Hamming distance which is commonly lower than the one obtained from the Gilbert-Varshamov bound.

To avoid these problems, new design rules based on parameters from information theory must be used instead of the balanced distance rule. We mainly focus on the capacity rule and the coding exponent rule.

4.5.2 Capacity rule

To achieve the capacity of a multilevel coding system, the individual rate R_i must be equal to the capacity of its corresponding equivalent channel (cf. section 4.3).

$$R_i = C_i \qquad i = 1, 2, \ldots, L \tag{4.15}$$

For example, in the case of an Ungerboeck partitioned 8-ASK constellation and for $R = C = 2.5$ bits/dimension, the code rates are $R_1 = 0.52$, $R_2 = 0.98$ and $R_3 = 1$. When compared with the rates obtained from the balanced distance rule, we obviously see that the rate of the first level is smaller. This improves the performance of the first level and therefore reduce the error propagation through higher levels.

4.5.3 Coding exponent rule

The capacity rule is well suited for the design of MLC schemes with large length N (codes that perform close to capacity). However, for small values of N, it is more convenient to use the random coding bound which provides a relation between N and the word error probability P_e of the system(cf. section 1.3.4 in chapter 1)

$$P_e = 2^{-NE(R)} \tag{4.16}$$

where R is in bits/dimension and $E(R)$ is the Gallager coding exponent

$$E(R) = \max_{0 \leq \rho \leq 1} [E_0(\rho) - \rho R] \tag{4.17}$$

For a discrete-input, continuous-output channel like the AWGN channel, $E_0(\rho)$ is defined by

$$E_0(\rho) = -log_2 \int_y \left[\sum_x P(x) P(y|x)^{\frac{1}{1+\rho}} \right]^{1+\rho} dy \tag{4.18}$$

Let $E_i(R_i)$ be the Gallager coding exponent of the equivalent channel i

$$E_i(R_i) = \max_{0 \leq \rho \leq 1} [E_{0i}(\rho) - \rho R_i] \qquad i = 1, \ldots, L \tag{4.19}$$

where $E_{0i}(\rho)$ is obtained by averaging the parameter $E_{0i}(\rho, x_i, x_{i-1}, \ldots, x_1)$ over all the subsets $A(x_i, x_{i-1}, \ldots, x_1)$ at level i

$$E_{0i}(\rho) = E_{x_i, x_{i-1}, \ldots, x_1} \{E_{0i}(\rho, x_i, x_{i-1}, \ldots, x_1)\} \tag{4.20}$$

and

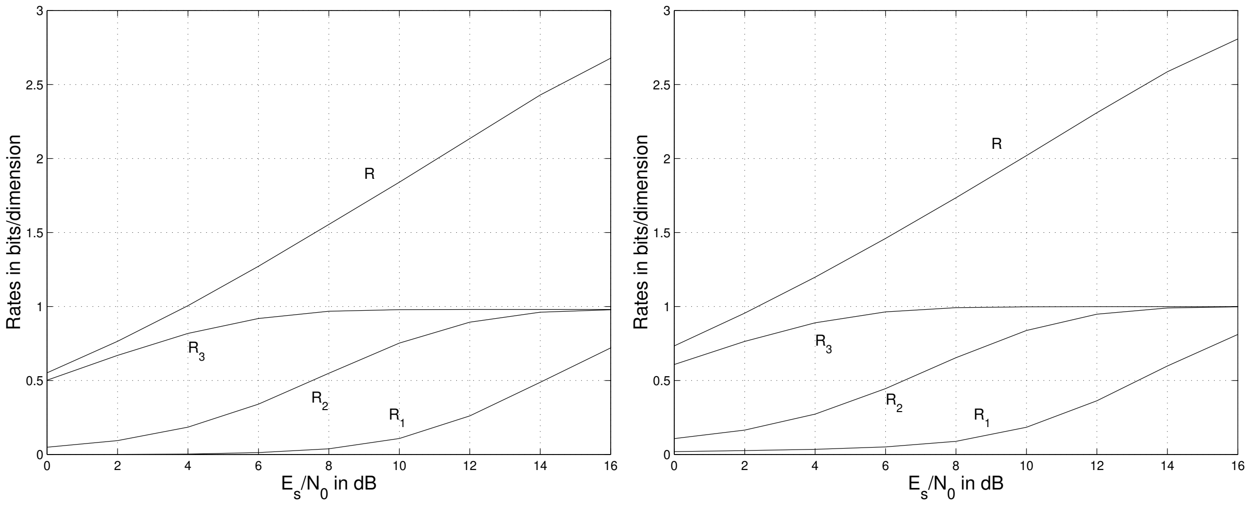
$$E_{0i}(\rho, x_i, x_{i-1}, \ldots, x_1) = -log_2 \int_y \left[\sum_{x_i=0}^{1} P(x_i) P(y|x_i)^{\frac{1}{1+\rho}} \right]^{1+\rho} dy \tag{4.21}$$

Then, to determine the code rates R_i, $E_i(R_i)$, and hence the word error probability P_e, is fixed to a constant value

$$E_i(R_i) = -\frac{1}{N} log_2 P_e = constant \tag{4.22}$$

For a fixed signal-to-noise ratio, the rate R_i that satisfies this equation is determined by evaluating $E_i(R_i)$. For the AWGN channel, $E_i(R_i)$ must be evaluated numerically.

Figures 4.9.a. and 4.9.b. show the total code rate R and the code rates R_1, R_2 and R_3 of the equivalent channels of an Ungerboeck partitioned 8-ASK as a function of the average signal-to-noise ratio per symbol E_s/N_0 in dB. The code length N is respectively equal to 500 and 100000. The word error probability is equal to 10^{-3}. This particular value of P_e guarantees that the bit error probability is less than 10^{-5} for both values of N. Note that for $N = 100000$, the curves coincide with those obtained using the capacity rule. In fact, the asymptotic values for $N \longrightarrow +\infty$ are those given by the capacity design rule. For $N = 500$, there is a significant difference between the results of the two design rules.



a. $N = 500$. b. $N = 100000$.

Figure 4.9: Total rate R of the 8-ASK MLC scheme (AWGN channel) and rates R_1, R_2 and R_3 of the three levels, derived from the coding exponent rule, as a function of $\frac{E_s}{N_0}$ in dB. $P_e = 10^{-3}$. Ungerboeck's partitioning is applied.

For $R = 2.5$ bits/dimension and $P_e = 10^{-3}$, the code rates R_i are sketched as a function of the code length N of the multilevel code (see figure 4.10). We observe that for $N > 1000$, the values of the capacity rule are a good approximation of the exact values calculated using the balanced distance rule. On the other hand, when N goes to zero the code rates R_i go to the values obtained using the balanced distance rule.

These design rules are the main ones that can be used to determine the code parameters of a multilevel coding scheme. Other design rules, like the cutoff rate rule and the equal error probability design rule [98], are not of great interest. The cutoff rate rule, $\forall\ i\ R_i = R_0^i$, is only useful for codes that cannot approach capacity but just the cutoff rate R_0^i of the equivalent channels. The equal error probability rule is limited to the class of codes for which an analytical expression of the word error probability can be determined for each of the equivalent channels. In this case, the rates are determined in a way that the word error probability is the same for all the levels of the multilevel coded modulation.

4.6 Labeling strategies

From section 4.3, we concluded that Ungerboeck's partitioning is not essential to approach capacity. Several labeling strategies can be used in practice. These labeling strategies are discussed in this section. We mainly focus on block partitioning and Gray labeling which both have some practical advantages. The performance of these

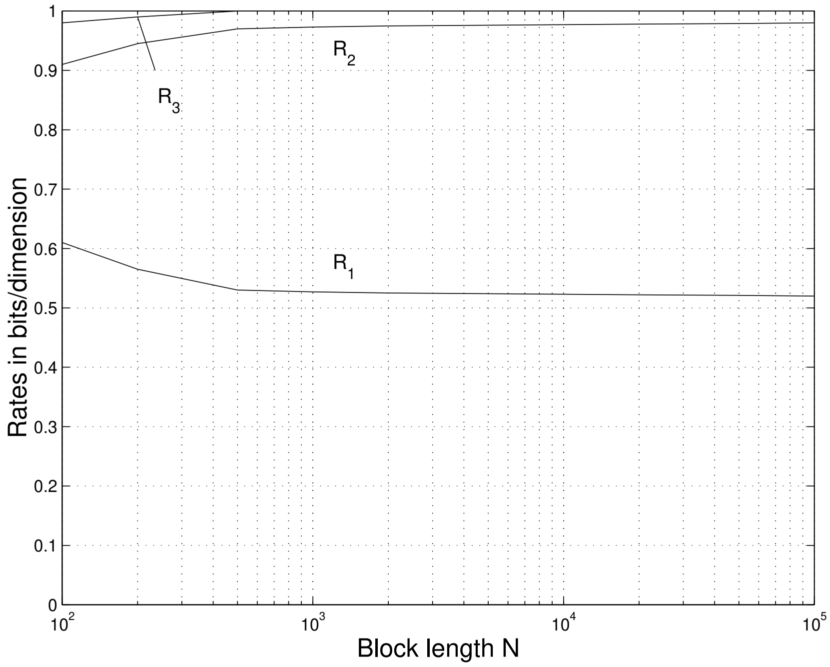


Figure 4.10: Distribution of rates R_1, R_2 and R_3 of the three levels of an 8-ASK MLC scheme (AWGN channel), derived from the coding exponent rule, as a function of the block length N. $P_e = 10^{-3}$. Ungerboeck's partitioning is applied.

labeling strategies is determined by calculating Gallager's coding exponent. This allows us to compare the performance of these labeling strategies for finite block length N.

4.6.1 Block partitioning

Usually, the partitioning of a signal set, *i.e.* a constellation, is done using Ungerboeck's partitioning which is based on maximizing the minimum intra-subset Euclidean distance. However an alternative labeling strategy, known as block partioning can be investigated. Block partitioning is based on minimizing the intra-subset variance. Figure 4.11 shows the block partitioning of an 8-ASK constellation. At each level, the minimum intra-subset Euclidean distance is the same.

As for Ungerboeck's partitioning, a signal point from the constellation is labelled by the vector $\mathbf{x} = (x_L, x_{L-1}, \ldots, x_i, \ldots, x_1)$ where x_i is the binary input of the i^{th} equivalent channel. However, on the opposite to Ungerboeck's partitioning, labeling of the subsets is done begining with x_L and then $x_{L-1}, \ldots, x_i, \ldots, x_1$. This yields to the same bijective mapping $a = M(\mathbf{x})$ obtained applying Ungerboeck's partitioning. For example, for an M-ary ASK constellation, the mapping can be represented by the same equation (4.1). However, in this case, the L equivalent channels are represented by the L binary random variables $X_L, X_{L-1}|X_L, \ldots, X_1|(X_L, \ldots, X_2)$. Therefore, it is quite obvious that MSD is processed by first decoding the higher level L, and then decoding the lower levels $L-1, \ldots, i, \ldots, 1$ (see figure 4.12).

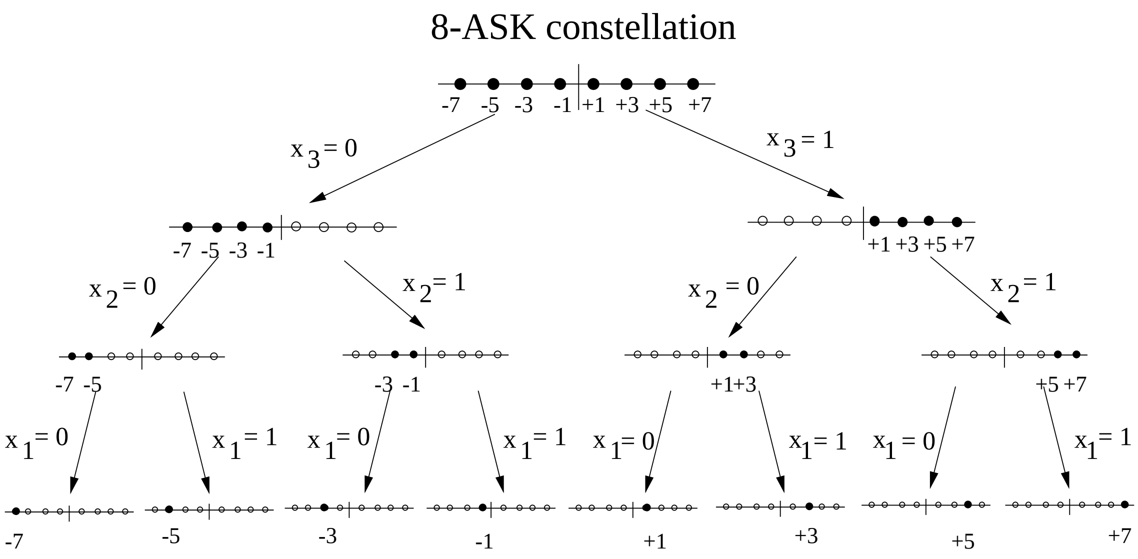


Figure 4.11: Block partitioning of an 8-ASK signal constellation.

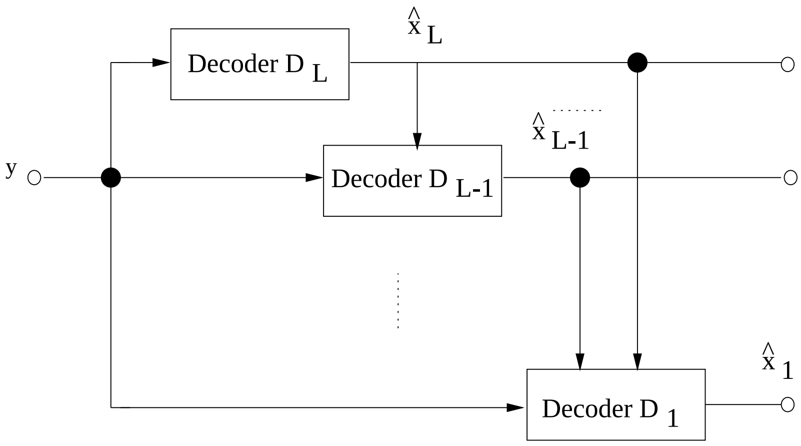


Figure 4.12: Multistage decoding of an MLC scheme when block partitioning is applied.

Block partitioning leads to equivalent channels with capacities C_i that have a smaller divergence (see figure 4.13).

This may facilitate implementation in some cases. For example, block partitioning can be applied to construct softly degrading MLC schemes where decoding in the MSD is done only down to the level that still delivers reliable data [98] [45]. A second application of block partitioning is to construct MLC schemes based on asymmetric constellations which have equal capacity at each level (see section 4.8).

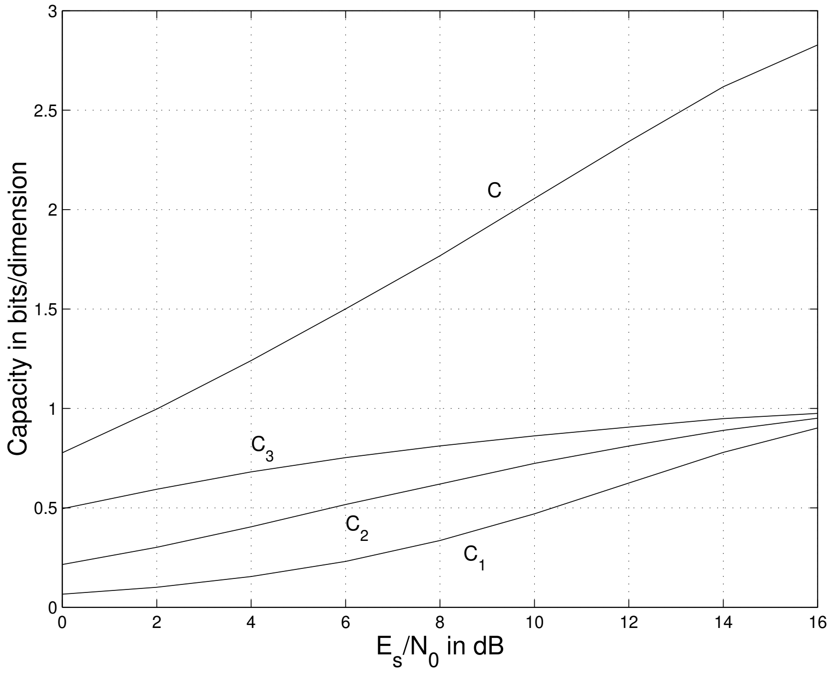


Figure 4.13: Capacity C of the 8-ASK modulation and capacities $C_i,\ i = 1, 2, 3$ of the equivalent channels as a function of $\frac{E_s}{N_0}$ in dB. Block partitioning is applied.

4.6.2 Gray labeling

Gray labeling is an interesting alternative first used in [101] to construct an 8-PSK modulation scheme for Rayleigh fading channels, that outperforms the best known trellis codes. The main idea is to use a single convolutional encoder that has a code rate equal to 2/3. The coded bits are randomly interleaved and then mapped to signal points using Gray labeling. This modulation technique is known as *Bit Interleaved Coded Modulation* (BICM).

In this section, we analyze an MLC scheme based on Gray labeling. This will be of great interest when a sub-optimal *parallel independent decoding* is used instead of MSD.

Figure 4.14 shows the partitioning of an 8-ASK constellation when Gray labeling is applied. As for block partitioning, the minimum intra-subset Euclidean distance remains the same for all partitioning levels. The difference between the capacities C_i of the equivalent channels is small (see figure 4.15). Moreover, a careful analysis done in the next two sections shows that MSD can be replaced by an independent *parallel decoding* of levels, without significant loss in capacity and performance.

4.6.3 Parallel independent decoding

In MLC with *parallel independent decoding* (PDL) of the individual levels (see figure 4.16), the decoder D_i makes no use of the decisions of other levels $i \neq j$.

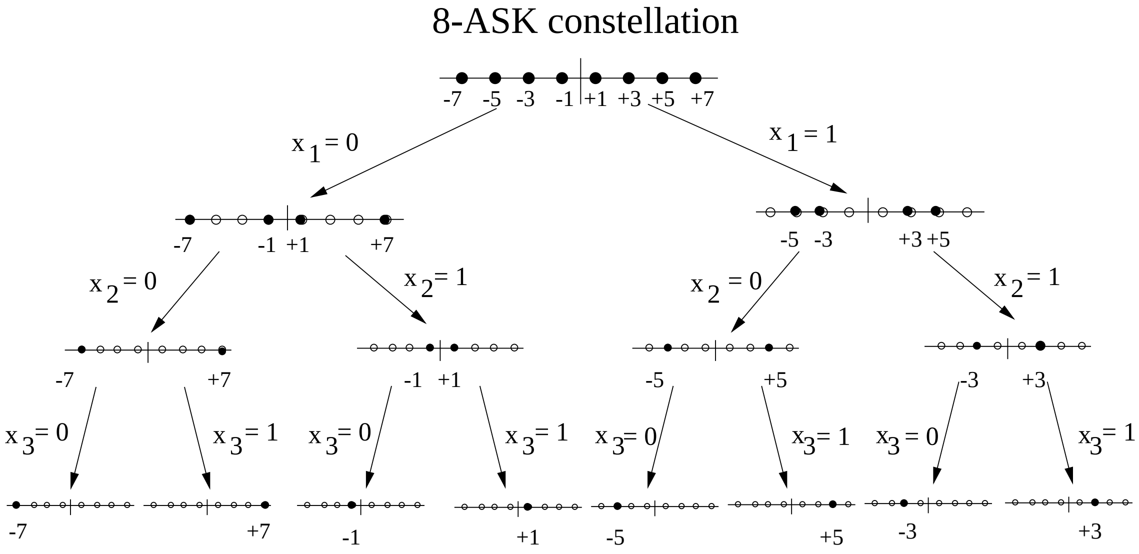


Figure 4.14: Partitioning of an 8-ASK signal constellation: Gray labeling.

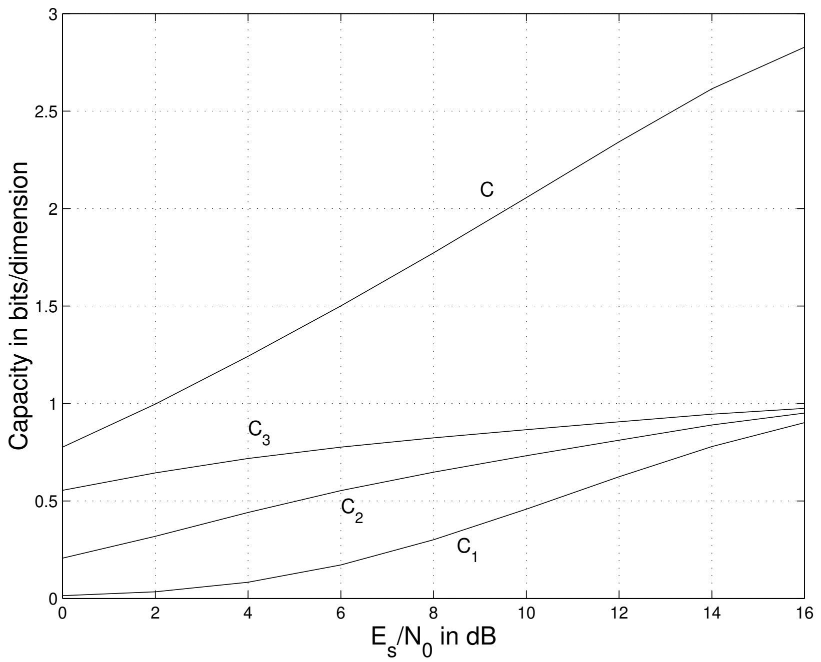


Figure 4.15: Capacity C of the 8-ASK modulation and capacities $C_i,\ i = 1, 2, 3$ of the equivalent channels as a function of $\frac{E_s}{N_0}$ in dB. Gray labeling is applied.

Therefore, the individual rates are bounded by

$$R_i \leq C_{PDL_i}, \qquad i = 1, 2, \ldots, L \tag{4.23}$$

where C_{PDL_i} is the maximum of the mutual information $I(X_i; Y_i)$, *i.e.* capacity of the equivalent channel i when PDL is applied. It is obvious that

$$I(X_i; Y_i) \leq I(X_i | (X_{i-1}, \ldots, X_1); Y_i) \tag{4.24}$$

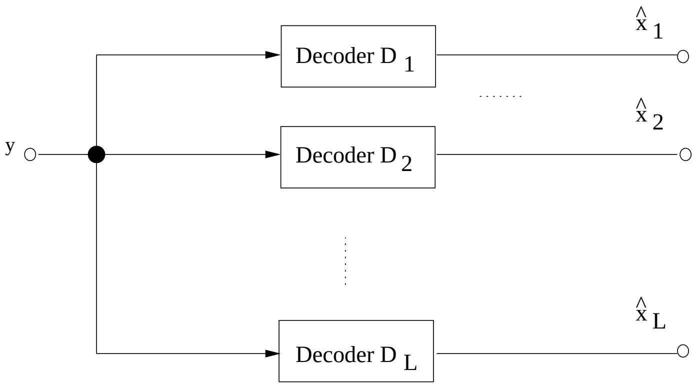


Figure 4.16: Parallel independent decoding of an MLC scheme.

Thus, $C_{PDL} \leq C$, where C is the total capacity of the MLC scheme when MSD is applied. Therefore, PDL approach can be viewed as a sub-optimum decoding technique of an optimum MLC scheme. The capacity C_{PDL} strongly depends on the labeling strategy. The gap to the total capacity C is small when Gray labeling is used. To understand why this gap is small, we can simply analyze the subsets of the set partitioning tree in figure 4.14 and verify that the minimum distance and the error coefficient stay unchanged when MSD is replaced by PDL. Figure 4.17 shows the capacities C, C_{PDL} and the capacities C_{PDL_i} of the equivalent channels when Gray labeling is applied to an 8-ASK. The gap between C and C_{PDL} goes to zero when $\frac{E_s}{N_0}$ goes to infinity. This can be explained by the fact that when $\frac{E_s}{N_0}$ goes to infinity, the inputs X_i, $i = 1, \ldots, L$, will be independent; as a result, equality will occur in equation (4.24).

Finally, we should note that a BICM modulation, implemented using an infinite length ideal interleaver, can be viewed as an MLC scheme based on Gray labeling and decoded using a PDL. Therefore, BICM and MLC combined to Gray labeling and PDL have the same capacity. The small degradation in capacity due to PDL is similar to the one found by Caire *et al* [13] [14] regarding the small capacity loss of BICM over the AWGN channel when Gray labeling is applied to an 8-PSK and a 16-QAM constellation.

4.6.4 Performance comparison for finite block length

The comparison between the different labeling strategies was done based on their capacities. However, for finite block length N, Ungerboeck's partitioning strategy

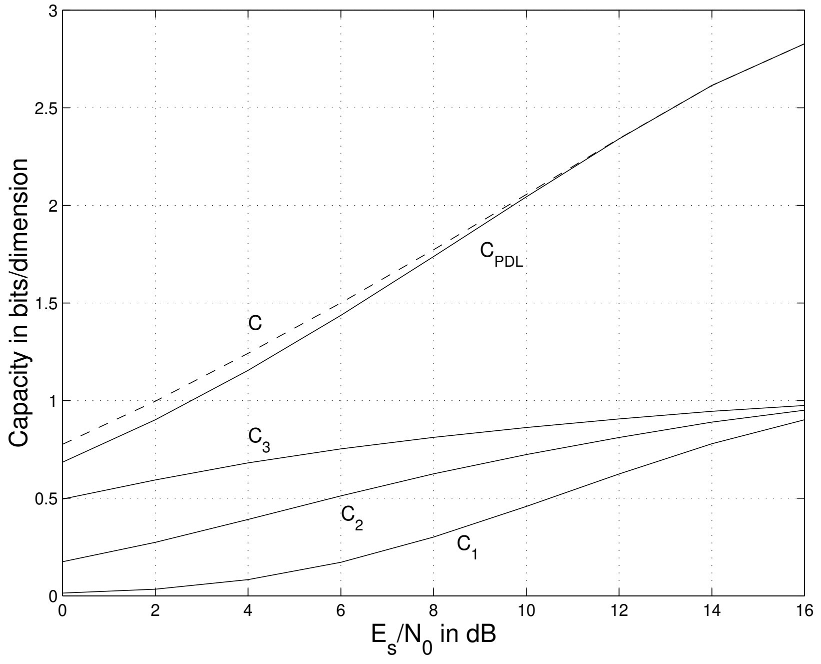
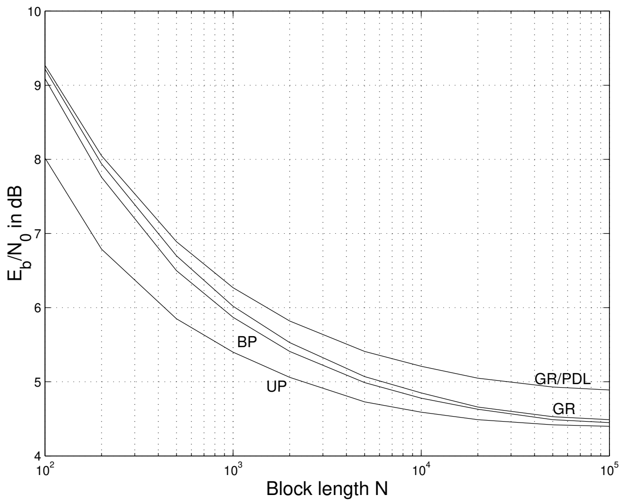


Figure 4.17: Capacity C of the 8-ASK modulation. Total capacity C_{PDL} and capacities C_{PDL_i}, $i = 1, 2, 3$ of the equivalent channels as a function of $\frac{E_s}{N_0}$ in dB, when Gray labeling with parallel independent decoding is applied.

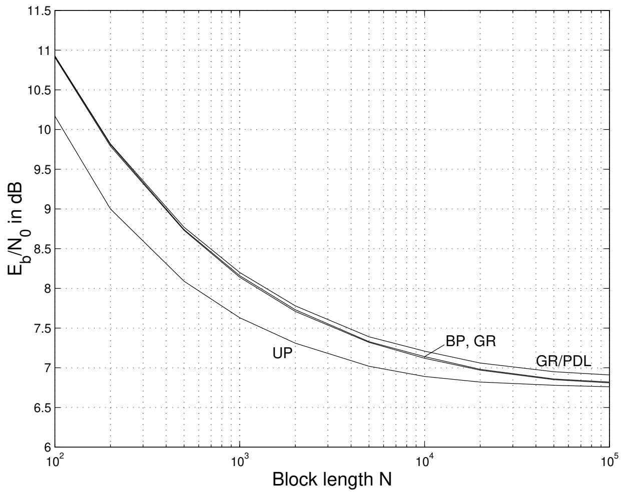
turns out to lead to highest performance. This result can be verified by calculating Gallager's random coding exponent.

For example, figures 4.18.a., 4.18.b. and 4.18.c. show the required signal-to-noise ratio per bit $\frac{E_b}{N_0}$ to achieve a word error probability $P_e = 10^{-3}$ versus the code word length N. The total rate is respectively equal to $R = 1.5$, $R = 2$ and $R = 2.5$ bits/dimension. For a fixed code length N, the gaps in performance between Ungerboeck's partitioning and the other labeling strategies are smaller for larger values of R. Moreover, as block length N goes to infinity, the gaps go to zero in the case of block partitioning and Gray Labeling since labeling strategy does not affect the capacity. For Gray labeling with PDL decoding, the gap is equal to the one we have when the capacities are compared (0.4 dB for $R = 1.5$ bits/dimension). We get from these results that, without a significant loss, MSD can be replaced by PDL when Gray labeling is applied; This specially true for high rates R and large block length N.

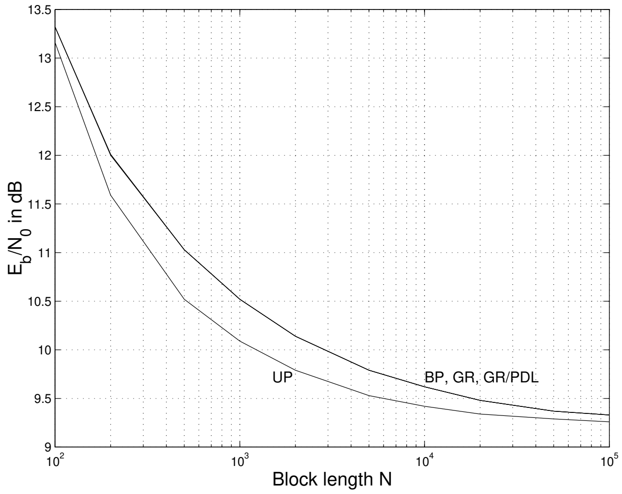
On the other hand, The performance of PDL when Gray labeling is applied are almost equal to the performance of BICM even for small block length N [98].



a. $R = 1.5$ bits/dimension.



b. $R = 2$ bits/dimension.



c. $R = 2.5$ bits/dimension.

Figure 4.18: Required $\frac{E_b}{N_0}$ in dB to achieve a word error probability $P_e = 10^{-3}$, as a function of the block length N. 8-ASK modulation. Labeling strategies: Ungerboeck's partitioning (UP), block partitioning (BP), Gray labeling (GR) and Gray labeling + PDL (GR/PDL).

4.7 Dimensionality of the constituent signal constellation

For band-limited channels, multidimensional modulations are largely employed. In the case of a TCM system, one redundant bit is introduced per D-dimensional symbol. The redundancy is equal to $1/D$ bits/dimension. However, for an MLC scheme the rate in bits/dimension is independent from the dimension D of the signal set. Therefore, in the design of a multilevel code, any D-dimensional modulation can be employed. The choice of D is based especially on performance and complexity. For example, let us consider a multilevel coded modulation system based on a *two*-dimensional M-ary squared QAM constellation. We consider that the total code rate is equal to R bits/dimension and that a codeword contains N M-ary coded symbols. The choice of N is fixed by the maximal data delay allowed during transmission.

This MLC scheme can be constructed in two different ways.

First we can choose $D = 2$ dimensions, apply one of the labeling strategies previously defined to the two-dimensional M-QAM constellation, and therefore obtain a $L = 4$-level MLC scheme. The code length of the binary block codes will be equal to N.

Secondly, we can choose $D = 1$, and therefore construct the MLC scheme using a 4-ASK constellation. In this case, the number of levels is $L = 2$ and the code length is $2N$. The $2 \times N$ ASK symbols are then combined into N two-dimensional QAM symbols.

The MLC scheme based on a one-dimensional ASK scheme is largely simpler than the first one. Moreover, simulation results done by Huber *et al* using turbo codes as component codes, showed that this system outperforms the one based on the 2-dimensional QAM constellation[1]. For $M = 16$ the gain is about 0.25 dB. Therefore, in the sequel, we focus our study on the one-dimensional ASK constellation and two-dimensional PSK which cannot be represented as two one-dimensional constellations.

4.8 MLC based on asymmetric 2^L-ASK modulation

Let us consider the one-dimensional ASK constellation with signal points a. After block partitioning, the bijective mapping $a = M(\mathbf{x})$ can be represented by the following equation:

$$a = 2^{L-1} \times \sum_{i=1}^{L} \alpha_i \ (2x_i - 1) \tag{4.25}$$

where the parameter α_i is an homothetic factor and $\alpha_L = 1$ is fixed. Let C be the capacity of the symmetric constellation obtained when the scaling factors $(\alpha_1, \alpha_2, \ldots, \alpha_{L-1})$ satisfy [2]

$$\alpha_i = \frac{1}{2^{L-i}} \qquad i = 1, 2, \ldots, L-1 \tag{4.26}$$

[1] This is true for block component codes like turbo codes. For a convolutional component code, the two-dimensional constellation performs better.

[2] In the case of a symmetric 8-ASK, $(\alpha_1, \alpha_2) = (0.25, 0.5)$.

Let C_a be the capacity of the asymmetric constellation obtained using arbitrary values of $\alpha_i \in [0,1[, \quad i = 1,2,\ldots,L-1$. We also define C_{ai} as the capacity of the equivalent channel at level i. The scaling factors can be determined in such a way that the capacities C_{ai} are equal. This is done numerically for each average signal-to-noise ratio per symbol $\frac{E_s}{N_0}$.

The two parameters (α_1, α_2) of an 8-ASK modulation reaches $(0.25, 0.5)$ when $\frac{E_s}{N_0}$ increases to infinity (see figure 4.19). These values correspond to the symmetric shape. A numerical search was made to find the values of the total capacities C_a and C. The result shows that the two capacities are almost equal (see figure 4.20). For $\frac{E_s}{N_0} \leq 13$ dB, C_a is even greater than C, which means that a small shaping gain is achieved by the asymmetric constellation.

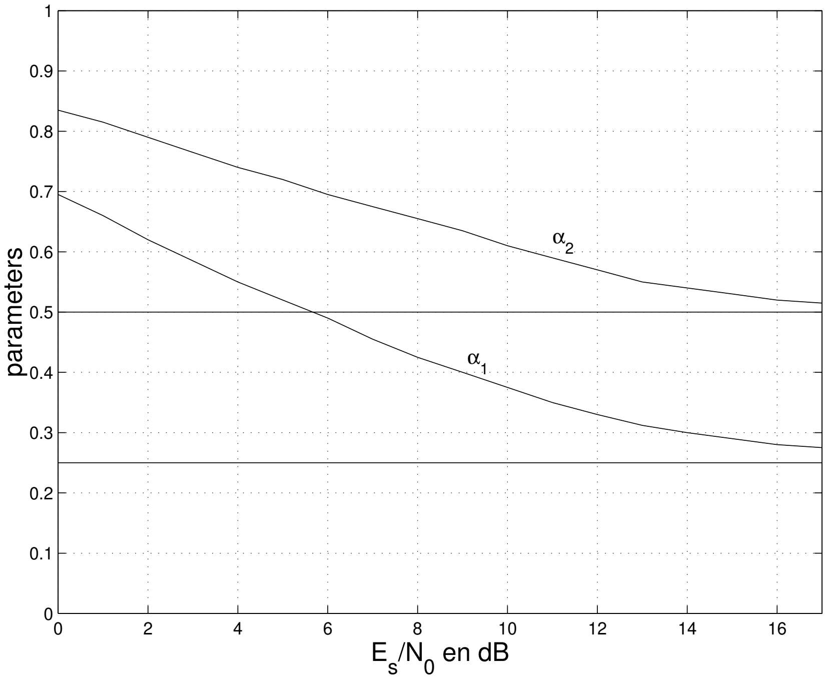


Figure 4.19: Parameters α_1 and α_2 of the asymmetric 8-ASK modulation as a function of $\frac{E_s}{N_0}$ in dB.

The above result is very interesting : the total capacity is achieved using an asymmetric constellation that has equal capacity at each level.

$$\forall \; i \qquad R_i = C_{ai} = \frac{C_a}{L} \cong \frac{C}{L} \tag{4.27}$$

Thus, instead of using L encoders and obviously L decoders, only one encoder and one decoder are sufficient to implement the MLC system. This reduces the complexity by a factor of L. No additional data delay will occur if we increase the speed of the encoder. There is no need to increase the speed of the decoder since MSD is applied.

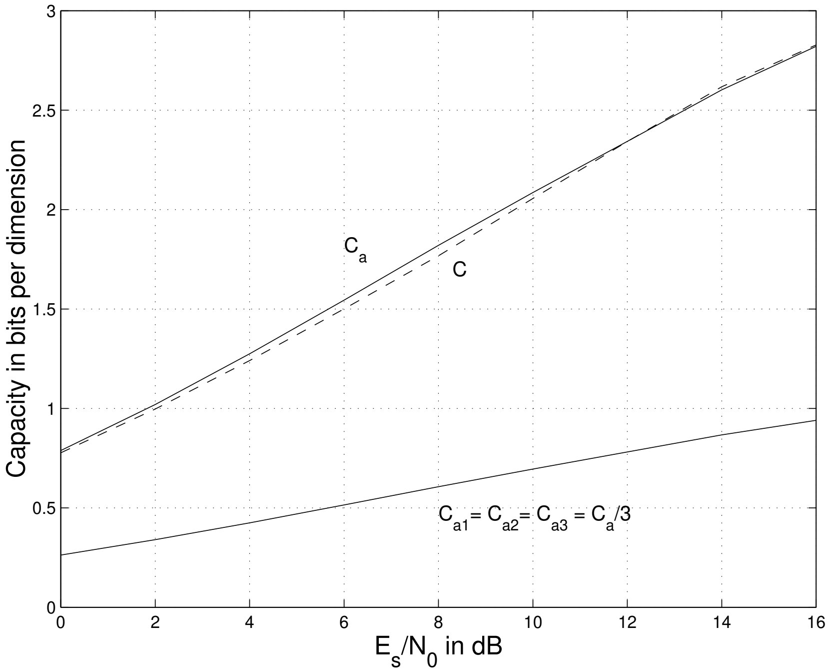


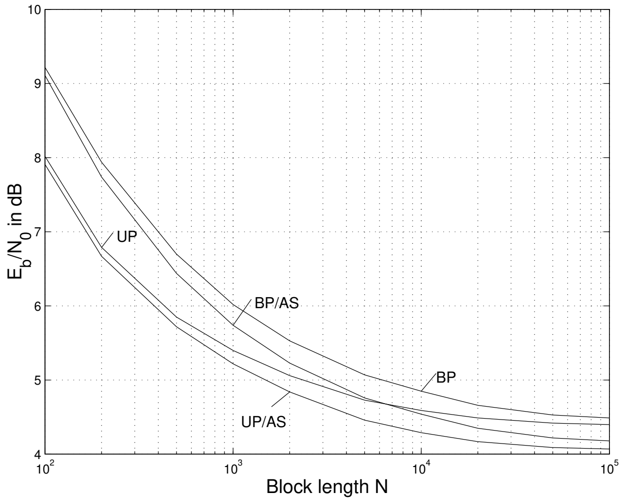
Figure 4.20: Capacity C of the 8-ASK modulation, capacity C_a of the asymmetric 8-ASK modulation and capacities $C_{a1} = C_{a2} = C_{a3} = C_a/3$ of the equivalent channels as a function of $\frac{E_s}{N_0}$ in dB.

However, even if the asymmetric constellation has the same capacity as for the symmetric one, a degradation in performance occurs for finite code length N. This degradation is due to block partitioning labeling strategy that must be applied instead of Ungerboeck's partitioning in order to get L equivalent channels that have the same capacity.

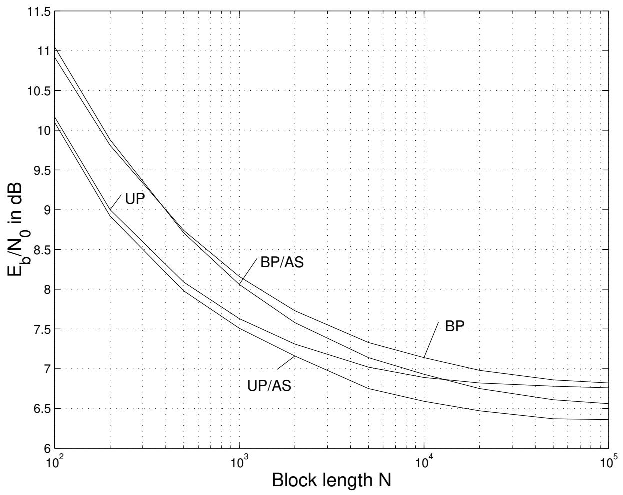
To evaluate this degradation in performance, we calculate the average signal-to-noise ratio per bit $\frac{E_b}{N_0}$ needed to achieve a word error probability equal to[3] $P_e = 10^{-3}$ at each level. This is done by finding the values of the L Gallager coding exponents [98] [45].

Figures 4.21.a., 4.21.b. and 4.21.c. show the average signal-to-noise ratio per bit $\frac{E_b}{N_0}$ versus the code length N for code rates respectively equal to 1.5, 2 and 2.5 bits/dimension. Curves correspond to : Ungerboeck's partitioning applied to a symmetric constellation (UP), Ungerboeck's partitioning applied to an asymmetric constellation which has the best performance (UP/AS), block partitioning applied to a symmetric constellation (BP) and block partitioning applied to an asymmetric constellation that has equal rate at each level (BP/AS). When N tends to infinity, $\frac{E_b}{N_0}$ approaches the value that corresponds to the capacity limit.

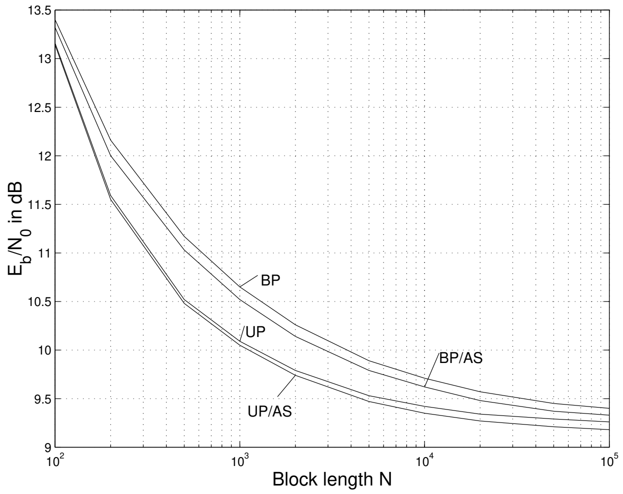
[3]For $P_e = 10^{-3}$ and large values of N, a bit error probability $P_b < 10^{-5}$ is achievable.



a. $R = 1.5$ bits/dimension.



b. $R = 2$ bits/dimension.



c. $R = 2.5$ bits/dimension.

Figure 4.21: Required $\frac{E_b}{N_0}$ in dB to achieve a word error probability $P_e = 10^{-3}$, as a function of the block length N. 8-ASK modulation.

We conclude from these results that :

1. For both symmetric and asymmetric constellations, Ungerboeck's partitioning performs better than block partitioning, especially for small code length N. When N goes to infinity, the two partitioning strategies have the same performance.

2. For both Ungerboeck's and block partitioning, a gain in signal-to-noise ratio is achieved with asymmetric constellations, especially when N goes to infinity and for moderate code rates ($R = 2$ bits/dimension). This can be viewed as a shaping gain introduced by the asymmetric constellations.

3. An MLC scheme based on asymmetric constellations with equal capacity at each level, can perform better than a symmetric one even if it is Ungerboeck partitioned. This is true when the code length N is large enough and for a code rate R not too close to the capacity of the modulation vehiculed by the AWGN channel. As an example, for $R = 1.5$ bits/dimension, a signal-to-noise ratio gain is obtained for $N \geq 7000$. For $R = 2$ bits/dimension, N has to be greater than 15000 to get a positive gain.

4.9 MLC based on asymmetric 8-PSK modulation

PSK modulations are widely used in practice specially for the transmission over non-linear channels like the satellite channel. Several coded modulations schemes based on PSK modulation were also proposed for the Rayleigh fading channel [101]. The purpose of this section is to study MLC systems based on $M = 2^L$-ary asymmetric PSK modulation and in particular the one based on asymmetric 8-PSK. We begin by determining the asymmetric 8-PSK constellation that leads to an MLC scheme that has equal capacity at each level.

An asymmetric 8-PSK constellation defined by two angles θ_1 and θ_2 is represented figure 4.22. As for asymmetric ASK constellations, and when block partitioning is applied, the angles (θ_1, θ_2) can be determined in such a way that the capacities $C_{ai},\ i = 1, 2, 3$ of the equivalent channels are equal. Numerically, we got that the two angles θ_1 and θ_2 are almost equal ($\theta_1 = \theta_2 \simeq \theta$). This can be explained by the fact that, for the symmetric shape and under block partitioning, the equivalent channels 2 and 3 already had the same capacity (see figure 4.24).

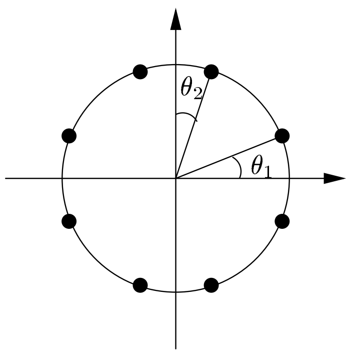


Figure 4.22: Asymmetric 8-PSK constellation

Figure 4.23 shows the angle θ in degrees as a function of the average signal-to-noise ratio $\frac{E_s}{N_0}$ in dB. When $\frac{E_s}{N_0}$ approaches infinity, θ goes to 22.5 degrees. This is the value of θ that corresponds to a symmetric 8-PSK constellation. The capacity C of the symmetric 8-PSK constellation as well as the capacity C_a of the asymmetric one that has the same capacity at each level are depicted in figure 4.24. Notice that there is a small degradation in capacity as in the case of a BICM modulation [13] [14].

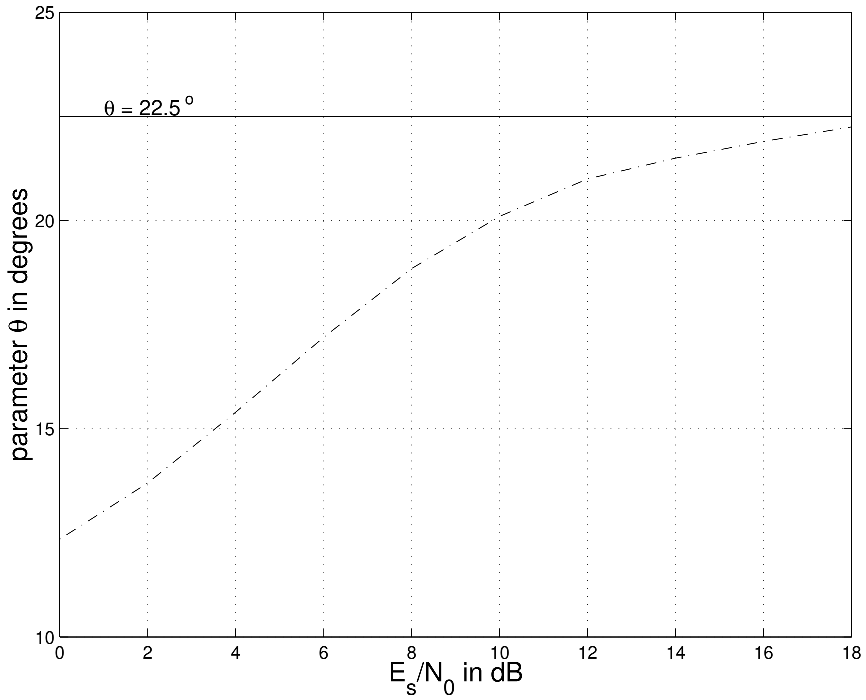


Figure 4.23: Parameter θ in degrees as a function of $\frac{E_s}{N_0}$ in dB. Asymmetric 8-PSK constellation. Block partitioning is applied.

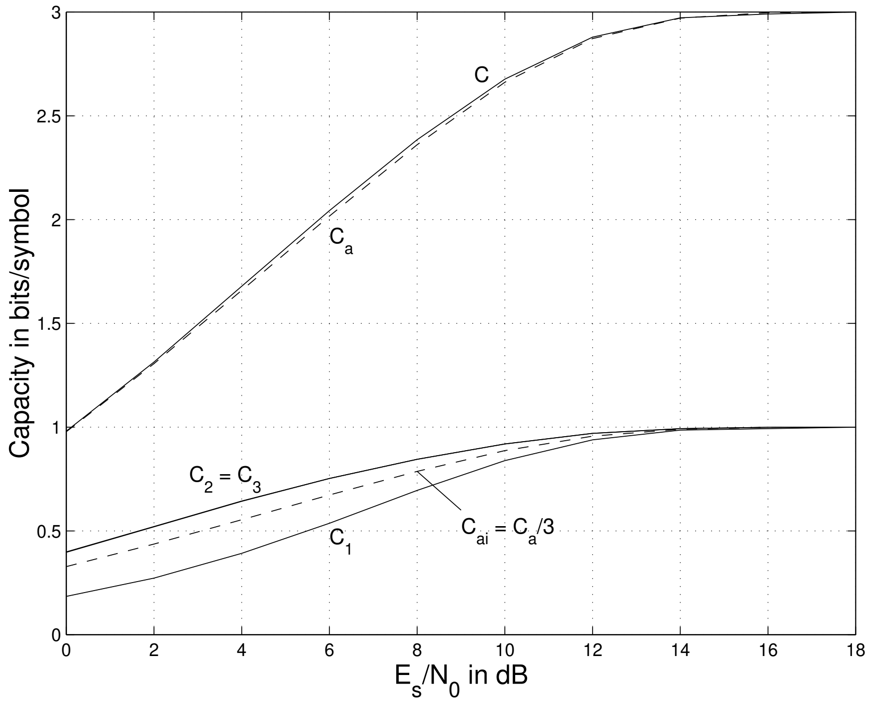
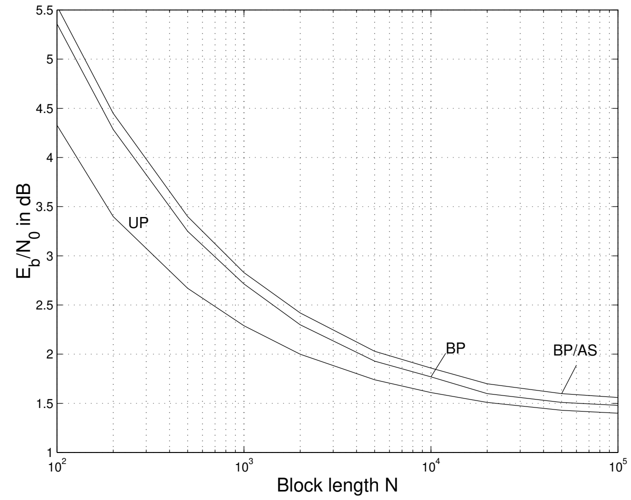


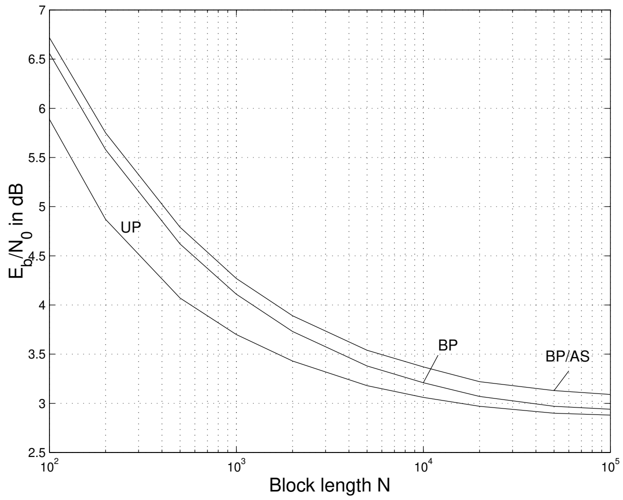
Figure 4.24: Capacities C and C_a of the symmetric and asymmetric, 8-PSK, capacities C_i and C_{ai}, $i = 1, 2, 3$ of their equivalent channels as a function of $\frac{E_s}{N_0}$ in dB. Block partitioning is applied.

For finite code length N, the degradation in performance of the asymmetric PSK is more significant. This can be verified analytically by computing Gallager's coding exponents. For rates equal to 1.5, 2 and 2.5 bits/symbol, figures 4.25.a., 4.25.b. and 4.25.c. show the average signal-to-noise ratio per bit $\frac{E_b}{N_0}$ needed to achieve a word error probability $P_e = 10^{-3}$ versus the code length N. Curves correspond to : Ungerboeck's partitioning and a symmetric constellation (UP), block partitioning with a symmetric constellation (BP) and block partitioning applied to an asymmetric constellation that has equal rate at each level (BP/AS).

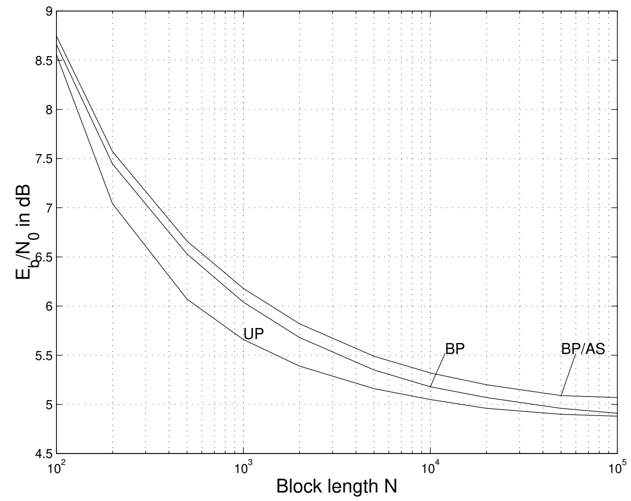
The same capacity equalization procedure described for an 8-PSK constellation can be applied to any $M = 2^L$-PSK signal set.



a. $R = 1.5$ bits/dimension.



b. $R = 2$ bits/dimension.



c. $R = 2.5$ bits/dimension.

Figure 4.25: Required $\frac{E_b}{N_0}$ in dB to achieve a word error probability $P_e = 10^{-3}$, as a function of the block length N. 8-PSK modulation.

4.10 Simulation results

In order to validate the results stated before, the performance of some MLC schemes over the AWGN channel is determined by Monte Carlo simulation. We consider in particular a 4-ASK with $R = 1.5$ bits/dimension, an 8-ASK with $R = 1.5$ and 2.5 bits/dimension, and an 8-PSK constellation with $R = 1$ bit/dimension ($R = 2$ bits/symbol). At each level, the encoder E_i is a punctured parallel turbo code of rate $R_i = R/L$ and block size $K_i = R_i.N$ where the code length N is equal to 2^{11} or 2^{17}. The recursive systematic convolutional constituent octal generators are $g = (31, 27)$. We consider both, Ungerboeck's partitioning of a symmetric constellation, and block partitioning of an asymmetric constellation with equal rate at each level. Multistage decoding is done. For the symmetric constellations, the rates R_i are chosen using the capacity design rule. For asymmetric constellations, the parameters $(\alpha_1, \alpha_2, \ldots, \alpha_{L-1})$ of the 2^L-ASK and the parameters (θ_1, θ_2) of the 8-PSK are chosen to minimize the bit error probability P_b of the multilevel coding system

$$P_b = \frac{1}{L}\sum_{i=1}^{L} P_{bi} \tag{4.28}$$

where P_{bi} is the bit error probability at level i.

In figure 4.26, we sketche the performance of an MLC scheme with different decoding strategies. We considered an asymmetric 8-ASK constellation. The total rate is $R = 1.5$ bits/dimension and the rate of the encoder E_i is $R_i = 1/2$. The block length is equal to $N = 2^{11}$. Results show that, even for small to moderate block length, multistage decoding, MSD with a posteriori probability propagation (APP) from each level, and MSD with iterative decoding of the L levels have similar performance. The performance of MSD without error propagation is also represented in figure 4.26. For this MLC scheme, the degradation in performance caused by error propagation is equal to 0.2 dB.

Figure 4.27 shows the performance of an MLC scheme based on a 4-ASK constellation. We considered both symmetric and asymmetric constellations. The total code rate is equal to $R = 1.5$ bits/dimension. This system is equivalent to a 16-QAM-based one with total rate $R = 3$ bits/symbol. For an 8-PSK, the performance is presented in figure 4.28. The total code rate is $R = 2$ bits/symbol.

Finally, the performance of different MLC schemes is presented in the bandwidth efficiency plane (see figure 4.29). Each point specifies the average signal-to-noise ratio per bit needed to achieve a bit-error probability P_b equal to 10^{-5}. These results show that performance close to Shannon limit is exhibited. For both symmetric Ungerboeck partitioned and asymmetric block partitioned constellations, the distance to the capacity of the underlying modulation transmitted over the AWGN channel is less than 1 dB. Moreover, the degradation between block and Ungerboeck's partitioning is close to the one found analytically by calculating Gallager's coding exponents specially for $L = 2$ levels. For $L = 3$, the degradation is a little bit greater. This is probably due to error propagation between the levels. On the other hand, a comparison is made with turbo trellis coded modulations introduced by Robertson and Wörz [69]. We considered Robertson's TCM (points of figure 4.29 plotted with a "+" symbol) based on a 4-ASK with rate $R = 1.5$ bits/dimension and block length $N = 5000$, an 8-ASK with $R = 2.5$ bits/dimension and $N = 3000$, and an 8-PSK with $R = 2$ bits/symbol and $N = 1024$ and 5000. We conclude from this comparison that the performance of MLC with Ungerboeck's partitioning and Robertson's TCM are almost identical. The small gain introduced by Robertson's TCM is probably due to the symbol-by-symbol interleaving that is applied instead of bit interleaving. The use of a turbo code or an irregular turbo code with symbol-based iterative decoding, as described in chapter2, will probably improve the performance of the MLC scheme and therefore, overcome this small degradation in performance.

4.11 Conclusion

The main result of this chapter is that multilevel coding, combined to multistage decoding, can reach the capacity of an ASK, a QAM or a PSK modulation transmitted over an additive white Gaussian noise channel using block partitioned asymmetric constellations with equal signaling rates at each level. Performance were derived analytically by calculating Gallager's coding exponents. The results showed that the degradation in performance, introduced since block partitioning was applied instead of Ungerboeck's partitioning, is relatively small. In the case of an ASK constellation, and for both Ungerboeck's and block partitioning, a gain in performance, and sometimes in capacity, was achieved by the asymmetric constellations. This can be viewed as a shaping gain introduced by these constellations. Moreover, for moderate signaling rates R and for large code length N, a block partitioned asymmetric ASK constellation which has equal capacity at each level, performed slightly better than an Ungerboeck partitioned symmetric one. These results were validated by Monte Carlo simulation of some MLC systems that use regular turbo codes as constituent codes. Performance close to the Shannon limit was reached.

Therefore, asymmetric constellations are well suited for multilevel coding schemes like the softly degrading scheme where decoding is done only to the level that delivers reliable data, or the unequal error protection (UEP) schemes. The complexity of the MLC system is largely reduced since only one architecture of encoding and decoding have to be implemented.

In the next chapter, We apply the principles of MLC to a multiple-input multiple-output flat fading channel. A MLC scheme based on a 2-level BICM is proposed for a *non-ergodic* block fading channel with 2 transmit and 2 receive antennas.

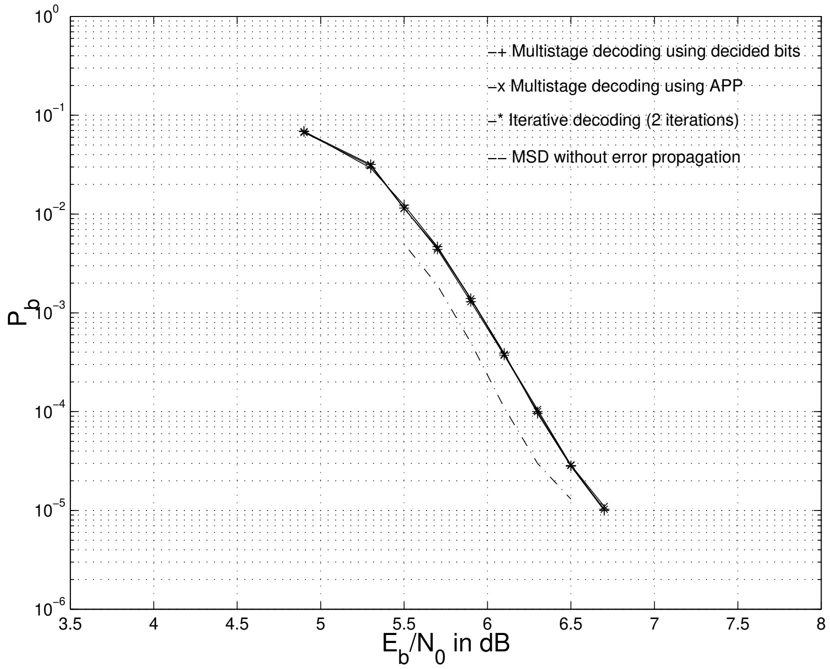


Figure 4.26: Bit error probability of an MLC with different decoding strategies. Block partitioned asymmetric 8-ASK. Total rate $R = 1.5$ bits/dimension. $N = 2^{11}$. AWGN channel.

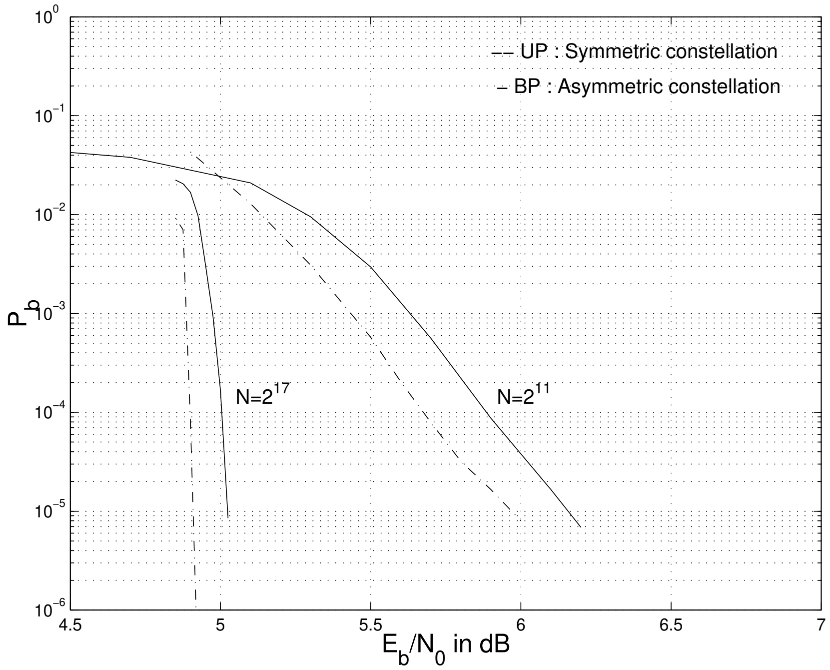


Figure 4.27: Bit error probability of a 4-ASK MLC scheme. $R = 1.5$ bits/ dimension. AWGN channel.

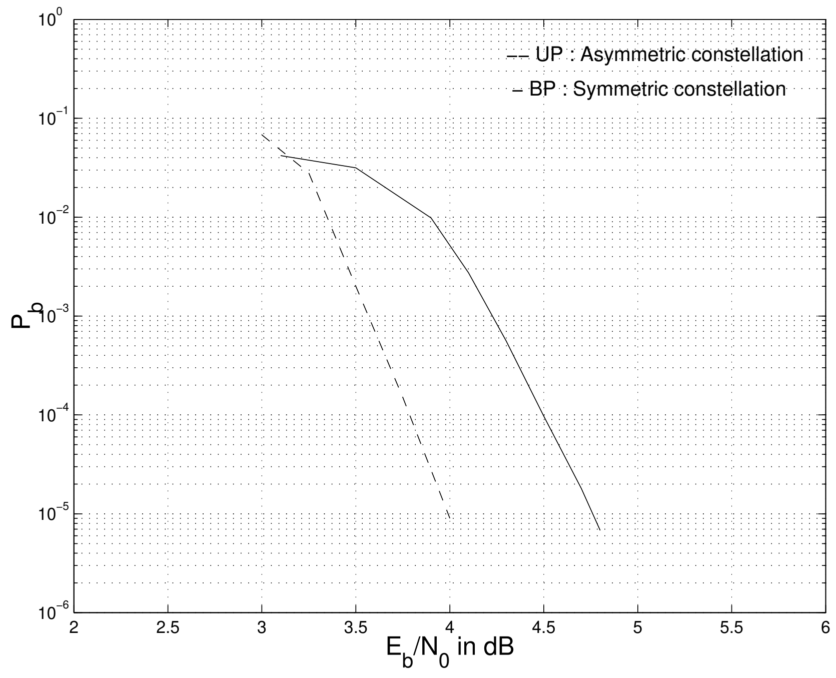


Figure 4.28: Bit error probability of an 8-PSK MLC scheme. $R = 1.5$ bits/ dimension. AWGN channel.

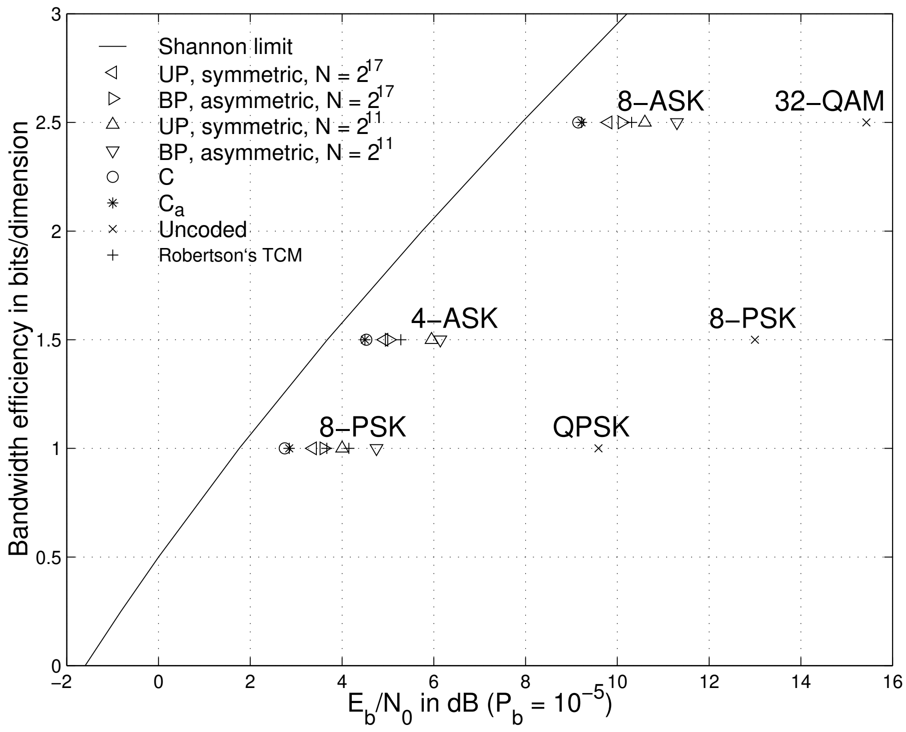


Figure 4.29: Simulation results for the AWGN channel, presented in the bandwidth efficiency plane.

Chapter 5

MLC in a multiple-antenna system

5.1 Introduction

The next generation of *wireless systems* are required to provide high quality and high data rates for wireless multimedia applications. However, achieving low error rates at high data transmission rates is extremely difficult over a *wireless multipath channel* that may suffer from sever attenuations.

The only practical way to improve the performance of such systems is by introducing some form of *diversity*. Several techniques, like time diversity, frequency diversity and antenna diversity, can be used to provide some replicas of the transmitted signal. We focus our study on *multiple-input multiple-output* (MIMO) channels that use multiple transmit and receive antennas.

We review some recent results regarding the capacity of MIMO channels [85] [31]. The capacity is also examined by formulating it in the context of multilevel coding as in chapter 4 [9]. We consider both *ergodic flat fading Rayleigh channels* and *non-ergodic block fading channels*. We assume that the channel coefficients between every pair of transmit and receive antenna are statistically independent and known by the receiver.

We propse two coding schemes based on *bit-interleaved coded modulations* [12] and multilevel coding. The performance of these systems is compared to the one we get using an *orthogonal design* [84] like the Alamouti scheme [1].

Notice that we didn't verify the effectiveness of the proposed systems for MIMO channels with very high data rates where more than two transmit antennas are used. For similar systems, the reader can consider using V-BLAST space-time codes [39] [40] [41] which achieves good performance using simple encoding and decoding schemes, or the *linear dispersion* space-time codes [43] that are designed to optimize the mutual information between the transmitted and the received signals. Finally, for unknown channel coefficients, the reader can refer to several studies mainly done by Marzetta and Hochwald [61] [44].

We begin this chapter by introducing the system model. In section 5.3, we review the calculation of the capacity of an ergodic Rayleigh flat fading channel. For a non-ergodic block fading channel, we determine in section 5.4 the outage probability which corresponds to the probability that the code rate is not supported by the channel. In section 5.5, we examine the capacity and the outage probability of a MIMO channel using multilevel coding. This is interesting for the design of a coding scheme for the MIMO channel. In section 5.6, we review the orthogonal designs and specially the Alamouti scheme which leads to significant improvement in performance using very simple encoding and decoding schemes. In order to achieve better performance, we study the effectiveness of bit-interleaved coded modulations in section 5.7. Both QPSK and 16-QAM constellations are considered. Finally, a two levels MLC scheme is applied to a 16-QAM constellation. At each level of the MLC, a bit-interleaved code is applied.

5.2 System model

Let us consider a flat fading multiple-input multiple-output (MIMO) channel with n_t transmit and n_r receive antennas (see figure 5.1). The received and the transmitted signals are related by

$$\mathbf{y} = \mathbf{H}.\mathbf{x} + \mathbf{n} \tag{5.1}$$

where $\mathbf{y} = (y_1, y_2, \ldots, y_{n_r})^t$ denotes the vector of complex received signals during any given channel use, $\mathbf{x} = (x_1, x_2, \ldots, x_j, \ldots, x_{n_t})^t$ denotes the vector of complex transmitted symbols. The superscript ^{t} stand for transpose. The superscript ^{h}, that will be used later, stands for transpose conjugate. The symbols x_j belong to a PSK or a QAM constellation of size $M = 2^m$. The channel matrix $\mathbf{H} = [h_{i,j}] \in \mathbb{C}^{n_r \times n_t}$ is assumed to be perfectly known at the receiver but not at the transmitter. The fading coefficients $h_{i,j} \in \mathbb{C}$ are Gaussian, circular, mutually independent and satisfy $E[|h_{i,j}|^2] = 1$. For an ergodic Rayleigh flat fading channel, the matrix $\mathbf{H}$ varies at each channel use. For a non-ergodic block fading channel, $\mathbf{H}$ is fixed in time. $\mathbf{n}$ denotes the vector of additive white complex Gaussian noise with zero mean and variance $2N_0$.

When T channel uses are considered, where T stands for time, equation (5.1) becomes

$$\mathbf{y} = \mathbf{H}.\mathbf{S} + \mathbf{n} \tag{5.2}$$

where $\mathbf{S}$ is a $n_t \times T$ matrix. Each column of $\mathbf{S}$ represents a transmit vector $\mathbf{x}$ at a given channel use. $\mathbf{y}$ and $\mathbf{n}$ are $n_r \times T$ channel matrices.

5.3 Capacity of an ergodic Rayleigh channel

We review in this section the calculation of the capacity of an *ergodic* Rayleigh flat fading MIMO channel [85]. Since $\mathbf{H}$ is known to the receiver and not to the

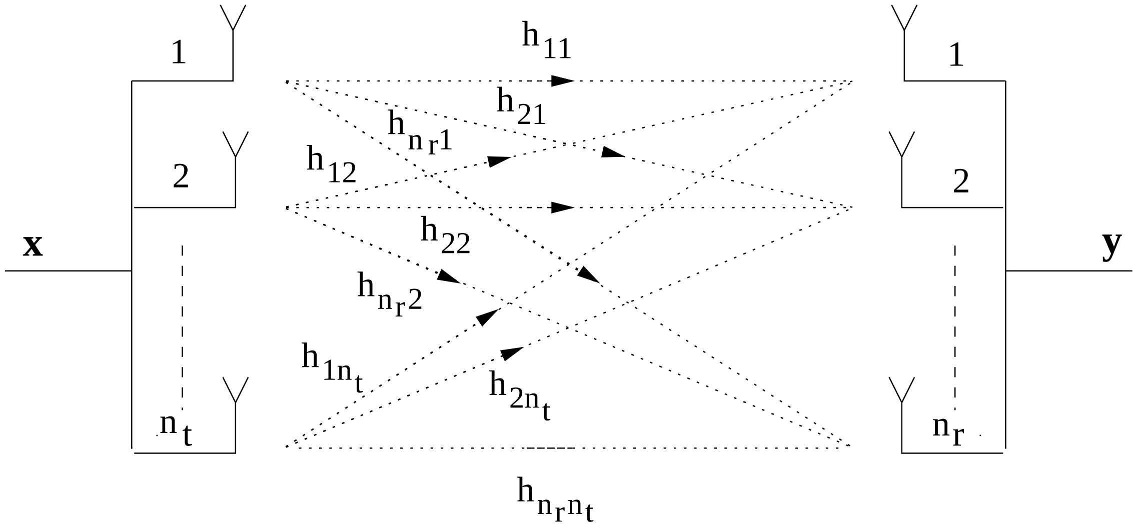


Figure 5.1: MIMO channel with n_t transmit and n_r receive antennas.

transmitter, it can be considered as part of the output. The capacity is therefore obtained by maximizing the mutual information $I(\mathbf{x};(\mathbf{y},\mathbf{H}))$. It can be shown after some simple manipulations that this quantity is equal to $E_{\mathbf{H}}\{I(\mathbf{x};\mathbf{y}|\mathbf{H}=H)\}$ where H denotes a realization of $\mathbf{H}$. The mutual information in this last expression is maximized when $\mathbf{x}$ is a zero mean complex circularly symmetric Gaussian vector (see [85] for the definition of a circularly symmetric random vector). This mutual information is given by

$$I_{\mathbf{R_x}}(\mathbf{x};\mathbf{y}|\mathbf{H}=H) = log_2(det(\mathbf{I_{n_r}} + \frac{1}{2N_0}\mathbf{H}\mathbf{R_x}\mathbf{H^h})) \tag{5.3}$$

where $\mathbf{R_x} = E\{\mathbf{x}\mathbf{x}^h\}$. Furthermore, the expectation which gives the capacity is maximum when the covariance matrix $\mathbf{R_x}$ is a multiple of identity. We shall write

$$\mathbf{R_x} = \frac{P}{n_t}\mathbf{I}_{n_t}$$

where obviously $P = E\{\mathbf{x^h}\mathbf{x}\}$ is the total transmitted energy per channel use. In short, the capacity on a Rayleigh fading channel is given by

$$C = E\{log_2(det(\mathbf{I_{n_r}} + \frac{P}{2N_0 n_t}\mathbf{H}.\mathbf{H^h}))\} \tag{5.4}$$

Since all elements $h_{i,j}$ are independent and satisfy $E\{|h_{i,j}|^2\} = 1$, the quantity $\rho = \frac{P}{2N_0}$ represents the SNR at each receiver antenna. Notice that C corresponds to the maximal number of information bits that can be transmitted per channel use $\mathbf{H}$ (one vector x is transmitted per channel use).

Figure 5.2 shows the capacity of some MIMO channels with continuous input as a function of the total signal-to-noise ratio at each receive antenna.

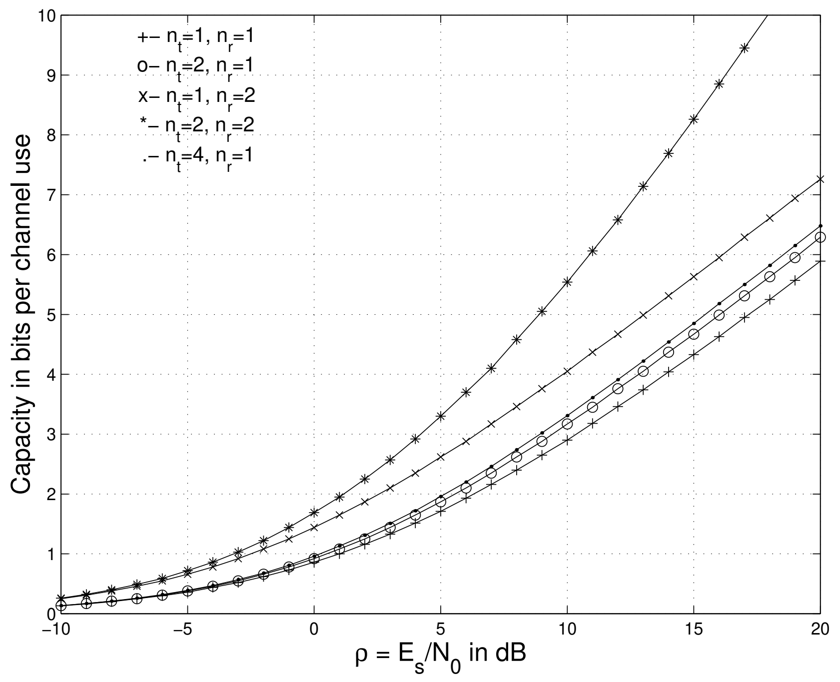


Figure 5.2: Capacity of a MIMO channel with continuous input as a function of the total signal-to-noise ratio at each receive antenna $\rho = \frac{E_s}{N_0}$ in dB.

Remarks

- For $n_t = n_r = 1$, we get the capacity of a single-input single-output Rayleigh fading channel

$$C = E_H\{log_2(1 + \rho|h_{11}|^2))\} \tag{5.5}$$

  where h_{11} is a complex scalar.

- The capacity increase with number of receive antennas. For example, for $n_t = 1$ transmit antenna, the capacity per channel use is equal to

$$C = E_H\{log_2(1 + \rho \sum_{i=1}^{n_r} |h_{i1}|^2))\} \tag{5.6}$$

  The coefficient $|h_{11}|^2$ that we have for $n_r = 1$ is replaced by the sum $\sum_{i=1}^{n_r} |h_{i1}|^2$ where the coefficients h_{i1} are complex independent Gaussian random variables. This is the effect of diversity on the capacity of the MIMO Rayleigh channel.

- Let us consider the case of $n_t = n_r$ parallel independent channels where the channel realization matrix is equal to the identity matrix ($\mathbf{H} = \mathbf{I_{n_r}}$). The capacity of this system is given by

$$C = \frac{n_t}{log(2)} log_2(1 + \frac{\rho}{n_t}) \tag{5.7}$$

For n_t sufficiently high, $C \rightarrow \frac{\rho}{ln(2)}$. On the other hand, the capacity is just equal to $C = log_2(1 + \rho)$ for $n_t = 1$. Therefore, the capacity for large value of n_t is not just only greater than the one we had for $n_t = 1$ but also increases linearly with the average signal-to-noise ratio ρ instead of logarithmic-ally as for $n_t = 1$. However, parallel independent channel is not feasible in practice.

- Tarokh [83] specifies that the capacity C grows at least linearly with the number of transmit antennas as long as the number of receive antennas is greater than or equal to the number of transmit antennas. From the results found by Foschini and Gans [31] regarding the capacity C, Tarokh also concluded that for one receive antenna little can be gained by using more than four transmit antennas. For two receive antennas, almost all the capacity increase can be obtained using $n_t = 6$ transmit antennas. These results can be validated using the capacity plots given by Telatar [85].

5.4 Outage probability of a non-ergodic block fading channel

The maximum mutual information has the meaning of capacity when the channel is memoriless *i.e.* when each use of the channel employs an independent realization of the channel matrix **H**.

For a non-ergodic block fading channel, **H** is chosen randomly at the beginning of all time and is held fixed for all the use of the channel, *i.e.* for the transmission of a whole code word **c**. In this case, there's always some realizations of **H** for which the capacity

$$C_H = log_2(det(\mathbf{I_{n_r}} + \frac{P}{2N_0 n_t}\mathbf{H}.\mathbf{H^h})) \tag{5.8}$$

of the corresponding Gaussian channel is smaller than the total code rate transmitted per channel use. A non-zero word error probability is obtained even if the code word length goes to infinity. Therefore, the capacity of the channel is equal to zero. The quantity to be used in order to evaluate the effectiveness of the channel is not its capacity anymore.

A good measure of the effectiveness of the channel is the *outage probability* P_{out}, that is, the probability that the code rate is not supported by the channel

$$P_{out} = Prob(C_H < R) \tag{5.9}$$

P_{out} is evaluated numerically by calculating, using equation (5.8), the capacity C_H for a large number of channel realizations **H**. For $R = 2$ and $R = 4$ bits per channel use, figure 5.3 shows the outage probability of a MIMO channel as a function

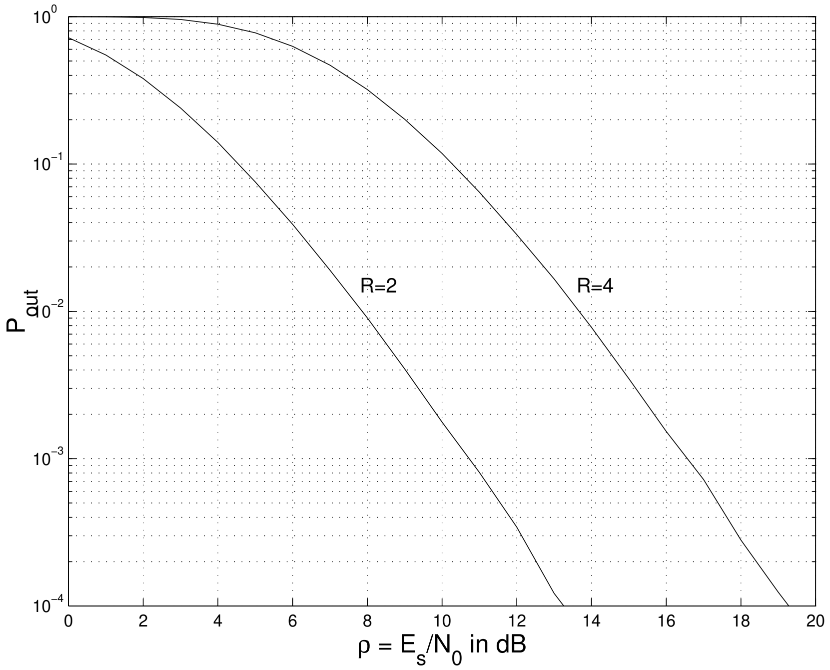


Figure 5.3: Outage probability of a continuous MIMO channel as a function of the average signal-to-noise ratio at each receive antenna $\rho = \frac{E_s}{N_0}$ in dB. $R = 2$ and $R = 4$ bits per channel use. $n_t = n_r = 2$.

of the total signal-to-noise ratio at each receive antenna. This curve is a lower bound on the word error probability of MIMO channels when coding is done.

5.5 Multiple antennas examined using MLC

In the previous chapter, we determined the capacity of a MLC scheme based on a M-ary constellation using the chain rule of mutual information which allows us to represent the transmission over the physical channel using L equivalent channels, one for each coded bit. The same procedure can be used in order to analyze a coded MIMO system in which a vector $\mathbf{c}$ of L coded bits or Q-ary symbols is mapped one-to-one onto the vector $\mathbf{x}$ transmitted using n_t antennas [9]. For a $M = 2^m$-ary modulation, we have

$$L = \frac{m.n_t}{log_2(Q)} \tag{5.10}$$

For example, let us consider $n_t = 2$ transmit antennas and a QPSK modulation ($M = 4$). The signal constellation observed by each receive antenna is a linear combination of two QPSK constellations. The number of signal points is equal to 16 (see figure 5.4). In the case of a quaternary code ($Q = 4$), The number of coded levels is $L = 2$, one level for each transmit antenna. Two equivalent channels are defined, one for each quaternary symbol. From the chain rule of mutual information we have

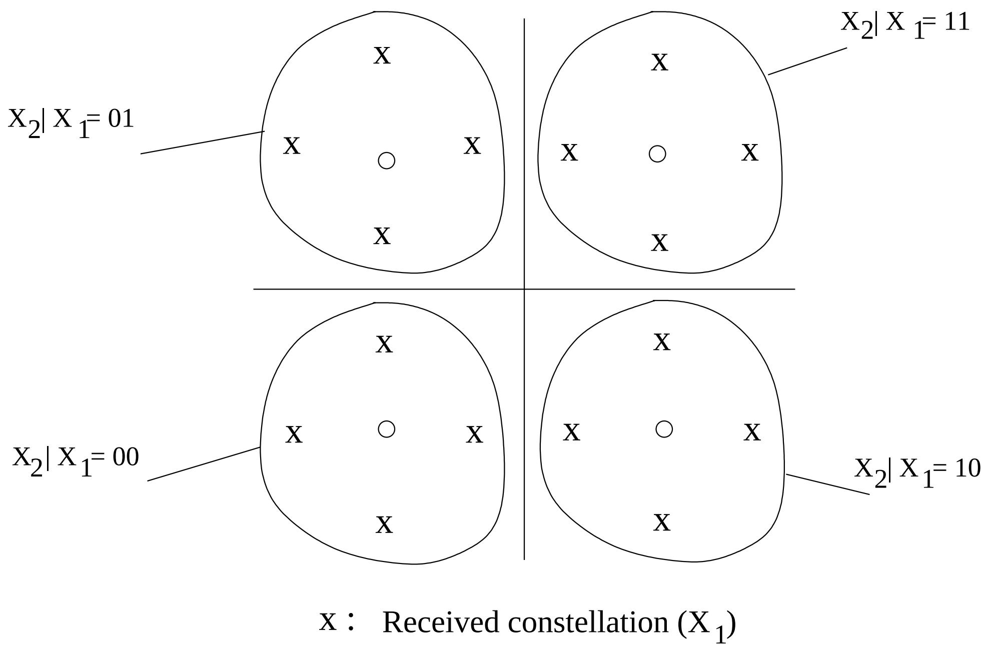


Figure 5.4: Example of receive constellation with QPSK and $n_t = 2$ transmit antennas.

$$I(X;Y) = I(X_1;Y) + I(X_2;Y|X_1) \tag{5.11}$$

X_1 is the random variable that represents the transmission of the first quaternary symbol over the first equivalent channel. X_2 represents the transmission of the second quaternary symbol. Multistage decoding can be done. We begin by decoding the first level that carries the first quaternary symbol. Assuming that the first quaternary symbol is known, we decode the second level. Notice that for the second level, the received constellation contains 4 points.

5.5.1 Capacity of the equivalent channels

Let us consider an ergodic Rayleigh fading channel. For a fixed realization $\mathbf{H}$, the output of the channel Y is a vector of Gaussian random variables.

$$p_H(\mathbf{y}|\mathbf{x}) \propto \exp -\frac{||\mathbf{y}-\mathbf{x}||^2}{N_0} \tag{5.12}$$

The calculation of the capacity of an equivalent channel is done as in section 4.3 of chapter 4 and by averaging over all the realizations of $\mathbf{H}$. We get

$$C_i = E_{\mathbf{H},x_1,\ldots,x_{i-1}}\{I(X_i,\ldots,X_L|(x_1,\ldots,x_{i-1});Y)\} - E_{\mathbf{H},x_1,\ldots,x_i}\{I(X_{i+1},\ldots,X_L|(x_1,\ldots,x_i);Y)\} \tag{5.13}$$

The total capacity of the discrete input MIMO channel is equal to the sum of the capacities of the L equivalent channels

$$C = \sum_{i=1}^{L} C_i \tag{5.14}$$

We represent in figure 5.5 the total capacity C and the capacities C_1 and C_2 of the equivalent channels of the ergodic MIMO channel with $n_t = n_r = 2$ and a QPSK constellation. These plots are obtained by Monte Carlo integration of 5.13 using 5000 noise samples and 5000 channel realizations of the fading channel matrix $\mathbf{H}$. We also represent the total capacity of the continuous input MIMO channel which is determined using equation (5.4).

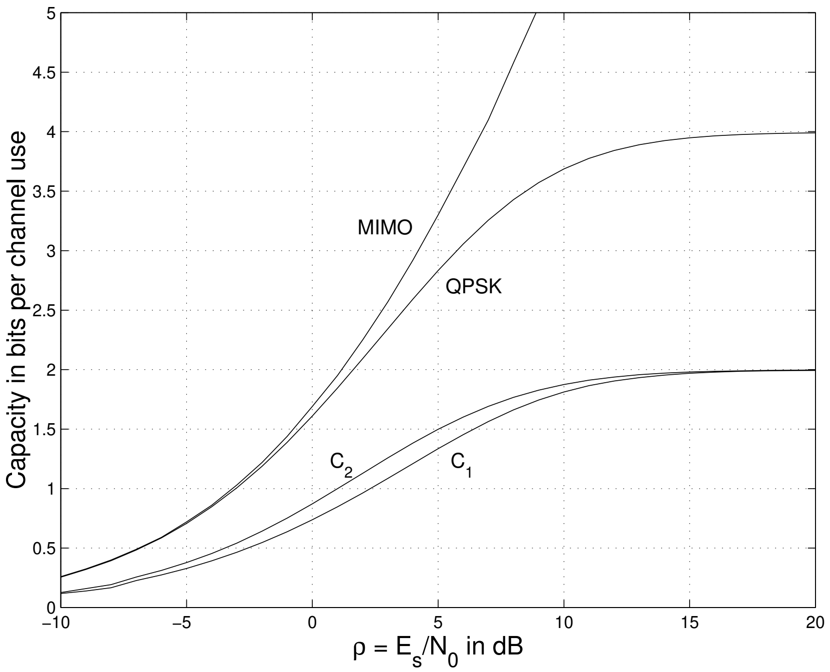


Figure 5.5: Total capacity C and capacities C_1 and C_2 of the equivalent channels of the MIMO system with a QPSK modulation and an ergodic Rayleigh fading channel. $n_t = n_r = 2$.

5.5.2 Capacity regions and rate design

The MIMO system can be considered as a multi-user system. The capacity region of the system can be determined as in section 4.4 of chapter 4. For two equivalent channels, the code rates R_1 and R_2 must satisfy

$$R_1 + R_2 \leq I((X_1, X_2); Y) \tag{5.15}$$

$$R_1 \leq I(X_1; Y|X_2), \quad R_2 \leq I(X_2; Y|X_1) \tag{5.16}$$

Points "A" and "B" of figure 4.8 correspond to the chain rule *i.e.* the capacity design rule. For these two points, the total capacity of the discrete input MIMO channel can be reached using Multistage decoding. For the points on the straight line connecting points "A" and "B", capacity can be reached using maximum likelihood decoding.

5.5.3 Outage probability

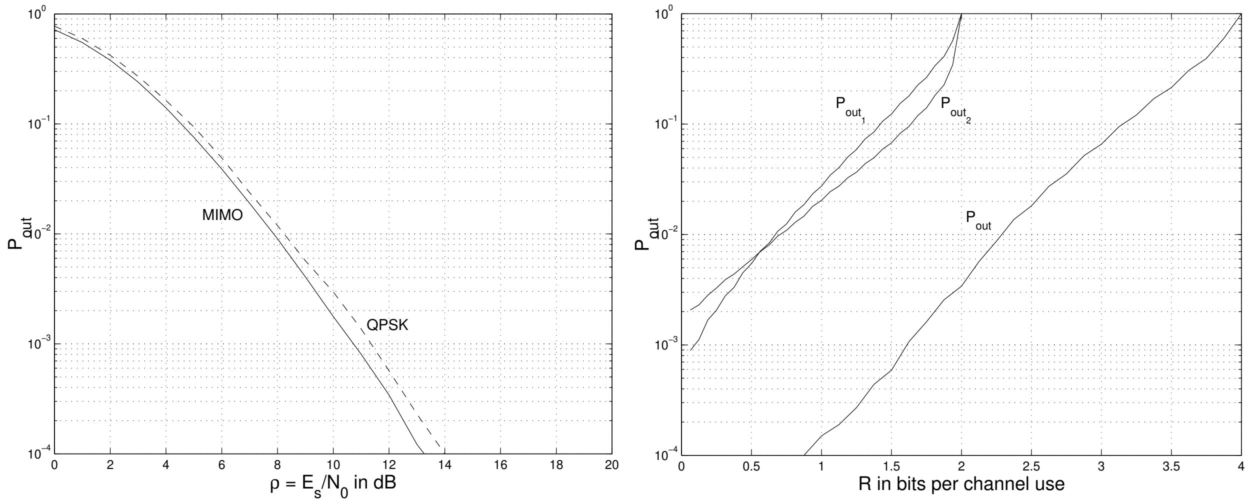
For a non-ergodic block fading channel, the outage probabilities of the equivalent channels are determined. When MSD decoding is applied, the outage probabilities P_{out_i} of the equivalent channels are defined by

$$P_{out_i} = Prob(R_i > C_i); \tag{5.17}$$

where $C_i = I((X_i, \ldots, X_L); Y|(X_1, \ldots, X_{i-1}))$ is determined using equation (5.13). The outage probability of the discrete input MIMO channel is equal to

$$P_{out} = Prob(R > C); \tag{5.18}$$

Figure 5.6 shows the outage probabilities of the equivalent channels and the MIMO channel considered previously, as a function of the code rate. Monte Carlo integration was performed. The average SNR is equal to 10 dB. These curves can be used in order to determine the code rates R_i. For a fixed outage probability $P_{out_1} = P_{out_2}$, the code rates R_1 and R_2 are determined. The total code rate will be $R = R_1 + R_2$. This value is less or equal to the one we get from equation (5.18). For example, for $P_{out_1} = P_{out_2} = P_{out} = 10^{-1}$, we have $R_1 = 1.35$, $R_2 = 1.65$ and $R = R_1 + R_2 = 3$ bits per channel use instead of $R = f^{-1}(P_{out}) = 3.2$ bits.



a. $R = 2$ bits per channel use.

b. $\rho = \frac{E_s}{N_0} = 10$ dB.

Figure 5.6: Outage probability of a QPSK. $n_t = n_r = 2$.

5.6 Orthogonal designs

In this section, we review the orthogonal designs that were originally proposed by Alamouti in 1998 [1] and later generalized by Tarokh in 1999 [84]. The resulting space-time block codes are easily decoded by a maximum-likelihood decoding algorithm which is based only on linear processing at the receiver.

Alamouti's orthogonal design is a simple two-branch transmit diversity scheme. For $n_t = 2$ transmit antennas and $n_r = 1$ receive antenna, the scheme provides a diversity order equal to 2 as for a maximal-ratio receiver combining (MRRC) with $n_t = 1$ transmit antenna and $n_r = 2$ receive antennas. Alamouti also showed that the scheme may be generalized to a $n_t = 2$ transmit antennas and n_r receive antennas scheme that provides a full diversity order of $2n_r$. We first study the MRRC scheme.

5.6.1 Maximal-ratio receiver combining

Consider a MIMO channel with $n_t = 1$ transmit antenna and $n_r = 2$ receive antennas. The received signal is $\mathbf{y} = \mathbf{H}.\mathbf{x} + \mathbf{n}$ where $\mathbf{y} = (y_1, y_2)$, $\mathbf{H} = (h_{11}, h_{21})^t$ and $\mathbf{x} = (x_1)$. We have

$$y_1 = h_{11}.x_1 + n_1 \tag{5.19}$$

$$y_2 = h_{21}.x_1 + n_2 \tag{5.20}$$

The ML decoder consists in finding the vector $\mathbf{x}$ that minimizes the euclideanne distance $||\mathbf{y} - \mathbf{H}.\mathbf{x}||^2 = |y_1 - h_{11}.x_1|^2 + |y_2 - h_{21}.x_1|^2$.

The MRRC scheme for two-branches computes the output $\tilde{x}$ as follows

$$\tilde{x} = h_{11}^*.y_1 + h_{21}^*.y_2 \tag{5.21}$$

we get

$$\tilde{x} = (|h_{11}|^2 + |h_{21}|^2).x_1 + h_{11}^*.n_1 + h_{21}^*.n_2 \tag{5.22}$$

$\tilde{x}$ is delivered to the ML decoder which finds the vector $\mathbf{x}$, i.e the symbol x_1 that minimizes the euclideanne distance $d(\tilde{x}, x_1)$. The diversity of the two-branch MRRC scheme is equal to 2.

5.6.2 Alamouti scheme

The transmitter in a space-time block code based on orthogonal design is represented in figure 5.7.

The "orthogonal design" block transforms the n_t symbols of a M-ary modulation into a $n_t \times T$ matrix $\mathbf{S}$. T stands for time. At each channel use, a column of the matrix $\mathbf{S}$ is transmitted by the n_t transmit antennas. The efficiency of the system is equal to $log_2(M).\frac{n_t}{T}$ bits per channel use. The most simple example is the one proposed by Alamouti [1] where $n_t = T = 2$ and S is given by

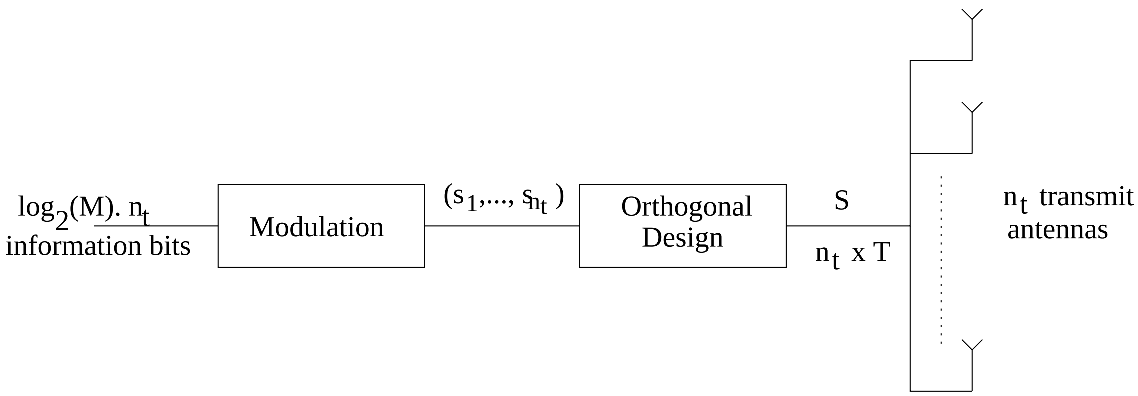


Figure 5.7: Transmitter in a space-time block code based on orthogonal design.

$$\mathbf{S} = \begin{bmatrix} s_1 & -s_2^* \\ s_2 & s_1^* \end{bmatrix} \tag{5.23}$$

This matrix is known as the "Alamouti matrix".

a) Alamouti scheme with $n_r = 1$ receive antenna

With $n_r = 1$, equation (5.2) becomes

$$\begin{bmatrix} y_1 & y_2 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} \end{bmatrix} . \begin{bmatrix} s_1 & -s_2^* \\ s_2 & s_1^* \end{bmatrix} + \begin{bmatrix} n_1 & n_2 \end{bmatrix} \tag{5.24}$$

where y_1 and y_2 are the received signals at the first and the second channel use. Equation (5.24) can be written as

$$\begin{bmatrix} y_1 \\ y_2^* \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} \\ h_{12}^* & -h_{11}^* \end{bmatrix} . \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} + \begin{bmatrix} n_1 \\ n_2^* \end{bmatrix} \tag{5.25}$$

We get a MIMO system that can be represented by the following equation

$$\mathbf{y_a} = \mathbf{H_a}.\mathbf{x_a} + \mathbf{n_a} \tag{5.26}$$

where $\mathbf{y_a} = (y_1, y_2^*)^t$, $\mathbf{x_a} = (s_1, s_2)^t$, $\mathbf{n_a} = (n_1, n_2*)^t$ and $\mathbf{H_a}$ is the equivalent channel matrix which verifies

$$\mathbf{H_a}.\mathbf{H_a^*} = \mathbf{H_a^*}.\mathbf{H_a} = (|h_{11}|^2 + |h_{12}|^2)\mathbf{I_2} \tag{5.27}$$

Hence, we just need to multiply $\mathbf{y_a}$ by $\mathbf{H_a^*}$ in order to get independent observations for s_1 and s_2

$$\tilde{\mathbf{y}}_\mathbf{a} = \mathbf{H_a^*}.\mathbf{y_a} = (|h_{11}|^2 + |h_{12}|^2).\mathbf{x_a} + \tilde{\mathbf{n}}_\mathbf{a} \tag{5.28}$$

This equation can be separated into two equations

$$\tilde{y}_{a1} = (|h_{11}|^2 + |h_{12}|^2).s_1 + \tilde{n}_{a1} \tag{5.29}$$

$$\tilde{y}_{a2} = (|h_{11}|^2 + |h_{12}|^2).s_2 + \tilde{n}_{a2} \tag{5.30}$$

Maximum likelihood decoding of s_1 and s_2 is therefore decoupled and the diversity is equal to 2 as for the MRRC scheme. On the other hand, the capacity of this orthogonal design can be determined from equation (5.4) by replacing the matrix $\mathbf{H}$ by the matrix $\mathbf{H_a}$ and by multiplying the result by a factor $\frac{1}{T} = \frac{1}{2}$ that is due to the orthogonal design. We find

$$\begin{aligned} C_a &= \frac{1}{2} E_{\mathbf{H}}\{log_2(det(\mathbf{I_2} + \frac{\rho}{2}(|h_{11}|^2 + |h_{12}|^2))\mathbf{I_2})\} \\ &= E_{\mathbf{H}}\{log_2(1 + \frac{\rho}{2}(|h_{11}|^2 + |h_{12}|^2))\} \end{aligned} \tag{5.31}$$

This is equal to the full channel capacity C of a MIMO channel with $n_t = 2$ transmit antennas and $n_r = 1$ receive antenna. Therefore, the orthogonal design is optimal not only from a diversity point of view but also from a capacity point of view.

b) Alamouti scheme with $n_r = 2$ or more receive antennas

The same calculations can be derived as for $n_r = 1$. For example , with $n_r = 2$, equation (5.2) becomes

$$\begin{bmatrix} y_{11} & y_{12} \\ y_{21} & y_{22} \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} . \begin{bmatrix} s_1 & -s_2^* \\ s_2 & s_1^* \end{bmatrix} + \begin{bmatrix} n_{11} & n_{12} \\ n_{21} & n_{22} \end{bmatrix} \tag{5.32}$$

which can be organized as

$$\mathbf{y_a} = \mathbf{H_a}.\mathbf{x_a} + \mathbf{n_a} \tag{5.33}$$

where $\mathbf{y_a} = (y_{11}, y_{12}^*, y_{21}, y_{22}^*)^t$, $\mathbf{x_a} = (s_1, s_2)^t$, $\mathbf{n_a} = (n_{11}, n_{21}^*, n_{12}, n_{22}^*)$ and the channel matrix $\mathbf{H_a}$ is equal to

$$\mathbf{H_a} = \begin{bmatrix} h_{11} & h_{12} \\ h_{12}^* & -h_{11}^* \\ h_{21} & h_{22} \\ h_{22}^* & -h_{21}^* \end{bmatrix} \tag{5.34}$$

$\mathbf{H_a}$ verifies

$$\mathbf{H_a}.\mathbf{H_a^*} = \mathbf{H_a^*}.\mathbf{H_a} = (|h_{11}|^2 + |h_{21}|^2 + |h_{12}|^2 + |h_{22}|^2)\mathbf{I_2} \tag{5.35}$$

$$\Rightarrow \tilde{\mathbf{y}}_{\mathbf{a}} = \mathbf{H_a^*}.\mathbf{y_a} = (|h_{11}|^2 + |h_{21}|^2 + |h_{12}|^2 + |h_{22}|^2).\mathbf{x_a} + \tilde{\mathbf{n}}_{\mathbf{a}} \tag{5.36}$$

A diversity order of 4 is reached. This is the full diversity of a MIMO system with $n_t = 2$ transmit and $n_r = 2$ receive antennas.

In general, for n_r received antennas, Alamouti scheme leads to a full diversity order of $2n_r$. However, unlike $n_r = 1$, the orthogonal design structure prohibits us from achieving capacity. To verify this, we compare the capacity C_a of the Alamouti scheme with the actual channel capacity of the MIMO system.

$$
\begin{aligned}
C_a &= \frac{1}{2}E_{\mathbf{H}}\{log_2(det(\mathbf{I_2} + \frac{\rho}{2}(|h_{11}|^2 + |h_{21}|^2 + |h_{12}|^2 + |h_{22}|^2))\mathbf{I_2})\} \\
&= E_{\mathbf{H}}\{log_2(1 + \frac{2\rho}{4}(|h_{11}|^2 + |h_{21}|^2 + |h_{12}|^2 + |h_{22}|^2))\} \\
&= E_{\mathbf{H}}\{log_2(1 + \frac{\rho'}{4}(|h_{11}|^2 + |h_{21}|^2 + |h_{12}|^2 + |h_{22}|^2))\} \qquad (5.37)
\end{aligned}
$$

This is the channel capacity of a MIMO system with $n_t = 4$ transmit antennas and $n_r = 1$ receive antenna at a signal-to-noise ratio $\rho' = 2\rho$. This capacity is smaller than the one we have for a MIMO channel with $n_t = n_r = 2$. Thus, when $n_r = 2$, Alamouti scheme is sub-optimal from a capacity point of view. The amount of loss in capacity is high at high signal-to noise ratios (see figure 5.8).

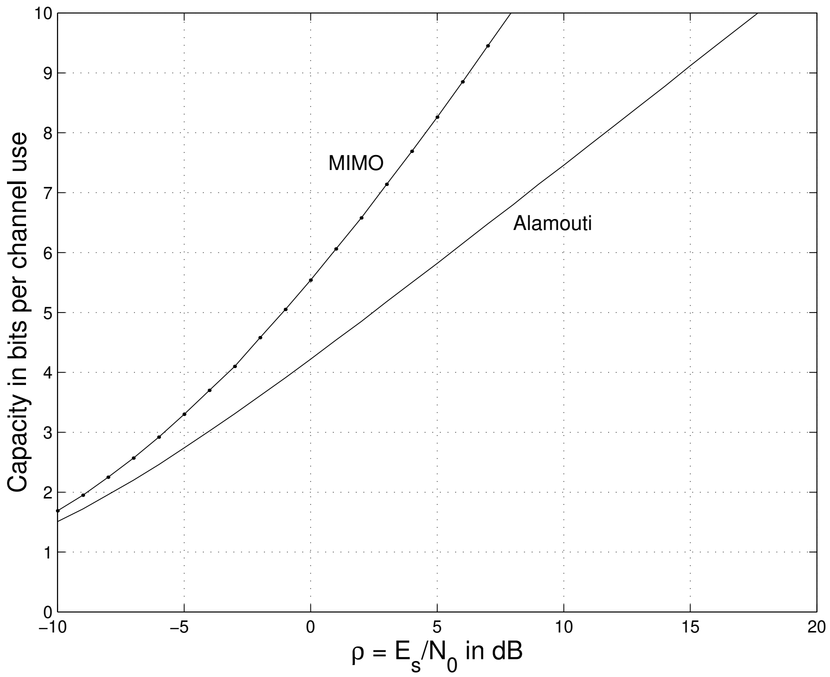


Figure 5.8: Capacity of the Alamouti scheme compared to the total capacity of a MIMO channel. $n_t = n_r = 2$.

For $n_r > 2$, similar results can be reached. We simply state that for $n_t = 2$ transmit antennas and n_r receive antennas, Alamouti scheme allows us to attain only the capacity of a MIMO system with $n_t = 2n_r$ transmit antennas and $n_r = 1$ receive antenna at a signal-to-noise ratio $\rho' = n_r\rho$.

5.6.3 Other orthogonal designs

In the previous section, we studied orthogonal designs with $n_t = 2$ transmit antennas. However, orthogonal designs also exist for $n_t > 2$ [84]. For example. for $n_t = 3$ and $T = 4$, we have the orthogonal design obtained using

$$\mathbf{S} = \begin{bmatrix} s_1 & -s_2^* & -s_3^* & 0 \\ s_2 & s_1^* & 0 & -s_3^* \\ s_3 & 0 & s_1^* & s_2^* \end{bmatrix} \tag{5.38}$$

It can be shown that the maximum likelihood decoding of the variables s_1, s_2, s_3 is decoupled. Using an argument similar to the one presented for $n_t = 2$, we can show that the capacity C_a for n_r receive antennas is equal to $\frac{3}{4}C$ where C is the full capacity of a MIMO system with $3n_r$ transmit antennas and 1 receive antenna at a signal-to-noise ratio $\rho' = n_r\rho$. C_a is much less than the true channel capacity. In fact the complex orthogonal designs for $n_t = 3$ are no longer "full rate" [84]. They generally perform poorly in the capacity that they can achieve, even when $n_r = 1$.

In section 5.7, we improve the performance of a MIMO system with $n_t = n_r = 2$, using bit-interleaved code modulations and multilevel coding.

5.7 Bit-interleaved coded modulations for MIMO systems

The study of the Alamouti scheme in the case of a MIMO channel with $n_t = 2$ transmit and $n_r = 2$ receive antennas showed that, although full-diversity is reached, the capacity of the scheme is largely reduced specially at high signal-to-noise ratios. In this section, we apply Bit-interleaved coded modulation to a MIMO system in order to achieve better performance [12].

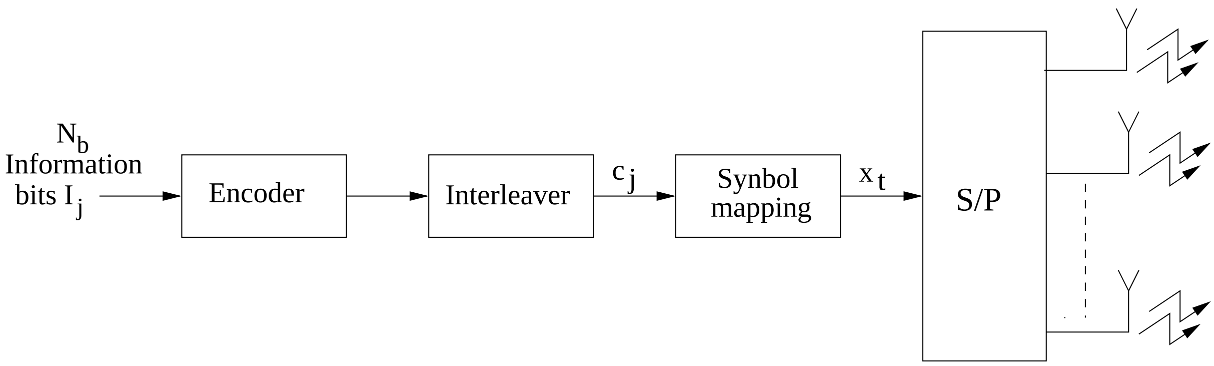


Figure 5.9: BICM transmitter.

The transmitter structure is illustrated in figure 5.9. The information bits $\mathbf{I} = (I_1, I_2, \ldots, I_{N_b})$ are encoded into a vector $\mathbf{c} = (c_1, c_2, \ldots, c_{N_c})$ of N_c coded bits which are then randomly interleaved and mapped into $M = 2^m$-ary PSK or QAM symbols x_t. The block of $N_s = N_c/m$ symbols to be transmitted is then divided into sub-blocks of length n_t and sent in parallel by the n_t transmit antennas. At every time index *i.e.* every channel use, the signal vector is function of $m \times n_t$ coded bits.

The motivation to this work is that, for most signal constellations with Gray labeling, the capacity of a BICM over a Rayleigh flat fading channel is almost equal to the capacity of the signal constellations themselves. This result was found by Caire *et al* [13] [14] who considered a Rayleigh fading channel with only one transmit and one receive antenna[1]. Results are derived in the next section for MIMO systems with more than one transmit and/or one receive antenna.

5.7.1 Capacity of a MIMO system with BICM

The calculation of the capacity is done under the assumption that infinite interleaving is realized at the encoder output. In this case, the capacity of a BICM is equal to the sum of the capacities of the $L = m.n_t$ equivalent channels of the MIMO system. This is the capacity of a L-level MLC scheme with parallel independent decoding (cf. chapter 4, section 4.6.3)

$$C_{BICM} = C_{PDL} = \sum_{i=1}^{L} C_i \tag{5.39}$$

where $C_i = \max_{P(X_i)} I(Y; X_i)$. X_i is the random variable that represents the transmission of the coded bit number i over the corresponding equivalent channel. The transmission of the L coded bits is represented by the random variable $X = (X_1, \ldots, X_i, \ldots, X_L)$. Notice that, form the chain rule of mutual information, we have

$$I(Y; X_i) = I(Y; X) - I(Y; X|X_i) \tag{5.40}$$

For example, in case of a *Gray labeled* QPSK constellation and $n_t = n_r = 2$ transmit and receive antennas, the symmetry in the constellation leads to $L = 4$ equivalent channels that have the same capacity *i.e.* $C_1 = C_2 = C_3 = C_4$. We get

$$C_{BICM} = 4C_1 = 4 \left(\max_{P(X_1)} I(Y; X_1) \right) \tag{5.41}$$

with

$$I(Y; X_1) = I(Y; X) - I(Y; X|X_1) \tag{5.42}$$

Results are represented in figure 5.11.a. The additional signal-to-noise ratio needed to achieve the same capacity of the MIMO system with discrete QPSK input is almost equal to 0.75 dB for $C = 2$ bits per channel use.

[1] We verified in chapter 4 that this result is also true for an AWGN channel.

Now let us consider a 16-QAM constellation ($n_t = n_r = 2$). Analyzing this constellation, we observe that the $L = 8$ equivalent channels can be divided into two classes. Each class contains 4 equivalent channels that have the same capacity. The first one contains the 4 "Good" equivalent channels of the better protected bits of the two 16-QAM symbols. Those bits are denoted by "a" in figure 5.10. "Bad" equivalent channels are those of the less protected bits labeled by "b" (the minimum euclideanne distance is the same for "Good" and "Bad" equivalent channels. However, for the "Good" equivalent channels, the kissing number is equal to 1/2 instead of 1 as for the "Bad" equivalent channels).

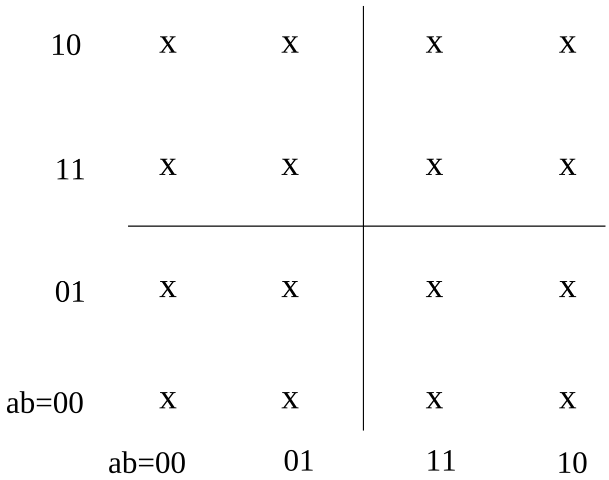


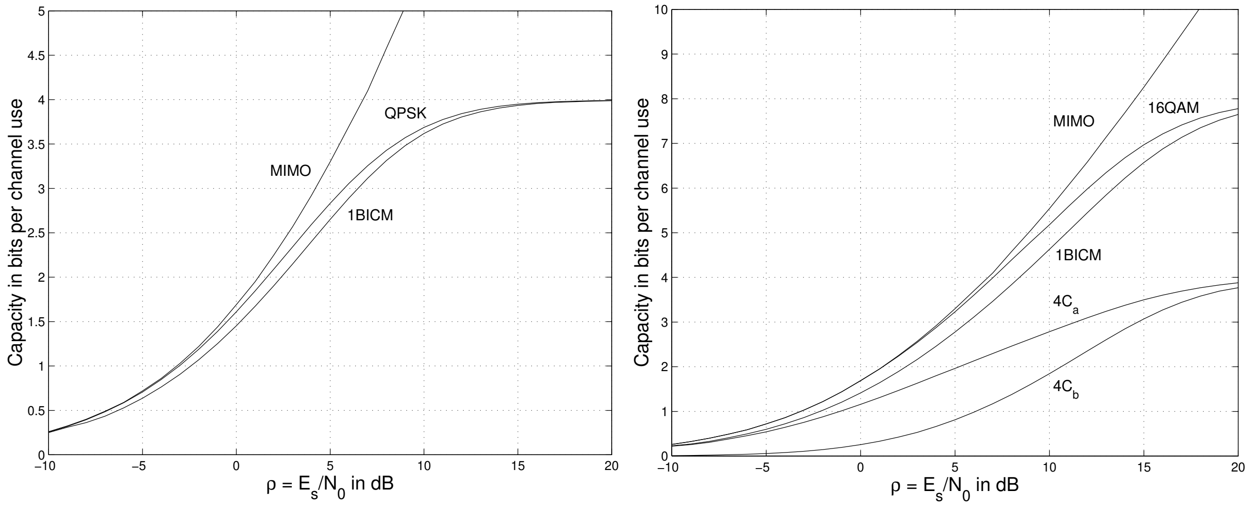
Figure 5.10: 16-QAM constellation. a → better protected bits. b → less protected bits.

The random variable X is now denoted $X = (X_a, X_b)$ where $X_a = (X_{a1}, \ldots, X_{a4})$ and $X_b = (X_{b1}, \ldots, X_{b4})$ respectively represent the transmission over the "Good" and the "Bad" equivalent channels. Due to symmetry, the 4 "Good" equivalent channels have the same capacity. This also true for the "Bad" equivalent channels. The total capacity of the MIMO system is therefore equal to

$$C_{BICM} = 4(C_a + C_b) \tag{5.43}$$

where $C_a = \max_{P(X_a)} I(Y; X_a)$ and $C_b = \max_{P(X_b)} I(Y; X_b)$ ($C_a > C_b$). Results are sketched in figure 5.11.b. The degradation in signal-to-noise ratio of this system over the MIMO system with 16-QAM input is equal to 1.4 dB for $C = 4$ bits per channel use.

Iterative decoding can significantly reduce this degradation. In fact, in this case, decoding is done with some knowledge, specially at high iterations, of the coded bits at the different levels of the modulation. This can be viewed as a sub-implementation of multistage decoding.



a. QPSK. b. 16-QAM.

Figure 5.11: Capacity of a MIMO channel with BICM. $n_t = n_r = 2$.

5.7.2 Iterative detection and APP decoding

To recover the binary stream, soft information on the coded bits needs to be extracted from the received signals, or more precisely from the contribution of each transmit antenna. Given the whole received signal over all time index $k = 1, \ldots, N_c/(m.n_t)$, and since coded bits have been randomly interleaved, it is possible to compute the APP of c_j which can be expressed as

$$\begin{aligned} APP(c_j) &= p(c_j|\mathbf{y}) \\ &= \frac{p(\mathbf{y}|c_j).\pi(c_j)}{p(\mathbf{y})} \quad j = 1, \ldots, m.n_t \end{aligned} \tag{5.44}$$

$$APP(c_j) \propto \pi(c_j).p(\mathbf{y}|c_j) = \pi(c_j).obs(c_j) \tag{5.45}$$

where $\pi(c_j)$ is the a priori probability of the bit c_j and the observation $obs(c_j) = p(\mathbf{y}|c_j)$. The conditional probability density $p(\mathbf{y}|c_j)$ is determined by marginalizing the joint density of all bits and the observation when taking into account that the received signal y_r are independent conditionally to the coded bits $c_1, \ldots, c_{m.n_t}$

$$\begin{aligned} p(\mathbf{y}|c_j) &= \sum_{c_i, i=1,\ldots,m.n_t, i\neq j} p(\mathbf{y}, c_1, \ldots, c_{j-1}, c_{j+1}, \ldots, c_{m.n_t}|c_j) \\ &= \sum_{c_i, i=1,\ldots,m.n_t, i\neq j} p(\mathbf{y}|c_1, \ldots, c_{m.n_t}) \prod_{l\neq j} \pi(c_l) \qquad (5.46) \\ &= \sum_{c_i, i=1,\ldots,m.n_t, i\neq j} \left(\prod_{r=1}^{n_r} p(y_r|c_1, \ldots, c_{m.n_t}) \prod_{l\neq j} \pi(c_l) \right) \qquad (5.47) \end{aligned}$$

The conditional density $p(y_r|c_1, \ldots, c_{m.n_t})$ is calculated using

$$p(y_r|c_1,\ldots,c_{m.n_t}) = \frac{e^{-\frac{||y_r-\sum_{t=1}^{n_t} h_{t,r}x_t||^2}{2\sigma^2}}}{2\pi\sigma^2} \tag{5.48}$$

The evaluation of the conditional likelihood $p(\mathbf{y}|c_j) = obs(c_j)$ corresponds to the detection stage located at the receiver front. This likelihood is then processed by a soft-input soft-output decoder. The SISO decoder generates an extrinsic information $Ext(c_j)$ which is equivalent to a new a priori probability $\pi(c_j)$ for the coded bit c_j. Hence, it is convenient to feed this a priori information back to the likelihood detector defined by equation (5.47). We get an iterative detection and decoding system which is an excellent way to improve the estimation of the a posteriori probabilities.

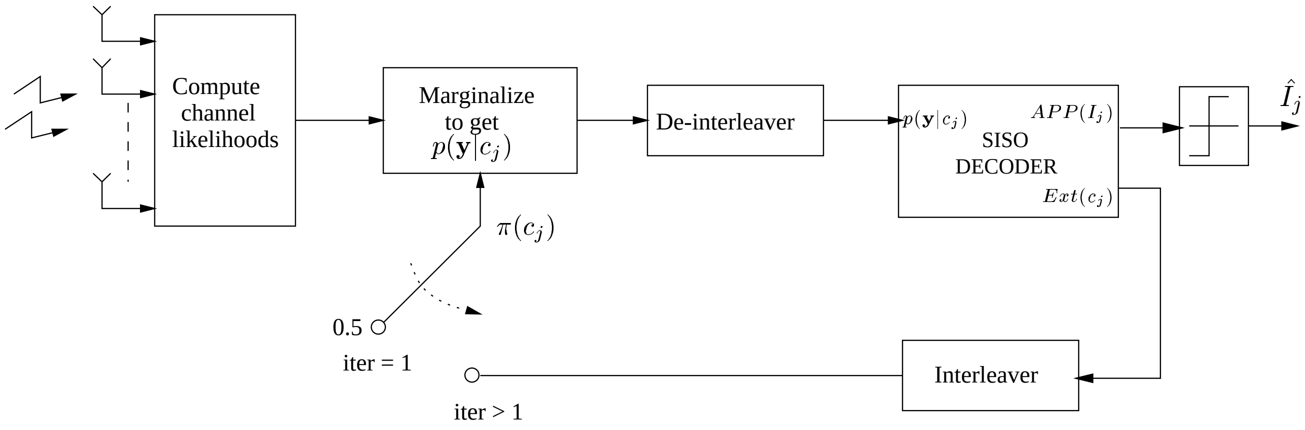


Figure 5.12: BICM iterative decoder.

Figure 5.12 illustrates the iterative detection and decoding receiver structure. Note that the receiver is separated into two parts: the first part is non iterative and computes the received signal conditional probability at every antenna r according to equation (5.48). The second receiver part is iterative and its input depends also on the a priori probabilities. The final decision is made out of the a posteriori probability generated by the SISO decoder at the last iteration.

5.7.3 Numerical results

We consider a BICM based on a rate-1/2, 16 state non-recursive non-systematic convolutional code with octal generators $g = (31, 27)$ and total frame length $N_c = 200$ bits when QPSK constellation is used, and $N_c = 400$ bits for a 16-QAM constellation. The trellis of the convolutional code is terminated to zero state. For both constellations, the number of coded symbols is $N_s = 100$.

Figures 5.16.a. and 5.16.b. represents the frame error rate (FER) of the BICM transmitted over a *non-ergodic* block fading channel. For a QPSK constellation, with a total rate $R = 2$ bits per channel use and $FER = 10^{-3}$, performance is at 2.1 dB from the performance limit of a MIMO channel with QPSK input. This corresponds to a gain of 3 dB over the performance of the Alamouti scheme that has the same code rate and the same frame length $N_s = 100$ symbols. Notice that better performance can be reached using a more complex convolutional code like the 64 state code with octal generators $g = (115, 117)$ [12]. For the 16-QAM constellation with $R = 4$ bits per channel use (see figure 5.16.b.), the gain over the Alamouti scheme is almost equal to 3.5 dB.

5.8 MLC based on a two stage BICM

The iterative decoder described previously allows us to approach the performance of the underlying constellation with multistage decoding. However, iterative decoding is not a direct implementation of MSD specially when the number of levels is relatively high as for a 16-QAM with $n_t = 2$ transmit antennas. We can try to achieve better performance using a multilevel coding scheme with two Bit-interleaved codes (see figure 5.13). The vectors of coded bits at the output of the two encoders have the same length $N_c/2$. The code rates verify $R_1 + R_2 = R/m$ where R is the total rate in bits per channel use. Hence, R_1 and R_2 can be optimized in a way to achieve the best performance. The length of the information bits at the input of the first and the second encoder are respectively equal to $N_{b1} = R_1 \times (N_c/2)$ and $N_{b2} = R_2 \times (N_c/2)$. The coded bits at the output of the first interleaver are used to select the "Good" bits a_j of a 16-QAM symbol. The coded bits at the output of the second interleaver select the "Bad" bits b_j.

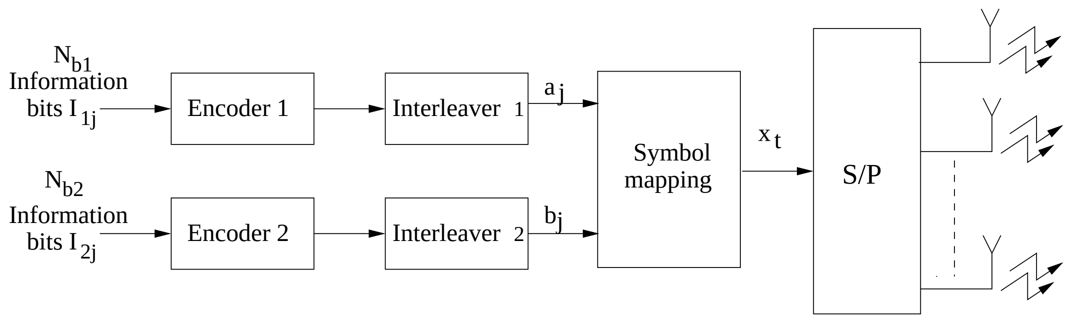


Figure 5.13: Two stage BICM transmitter.

MSD consists in decoding the first level of "Good" bits then the second level of "Bad" bits knowing the first level. The capacity of this scheme is equal to

$$C_{2BICM} = 4(C_a + C_{b|a}) \tag{5.49}$$

where $C_{b|a} = \max_P(X_{b1})I(Y; X_{b1}|X_a)$ is the capacity of one of the "Bad" equivalent channels knowing the first level *i.e.* knowing the 4 "Good" equivalent channels.

From figure 5.14, we get that the degradation in signal-to-noise ratio over the MIMO system with 16-QAM input is equal to 0.5 dB for a total code rate $R = 4$ bits per channel use. This corresponds to gain of 0.9 dB over the system with only

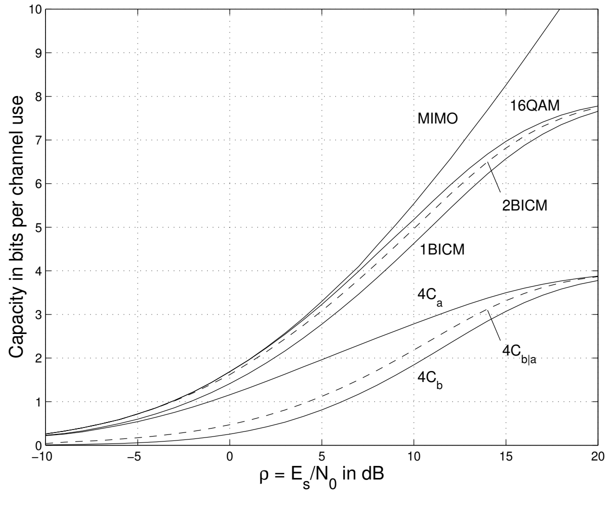


Figure 5.14: Capacity of a MIMO channel with a two stage BICM. 16-QAM constellation. $n_t = n_r = 2$.

one BICM. Moreover, iterative decoding at each level will significantly reduce the 0.5 dB degradation.

However, MSD decoding of the two levels with separate iterative decoding at each level is not recommended. This is due to the interaction between the iterative decoders and the signal constellation. The proposed structure of the decoder is represented in figure 5.15. The whole iterative receiving scheme is performed as follows :

- Initialization : Precompute the $\frac{N_c}{m.n_t} \times n_r$ likelihoods from the channel output. Set the N_c a priori probabilities $\pi(a_j)$ and $\pi(b_j)$ at the input of the "Marginalize" block to 1/2.

- At each iteration :

  1. Compute the $N_c/2$ observations $p(\mathbf{y}|a_j)$ from the channel output using equation (5.47). The SISO decoding at the first level calculate the $N_c/2$ a posteriori probabilities $APP(a_j) \propto Ext(a_j) \times obs(a_j)$ over the coded bits and the N_{b1} a posteriori probabilities $APP(I_{1j})$ over the information bits. The extrinsic information at the output of the SISO is used as a new a priori information.

  2. Compute the $N_c/2$ observations $p(\mathbf{y}|b_j)$ from the channel output using the newly updated a priori information. Then SISO decoding is done as for the first level.

- At the last iteration : decision is made over the information bits using $APP(I_{1j})$ and $APP(I_{2j})$.

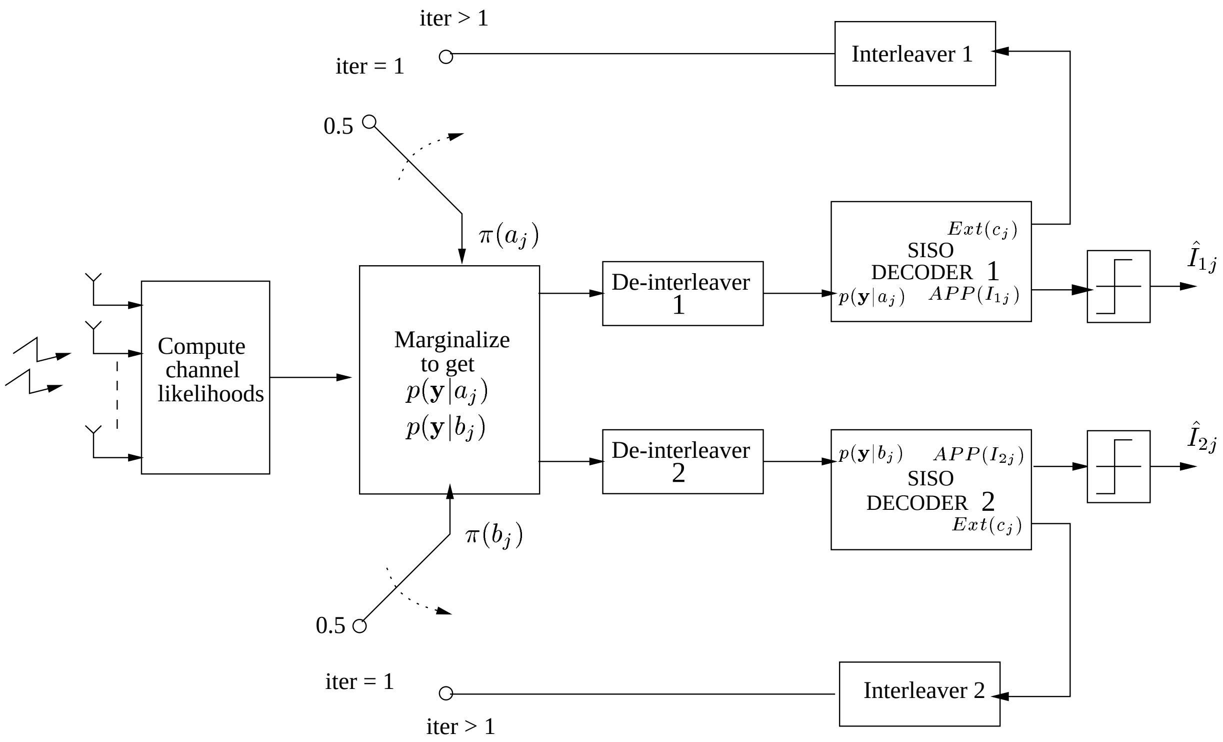


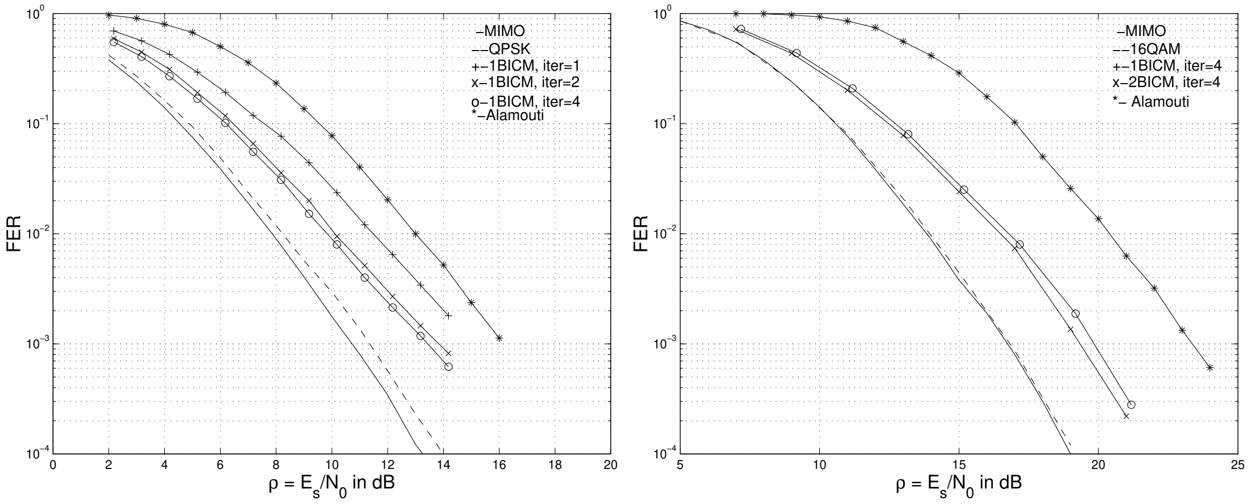
Figure 5.15: Two stage BICM iterative decoder.

The performance of the system over the *non-ergodic* block fading channel is plotted in figure 5.16.b. The gain achieved by this system over the system with one BICM is less than 0.5 dB. We didn't get better performance probably because of the error propagation between the two levels of the MLC scheme. The price to pay in order to achieve this small gain is in complexity. Therefore, the two stage BICM is not of great interest for MIMO channels with only $n_t = 2$ transmit antennas.

5.9 Conclusion

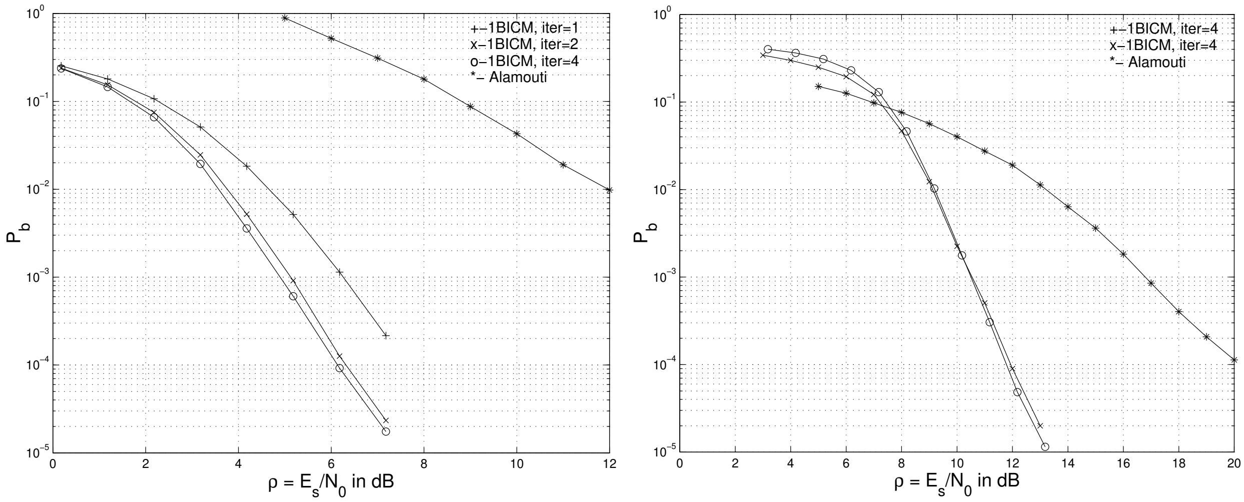
In this chapter, we briefly studied the transmission over a multiple-antenna channel. the system was examined using multilevel coding. We specially considered the case of a non-ergodic block fading channel with $n_t = 2$ transmit and $n_r = 2$ receive antennas. Performance close to the limit of a MIMO channel with discrete input is reached using a bit-interleaved coded modulation. For a QPSK constellation with a total rate $R = 2$ bits per channel use and a frame length $N_s = 100$ symbols, the gain in signal-to-noise ratio of a BICM over the Alamouti scheme is equal to 3 dB at a frame error rate equal to 10^{-3}. In the case of a 16-QAM constellation with $R = 4$ bits per channel use, the gain is equal to 3.5 dB. The price to pay in order to achieve these performance is in complexity.

Finally, a more complex coding scheme based on a two stage BICM was applied to a 16-QAM constellation in order to improve the performance of the one based on one BICM. However, the gain in signal-to-noise ratio was less than 0.5 dB. This was probably due to error propagation between the two levels of the system.



a. QPSK. b. 16-QAM.

Figure 5.16: Frame error rate of a BICM over a non-ergodic MIMO channel. $n_t = n_r = 2$.



a. QPSK. b. 16-QAM.

Figure 5.17: Bit error rate of a BICM over a MIMO ergodic Rayleigh fading channel. $n_t = n_r = 2$.

Conclusions and perspectives

The problem of capacity-approaching was largely studied in this thesis especially for the AWGN channel. The major work was divided into four parts :

First, the limits in terms of channel capacity and word error probability were reviewed. The word error probability of finite length optimal spherical codes, over the continuous AWGN channel, was calculated using the upper and the lower bound derived by Shannon using respectively a geometrical approach and general random coding techniques (For a code length N greater than 100, the two bounds are almost equal). These are the best performance that can be achieved by a finite length spherical code. For the BSC channel, the upper bound derived by Gallager seems to be a good approximation of the performance of optimal codes. These results determine the best achievable performance. Digital communications systems with performance close to these limits must be designed.

As shown in chapter 1, this can be done using a structured codes like lattice codes. In fact, it has been demonstrated that lattice codes can achieve, not only capacity, but also the performance of optimal spherical codes. Therefore, lattice codes are optimal for finite and infinite code length. However, methods for decoding lattice codes with a large code length are not practical.

Secondly, the performance of finite length turbo codes and irregular turbo codes were improved using symbol-based iterative decoding. The gain in performance is due to the reduction of the short cycles in the graph of the code.

The error floor that appears in the performance of the rate-1/2 irregular turbo code proposed by Frey [34] was lowered by replacing the punctured rate-1/2 RSC constituent code, by a rate-2/3 RSC code and by applying symbol-based iterative decoding. For a bit error probability equal 10^{-5}, the performance was better than the best known LDPC codes even for very large values of N. In any case, for small values of N, a regular turbo code can achieve better performance than LDPC.

For infinite code length, capacity of the AWGN channel with a BPSK input was reached as for LDPC codes. This was done using irregular turbo codes. Code rates equal to 1/3 and 1/2 were considered.

In summary, the advantages of turbo codes over LDPC codes are not only in the simplicity of the encoder and the performance for small code length, but also in the performance for large code length.

Further investigations need to be carried out. For example, the study of iterative decoding on a graph that contains cycles must be developped.

On other hand, although capacity was achieved, however for finite code length, the performance in terms of bit error probability are relatively degraded if compared to the bit error probability of an optimal spherical code. The degradation is more important when the word error probabilities are compared. This is due to the poor minimum Hamming distance achieved by turbo codes and compound codes in general. Instead of applying symbol-based iterative decoding to a binary code, an improvement could be reached by considering *non-binary* codes. This would be interesting when a small word error probability or a small bit error probability is needed (for example $P_e \simeq 10^{-5}$ or $P_b \simeq 10^{-8}$).

Thirdly, in the case of M-ary coded-modulations, it is shown that multilevel coding, combined to multistage decoding, can reach the capacity of an ASK, a QAM or a PSK modulation transmitted over an additive white Gaussian noise channel using block partitioned asymmetric constellations with equal signaling rates at each level. Performance were derived analytically using Gallager's coding exponents. Monte Carlo simulations showed that performance close to the Shannon limit are reached.

Therefore, asymmetric constellations are well suited for multilevel coding. The complexity of the MLC system, for example a multilevel UEP system, is largely reduced since only one encoder and one decoder must be implemented. For moderate code length N, the price to pay in order to reduce complexity is a relatively small degradation in performance due to the use of block partitioning instead of Ungerboeck's partitioning.

For finite code length, the use of irregular turbo codes and symbol-based iterative decoding would probably improve the performance of MLC. Non-binary codes must also be considered.

For infinite code length, the performance limits of an MLC scheme must be determined in order to find if capacity can be reached using MSD with iterative decoding at each level.

Note that, other turbo coded-modulations techniques like the pragmatic approach [38], the bit-interleaved coded modulations [12], as described in chapter 5 for MIMO channels , or Robertson's trellis coded modulations [69] could be more efficient than MLC.

Finally, the work done in chapter 5, concerning MIMO channels, is just an introduction to this field. For a non-ergodic block fading channel with $n_t = 2$ transmit and $n_r = 2$ receive antennas, performance close to the limit of a MIMO channel with discrete input is reached using a bit-interleaved coded modulation. The gain over the Alamouti scheme that has the same code rate and the same frame length is relatively important.

For a 16-QAM constellation, a small additional gain is obtained When the BICM is replaced by a more complex 2-level MLC scheme with a bit-interleaved code at each level. The price to pay in order to achieve this small gain is the increase of the complexity of the system. Therefore, the use of this scheme cannot be justified, although this not necessarily the case for more than 2 transmit antennas. This idea has to be discussed in the future.

Appendix A

Joint convergence of total and partial APP

We first prove the following result :

Result A.1
Let $x,\ y_m,\ z_m$ be 3 real random variables such that

$$z_m = x + y_m \ ,$$

where $m \in \mathbb{N}$. We consider that the random variable x verify the following two hypotheses :

- $\forall\ \epsilon > 0,\ \exists\ A \in \mathbb{R},\ \ Pr(x \geq A) < \epsilon$
- $\forall\ \epsilon > 0,\ \exists\ A \in \mathbb{R},\ \ Pr(x \leq A) < \epsilon$

Then, the two following statements are equivalent :

1. $\forall\ \alpha \in \mathbb{R},\ \forall\ \epsilon > 0,\ \exists\ m_0,\ \forall m > m_0\ \ Pr(y_m \geq \alpha) < \epsilon$
2. $\forall\ \beta \in \mathbb{R},\ \forall\ \epsilon > 0,\ \exists\ m_0,\ \forall m > m_0\ \ Pr(z_m \geq \beta) < \epsilon$

Proof :

1. Consider that statement (1) is verified :

$$\forall\ \alpha \in \mathbb{R},\ \forall\ \epsilon > 0,\ \exists\ m_0,\ \forall m > m_0\ \mathrm{Pr}(y_m \geq \alpha) < \epsilon \ .$$

Let $\epsilon' = f(\epsilon)$. From the two hypotheses defined for the variable x, we get

$$\exists\ A_{\epsilon'} \in \mathbb{R},\ \ \mathrm{Pr}(x \geq A_{\epsilon'}) < \epsilon' \ .$$

Let β and ϵ be two strictly positive real variables. We have

$$\mathrm{Pr}(z_m \geq \beta) = \mathrm{Pr}(z_m \geq \beta | x \leq A_{\epsilon'})\mathrm{Pr}(x \leq A_{\epsilon'}) + \mathrm{Pr}(z_m \geq \beta | x \geq A_{\epsilon'})\mathrm{Pr}(x \geq A_{\epsilon'}) \ .$$

This probability can be upper bounded by

$$\Pr(z_m \geq \beta) \leq \Pr(z_m \geq \beta | x \leq A_{\epsilon'}) + \epsilon'$$

(The other probabilities were upper bounded by 1). On the other hand,

$$x \leq A_{\epsilon'} \Leftrightarrow \beta - x \geq \beta - A_{\epsilon'}$$

hence

$$\Pr(z_m \geq \beta) \leq \Pr(y_m \geq \beta - A_{\epsilon'}) + \epsilon' \ .$$

Now defining $\epsilon' = \frac{\epsilon}{2}$ and $\alpha = \beta - A_{\epsilon'}$, we get the statement (2).

2. Consider that statement (2) is verified :

$$\forall \ \beta \in \mathbb{R}, \ \forall \ \epsilon > 0, \ \exists \ m_0, \ \forall m > m_0 \ \Pr(z_m \geq \beta) < \epsilon \ .$$

Let $\epsilon' = f(\epsilon)$. From the two hypotheses defined for the variable x, we get

$$\exists \ A_{\epsilon'} \in \mathbb{R}, \ \Pr(x \leq A_{\epsilon'}) < \epsilon' \ .$$

Let α and ϵ be two strictly positive real variables. We have

$$\Pr(y_m \geq \alpha) = \Pr(y_m \geq \alpha | x \leq A_{\epsilon'})\Pr(x \leq A_{\epsilon'}) + \Pr(y_m \geq \alpha | x \geq A_{\epsilon'})\Pr(x \geq A_{\epsilon'}) \ .$$

This probability can be upper bounded by

$$\Pr(y_m \geq \alpha) \leq \epsilon' + \Pr(y_m \geq \alpha | x \geq A_{\epsilon'})$$

(The other probabilities were upper bounded by 1). On the other hand,

$$x \geq A_{\epsilon'} \Leftrightarrow \alpha + x \geq \alpha + A_{\epsilon'}$$

hence

$$\Pr(y_m \geq \alpha) \leq \Pr(y_m \geq \alpha + A_{\epsilon'}) + \epsilon' \ .$$

Now defining $\epsilon' = \frac{\epsilon}{2}$ and $\beta = \alpha + A_{\epsilon'}$, we get the statement (1).

QED.

Result A.2 *Joint convergence of total and partial APP*
The sequence $\mathbf{x} = (-1, \ldots, -1)$ *is transmitted over an AWGN channel. Consider the two following statements :*

1. $\forall \epsilon > 0, \exists \ m_0$ *such that* $\forall m > m_0$, $Pr(\mathcal{B}_m > 0) < \epsilon$

2. $\forall \epsilon > 0, \exists \ m_0$ *such that* $\forall m > m_0$, $Pr(LR_m > 0) < \epsilon$

Then, statement (1) *implies statement* (2).

Proof :
Consider the case where
$$LR_m = \mathcal{B}_m + extLR_m \ ,$$
and
$$\mathcal{B}_m = obsLR + extLR_{m-1} \ .$$
The case where $\mathcal{B}_m$ is obtained from $d_b - 1$ independent and identically distributed extrinsic informations can be directly deduced from the following proof. Hence, we consider
$$z_m = LR_m \qquad y_m = extLR_m + extLR_{m-1} \qquad x = obsLR \ .$$
The variable x verifies the two hypotheses of Result A.1 since x is a Gaussian random variable. Therefore :
$$\forall\ \epsilon' > 0,\ \exists\ A \in \mathbb{R},\ \Pr(x \geq A) < \epsilon'$$
and
$$\forall\ \epsilon' > 0,\ \exists\ A \in \mathbb{R},\ \Pr(x \leq A) < \epsilon' \ .$$
Consider that statement (1) is verified. From Result A.1 applied first on variables $(\mathcal{B}_m, obsLR, extLR_{m-1})$, we have :
$$\forall\ \alpha \in \mathbb{R},\ \forall\ \epsilon' > 0,\ \exists\ m_0',\ \forall m > m_0'\ \Pr(extLR_{m-1} \geq \alpha) < \epsilon' \ . \tag{A.1}$$
For a fixed ϵ, and $m_0'' = \max(m_0', m_0)$, we get $\forall\ m > m_0''$:
$$\begin{aligned}\Pr(y_m > 0) &= \Pr(extLR_m + extLR_{m-1} > 0 | extLR_{m-1} \leq \alpha)\Pr(extLR_{m-1} \leq \alpha) \\ &+ \Pr(extLR_m + extLR_{m-1} > 0 | extLR_{m-1} \geq \alpha)\Pr(extLR_{m-1} \geq \alpha) \ .\end{aligned}$$
This probability can be upper bounded by
$$\Pr(y_m > 0) < \Pr(extLR_m + extLR_{m-1} > 0 | extLR_{m-1} \leq \alpha) + \epsilon' \ ,$$
thus,
$$\Pr(y_m > 0) < \Pr(extLR_m > -\alpha) + \epsilon' \ .$$
From equation (A.1), $\exists\ m_0^{'''}$, $\forall\ m > m_0^{'''}$:
$$\Pr(extLR_m > -\alpha) < \epsilon',$$
now defining $\epsilon' = \frac{\epsilon}{2}$, we have
$$\Pr(y_m > 0) < \epsilon \ .$$
Applying Result A.1 to variables (z_m, x, y_m), we get the statement (2).
QED.

Finally, the Gaussian approximation applied to the extrinsic informations allows us to prove that statements (1) and (2) are equivalent.

Bibliography

[1] S. M. Alamouti : "A simple transmit diversity technique for wireless communications," *IEEE Journal on Select Areas in Communications*, vol. 16, pp. 1451-1458, October 1998.

[2] L. R. Bahl, J. Cocke, F. Jelinek, J. Raviv : "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Transactions on Information Theory*, vol. 20, pp. 284-287, March 1974.

[3] S. Benedetto, G. Montorsi : "Unveiling turbo codes : Some results on parallel concatenated coding schemes," *IEEE Transactions on Information Theory*, vol. 42, pp. 409-429, March 1996.

[4] S. Benedetto, G. Montorsi : "Design of parallel concatenated convolutional codes," *IEEE Transactions on Communications*, vol. 44, pp. 591-600, May 1996.

[5] S. Benedetto, G. Montorsi, D. Divsalar, F. Pollara : "Serial concatenation of interleaved codes : Performance analysis, design and iterative decoding," *TDA Progress Report*, 42-126, JPL, August 1995.

[6] C. Berrou, A. Glavieux, P. Thitimajshima : "Near Shannon limit error-correcting : Turbo-codes," *Proceedings of the IEEE International Conference on Communications*, ICC, Geneva, Switzerland, pp. 1064-1070, May 1993.

[7] C. Berrou, M. Jezequel : "Non binary convolutional codes for turbo coding," *IEEE Electronics Letters*, vol. 35, pp. 39-40, January 1999.

[8] E. Biglieri : Introduction to treillis-coded modulations with applications. Macmillan, 1991.

[9] E. Biglieri : "An information-theoretical approach to coding for multiple antennas," Draft copy.

[10] M. Bingeman, A. K. Khandani : "Symbol-based turbo codes," *IEEE Communications Letters*, vol. 3, pp. 285-287, October 1999.

[11] J. J. Boutros : Lattice codes for Rayleigh fading channels. PhD thesis, ENST - Paris, France, May 1996.

[12] J. J. Boutros, F. Boixadera, C. Lamy : "Bit-interleaved coded modulations for multiple-input multiple-output channels," *Proceedings of the 6th IEEE International Symposium on Spread-Spectrum Technics and Applications*, NJIT, New Jersey, USA, September 2000.

[13] G. Caire, G. Taricco, E. Biglieri : "Capacity of bit-interleaved channel," *IEEE Electronics Letters*, vol. 32, pp. 1060-1061, June 1996.

[14] G. Caire, G. Taricco, E. Biglieri : "Bit-interleaved coded modulation," *IEEE Transactions on Information Theory*, vol. 44, pp. 927-946, May 1998.

[15] S. Chung, G. D. Forney, T. Richardson, R. Urbanke : "On the design of low-density parity check codes within 0.0045 dB of the Shannon limit," *IEEE Communications Letters*, vol. 5, pp. 58-60, February 2001.

[16] S. Chung, T. Richardson, R. Urbanke : "Analysis of the sum-product decoding of low-density parity-check codes using a Gaussian approximation," *IEEE Transactions on Information Theory*, vol. 47, pp. 657-670, February 2001.

[17] J. H. Conway, N. J. A. Sloane : Sphere packings, lattices and groups. Springer-Verlag, New York, 2nd edition, 1993.

[18] T. M. Cover, J. A. Thomas : Elements of information theory. Wiley, 1991.

[19] R. De Buda : "The upper error bound of a new near-optimal code," *IEEE Transactions on Information Theory*, vol. 21, pp. 441-445, July 1975.

[20] R. De Buda : "Some optimal codes have structure," *IEEE Journal on Selected Areas in Communications*, vol. 6, pp. 893-899, August 1989.

[21] D. Divsalar, S. Dolinar, R. J. McEliece, F. Pollara : "Transfer function bounds on the performance of turbo codes," *TDA Progress Report*, 42-122, JPL, August 1995.

[22] S. Dolinar, D. Divsalar, F. Pollara : "Code performance as function of block size," *TMO Progress Report*, 42-133, May 1998.

[23] P. Elias : "Error-free coding," *IRE Transactions on Information Theory*, pp. 29-37, 1954.

[24] H. El Gamal, A. R. Hommos : "Analysing the turbo decoder using the Gaussian approximation," *IEEE Transactions on Information Theory*, vol. 47, pp. 671-686, February 2001.

[25] A. J. Felstrom, K. Sh. Zigangirov : "Time-varying periodic convolutional codes with low-density parity-check matrix," *IEEE Transactions on Information Theory*, vol. 45, pp. 2181-2191, September 1999.

[26] R. Fischer, J. Huber, U. Washmann : "Multilevel coding : Aspects from information theory," *Proceedings of the IEEE Information Theory Workshop*, ITW, Haifa, Israel, pp. 26-30, June 1996.

[27] R. Fischer, U. Washmann, J. Huber : "On the combination of multilevel coding and signal shaping," *Proceedings of ITG-Fachbericht : Codierung*, Aachen, Germany, pp. 273-278, March 1998.

[28] G. D. Forney Jr., L.-F. Wei : "Multidimensional constellations : Introduction, figures of merit, and generalized cross-constellations," *IEEE Journal on Selected Areas in Communications*, vol. 7, pp. 877-892, 1989.

[29] G. D. Forney : "Coset codes — PartII : Binary lattices and related codes," *IEEE Transactions on Information Theory*, vol.34, pp. 1152-1187, September 1988.

[30] G. D. Forney : Concatenated codes. MIT Press, 1966.

[31] G. J. Foschini, M. J. Gans : "On limits of wireless communications in a fading environment when using multiple antennas," *Wireless Personal Communications*, March 1998.

[32] B. Frey, D. MacKay : "Irregular turbo codes," *Proceedings of the 37th Allerton Conference*, Illinois, USA, September 1999.

[33] B. J. Frey, F. R. Kschischang : "Probability propagation and iterative decoding," *Proceedings of the 34th Allerton Conference*, Illinois, USA, October 1996.

[34] B. Frey, D. MacKay : "Irregular turbo-like codes," *Proceedings of the International Symposium on Turbo Codes*, Brest, France, September 2000.

[35] R. G. Gallager : Low-density parity-check codes. MIT Press, 1963.

[36] R. Gallager : "A simple derivation of the coding theorem and some applications," *IEEE Transactions on Information Theory*, vol. 11, pp. 3-18, January 1965.

[37] R. Gallager : Information Theory and Reliable Communication. J. Wiley, 1968.

[38] S. Goff, A. Glavieux, C. Berrou : "Turbo codes and high-spectral efficiency modulation," *Proceedings of the IEEE International Conference on Communications*, ICC, New Orleans, Louisiana, USA, pp. 645-9, May 1994.

[39] G. D. Golden, G. J. Foschini, P. W. Wolniansky, R. A. Valenzuela : "V-BLAST : A high capacity space-time architecture for the rich-scattering wireless channel," *Proceedings of the International Symposium on Advandced Radio Technologies*, Boulder, Colorad, USA, September 1998.

[40] G. D. Golden, G. J. Foschini, R. A. Valenzuela, P. W. Wolniansky : "V-BLAST : An architecture for realizing very high data rates over the rich-scattering wireless channel," *Proceeding of IEEE International Symposium on Signals, Systems, and Electronics*, ISSSE, Pise, Italy, September 1998.

[41] G. D. Golden, G. J. Foschini, R. A. Valenzuela, P. W. Wolniansky : "Detection algorithm and initial laboratory results using V-BLAST space-time communication architecture," *IEEE Electronics Letters*, vol. 35, pp. 14-16, January 1999.

[42] J. Hagenauer, P. Hoeher : "A Viterbi algorithm with soft-decision outputs and its applications," *Proceedings of the IEEE Global Telecommunication Conference*, GLOBECOM, Dallas, Texas, pp. 47.1.1-47.1.7, November 1989.

[43] B. Hassibi, B. M. Hochwald : "High-rate codes that are linear in space and time," Bell Laboratories, April 2001.

[44] B. M. Hochwald, T. L. Marzetta : "Unitary Space-time modulation for multiple-antenna communications in Rayleigh flat fading," *IEEE Transactions on Information Theory*, vol. 46, pp. 543-564, March 2000.

[45] J. Huber : "Multilevel codes : Distance profiles and channel capacity," *Proceedingd of ITG-Fachbericht*, Munchen, Germany, pp. 305-319, October 1994.

[46] H. Imai, S. Hirakawa : "A new multilevel coding method using error correcting codes," *IEEE Transactions on Information Theory*, vol. 23, pp. 371-377, May 1977.

[47] F. R. Kschischang, B. J. Frey : "Iterative decoding of compound codes by probability propagation in graphical models," *IEEE Journal on Selected Areas in Communications*, vol. 16, pp. 219-230, February 1998.

[48] F. R. Kschischang, B. J. Frey, H.-A. Loeliger : "Factor graphs and the sum-product algorithm," *IEEE Transactions on Information Theory*, vol. 47, pp. 498-516, February 2001.

[49] C. Lamy, J. Boutros : "Décodage à sortie souple des réseaux de points," Motorola Semiconductors, Toulouse, France, 1998.

[50] C. Lamy: Communications à grande efficacité spectrale sur le canal à évanouissements. PhD thesis, ENST - Paris, France, April 2000.

[51] D. E. Lazic, M. Calic : "How close are turbo codes to optimal codes," *Proceedings of the International Symposium on Turbo Codes*, Brest, France, 1997.

[52] X. Li, A. Ritcey : "Trellis-coded modulation with bit-interleaving and iterative decoding," *IEEE Journal on Selected Areas in Communications*, vol. 17, pp. 715-724, April 1999.

[53] T. Linder, C. Schlegel, K. Zeger : "Corrected proof of Buda's theorem," *IEEE Transactions on Information Theory*, vol. 39, pp. 1735-1737, September 1993.

[54] H.-A Loeliger : "Averaging bounds for lattices and linear codes," *IEEE Transactions on Information Theory*, vol. 43, pp. 1767-1773, November 1997.

[55] M. Luby, M. Mitzenmacher, M. A. Shokrollahi, D. A. Spielman : "Improved low-density parity-check codes using irregular graphs," *IEEE Transactions on Information Theory*, vol. 47, pp. 569-584, February 2001.

[56] D. J. C. MacKay : "Good error-correcting codes based on very sparse matrices", *IEEE Transactions on Information Theory*, vol. 45, pp. 399-431, March 1999.

[57] D. J. C. MacKay, S. T. Wilson, M. C. Davey : "Comparison of constructions of irregular Gallager codes," *IEEE Transactions on Communications*, vol. 47, pp. 1449-1454, October 1999.

[58] F. J. MacWilliams, N. J. A. Sloane : The theory of error correcting codes. North-Holland, Amsterdam, 1977.

[59] H. Magalhaẽs de Oliveira, G. Battail : "A capacity theorem for lattice codes on Gaussian channels," *Proceedings of SBT/IEEE International Telecommunications Symposium*, Rio de Janeiro, Brazil, September 1990.

[60] H. Magalhaẽs de Oliveira, G. Battail : "Performance of lattice codes over the Gaussian channel," *Annales des Télécommunications*, vol. 47, pp. 293-305, 1992.

[61] T. L. Marzetta, B. M. Hochwald : "Capacity of a mobile multiple-antenna communication link in Rayleigh flat fading," *IEEE Transactions on Information Theory*, vol. 45, pp. 139-157, January 1999.

[62] G. Poltyrev : "On coding without restrictions for the AWGN channel," *IEEE Transactions on Information Theory*, vol. 40, pp. 409-417, March 1994.

[63] J. G. Proakis : Digital communications. McGraw-Hill, New York, 3rd edition, 1995.

[64] T. Richardson, R. Urbanke : "The capacity of low-density parity check codes under message-passing decoding," Bell Laboratories, November 1998.

[65] T. Richardson, M. A. Shokrollahi, R. Urbanke : "Design of capacity-approaching low-density parity check codes," *IEEE Transactions on Information Theory*, vol. 47, pp. 619-637, February 2001.

[66] T. Richardson, R. Urbanke : "Efficient encoding of low-density parity check codes," *IEEE Transactions on Information Theory*, vol. 47, pp. 638-656, February 2001.

[67] P. Robertson, P. Hoeher, E. Villeburn : "A comparison of optimal and sub-optimal MAP decoding algorithms operating in the log-domain," *Proceedings of IEEE International Conference on Communications*, ICC, Seattle, Washington, USA, pp. 1009-1013, February 2001.

[68] P. Robertson, E. Villeburn, P. Hoeher : "Optimal and sub-optimal a posteriori algorithms suitable for turbo decoding," *European Transactions on Telecommunications*, vol. 8, pp. 119-125, March - April 1997.

[69] P. Robertson, T. Wörz : "Bandwidth-efficient turbo trellis-coded modulation using punctured component codes," *IEEE Journal on Selected Areas in Communications*, vol. 16, pp. 206-218, February 1998.

[70] C. A. Rogers : Packing and covering. Cambridge University press, United Kingdom, 1964.

[71] H. E. Sawaya, J. J. Boutros : "Error probability of optimal lattice codes," *Proceedings of the 22nd Symposium on Information Theory and its Applications*, SITA, vol. 2, pp. 587-590, December 1999

[72] H. E. Sawaya, J. J. Boutros : "New concepts for the design of multilevel codes," *Proceedings of the International Conference on Reaserch Trends in Science and Technology*, RTST, Beirut, Lebanon, March 2000.

[73] H. E. Sawaya, J. J. Boutros : "Multilevel coded modulations based on asymmetric constellations," *Proceedings of the IEEE International Symposium on Information Theory*, ISIT, Washington, USA, June 2001.

[74] H. E. Sawaya, S. Vialle, J. J. Boutros, G. Zemor : "Performance limits of compound codes with symbol-based iterative decoding," *Proceeding of the Work Shop on Coding and Cryptography*, WCC, Paris, France, January 2001.

[75] H. E. Sawaya, J. J. Boutros : "Performance of irregular turbo codes with symbol-based iterative decoding," *Submitted to the International Symposium on Information Theory*, ISIT, Lauzanne, Switzeland, June 2002.

[76] H. E. Sawaya, J. J. Boutros : "xxxx," *Submitted to IEEE Electronics Letters.*

[77] S. I. Sayegh : "A class of optimum block codes in signal space," *IEEE Transactions on Communications*, vol. 34, pp. 1043-1045, October 1986.

[78] C. E. Shannon : "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, pp. 379-623, 1948.

[79] C. E. Shannon : "Probability of error for optimal code in Gaussian channel," *Bell System Technical Journal*, vol. 38, May 1959.

[80] D. Slepian : "Bounds on communication," *Bell System Technical Journal*, vol. 42, pp. 681-707, 1963.

[81] R. M. Tanner : "A recursive approach to low complexity codes," *IEEE Transactions on Information Theory*, vol. 27, pp. 533-547, September 1981.

[82] V. Tarokh, A. Vardy, K. Zeger : "Universal Bound on the performance of Lattice Codes," *IEEE Transactions on Information Theory*, vol. 45, pp. 670-681, March 1999.

[83] V. Tarokh, N. Seshadri, A. R. Calderbank : "Space-time codes for high data rate wireless communication : Performance criterion and code construction," *IEEE Transactions on Information Theory*, vol. 44, pp. 744-765, March 1998.

[84] V. Tarokh, H. Jafarkhani, A. R. Calderbank : "Space-time block codes from orthogonal designs," *IEEE Transactions on Information Theory*, vol. 45, no. 5, July 1999.

[85] E. Telatar : "Capacity of multi-antenna Gaussian channels," *AT&T-Bell Laboratories Internal Technical Memorandum*, June 1995.

[86] S. Ten Brink : "Iterative decoding trajectories of parallel conacatenated codes," *Proceeding of the 3rd IEEE ITG Conference on Source and Channel Coding*, Munich, Germany, January 2000.

[87] S. Ten Brink : "Design of serially concatenated codes based on iterative decoding convergence," *Proceedings of the International Symposium on Turbo Codes*, Brest, France, September 2000.

[88] S. Ten Brink : "Rate one-half code approaching the Shannon limit by 0.1 dB," *IEEE Electronics Letters*, vol. 36, pp. 1293-1294, July 2000.

[89] S. Ten Brink : "Code doping for triggering iterative decoding convergence," *Proceedings of the International Symposium on Information Theory*, ISIT, Washington, USA, June 2001.

[90] G. Ungerboeck : "Channel coding with multilevel/phase signals," *IEEE Transactions on Information Theory*, vol. 28, pp. 55-67, January 1982.

[91] G. Ungerboeck : "Trellis coded modulation with redundant signal sets, Part I," *IEEE Communications Magazine*, vol. 25, pp. 5-11, February 1987.

[92] G. Ungerboeck : "Trellis coded modulation with redundant signal sets, Part II ," *IEEE Communications Magazine*, vol. 25, pp. 12-21, February 1987.

[93] R. Urbanke, B. Rimoldi : "Lattice codes can achieve capacity on the AWGN channel," *IEEE Transactions on Information Theory*, vol. 44, pp. 273-278, January 1998.

[94] S. Vialle, J. J. Boutros : Travail sur la capacité sur canal Gaussien et canal de Rayleigh. ENST - Paris, France, February 1998.

[95] S. Vialle, J. J. Boutros : "Performance limits of concatenanted codes with iterative decoding," *Proceedings of the International Symposium on Information theory*, ISIT, Sorrento, Italy, June 2000.

[96] C. Lamy: Construction et analyse de nouvelles structures de codage adaptées au traitement itŕatif.. PhD thesis, ENST - Paris, France, December 2000.

[97] A. J. Viterbi, J. K. Omura : Principles of digital communications and coding. McGraw-Hill, New York, 1979.

[98] U. Washmann, R. Fischer, J. Huber : "Multilevel codes : Theoretical concepts and practical design rules," *IEEE Transactions on Information Theory*, vol. 45, pp. 1361-1391, July 1999.

[99] U. Washmann, J. Huber : "Power and bandwidth efficient digital communication using turbo codes in multilevel codes," *European Transactions on Telecommunications*, vol. 6, pp. 557-567, September - October 1995.

[100] T. Wörz, J. Hagenauer : "Iterative decoding for multilevel codes using reliability information," IN *Proceedings of the IEEE Global Telecommunication Conference*, GLOBECOM, Orlando, USA, December 1992.

[101] E. Zehavi : "8-PSK trellis codes for a Rayleigh channel," *IEEE Transactions on Communications*, vol. 40, pp. 873-884, May 1992.