

## Chapter 6

# Memoryless Channels and LDPC Codes

### 6.1 Introduction

As stated in Chapter 1, the introduction of low-density parity-check (LDPC) codes has allowed to achieve near-capacity transmission over some simple channels, such as, for example, the binary input additive white Gaussian noise (BI-AWGN) channel, the binary erasure channel (BEC), or the binary symmetric channel (BSC) [9, 12, 13, 33]. Although the performance of LDPC codes transmitted over binary-input memoryless channels is known and well studied in the literature, a formal proof of their potential to achieve channel capacity is still lacking. Moreover, the application of LDPC codes to generic memoryless channels, and, in particular, for medium to high spectral efficiency signaling over the AWGN channel, represents a promising evolution of established coding techniques, such as, for example, trellis coded modulations (TCM), and have been the subject of attention in the scientific community.

In this chapter, we address the performance of LDPC codes transmitted through a memoryless channel, as described in the following paragraphs.

In Section 6.2, we consider an extrinsic information transfer (EXIT) chart-based analysis of the convergence of the belief propagation decoding algorithm of LDPC codes [69, 76, 77] for binary input memoryless channels. An in-depth investigation of this analysis technique suggests that the performance of LDPC codes depends marginally on the characteristics of the particular memoryless channel, whereas it is dominated by the mutual information (MI) between the input and the output of the channel. Related work also appears in [100–102]. In the following, we will refer to this MI as *constrained input*

channel capacity  $C_{ci}$ , in order to distinguish it from the MI used in EXIT chart-based analyses, which refers to the MI between the generic message in the graph and the corresponding codeword bit and is used to track the convergence of the decoding process. Each investigated communication scheme is characterized in terms of the parameter  $C_{ci}$ . Moreover, a uniform input binary distribution is assumed, i.e., the *a priori* distribution of an information bit  $X$  is such that  $P\{X = 1\} = P\{X = 0\} = 1/2$ . We will also assume uniform distribution for the LDPC encoded bits, a property that in general holds for every codeword bit in a binary linear code whose generator matrix does not contain all-zero lines.

In Section 6.3, we will use the results obtained in the first part of this chapter to derive an efficient design algorithm for multilevel coding (MLC) [103–105]. The proposed algorithm exploits the partition into memoryless subchannels induced by MLC, and selects, from a given library of LDPC codes, a subset of codes. The selected subset of LDPC codes is tailored for MLC, and is optimal, in the sense that it maximizes the spectral efficiency guaranteeing a fixed bit error rate (BER) performance above a given SNR. This technique can therefore be used as a practical tool to achieve high spectral efficiency using LDPC codes.

## 6.2 Performance of LDPC Codes on Binary-Input Memoryless Channels

The assumptions considered in [9] for the derivation of the Gallager A, B, and C iterative decoding algorithms for LDPC codes are valid for a binary input *memoryless* channel. In [69–71], practical approximations of EXIT charts are proposed and used for LDPC code design. Tight upper and lower bounds for EXIT charts have been derived in [62–64, 106, 107]. These bounds allow to find transmission conditions, in terms of MI between the input and the output of the channel, for which it is possible to guarantee convergence regardless of the specific channel. Nevertheless, these bounds do not completely reflect the actual behavior of the decoding process of LDPC codes, which, as the number of iterations increases, seems to converge to that of the BEC bound regardless of the specific channel [93]. This behavior, which has been experimentally observed, seems related to the fact that in the last iterations the BER is low.

In the following, we show that a performance analysis of LDPC codes based on EXIT charts suggests that the behavior of an *ensemble* of LDPC codes (i.e., a set of codes with given degree distributions) does not depend appreciably on the particular memoryless channel, but only on the MI between the input

and the output of the channel. This would imply that a code from a given ensemble will exhibit *similar* convergence threshold and performance on any memoryless channel—in other words, different memoryless channels exhibit minor performance differences for a given value of MI. This observation is supported by simulation results relative to various LDPC codes and several memoryless channels. The considered channels are both *symmetric*—binary-input AWGN channel, BSC and BEC—and *asymmetric*—binary asymmetric channel (BAC) and Z channel (ZC). This confirms the early remark in [13], where it was observed that LDPC codes optimized for the AWGN channel show good performance for other memoryless channels, such as BSC and BEC. Similar conclusions were drawn in [100, 101], where the authors show that the performance of LDPC codes over any Gaussian channel, not necessarily AWGN, depends only on the MI between the input and the output of the channel. We remark that the EXIT chart-based analysis of LDPC codes, interpreted as functions of their degree distributions, underlies the implicit assumption that the graphs of the corresponding LDPC codes do not contain short cycles. This condition can be achieved, for example, by choosing a sufficiently long codeword length.

### 6.2.1 The Start Point in EXIT Charts

In Chapter 4, we introduced a statistical characterization of the convergence behaviour of LDPC coded schemes on the basis of EXIT charts. We now ask ourselves the following question: what is the meaning of the start point of the decoding trajectory in an EXIT chart? The messages at the output of variable nodes at the very first iteration correspond to the logarithmic likelihood ratios (LLRs), based on channel observations, of the transmitted symbols [9]. These quantities are *sufficient statistics* for an optimal decision on the transmitted sequence. This means that the MI between the transmitted binary sequence and these LLRs is equal to the MI between the transmitted binary sequence and the channel output. Since the transmitted bits are assumed to be 0 or 1 with probability 1/2, this MI can also be interpreted as constrained-input channel capacity  $C_{ci}$ . Hence, at the first iteration, the MI generated at the output of the variable node detector (VND) is  $I_V = C_{ci}$ . As stated in Section 5.4, this value corresponds to the start point of the EXIT chart decoding trajectory in Figure 5.4.

Since simple EXIT chart-based analyses assume that the MI at the output of a block is independent of the particular distribution of the messages, but depends only on the MI at its input, the check node detector (CND) EXIT curve (i.e., the curve  $I_B$  in Figure 5.4) does not depend on the particular channel.

Nevertheless, the VND EXIT curve may depend on the channel. Experience suggests, however, that the VND EXIT curve (i.e.,  $I_V$ ) depends on the channel through  $C_{ci}$  only, whereas it depends weakly on the particular channel type. This is taken into account in the practical approximations currently used for LDPC code design in [69], where the author expresses  $I_V$  as a function of the MI between the input and the output of the channel and the MI of the messages coming from the VND. Since, as previously stated,  $C_{ci} = I_V(0)$ , i.e., the start point of the EXIT chart decoding trajectory, from the considerations above, it follows that the entire function  $I_V(I)$  can be characterized by this start point.

Assuming that the EXIT chart-based analysis is accurate, one can conclude that the convergence of the decoding process for an ensemble of LDPC codes, described by their degree distributions, depends only on the constrained-input channel capacity and not on the particular channel. This means that *if* a randomly chosen code of a given ensemble shows, with high probability, a good BER performance when transmitted over a memoryless channel with given  $C_{ci}$ , *then* this code, with high probability, will guarantee good BER performance also when used for transmission over any other memoryless channel with equal  $C_{ci}$ . The previous consideration is valid *provided that* there is no feedback from the VND to the soft demapper or that the presence of this feedback would not change the MI between the message set at the output of the soft demapper and the generic codeword bit. This occurs in binary input memoryless channels and other significant scenarios which will be addressed in Chapter 7.

It is important to note that these considerations rely on an approximated method, and their accuracy is strictly related to the accuracy of the EXIT chart-based analysis. In the next section, simulation results will be presented that allow to understand to what extent LDPC codes belonging to the same ensemble and transmitted over a memoryless channel show similar performance *regardless* of the specific channel.

## 6.2.2 Numerical Evidence

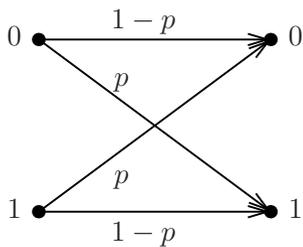
We consider Monte Carlo simulation-based performance analysis of three LDPC codes transmitted over five different memoryless channels. The considered codes have rates  $1/4$ ,  $1/2$ , and  $3/4$ , and are generated starting from the degree distributions, optimized for the binary-input AWGN channel, found in [97]. The used degree distributions (for variable and check nodes) are given in Table 6.1. The codeword length is set to 10000 binary symbols in all cases. The decoding process stops if a codeword is obtained or a maximum allowed

Table 6.1: Variable and check nodes degree distributions for the considered LDPC codes.

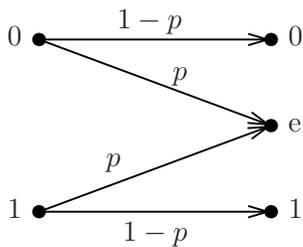
Code Rate	Variable Node Degree Distribution		Check Node Degree Distribution	
	$i$	$\lambda_i$	$j$	$\rho_j$
$\frac{1}{4}$	2	0.4161610	4	1
	3	0.2355160		
	5	0.0759725		
	6	0.061250		
	7	0.0013665		
	9	0.0158465		
	10	0.1938870		
$\frac{1}{2}$	2	0.272536	7	0.7
	3	0.237552		
	4	0.070380		
	10	0.419532		
$\frac{3}{4}$	2	0.201224	16	1
	3	0.276439		
	4	0.033386		
	10	0.488951		

number (equal to 100) of iterations is reached. In both cases, a decision on a binary symbol is made according to the final corresponding LLR value of the symbol, computed as the sum of all the messages sent to its corresponding variable node at the last iteration. The receiver is assumed to know the channel statistics. The soft demapper thus computes the exact APP.

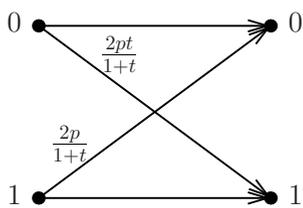
Figure 6.1 shows the considered memoryless channels, which comprise three symmetric channels (binary input AWGN channel, BSC and BEC) and two asymmetric channels (BAC and ZC). We evaluate the BER performance of each considered code over each channel as a function of the constrained-input capacity. For the BAC, the transition probability  $P\{0 \rightarrow 1\}$  is different from the transition probability  $P\{1 \rightarrow 0\}$ . Two parameters are then necessary to describe this channel (and to compute  $C_{ci}$ ). We choose to specify the ratio  $t \triangleq P\{0 \rightarrow 1\}/P\{1 \rightarrow 0\}$  as a given constant parameter. The ZC can be interpreted as a particular instance of the BAC with  $t = 0$ . It is then possible to express  $C_{ci}$  for every channel as a function of a single parameter, namely the SNR  $\gamma$  for the AWGN channel, the transition probability for the



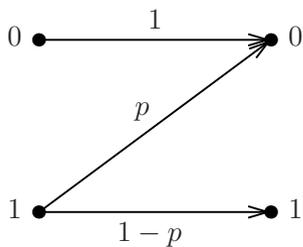
(a)



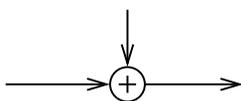
(b)



(c)



(d)



(e)

Figure 6.1: Pictorial representation of the considered memoryless channels: (a) the BSC, (b) the BEC, (c) the BAC, (d) the ZC and (e) the BI-AWGN.

BSC, the erasure probability for the BEC, the average transition probability ( $P\{1 \rightarrow 0\} + P\{0 \rightarrow 1\}$ )/2 for both BAC and ZC (all these probabilities are denoted by  $p$  in the following expressions). Summarizing, the constrained-input capacities of the considered channels can be obtained using standard methods and have the following expressions:

$$C_{\text{ci}}^{\text{AWGN}} = \frac{1}{2} \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi\gamma}} e^{-\frac{(y-2\gamma)^2}{8\gamma}} \log_2 \frac{2}{1+e^{-y}} dy \quad (6.1)$$

$$C_{\text{ci}}^{\text{BSC}} = 1 + p \log(p) + (1-p) \log(1-p) \quad (6.2)$$

$$C_{\text{ci}}^{\text{BEC}} = 1 - p \quad (6.3)$$

$$C_{\text{ci}|t}^{\text{BAC}} = \frac{1}{2(1+t)} \left\{ 2(1+t) + pt(1+t) \log \left[ \frac{pt(1+t)}{(1+t)(1-p+pt)} \right] \right. \\ \left. + (1+t)(1-p) \log \left[ \frac{(1+t)(1-p)}{(1+t)(1-p+pt)} \right] \right. \quad (6.4)$$

$$+ p(1+t) \log \left[ \frac{p(1+t)}{(1+t)(1-p+pt)} \right] \\ \left. + (1-pt)(1+t) \log \left[ \frac{(1+t)(1-pt)}{(1+t)(1-p+pt)} \right] \right\} \quad (6.5)$$

$$C_{\text{ci}}^{\text{ZC}} = C_{\text{ci}|t=0}^{\text{BAC}} = \frac{1}{2} [2 + p \log p - (1+p) \log(1+p)]. \quad (6.6)$$

We remark that for symmetric channels,  $C_{\text{ci}}$  equals the unconstrained capacity [1].

In Figure 6.2, the BER curves of all three codes, transmitted over the considered five channels, are shown as functions of  $C_{\text{ci}}$ —note that  $C_{\text{ci}}$  can assume values between 0 and 1, since the transmitted symbols are binary. For the BAC, the ratio  $t = 3$  is chosen as representative. From the results in Figure 6.2, one can conclude that the convergence threshold, in terms of  $C_{\text{ci}}$ , basically depends only on the code and, in a very limited way, on the channel. Interestingly, this conjecture holds for asymmetric channels as well. The slight differences between the BER curves relative to a specific code cannot be predicted by the EXIT chart-based analysis. In fact, the BER curves depend on the actual code structure, which may contain short cycles [13], and on the statistical distribution of the LLRs at the output of the channel, which are not taken into account by the EXIT charts. Moreover, by considering Figure 6.2 one can quantify the actual difference between the performance of a code transmitted over the considered channels in terms of small fractions of bits per channel use (within a few hundredths).

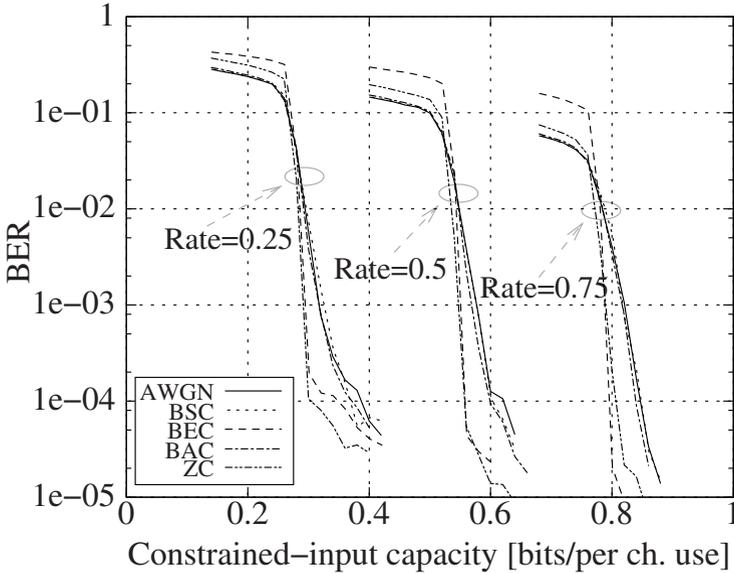


Figure 6.2: BER versus constrained-input capacity for three LDPC codes, characterized by rates  $1/4$ ,  $1/2$  and  $3/4$ , respectively. For each code, the performance for transmission over five memoryless channels is shown. The codeword length is 10000 and the codes are optimized for transmission on the AWGN channel.

### 6.2.3 Implications

In the previous subsections, we have shown that, with good approximation, LDPC codes which are good for a particular memoryless channel are also good for any memoryless channel, in the sense that they guarantee similar BER performance in the same  $C_{ci}$  region, regardless of the channel type.

This allows a characterization of the coding gain in terms of bits per channel use. Since slight differences between different channels have been observed, one can expect that a given LDPC code will exhibit equal performance, within a small fraction of bits per channel use, over different memoryless channels. More rigorous claims regarding our conjecture would involve the derivation of new bounds on EXIT curves of LDPC codes, which extend the results in [62–64, 106, 107], taking into account how the distribution of the messages varies at each iteration. The fact that the performance of an LDPC code has a small dependence on the particular channel has some interesting implications, described in the following.

### LDPC Codes Libraries

The independence of the LDPC code performance from the particular memoryless channel implies that good LDPC codes for memoryless channels could be collected in code libraries and reused for several different applications, thus separating the tasks of (i) designing LDPC codes and (ii) fitting them to the considered scenario, which may consist of the concatenation of one or more LDPC codes with a high spectrally efficient modulator .

### Soft Demapper without Feedback from VND

The presented conjecture may also impact the design of LDPC codes to be used in a bit interleaved coded modulation (BICM) scheme, which maps binary symbols onto high-order modulation formats [81]. At the receiver side, a soft demapper could generate reliability values for the mapped bits to be passed to the LDPC decoder, which would treat them as channel outputs. In this case, the decoding process would not depend on the particular mapper or channel but only on the MI at the output of the soft demapper. Assuming that iterations between demapper and decoder are not performed or are not useful, LDPC codes designed for a simple memoryless channel (e.g., BSC) will be a good choice also if mapped into high-order modulations. This is due to the fact that, in BICM schemes, the presence of the interleaver after the binary encoder transforms the channel, as seen by the encoder/decoder, into a memoryless channel.<sup>1</sup> However, if iterations are allowed, these claims are no longer valid, since at every iteration the LDPC VND operates on a different input from the soft demapper.

### LDPC-MLC Design

Another application of the described property is presented in the following section, where it is shown how to select LDPC codes belonging to a library of good codes for memoryless channels, in order to design an LDPC coded MLC scheme.

## 6.3 Multilevel Code Design

We now use the concepts developed in the previous sections in order to design MLC schemes [103–105]. In particular, given a set of LDPC codes which exhibit good performance on memoryless channels and featuring several code

---

<sup>1</sup>This holds exactly only if an ideal random interleaver is used.

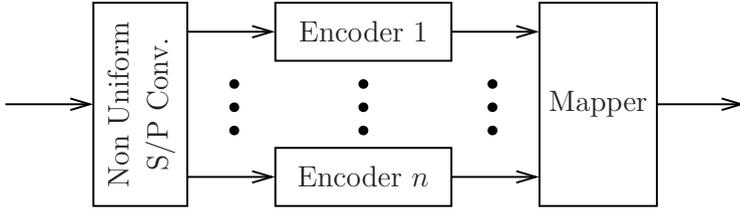


Figure 6.3: MLC transmission scheme.

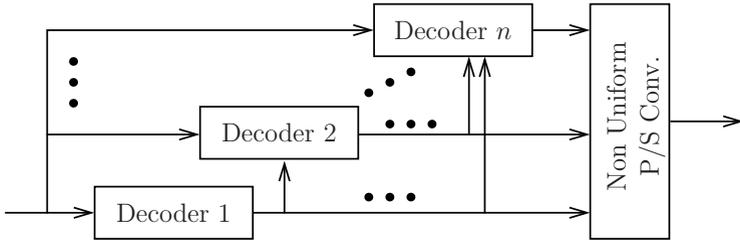


Figure 6.4: Multi-stage decoding scheme: each decoder passes its decisions to every following decoder, which uses this information to compute its own decisions.

rates, i.e., an LDPC code library, we show how to optimally choose the tuple of codes for a MLC scheme operating on a given constellation.

### 6.3.1 Multilevel Scheme Overview

In Figure 6.3, a MLC transmission scheme is shown for transmission over a  $2^n$ -point constellation. The information bits are split into  $n$  different streams by a serial to parallel (S/P) conversion block. Each information bit stream is encoded with a properly chosen LDPC code. The S/P conversion block adjusts the bit rate delivered to each encoder in order to obtain, at the output of each encoder the same binary symbol rate, regardless of the code rate. Hence, the label “Non Uniform S/P.” The obtained encoded bit streams feed a mapper which encodes  $n$  code bits into a constellation symbol. This coding technique was first proposed in [103], along with a proper decoding algorithm, referred to as multi-stage decoding (MSD). MLC caught growing attention after [105], where it is shown how to determine the rate of each component code in order to retain the full information rate allowed by the considered modulation.

In Figure 6.4, an MSD scheme is shown. For each encoder at the trans-

mitter side, there is a matched decoder at the receiver side. The received observables are fed to each decoder. The decoding process starts at decoder 1, which performs a codeword decision based on the observables from the channels. The decided bits are passed to the remaining decoders. At each stage, the  $i$ -th decoder makes a decision on the corresponding codeword, based on the observables from the channel and the codeword decisions corresponding to codes  $1, \dots, i - 1$ .

Following an information-theoretic viewpoint proposed in [105], we denote the  $n$  bits forming a constellation symbol as  $X_1, \dots, X_n$  and the corresponding noisy received observable as  $Y$ . The MI between the input and the output of the channel can be expressed as follows:

$$I(X_1, \dots, X_n; Y) = \sum_{i=1}^n I(X_i; Y | X_{i-1}, \dots, X_1) \quad (6.7)$$

where the right hand side is obtained by applying the chain rule of MI [1]. Equation (6.7) hides a very powerful concept. In fact, it is possible to interpret the stochastic relation between the transmitted bit at the  $i$ -th stage  $X_i$  and the received observable  $Y$ , given exact knowledge of  $X_1, \dots, X_{i-1}$ , as a particular memoryless channel. In fact, although it is true that there is dependence between the bits corresponding to a specific constellation symbol, the successive realizations are conditionally independent over time. The overall  $2^n$ -ary input channel can then be seen as a cascade of channels. Assuming that it is possible to practically achieve “error-free” performance on a memoryless channel through channel coding techniques, one can design a good code for the channel  $(X_1; Y)$ , then a good code for the channel  $(X_2; Y)$  given the knowledge of  $X_1$ , and so on. MLC-MSD is the straightforward decoding solution for this scheme. In particular, for each (sub)channel it is possible to achieve error free performance at any rate below the MI of the (sub)channel (uniform input is assumed). With MLC-MSD, it is therefore possible to achieve error free performance at any rate below the MI  $I(X_1, \dots, X_n; Y)$  of the overall channel. We refer the interested reader to [105], where information theoretic aspects of MLC-MSD are covered in details. Among the possible code design techniques for MLC-MSD, selection of the code rate based on the MI of the (sub)channels is referred to as *MI rule* in [105]. According to this rule, efficient coding can be achieved by choosing for the  $i$ -th (sub)channel,  $i = 1, \dots, n$ , a powerful (and possibly *ad hoc*) binary code with rate slightly lower than  $I(X_i; Y | X_{i-1}, \dots, X_1)$ .

### 6.3.2 Code Selection with the MI Rule

We now show how the results in Section 6.2 affect MLC design based on the MI rule and LDPC coding. Since, over a memoryless channel, the performance of a particular LDPC code without short cycles in the code graph depends almost only on the MI between the input and the output of the channel, as discussed in Section 6.2, we collect a library of LDPC codes, characterized by various rates.

For simplicity, we restrict ourselves to an AWGN channel and linear modulations with given  $2^n$ -ary constellations. The goal is to select an ordered  $n$ -uple of LDPC codes in the code library, in order to obtain an overall MLC whose performance can be considered error-free for SNR values below a given threshold. The obtained MLC should have the highest possible rate.

We start with the following consideration: given a constellation, if it is possible to specify a BER threshold  $P_b^*$  and to guarantee that at each level in the MLC scheme the error rate is below this threshold, *then* by specifying a sufficiently low value of  $P_b^*$ , it is possible to obtain arbitrary small BER for the overall MLC scheme. This reasonable assumption has an important impact on the code design algorithm described in the following. In fact, consider the property described in Section 6.2, i.e., that the performance of LDPC codes does not depend on the particular channel but only on the channel MI. Every code will be characterized by its own BER versus MI curve. If two LDPC codes have the same rate, for a specified value of  $P_b^*$  one of them will have a better or equal performance, measured as input MI needed to achieve a BER equal to  $P_b^*$ , than the other. If both codes fulfill the required design constraints, there is no point in using the code that requires higher MI to achieve the specified BER equal to  $P_b^*$ . Thus, for each code rate in the LDPC code library, it is possible to eliminate all codes but the best one. We can thus assume that in the code library the codes will have different rates.

In Figure 6.5, an 8-PSK constellation with natural bit mapping is shown. In Figure 6.6, the MIs of the three sub-channels, as well as with the overall MI, are shown for an 8-PSK constellation transmitted through an AWGN channel. The MI is shown as a function of the SNR  $E_s/N_0$ , where  $E_s$  is the average 8-PSK symbol energy and  $N_0$  is the one-sided noise power spectral density. The three curves are monotonically non-decreasing. The horizontal dotted line represents the MI needed for a particular code to achieve a BER equal to  $P_b^*$ . One can observe that the first subchannel intersects the horizontal line in the rightmost point, the third in in the leftmost and the second in the middle. The meaning of each intersection is that beyond the corresponding SNR, the considered code will guarantee a BER lower than  $P_b^*$  if used for

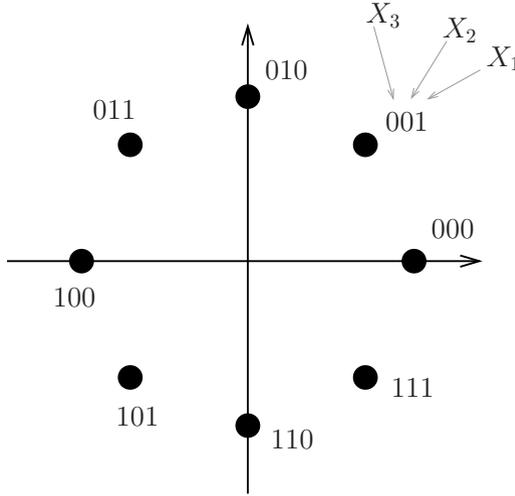


Figure 6.5: Pictorial representation of an 8-PSK constellation with natural mapping.

channel coding on the corresponding sub-channel. If the actual SNR is below that corresponding to the intersection, then the system is in *outage*.

The previous considerations allow one to derive the following MLC design algorithm based on the selection of the proper LDPC codes. The procedure is graphical and the basic graph, shown in Figure 6.7 for a specific scenario (relative to Example 6.1 considered in the following), can be constructed according to the following steps.

- On a graph, plot the MIs of the subchannels versus the SNR. The aggregate MI should be plotted as well.
- For each code in the LDPC code library, plot a horizontal line intersecting the MI axis at a MI equal to the value needed by the LDPC code to obtain a BER equal  $P_b^*$ .
- Find the intersection of each code line with each sub-channel MI curve and draw the projection of each intersection on the SNR axis.

In order to find the best code, i.e. the highest-rate code with SNR outage threshold below a given  $\text{SNR}^*$ , it is sufficient, for each sub-channel, to find the intersection of the corresponding MI curve with the code line which has the

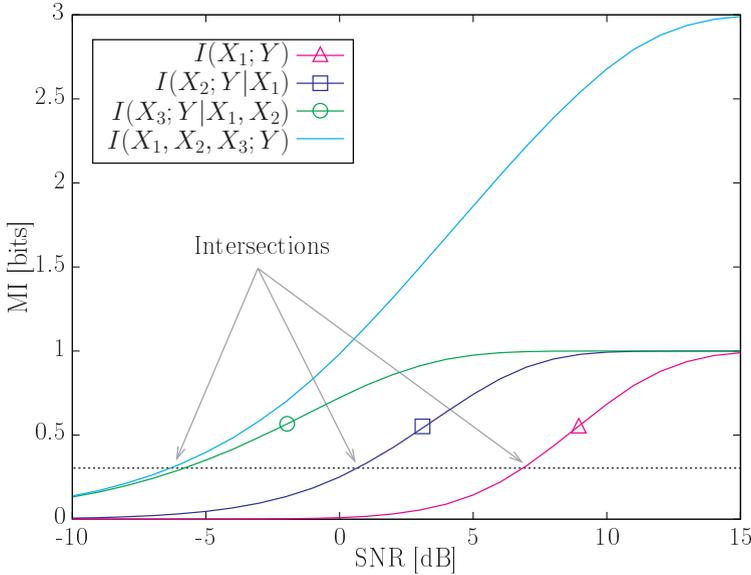


Figure 6.6: MIs  $I(X_1; Y)$ ,  $I(X_2; Y|X_1)$  and  $I(X_3; Y|X_1, X_2)$  of the subchannels and MI  $I(X_1, X_2, X_3; Y)$  of the overall channel for an 8-PSK constellation with natural mapping.

highest SNR below  $\text{SNR}^*$ .

**Example 6.1** In Figure 6.7, the proposed graphical algorithm is illustrated for an 8-PSK constellation with natural mapping. Three horizontal lines are plotted, assuming a library which contains only three codes with rates 0.25, 0.57, and 0.88, respectively, for ease of exposition. However, we remark that the approach is general and the use of larger libraries can lead to better results. The intersections are shown on the SNR axis. The intersections corresponding to the first subchannel are marked with triangles, those corresponding to the second subchannel with squares, and those corresponding to the third subchannel are marked with circles. For a given  $\text{SNR}^*$  value, it is sufficient to find the rightmost triangle, the rightmost square and the rightmost circle to the left of this  $\text{SNR}^*$  value. Each symbol, either triangle, circle or square, identifies exactly a subchannel and a code in the code library.

Three LDPC codes were selected using the graph in Figure 6.7 and considering  $\text{SNR}^*$  equal to 10.5 dB, as reported in the figure. The resulting codes can be characterized as follows:

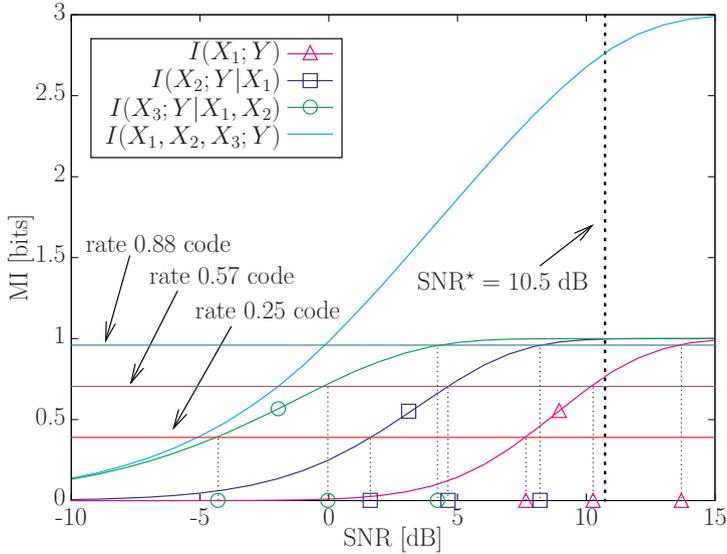


Figure 6.7: Graphical scheme used for MLC code selection.

channel 1 ( $X_1; Y$ ): rate 0.57 code  
channel 2 ( $X_2; Y|X_1$ ): rate 0.88 code  
channel 3 ( $X_3; Y|X_1, X_2$ ): rate 0.88 code.

Had we chosen  $\text{SNR}^*$  equal to 9 dB, the selected codes would have been as follows:

channel 1 ( $X_1; Y$ ): rate 0.25 code  
channel 2 ( $X_2; Y|X_1$ ): rate 0.88 code  
channel 3 ( $X_3; Y|X_1, X_2$ ): rate 0.88 code.

In Figure 6.8, the BER performance of the system designed considering  $\text{SNR}^*$  equal to 10.5 dB is shown. The component codes, with codeword length 10000 and rates 0.57, 0.88, and 0.88, correspond to three regular LDPC codes: (3, 7), (3, 25), and (3, 25), respectively. The overall code rate is  $0.57 + 0.88 + 0.88 \simeq 2.33$  bits per channel use. The predicted convergence threshold, i.e., the SNR of the rightmost intersection point within the selected codes, is 10.11 dB, corresponding to a bit SNR  $E_b/N_0 = 6.43$ . The BER performance of each sub-channel is shown as well. One can observe that there is a good match between the design outage threshold and the actual outage threshold.

We remark that, in order to achieve good performance, the first decoding stages should not introduce errors, since this would affect the next decoding

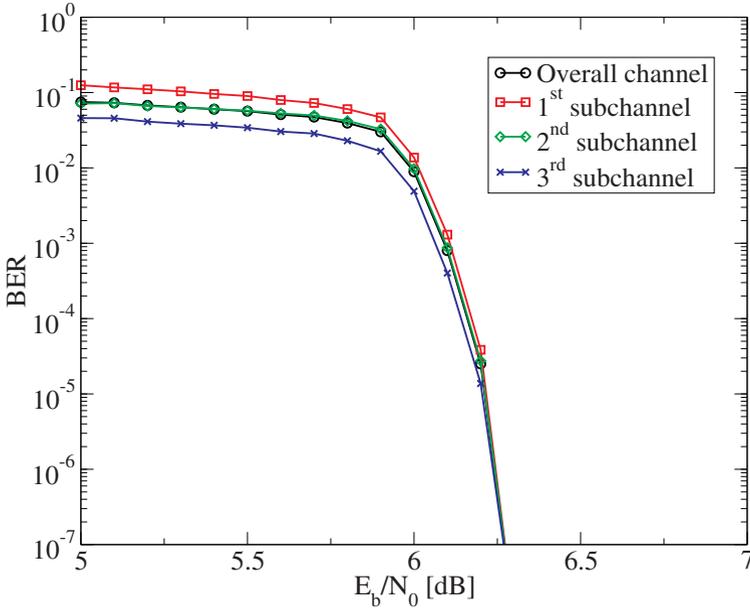


Figure 6.8: BER performance of an 8-PSK MLC-MSD system designed for convergence at SNR equal to 10.11 dB ( $E_b/N_0 = 6.43$  dB).

stages, possibly causing avalanche error propagation. The presented numerical results were obtained with *regular* and *quasi-regular* LDPC codes.<sup>2</sup> This choice is due to the fact that the construction of codes without short cycles is easier for regular LDPC codes, especially in the case of low-degree variable and check nodes. The absence of short cycles is useful in order to lower (or make disappear) the error floor characterizing most powerful codes.

## 6.4 Concluding Remarks

In this chapter, the performance of LDPC codes over memoryless channels has been discussed. The fact that, with a good approximation, the performance of an LDPC code transmitted over a binary-input memoryless channel does not depend on the particular channel but only on the MI between the input and the output of the channel, has been highlighted. This property has several implications, among which that of enabling efficient design of multilevel codes.

<sup>2</sup>We denote as *quasi-regular* LDPC a code with only two, possibly contiguous, allowed variable or check node degrees

In particular, an algorithm for multilevel code design based on the selection of a group of good LDPC codes from an LDPC code library has been described, and some design examples have been given.