POLITECNICO DI TORINO

SCUOLA DI DOTTORATO
Dottorato in Matematica per le Scienze dell'Ingegneria – XX ciclo
Settore scientifico-disciplinare: MAT/05 ANALISI MATEMATICA

Ph.D. Thesis

Generalized serial turbo coding ensembles: analysis and design

Federica Garin

Advisor
prof. Fabio Fagnani

Director of the doctoral program
prof. Nicola Bellomo

March 2008

Abstract

The topic of this thesis falls within modern channel coding theory, and consists in the analysis of a wide class of serial turbo codes. Instead of binary codes only, we consider general codes constructed from an arbitrary finite Abelian group, in order to match the symmetries of a large family of channels. Moreover, with respect to classical serial turbo schemes, we relax some assumptions on the constituent encoders of the concatenation and we allow for more freedom in the choice of the interleaver. This setting includes as special cases usual binary serial and parallel turbo codes, as well as turbo trellis-coded modulation for Gaussian channels with geometrically uniform input constellation, e.g. m-PSK.

In this general setting, we prove rigorous bounds on the average error probability, generalizing an upper bound already known in the binary case and providing a matching lower bound which is new even for binary codes. We obtain the interleaver gain, i.e. the asymptotic decay of average error probability when the codewords' length tends to infinity, and moreover we study the behaviour with respect to channel's signal-to-noise ratio.

In the classical binary setting, we give a more detailed analysis: by the study of the minimum distance distribution, together with expurgation techniques, we find the asymptotic behaviour of typical error probability. Typical error probability decays sub-exponentially fast, as opposed to the polynomial decay of the average, thus showing that the average is strongly affected by a small fraction of bad codes. However, the design parameters suggested by the average-based analysis for the constituent encoders are confirmed also in the typical behaviour.

Then, we consider another family of binary codes that belongs to the generalized setting we introduced, and at the same time can be seen as structured Low-Density Parity-Check codes (a generalized version of Repeat-Accumulate codes). We discuss average-based analysis, including an expurgated sub-ensemble, and we compare theoretical predictions, which are average-based and obtained for maximum likelihood decoding, with simulation results which use message passing decoding. We propose a new decoding algorithm which both improves performance and allows a density-evolution analysis. Summarizing theoretical and simulative results, we describe some guidelines for the design of the encoders.

Acknowledgments

First of all, I wish to thank my advisor: for having been an enthusiastic teacher; for many hours spent together in front of a blackboard; for miraculous travel-funding; for organizing interesting classes and seminars; and also for the fun outside work-hours.

In addition to my advisor, other people contributed to the research presented in this thesis: Daniele Capirone, Giacomo Como and prof. Roberto Garello. I am very indebted with Giacomo, who taught me how to study typical error probability using expurgation and how to analyze message-passing algorithms using density evolution, and with Daniele, who helped me in the study of structured LDPC codes, especially in implementing different decoding algorithms and density evolution.

I also wish to thank prof. Paul Siegel, who came from California to Torino to teach a very interesting class, and who invited me for a visit at University of California at San Diego: the nine months I spent at UCSD have been an exciting opportunity to attend classes and seminars from world-renown professors, and to be a member of a big and active research group.

Apart from academic matters, my staying in San Diego has also been a great life experience. A big hug to all the friends who made me feel so happy there: my labmates, the Italian friends, the International Cooking Club mates, those who hosted me (in San Diego and in Berkeley), and those who shared with me some journey.

In Torino, too, my everyday routine has been made much happier by my office-mates and friends: thank you! A special thanks goes to Giacomo: I have been very lucky to share my studies on coding theory with him, from whom I have learnt so much. And thanks to Paolo and Sophie, for patiently checking correctness of some proofs.

Last but not least, thanks to my family, who makes all my dreams possible.

Table of contents

Chapter 1

Introduction

1.1 Modern channel coding: introduction and state of the art

Channel coding theory is the study of how to add redundancy to a source message in order to obtain a correct received message after the decoding, even when the communication channel is noisy. In his seminal work [70], Shannon gave a mathematical formalization of the problem of digital communication, introducing probabilistic models of sources and channels and studying both source coding (data compression) and channel coding (error correction).

It is clear that error probability can be made arbitrarily small by adding more and more redundancy, and letting the ratio of the lengths of information words and corresponding codewords (called rate) go to zero. More surprisingly, Shannon's channel coding theorem guarantees that asymptotically vanishing error probability can be obtained also by introducing only bounded redundancy, at the price of augmenting code complexity: there is a threshold associated with the channel, called capacity, such that for any rate below capacity you can find a sequence of codes with such fixed rate and with growing length achieving asymptotically vanishing error probability. Shannon's proof was based on probabilistic arguments, and showed that randomly constructed codes are good with high probability. However, random codes have in general an unfeasible complexity, particularly in the decoding process, which usually requires a time exponential in the codewords length. Thus, for more than forty years, classical algebraic coding theory (see e.g. [50]) focused on the construction of codes with very strong structural properties, in order to allow an easy decoding, but the performance of such codes was far away from the theoretical limit given by Shannon capacity.

Two modern classes of codes give an excellent performance/complexity tradeoff:

turbo codes, introduced in 1993 [9] (see also [10]), and Low-Density Parity-Check (LDPC) codes, introduced in 1963 [33] but widely studied only recently, after their re-discovering in [49] (see also [48]). The success of these two families of codes is due to a good balance of enough randomness, which gives good performance, and enough structure, which is exploited by a suitable iterative decoding algorithm. Iterative decoding is suboptimal with respect to the best possible decoder ('maximum likelihood', which minimizes the error probability for given code and channel), but its performance usually approaches the optimal one. The sub-optimal decoding significantly reduces the decoding complexity, from exponential to linear time with respect to the codewords length, thus allowing the use of very long codes and so improving performance.

The idea to concatenate simple constituent encoders in order to get complex but structured codes was introduced by Forney [28], who considered serial concatenation of convolutional codes. The turbo schemes by Berrou et al. [9] add into the concatenation also a random permutation, thus obtaining a very good trade-off in between randomness and structure, which is the key of modern channel coding. The classical binary turbo schemes are obtained by a parallel concatenation of two encoders: the information bits are fed in two distinct encoders and the codeword is the juxtaposition of the two codewords. However, before entering in the second encoder, the information bits are 'interleaved', i.e. permuted, and this adds to the error-correcting capability of the scheme. The iterative decoder takes advantage of the concatenated structure, by decoding separately and optimally each individual code, and iteratively exchanging information from one decoder to the other, thus reaching a suboptimal but good decoding of the overall scheme.

After the amazing success of turbo codes, many different turbo-like schemes have been proposed in the literature. Serial turbo codes were introduced by Benedetto et al. in [3], putting together Forney's serial concatenations with the interleaving technique of Berrou et al. and devising an ad-hoc iterative decoder for these new schemes. Since then, concatenations of more than two encoders and more than one interleaver have been considered, both in a serial structure or in a mixed serial and parallel way (see e.g. [4, 39, 41]). All these schemes go under the name of 'turbo-like' codes. Despite very good intrinsic properties of some multiple concatenated codes (see e.g. [56, 41, 2]), these schemes have not yet had a significant impact in the applications because of the difficulty to find efficient iterative decoding algorithms (see e.g. [13] for a discussion of different decoding algorithms and their behaviour).

Low-Density Parity-Check codes are linear codes defined as the kernel of a sparse matrix (the parity-check matrix). The constraints defining the codewords can equivalently be represented by a graph, called the Tanner graph, and the decoding algorithm can be seen as a message-passing algorithm running on the graph. Sparseness

of the matrix (and hence of the associated graph), allows to use the approximation that all incoming messages in each node are independent, thus highly reducing the complexity. The classical regular family of LDPC codes introduced by Gallager has a parity-check matrix chosen uniformly at random among matrices with fixed number of ones per row and per column (equivalently degree of nodes in the Tanner graph), while more general irregular families have been considered recently [47], where the degrees are allowed to vary according to different distributions. This additional degree of freedom in the design of the coding scheme has allowed to find capacity-achieving families of LDPC codes [61, 57].

One drawback of LDPC codes is that, despite their linear decoding complexity, in general they have quadratic encoding complexity, because encoding requires the multiplication of the input vector times the generating matrix, which is not sparse. On the contrary, for turbo-like codes, the constituent encoders are usually convolutional encoders, that can be seen as finite-state machines with linear update of the state and of the output, so that the encoding complexity is linear in the length. The issue of encoding complexity of LDPC codes has been addressed in two different ways. On one side there are the results in [62], which allow to construct, for given generic LDPC matrix, equivalent generating matrices with lower encoding complexity. On the other side, there are the constructions of parity check matrices structured in a such a way that allows easy encoding. A successful construction is the one using matrices with a staircase part (i.e. a sub-matrix with ones on the diagonal and on the lower diagonal, and zeros everywhere else), so that the encoder can be seen as a serial concatenation of a repetition code, an interleaver and an accumulator: this gives Repeat-Accumulate codes and their generalization, the Irregular Repeat-Accumulate (IRA) codes introduced in [38].

The extremely good performance of turbo and LDPC codes, shown by Monte-Carlo simulations, has attracted a lot of interest in the theoretical analysis of such codes and of the associated decoding algorithms, in order to get a deeper understanding of their behaviour and to guide code design. On the one side, analysis can focus on intrinsic properties of the codes, and on the other side it can look for properties of the decoding algorithms. For turbo codes, both parallel and serial, the study of the code has been initiated by Benedetto et al. [6, 7, 3]: they studied the behaviour of the error probability under the theoretical assumption of optimal maximum likelihood (ML) decoding, and averaging among all interleavers picked uniformly at random (uniform interleaver). Results along these lines are also in [75, 19, 17, 39, 23]. Another interesting study of the intrinsic properties of turbo codes concerns the minimum distance, which dominates the ML error probability at high signal-to-noise ratio (SNR). For classical parallel turbo codes, Breiling [14] showed that minimum distance can grow at most logarithmically with the length,

while with more of two encoders, or for serial concatenations, better minimum distance can be achieved [41, 2, 55]; results on the distance spectrum of turbo codes can be found in [68]. For LDPC codes, the study of the minimum distance and ML error probability has been considered in Gallager's thesis [33], and new interesting results can be found e.g. in [52].

The study of iterative decoding algorithms is one of the most challenging current open problems in coding theory. A huge literature is devoted to this subject, and many interesting results have been presented, even though a full understanding of this topic has not yet been reached. We will quote here only a very small and clearly non-exhaustive selection from this literature, identifying some of the main research lines; a satisfactory summary of results in this research are can be found in the book [63]. The first analysis of a suboptimal iterative decoding algorithm dates back to Gallager's thesis [33], while the first important results on turbo decoding are given in [59]. A detailed description of how graphical models can describe codes and their decoding algorithm can be found in [78, 73]. The various iterative algorithms proposed in the coding literature have been recognized as instances of Pearl's 'belief propagation' algorithm [51], and for some of the most popular algorithms the solutions to which the algorithm could converge have been characterized with variational methods imported from statistical physics, such as the Bethe free energy approximation (see e.g. [79]). A very important tool that allows to predict convergence of the decoder is density evolution, introduced in [60] for LDPC codes, and applied in [18] also to turbo codes: it consists in considering the probability distribution (density) of messages sent while running the algorithm, under the assumption that no loop had occurred up to that given time, and describing its evolution as a discrete-time dynamical system. This technique is particularly effective in predicting the decoding behaviour on the binary erasure channel (BEC), where the messages are very simple and the density evolution system is finite-dimensional (usually one-dimensional). For other channels, the dynamical system is infinite-dimensional, and it is necessary to find some suitable projection with lower dimension whose behaviour describes well the system, or to make approximations which reduce the number of possible exchanged messages; the first approach is used in so-called exit charts method [76], and the second in the Gaussian approximation [21], both particularly used for turbo codes. A more refined analysis, giving better prediction than density evolution, is the so-called 'finite length analysis', introduced in [1].

The largest part of coding literature is devoted to linear binary codes, i.e. vector spaces over the finite field with only two elements GF(2). This is due both to the simplicity of such codes and to the importance of bits in digital communications. However, in many applications, the actual transmission is not binary: in order to gain spectral efficiency, the modulation uses a larger set of signals. In many cases, a pragmatic approach is used, which consists in designing a good binary code,

optimized for a binary-input output-symmetric channel, and then using it also in different contexts, by mapping blocks of bits to symbols in the input constellation for the channel. In many cases, though, is is possible to design ad-hoc codes for the given channel, particularly when the channel has symmetries which can be fully exploited by imposing a suitable algebraic structure on the code, e.g. using a module over some ring, or simply a group.

The study of group codes was initiated by Slepian, who actually formulated most of his coding results in this general setting (see e.g. [71, 72]). Classical results were established by Ingemarsson and Ungerboeck (see e.g. [37, 77]). More recently, there has been a vivid interest in convolutional codes over rings and over groups, the so-called trellis-coded modulation (TCM), where 'trellis' refers to the linear finite-state machine describing the convolutional encoder: see [30, 5, 46, 40, 25, 26, 27, 31] and see [29] and [45] for a mathematical formulation of a family of channels for which these codes are well-suited. Most results on group convolutional codes are restricted to finite Abelian groups, because in this setting it is possible to prove deeper results, by exploiting the ring structure of the cyclic group $\mathbb{Z}_m$ and then extending the results by the Kronecker decomposition theorem.

With the advent of modern high-performance codes, most of the literature has focused on a pragmatic design, where binary turbo or LDPC codes are designed and optimized for binary-input output-symmetric channels, and then applied to all channels. Most schemes presented in the literature belong to this class; performances are obtained by simulations and most of the research is focused on optimizing the mapping of the coded bits into points of the constellation (see e.g. [44, 64, 32]). A few works have introduced some results on non-binary modern codes. For turbo codes, this study, which goes under the name of turbo-TCM, has been initiated in the pioneering works [34, 53] and recently in [24]. For non-binary LDPC codes, some interesting results have been obtained, but this research mostly focuses on non-binary fields, not on more general groups [16, 8, 22]; an interesting exception is [74], where rings are considered.

It is interesting to note that also some more classical information-theoretic questions about symmetric channels and group codes have appeared in the literature very recently (see e.g. [15]), showing that there are still many challenging open questions in this area.

1.2 Overview of results and thesis outline

In this dissertation, we present a very general serial turbo scheme, which is constructed on a generic finite Abelian group in order to match the symmetries of a wide class of channels. In this general setting, we study ML error probability, and in particular its asymptotic properties with respect to the codewords' length and to

the channel's SNR. Following the lines of [3], we study performance averaged over ensembles where the constituent encoders are fixed while the interleaver is uniformly distributed. However, we allow a more general choice of the set to which the interleaver belongs: it can be the group of permutations, or one of its subgroups, or even a more general group whose action satisfies some suitable regularity properties. This setting includes as special cases usual binary serial and parallel turbo codes, as well as turbo trellis-coded modulation for Gaussian channels with geometrically uniform input constellation, e.g. m-PSK. We prove rigorous bounds on the average error probability, generalizing an upper bound already known in the binary case [7, 3, 39, 23] and providing a matching lower bound which is new even for binary codes. We obtain that, under some assumptions on the constituent encoders and at sufficiently high but fixed SNR, average error probability is vanishing when the length goes to infinity, with a polynomial decay whose speed is described by the solution of a combinatorial optimization problem, involving in general both constituent encoders. Our bounds also underline the dependence on the SNR, characterized as the solution of a second optimization problem. In special cases, such as in the classical binary setting, these two optimization problems simplify and involve separately the two encoders, and thus provide simple design criteria.

In the special case of classical serial turbo schemes, we can go further in the study of the ensemble and find not only the average, but also the typical error probability. To do so, it is essential to obtain refined bounds on the constituent encoders' weight enumerators, with techniques from [41]. First we study the distribution of minimum distance: we find bounds for its left tail, with techniques mostly from [41], and we find a deterministic upper bound which generalizes a result in [2], there obtained for Repeat-Convolute codes. The new deterministic upper bound is asymptotically tighter than the best known bound for minimum distance of serial turbo codes, presented in [55]. Then, from the results on minimum distance, we obtain the typical asymptotic behaviour of ML error probability at high SNR, using a conditioning technique known as 'expurgation'. This approach is classical in information theory and in the LDPC literature (see e.g. [33]), but is new for turbo-like codes. Our analysis shows that typical error probability decays sub-exponentially fast, as opposed to the polynomial decay of the average, thus showing that the average is strongly affected by a small fraction of bad codes. However, the design parameters suggested by the average-based analysis for the constituent encoders are confirmed also in the typical behaviour: the speed of the decay increases with the free distance of the outer encoder, while performance is also improving when the effective free distance of the inner encoder (i.e. the minimum distance if the encoder is restricted to input weight two), but the dependence appears only as a multiplicative term.

In the binary case, we consider also another family of coding schemes belonging to the general serial turbo scheme, generalizing Repeat-Accumulate codes and which is a family of linear-time encodable and decodable LDPC codes. We discuss

average-based analysis, and we note that in order to find a design parameter for the inner encoder we need to consider a smaller sub-ensemble. We compare theoretical predictions, which are average-based and obtained for maximum likelihood decoding, with simulation results which use iterative decoding: we find that in some cases a bad behaviour of the decoder strongly deteriorates performance. We conjecture that this is related to the high number of cycles in the structured part of the graph. We propose a different decoding algorithm, which runs on a modified graph where some nodes are gathered together, in such a way to destroy structured cycles: this both improves performance and allows a density-evolution analysis.

Thesis outline

In Chapter 2, we describe the channel model we are considering (memoryless G-symmetric channels) and group codes. Particularly, we focus on convolutional codes over Abelian groups, which are the constituent elements of turbo concatenations. We give here some properties of convolutional codes which will be instrumental to the derivations in the next chapters; some of them are classical, and some are new.

In Chapter 3, we introduce a general serial turbo coding ensemble and we analyze its average word and symbol error probability, providing an upper and a lower bound, which are asymptotically tight when the length goes to infinity.

In Chapters 4 and 5 we restrict our attention to two particular cases of the very general ensemble of Chapter 3, both of them binary.

Chapter 4 is joint work with Giacomo Como. It presents a detailed analysis of classical binary uniform interleaver serial ensemble, for which we find precise estimations of the minimum distance distribution and then, by expurgation techniques, the typical behaviour of error probability.

Chapter 5 is joint work with Daniele Capirone and Giacomo Como. It deals with a family of codes that generalize Repeat-Accumulate codes, and can be seen both as particular systematic serial turbo codes and as structured LDPC codes. We discuss average-based analysis, including an expurgated sub-ensemble, and we compare ML and average-based predictions with simulation results using message passing. We propose a new decoding algorithm which both improves performance and allows a density-evolution analysis, and we describe some directions on encoders design.

Finally, in Chapter 6, we summarize the results presented in this thesis, and we describe future research directions that arise naturally from this work.

This thesis is partly based on the following papers:

- F. Fagnani, F. Garin, "Analysis of serial concatenation schemes for non-binary modulations", in Proceedings of ISIT 2005, pp. 745-749 (Adelaide, SA, Australia), September 5-9, 2005.

- G. Como, F. Garin, F. Fagnani, "ML performances of serial turbo codes do not concentrate!", in Proceedings of 4th International Symposium on Turbo Codes (Munich, Germany), April 3-7, 2006.
- F. Fagnani, R. Garello, F. Garin, "Average ML Asymptotic Performances of Different Serial Turbo Ensembles", in Proceedings of ISIT 2006, pp. 572-576 (Seattle, WA, USA), July 9-14, 2006.
- F. Garin and F. Fagnani, "Analysis of serial turbo codes over Abelian groups for geometrically uniform constellations", submitted to *Siam Journal on Discrete Mathematics*, April 2007. Pre-print available on-line: `http://calvino.polito.it/rapporti/2007/pdf/20_2007/art_20_2007.pdf`
- F. Garin, G. Como, F. Fagnani, "Staircase and other structured linear-time encodable LDPC codes: analysis and design", in Proceedings of ISIT 2007, pp. 1226-1230, (Nice, France), June 25-29, 2007.
- D. Capirone, G. Como, F. Fagnani, F. Garin, "Nonbinary decoding of structured LDPC codes", to appear in Proceedings of 2008 International Zurich Seminar on Communications, (Zurich, CH), March 12-14, 2008.
- D. Capirone, G. Como, F. Fagnani, F. Garin, "Density Evolution of Nonbinary Decoding Applied to Structured LDPC Codes", submitted, ISIT 2008.
- F. Garin, G. Como, F. Fagnani, "Typical minimum distance and ML error probability of serial turbo codes", in preparation.

Chapter 2

Group codes

In this chapter, we introduce the channel model which we will be considering throughout this thesis and we introduce group codes, whose algebraic properties are perfectly matching the symmetries of the channel. After some general properties of all group codes, we focus on convolutional group codes: we gather here many properties, some classical and some new, which will be instrumental to the derivations in next chapters.

2.1 Notation

We fix here some notation that will be used throughout this thesis.

Given a set Ω and $A \subseteq \Omega$, the symbol $\mathbb{1}_A : \Omega \to \{0,1\}$ will denote the indicator function of A, i.e. $\mathbb{1}_A(x) = 1$ if and only if $x \in A$. $|A|$ will denote the cardinality of A.

We will denote by $\mathbb{N}$ the set of non-negative integers, and by $\mathbb{N}^*$ the set of positive integers.

Vectors will always be column vectors, and will be denoted by boldface letters. We will denote by $\boldsymbol{e}_j$ a vector of the appropriate length (clear by the context or explicitly stated) made by all zeros except a one in position j. Given two sets A,B, B^A will denote vectors with entries in B, having length $|A|$ and components indexed by elements of A instead of integers $1, \ldots, |A|$.

By log and exp we will denote logarithm and exponential with respect to the same basis $b > 1$.

Given groups G and H, $\mathrm{Hom}(G,H)$ will denote the group of all homomorphisms from G to H, while $\mathrm{Aut}(G)$ will be the group of automorphisms of G.

2.2 Symmetric channels and block encoders over Abelian groups

2.2.1 Weights

In this thesis, we will deal different kinds of weight. We propose here a general definition. First a notation: for any $\boldsymbol{w} \in \mathbb{N}^k$, we put $|\boldsymbol{w}| = \sum_j \boldsymbol{w}_j$.

Definition 2.1 A *weight* on an Abelian group Z consists of a positive integer ρ and of a map $\boldsymbol{w} : Z \to \mathbb{N}^\rho$ satisfying the following properties:

1. $\boldsymbol{w}(0) = \mathbf{0}$;
2. $|\boldsymbol{w}(z^1 + z^2)| \leq |\boldsymbol{w}(z^1)| + |\boldsymbol{w}(z^2)|$ for every z^1,$z^2 \in Z$;
3. $\{\boldsymbol{e}_1, \ldots, \boldsymbol{e}_\rho\} \subseteq \boldsymbol{w}(Z)$ (here $\boldsymbol{e}_j \in \mathbb{N}^\rho$).

A few considerations on the above definition:

- Item 2. simply says that summation in Z can not create any extra weight;
- Item 3. is a simple minimality assumption which ensures that the full semigroup structure of $\mathbb{N}^\rho$ is used.

Whenever we have a weight $\boldsymbol{w}$ we will consider its natural extension to vectors by componentwise sum

$$\boldsymbol{w} : Z^N \to \mathbb{N}^\rho\,, \quad \boldsymbol{w}(\boldsymbol{z}) = \sum_j \boldsymbol{w}(z_j)\,.$$

Given $\boldsymbol{h} \in \mathbb{N}^\rho$, we will use the following notation

$$Z^N_{\boldsymbol{h}} = \{\boldsymbol{z} \in Z^N \ : \ \boldsymbol{w}(\boldsymbol{z}) = \boldsymbol{h}\}$$

Moreover, if $\boldsymbol{h} \in \mathbb{N}^\rho$ we will use the notation

$$\binom{N}{\boldsymbol{h}} = \begin{cases} \frac{N!}{\boldsymbol{h}_1! \cdots \boldsymbol{h}_\rho! (N - |\boldsymbol{h}|)!} & \text{if } |\boldsymbol{h}| \leq N \\ 0 & \text{otherwise} \end{cases}$$

The following result will be useful later

Lemma 2.1 Suppose $\boldsymbol{w}$ is a weight on Z. For every $\boldsymbol{h} \in \mathbb{N}^\rho$ we have that

$$\binom{N}{\boldsymbol{h}} \leq |Z^N_{\boldsymbol{h}}| \leq \big(|Z|\, N\big)^{|\boldsymbol{h}|}$$

□

Proof: For $i = 1, \ldots, \rho$, let $\eta_i \in Z$ be such that $\boldsymbol{w}(\eta_i) = \boldsymbol{e}_i \in \mathbb{N}^\rho$ (they surely exist by point 3. of definition of weight). The lower bound is trivially true if $|\boldsymbol{h}| > N$. Otherwise, consider the words in Z^N with support cardinality $|\boldsymbol{h}|$ made by exactly $\boldsymbol{h}_j$ times η_j, for $j = 1, \ldots, \rho$: there are $\binom{N}{\boldsymbol{h}}$ such words, and all of them have invariants vector weight $\boldsymbol{h}$.

The upper bound is clearly true if $\boldsymbol{h} = \boldsymbol{0}$. Assume therefore that $\boldsymbol{h} \neq \boldsymbol{0}$ For any $\boldsymbol{z} \in Z^N_{\boldsymbol{h}}$ consider the subset $\mathcal{J}$ of indices $j \in \{1, \ldots, N\}$ for which $\boldsymbol{z}_j \neq 0$. Clearly, $1 \leq |\mathcal{J}| \leq |\boldsymbol{h}|$. It thus follows that the number of elements in $Z^N_{\boldsymbol{h}}$ can be upper bound considering all possible subsets $\mathcal{J}$ of cardinality $1 \leq |\mathcal{J}| \leq |h|$ and all the possible elements of Z in the positions in $\mathcal{J}$. In other words

$$|Z^N_{\boldsymbol{h}}| \leq \sum_{j=1}^{|\boldsymbol{h}|} \binom{N}{j} |Z|^j \leq |Z|^{|\boldsymbol{h}|} \sum_{j=1}^{|\boldsymbol{h}|} \binom{N}{j}$$

It is now sufficient to use the inequality $\sum_{j=1}^{|\boldsymbol{h}|} \binom{N}{j} \leq N^{|\boldsymbol{h}|}$ to obtain the result. ■

Two examples of weights, always available on any set Z, are the following:

- **Hamming weight:** $\rho = 1$, $\mathrm{w_H}(z) = 1 - \mathbb{1}_0(z)$;
- **Type weight:** $\rho = |Z| - 1$, or better $\boldsymbol{w}_\mathrm{T}(z) \in \mathbb{N}^{Z \setminus \{0\}}$, because we prefer indexing the components of $\boldsymbol{w}_\mathrm{T}(z)$ directly by the elements in $Z \setminus \{0\}$ instead of by integers $1, \ldots, |Z| - 1$; define $\boldsymbol{w}_\mathrm{T}(z)_a = \mathbb{1}_a(z)$ for every $a \in Z \setminus \{0\}$.

With this notation, we clearly have $|\boldsymbol{w}_\mathrm{T}(\boldsymbol{z})| = \mathrm{w_H}(\boldsymbol{z})$. Notice moreover, that for any weight on Z, it necessary holds

$$\mathrm{w_H}(\boldsymbol{z}) \leq |\boldsymbol{w}(\boldsymbol{z})| \leq \mathrm{w_{max}} \mathrm{w_H}(\boldsymbol{z}) .$$

where $\mathrm{w_{max}} = \max_{z \in Z} |\boldsymbol{w}(z)|$.

On Abelian groups, it will be particularly important to consider the weights compatible with the algebraic structure, as defined below.

Definition 2.2 Given an Abelian group U, a distance d on U is called compatible with the group structure of U if $\mathrm{d}(u,v) = \mathrm{d}(u+w, v+w)$ for all $u, v, w \in U$.

A weight $\boldsymbol{w} : U \to \mathbb{N}$ is called compatible with the group U if there exists a distance d compatible with U such that, for all $u \in U$, $\boldsymbol{w}(u) = \mathrm{d}(u, 0)$.

Notice that if d is a compatible distance on U, the natural extension (by componentwise summation) on U^k remains compatible: it will be denoted with the same symbol d, as well as the associated weight w.

Notice that the Hamming and the type weights are always compatible with any fixed group U.

2.2.2 Symmetric channels

A memoryless channel is described by: an input alphabet $\mathcal{X}$ (which we will always assume is finite), an output alphabet $\mathcal{Y}$, endowed with a σ-algebra $\mathcal{B} \subseteq 2^{\mathcal{Y}}$ and a probability measure μ; a family of transition probability densities $W(\cdot|x)$ on $\mathcal{Y}$, indexed by the inputs $x \in \mathcal{X}$. Such a channel will be denoted by $(\mathcal{X},\mathcal{Y},W)$. In most applications, either $\mathcal{Y}$ is finite, and μ is the counting measure, so $W(\cdot|x)$ are simply probability vectors, or $\mathcal{Y} = \mathbb{R}^n$ and μ is the Lebesgue measure.

To give a formal definition of symmetric memoryless channels, we need to recall some definitions of group actions. Given a group $(G,+)$ with neutral element 0, and given a set A, G acts on A if for every $g \in G$ there exists a map $a \mapsto ga$ from A to A, such that $(h+g)a = h(ga)$ for all $h,g \in G$, and $a \in A$, and $0a = a$ for all $a \in A$. For finite A, the group action of G on A is said to be (simply) transitive if for every $a,b \in A$, there exists a (unique) element $g \in G$ such that $ga = b$. If G acts simply transitively on A, G and A are in bijection, through the map $\theta : G \to A$ defined by $\theta(g) = ga_0$ for any fixed $a_0 \in A$.

Given a probability space $\mathcal{Y}$, with σ-algebra $\mathcal{B}$ and probability measure μ, we say that a group G acts isometrically on $\mathcal{Y}$ if there exists an action of G on $\mathcal{Y}$ consisting of measurable bijections such that $\mu(gA) = \mu(A) \forall A \in \mathcal{B}, \forall g \in G$. If $\mathcal{Y}$ is finite, then all group actions on $\mathcal{Y}$ are isometric. If $\mathcal{Y} = \mathbb{R}^n$, then an action is isometric when all maps $y \mapsto gy$ are isometries of $\mathbb{R}^n$.

Given a group G, a memoryless channel $(\mathcal{X},\mathcal{Y},W)$ is called G-symmetric if:

1. G acts simply transitively on $\mathcal{X}$;
2. G acts isometrically on $\mathcal{Y}$;
3. $W(y|x) = W(gy|gx)$ for every $g \in G$, $x \in \mathcal{X}$, $y \in \mathcal{Y}$.

In this case, the bijection $\theta : G \to \mathcal{X}$ defined by $\theta(g) = gx_0$ for some fixed $x_0 \in \mathcal{X}$ is called an isometric labeling.

The most common examples of G-symmetric channels are the following.

Binary-input output-symmetric channels. $\mathbb{Z}_2$-symmetric channels are known in the coding literature as binary-input output-symmetric (BIOS) channels. Well-known examples are binary symmetric channel (BSC), binary erasure channel (BEC), and binary-input AWGN (BIAWGN) channel.

Geometrically uniform AWGN channels. A n–dimensional constellation is a finite subset $S \subset \mathbb{R}^n$ that spans $\mathbb{R}^n$; we denote with $\Gamma(S)$ its symmetry group, i.e. the group of the Euclidean isometries of $\mathbb{R}^n$ mapping S into S itself. A constellation S is said to be geometrically uniform (GU) with generating group G if G is a subgroup of $\Gamma(S)$ whose action on S is simply transitive.

The simplest example of GU constellation is the 1-dimensional antipodal constellation $\{-1,1\}$ (a.k.a. 2-points Pulse Amplitude Modulation, 2–PAM). A bi-dimensional example is the m-PSK constellation

$$S = \{e^{\frac{2\pi i l}{m}} \ : \ l = 0, \dots m-1\} \subseteq \mathbb{C} \simeq \mathbb{R}^2$$

which always has the generating group $\mathbb{Z}_m$ (seen as rotations of angles multiple of $2\pi/m$) and for even m also has the non-Abelian generating group $D_{m/2}$. For a complete theory of GU constellations and generating groups, see [29] and [45].

Given a GU constellation $S \subset \mathbb{R}^q$ with generating group G, define the *S–AWGN channel* as the memoryless channel $(S,\mathbb{R}^n,W)$ where the family W of n-dimensional transition densities is given by, for every $x \in S$,

$$W(y|x) = N(y-x)\,,$$

where $N(\cdot)$ is the density of a n-dimensional diagonal Gaussian random variable:

$$N(y) = \frac{1}{(2\pi\sigma^2)^{n/2}} e^{-\frac{||y||^2}{2\sigma^2}}\,.$$

The interpretation is that, if $x \in S$ is the transmitted symbol, the received symbol is given by $x+Z$ where Z is a Gaussian random variable of density N.

Other examples of G-symmetric channels can be obtained from the S-AWGN by suitable symmetric quantizations of the channel output, e.g. quantizing with respect to the Voronoi regions of the same constellation S.

m-ary symmetric channels. This is a simple generalization of the BSC: $\mathcal{X} = \mathcal{Y} = \{0,1,\dots,m-1\}$ and $W(y|x) = \epsilon/(m-1)$ if $y \neq x$, $W(y|x) = 1-\epsilon$ if $y = x$. This channel is G-symmetric for any group G with $|G| = m$; in particular, for $G = \mathbb{Z}_m$.

In the study of G-symmetric channels, a key element is the pairwise equivocation probability of a word $\boldsymbol{c} \in G^n$, $\mathrm{P}(\mathbf{0} \to \boldsymbol{c})$, defined as the probability that, for some fixed decoding rule, the decoder will prefer $\boldsymbol{c}$ to $\mathbf{0}$, given that $\boldsymbol{\theta}(\mathbf{0})$ was transmitted. In this thesis, we consider maximum likelihood decoding, with the choice to break ties uniformly at random (or with any given rule on channels such as S-AWGN where ties occur with probability zero), so that

$$\begin{aligned}\mathrm{P}(\mathbf{0} \to \boldsymbol{c}) = &\int_{\mathcal{Y}^n} W_n(\cdot|\boldsymbol{\theta}(\mathbf{0}))\mathbb{1}_{W_n(\cdot|\boldsymbol{\theta}(\boldsymbol{c}))>W_n(\cdot|\boldsymbol{\theta}(\mathbf{0}))}\ \mathrm{d}\mu_n \\ &+ \tfrac{1}{2}\int_{\mathcal{Y}^n} W_n(\cdot|\boldsymbol{\theta}(\mathbf{0}))\mathbb{1}_{W_n(\cdot|\boldsymbol{\theta}(\boldsymbol{c}))=W_n(\cdot|\boldsymbol{\theta}(\mathbf{0}))}\ \mathrm{d}\mu_n\end{aligned}$$

where W_n, μ_n and $\boldsymbol{\theta}$ are the natural extensions to multiple uses of the channel of W, μ and θ. Note that, under this decoding rule, $\mathrm{P}(\mathbf{0} \to \boldsymbol{c})$ depends only on the type $\boldsymbol{w}_{\mathrm{T}}(\boldsymbol{c})$, and given a type $\boldsymbol{w}$ we will use the notation $Q(\boldsymbol{w})$ to denote $\mathrm{P}(\mathbf{0} \to \boldsymbol{c})$ for any $\boldsymbol{c}$ with $\boldsymbol{w}_{\mathrm{T}}(\boldsymbol{c}) = \boldsymbol{w}$.

The well-known Bhattacharyya bound is the following upper bound for pairwise equivocation probability:

$$\begin{aligned}\mathrm{P}(\mathbf{0} \to \boldsymbol{c}) &\leq \int_{\mathcal{Y}^n} W_n(\cdot|\boldsymbol{\theta}(\mathbf{0}))\mathbb{1}_{W_n(\cdot|\boldsymbol{\theta}(\boldsymbol{c}))\geq W_n(\cdot|\boldsymbol{\theta}(\mathbf{0}))}\,\mathrm{d}\mu_n \\ &\leq \prod_{i=1}^{n}\int_{\mathcal{Y}} \sqrt{W(\cdot|\theta(0))W(\cdot|\theta(c_i))}\,\mathrm{d}\mu \\ &\leq \gamma^{\mathrm{w_H}(\boldsymbol{c})}\end{aligned}$$

where γ is the (worse) Bhattacharyya noise parameter of the channel defined as:

$$\gamma = \max_{g\neq 0}\int_{\mathcal{Y}} \sqrt{W(\cdot|\theta(0))W(\cdot|\theta(g))}\,\mathrm{d}\mu$$

On the other side, a lower bound for pairwise equivocation probability is easily obtained:

$$\begin{aligned}\mathrm{P}(\mathbf{0} \to \boldsymbol{c}) &\geq \int_{\mathcal{Y}^n} W_n(\cdot|\boldsymbol{\theta}(\mathbf{0}))\mathbb{1}_{W_n(\cdot|\boldsymbol{\theta}(\boldsymbol{c}))> W_n(\cdot|\boldsymbol{\theta}(\mathbf{0}))}\,\mathrm{d}\mu_n \\ &\geq \prod_{i=1}^{n}\int_{\mathcal{Y}} W(\cdot|\theta(0))\mathbb{1}_{W(\cdot|\theta(c_i))> W(\cdot|\theta(0))}\,\mathrm{d}\mu \\ &\geq p^{\mathrm{w_H}(\boldsymbol{c})}\end{aligned}$$

where p is the (worse) equivocation probability of the channel, defined as:

$$p = \min_{g\neq 0}\int_{\mathcal{Y}} W(\cdot|\theta(0))\mathbb{1}_{W(\cdot|\theta(c_i))> W(\cdot|\theta(0))}\,\mathrm{d}\mu$$

Let's see what these definitions give in the examples of G-symmetric channels we have presented.

BIOS channels. The names Bhattacharyya parameter and equivocation probability for γ and p are mostly used only in this context, where there is only one non-zero $g \in G$ and so there is no maximization (resp. minimization) in the definition of γ (p).

For BSC with cross-over probability ϵ, if $\mathrm{w_H}(\boldsymbol{c}) = w$ is odd, $\mathrm{P}(\mathbf{0} \to \boldsymbol{c}) = Q(w) = \sum_{r=\lceil w/2\rceil}^{w}\binom{w}{r}\epsilon^r(1-\epsilon)^{w-r}$, while if $\mathrm{w_H}(\boldsymbol{c}) = w$ is even, $\mathrm{P}(\mathbf{0} \to \boldsymbol{c}) =$

$Q(w) = \sum_{r=1+w/2}^{w} \binom{w}{r} \epsilon^r (1-\epsilon)^{w-r} + \frac{1}{2}\binom{w}{w/2} \epsilon^{w/2}(1-\epsilon)^{w/2}$ (the last term because of breaking ties). The Bhattacharyya parameter is $\gamma = 2\sqrt{\epsilon(1-\epsilon)}$, while the equivocation probability is $p = \epsilon$.

For BEC with erasure probability ϵ, the only terms in $\mathrm{P}(\mathbf{0} \to \boldsymbol{c})$ come from breaking ties: if $\mathrm{w_H}(\boldsymbol{c}) = w$, $\mathrm{P}(\mathbf{0} \to \boldsymbol{c}) = Q(w) = \frac{1}{2}\epsilon^w$. Here $\gamma = \epsilon$, while $p = 0$.

For BIAWGN channel, see below.

S-AWGN channels. Here, with $S \subset \mathbb{R}^d$ and codewords of length n,

$$\begin{aligned}\mathrm{P}(\mathbf{0} \to \boldsymbol{c}) &= \int_{\mathbb{R}^{dn}} W_n(\cdot|\boldsymbol{\theta}(\mathbf{0}))\mathbb{1}_{W_n(\cdot|\boldsymbol{\theta}(\boldsymbol{c}))>W_n(\cdot|\boldsymbol{\theta}(\mathbf{0}))}\ \mathrm{d}\mu_n \\ &= \int_{\frac{\|\boldsymbol{\theta}(\boldsymbol{c})-\boldsymbol{\theta}(\mathbf{0})\|}{2}}^{+\infty} \frac{1}{2\pi\sigma^2} e^{-y^2/(2\sigma^2)} \mathrm{d}y \\ &= \frac{1}{2}\,\mathrm{erfc}\left(\frac{\|\boldsymbol{\theta}(\boldsymbol{c}) - \boldsymbol{\theta}(\mathbf{0})\|}{2\sqrt{2\sigma^2}}\right)\end{aligned}$$

where $\mathrm{erfc}(x) = \frac{2}{\sqrt{\pi}} \int_x^{+\infty} e^{-t^2} \mathrm{d}t$ and $\|\cdot\|$ denotes Euclidean norm.

It is well-known that all points of a geometrically uniform constellation lie on a sphere and it is usually assumed that constellations have barycenter in the origin, so the radius of the sphere, squared, is the signal energy per transmitted symbol E_s: this remark allows to find explicit dependence of $\mathrm{P}(\mathbf{0} \to \boldsymbol{c})$ on the re-scaled Euclidean distance and on the SNR:

$$\mathrm{P}(\mathbf{0} \to \boldsymbol{c}) = \frac{1}{2}\,\mathrm{erfc}\left(\frac{\|\boldsymbol{\theta}(\boldsymbol{c}) - \boldsymbol{\theta}(\mathbf{0})\|}{2\sqrt{E_s}}\sqrt{\frac{E_s}{N_0}}\right)$$

where E_s/N_0 is the signal-to-noise ratio per transmitted symbol.

You can compute

$$\gamma = \max_{g \neq 0} e^{-\|\theta(g)-\theta(0)\|^2/(8\sigma^2)} = (e^{-E_s/(4N_0)})^{\min_{g\neq 0}\|\theta(g)-\theta(0)\|^2/E_s}$$

Often, γ is defined in a slightly different way, as $\gamma = e^{-E_s/(4N_0)}$, in order to underline the role of the re-scaled squared Euclidean weights in the exponent.

Finally,

$$p = \min_{g\neq 0} \frac{1}{2}\,\mathrm{erfc}\left(\frac{\|\theta(g) - \theta(0)\|}{2\sqrt{2\sigma^2}}\right) = \frac{1}{2}\,\mathrm{erfc}\left(\frac{\max_{g\neq 0}\|\theta(g) - \theta(0)\|}{\sqrt{E_s}}\sqrt{\frac{E_s}{4N_0}}\right)$$

m-ary symmetric channels. A symmetric channel with alphabet size m and mistake probability ϵ has

$$\mathrm{P}(\mathbf{0} \to \boldsymbol{c}) = \sum_{s=1}^{w} \tbinom{w}{s} [(1 - \tfrac{1}{m-1})\epsilon]^{w-s} \sum_{r=\lfloor s/2 \rfloor + 1}^{s} \tbinom{s}{r} \left(\tfrac{\epsilon}{m-1}\right)^{r} (1-\epsilon)^{s-r} + \tfrac{1}{2} \sum_{s=0}^{\lfloor w/2 \rfloor} \tbinom{w}{2s} [(1 - \tfrac{1}{m-1})\epsilon]^{w-2s} \tbinom{2s}{s} \left(\tfrac{\epsilon(1-\epsilon)}{m-1}\right)^{s}$$

In this case, $\gamma = \sqrt{\frac{\epsilon}{m-1}} \left(2\sqrt{1-\epsilon} + (m-2)\sqrt{\frac{\epsilon}{m-1}}\right)$ and $p = \frac{\epsilon}{m-1}$.

2.2.3 Block codes over Abelian groups

We fix an Abelian group Γ and we consider transmission on a memoryless Γ-symmetric channel. Given another Abelian group U, we define a block encoder of rate k/n, over Γ with inputs in U, to be any injective group homomorphism $\phi : U^k \to \Gamma^n$; we define the corresponding code to be the image of the encoder.

We let ξ to be a r.v. uniformly distributed on U^k (ξ is the word to be sent) and independent from the channel noise. We let $\hat{\xi}$ to be the ML estimate of ξ from the received word y. In this setting, we can clearly define the word error probability of our code in the usual way:

$$P_w(e|\boldsymbol{u}) = \mathbb{P}(\hat{\xi} \neq \boldsymbol{u} | \xi = \boldsymbol{u})$$

and

$$P_w(e) = \mathbb{P}(\hat{\xi} \neq \xi) = \frac{1}{|U|^k} \sum_{\boldsymbol{u} \in U^k} P_w(e|\boldsymbol{u})\,.$$

Our assumptions ensure that the Uniform Error Property holds, i.e. the word error probability does not depend on which word has been sent and, in particular, we can assume that the all-zero word has been sent: $P_w(e) = P_w(e|\mathbf{0})$

Another interesting property of a code (or, more precisely, of an encoder) is its bit error rate. In our abstract setting, it is more convenient to consider a symbol error rate, where symbols can be, for example, the elements of U or, as we will see, also something 'smaller'. We propose the following definition.

Given a distance d compatible with U and such that $\mathrm{d}(u,0) \neq 0$ for all $u \neq 0$, we define a *symbol error rate* with respect to d as

$$P_s(e|\boldsymbol{u}) = \sum_{\hat{\boldsymbol{u}} \in U^k} \frac{\mathrm{d}(\hat{\boldsymbol{u}},\boldsymbol{u})}{k\rho_U} \mathbb{P}(\hat{\xi} = \hat{\boldsymbol{u}} | \xi = \boldsymbol{u})$$

where ρ_U is the diameter of U with respect to d. Moreover, we put

$$P_s(e) = \frac{1}{|U|^k} \sum_{\boldsymbol{u} \in U^k} P_s(e|\boldsymbol{u})\,.$$

The compatibility of the distance with U, together with the previous assumptions, ensures that also for $P_s(e)$ the Uniform Error Property holds true:

$$P_s(e) = P_s(e|\mathbf{0}) = \sum_{\hat{\boldsymbol{u}} \in U^k} \frac{\boldsymbol{w}(\hat{\boldsymbol{u}})}{k\rho_U} \mathbb{P}(\hat{\xi} = \hat{\boldsymbol{u}} | \xi = \mathbf{0}) \,,$$

where $\boldsymbol{w}$ is the weight associated with the distance d. In this case, $\rho_U = \max_{u \in U} \boldsymbol{w}(u)$, and we have the inequality

$$P_s(e) \geq \frac{1}{N} \frac{\min\limits_{u \in U, u \neq 0} \boldsymbol{w}(u)}{\max\limits_{u \in U} \boldsymbol{w}(u)} P_w(e)$$

When d and $\boldsymbol{w}$ are Hamming distance and weight respectively, the above definition simply gives the usual Symbol Error Rate, where symbols are elements in U, and if $U = \mathbb{Z}_2$ this is the classical Bit Error Rate. When $U = \mathbb{Z}_2^a$, in addition to the Symbol Error Rate, we can find also the Bit Error Rate taking as distance the number of different bits (Hamming weight in $(\mathbb{Z}_2^a)^k$ identified with $\mathbb{Z}_2^{ak}$) instead of the number of different symbols.

2.3 Convolutional encoders over Abelian groups

In this section we will recall some basic facts of the theory of convolutional codes over Abelian groups which will be needed for the sequel. Further details can be found in [40, 25, 26, 27] and the reference therein.

2.3.1 State maps and error events

Let U and Y be two Abelian groups. Consider the spaces of sequences $U^{\mathbb{N}}$ and $Y^{\mathbb{N}}$, respectively, both equipped with the componentwise group structure. Convolutional codes will be for us homomorphic maps $\phi : U^{\mathbb{N}} \to Y^{\mathbb{N}}$ satisfying certain properties which are introduced below. In coding theory the only maps between sequence spaces which are really relevant are those which admit a realization through finite state maps.

A *(homomorphic) state map* η from $U^{\mathbb{N}}$ to $Y^{\mathbb{N}}$ consists in another Abelian group X and in four homomorphisms

$$\begin{aligned} F &: X \to X \,, & L &: U \to X \\ R &: X \to Y \,, & S &: U \to Y \end{aligned}$$

X is called the *state space* of the state map and if X is finite, then the state map is said to be a finite state map. A state map is formally denoted by the quadruple

$\eta = (F,L,R,S)$. A finite state map can be pictorially described by a trellis, in the usual way: at each time step, we draw vertices corresponding to the elements of X, then we draw an edge from vertex x at time t to vertex x' at time $t+1$, with input tag u and output label y if and only if $x' = Fx + Lu$ and $y = Rx + Su$.

Given a homomorphic state map η and a state $x \in X$, we can define a map $\eta_x : U^{\mathbb{N}} \to Y^{\mathbb{N}}$ mapping $\mathbf{u} \in U^{\mathbb{N}}$ into $\mathbf{y} = \eta_x(\mathbf{u})$ computed recursively starting from the initial condition $x_0 = x$, as follows:

$$\begin{cases} x_{t+1} &= Fx_t + Lu_t \\ y_t &= Rx_t + Su_t \end{cases} \qquad \forall t \in \mathbb{N}. \tag{2.1}$$

Explicitly, we can write

$$y_t = RF^t x + R\sum_{j=1}^{t} F^j L u_{t-j} + Su_t\,. \tag{2.2}$$

Notice that η_0 is a homomorphism.

A homomorphic map $\phi : U^{\mathbb{N}} \to Y^{\mathbb{N}}$ is said to be a *convolutional encoder* if there exists a homomorphic finite state map $\eta = (F,L,R,S)$ such that $\phi = \eta_0$. In this case η is said to be a *state space realization* of ϕ. Given a convolutional encoder $\phi : U^{\mathbb{N}} \to Y^{\mathbb{N}}$, there may exist many homomorphic finite state maps realizing ϕ. A state map η is said to be a minimal realization of ϕ if it has the state space with the minimal number of states among the possible realizations of ϕ. An important consequence of minimality ([20, pag. 48] or [42, pag. 192]) are the following properties:

- **Observability:** Let $\mathbf{u}',\mathbf{u}'' \in U^{\mathbb{N}}$ and $\mathbf{x}',\mathbf{x}'' \in X^{\mathbb{N}}$ be such that both pairs $(\mathbf{u}',\mathbf{x}')$ and $(\mathbf{u}',\mathbf{x}')$ satisfy the first relation of (2.1). Let $\mathbf{y}',\mathbf{y}''$ be the corresponding output sequences. Then, if ${\mathbf{u}'}_t = {\mathbf{u}''}_t$ and ${\mathbf{y}'}_t = {\mathbf{y}''}_t$ for all $t = 0,\ldots,|X|-1$, necessarily, it must hold ${\mathbf{x}'}_0 = {\mathbf{x}''}_0$.
- **Reachability:** For any $x, x' \in X$ there exist $t \leq |X|-1$, $\mathbf{u} \in U^{\mathbb{N}}$ and $\mathbf{x} \in X^{\mathbb{N}}$ satisfying the first relation in (2.1) such that $\mathbf{x}_0 = x$ and $\mathbf{x}_t = x'$. The smallest t for which this condition holds for any $x, x' \in X$ is called the *reachability index* of η and denoted by ν.

From now on, whenever we are considering a convolutional encoder $\phi : U^{\mathbb{N}} \to Y^{\mathbb{N}}$, we will assume that an underlying minimal state space representation η has been fixed once and for all: in particular, to any given $\boldsymbol{u} \in U^{\mathbb{N}}$, and initial state x we can unambiguously associate a state sequence $\boldsymbol{x} \in X^{\mathbb{N}}$. Whenever the initial state x is not explicitly mentioned, we assume that $x = 0$. Notice moreover that x_t only depends on $\boldsymbol{u}$ up to time $t-1$.

We now define the key concept of error event.

Definition 2.3 Let $\boldsymbol{u} \in U^{\mathbb{N}}$ be an input sequence with associated state sequence $\boldsymbol{x}$. $\boldsymbol{u}$ is said to be an *input error event* for ϕ if there exist $t_1 \leq t_2$ such that

(i) $u_t = 0$ for all $t < t_1$ and $t > t_2$.

(ii) $x_t = 0$ for all $t \leq t_1$ and $t > t_2$.

(iii) $x_t \neq 0$ for all $t \in]t_1,t_2]$.

The corresponding codeword $\boldsymbol{y} = \phi(\boldsymbol{u})$ is said to be an *error event*. We call $[t_1,t_2]$ the *active window* and $t_2 - t_1$ the *length* of the (input) error event and we denote it by $l(\boldsymbol{u})$ or by $l(\boldsymbol{y})$.

The following property shows that the length of an error event cannot grow unbounded. We omit the proof since it is a straightforward generalization of Lemma 20 in [23] (binary case) using the observability property of the minimal realization.

Proposition 2.1 Given a convolutional encoder $\phi : U^{\mathbb{N}} \to Y^{\mathbb{N}}$, there exists a constant $L > 0$ such that any error event $\boldsymbol{u}$ has length $l(\boldsymbol{u}) \leq L\,(w_H(\boldsymbol{u}) + w_H(\phi(\boldsymbol{u}))$ □

The *support* of a sequence $\boldsymbol{u} \in U^{\mathbb{N}}$ is defined by

$$\mathrm{supp}(\boldsymbol{u}) = \{t \in \mathbb{N} \ : \ u_t \neq 0\}\,.$$

$\boldsymbol{u}$ is said to have *finite support* if its support has finite cardinality. Notice that the cardinality of the support of a sequence coincide with the Hamming weight.

2.3.2 Laurent series formalism

In many situations the description of a convolutional encoder through a state representation or the corresponding trellis is sufficient and quite appropriate. As in the classical binary case there are also more algebraic but equivalent ways to describe convolutional codes which, on the other hand, turn out to be quite useful in investigating concepts like recursiveness, non-catastrophicity etc. This is what we are going to do next.

Given a group U, we consider the group of *Laurent series*

$$U((D)) = \left\{ \sum u_k D^k \ : \ u_k \in U \,, \exists\ k_0 \in \mathbb{Z}\ u_k = 0\ \forall k < k_0 \right\}\,.$$

Inside $U((D))$ there are two relevant subgroups: the polynomials $U[D]$ and the usual formal power series $U[[D]]$.

Relation (2.2), for $x = 0$, can be interpreted as a multiplicative operator (product being defined in the Cauchy style) from $U((D))$ to $G((D))$ with the multiplicative symbol given by

$$\phi(D) = \sum_{j=1}^{\infty} (RF^j L) D^j + S \in \mathrm{Hom}(U,Y)[[D]] \,. \tag{2.3}$$

$\phi(D)$ is called the *transfer function* associated with ϕ. Conversely, given a generic $\phi(D) \in \mathrm{Hom}(U,Y)[[D]]$, we can ask if it is the transfer function of a convolutional encoder. The answer is that this is true if and only if $\phi(D)$ is rational. Rationality is defined similarly to the field case. Consider the ring $\mathbb{Z}((D))$ of Laurent series with coefficients in $\mathbb{Z}$. The invertible elements in $\mathbb{Z}((D))$ are those Laurent series whose trailing coefficient is equal to 1 or -1: we denote this subset with the symbol $\mathbb{Z}((D))^*$. Given any Abelian group U, $U((D))$ is naturally a $\mathbb{Z}((D))$–module. We define the submodule of *rational elements* of $U((D))$ as

$$U(D) = \{u(D) \in U((D)) \ : \ \exists p(D) \in \mathbb{Z}[D] \cap \mathbb{Z}((D))^* \,, \, p(D)u(D) \in U[D]\} \,.$$

Notice that rational Laurent series can always be represented in the usual fraction style

$$u(D) = \frac{1}{p(D)} v(D)$$

for some suitable polynomials $p(D) \in \mathbb{Z}[D] \cap \mathbb{Z}((D))^*$ and $v(D) \in U[D]$. The assumption on $p(D)$ is exactly to make sure that $1/p(D)$ is a meaningful element of $\mathbb{Z}((D))$. It can be proven that $\phi(D) = \sum_{k=0}^{\infty} \phi_k D^k \in \mathrm{Hom}(U,Y)[[D]]$ is the transfer function of a convolutional encoder if and only if it is rational (see Proposition 5.2 in [25]). Rationality has a useful characterization at the level of the underlying sequence ϕ_k: it is equivalent to the fact that ϕ_k is periodic for sufficiently large k. A special type of convolutional encoders are the polynomial ones, namely those for which $\phi(D) \in \mathrm{Hom}(U,Y)[D]$.

In the sequel we will often 'confuse' the group sequence $U^{\mathbb{N}}$ with the formal power series $U[[D]]$ through the one-to-one correspondence

$$(u_t)_{t \in \mathbb{N}} \leftrightarrow \sum_t u_t D^t \,.$$

In particular $u_0 D^{t_0}$ will often be used to denote the sequence $\boldsymbol{u}$ which is equal to u_0 at time t_0 and equal to 0 otherwise. Notice that finite support sequences are in this way represented by polynomials in D and polynomial encoders transform polynomials into polynomials.

2.3.3 Properties of convolutional encoders

In this section we describe how some classical properties can be generalized to our setting; we will need them in analyzing our concatenated schemes. Some further properties, specific for the case when the input and output groups are free $\mathbb{Z}_m$–modules, will be given in the Appendix 2.4.

Non-catastrophicity

The classical definition of non-catastrophic encoders is the following.

Definition 2.4 A convolutional encoder $\phi : U^{\mathbb{N}} \to V^{\mathbb{N}}$ is *non-catastrophic* if, for all $\boldsymbol{u} \in U^{\mathbb{N}}$

$$\mathrm{w_H}(\phi(\boldsymbol{u})) < \infty \;\Rightarrow\; \mathrm{w_H}(\boldsymbol{u}) < \infty$$

An useful remark is that systematic encoders are surely non-catastrophic. Also, non-catastrophic encoders have the following nice characterization (direct consequence of [27, Coroll. 1, p. 41])

Proposition 2.2 Let $\phi : U^{\mathbb{N}} \to V^{\mathbb{N}}$ be a convolutional encoder. The following conditions are equivalent:

1. ϕ is non-catastrophic;
2. ϕ admits a polynomial left inverse.
3. there exists a constant $\zeta > 0$ such that, for all $\boldsymbol{u} \in U^{\mathbb{N}}$

$$\mathrm{w_H}(\boldsymbol{u}) \leq \zeta\, \mathrm{w_H}(\phi(\boldsymbol{u}))\,.$$

□

Notice that condition 2. gives a practical tool for testing if an encoder is non-catastrophic, and it also shows that non-catastrophicity is a property stronger than injectivity. Instead condition 3. is a sort of continuity reformulation.

Recursiveness

Binary convolutional encoders are defined to be recursive when no input word with Hamming weight one can give a finite-weight output; this property can be easily generalized to our setting.

Definition 2.5 Given a weight $\boldsymbol{w} : U \to \mathbb{N}^{\rho}$, a convolutional encoder $\phi : U^{\mathbb{N}} \to Y^{\mathbb{N}}$ is $\boldsymbol{w}$–recursive if, for all $\boldsymbol{u} \in U^{\mathbb{N}}$,

$$|\boldsymbol{w}(\boldsymbol{u})| = 1 \;\Rightarrow\; \mathrm{w_H}(\phi(\boldsymbol{u})) = +\infty\,.$$

When $\boldsymbol{w}$ is the Hamming weight, this is the usual definition of recursiveness.

See Appendix 2.4 for a characterization of recursive encoders on free $\mathbb{Z}_m$–modules which allows to easily test for recursiveness.

Small input-weight codewords

All convolutional encoders, including the recursive ones, admit non-zero finite support input sequences whose image also has finite support. This fact is obvious from the rationality property. Indeed, if the transfer function $\phi(D)$ is of type

$$\phi(D) = \frac{1}{p(D)}\phi'(D)$$

where $p(d) \in \mathbb{Z}((D))^* \cap \mathbb{Z}[D]$ and $\phi'(D) \in \mathrm{Hom}(U,Y)[D]$, we can observe that any polynomial input of type $u(D) = p(D)v(D)$ for some $v(D) \in U[D]$ is transformed into another polynomial $\phi(D)u(D) = \phi'(D)v(D)$.

We now present a sharper result which shows how to construct input sequences with support of cardinality 2, whose image has finite support: this will be useful later on.

Proposition 2.3 Let $\phi : U^{\mathbb{N}} \to Y^{\mathbb{N}}$ be e a convolutional encoder and let $u_1, \ldots, u_r \in U$ be such that $\sum u_i = 0$. We can find time instants $t_1, \ldots t_r$ such that given $\boldsymbol{u} = \sum u_j D^{t_j}$ we have that $\phi(\boldsymbol{u})$ has finite support. □

Proof:

Consider the transfer function $\phi(D) = \sum_k \phi_k D^k$. By rationality we know that there exists $k_0 \in \mathbb{N}$ and $T \in \mathbb{N}$ such that $\phi_k = \phi_{k+T}$ for every $k \geq k_0$. Consider now the input sequence $\boldsymbol{u} = \sum_j u_j D^{(j-1)T}$. We have that

$$\phi(\boldsymbol{u})_t = \sum_{j=1}^{r} \phi_{t-(j-1)T} u_j$$

Notice that if we choose $t \geq k_0 + (r-1)T$, we easily obtain that $\phi_{t-(j-1)T} = \phi_t$ for every j so that, $\phi(\boldsymbol{u})_t = 0$. This proves the result. ■

From the above result we obtain as an immediate corollary the following property, well-known at least for the binary case.

Proposition 2.4 Given a recursive convolutional encoder $\phi : U^{\mathbb{N}} \to Y^{\mathbb{N}}$, there exists $\delta \in \mathbb{N}$ such that, for any $u \in U$ the input sequence $\boldsymbol{u} = u - uD^{\delta}$ is an error event. ■

Free distance

In the classical analysis by Benedetto et al. [3], an essential design parameter is the free distance of the outer encoder. When the concatenating group is not the group of all permutations (the classical uniform interleaver), we have to consider a slightly different parameter: instead of taking the minimum Hamming weight among non-zero outer codewords, we minimize some other proper weight.

Definition 2.6 Given a convolutional encoder $\phi : U^{\mathbb{N}} \to Y^{\mathbb{N}}$ and a weight $\boldsymbol{w} : Y^{\mathbb{N}} \to \mathbb{N}^{\rho}$, we define the $\boldsymbol{w}$–free distance of ϕ to be

$$d_f(\phi,\boldsymbol{w}) = \min\{|\boldsymbol{w}(\boldsymbol{c})| : \boldsymbol{c} = \phi(\boldsymbol{u}),\boldsymbol{u} \in U^{\mathbb{N}},\boldsymbol{u} \neq \boldsymbol{0}\}$$

The classical free distance is the $\mathrm{w_H}$–free distance.

2.3.4 Terminated convolutional encoders

Suppose $\phi : U^{\mathbb{N}} \to Y^{\mathbb{N}}$ is a convolutional encoder with minimal state space X. We now define the terminated block codes associated with ϕ as follows.

Fix $N \in \mathbb{N}^*$. Given a vector $\boldsymbol{u} = (u_0,\dots,u_{N-1}) \in U^N$, let x_N be the corresponding state at time N. Because of the reachability condition it is possible to find input elements $\tilde{u}_N,\dots,\tilde{u}_{N+\nu-1}$ such that the state at time $N+\nu-1$ is equal to 0. This input string may not be unique and we assume we have fixed a specific one as a function of the terminal state x_N we had reached in such a way that the mapping

$$x_N \mapsto (\tilde{u}_N,\dots,\tilde{u}_{N+\nu-1})$$

is a homomorphism. It is a straightforward algebraic verification that this is always indeed possible. Given $\boldsymbol{u} = (u_0,\dots,u_{N-1}) \in U^N$ we now consider the associated input sequence

$$\tilde{\boldsymbol{u}} = (u_0,\dots,u_{N-1},\tilde{u}_N,\dots,\tilde{u}_{N+\nu-1},0,0,\dots)$$

We then define the *N-terminated block encoder* as

$$\phi^N : U^N \to Y^{N+\nu}\,, \quad \phi^N(\boldsymbol{u}) = \phi(\tilde{\boldsymbol{u}})|_{[0,N+\nu-1]}\,.$$

For the assumptions made, ϕ^N is also a homomorphism. $\mathcal{C}^N = \operatorname{Im}\phi^N$ is called the *N-block code* associated to ϕ^N.

An input vector $\boldsymbol{u} \in U^N$ is an input error event for ϕ^N if $\tilde{\boldsymbol{u}} \in U^{\mathbb{N}}$ is an input error event for ϕ. In this case $\boldsymbol{c} = \phi^N(\boldsymbol{u})$ is called an error event for ϕ^N. Suppose the active window of $\tilde{\boldsymbol{u}}$ is equal to $[t_1,t_2]$. Then, the (input) error event is said to be *regular* if $t_2 \leq N$, otherwise it is called *terminated*. For a terminated error event, we call $N - t_1$ its length.

Notice that any codeword $\boldsymbol{c} \in \mathcal{C}^N$ can be written as $\boldsymbol{c} = \sum_{j=1}^{n+1} \boldsymbol{c}_j$ where $\boldsymbol{c}_j$ are regular error events for $j = 1, \ldots, n$ and $\boldsymbol{c}_{n+1}$ is either zero or a terminating error event and the active windows of all these events are disjoint. We will use the notation $n(\boldsymbol{c})$ to denote the number of regular error events in the above decomposition of $\boldsymbol{c}$. Also notice that the above decomposition is unique, up to a permutation of the regular error events.

Some codewords have a decomposition in error events which is the same up to shifts of their error events, and for this reason share many important properties. More formally, we propose the following definition:

- two error events $\boldsymbol{c} = \phi^N(\boldsymbol{u})$ and $\boldsymbol{c}' = \phi^{N'}(\boldsymbol{u}')$ (notice that possibly $N \neq N'$) are said to be shift equivalent if the corresponding extended inputs $\tilde{\boldsymbol{u}}, \tilde{\boldsymbol{u}}' \in U^{\mathbb{N}}$ differ only by a shift.
- two codewords $\boldsymbol{c} = \phi^N(\boldsymbol{u})$ and $\boldsymbol{c}' = \phi^{N'}(\boldsymbol{u}')$ are said to be shift equivalent if there exist error event decompositions $\boldsymbol{c} = \sum_{j=1}^{n+1} \boldsymbol{c}_j$ and $\boldsymbol{c}' = \sum_{i=1}^{n'+1} \boldsymbol{c}'_i$ such that $\boldsymbol{c}_i$ and $\boldsymbol{c}'_i$ are shift equivalent for all i.

Notice that, given two shift equivalent codewords $\boldsymbol{c}$ and $\boldsymbol{c}'$, clearly $n(\boldsymbol{c}) = n(\boldsymbol{c}')$ and moreover, given a weight $\boldsymbol{w}$ on the alphabet Y, $\boldsymbol{w}(\boldsymbol{c}) = \boldsymbol{w}(\boldsymbol{c}')$.

Remark 2.1 Now we want to underline a property which is somehow similar to an inclusion of $\mathcal{C}^N$ in $\mathcal{C}^{N'}$ for $N \leq N'$ (while strictly speaking an inclusion cannot occur, as the two codes are subsets of different spaces). If $N \leq N'$, for all $\boldsymbol{c} \in \mathcal{C}^N$ we can construct $\boldsymbol{c}' \in \mathcal{C}^{N'}$ such that $\boldsymbol{c}$ and $\boldsymbol{c}'$ are shift equivalent, by properly adding zeros. ■

2.3.5 Enumerating functions and growth estimates

A fundamental concept for all encoders is the so called weight enumerating function, since it is well known to play a basic role in all performance evaluations. While in the binary case, there is only one possible weight to be considered, namely the Hamming one, in our setting many choices are possible and we will need to consider different possibilities in later sections. We start defining the basic one based on the Hamming weights in the input and output groups.

Definition 2.7 Given a convolutional encoder $\phi : U^{\mathbb{N}} \to Y^{\mathbb{N}}$, consider its associated N-terminated block encoder $\phi^N : U^N \to Y^{N+\nu}$. Define its input/output support enumerating coefficients as:

$$\Lambda^N_{w,d,n} = |\{\boldsymbol{u} \in U^N : \mathrm{w_H}(\boldsymbol{u}) = w, \mathrm{w_H}(\phi^N(\boldsymbol{u})) = d, n(\phi^N(\boldsymbol{u})) = n\}|$$

□

In some cases, we will need to replace the Hamming weight with other possible weights in the input and in the output. We will use the notation $A^N_{\boldsymbol{w},\boldsymbol{d},n}$ to denote enumerating coefficients relative to some specified input weight $\boldsymbol{w}$ and output weight $\boldsymbol{d}$.

The following proposition gives a growth estimation for input/output support enumerating coefficients: this will allow us to have general bounds (even if quite loose) on all the different weight enumerators. We omit the proof, which is a straightforward generalization of Proposition 10 in [23].

Proposition 2.5 There exists two positive constants a and b such that

$$\Lambda^N_{w,d,n} \leq \binom{N+n}{n} a^w b^d$$

□

2.4 Properties of free $\mathbb{Z}_m$ convolutional encoders

In this Section, we consider convolutional encoders $\phi : \mathbb{Z}_m^{k\mathbb{N}} \to \mathbb{Z}_m^{n\mathbb{N}}$ which can be represented as matrices $\phi \in \mathbb{Z}_m^{k\times n}(D) \simeq \mathbb{Z}_m(D)^{k\times n}$. We will call them free $\mathbb{Z}_m$ convolutional encoders. They are the most straightforward generalization of classical binary convolutional encoders, and they have some interesting properties. Let us start with a simple algebraic remark: we know that ϕ can be represented as $\phi = p(D)^{-1}q(D)$ for some $p(D) \in \mathbb{Z}[D] \cap \mathbb{Z}((D))^*$ and $q(D) \in \mathbb{Z}_m[D]^{k\times n}$. Since all the algebraic structures involved are also $\mathbb{Z}_m$-modules, it turns out that we can as well assume that $p(D) \in \mathbb{Z}_m[D] \cap \mathbb{Z}_m((D))^*$ which in practice means that $p(D)$ has all coefficients in $\mathbb{Z}_m$ and the trailing coefficient is in $\mathbb{Z}_m^*$.

In Sect. 2.3.3, we gave a general definition of recursiveness. In the binary case (for simplicity consider scalar input, i.e. $\phi : \mathbb{Z}_2^{\mathbb{N}} \to \mathbb{Z}_2^{n\mathbb{N}}$), there are well-known characterizations of $\mathrm{w_H}$–recursive encoders: ϕ is recursive when its shift-register state representation has a feedback, or equivalently if $\phi = \frac{1}{q(D)}[p_1(D),\ldots,p_n(D)]$, with $\gcd\{q,p_1,\ldots,p_n\} = 1$ has non-trivial denominator, i.e. $q(D) \neq D^h$. This latter characterization allows to check very easily if an encoder is recursive and we will now generalize it to recursiveness of free $\mathbb{Z}_m$ encoders with respect to Hamming or equivalently to type weight in $\mathbb{Z}_m$ (not the Hamming weight in $\mathbb{Z}_m^k$).

First of all, without loss of generality we can restrict ourselves to considering scalar encoders $\phi : \mathbb{Z}_m^{\mathbb{N}} \to \mathbb{Z}_m^{\mathbb{N}}$: if not so, notice that $\phi : \mathbb{Z}_m^{k\mathbb{N}} \to \mathbb{Z}_m^{n\mathbb{N}}$ is $\boldsymbol{w}$–recursive ($\boldsymbol{w}$ being the Hamming or the type weight in $\mathbb{Z}_m$) if and only if each column of its matrix has at least one entry which is a scalar $\boldsymbol{w}$–recursive encoder.

Then, if m is a prime (so that $\mathbb{Z}_m$ is a field), $\phi = p(D)/q(D)$ with $p(D),q(D) \in \mathbb{Z}_m[D]$ and $\gcd(p,q) = 1$ is $\boldsymbol{w}$–recursive if and only if $q(D) \neq D^t$: as in the binary

case, we can identify recursive encoders at a glance, just looking at their denominator.

If m is not a prime, let $m = p_1^{\alpha_1} \dots p_l^{\alpha_l}$ be its prime factors decomposition and let $\phi_i : \mathbb{Z}_{p_i}^{\mathbb{N}} \to \mathbb{Z}_{p_i}^{\mathbb{N}}$ be obtained by taking the restriction of ϕ to inputs in $\frac{m}{p_i}\mathbb{Z}_m$ and then identifying $\frac{m}{p_i}\mathbb{Z}_m$ with $\mathbb{Z}_{p_i}$ through the natural fields isomorphism mapping.

Proposition 2.6 ϕ is $\boldsymbol{w}$–recursive if and only if $\phi_1, \dots, \phi_l$ are $\boldsymbol{w}$–recursive. □

Proof: The first implication is trivial. Conversely, knowing that $\phi_1, \dots, \phi_l$ are recursive, we want to show that $\mathrm{w_H}(\phi(D)u) = +\infty$ for any $u \in \mathbb{Z}_m \setminus \{0\}$. Since $u < m$, there exists $i \in \{1, \dots, l\}$ and $r \in \mathbb{N}$ such that $p_i^r | u$, $p_i^{r+1} \nmid u$, and $p_i^{r+1} | m$. Consider $\tilde{u} = (p_i^{-1}m)(p_i^{-r}u) = (p_i^{-r-1}m)u$. Clearly, $\tilde{u} \neq 0$ and, by the assumptions made, $\mathrm{w_H}(\phi(D)\tilde{u}) = \infty$. This clearly implies that also $\mathrm{w_H}(\phi(D)u) = \infty$. ■

The characterization given by Prop. 2.6 is helpful because the encoders $\phi_1, \dots, \phi_l$ can be obtained very easily from ϕ: if you write $\phi(D) = p(D)/q(D)$ with $p(D), q(D) \in \mathbb{Z}_m[D]$, you have $\phi_j(D) = \tilde{p}(D)/\tilde{q}(D)$ where $\tilde{p}(D), \tilde{q}(D)$ are polynomials in $\mathbb{Z}_{p_j}$ obtained multiplying each coefficient of $p(D)$ (resp. $q(D)$) by m/p_j (modulo m) and then identifying the corresponding element in $\mathbb{Z}_{p_j}$.

For example, the encoder $\phi : \mathbb{Z}_8^{\mathbb{N}} \to \mathbb{Z}_8^{\mathbb{N}}$ defined by $\phi(D) = \frac{1+3D^2}{1+7D}$ is not recursive. You cannot tell it simply looking at the denominator, which is non-trivial. You can see it using the definition given in Sect. 2.3.3: notice that $\phi(D) = (1+3D^2)\sum_{t\geq 0} D^t$ and then input $u(D) = 2$ produces output $2\phi(D) = 2 + 2D \in \mathbb{Z}_8[D]$. You can also check the recursiveness of ϕ using Prop. 2.6: as $m = 8$ has only one prime divisor $p_1 = 2$, you need to check only one encoder $\phi_1 = \frac{1+D^2}{1+D} = 1 + D$ which clearly isn't recursive.

By the same technique of looking at the encoders $\phi_1, \dots, \phi_l$ defined above, we can obtain a characterization of the free distance of ϕ with respect to Hamming or type weight in $\mathbb{Z}_m$. This characterization is not interesting under a computational point of view, as the computation of the free distance of encoders over fields or rings does not have a different complexity, but it is essential to find tight bounds for the interleaver gain of free $\mathbb{Z}_m$ serial schemes (Prop. 3.12 and Coroll. 3.1).

Proposition 2.7 Let d_f be the $\boldsymbol{w}$–free distance of ϕ and $d_f(\phi_j)$ be the $\boldsymbol{w}$–free distance of ϕ_j, where $\boldsymbol{w}$ is the Hamming or the type weight in $\mathbb{Z}_m$ and $\mathbb{Z}_{p_j}$ respectively. Then:

$$d_f = \min_{j=1,\dots,l} \{d_f(\phi_j)\}$$

Proof: Clearly, for all $j = 1, \dots, l$, $d_f(\phi_j) = d_f(\frac{m}{p_j}\phi) \geq d_f$. Now we will prove that there exists at least one j such that $d_f(\phi_j) = d_f$.

Let $\mathcal{C} = \phi\left(\mathbb{Z}_m^k((D))\right)$ and let $\boldsymbol{x} \in \mathcal{C}$ be a codeword such that $\mathrm{w_H}(\boldsymbol{x}) = d_f$. The key remark is that $\mathrm{w_H}(\boldsymbol{x}) = d_f$ implies that all non-zero symbols (i.e. elements of $\mathbb{Z}_m$) of $\boldsymbol{x}$ have the same annihilator. In fact, $\mathrm{w_H}(\boldsymbol{x}) = d_f$ means that $\forall \boldsymbol{y} \in \mathcal{C} \setminus \{\boldsymbol{0}\}$ $\mathrm{w_H}(\boldsymbol{y}) \geq \mathrm{w_H}(\boldsymbol{x})$, which implies that

$$\nexists a \in \mathbb{Z}_m \text{ such that } 0 < \mathrm{w_H}(a\boldsymbol{x}) < \mathrm{w_H}(\boldsymbol{x})$$

and so, for all $a \in \mathbb{Z}_m$, either $a\boldsymbol{x} = \boldsymbol{0}$ or $\mathrm{w_H}(a\boldsymbol{x}) = d_f$ i.e. $a\boldsymbol{x}_i \neq 0$ for all $\boldsymbol{x}_i \neq 0$.

This remark implies that there exists $d|m$ (possibly $d = 1$) and there exists p_j a prime factor of m such that $\mathrm{w_H}(d\boldsymbol{x}) = \mathrm{w_H}(\boldsymbol{x})$ and $p_j\, d\, \boldsymbol{x} = \boldsymbol{0}$. Now, choosing $\boldsymbol{c} = d\,\boldsymbol{x}$ we have a codeword $\boldsymbol{c} \in \frac{m}{p_j}\mathcal{C}$ such that $\mathrm{w_H}(\boldsymbol{c}) = d_f$, so that we can conclude: $d_f(\phi_j) = d_f(\frac{m}{p_j}\phi) = d_f$. ■

Finally, when proving Prop. 3.12 we need also the following simple lemma, even though it is just a property of $\mathbb{Z}_m$ and not of convolutional codes.

Lemma 2.2 Given $a_1, \ldots, a_m \in \mathbb{Z}_m \backslash \{0\}$, there exist indexes $\{i_1, \ldots, i_n\} \subseteq \{1, \ldots, m\}$ such that $a_{i_1} + \ldots + a_{i_n} = 0 \mod m$. □

Proof: By contradiction, assume that $\sum_{i \in \mathcal{I}} a_i \neq 0 \mod m$ for all non-empty $\mathcal{I} \subseteq \{1, \ldots, m\}$. Then, in particular, $\sum_{i=1}^{n} a_i \neq 0 \mod m$ for all $n = 2, \ldots, m$ and so $a_1 \notin \{-a_2, -a_2 - a_3, \ldots, -\sum_{j=2}^{m} a_i\}$, which, being a set of $m-1$ distinct non-zero elements of $\mathbb{Z}_m$ is $\mathbb{Z}_m \setminus \{0\}$ itself, contradicting $a_1 \in \mathbb{Z}_m \setminus \{0\}$. ■

Chapter 3

Generalized serial turbo ensemble

In this chapter we introduce a wide class of generalized serial turbo schemes, coupling two convolutional encoders over groups through an interleaver respecting the group structure; these codes are designed to be used on symmetric channels, where the group structure of the encoder and the channel is matching. A particularly relevant example is the case when the convolutional codes are modules on $\mathbb{Z}_m$, the interleaver is a permutation and the channel is AWGN with m–PSK input constellation.

We introduce an ensemble of coding schemes, and we study its average ML performance: we obtain the exact asymptotic decay of the average symbol and word error probability when the interleaver length goes to infinity and also the behavior when the SNR goes to infinity. The performance is characterized by two parameters, the interleaver gain μ and the effective distance q^*, which are defined as the solution of an optimization problem and in general jointly depend on both constituent encoders, differently from the binary case. To make clear the meaning of these parameters, we have explicitly computed them in some examples encompassing most of the relevant scenarios.

3.1 Ensemble description

3.1.1 Serial interconnections

We now precisely define the serial interconnected schemes we are going to consider. We start with a Γ-symmetric channel. We also fix the *input* Abelian group U. All encoders we will consider will be driven by words on U and will output symbols in Γ. The interconnection will take place through a third Abelian group, say Y called the *interconnection* group. We now fix two convolutional encoders denoted, respectively, the *outer* and *inner* encoder:

$$\phi_o : U^{\mathbb{N}} \to (Y^r)^{\mathbb{N}}, \quad \phi_i : (Y^s)^{\mathbb{N}} \to (\Gamma^l)^{\mathbb{N}}.$$

Denote by ν_o and ν_i the reachability indices of ϕ_o and ϕ_i respectively, and define the set

$$\mathcal{N} = \{N \in \mathbb{N}^* \ : \ s | r(N + \nu_o)\} .$$

Consider now the terminations, for $N \in \mathcal{N}$:

$$\phi_o^N : U^N \to Y^{r(N+\nu_o)} , \quad \phi_i^N : Y^{sM_N} \to I^{l(M_N+\nu_i)} ,$$

where $sM_N = r(N + \nu_o)$. We now fix, for every $N \in \mathbb{N}^*$, a subgroup $G_N \subseteq \mathrm{Aut}(Y^{r(N+\nu_o)})$. The triple $(\phi_o,\phi_i,(G_N)_{N\in\mathcal{N}})$ is said to be a *serial interconnected ensemble*. The asymptotic rate of the serial interconnected ensemble above is defined by the product

$$R = \frac{\log |U|}{r} \frac{s}{l} \text{ bits per channel use} .$$

To the serial interconnected ensemble above we can associate a random sequence of encoders and codes as follows. Define Π_N to be a r.v. uniformly distributed over G_N and consider the corresponding homomorphic encoder $\Phi^N = \phi_i^N \circ \Pi_N \circ \phi_o^N$ and group code $\mathcal{C}^N = \mathrm{Im}(\Phi^N)$: they are called, respectively, the *random encoder* and the *random code* associated with the given ensemble.

The following picture describes the above construction.

$$\xrightarrow{U^N} \boxed{\phi_o^N} \xrightarrow{Y^{r(N+\nu_o)}} \boxed{\pi_N} \xrightarrow{Y^{sM_N}} \boxed{\phi_i^N} \xrightarrow{I^{l(M_N+\nu_i)}}$$

In the sequel we will denote by $\mathbb{P}$ and $\mathbb{E}$ probability and expected value, respectively, made with respect to the probabilistic space underlying the sequence Π_N. We will also use the notation $\overline{P_w(e)}$ and $\overline{P_s(e)}$, respectively, for the average word and symbol error probabilities.

3.1.2 Regular ensembles

Our aim is to give asymptotic results for $\overline{P_w(e)}$ and $\overline{P_s(e)}$ when $N \to \infty$, keeping fixed the constituent encoders. To do so, we need to make further assumptions on the groups G_N: roughly, we need to enforce some compatibility among the groups as N varies and that the number of invariants of the group action does not grow with N. Following [23] we propose the following definition.

Definition 3.1 The sequence of groups G_N (and the corresponding ensemble) is said to be regular if there exists a weight $\boldsymbol{w}_G : Y \to \mathbb{N}^\rho$ such that, for every N and for all $\boldsymbol{y}, \boldsymbol{z}, \in Y^{r(N+\nu_0)}$, it holds

$$\boldsymbol{w}_G(\boldsymbol{y}) = \boldsymbol{w}_G(\boldsymbol{z}) \Leftrightarrow \quad \exists \sigma \in G_N \ : \sigma \boldsymbol{y} = \boldsymbol{z}$$

$\boldsymbol{w}_G(\boldsymbol{y})$ will be called the invariants weight vector of $\boldsymbol{y} \in Y^{r(N+\nu_0)}$.

Property of regularity simply says that all actions of the groups G_N on the sets $Y^{r(N+\nu_0)}$ can be described through a finite (constant) family of invariants: the ρ components of the weight $\boldsymbol{w}_G$. We will use the notation $Y_{\boldsymbol{h}}^L = \{\boldsymbol{x} \in Y^L : \boldsymbol{w}_G(\boldsymbol{x}) = \boldsymbol{h}\}$. Moreover we denote by $G_N(\boldsymbol{y}, \boldsymbol{z})$ the subset of elements in G_N mapping $\boldsymbol{y}$ to $\boldsymbol{z}$. Using standard results on group actions (the class formula) [36], we can show that

Remark 3.1

$$\frac{|G_N(\boldsymbol{u},\boldsymbol{v})|}{|G_N|} = \begin{cases} 0 & \text{if } \boldsymbol{w}_G(\boldsymbol{u}) \neq \boldsymbol{w}_G(\boldsymbol{v}) \\ 1/|Y_{\boldsymbol{h}}^{r(N+\nu_0)}| & \text{if } \boldsymbol{w}_G(\boldsymbol{u}) = \boldsymbol{w}_G(\boldsymbol{v}) = \boldsymbol{h} \end{cases}$$

■

This technical result will be needed later.

Lemma 3.1 Assume that $\boldsymbol{y},\boldsymbol{z} \in Y^{r(N+\nu_o)}$ are such that for every index $i \in \{0,\ldots,N+\nu_o-1\}$, $\boldsymbol{y}_i \neq \boldsymbol{0}$ yields $\boldsymbol{z}_i = \boldsymbol{0}$. Then, given any $\sigma \in G_N$ and given any i we have that

$$(\sigma\boldsymbol{y})_i \neq \boldsymbol{0} \;\Rightarrow\; (\sigma\boldsymbol{z})_i \neq (\sigma\boldsymbol{y})_i \,.$$

Proof: Notice that

$$|\boldsymbol{w}_G(\sigma\boldsymbol{y} + \sigma(-\boldsymbol{z}))| = |\boldsymbol{w}_G(\boldsymbol{y} - \boldsymbol{z})| = |\boldsymbol{w}_G(\boldsymbol{y})| + |\boldsymbol{w}_G(-\boldsymbol{z})| \,.$$

On the other hand, if $\sigma\boldsymbol{y}$ and $\sigma\boldsymbol{z}$ were equal in an index where they are not equal to $\boldsymbol{0}$, by point 2. of Definition 2.1, we would have

$$|\boldsymbol{w}_G(\sigma\boldsymbol{y} + \sigma(-\boldsymbol{z}))| < |\boldsymbol{w}_G(\sigma\boldsymbol{y})| + |\boldsymbol{w}_G(\sigma(-\boldsymbol{z}))| = |\boldsymbol{w}_G(\boldsymbol{y})| + |\boldsymbol{w}_G(-\boldsymbol{z})| \,.$$

This ends the proof. ■

We now present two fundamental examples of regular actions.

1. *Symbol permutation* In this case we simply take $G_N = S_{r(N+\nu_0)}$ the full symmetric group acting on $Y^{r(N+\nu_0)}$ by standard permutation. In this case the invariant weight is the type weight: $\rho = |Y| - 1$ and $\boldsymbol{w}_G(y) \in \mathbb{N}^\rho$ by $(\boldsymbol{w}_G(y))_a = \mathbb{1}_a(y)$ as a varies in $Y \setminus \{0\}$. Notice that

$$\left|Y_{\boldsymbol{h}}^{r(N+\nu_o)}\right| = \binom{r(N+\nu_o)}{\boldsymbol{h}} \,.$$

2. *Separate channels symbol permutation* Assume $Y = Y_1 \times Y_2$ and assume $G_{N,1}$ and $G_{N,2}$ are sequences of groups acting regularly on $Y_1^{r(N+\nu_0)}$ and $Y_2^{r(N+\nu_0)}$

respectively with invariant weights $\boldsymbol{w}_G^1 : Y_1 \to \mathbb{N}^{\rho_1}$ and $\boldsymbol{w}_G^2 : Y_2 \to \mathbb{N}^{\rho_2}$ respectively. Then, we can consider a regular action given by $G_N = G_{N,1} \times G_{N,2}$ acting componentwise on $Y_1 \times Y_2$. Its invariant weight is given by $\boldsymbol{w}_G : Y_1 \times Y_2 \to \mathbb{N}^{\rho_1+\rho_2}$, $\boldsymbol{w}_G(y_1,y_2) = (\boldsymbol{w}_G^1(y_1),\boldsymbol{w}_G^2(y_2))$. Notice that in this case

$$\left|Y^{r(N+\nu_o)}_{\boldsymbol{h}_1,\boldsymbol{h}_2}\right| = \binom{r(N+\nu_o)}{\boldsymbol{h}_1}\binom{r(N+\nu_o)}{\boldsymbol{h}_2}.$$

3.1.3 Examples of serial ensembles

In the examples below we assume $\Gamma = \mathbb{Z}_m$.

Repeat-Convolute codes We choose $U = Y = \mathbb{Z}_m$ and $\phi_o = \mathrm{Rep}_r : \mathbb{Z}_m^{\mathbb{N}} \to (\mathbb{Z}_m^r)^{\mathbb{N}}$ to be the r-repetition encoder

$$\mathrm{Rep}_r(\boldsymbol{u})_t = (u_t, \dots u_t)\,.$$

We let $\phi_i : (\mathbb{Z}_m^s)^{\mathbb{N}} \to (\mathbb{Z}_m^s)^{\mathbb{N}}$ be a rate-1 non-catastrophic convolutional encoder. Finally we choose for the coupling interleavers the symbol permutation groups $G_N = S_{rN}$. The corresponding invariant weight is thus the type weight $\boldsymbol{w}_{\mathrm{T}} : \mathbb{Z}_m^r \to \mathbb{N}^{m-1}$, where $(\boldsymbol{w}_{\mathrm{T}}(\boldsymbol{y}))_j$ is the number of elements equal to j in the vector $\boldsymbol{y} \in \mathbb{Z}_m^r$. The rate of the scheme is

$$R = \frac{\log m}{r} \text{ bits/ch. use.}$$

For this ensemble, we will need the assumption that ϕ_i is $\boldsymbol{w}$-recursive, which is the same as asking it is $\mathrm{w_H}$-recursive.

Structured LDPC codes We choose $U = \mathbb{Z}_m$, $Y = \mathbb{Z}_m^c \times \mathbb{Z}_m$ and ϕ_o as the systematic encoder

$$\phi_o : \mathbb{Z}_m^{\mathbb{N}} \to (\mathbb{Z}_m^c \times \mathbb{Z}_m)^{\mathbb{N}}, \quad \phi_o(\boldsymbol{u}) = (\mathrm{Rep}_c(\boldsymbol{u}),\boldsymbol{u})\,.$$

Instead ϕ_i is itself the serial interconnection of two encoders. We consider $\mathrm{Sum}_d : (\mathbb{Z}_m^d)^{\mathbb{N}} \to \mathbb{Z}_m^{\mathbb{N}}$ defined by

$$\mathrm{Sum}_d(\boldsymbol{y}) = (y_1 + \cdots + y_d, y_{d+1} + \cdots + y_{2d}, \dots)\,,$$

and a $\mathrm{w_H}$-recursive non-catastrophic rate-1 convolutional encoder $\psi : \mathbb{Z}_m^{\mathbb{N}} \to \mathbb{Z}_m^{\mathbb{N}}$. Finally we take $\phi_i : (\mathbb{Z}_m^d \times \mathbb{Z}_m)^{\mathbb{N}} \to (\mathbb{Z}_m \times \mathbb{Z}_m)^{\mathbb{N}}$ defined by

$$\phi_i(\boldsymbol{y}^1,\boldsymbol{y}^2) = ((\psi \circ \mathrm{Sum}_d)(\boldsymbol{y}^1),\boldsymbol{y}^2)\,.$$

When taking the truncated versions of these encoders, we must make sure to have suitable lengths, so we take: $\phi_o^N : \mathbb{Z}_m^{dN} \to \mathbb{Z}_m^{cdN} \times \mathbb{Z}_m^{dN}$ and $\phi_i^N : \mathbb{Z}_m^{cdN} \times \mathbb{Z}_m^{dN} \to \mathbb{Z}_m^{cN+\nu_\psi} \times \mathbb{Z}_m^{dN} = \Gamma^{(c+d)N+\nu_\psi}$. So, the design rate of the serial encoder $\phi^N = \phi_i^N \circ \Pi_N \circ \phi_o^N$ is $R = \log m \, \frac{d}{c+d}$.

As interconnection group, we chose the separated channels symbol permutation $G_N = S_{cdN} \times S_{dN}$.

This family of codes is a generalization of Repeat-Convolute codes: the additional summator Sum_s is the same as the grouping factor introduced in Irregular Repeat Accumulate codes.

If we construct the parity check matrix for the code $\mathcal{C}^N = \mathrm{Im}(\Phi^N) \subseteq \mathbb{Z}_m^{(c+d)N+\nu_\psi}$, we can see that it is sparse, and it has a structured and a random part, so that we have a structured LDPC ensemble, generalizing staircase LDPC codes. In fact, notice that

$$\begin{aligned}(\boldsymbol{c}^1,\boldsymbol{c}^2) \in \mathcal{C}^N &\Leftrightarrow \boldsymbol{c}^1 = \psi^N \circ \mathrm{Sum}_d \circ \pi_N^1 \circ \mathrm{Rep}_c \circ (\pi_N^2)^{-1}(\boldsymbol{c}^2) \\ &\Leftrightarrow (\psi^N)^{-1}(\boldsymbol{c}^1) = \mathrm{Sum}_d \circ \pi_N^1 \circ \mathrm{Rep}_c \circ (\pi_N^2)^{-1}(c^2)\end{aligned}$$

It is clear that the permutation π_N^2 does not play any essential role: we needed it only to fit this scheme in our assumptions, but we can take it out without changing the performance of the scheme.

Note that the non-catastrophicity of ϕ_i is needed to make the syndrome matrix 'low density' , i.e. with a number of non-zero elements per row and per column which is small and does not grow with N. More precisely, the matrix $H_2 = \mathrm{Sum}_d \pi_N^1 \mathrm{Rep}_d$ is a random low density matrix with entries in $\{0,1\}$, depending only on c, d and π_N, with at most c elements equal to 1 in each column and at most d on any row. Instead $H_1 = (\psi^N)^{-1}$ depends on the choice of ψ, and is also low density, having a number of non-zero elements per row and per column at most equal to the degree of the polynomial $\psi^{-1}(D)$.

3.2 Main result: interleaver gain

The well-known analysis by Benedetto et al. [3] showed an interleaver gain, in the sense that average error probability is asymptotically vanishing when the interleaver length grows, and their result was true under the assumption that both constituent encoders where systematic recursive convolutional encoders and that the free distance of the outer encoder was $d_f^o \geq 2$ to ensure $\overline{P_b(e)} \to 0$ and $d_f^o \geq 3$ to have also $\overline{P_w(e)} \to 0$.

In this section, we will comment on how the classical assumptions on the constituent encoders can be adapted to our setting, and we will state our results about the interleaver gain. All the proofs will be given in Section 3.3.

From now on, we will be always considering a regular serial ensemble (see Definition 3.1), with outer encoder $\phi_o : U^{\mathbb{N}} \to (Y^r)^{\mathbb{N}}$ and inner encoder $\phi_i : (Y^s)^{\mathbb{N}} \to (\Gamma^l)^{\mathbb{N}}$, and with a family of interconnection groups (G_N) with invariants weight $\boldsymbol{w}_G$. The symbol error probability will be with respect to a fixed weight on the input group U, denoted $\boldsymbol{w}_{\text{in}}$, with the requirement that $\boldsymbol{w}_{\text{in}}(u) \neq 0$ for all $u \neq 0$.

First of all, we have to generalize the assumptions about the constituent encoders introduced in [3].

When considering one single convolutional encoder, non-catastrophicity is usually needed to ensure good asymptotic properties. However, when dealing with a concatenated scheme the assumption that all constituent encoders are non-catastrophic can be slightly weakened, as it was already recognized for example in [23] and in [35] (in the latter, the authors consider serial schemes where the inner encoder is heavily punctured and becomes not injective). The essential assumption is that the overall scheme is non-catastrophic, and this can be obtained by asking classical non-catastrophicity of the outer encoder and a weaker property of the inner encoder: ϕ_i must be non-catastrophic when restricted to the inputs he will actually receive, i.e. the permuted outer codewords.

When we are dealing with ensembles of concatenated codes, each code of the ensemble must be non-catastrophic, in the sense specified above. This leads to the following definition.

Definition 3.2 A regular serial ensemble with constituent encoders $\phi_o : U^{\mathbb{N}} \to (Y^r)^{\mathbb{N}}$ and $\phi_i : (Y^s)^{\mathbb{N}} \to (\Gamma^l)^{\mathbb{N}}$ and regular group family (G_N) is *concatenatedly non-catastrophic* if there exist two positive constants ζ_o and ζ_i such that, for all $N \in \mathbb{N}^*$ and for all $\boldsymbol{u} \in U^N$:

1. $\mathrm{w_H}(\boldsymbol{u}) \leq \zeta_o \left| \boldsymbol{w}_G \left(\phi_o^N(\boldsymbol{u}) \right) \right|$;
2. for all $\pi \in G_N$, $\left| \boldsymbol{w}_G \left(\phi_o^N(\boldsymbol{u}) \right) \right| \leq \zeta_i \mathrm{w_H} \left(\phi_i^N \circ \pi \circ \phi_o^N(\boldsymbol{u}) \right)$.

Notice that the requirement (1) is equivalent to asking that the convolutional encoder ϕ_o is non-catastrophic (see Prop. 2.2). In the examples introduced in previous section, we have an example where both encoders are non-catastrophic (Repeat-Convolute), and an example where only concatenated non-catastrophicity is true (Structured LDPC). In fact, in this second example, non-catastrophicity of ψ ensures sparsity of the parity-check matrix, but due to non-injectivity of $\mathrm{Sum_s}$ the inner encoder is indeed catastrophic; overall non-catastrophicity of the concatenated scheme is ensured by the systematic branch.

About the other classical assumptions on the constituent encoders ($d_f^o \geq 3$ and recursiveness of ϕ_i), they clearly must be re-stated considering the suitable connecting weight $\boldsymbol{w}_G$ instead of Hamming weight, using the definitions introduced in

Sect. 2.3. However, we will comment later in this section why these assumptions are sufficient and not necessary to obtain some interleaver gain, and how they can be weakened.

Now we will introduce some useful definitions, and then state the interleaver gain result, which will answer to the question: 'Is the average error probability asymptotically vanishing when the interleaver length grows to infinity? And if so, how fast is the decay?'. From now on, we will always assume that we are considering a concatenatedly non-catastrophic ensemble.

Let $\mathcal{C}_o^N = \phi_o^N(U^N)) \subseteq Y^{r(N+\nu_o)}$ be the outer block code, and let

$$H = \{\boldsymbol{w}_G(\boldsymbol{c}) : \boldsymbol{c} \in \mathcal{C}_o^N \text{ for some } N\,,\, \boldsymbol{c} \neq \boldsymbol{0}\}\,.$$

Notice that with this notation the requirement (2) in Definition 3.2 is equivalent to the following:

$$\forall N \in \mathbb{N}^*, \forall \boldsymbol{h} \in H, \forall \boldsymbol{c} \in Y^{r(N+\nu_o)} \text{ such that } \boldsymbol{w}_G(\boldsymbol{c}) = \boldsymbol{h},\, |\boldsymbol{h}| \leq \zeta_i |\boldsymbol{w}_{\mathrm{T}}\left(\phi_i^N(\boldsymbol{c})\right)|$$

Given $\boldsymbol{h} \in H$, we look at the decomposition of codewords in error events, as defined in Sections 2.3.1 and 2.3.4, and we define:

- $n_o(\boldsymbol{h}) = \max\{n(\boldsymbol{c}) \text{ such that } \exists N, \exists \boldsymbol{c} \in \mathcal{C}_o^N : \boldsymbol{w}_G(\boldsymbol{c}) = \boldsymbol{h}\}$
- $n_i(\boldsymbol{h}) = \max\{n(\boldsymbol{x}) \text{ s.t. } \exists N, \exists \boldsymbol{u} \in Y^{r(N+\nu_o)} : \boldsymbol{x} = \phi_i^N(\boldsymbol{u}), \boldsymbol{w}_G(\boldsymbol{u}) = \boldsymbol{h}\}$
- $f(\boldsymbol{h}) = 1 + |\boldsymbol{h}| - n_o(\boldsymbol{h}) - n_i(\boldsymbol{h})$

Remark 3.2 Both maxima in the above definition are well defined, since we clearly have $n(\boldsymbol{c}) \leq |\boldsymbol{h}|$, $n(\boldsymbol{x}) \leq |\boldsymbol{h}|$. Moreover, notice that, because of Remark 2.1 in 3.4, the sequence of sets

$$\{n(\boldsymbol{c}) \text{ such that } \exists \boldsymbol{c} \in \mathcal{C}_o^N : \boldsymbol{w}_G(\boldsymbol{c}) = \boldsymbol{h}\}$$

is increasing in N and so there exists $\bar{N}(\boldsymbol{h})$ such that

$$n_o(\boldsymbol{h}) = \max\{n(\boldsymbol{c}) \text{ such that } \exists \boldsymbol{c} \in \mathcal{C}_o^{\bar{N}(\boldsymbol{h})} : \boldsymbol{w}_G(\boldsymbol{c}) = \boldsymbol{h}\}$$

An analogous statement holds true for $n_i(\boldsymbol{h})$.

It is also clear that for what $n_o(\boldsymbol{h})$ is concerned, maximum can always be obtained with a codeword which only admits regular error events, while this is not necessarily true for $n_i(\boldsymbol{h})$. ■

Finally, we define:

$$\mu = \inf\{f(\boldsymbol{h}), \boldsymbol{h} \in H\}\,. \tag{3.1}$$

Notice that the function f takes values in $\mathbb{Z}$ and H is non-empty, so either $\mu = -\infty$ or $\mu = \min\{f(\boldsymbol{h}), \boldsymbol{h} \in H\}$. We will use the assumptions about the constituent encoders to ensure that we are in the interesting case when μ is positive.

Our main result (formally stated in Theorem 3.1) is that, for sufficiently good channels, if $\mu \geq 1$,

$$\overline{P_s(e)} \asymp N^{-\mu} \quad \text{and} \quad \overline{P_w(e)} \asymp N^{-\mu+1} \quad \text{for } N \to \infty$$

In addition to this interleaver gain result, we want to underline also the dependence of the error probability on the channel, following the steps of Benedetto et al. [3] and looking for an analogous of the classical effective free distance.

We define the set of the vectors $\boldsymbol{h}$ minimizing $f(\boldsymbol{h})$:

$$\mathcal{H} = \{\boldsymbol{h} \in H : f(\boldsymbol{h}) = \mu\}$$

and we define:

- $q^*(\boldsymbol{h}) = \max\{\mathrm{P}(\mathbf{0} \to \boldsymbol{x}) \colon \exists N, \exists \boldsymbol{u} \in Y^{r(N+\nu_o)} : \boldsymbol{x} = \phi_i^N(\boldsymbol{u}), \boldsymbol{w}_G(\boldsymbol{u}) = \boldsymbol{h}, n(\boldsymbol{x}) = n_i(\boldsymbol{h})\}$
- $q^* = \max\limits_{\boldsymbol{h} \in \mathcal{H}} \{q^*(\boldsymbol{h})\}$.

Remark 3.3

- We can prove that maxima in the definitions of $q^*(\boldsymbol{h})$ and q^* are well-defined. In principle the number of $\boldsymbol{x}$ involved in the maximum defining $q^*(\boldsymbol{h})$ is infinite. However, we can always restrict the search to a finite set, in the following way. As a first step, we can find an upper bound on the values of $\mathrm{P}(\mathbf{0} \to \boldsymbol{x})$ to consider, trivially by computing $\bar{q} = \mathrm{P}(\mathbf{0} \to \bar{\boldsymbol{x}})$ for one admissible $\bar{\boldsymbol{x}}$. Then we restrict our search to the set:

  $$X = \{\boldsymbol{x} \colon \mathrm{P}(\mathbf{0} \to \boldsymbol{x}) \geq \bar{q} \text{ and } \exists N, \exists \boldsymbol{u} \in Y^{r(N+\nu_o)} : \boldsymbol{x} = \phi_i^N(\boldsymbol{u}), \boldsymbol{w}_G(\boldsymbol{u}) = \boldsymbol{h}, n(\boldsymbol{x}) = n_i(\boldsymbol{h})\}$$

  Now note that $\mathrm{P}(\mathbf{0} \to \boldsymbol{x}) \geq \bar{q}$ implies $\gamma^{\mathrm{w_H}(\boldsymbol{x})} \geq \bar{q}$, i.e. $\mathrm{w_H}(\boldsymbol{x}) \leq \log \bar{q} / \log \gamma$. Now, by Prop. 2.1, we can bound the length of all error events in the decomposition of $\boldsymbol{x} \in \mathcal{X}$. This implies that, up to shift equivalence, the family of all possible error events appearing in $\boldsymbol{x} \in \mathcal{X}$ is finite. Therefore, also $\mathcal{X}$, up to shift equivalence, is finite. The same argument applies also to q^*.

- Later, we will also see that under suitable assumptions $\mathcal{H}$ is a finite set (Prop. 3.2). ■

Using the definition of q^*, we can state the interleaver gain result in a stronger way that underlines, additionally to the decay with N, also the dependence on the channel.

Theorem 3.1 Consider a regular and concatenatedly non-catastrophic serial ensemble $(\phi_o,\phi_i,(G_N)_{N\in\mathcal{N}})$, corresponding to the encoding scheme

$$\xrightarrow{U^N} \boxed{\phi_o^N} \xrightarrow{Y^{r(N+\nu_o)}} \boxed{\pi_N} \xrightarrow{Y^{sM_N}} \boxed{\phi_i^N} \xrightarrow{I^{l(M_N+\nu_i)}}$$

If $\mu \geq 1$, there exist positive constants c,c_1,c_2 and γ_0 (depending only on ϕ_o,ϕ_i and (G_N)) such that, for all Γ-symmetric channels with Bhattacharyya parameter $\gamma < \gamma_0$:

$$c'\, q^* N^{-\mu} \leq \overline{P_w(e)} \leq c_1 q^* c_2^{(\log q^*/\log\gamma)} N^{-\mu} + O(N^{-\mu-1})$$

Moreover, for a given input weight $\boldsymbol{w}_{\text{in}}$ (compatible with U and satisfying $\boldsymbol{w}_{\text{in}}(u) \neq 0$ for all $u \neq 0$),

$$c' \frac{\boldsymbol{w}_{\text{in}}^{\max}}{\boldsymbol{w}_{\text{in}}^{\min}}\, q^* N^{-\mu+1} \leq \overline{P_s(e)} \leq c_1 \frac{\boldsymbol{w}_{\text{in}}^{\max}}{\boldsymbol{w}_{\text{in}}^{\min}}\, q^* c_2^{(\log q^*/\log\gamma)} N^{-\mu+1} + O(N^{-\mu})$$

where $\boldsymbol{w}_{\text{in}}^{\max} = \max\limits_{u\in U} \boldsymbol{w}_{\text{in}}(u)$ and $\boldsymbol{w}_{\text{in}}^{\min} = \min\limits_{u\in U\setminus\{0\}} \boldsymbol{w}_{\text{in}}(u)$. □

The terms q^* in the lower bound and $q^* \frac{\log q^*}{\log\gamma}$ in the upper bound describe the behaviour of $\overline{P_s(e)}$ and $\overline{P_w(e)}$ with respect to the channel's noise. Note that $q^* = \mathrm{P}(\mathbf{0} \to \boldsymbol{c})$ for some word $\boldsymbol{c}$, so that if you denote $w^* = \mathrm{w_H}(\boldsymbol{c})$, $q^* \leq \gamma^{w^*}$ and $\frac{\log q^*}{\log\gamma} \leq w^*$. Hence, for a family of channels where you let $\gamma \to 0$ in such a way that the decreasing noise does not affect which words minimize $\mathbb{P}(\mathbf{0} \to \boldsymbol{c})$ (e.g. BSC, BEC, S-AWGN channel for fixed S, m-symmetric channel), all the information on how fast $\overline{P_w(e)}$ and $\overline{P_s(e)}$ tend to zero is contained in q^*.

Now we will show how the free distance of ϕ_o and the recursiveness of ϕ_i come into the picture. First of all, we generalize the classical assumptions in the most natural way, simply replacing Hamming weight with the interconnection weight $\boldsymbol{w}_G$: this will ensure that there is an interleaving gain (namely that $\mu \geq 1$). From now on, let's denote by d_f^o the $\boldsymbol{w}_G$–free distance of ϕ_o.

Proposition 3.1 Assume that $d_f^o \geq 2$ and ϕ_i is $\boldsymbol{w}_G$–recursive. Then

$$\lfloor (d_f^o + 1)/2 \rfloor \leq \mu \leq d_f^o\,.$$

In particular, $\mu \geq 1$ and if $d_f^o \geq 3$ then $\mu \geq 2$. □

In some particular cases we will give tighter upper bounds on μ (see Sect. 3.4).

The strong assumptions used in Prop. 3.1 have also another interesting consequence:

Proposition 3.2 If $d_f^o \geq 3$ and ϕ_i is $\boldsymbol{w}_G$–recursive, then $\mathcal{H}$ is a finite set. □

However, these assumptions are not necessary to obtain an interleaver gain. For example, in the case of parallel concatenations with multiple branches, those assumptions would mean that all constituent encoders are recursive, while it is known that there is an interleaver gain, even if smaller, also when only some of them are recursive. Also, a relaxation of the classical assumptions will allow us to give results about very interesting examples, such as the heavily punctured serial schemes considered in [35], or the class of structured LDPC interpreted as serial schemes that we introduced as Example (E2). Thus, we are interested in a generalization of Prop. 3.1.

Proposition 3.3 Assume that the interconnection weight has the structure $\boldsymbol{w}_G = (\boldsymbol{w}_1,\boldsymbol{w}_2) : Y^{sM} \to \mathbb{N}^{\rho_1} \times \mathbb{N}^{\rho_2}$ (possibly $\rho_2 = 0$, but $\rho_1 \geq 1$); denote by $d^o_{f,1}$ the $\boldsymbol{w}_1$–free distance of ϕ_o. Assume that $d^o_{f,1} \geq 2$ and ϕ_i is $\boldsymbol{w}_1$–recursive. Then,

$$\lfloor (d^o_{f,1} + 1)/2 \rfloor \leq \mu \leq d^o_f \,.$$

In particular, $\mu \geq 1$, and if $d^o_{f,1} \geq 3$ then $\mu \geq 2$. □

Notice that Prop. 3.1 is a particular case of Prop. 3.3, where $\rho_2 = 0$ and so $d^o_f = d^o_{f,1}$.

3.3 Proofs of the main results

In this section, we prove our main results, i.e. Theorem 3.1 and Proposition 3.3. We prove the upper bound for $\overline{P_s(e)}$ and the lower bound for $\overline{P_w(e)}$; the whole result stated in Theorem 3.1 is then obtained by the simple remark

$$\overline{P_s(e)} \geq \frac{1}{N} \frac{\boldsymbol{w}_{\text{in}}^{\min}}{\boldsymbol{w}_{\text{in}}^{\max}} \overline{P_w(e)} \,.$$

3.3.1 Upper bound

This proof is based on the union-Bhattacharyya bound (see e.g. [39]) and on estimations of the weight enumerating coefficients of the constituent encoders.

We will consider only the case when the symbol error rate $P_s(e)$ is defined with respect to Hamming input weight ($\boldsymbol{w}_{\text{in}} = \mathrm{w_H}$); however this will give results true for every other compatible weight, up to a positive constant factor.

The well-known union bound gives

$$\overline{P_s(e)} \leq \sum_w \sum_{\boldsymbol{d}} \frac{w}{N} \overline{A_{w,d}}^N Q(\boldsymbol{d}) \tag{3.2}$$

where $\overline{A_{w,\boldsymbol{d}}}^N$ is the average number of codewords of a serial ensemble with input Hamming weight w and output type weight $\boldsymbol{d}$.

The standard technique (see [39, 3]) is to express $\overline{A_{w,\boldsymbol{d}}}^N$ as a function of suitable enumerating coefficients of the constituent encoders. Here, we need:

- $A^{o,N}_{w,\boldsymbol{h}}$ the number of codewords of ϕ_o^N with input Hamming weight $\boldsymbol{w}$ and output invariants vector weight $\boldsymbol{h}$;
- $A^{i,N}_{\boldsymbol{h},\boldsymbol{d}}$ the number of codewords of ϕ_i^N with input invariants vector weight $\boldsymbol{h}$ and output type weight $\boldsymbol{d}$.

Proposition 3.4 $$\overline{A_{w,\boldsymbol{d}}}^N = \sum_{\boldsymbol{h}\in H} \frac{A^{o,N}_{w,\boldsymbol{h}} A^{i,N}_{\boldsymbol{h},\boldsymbol{d}}}{|Y_{\boldsymbol{h}}^{r(N+\nu_o)}|}.$$ □

Proof:

$$\overline{A_{w,d}}^N = \sum_{\boldsymbol{u}:\mathrm{w_H}(\boldsymbol{u})=w} \sum_{\boldsymbol{v}:\boldsymbol{w}_\mathrm{T}(\phi_i^N(\boldsymbol{v}))=\boldsymbol{d}} \mathbb{P}(\Pi_N(\phi_o^N(\boldsymbol{u})) = \boldsymbol{v})$$

By Remark 3.1,

$$\mathbb{P}(\Pi_N(\phi_o^N(\boldsymbol{u})) = \boldsymbol{v}) = \frac{|G_N(\phi_o^N(\boldsymbol{u}),\boldsymbol{v})|}{|G_N|} = \begin{cases} 0 & \text{if } \boldsymbol{w}_G(\phi_o^N(\boldsymbol{u})) \neq \boldsymbol{w}_G(\boldsymbol{v}) \\ \frac{1}{|Y_{\boldsymbol{h}}^{r(N+\nu_o)}|} & \text{if } \boldsymbol{w}_G(\phi_o^N(\boldsymbol{u})) = \boldsymbol{w}_G(\boldsymbol{v}) = \boldsymbol{h} \end{cases}$$

Substituting in the expression above, we obtain the thesis. ■

By Lemma 2.1, we know that $|Y_{\boldsymbol{h}}^{r(N+\nu_o)}| \geq \binom{r(N+\nu_o)}{\boldsymbol{h}}$. Thus, by the inequality (3.2) and Prop. 3.4 we have

$$\overline{P_s(e)} \leq \sum_{w,\boldsymbol{h},d} \frac{w}{N} \frac{1}{\binom{r(N+\nu_o)}{\boldsymbol{h}}} A^{o,N}_{w,\boldsymbol{h}} A^{i,N}_{\boldsymbol{h},\boldsymbol{d}} Q(\boldsymbol{d}) \tag{3.3}$$

We have some inequalities involving the indexes $w,\boldsymbol{h},\boldsymbol{d}$ which are necessary conditions to have non-zero $A^{o,N}_{w,\boldsymbol{h}} A^{i,N}_{\boldsymbol{h},\boldsymbol{d}}$. They are listed in Definition 3.3 and Prop. 3.5.

Definition 3.3 Let $I \subseteq \mathbb{N}^* \times H \times \mathbb{N}^{\Gamma\setminus\{0\}}$ be the set of triples $(w,\boldsymbol{h},\boldsymbol{d})$ satisfying the following conditions:

- $1 \leq w \leq N$;
- $|d| \leq l\,(M_N + \nu_i)$;
- $w \leq \zeta_o|\boldsymbol{h}|$ and $|\boldsymbol{h}| \leq \zeta_i|\boldsymbol{d}|$ (ζ_o and ζ_i as in Def. 3.2).

□

Proposition 3.5 If $(w,\boldsymbol{h},\boldsymbol{d}) \notin I$, then $A^{o,N}_{w,\boldsymbol{h}} A^{i,N}_{\boldsymbol{h},\boldsymbol{d}} = 0$. □

Proof: The first two inequalities are trivial remarks about the length of the input and code words and the definition of free distance; the last one is the concatenated non-catastrophicity of the ensemble (see Def. 3.2). ■

Now, we need to estimate the product $A^{o,N}_{w,\boldsymbol{h}} A^{i,N}_{\boldsymbol{h},\boldsymbol{d}}$ when it is non-zero. We start with the following inequalities deriving from Prop. 2.5.

Proposition 3.6 There exist some positive constants a_o,a_i,b_o,b_i such that, for every $(w,\boldsymbol{h},\boldsymbol{d}) \in I$:

1. $A^{o,N}_{w,\boldsymbol{h}} \leq \sum_{n_o=1}^{n_o(\boldsymbol{h})} \binom{N+n_o}{n_o} a_o^w b_o^{|\boldsymbol{h}|}$

2. $A^{i,N}_{\boldsymbol{h},\boldsymbol{d}} \leq \sum_{n_i=0}^{n_i^{\max}} \binom{N+n_i}{n_i} a_i^{|\boldsymbol{h}|} b_i^{|\boldsymbol{d}}$, where $n_i^{\max} = \begin{cases} n_i(\boldsymbol{h}) & \text{if } Q(\boldsymbol{d}) \leq q^*(\boldsymbol{h}) \\ n_i(\boldsymbol{h}) - 1 & \text{if } Q(\boldsymbol{d}) > q^*(\boldsymbol{h}) \end{cases}$

Proof: Let $\mathrm{w}_{\max} = \max\{|\boldsymbol{w}_G(v)| : v \in Y^r\}$, so that $|\boldsymbol{w}_G(\boldsymbol{v})|/\mathrm{w}_{\max} \leq \mathrm{w}_{\mathrm{H}}(\boldsymbol{v}) \leq |\boldsymbol{w}_G(\boldsymbol{v})|$ for all $\boldsymbol{v} \in Y^{r(N+\nu_o)}$. Then:

$$A^{o,N}_{w,\boldsymbol{h}} \leq \sum_{n_o=1}^{n_o(\boldsymbol{h})} \sum_{h'=\lfloor |\boldsymbol{h}|/\mathrm{w}_{\max} \rfloor}^{|\boldsymbol{h}|} \Lambda^{o,N}_{w,h',n_o}$$

where $\Lambda^{o,N}_{w,h',n_o}$ is the input/output support enumerating coefficient of ϕ_o^N, as defined in Sect. 2.3.5. The conclusion now follows in a straightforward way from Prop. 2.5.

The proof for the inner encoder is similar, but we want to exploit the fact that, by definition of $q^*(\boldsymbol{h})$, there are no codewords of input weight $\boldsymbol{h}$, output weight $\boldsymbol{d}$ such that $Q(\boldsymbol{d}) > q^*(\boldsymbol{h})$ having $n_i(\boldsymbol{h})$ error events in their decomposition. To do so, we need to define $A^{i,N}_{\boldsymbol{h},\boldsymbol{d},n}$ to be the number of codewords of ϕ_i^N with input invariants vector weight $\boldsymbol{h}$, output type weight $\boldsymbol{d}$, and n error events in the decomposition, so that

$$A^{i,N}_{\boldsymbol{h},\boldsymbol{d}} = \sum_{n_i=0}^{n_i(\boldsymbol{h})} A^{i,N}_{\boldsymbol{h},d,\boldsymbol{n}_i} = \sum_{n_i=0}^{n_i^{\max}} A^{i,N}_{\boldsymbol{h},\boldsymbol{d},n_i}$$

because $A^{i,N}_{\boldsymbol{h},\boldsymbol{d},n_i(\boldsymbol{h})} = 0$ if $Q(\boldsymbol{d}) > q^*(\boldsymbol{h})$. Then we conclude the proof as for the outer encoder, with $\hat{\mathrm{w}}_{\max} = \max\{|\boldsymbol{w}_{\mathrm{T}}(g)| : g \in \Gamma^l\}$:

$$A^{i,N}_{\boldsymbol{h},\boldsymbol{d}} \leq \sum_{n_i=1}^{n_i^{\max}} \sum_{h'=\left\lfloor \frac{|\boldsymbol{h}|}{\mathrm{w}_{\max}} \right\rfloor}^{|\boldsymbol{h}|} \sum_{d'=\left\lfloor \frac{|\boldsymbol{d}|}{\hat{\mathrm{w}}_{\max}} \right\rfloor}^{|\boldsymbol{d}|} \Lambda^{i,N}_{h',d',n_i}$$

■

We now prove the following combinatorial inequality.

Proposition 3.7 There exists a constant $C > 0$ such that, for all $\boldsymbol{h} \in H$ with $|\boldsymbol{h}| \leq \mathrm{w}_{\max} r(N+\nu_o)$ (where $\mathrm{w}_{\max} = \max\{|\boldsymbol{w}_G(v)| : v \in Y^r\}$):

$$\frac{1}{\binom{r(N+\nu_o)}{\boldsymbol{h}}} \sum_{n_o=1}^{n_o(\boldsymbol{h})} \sum_{n_i=0}^{n_i^{\max}} \binom{N+n_o}{n_o} \binom{N+n_i}{n_i} \leq \begin{cases} C^{|\boldsymbol{h}|} \frac{|\boldsymbol{h}|^{f(\boldsymbol{h})-1}}{N^{f(\boldsymbol{h})-1}} & \text{if } n_i^{\max} = n_i(\boldsymbol{h}) \\ C^{|\boldsymbol{h}|} \frac{|\boldsymbol{h}|^{f(\boldsymbol{h})}}{N^{f(\boldsymbol{h})}} & \text{if } n_i^{\max} = n_i(\boldsymbol{h}) - 1 \end{cases}$$

Proof: First, we have $\binom{r(N+\nu_o)}{\boldsymbol{h}} \geq \left[\frac{r(N+\nu_o)}{e|\boldsymbol{h}|}\right]^{|\boldsymbol{h}|}$.

This gives $\frac{1}{\binom{r(N+\nu_o)}{\boldsymbol{h}}} \leq C^{|\boldsymbol{h}|} \left[\frac{|\boldsymbol{h}|}{N}\right]^{|\boldsymbol{h}|}$ for some constant $C > 0$.

For the other terms, we use the following combinatorial inequalities:

- $\binom{N+n}{n} \leq \left[\frac{s}{N}\right]^{s-n} \binom{N+s}{s}$ for all $n \geq 0$, s,$N \geq 1$ satisfying $s \geq n$;
- there exists a constant $c > 0$ such that $\binom{N+n}{n} \leq c \left[\frac{N+n}{N}\right]^N \left[\frac{N+n}{n}\right]^n$ for all n,$N \geq 1$;
- $\left[\frac{N+n}{N}\right]^N \leq e^n$ for all $n \geq 0$, $N \geq 1$.

As $n_o(\boldsymbol{h}) \leq |\boldsymbol{h}|$, these inequalities give

$$\begin{aligned} \sum_{n_o=1}^{n_o(\boldsymbol{h})} \binom{N+n_o}{n_o} &\leq \sum_{n_o=1}^{n_o(\boldsymbol{h})} \left[\frac{|\boldsymbol{h}|}{N}\right]^{|\boldsymbol{h}|-n_o} c e^{|\boldsymbol{h}|} \left[\frac{N+|\boldsymbol{h}|}{|\boldsymbol{h}|}\right]^{|\boldsymbol{h}|} \\ &\leq c_o^{|\boldsymbol{h}|} \sum_{n_o=1}^{n_o(\boldsymbol{h})} |\boldsymbol{h}|^{-n_o} N^{n_o} \\ &\leq C_o^{|\boldsymbol{h}|} |\boldsymbol{h}|^{-n_o(\boldsymbol{h})} N^{n_o(\boldsymbol{h})} \end{aligned}$$

(for some positive constants c_o and C_o). The second inequality is true thanks to the assumption that $|\boldsymbol{h}| \leq \mathrm{w}_{\max} r(N+\nu_o)$. Similar estimation can be obtained for the summation relative to the inner part and this yields the result. ■

If we substitute the estimations given by Propositions 3.6 and 3.7 into the expression (3.3) and we use Prop. 3.5, we get, for some positive constants C_1,C_2,C_3:

$$\overline{P_s(e)} \leq \sum_{\substack{(w,\boldsymbol{h},\boldsymbol{d}) \in I: \\ Q(\boldsymbol{d}) \leq q^*(\boldsymbol{h})}} \frac{|\boldsymbol{h}|^{f(\boldsymbol{h})-1}}{N^{f(\boldsymbol{h})}} C_1^w C_2^{|\boldsymbol{h}|} C_3^{|\boldsymbol{d}|} Q(\boldsymbol{d}) + \sum_{\substack{(w,\boldsymbol{h},\boldsymbol{d}) \in I: \\ Q(\boldsymbol{d}) > q^*(\boldsymbol{h})}} \frac{|\boldsymbol{h}|^{f(\boldsymbol{h})}}{N^{f(\boldsymbol{h})+1}} C_1^w C_2^{|\boldsymbol{h}|} C_3^{|\boldsymbol{d}|} Q(\boldsymbol{d}) \tag{3.4}$$

Now, we split the first summation into two terms, separating $\boldsymbol{h} \in \mathcal{H}$ from $\boldsymbol{h} \notin \mathcal{H}$. Define:

- $I_\mu = \{(w,\boldsymbol{h},\boldsymbol{d}) \in I : f(\boldsymbol{h}) = \mu, Q(\boldsymbol{d}) \le q^*(\boldsymbol{h})\}$,
- $I_> = \{(w,\boldsymbol{h},\boldsymbol{d}) \in I : f(\boldsymbol{h}) > \mu, Q(\boldsymbol{d}) \le q^*(\boldsymbol{h})\}$,
- $I_* = \{(w,\boldsymbol{h},\boldsymbol{d}) \in I : Q(\boldsymbol{d}) > q^*(\boldsymbol{h})\}$.

Eq. (3.4) can be re-written as follows:

$$\begin{aligned}\overline{P_s(e)} \le{}& \frac{1}{N^\mu} \sum_{(w,\boldsymbol{h},\boldsymbol{d})\in I_\mu} |\boldsymbol{h}|^{\mu-1} C_1^w C_2^{|\boldsymbol{h}|} C_3^{|\boldsymbol{d}|} Q(\boldsymbol{d}) \\ &+ \frac{1}{N^{\mu+1}} \sum_{(w,\boldsymbol{h},\boldsymbol{d})\in I_>} C_1^w \left(\frac{|\boldsymbol{h}|}{N}\right)^{f(\boldsymbol{h})-\mu-1} |\boldsymbol{h}|^\mu C_2^{|\boldsymbol{h}|} C_3^{|\boldsymbol{d}|} Q(\boldsymbol{d}) \\ &+ \frac{1}{N^{\mu+1}} \sum_{(w,\boldsymbol{h},\boldsymbol{d})\in I_*} C_1^w \left(\frac{|\boldsymbol{h}|}{N}\right)^{f(\boldsymbol{h})-\mu} |\boldsymbol{h}|^\mu C_2^{|\boldsymbol{h}|} C_3^{|\boldsymbol{d}|} Q(\boldsymbol{d})\end{aligned}$$

In the following we show that the first summation is bounded by $cq^* \exp(\log q^* / \log p)$ (Prop. 3.8), while the second and the third one are bounded by $c'(\gamma)$ (Prop. 3.9), thus ending the proof of the upper bound.

Proposition 3.8 There exist some positive constants γ_0 and c such that, for all BIOS channel with $\gamma < \gamma_0$,

$$\sum_{(w,\boldsymbol{h},\boldsymbol{d})\in I_\mu} |\boldsymbol{h}|^{\mu-1} C_1^w C_2^{|\boldsymbol{h}|} C_3^{|\boldsymbol{d}|} Q(\boldsymbol{d}) \le cq^* \exp(\log q^* / \log p)$$

□

Proof: Recall that $(w,\boldsymbol{h},\boldsymbol{d}) \in I_\mu$ implies that $w \le \zeta_o|\boldsymbol{h}|$, $|\boldsymbol{h}| \le \zeta_i|\boldsymbol{d}|$ and $Q(\boldsymbol{d}) \le q^*(\boldsymbol{h}) \le q^*$. So:

$$\begin{aligned}&\sum_{(w,\boldsymbol{h},\boldsymbol{d})\in I_\mu} |\boldsymbol{h}|^{\mu-1} C_1^w C_2^{|\boldsymbol{h}|} C_3^{|\boldsymbol{d}|} Q(\boldsymbol{d}) \\ &\quad\le \sum_{\substack{\boldsymbol{d}\in\mathbb{N}^{\Gamma\setminus\{0\}} \\ Q(\boldsymbol{d})\le q^*}} \sum_{\substack{\boldsymbol{h}\in\mathbb{N}^\rho \\ |\boldsymbol{h}|\le\zeta_i d}} |\boldsymbol{h}|^{\mu-1} C_2^{|\boldsymbol{h}|} \Big(\sum_{w\le\zeta_o|\boldsymbol{h}|} C_1^w \Big) C_3^{|\boldsymbol{d}|} Q(\boldsymbol{d}) \\ &\quad\le \sum_{\substack{\boldsymbol{d}\in\mathbb{N}^{\Gamma\setminus\{0\}} \\ Q(\boldsymbol{d})\le q^*}} \sum_{\substack{\boldsymbol{h}\in\mathbb{N}^\rho \\ |\boldsymbol{h}|\le\zeta_i d}} |\boldsymbol{h}|^{\mu-1} C_2^{|\boldsymbol{h}|} \zeta_o|\boldsymbol{h}| C_1^{|\boldsymbol{h}|} C_3^{|\boldsymbol{d}|} Q(\boldsymbol{d}) \\ &\quad\le \sum_{\substack{\boldsymbol{d}\in\mathbb{N}^{\Gamma\setminus\{0\}} \\ Q(\boldsymbol{d})\le q^*}} K^{|\boldsymbol{d}|} Q(\boldsymbol{d}) \qquad \text{(for some suitable } K > 0).\end{aligned}$$

Now, we split the summation, recalling the Bhattacharyya bound $Q(\boldsymbol{d}) \leq \gamma^{|\boldsymbol{d}|}$:

$$\sum_{\substack{\boldsymbol{d}\in\mathbb{N}^{\Gamma\setminus\{0\}}\\ Q(\boldsymbol{d})\leq q^*}} K^{|\boldsymbol{d}|}Q(\boldsymbol{d}) = \sum_{\substack{\boldsymbol{d}\in\mathbb{N}^{\Gamma\setminus\{0\}}\\ Q(\boldsymbol{d})\leq q^*,\gamma^{|\boldsymbol{d}|}>q^*}} K^{|\boldsymbol{d}|}q^* + \sum_{\substack{\boldsymbol{d}\in\mathbb{N}^{\Gamma\setminus\{0\}}\\ \gamma^{|\boldsymbol{d}|}\leq q^*}} K^{|\boldsymbol{d}|}\gamma^{|\boldsymbol{d}|}$$

Now let's find a bound for the number of $\boldsymbol{d}$'s involved in the first summation: $Q(\boldsymbol{d}) \leq q^*$ implies $\gamma^{|\boldsymbol{d}|} \leq q^*$, so $|\boldsymbol{d}| \leq \log q^*/\log\gamma$ and so there are less than $(|\Gamma|-1)^{\log q^*/\log p}$ type weights satisfying this inequality:

$$\sum_{\substack{\boldsymbol{d}\in\mathbb{N}^{\Gamma\setminus\{0\}}\\ Q(\boldsymbol{d})\leq q^*,\gamma^{|\boldsymbol{d}|}\geq q^*}} K^{|\boldsymbol{d}|}q^* \leq ((|\Gamma|-1)K)^{\log q^*/\log p}q^*$$

For the second term, note that, for $\gamma < 1/K$, the series is convergent, and bounded by a constant times its first term, which has $|\boldsymbol{d}| = \log\gamma/\log q^*$, i.e.

$$\sum_{\substack{\boldsymbol{d}\in\mathbb{N}^{\Gamma\setminus\{0\}}\\ \gamma^{|\boldsymbol{d}|}\leq q^*}} K^{|\boldsymbol{d}|}\gamma^{|\boldsymbol{d}|} \leq CK^{(\log\gamma/\log q^*)}q^*$$

■

Proposition 3.9 There exists a constant $\gamma_0 > 0$, depending on ϕ_o,ϕ_i and (G_N) and there exists $c'(\gamma) > 0$ depending only on γ such that, for all $\gamma < \gamma_0$

$$\sum_{(w,\boldsymbol{h},\boldsymbol{d})\in I_>} C_1^w\left(\frac{|\boldsymbol{h}|}{N}\right)^{f(\boldsymbol{h})-\mu-1}|\boldsymbol{h}|^\mu C_2^{|\boldsymbol{h}|}C_3^{|\boldsymbol{d}|}Q(\boldsymbol{d}) + \sum_{(w,\boldsymbol{h},\boldsymbol{d})\in I_*} C_1^w\left(\frac{|\boldsymbol{h}|}{N}\right)^{f(\boldsymbol{h})-\mu}|\boldsymbol{h}|^\mu C_2^{|\boldsymbol{h}|}C_3^{|\boldsymbol{d}|}Q(\boldsymbol{d}) \;\leq\; c'(\gamma)$$

Proof: Notice that, for $(w,\boldsymbol{h},\boldsymbol{d}) \in I_>$, $0 \leq f(\boldsymbol{h}) - \mu - 1 \leq |\boldsymbol{h}| \leq cN$, where the first inequality holds true because $\boldsymbol{h} \in H \setminus \mathcal{H}$, the second immediately follows from the definitions of $f(\boldsymbol{h})$ and μ, the third is true, for a suitable $c > 1$, because $|\boldsymbol{h}| \leq r(N + \nu_o)$ for all $\boldsymbol{h} \in H$. These inequalities imply that

$$(|\boldsymbol{h}|/N)^{f(\boldsymbol{h})-\mu-1} \leq c^{|\boldsymbol{h}|}$$

Analogously, for all $(w,\boldsymbol{h},d) \in I_*$, $0 \leq f(\boldsymbol{h}) - \mu \leq |\boldsymbol{h}| \leq cN$ and so

$$(|\boldsymbol{h}|/N)^{f(\boldsymbol{h})-\mu} \leq c^{|\boldsymbol{h}|}$$

This gives:

$$\begin{aligned}
&\sum_{(w,\boldsymbol{h},\boldsymbol{d})\in I_>} C_1^w \left(\frac{|\boldsymbol{h}|}{N}\right)^{f(\boldsymbol{h})-\mu-1} |\boldsymbol{h}|^\mu C_2^{|\boldsymbol{h}|} C_3^{|\boldsymbol{d}|} Q(\boldsymbol{d}) \\
&\qquad + \sum_{(w,\boldsymbol{h},d)\in I_*} C_1^w \left(\frac{|\boldsymbol{h}|}{N}\right)^{f(\boldsymbol{h})-\mu} |\boldsymbol{h}|^\mu C_2^{|\boldsymbol{h}|} C_3^{|\boldsymbol{d}|} Q(\boldsymbol{d}) \\
&\leq \sum_{(w,\boldsymbol{h},\boldsymbol{d})\in I} C_1^w c^{|\boldsymbol{h}|} C_2^{|\boldsymbol{h}|} C_3^{|\boldsymbol{d}|} Q(\boldsymbol{d})
\end{aligned}$$

Noticing also that $\sum_{w\leq\zeta_o|\boldsymbol{h}|} C_1^w \leq \zeta_o|\boldsymbol{h}|C_1^{|\boldsymbol{h}|}$, we have:

$$\sum_{(w,\boldsymbol{h},\boldsymbol{d})\in I} C_1^w c^{|\boldsymbol{h}|} C_2^{|\boldsymbol{h}|} C_3^{|\boldsymbol{d}|} Q(\boldsymbol{d}) \leq \sum_{\boldsymbol{d}\in\mathbb{N}^{\Gamma}\setminus\{0\}} \sum_{\boldsymbol{h}:|\boldsymbol{h}|\leq\zeta_i|\boldsymbol{d}|} \zeta_o|\boldsymbol{h}|C_1^{|\boldsymbol{h}|} c^{|\boldsymbol{h}|} |\boldsymbol{h}|^\mu C_2^{|\boldsymbol{h}|} C_3^{|\boldsymbol{d}|} Q(\boldsymbol{d})$$

Now notice that, for some $K > 1$,

$$\sum_{\boldsymbol{h}:|\boldsymbol{h}|\leq\zeta_i|\boldsymbol{d}|} \zeta_o|\boldsymbol{h}|C_1^{|\boldsymbol{h}|} c^{|\boldsymbol{h}|} |\boldsymbol{h}|^\mu C_2^{|\boldsymbol{h}|} \leq K^{|\boldsymbol{d}|}$$

Finally, we use the Bhattacharyya bound.

$$\sum_{\boldsymbol{d}\in\mathbb{N}^{\Gamma}\setminus\{0\}} (KC)_3^{|\boldsymbol{d}|} Q(\boldsymbol{d}) \leq \sum_{d\in\mathbb{N}} \sum_{\boldsymbol{d}\in\mathbb{N}^{\Gamma}\setminus\{0\}:|\boldsymbol{d}|=d} (KC_3\gamma)^d = c'(\gamma) < \infty$$

if γ is sufficiently small to ensure convergence. ■

3.3.2 Lower bound

The lower bound is based on the following simple remark involving the equivocation probability.

Remark 3.4 If $\boldsymbol{c} \in \mathcal{C}^N$, then $P_w(e) \geq \mathrm{P}(\boldsymbol{0} \to \boldsymbol{c})$. So, if you define $Q_{\max}(\pi_N) := \max\{\mathrm{P}(\boldsymbol{0} \to \boldsymbol{c}), \boldsymbol{c} \in \phi_i^N \circ \pi_N \circ \phi_o^N(U^N)\}$, for any q,

$$\overline{P_w(e)} \geq q\,\mathbb{P}\big(Q_{\max}(\Pi_N) \geq q\big)$$

■

We focus our attention on the value $q = q^*$, and we find the following lower bound to $\mathbb{P}\big(Q_{\max}(\Pi_N) \geq q\big)$, thus ending the proof of the lower bound in Theorem 3.1.

Proposition 3.10 If $\mu \geq 1$, there exists a constant $C > 0$ such that

$$\mathbb{P}\big(Q_{\max}(\Pi_N) \geq q\big) \geq CN^{-\mu+1}\,.$$

□

In the remainder of this section, we will prove Prop. 3.10. To do so, we need to define some particular codewords which are essential for the bound. We start fixing once and for all the following objects:

1. A weight vector $\boldsymbol{h} \in H$ such that $q^*(\boldsymbol{h}) = q^*$.
2. An outer codeword $\boldsymbol{c}^* \in \phi_o^N(U^N)$ such that $\boldsymbol{w}_G(\boldsymbol{c}^*) = \boldsymbol{h}$ and $n(\boldsymbol{c}) = n_o(\boldsymbol{h})$. Let $n_o = n_o(\boldsymbol{h})$ and let $\boldsymbol{c}^* = \boldsymbol{c}_1^* + \ldots + \boldsymbol{c}_{n_o+1}^*$ be an error event decomposition of $\boldsymbol{c}^*$ (see Sect. 2.3.4). Denote by l_k the length of $\boldsymbol{c}_k^*$ and let $l_{\max} = \max\{l_1, \ldots, l_{n_o}\}$. If $\mu = 1$, we need the different definition

   $$l_{\max} = \max\{l_1, \ldots, l_{n_o}, \tfrac{2e|\boldsymbol{h}|}{n_o r}\big(\sqrt[n_o]{1 + (2e|\boldsymbol{h}||Y|^{|\boldsymbol{h}|})^{-1/2}} - 1\big)^{-1}\}$$

   (the reason will be clear at the end of the proof).
3. An input word $\boldsymbol{u}^*$ for the inner encoder, such that $\boldsymbol{w}_G(\boldsymbol{u}^*) = \boldsymbol{h}$ and such that $\boldsymbol{x}^* = \phi_i^N(\boldsymbol{u}^*)$ has equivocation $\mathrm{P}(\boldsymbol{0} \to \boldsymbol{x}^*) = q^*(\boldsymbol{h})$ and $n(\boldsymbol{x}^*) = n_i(\boldsymbol{h})$. Let $n_i = n_i(\boldsymbol{h})$ and let $\boldsymbol{x}^* = \boldsymbol{x}_1^* + \ldots + \boldsymbol{x}_{n_i+1}^*$ be an error event decomposition of $\boldsymbol{x}^*$. Denote by $\boldsymbol{u}_k^*$ the input error event corresponding to $\boldsymbol{x}_k^*$ and by λ_k its length (with $\lambda_{n_i+1} = 0$ if there is no terminating event). Let $\lambda_{\max} = \max\{\lambda_1, \ldots, \lambda_{n_i}\}$, modified as $\lambda_{\max} = \max\{\lambda_1, \ldots, \lambda_{n_i}, \frac{2e|\boldsymbol{h}|}{n_o r}\big(\sqrt[n_i]{1 + (2e|\boldsymbol{h}||Y|^{|\boldsymbol{h}|})^{-1/2}} - 1\big)^{-1}\}$ when $\mu = 1$.

Notice that $\boldsymbol{c}^*$ can be chosen in such a way that it doesn't have any terminating event and that it does not depend on N, while this may not be possible for $\boldsymbol{u}^*$. However, we can assume that the error events $\boldsymbol{x}_k^*$ and their inputs $\boldsymbol{u}_k^*$ remain the same apart from some possible translations (see Remark 2.1). Also remember that $n_o \geq 1$, $n_i \geq 0$.

Now, we select a sufficiently big set of shift equivalent words for both $\boldsymbol{c}^*$ and $\boldsymbol{x}^*$, choosing many positions for the error events of $\boldsymbol{c}^*$ and for the input error events of $\boldsymbol{u}^*$, across all the time axis $[0, N + \nu_o - 1]$ for $\boldsymbol{c}^*$ and $[0, M_N - 1]$ for $\boldsymbol{u}^*$.

Let's start with $\boldsymbol{c}^*$. Define $\mathcal{A} = [0, \left\lfloor \frac{N}{n_o l_{\max}} \right\rfloor - 1]$. Given $\boldsymbol{a} \in \mathcal{A}^{n_o}$, we define $\boldsymbol{c}_{\boldsymbol{a}}^*$ to be the outer codeword which, for every $k = 1, \ldots, n_o$, contains exactly one shifted copy of the error event $\boldsymbol{c}_k^*$ starting at time $a_k l_{\max} + (k-1)|\mathcal{A}| l_{\max}$. Clearly by construction all error events in $c_{\boldsymbol{a}}^*$ have disjoint support.

In the same way, we consider the inner input word $\boldsymbol{u}^*$. For $n_i \geq 1$, define $\mathcal{B} = [0, \left\lfloor \frac{M_N - \lambda_{n_i+1}}{n_i \lambda_{\max}} \right\rfloor - 1]$. Given $\boldsymbol{b} \in \mathcal{B}^{n_i}$ we define $\boldsymbol{u}_{\boldsymbol{b}}^*$ to be the inner input word

which, for every $k = 1, \ldots, n_i$, contains exactly one translated copy of the input error event $\boldsymbol{u}_k^*$ starting at time $b_k\lambda_{\max} + (k-1)|\mathcal{B}|\lambda_{\max}$, while the terminating event $\boldsymbol{u}_{n_i+1}^*$ (if there is one) remains fixed in its position in the interval $[M_N - \lambda_{n_i+1} - 1, M_N - 1]$. Let $\boldsymbol{x}_{\boldsymbol{b}}^*$ be the output $\boldsymbol{x}_{\boldsymbol{b}}^* = \phi_i^N(\boldsymbol{u}_{\boldsymbol{b}}^*)$.

Given $\boldsymbol{a} \in \mathcal{A}^{n_o}$ and $\boldsymbol{b} \in \mathcal{B}^{n_i}$, if $n_i \geq 1$ we define the event

$$E_{\boldsymbol{a},\boldsymbol{b}} = \{\Pi_N(\boldsymbol{c}_{\boldsymbol{a}}^*) = \boldsymbol{u}_{\boldsymbol{b}}^*\} = G_N(\boldsymbol{c}_{\boldsymbol{a}}^*, \boldsymbol{u}_{\boldsymbol{b}}^*)$$

and we also define

$$E_{\boldsymbol{a}} = \bigcup_{\boldsymbol{b} \in \mathcal{B}} E_{\boldsymbol{a},\boldsymbol{b}}$$

(notice that this is an union of disjoint events). If $n_i = 0$, we simply let $E_{\boldsymbol{a}} = \{\Pi_N(\boldsymbol{c}_{\boldsymbol{a}}^*) = \boldsymbol{u}^*\} = G_N(\boldsymbol{c}_{\boldsymbol{a}}^*, \boldsymbol{u}^*)$.

Remark 3.5 Clearly $\pi_N \in E_{\boldsymbol{a},\boldsymbol{b}}$ implies $Q_{\max}(\pi_N) \geq \mathrm{P}(\boldsymbol{0} \to \boldsymbol{x}_{\boldsymbol{b}}^*) = q^*$. Hence,

$$\mathbb{P}\big(Q_{\max}(\Pi_N) \geq q^*\big) \geq \mathbb{P}\Big(\bigcup_{\boldsymbol{a} \in \mathcal{A}^{n_o}} E_{\boldsymbol{a}}\Big).$$

■

Our aim is now to estimate this last probability, using:

$$\mathbb{P}\Big(\bigcup_{\boldsymbol{a} \in \mathcal{A}^{n_o}} E_{\boldsymbol{a}}\Big) \geq \sum_{\boldsymbol{a} \in \mathcal{A}^{n_o}} \mathbb{P}(E_{\boldsymbol{a}}) - \sum_{\substack{\boldsymbol{a},\boldsymbol{a}' \in \mathcal{A}^{n_o} \\ \boldsymbol{a} \neq \boldsymbol{a}'}} \mathbb{P}(E_{\boldsymbol{a}} \cap E_{\boldsymbol{a}'})$$

We will prove a lower bound for the first term (Lemma 3.2) and an upper bound for the second term (Lemma 3.3).

Lemma 3.2 With the convention $|\mathcal{B}| = 1$ if $n_i = 0$,

$$\sum_{\boldsymbol{a} \in \mathcal{A}^{n_o}} \mathbb{P}(E_{\boldsymbol{a}}) \geq |\mathcal{A}|^{n_o} \, |\mathcal{B}|^{n_i} \, \frac{1}{[|Y|r(N+\nu_o)]^{|\boldsymbol{h}|}}$$

□

Proof:

$$\mathbb{P}(E_{\boldsymbol{a}}) = \frac{|E_{\boldsymbol{a}}|}{|G_N|} = \frac{|\mathcal{B}|^{n_i} \, |G_N(\boldsymbol{c}^*, \boldsymbol{u}^*)|}{|G_N|}$$

By Remark 3.1 and Lemma 2.1,

$$\frac{|G_N(\boldsymbol{c}^*, \boldsymbol{u}^*)|}{|G_N|} = \frac{1}{|Y_{\boldsymbol{h}}^{r(N+\nu_o)}|} \geq \frac{1}{[|Y|r(N+\nu_o)]^{|\boldsymbol{h}|}}$$

■

Lemma 3.3 If $\mu \geq 2$:

$$\sum_{\substack{\boldsymbol{a},\boldsymbol{a}' \in \mathcal{A}^{n_o} \\ \boldsymbol{a} \neq \boldsymbol{a}'}} \mathbb{P}(E_{\boldsymbol{a}} \cap E_{\boldsymbol{a}'}) \leq |\mathcal{A}|^{2n_o} |\mathcal{B}|^{2n_i} \left(\frac{2e|\boldsymbol{h}|}{r(N+\nu_o)} \right)^{2|\boldsymbol{h}|}$$

while if $\mu = 1$:

$$\sum_{\substack{\boldsymbol{a},\boldsymbol{a}' \in \mathcal{A}^{n_o} \\ \boldsymbol{a} \neq \boldsymbol{a}'}} \mathbb{P}(E_{\boldsymbol{a}} \cap E_{\boldsymbol{a}'}) < \frac{1}{|Y_{\boldsymbol{h}}^{r(N+\nu_o)}|}$$

□

Proof: We have

$$E_{\boldsymbol{a}} \cap E_{\boldsymbol{a}'} = \bigcup_{\boldsymbol{b},\boldsymbol{b}' \in \mathcal{B}^{n_i}} (E_{\boldsymbol{a},\boldsymbol{b}} \cap E_{\boldsymbol{a}',\boldsymbol{b}'}) \ .$$

Consequently, by the union bound,

$$\sum_{\substack{\boldsymbol{a},\boldsymbol{a}' \in \mathcal{A}^{n_o} \\ \boldsymbol{a} \neq \boldsymbol{a}'}} \mathbb{P}(E_{\boldsymbol{a}} \cap E_{\boldsymbol{a}'}) \leq \sum_{\substack{\boldsymbol{a},\boldsymbol{a}' \in \mathcal{A}^{n_o} \\ \boldsymbol{a} \neq \boldsymbol{a}'}} \sum_{\boldsymbol{b},\boldsymbol{b}' \in \mathcal{B}^{n_i}} \mathbb{P}(E_{\boldsymbol{a},\boldsymbol{b}} \cap E_{\boldsymbol{a}',\boldsymbol{b}'})$$

Now we need to deal with $\mathbb{P}(E_{\boldsymbol{a},\boldsymbol{b}} \cap E_{\boldsymbol{a}',\boldsymbol{b}'})$. First of all note that if there is an incomplete error event in $\boldsymbol{u}^*$, surely $\mathbb{P}(E_{\boldsymbol{a},\boldsymbol{b}} \cap E_{\boldsymbol{a}',\boldsymbol{b}'}) = 0$ for all $\boldsymbol{a} \neq \boldsymbol{a}'$ and for all $\boldsymbol{b}$,$\boldsymbol{b}'$. Now consider the case of only regular events. Fix any $\boldsymbol{a} \neq \boldsymbol{a}'$ and $\boldsymbol{b}$,$\boldsymbol{b}'$ such that there exists $\pi \in E_{\boldsymbol{a},\boldsymbol{b}} \cap E_{\boldsymbol{a}',\boldsymbol{b}'}$. By the definition of $\boldsymbol{c}^*_{\boldsymbol{a}}$ and $\boldsymbol{c}^*_{\boldsymbol{a}'}$, we can find outer codewords $\tilde{\boldsymbol{c}}^*$, $\tilde{\boldsymbol{c}}^*_{\boldsymbol{a}}$, $\tilde{\boldsymbol{c}}^*_{\boldsymbol{a}'}$ (possibly $\tilde{\boldsymbol{c}}^* = \boldsymbol{0}$) having disjoint supports, each consisting of some of the error events $\boldsymbol{c}^*_k$, such that $\boldsymbol{c}^*_{\boldsymbol{a}} = \tilde{\boldsymbol{c}}^* + \tilde{\boldsymbol{c}}^*_{\boldsymbol{a}}$ and $\boldsymbol{c}^*_{\boldsymbol{a}'} = \tilde{\boldsymbol{c}}^* + \tilde{\boldsymbol{c}}^*_{\boldsymbol{a}'}$. More precisely, letting $\tilde{n}_o = \mathrm{d_H}(\boldsymbol{a},\boldsymbol{a}')$, i.e. the number of i's such that $\boldsymbol{a}_i \neq \boldsymbol{a}'_i$, $\tilde{\boldsymbol{c}}^*$ consists of $n_o - \tilde{n}_o$ error events, and $\tilde{\boldsymbol{c}}^*_{\boldsymbol{a}}$ consists of $\tilde{n}_o$ error events and is shift equivalent to $\tilde{\boldsymbol{c}}^*_{\boldsymbol{a}'}$. Clearly, $\boldsymbol{w}_G(\tilde{\boldsymbol{c}}^*_{\boldsymbol{a}}) = \boldsymbol{w}_G(\tilde{\boldsymbol{c}}^*_{\boldsymbol{a}'}) = \boldsymbol{h} - \boldsymbol{w}_G(\tilde{\boldsymbol{c}}^*)$.

Similarly, we can find inner input words $\tilde{\boldsymbol{u}}^*$, $\tilde{\boldsymbol{u}}^*_{\boldsymbol{b}}$, $\tilde{\boldsymbol{u}}^*_{\boldsymbol{b}'}$ (possibly $\tilde{\boldsymbol{u}}^* = \boldsymbol{0}$ or $\tilde{\boldsymbol{u}}^*_{\boldsymbol{b}} = \tilde{\boldsymbol{u}}^*_{\boldsymbol{b}'} = \boldsymbol{0}$) having disjoint supports, each consisting of some of the input error events $\boldsymbol{u}^*_k$, such that $\boldsymbol{u}^*_{\boldsymbol{b}} = \tilde{\boldsymbol{u}}^* + \tilde{\boldsymbol{u}}^*_{\boldsymbol{b}}$ and $\boldsymbol{u}^*_{\boldsymbol{b}'} = \tilde{\boldsymbol{u}}^* + \tilde{\boldsymbol{u}}^*_{\boldsymbol{b}'}$. Letting $\tilde{n}_i = \mathrm{d_H}(\boldsymbol{b},\boldsymbol{b}')$, $\tilde{\boldsymbol{u}}^*$ has $n_i - \tilde{n}_i$ error events and $\tilde{u}^*_{\boldsymbol{b}}$ has $\tilde{n}_i$ error events and is shift equivalent to $\tilde{\boldsymbol{u}}^*_{\boldsymbol{b}'}$. Clearly, $\boldsymbol{w}_G(\tilde{\boldsymbol{u}}^*_{\boldsymbol{b}}) = \boldsymbol{w}_G(\tilde{\boldsymbol{u}}^*_{\boldsymbol{b}'}) = \boldsymbol{h} - \boldsymbol{w}_G(\tilde{\boldsymbol{u}}^*)$.

As a consequence of Lemma 3.1, if $\pi \in E_{\boldsymbol{a},\boldsymbol{b}} \cap E_{\boldsymbol{a}',\boldsymbol{b}'}$, then $\pi(\tilde{\boldsymbol{c}}^*) = \tilde{\boldsymbol{u}}^*$, $\pi(\tilde{\boldsymbol{c}}^*_{\boldsymbol{a}}) = \tilde{\boldsymbol{u}}^*_{\boldsymbol{b}}$ and $\pi(\tilde{\boldsymbol{c}}^*_{\boldsymbol{a}'}) = \tilde{\boldsymbol{u}}^*_{\boldsymbol{b}'}$. This implies that $\boldsymbol{w}_G(\tilde{\boldsymbol{u}}^*) = \boldsymbol{w}_G(\tilde{\boldsymbol{c}}^*)$ and that $\boldsymbol{w}_G(\tilde{\boldsymbol{u}}^*_{\boldsymbol{b}}) = \boldsymbol{w}_G(\tilde{\boldsymbol{u}}^*_{\boldsymbol{b}'}) = \boldsymbol{w}_G(\tilde{\boldsymbol{c}}^*_{\boldsymbol{a}}) = \boldsymbol{w}_G(\tilde{\boldsymbol{c}}^*_{\boldsymbol{a}'}) = \boldsymbol{h} - \boldsymbol{w}_G(\tilde{\boldsymbol{c}}^*)$. We will use the notation $\tilde{\boldsymbol{h}} = \boldsymbol{w}_G(\tilde{\boldsymbol{u}}^*_{\boldsymbol{b}})$. Note that if $\mathbb{P}\big(E_{\boldsymbol{a},\boldsymbol{b}} \cap E_{\boldsymbol{a}',\boldsymbol{b}'}\big) \neq 0$ and $(\boldsymbol{a},\boldsymbol{b}) \neq (\boldsymbol{a}',\boldsymbol{b}')$ then surely both $\boldsymbol{a} \neq \boldsymbol{a}'$ and $\boldsymbol{b} \neq \boldsymbol{b}'$. Also note that

$$\mathbb{P}\big(E_{\boldsymbol{a},\boldsymbol{b}} \cap E_{\boldsymbol{a}',\boldsymbol{b}'}\big) \leq \mathbb{P}\big(\mathit{\Pi}_N(\tilde{\boldsymbol{c}}^* + \tilde{\boldsymbol{c}}^*_{\boldsymbol{a}} + \tilde{\boldsymbol{c}}^*_{\boldsymbol{a}'}) = \tilde{\boldsymbol{u}}^* + \tilde{\boldsymbol{u}}^*_{\boldsymbol{b}} + \tilde{\boldsymbol{u}}^*_{\boldsymbol{b}'}\big) = \frac{1}{Y_{\boldsymbol{h}+\tilde{\boldsymbol{h}}}^{r(N+\nu_o)}}$$

In the simple case when $\tilde{\boldsymbol{h}} = \boldsymbol{h}$, this gives

$$\mathbb{P}(E_{\boldsymbol{a},\boldsymbol{b}} \cap E_{\boldsymbol{a}',\boldsymbol{b}'}) \leq= \frac{1}{|Y_{2\boldsymbol{h}}^{r(N+\nu_o)}|} \leq \left(\frac{2e|\boldsymbol{h}|}{r(N+\nu_o)}\right)^{2|\boldsymbol{h}|}$$

where the last line comes from Lemma 2.1.

Now notice that $\tilde{\boldsymbol{h}} \in H$ and $\boldsymbol{h} - \tilde{\boldsymbol{h}} \in H \cup \{\boldsymbol{0}\}$, so that

$$1 + |\tilde{\boldsymbol{h}}| - \tilde{n}_o - \tilde{n}_i \geq f(\tilde{\boldsymbol{h}}) \geq \mu = 1 + |\boldsymbol{h}| - n_o - n_i \tag{3.5}$$

and, if $\tilde{\boldsymbol{h}} \neq \boldsymbol{h}$,

$$1 + |\boldsymbol{h} - \tilde{\boldsymbol{h}}| - (n_o - \tilde{n}_o) - (n_i - \tilde{n}_i) \geq f(\boldsymbol{h} - \tilde{\boldsymbol{h}}) \geq \mu = 1 + |\boldsymbol{h}| - n_o - n_i\,. \tag{3.6}$$

Equations (3.5) and (3.6) together are possible only in the case when $\mu = 1$. So, for $\mu \geq 2$, surely $\boldsymbol{h} = \tilde{\boldsymbol{h}}$, and this ends the proof:

$$\sum_{\boldsymbol{a},\boldsymbol{a}' \in \mathcal{A}^{n_o}, \boldsymbol{a} \neq \boldsymbol{a}'} \sum_{\boldsymbol{b},\boldsymbol{b}' \in \mathcal{B}^{n_i}, \boldsymbol{b} \neq \boldsymbol{b}'} \mathbb{P}(E_{\boldsymbol{a},\boldsymbol{b}} \cap E_{\boldsymbol{a}',\boldsymbol{b}'}) \leq |\mathcal{A}|^{2n_o} |\mathcal{B}|^{2n_i} \left(\frac{2e|\boldsymbol{h}|}{r(N+\nu_o)}\right)^{2|\boldsymbol{h}|}$$

For $\mu = 1$, instead, note that in this case $\tilde{n}_0 + \tilde{n}_i = |\tilde{\boldsymbol{h}}|$ and $n_0 + n_i = |\boldsymbol{h}|$. We can estimate:

$$\begin{aligned}
\sum_{\substack{\boldsymbol{a},\boldsymbol{a}' \in \mathcal{A}^{n_o} \\ \boldsymbol{a} \neq \boldsymbol{a}'}} \mathbb{P}(E_{\boldsymbol{a}} \cap E_{\boldsymbol{a}'}) &= \sum_{1 \leq \tilde{n}_o \leq n_o} \sum_{\substack{\boldsymbol{a},\boldsymbol{a}' \in \mathcal{A}^{n_o} \\ 1 \leq \mathrm{d_H}(\boldsymbol{a},\boldsymbol{a}') \leq \tilde{n}_o}} \sum_{1 \leq \tilde{n}_i \leq n_i} \sum_{\substack{\boldsymbol{b},\boldsymbol{b}' \in \mathcal{B}^{n_i} \\ 1 \leq \mathrm{d_H}(\boldsymbol{b},\boldsymbol{b}') \leq \tilde{n}_i}} \mathbb{P}(E_{\boldsymbol{a},\boldsymbol{b}} \cap E_{\boldsymbol{a}',\boldsymbol{b}'}) \\
&\leq \sum_{1 \leq \tilde{n}_o \leq n_o} \binom{n_o}{\tilde{n}_o} |\mathcal{A}|^{n_o + \tilde{n}_o} \sum_{1 \leq \tilde{n}_i \leq n_i} \binom{n_i}{\tilde{n}_i} |\mathcal{B}|^{n_i + \tilde{n}_i} \left(\frac{e(|\boldsymbol{h}| + \tilde{n}_o + \tilde{n}_i)}{r(N+\nu_o)}\right)^{|\boldsymbol{h}| + \tilde{n}_o + \tilde{n}_i} \\
&\leq \frac{|\mathcal{A}|^{n_o} |\mathcal{B}|^{n_i}}{r(N+\nu_o)^{|\boldsymbol{h}|}} 2e|\boldsymbol{h}| \left[\left(1 + \frac{|\mathcal{A}|2e|\boldsymbol{h}|}{r(N+\nu_o)}\right)^{n_o} - 1\right] \left[\left(1 + \frac{|\mathcal{B}|2e|\boldsymbol{h}|}{r(N+\nu_o)}\right)^{n_i} - 1\right] \\
&< \frac{|\mathcal{A}|^{n_o} |\mathcal{B}|^{n_i}}{r(N+\nu_o)^{|\boldsymbol{h}|}} \frac{1}{|Y_{\boldsymbol{h}}^{r(N+\nu_o)}|}
\end{aligned}$$

where the last line is due to the suitable choice of $l_{\max}$ and $\lambda_{\max}$ for the case $\mu = 1$, ensuring that $|\mathcal{A}|$ and $|\mathcal{B}|$ are small enough. ■

Now we can conclude the proof of Prop. 3.10. Using Lemmas 3.2 and 3.3, for $\mu \geq 2$ we get

$$\sum_{\boldsymbol{a} \in \mathcal{A}^{n_o}} \mathbb{P}(E_{\boldsymbol{a}}) - \sum_{\substack{\boldsymbol{a},\boldsymbol{a}' \in \mathcal{A}^{n_o} \\ \boldsymbol{a} \neq \boldsymbol{a}'}} \mathbb{P}(E_{\boldsymbol{a}} \cap E_{\boldsymbol{a}'}) \geq \frac{|\mathcal{A}|^{n_o} |\mathcal{B}|^{n_i}}{[r(N+\nu_o)]^{|\boldsymbol{h}|}} \left(\frac{1}{|Y|^{|\boldsymbol{h}|}} - |\mathcal{A}|^{n_o} |\mathcal{B}|^{n_i} \frac{(2e|\boldsymbol{h}|)^{2|\boldsymbol{h}|}}{[r(N+\nu_o)]^{|\boldsymbol{h}|}}\right)$$

For $N \to \infty$, as $|\mathcal{A}| \asymp N$ and $|\mathcal{B}| \asymp N$, we have $\frac{|\mathcal{A}|^{n_o}\,|\mathcal{B}|}{[r(N+\nu_o)]^{|\boldsymbol{h}|}} \asymp N^{-\mu+1}$. We conclude the proof by noticing that $|\mathcal{A}|^{n_o}\,|\mathcal{B}|^{n_i}\frac{(2e|\boldsymbol{h}|)^{2|\boldsymbol{h}|}}{[r(N+\nu_o)]^{|\boldsymbol{h}|}} \asymp N^{-\mu+1} \to 0$.

For $\mu = 1$, Lemmas 3.2 and 3.3, together with the remark that $n_o + n_i = |\boldsymbol{h}|$, give that $\sum_{\boldsymbol{a}\in\mathcal{A}^{n_o}} \mathbb{P}(E_{\boldsymbol{a}}) - \sum_{\substack{\boldsymbol{a},\boldsymbol{a}'\in\mathcal{A}^{n_o}\\ \boldsymbol{a}\neq\boldsymbol{a}'}} \mathbb{P}(E_{\boldsymbol{a}}\cap E_{\boldsymbol{a}'})$ is bounded from below by a strictly positive constant.

3.3.3 Proof of Propositions 3.1, 3.2 and 3.3

We prove here Prop. 3.2 and Prop. 3.3; clearly the latter one also implies the weaker Prop. 3.1, which can be obtained as a special case just taking $\rho_2 = 0$ so that $\boldsymbol{w}_G = \boldsymbol{w}_1$ and $d_f^o = d_{f,1}^o$.

Lemma 3.4 Under the same assumptions as in Prop. 3.3, for all $\boldsymbol{h} = (\boldsymbol{h}_1,\boldsymbol{h}_2) \in H \subseteq \mathbb{N}^{\rho_1} \times \mathbb{N}^{\rho_2}$:

- $1 \le n_o(\boldsymbol{h}) \le \lfloor |\boldsymbol{h}_1|/d_{f,1}^o \rfloor$;
- $0 \le n_i(\boldsymbol{h}) \le \lfloor |\boldsymbol{h}_1|/2 \rfloor + |\boldsymbol{h}_2|$;
- $1 + |\boldsymbol{h}_1| - \lfloor |\boldsymbol{h}_1|/d_{f,1}^o \rfloor - \lfloor |\boldsymbol{h}_1|/2 \rfloor \le f(\boldsymbol{h}) \le |\boldsymbol{h}|$ □

Proof: The upper bounds for $n_o(\boldsymbol{h})$ and $n_i(\boldsymbol{h})$ are an immediate consequence of the definition of $d_{f,1}^o$ and of the $\boldsymbol{w}_1$–recursiveness of ϕ_i. For the lower bounds, see Remark 3.2. Then, the estimations for $f(\boldsymbol{h})$ directly follow. ■

The definitions of d_f^o and H now clearly imply that $\mu \le d_f^o$, while the lower bound for μ comes from the following property.

Proposition 3.11 Given any constant $c \ge 2$, the function $g : \mathbb{N} \to \mathbb{N}$ defined by $g_c(h) = 1 + h - \lfloor h/c \rfloor - \lfloor h/2 \rfloor$ and restricted to $h \ge c$ has minimum value $\lfloor (c+1)/2 \rfloor$ and

$$\arg\min_{h\ge c} g_c(h) = \begin{cases} 2\mathbb{N}^* & \text{if } c = 2; \\ \{c, c+1, 2c\} & \text{if } c = 3; \\ \{c, c+1\} & \text{if } c \text{ is odd, } c \ge 5; \\ \{c\} & \text{if } c \text{ is even, } c \ge 4. \end{cases}$$

□

Proof:

Step 1: consider $g_c(h)$ restricted to multiples of c: $h = ac$ where a varies in $\mathbb{N}^*$. Then

$$g_c(h) = \begin{cases} 1 + \left(\frac{c}{2} - 1\right) a & \text{if } ac \text{ is even} \\ \frac{3}{2} + \left(\frac{c}{2} - 1\right) a & \text{if } ac \text{ is odd} \end{cases}$$

The minimum is obtained:

- for any $a \in \mathbb{N}^*$ if $c = 2$;
- for $a = 1$ and $a = 2$ if $c = 3$;
- only for $a = 1$ if $c > 3$.

Step 2: for any fixed $a \in \mathbb{N}^*$, consider $g_c(h)$ restricted to $h \in [ac,(a+1)c)$. In this case, $g_c(h) = 1 - a + \lfloor (h+1)/2 \rfloor$. The minimum is then obtained for $h = ac$ if ac is even, for $h = ac$ and also for $h = ac + 1$ if ac is odd.

Step 3: combine steps 1 and 2 to obtain the complete list of h's minimizing $g_c(h)$; notice that this always includes $h = c$ and then

$$\min_{h \geq g} g_c(h) = g_c(c) = \lfloor (c+1)/2 \rfloor .$$

■

The estimations for $f(\boldsymbol{h})$ in Lemma 3.4 can be re-written as $f(\boldsymbol{h}) \geq g_{d^o_{f,1}}(|\boldsymbol{h}_1|)$. For all $\boldsymbol{h} \in H$, clearly $|\boldsymbol{h}_1| \geq d^o_{f,1}$ and so, by Prop. 3.11,

$$\mu = \min_{\boldsymbol{h} \in H} f(\boldsymbol{h}) \geq \lfloor (d^o_{f,1} + 1)/2 \rfloor .$$

Clearly this lower bound for μ immediately means that $d^o_{f,1} \geq 2$ gives $\mu \geq 1$ and $d^o_{f,1} \geq 3$ gives $\mu \geq 2$.

Finally we prove that $\mathcal{H}$ is a finite set, under the assumption that $\rho_2 = 0$ and $d^o_f \geq 3$. For any $\boldsymbol{h} \in \mathcal{H}$, i.e. such that $f(\boldsymbol{h}) = \mu$, by Lemma 3.4 we get:

$$\mu = f(\boldsymbol{h}) \geq 1 + |\boldsymbol{h}| - \lfloor |\boldsymbol{h}|/d^o_f \rfloor - \lfloor |\boldsymbol{h}|/2 \rfloor \geq 1 + |\boldsymbol{h}| \left(\tfrac{1}{2} - \tfrac{1}{d^o_f}\right)$$

which gives $|\boldsymbol{h}| \leq (\mu - 1)\frac{2d^o_f}{d^o_f - 2}$, ending the proof.

3.4 Examples

In this section we consider particular cases, where we can characterize μ and q^* exactly or we can give tighter bounds than the general ones. We will particularly focus on the relevant examples introduced in Sect. 3.1.3. Throughout this section, we will consider $\Gamma = \mathbb{Z}_m$; in some cases we will restrict our attention to m-PSK–AWGN channels.

3.4.1 Classical free $\mathbb{Z}_m$ serial scheme

We call this scheme classical, because it is the simplest and the most natural generalization of the classical binary serial concatenations introduced in [3].

In the general scheme, take $U = \mathbb{Z}_m^k$, $Y = \mathbb{Z}_m$, $\Gamma = \mathbb{Z}_m$, and consider constituent encoders which are rational matrices $\phi_o \in \mathbb{Z}_m(D)^{k\times r}$ and $\phi_i \in \mathbb{Z}_m(D)^{s\times l}$. See Appendix 2.4 for properties of convolutional encoders in this particular setting.

Consider symbol error probability with respect to Hamming weight on $\mathbb{Z}_m$ (extended component-wise). Take as interconnection group $G_N = S_{r(N+\nu_o)}$, i.e. all the permutations moving around the elements of $\mathbb{Z}_m$. Clearly the invariant weight $\boldsymbol{w}_G$ will be the type weight $\boldsymbol{w}_{\mathrm{T}}$ on $\mathbb{Z}_m$ (extended component-wise). Notice that in this scheme, we can think 'symbols' in the most intuitive way, i.e. to be the elements of $\mathbb{Z}_m$, both in input, in the interconnection and at the output. Clearly, if you take $m = 2$, symbols are just bits, type weight and Euclidean weight are equal to Hamming weight and so we get the classical binary schemes introduced in [3].

For this ensemble, we have an explicit expression for μ if m is a power of 2, and tight bounds for μ for general m; we also have simple examples showing that, without more information about the constituent encoders, nothing tighter than these bounds can be found.

Let $m = p_1^{\alpha_1} \dots p_l^{\alpha_l}$ be the prime factors decomposition of m and let $\phi_{j,o} : \mathbb{Z}_{p_j}^{k\mathbb{N}} \to \mathbb{Z}_{p_j}^{r\mathbb{N}}$ be obtained by taking the restriction of ϕ_o to inputs in $\frac{m}{p_j}\mathbb{Z}_m^k$ and then identifying $\frac{m}{p_j}\mathbb{Z}_m$ with $\mathbb{Z}_{p_j}$ through the natural fields isomorphism. With this notation, the following bounds for μ hold true.

Proposition 3.12 For the classical free $\mathbb{Z}_m$ ensemble,

$$\lfloor (d_f^o + 1)/2 \rfloor \le \mu \le d_f^o - \lfloor d_f^o / p_{\min} \rfloor$$

where $p_{\min} = \min\{p_j : d_f^o = d_f(\phi_{j,o})\}$. □

Proof: We already have the bound $\lfloor (d_f^o + 1)/2 \rfloor \le \mu \le d_f^o$ (Prop. 3.1), so we just need to prove the tighter upper bound.

Notice that, by Prop. 2.7, $\mathcal{P} := \{p_j : d_f^o = d_f(\phi_{j,o})\} \neq \emptyset$. We want to prove that $\mu \le d_f^o - \lfloor d_f^o/p \rfloor$ for all $p \in \mathcal{P}$. So, fix any $p \in \mathcal{P}$; consider a word $\boldsymbol{c} \in \frac{m}{p}\phi_o^N(\mathbb{Z}_m^{kN})$ such that $\mathrm{w_H}(\boldsymbol{c}) = d_f^o$; let $\boldsymbol{h} = \boldsymbol{w}_{\mathrm{T}}(\boldsymbol{c})$.

Let $a_1, \dots, a_{d_f^o} \in \frac{m}{p}\mathbb{Z}_m \setminus \{0\}$ be the non-zero symbols of the word $\boldsymbol{c}$ (possibly with the same symbol repeated many times). Consider $a_1, \dots, a_p$: by Lemma 2.2 (applied to $\mathbb{Z}_p$), there exist indexes $\{j_1, \dots, j_n\} \subseteq \{1, \dots, p\}$ such that $a_{j_1} + \dots, a_{j_n} = 0 \mod m$. Then, by Prop. 2.3, there exist distinct times $t_1, \dots, t_n$ such that $\phi_i(a_{j_1} D^{t_1} + \dots + a_{j_n} D^{t_n})$ has finite support, i.e. is formed by some (at least one) error events. By applying the same argument to $a_{p+1}, \dots, a_{2p}$ and so on up to $a_{(\lfloor d_f^o/p \rfloor - 1)p+1}, \dots, a_{\lfloor d_f^o/p \rfloor p}$, we obtain that $n_i(\boldsymbol{h}) \ge \lfloor d_f^o/p \rfloor$. Clearly $n_o(\boldsymbol{h}) = 1$ and so

we can conclude: $\mu \leq f(\boldsymbol{h}) \leq 1 + d_f^o - 1 - \lfloor d_f^o/p \rfloor$. ■

From Prop. 3.12, together with Prop. 2.7 (see Appendix 2.4), we get the exact value of μ for the case when m is a power of 2:

Corollary 3.1 For the classical free $\mathbb{Z}_m$ ensemble, if m is a power of 2,

$$\mu = \left\lfloor (d_f^o + 1)/2 \right\rfloor .$$

■

It is more difficult to get an explicit formula for q^*. We can just notice that whenever $\mu = \lfloor (d_f^o + 1)/2 \rfloor$ (so in particular for the classical free $\mathbb{Z}_m$ ensemble when m is a power of 2), the inequalities $n_o(\boldsymbol{h}) \leq \lfloor |\boldsymbol{h}|/d_f^o \rfloor$ and $n_i(\boldsymbol{h}) \leq \lfloor |\boldsymbol{h}|/2 \rfloor$ (Lemma 3.4) and Prop. 3.11, make simpler the description of $\mathcal{H}$. When $m = 2$, the description of $\mathcal{H}$ gets even simpler, because the type weight is a scalar, equal to the Hamming weight, and for all $h \in H$, $n_i(h) = \lfloor h/2 \rfloor$. This allows to find the following explicit formula $q^* = Q(d^*)$ in the binary classical ensemble (d^* was already described by Benedetto et al. [3], but here our result is more precise for odd values of d_f^o). Define $d_{f,2}^i$ and $d_{f,3}^i$ to be the minimum output Hamming weight of a regular error event of the inner encoder constrained to input Hamming weight 2 and 3 respectively ($d_{f,3} = +\infty$ if such an event does not exist). Also define $d_{1,\mathrm{term}}^i$ to be the minimum output Hamming weight of a terminated error event with input Hamming weight 1.

Proposition 3.13 For the binary classical ensemble, $q^* = Q(d^*)$, where:

- if d_f^o is even, $\quad d^* = \frac{1}{2} d_f^o d_{f,2}^i$;
- if d_f^o is odd ($d_f^o \geq 5$),

$$d^* = \begin{cases} \frac{d_f^o - 3}{2}\, d_{f,2}^i + \min\left\{ d_{f,2}^i + d_{1,\mathrm{term}}^i, d_{f,3}^i, 2d_{f,2}^i \right\} & \text{if } d_f^o + 1 \in H \\ \frac{d_f^o - 3}{2}\, d_{f,2}^i + \min\left\{ d_{f,2}^i + d_{1,\mathrm{term}}^i, d_{f,3}^i \right\} & \text{if } d_f^o + 1 \notin H \end{cases}$$

- if $d_f^o = 3$,

$$d^* = \begin{cases} \min\left\{ d_{f,2}^i + d_{1,\mathrm{term}}^i, d_{f,3}^i, 2d_{f,2}^i \right\} & \text{if } 4 \in H \\ \min\left\{ d_{f,2}^i + d_{1,\mathrm{term}}^i, d_{f,3}^i, 3d_{f,2}^i \right\} & \text{if } 4 \notin H \end{cases}$$

■

Now we give three examples of simple choices of the constituent encoders, for which we consider general m. We compute μ and then, for the computation of q^*, we consider the specific case of the m-PSK–AWGN channel, for which we can find

an explicit expression. Recall that for S–AWGN channel, given a type d, $Q(\boldsymbol{d}) = \frac{1}{2}\operatorname{erfc}\sqrt{(\sum_g d_g\omega_g)E_s/N_0}$ where $\omega_g = \|\theta(g)-\theta(0)\|^2/(4E_s)$; when S is m-PSK, $\omega_j = \sin^2(j\pi/m)$.

Example 3.1 [**Repeat-Accumulate codes**] The encoders are $\phi_o = \mathrm{Rep}_r$ (with $r \geq 2$) and $\phi_i = \frac{1}{1-D}$. We assume that the termination rule for the accumulator is the one that always brings to the zero state in one trellis step (using the input $-a$ if we are in state a).

We obtain $\mu = \min\{r-1, r-\lfloor r/p \rfloor\}$, where p is the smallest prime divisor of m, and $q^* = \frac{1}{2}\operatorname{erfc}\sqrt{d^*E_s/N_0}$ where:

- for $m = 2$, $d^* = \lfloor (r+1)/2 \rfloor$;
- for even $m \geq 4$,
$$d^* = \begin{cases} r\omega_1 & \text{if } r = 2 \text{ or } r = 3 \\ \lfloor (r+1)/2 \rfloor & \text{if } r \geq 4 \end{cases}$$
- for odd $m \geq 3$, let p be the smallest prime divisor of m and let $n = m/p$. Define
$$d_*(r,m) = \left\lfloor \tfrac{r}{p} \right\rfloor \sum_{i=1}^{p-1} \omega_{in} + \min_{1\leq j\leq p-1} \sum_{i=1}^{r \bmod p} \omega_{ijn \bmod p}$$
Then:
$$d^* = \begin{cases} r\omega_1 & \text{if } r < p \\ d_*(r,m) & \text{if } r \geq 2p \\ \min\{r\omega_1, d_*(r,m)\} & \text{if } p \leq r < 2p \end{cases}$$

Sketch of how to get this result:

- $m = 2$:
We can use the explicit expressions we have for μ and d^* in the binary case. For Rep_r, $d_f^o = r$ and $d_f^o + 1 \notin \mathcal{H}$; for the accumulator, $d_{f,2}^i = d_{1,\mathrm{term}}^i = 1$ and $d_{f,3}^i = +\infty$.
- even $m \geq 4$:
Notice that $d_f(\frac{m}{p_i}\mathrm{Rep}_r) = r$ for all prime $p_i | m$, so that, if $2|m$, by Prop. 3.12 $\mu = \lfloor (r+1)/2 \rfloor$. Then compute:
  - $H = (r\mathbb{N})^{m-1} \setminus \{\mathbf{0}\}$.
  - If $r \geq 4$, $\mathcal{H} = \{\boldsymbol{h} \in H : |\boldsymbol{h}| = r, n_o(\boldsymbol{h}) = 1, n_i(\boldsymbol{h}) = \lfloor |\boldsymbol{h}|/2 \rfloor\} = \{\boldsymbol{k}\}$ where $\boldsymbol{k}_{m/2} = r$ and $\boldsymbol{k}_i = 0$ for all $i \neq m/2$. This gives $d^* = \lfloor (r+1)/2 \rfloor \omega_{m/2}$. Notice that $\omega_{m/2} = 1$.

  - If $r = 3$, we have $\mathcal{H} = \{\boldsymbol{k}, 2\boldsymbol{k}, \boldsymbol{k}^{(1)}, \ldots, \boldsymbol{k}^{(m/2-1)}\}$ with $\boldsymbol{k}$ as above and $\boldsymbol{k}^{(j)}$ defined by $\boldsymbol{k}^{(j)}_j = \boldsymbol{k}^{(j)}_{m-j} = r$ and $\boldsymbol{k}^{(j)}_i = 0$ for all $i \notin \{j, m-j\}$. Then:

$$d^* = \min\{\lfloor (r+1)/2 \rfloor \omega_{m/2}, r\omega_1, \ldots, r\omega_m\} = \min\{2\omega_{m/2}, 3\omega_1\}$$

    Then $m \geq 4$ implies $\omega_1 \leq 1/2 = \omega_{m/2}/2$, so $d^* = 3\omega_1$.
  - If $r = 2$, $\mathcal{H} = \{\boldsymbol{h} \in H : \boldsymbol{h}_i = \boldsymbol{h}_{m-i} \forall i = 1, \ldots, m/2-1\}$ and, with the same reasonings as above, we find again: $d^* = \lfloor (r+1)/2 \rfloor$ for $m = 2$ and $d^* = d^*((2,0,\ldots,0,2)) = 2\mathrm{w_E}(\phi_i(1-D)) = 2\omega_1$ for $m \geq 4$.

- odd $m \geq 3$:
  - $H = (r\mathbb{N})^{m-1} \setminus \{\boldsymbol{0}\}$.
  - Compute:

$$\min_{\boldsymbol{h} \in H : |\boldsymbol{h}| = kr} = \begin{cases} 1 + kr - k - \frac{k}{2} r & \text{if } k \text{ even} \\ 1 + kr - k - \frac{k-1}{2} r - \lfloor r/p \rfloor & \text{if } k \text{ odd} \end{cases}$$

    Notice that both expressions are non decreasing in k, and increasing in k if $r \geq 3$, so that $\mu = \min\{r-1, r - \lfloor r/p \rfloor\}$
  - if $r = 2$, $\mu = 1$ and $\mathcal{H} = \{\boldsymbol{h} \in H : \boldsymbol{h}_i = \boldsymbol{h}_{m-i} \forall i = 1, \ldots, m/2-1\}$, so that $d^* = d^*(r\boldsymbol{e}_1 + r\boldsymbol{e}_{-1}) = r\omega_1$;
  - if $2 < r < p$, $\mu = r - 1$ and $\mathcal{H} = \{\boldsymbol{k}^{(1)}, \ldots, \boldsymbol{k}^{((m-1)/2)}\}$, the $\boldsymbol{k}^j$'s defined as for even m. So again $d^* = d^*(r\boldsymbol{e}_1 + r\boldsymbol{e}_{-1}) = r\omega_1$.
  - if $r \geq 2p$, $\mu = r - \lfloor r/p \rfloor$ and $\mathcal{H} = \{r\boldsymbol{e}_m, r\boldsymbol{e}_{2m/p}, \ldots, r\boldsymbol{e}_{(p-1)m/p}\}$, from which comes the expression for d^*
  - if $p \leq r < 2p$, $\mu = r - 1 = r - \lfloor r/p \rfloor$, so $\mathcal{H}$ is the union of the set $\mathcal{H}$ computed for $r < p$ and the one computed for $r \geq 2p$; thus, d^* is the minimum of the two values obtained before. □

The Repeat-Accumulate on $\mathbb{Z}_3$ ($r \geq 3$) is an example where the upper bound $\mu \leq d^o_f - \lfloor d^o_f/3 \rfloor$ is reached with equality. Now, we show another simple Repeat-Convolute code on $\mathbb{Z}_3$ such that the lower bound $\mu \geq \lfloor (d^o_f + 1)/2 \rfloor$ is reached with equality, showing that the bounds in Prop. 3.12 are the best possible for general m.

Example 3.2 Consider $m = 3$ and $\phi_o = \mathrm{Rep}_r$ ($r \geq 2$) and $\phi_i = 1/(1+D) = \sum_{t \geq 0} D^{2t} + 2D^{2t+1}$, with the termination rule that always bring to the zero state in one trellis step (i.e. if at time t the codeword has $c_t = a$, we terminate using the input $u_{t+1} = -a$ if t is even, a if t is odd).

Then, as for the Repeat-Accumulate, $H = r\mathbb{N}^2 \setminus \{(0,0)\}$ and given $\boldsymbol{h} = (rh_1, rh_2)$ we have $n_o(\boldsymbol{h}) = h_1 + h_2$. But now, when we look at the inner encoder to compute

$n_i(\boldsymbol{h})$, we find $n_i(\boldsymbol{h}) = \lfloor |\boldsymbol{h}|/2 \rfloor$, because all the following inputs produce a complete error event of ϕ_i: $\boldsymbol{u} = D^t + D^{t+1}$, $\boldsymbol{u} = 2D^t + 2D^{t+1}$ $\boldsymbol{u} = D^t + 2D^{t+2}$ and $\boldsymbol{u} = 2D^t + D^{t+2}$.

As a consequence,

$$\mu = \lfloor (r+1)/2 \rfloor$$

Let's compute $q^* = \frac{1}{2}\operatorname{erfc}\sqrt{d^* E_s/N_0}$ for this example. First of all we need $\mathcal{H}$:

$$\mathcal{H} = \begin{cases} H & \text{if } r = 2 \\ \{(r,0),(0,r),(r,r),(2r,0),(0,2r)\} & \text{if } r = 3 \\ \{(r,0),(0,r)\} & \text{if } r > 3 \end{cases}$$

Now consider that: $\mathrm{w_E}(\phi_i(D^t + D^{t+1})) = \mathrm{w_E}(D^t) = \omega_1 = 3/4$; analogously $\mathrm{w_E}(\phi_i(2D^t + 2D^{t+1})) = \omega_2 = 3/4$; while $\mathrm{w_E}(\phi_i(D^t + 2D^{t+2})) = \mathrm{w_E}(D^t + 2D^{t+1}) = \omega_1 + \omega_2 = 3/2$ and the same for $\mathrm{w_E}(\phi_i(2D^t + D^{t+2})) = 3/2$. Assuming that termination is always done in one single step, we also have that $\mathrm{w_E}(\phi_i^N(D^{N-1})) = \mathrm{w_E}(\phi_i^N(2D^{N-1})) = 3/4$. Finally, we get $d^* = \frac{3}{4}\left\lfloor \frac{r+1}{2} \right\rfloor$. □

Notice that for Repeat-Accumulate codes (Example 3.1) $\mu = \lfloor (r+1)/2 \rfloor$ for all even m. This is true for all Repeat-Convolute codes, by Prop. 3.12 together with the remark that $d_f(\frac{m}{p_i}\mathrm{Rep}_r) = r$ for all prime $p_i | m$. However, for a general outer encoder ϕ_o this is not true: the assumption that m is a power of two is essential in Coroll. 3.1, as shown by the following example.

Example 3.3 Let $m = 6$. Consider ϕ_o which is the following slight variation of a Repeat code: $\phi_o = [1,1,1,1,3]^T$. Let the inner encoder be the Accumulator $\phi_i = \frac{1}{1-D}$. For $p_1 = 2$ we have $\phi_{o,1} = [1,1,1,1,1]^T$, which has $d_f(\phi_{o,1}) = 5$, while for $p_2 = 3$ we have $\phi_{o,2} = [1,1,1,1,0]^T$, which has $d_f(\phi_{o,2}) = 4$, and so $d_f^o = d_f(\phi_{o,2}) = 4$. The bounds given in Prop. 3.12 give us $2 \le \mu \le 3$ and now we will show that $\mu = 3$. Notice that $f(\boldsymbol{h}) = \mu$ implies that $1 + |\boldsymbol{h}| - \frac{|\boldsymbol{h}|}{d_f^o} - \frac{|\boldsymbol{h}|}{2} \le \mu \le 3$ and then $|\boldsymbol{h}| \le 8$, so $\mathcal{H} \subseteq \{\boldsymbol{h} \in H : |\boldsymbol{h}| \le 8\}$. There are seven elements in H with $|\boldsymbol{h}| \le 8$. By computing $f(\boldsymbol{h})$ for the all of them, we get $\mu = 3$ and $\mathcal{H} = \{(0,4,0,0,0),(0,0,5,0,0),(0,0,0,4,0)\}$ and finally $q^* = \frac{1}{2}\operatorname{erfc}\sqrt{d^* E_s/N_0}$ with $d^* = 2\omega_2 + \omega_4 = 9/4$, reached when $\boldsymbol{h} = (0,4,0,0,0)$ and $\boldsymbol{h} = (0,0,0,4,0)$. □

3.4.2 Subgroups of permutations for the $\mathbb{Z}_m$ scheme

In the previous section, we have considered $\mathbb{Z}_m$-schemes

$$\xrightarrow{\mathbb{Z}_m^{kN}} \boxed{\phi_o^N} \xrightarrow{\mathbb{Z}_m^{r(N+\nu_o)}} \boxed{\pi_N} \xrightarrow{\mathbb{Z}_m^{sM_N}} \boxed{\phi_i^N} \xrightarrow{\mathbb{Z}_m^{l(M_N+\nu_i)}}$$

by taking $U = \mathbb{Z}_m^k$, $Y = \mathbb{Z}_m$, $\Gamma = \mathbb{Z}_m$ in the general serial scheme. However, we can also obtain some $\mathbb{Z}_m$ schemes by taking $Y = \mathbb{Z}_m^a$. Then, if we consider on $\mathbb{Z}_m^a$ a weight given by the component-wise extension of the type weight on $\mathbb{Z}_m$, we get again the same scheme as above. However, in this case we can also consider permutations moving around not single elements of $\mathbb{Z}_m$, but only the vectors in $\mathbb{Z}_m^a$, so that the invariant weight is the type weight on $\mathbb{Z}_m^a$. Or, on the contrary, we can consider a 'separate channels permutation': the invariant weight is $\boldsymbol{w} \in (\mathbb{N}^{m-1})^a$ given by the type weight on each separate component of $\mathbb{Z}_m^a$.

Even though these schemes are quite similar to the classical one, differing only for a restriction of the permutations to a subgroup of $S_{r(N+\nu_o)}$, Prop. 2.7 and Coroll. 3.1 do not hold true. We give here a simple example, for the binary case $m = 2$, and for the 'separate channels' permutation, where $\mu > \lfloor (d_f^o + 1)/2 \rfloor$.

Example 3.4 Consider the following outer and inner binary encoders:

$$\phi_o = \left[1, \tfrac{1}{1+D+D^3}\right]^T \qquad \phi_i = \begin{bmatrix} \frac{1}{1+D} & 0 \\ 0 & \frac{1}{1+D} \end{bmatrix}$$

and consider the 'separate channels permutation' ensemble (here $m = 2$ and $a = 2$ and so $\boldsymbol{w} \in \mathbb{N}^2$ is the Hamming weight of the two streams). The outer encoder has free distance $d_f^o = 4$ and all the words $\boldsymbol{c}$ of the outer code such that $\mathrm{d_H}(\boldsymbol{c}) = d_f^o$ are obtained when input is $1 + D + D^3$ or its shifts and have $\boldsymbol{w}(\boldsymbol{c}) = (3,1)$. The inner encoder is simply the rate-1 Accumulator, but acting separately on the two input streams.

We claim that for this scheme $\mu = 3 > \lfloor (d_f^o + 1)/2 \rfloor = 2$. In fact, we know that $\mu \geq \lfloor (d_f^o + 1)/2 \rfloor = 2$, where equality could be reached only if there was $\boldsymbol{h} \in H$ such that $|\boldsymbol{h}| = 4$, $n_o(|\boldsymbol{h}|) = 1$, $n_i(\boldsymbol{h}) = 2$, but this is not possible, as the only $\boldsymbol{h} \in H$ such that $|\boldsymbol{h}| = 4$ is $\boldsymbol{h} = (3,1)$, which has $n_o(\boldsymbol{h}) = 1$ but $n_i(\boldsymbol{h}) = 1$, giving $f(\boldsymbol{h}) = 3$ and so $\mu = 3$.

By an exhaustive listing of all small-weight codewords, we can also find $\mathcal{H}$, noting that $\boldsymbol{h} \in \mathcal{H}$ implies $|\boldsymbol{h}| \leq 8$, and then we can find $q^* = Q(3)$. □

Remark 3.6 The 'separate channels' ensemble is particularly interesting because it allows to include in our generalized serial concatenations also the most traditional parallel turbo codes (as it was already noticed e.g. in [2]): a turbo code with a parallel branches, each with an encoder ψ_j of rate k_j/n_j, can always be seen as Repeat-Convolute scheme, where $\phi_o = \mathrm{Rep}_r$ and $r = \sum k_j$, the interleaver acts separately on the a streams of $k_i \times N$ bits and ϕ_i is a block diagonal matrix, where the blocks are the ψ_j's. ■

3.4.3 Structured LDPC ensemble

For a description of these schemes, see Section 3.1.3. Here we give some statements about the parameters μ and d^*.

First of all, we have the following tight bounds for μ.

Proposition 3.14 For the structured LDPC ensemble,

$$\lfloor (c+1)/2 \rfloor \leq \mu \leq c - \lfloor c/p_{\min} \rfloor$$

where $p_{\min} = \min\{p_j \geq 2 : p_j | m\}$. □

By Prop. 3.3, we have $\mu \geq \lfloor (d^o_{f,1}+1)/2 \rfloor = \lfloor (c+1)/2 \rfloor$. The proof of the upper bound is similar to the proof of Prop. 3.12. Notice that here $p_{\min}$ is computed considering all prime factors of m because the outer encoder is a simple repetition code. ■

In particular, this Proposition implies that for all even m the interleaver gain is:

$$\mu = \lfloor (c+1)/2 \rfloor .$$

In the binary case ($m = 2$), we can also characterize q^*. In fact, we can easily describe $\mathcal{H}$:

$$\mathcal{H} = \begin{cases} \{(2w,w) : w \in \mathbb{N}^*\} & \text{if } c = 2 \\ \{(3,1),(6,2)\} & \text{if } c = 3 \\ \{(c,1)\} & \text{if } c \geq 4 \end{cases}$$

and then compute $q^* = Q(d^*)$:

$$d^* = \begin{cases} 1 & \text{if } c \text{ is even} \\ 2 & \text{if } c = 3 \\ 1 + \min\{d_{1,\text{term}}(\psi), d_{f,3}(\psi)\} & \text{if } c \text{ is odd, } c \geq 5 \end{cases}$$

where $d_{1,\text{term}}(\psi), d_{f,3}(\psi)$ are defined as $d^i_{1,\text{term}}, d^i_{f,3}$ in Prop. 3.13 but referring here to ψ instead of ϕ_i. If the inner encoder is truncated instead of terminated, $d^* = 2$ for all odd c.

Notice that the choice of ψ has almost no influence on d^*. This happens because pairs of bits which are repetition of a same information bit can be permuted by some interleaver in such a way that they are summed up by Sum_d, producing a zero output. The value of d^* is given by this worse case scenario. This remark suggests to consider interleavers with a better spread, enforcing the fact that 1's coming from the same error event of Rep_c cannot end up in positions where they would be summed up by Sum_d. However, the analysis of such a smaller ensemble, with a set of interleavers which is not a group, requires some different proof techniques, and

will be presented in Chapter 5.

For general m there is no explicit simple characterization of q^*, and neither there is one for all even m. We can just notice that, on m-PSK–AWGN channels, by the same argument used for $m = 2$, if both m and c are even then $q^* = \frac{1}{2}\operatorname{erfc}\sqrt{d^* E_s/N_0}$ with $d^* \leq \omega_{m/2} = 1$. This upper bound is achieved, for example, simply taking $\psi = 1/(1 - D)$. To see that this upper bound is not always achieved, take for example $m = 6$ and $\psi = 1/(1 + D)$: we have that $\psi(1 + D) = 1$ which is an error event of Euclidean weight $\omega_1 = 1/4$, so that for $c = 2$ or $c = 4$ we have $d^* = \frac{1}{4} + \frac{c}{2}\frac{1}{4} < 1$.

Chapter 4

Binary serial turbo ensembles: typical performance analysis

In this chapter we focus on the classical binary setting (binary serial turbo codes for binary-input output-symmetric channels) in order to find more refined probabilistic results in addition to the average error probability.

We investigate the typical behaviour of minimum distance and ML word error probability of a serial turbo concatenation with random interleaver, when the interleaver length N goes to infinity. Since the average-based analysis seemed to agree with simulation results in the sense that hierarchies of the design parameters were respected, it could be expected that a typical serial turbo code has an analogous behaviour, i.e. there is a concentration phenomenon. In this chapter we will show that in fact there is no concentration of the ML error probability around its average value, since the ratio $P(e)/\mathbb{E}[P(e)]$ converges to zero with probability one, thus showing that the average error probability is dominated by an asymptotically negligible fraction of bad interleavers. More precisely we shall prove that a typical sequence of serial turbo codes has error probability subexponentially decreasing to zero in N: with probability one the sequence $\log(-\log(P(e)))/\log N$ approaches an interval $[\alpha,\beta] \subset (0,1)$. The parameters α and β are increasing functions of the free distance of the outer encoder, which is thus confirmed as the main design parameter for these coding schemes, as was already suggested by the average-based analysis.

Our analysis is based on a precise estimation of the probability distribution of minimum distances, inspired both by the the tail estimations of [41] and the deterministic upper bounding techniques devised in [2]. A closer look to these bounds, with a careful estimation of the constants involved, allows to find also a design parameter for the inner encoder: its effective free distance, i.e. smallest weight of codewords corresponding to input weight two. This confirms the importance of this parameter that showed up in the analysis of the average error probability, but up to now had not been noticed in the study of minimum distance.

Our result has to be considered as analogous of the well known behaviour of ML-decoded LDPC codes (see [33], [48]): for the (c,d)-regular LDPC ensemble the average error probability is known to decrease to zero as $N^{1-c/2}$ for even c and N^{2-c} for odd c, while the error probability of a typical code goes to zero exponentially fast.

Our proofs rely on estimations of the weight enumerating coefficients of the constituents encoders which are tighter than those given in Section 2.3.5 in the case when the output weight is not constant with respect to N. We will discuss these bounds in Section 4.2; their proofs, based on techniques from [41], are specific for the binary case.

4.1 Problem setting

In this chapter, we consider the ensemble described in Section 3.1, but specialized to the most classical case: binary encoders and interleaver uniformly drawn among all permutation of the suitable length.

For the sake of clarity, we recall here the scheme, in the particular case we are now considering:

$$\xrightarrow{\mathbb{Z}_2^{kN}} \boxed{\phi_o^N} \xrightarrow{\mathbb{Z}_2^{r(N+\nu_o)}} \boxed{\pi_N} \xrightarrow{\mathbb{Z}_2^{sM_N}} \boxed{\phi_i^N} \xrightarrow{\mathbb{Z}_2^{l(M_N+\nu_i)}}$$

where

- $\phi_o : \mathbb{Z}_2^k(D) \to \mathbb{Z}_2^r(D)$;
- $\phi_i : \mathbb{Z}_2^s(D) \to \mathbb{Z}_2^l(D)$;
- ϕ_o is terminated after N trellis steps, with N such that s divides $r(N+\nu_o)$, obtaining
$$\phi_o^N : \mathbb{Z}_2^{kN} \to \mathbb{Z}_2^{r(N+\nu_o)}$$
- ϕ_i terminated after M_N trellis steps, with M_N such that $sM_N = r(N+\nu_o)$
$\phi_i^N : \mathbb{Z}_2^{sM_N} \to \mathbb{Z}_2^{l(M_N+\nu_i)}$
- the interleaver is a permutation $\pi_N \in S_{sM_N}$

In addition, we will also use the notation:

- $L_N := r(N+\nu_o) = sM_N$ the interleaver length
- $K_N := l(M_N+\nu_i) = l(\frac{r}{s}(N+\nu_o)+\nu_i)$ the blocklength.

From this binary serial scheme we get the classical binary 'uniform interleaver' ensemble by letting the interleaver Π_N be a random variable uniformly distributed on S_{L_N}.

In order to avoid extremely cumbersome notation, we will at first expose our results in full detail under some simplifying assumptions, and later (Section 4.5) we will discuss how most of the assumptions can be weakened.

So, from now on, we will assume:

- ϕ_o is non-catastrophic;
- ϕ_i is non-catastrophic and recursive;
- d_f^o is even;
- ϕ_i has scalar input ($s = 1$) and is proper rational, i.e. $\phi_i = \frac{1}{q(D)}[p_1(D), \ldots, p_l(D)]^T$ with $\deg(p_i) < \deg(q_i)$ for all i

although the only essential assumptions are non-catastrophicity (at least the concatenated non-catastrophicity of the serial scheme described in Section 3.2) and recursiveness of ϕ_i.

In most results we will assume $d_f^o \geq 3$, and in some also $d_f^o \geq 5$, we will comment on the way on this requirement.

Throughout the chapter we will have quantities depending on many parameters: w,d,N,n .... We will implicitly assume that the all the parameters are depending on N, but we will avoid heavy notation w_N,d_N .... So a statement such as '$f(w,d,N) = o(N^a)$ for $N \to \infty$, $d = o(N)$ and $w \leq d$' means that if $d = d_N$, $w = w_N$ satisfying $w_N \leq d_N$ and $d_N/N \to 0$ when $N \to \infty$, then $\lim_{N\to\infty} f(w_N,d_N,N)/N = 0$. When we say '$w$ is constant' we mean it does not depend on N.

4.2 Estimations of the weight enumerating coefficients of the constituent encoders

In this section, we present the bounds on the weight enumerating coefficients. As the proofs are long, we postpone them to Section 4.2.4.

4.2.1 Preliminaries

We recall here some notation and properties, adding some new definitions.

First of all, we remind two very important parameters:

- d_f^o is the free distance of the outer encoder, i.e. the minimum Hamming weight of its non-zero codewords

- d_2^i is the effective free distance of the inner encoder, i.e. the minimum Hamming weight among its codewords corresponding to input weight two.

We give here a slightly different version of Propositions 2.2 and 2.1 (from which it follows immediately).

Lemma 4.1 Given a non-catastrophic convolutional encoder, there exists a constant η such that any error event with output weight w has length not greater than ηw trellis steps

We will denote η_o such constant for ϕ_o and η_i the one for ϕ_i, or, more precisely, we will define η_o and η_i in such a way that any regular or terminating error event of ϕ_o^N and ϕ_i^N respectively has length bounded by η_o (resp. η_i) times the output weight. We will also use the notation $\mu_o = k\eta_o$ and $\mu_i = s\eta_i$, so that any regular or terminating error event of ϕ_o^N and ϕ_i^N respectively has input weight bounded from above by μ_o (resp. μ_i) times the output weight.

In the following sections, we will give estimations of some weight enumerating coefficients of ϕ_o^N, and ϕ_i^N, using techniques from [41]. We will use the notation:

- $A_d^{o,N}$ = number of codewords of ϕ_o^N with (output) weight d;
- $A_{w,\leq d}^{i,N}$ = number of codewords of ϕ_i^N with input weight w and output weight not greater than d;
- $R_{w,\leq d,n}^{i,N}$ = number of codewords of ϕ_i^N with input weight w and output weight not greater than d, consisting of exactly n regular error events, and no terminating event;
- $T_{w,\leq d,n}^{i,N}$ = number of codewords of ϕ_i^N with input weight w and output weight not greater than d, consisting of exactly $n-1$ regular error events, plus one terminating event

4.2.2 Outer encoder

For the outer encoder, we need only the following simple upper bound, which holds true for all non-catastrophic terminated convolutional encoders

Lemma 4.2 ([41], Lemma 3) If $\lfloor d/d_f^o \rfloor < N/2$,

$$A_d^{o,N} \leq 2^{(k\eta_o+\eta_o+1)d+1} \binom{N}{\lfloor d/d_f^o \rfloor}$$

In the particular case when $d = d_f^o$, also the following tighter estimation is true:

$$A_{d_f^o}^{o,N} \leq m_f^o N$$

where m_f^o is the number of different error events producing output weight d_f^o (all starting at time 0) □

Note: we know two estimations for m_f^o. One is the same used in the proof of this lemma, $m_f^o \leq 2^{kd_f^o\eta_o}$, the other (usually tighter) is $m_f^o \leq \binom{r\eta_o d_f^o}{d_f^o} \leq (er\eta_o)^{d_f^o}$.

4.2.3 Inner encoder

Using the recursiveness of ϕ_i, tighter bounds can be obtained, exploiting the limitation on the number of error events given by the restriction that each of them must have input weight at least two.

First of all we need the following well-known property, which is the binary version of Propositions 2.3 and 2.4

Lemma 4.3 Given ϕ a convolutional encoder with scalar input, there exists a constant $\delta \in \mathbb{N}^*$ such that $\mathrm{w_H}(\phi(1+D^\delta)) < \infty$. Moreover, if you denote by $\mathcal{D}$ the set of all such constants, and define $\bar{\delta} = \min \mathcal{D}$, you have $\mathcal{D} = \bar{\delta}\mathbb{N}^*$.

Moreover, the following inequality holds true $\mathrm{w_H}(\phi(1+D^{a\bar{\delta}})) \leq a\mathrm{w_H}(1+D^{a\bar{\delta}})$ and equality is guaranteed if $\phi(D)$ is proper, i.e. its numerator has strictly smaller degree than its denominator. In this case, if you define d_2 to be the smallest output weight when input weight is forced to be 2, you also have $d_2 = \mathrm{w_H}(1+D^{\bar{\delta}})$. □

Note that this property is trivially true, with $\delta = 1$, if ϕ is polynomial, and is interesting only when ϕ is recursive.

We will use the notation δ^i to denote $\bar{\delta}$ for the encoder ϕ_i and d_2^i to denote the effective free distance d_2 of ϕ_i.

Now, for scalar-input ϕ_i, define:

$$I_i = \inf_j \frac{\mathrm{w_H}\big(\phi_i(1+D^{j\delta^i})\big)}{j}$$

Remarks:

- As ϕ_i is recursive and has scalar-input, $\mathrm{w_H}\big(\phi_i(1+D^t)\big) < \infty$ if and only if $t = j\delta^i$, $j \in \mathbb{N}$.
- If $\phi_i(D) = \frac{p(D)}{q(D)}$ with $\deg p < \deg q$, then $\mathrm{w_H}\big(\phi_i(1+D^{j\delta^i})\big) = j\mathrm{w_H}\big(\phi_i(1+D^{\delta^i})\big)$. Proof:
  $(1+D^{j\delta^i})\frac{p(D)}{q(D)} = \sum_{t=0}^{j-1} D^{t\delta^i}(1+D^{\delta^i})\frac{p(D)}{q(D)}$. As $\deg p < \deg q \leq \delta^i$, the error events $D^{t\delta^i}(1+D^{\delta^i})\frac{p(D)}{q(D)}$ have disjoint supports, so that the weight of

$\sum_{t=0}^{j-1} D^{t\delta^i}(1+D^{\delta^i})\frac{p(D)}{q(D)}$ is the sum of the individual weights of the j error events, which are all equal to $\mathrm{w_H}\big(\phi_i(1+D^{\delta^i})\big)$.

- Clearly, the same holds also for $\phi_i(D) = \frac{1}{q(D)}[p_1(D),\ldots,p_l(D)]^T$ with $\deg p_j < \deg q\,\forall j$ (ϕ_i has scalar input and is proper rational)
- As a consequence, if ϕ_i has scalar input and is proper rational, then

$$I_i = \mathrm{w_H}\big(\phi_i(1+D^{\delta^i})\big) = d_2^i$$

Lemma 4.4 ([41], Lemma 1) Let $1 \le w \le \mu_i d$, $1 \le d \le K_N$.

- If w is even,

$$R^{i,N}_{w,\le d,w/2} \le \frac{(2e)^w}{w^w} M_N^{w/2} \left\lfloor \frac{d}{d_2^i} \right\rfloor^{w/2}$$

- If w is even, for $N \to \infty$, if $d = o(N)$,

$$A^{i,N}_{w,\le d} = R^{i,N}_{w,\le d,w/2} + o\left(\frac{C^w}{w^w} M_N^{w/2} \left\lfloor \frac{d}{d_2^i} \right\rfloor^{w/2} \right)$$

  where $C = 2e^3\sqrt{e(l+1)\eta_i}$.

- If w is odd, for $N \to \infty$, if $d = o(N)$,

$$A^{i,N}_{w,\le d} = O\left(\frac{C^w}{w^w} N^{\lfloor w/2 \rfloor} \left\lfloor \frac{d}{d_2^i} \right\rfloor^{\lfloor w/2 \rfloor - 1} d^2 \right)$$

  where $C = \max\left(2e^3\sqrt{(l+1)\eta_i}, \mu_i^2\sqrt{e}, 2e^2\eta_i\right)$ □

Lemma 4.5 ([41], Lemma 2) If w is even, $2 \le w \le \frac{N}{\eta_o}$ and $\frac{d_2^i w}{2} \le d \le \frac{d_2^i M_N}{2\delta^i}$

$$R^{i,N}_{w,\le d,w/2} \ge \binom{M_N - \delta^i \lfloor d/d_2^i \rfloor}{w/2} \binom{\lfloor d/d_2^i \rfloor}{w/2}$$

which implies also

$$R^{i,N}_{w,\le d,w/2} \ge \frac{2^{w/2}}{w^w} M_N^{w/2} \left\lfloor \frac{d}{d_2} \right\rfloor^{w/2}$$

□

Clearly $A^{i,N}_{w,\le d} \ge R^{i,N}_{w,\le d,w/2}$, so this lemma gives also a lower bound for $A^{i,N}_{w,\le d}$; looking at Lemma 4.4 you see that asymptotically it is a tight bound.

4.2.4 Proofs

Proof of Lemma 4.2 ([41], Lemma 3):

$$A_d^{o,N} = R_d^{o,N} + T_d^{o,N}$$

For the regular events, use the estimation

$$R_{(d_1,\dots,d_n)}^{o,N} \leq 2^{kd\eta_o} \binom{N}{n}$$

In fact, we are considering n error events, with lengths at most $d_1\eta_o$, ..., $d_n\eta_o$ respectively, so that the sum of their lengths is bounded by $d\eta_o$: if you consider the codewords restricted to these at most $d\eta_o$ trellis steps (removing the zeros corresponding to state zero–state zero transitions in between), you have at most $2^{kd\eta_o}$ words (the number of possible inputs for $d\eta_o$ trellis steps). Then, the starting position for n error events in between less then $N-n$ zero state–zero state transitions can be chosen in at most $\binom{N}{n}$ ways. So finally

$$\begin{aligned} R_d^{o,N} &= \sum_{n=1}^{\lfloor d/d_f^o \rfloor} \sum_{\substack{d_1,\dots,d_n: \\ \sum_i d_i = d, d_i \geq 1}} R_{(d_1,\dots,d_n)}^{o,N} \\ &\leq \sum_{n=1}^{d} \binom{d}{n} 2^{kd\eta_o} \binom{N}{\lfloor d/d_f^o \rfloor} \\ &\leq 2^{(k\eta_o+1)d} \binom{N}{\lfloor d/d_f^o \rfloor} \end{aligned}$$

where the last line uses the simple remark that $\sum_{n=0}^{d} \binom{d}{n} = 2^d$.

For the terminated events,

$$T_{(d_1,\dots,d_n)}^{o,N} \leq 2^{kd\eta_o} \binom{N}{n-1} d\eta_o$$

with a proof analogous to the previous, but considering that the $(n+1)$-th event, being terminated and having length at most $d\eta_o$, starts in a position between $N - d\eta_o$

and $N-1$ on the trellis. So,

$$\begin{aligned} T_d^{o,N} &= \sum_{n=1}^{\lceil d/d_f^o \rceil} \sum_{\substack{d_1,\ldots,d_n: \\ \sum_i d_i = d, d_i \geq 1}} T_{(d_1,\ldots,d_n)}^{o,N} \\ &\leq 2^d 2^{kd\eta_o} \binom{N}{\lceil d/d_f^o \rceil - 1} d\eta_o \\ &\leq 2^{(k\eta_o+\eta_o+1)d} \binom{N}{\lfloor d/d_f^o \rfloor} \end{aligned}$$

In the case when $d = d_f^o$ we can get a tighter estimation by noticing that the only two possibilities are to have one single regular error event of output weight d_f^o or no regular event at all and just one terminating event also of weight d_f^o, then being the same as some regular event. ■

Proof of Lemma 4.4 ([41], Lemma 1).
We study $A_{w,\leq d}^{i,N}$, separating the case when w is even or odd. Throughout the proof we will use the following simple but useful estimations:

$$\frac{n^m}{m^m} \leq \binom{n}{m} \leq \frac{(en)^m}{m^m} \tag{4.1}$$

$$\binom{n-m}{m} \leq e^{n+m} \tag{4.2}$$

$$t^t(w-t)^{w-t} \geq (w/2)^w \quad \text{for all } t \in [0,w] \tag{4.3}$$

$$\frac{1}{(t-1)^{(t-1)}} \leq \frac{e\,t}{t^t} \tag{4.4}$$

Proof when w is even:

$$A_{w,\leq d}^{i,N} = R_{w,\leq d}^{i,N} + T_{w,\leq d}^{i,N}$$

$$R_{w,\leq d}^{i,N} = \sum_{n=1}^{w/2} R_{w,\leq d,n}^{i,N}$$

where $R_{w,\leq d,n}^{i,N}$ is the number of words of $\mathbb{Z}_2^{L_N}$ with weight w, producing codewords of ϕ_i^N of weight $\leq d$ made by exactly n regular error events and no terminating event (the events possibly spaced with zeros in between).

First we find an estimation for $R_{w,\leq d,w/2}^{i,N}$. Having $w/2$ error events and input weight w, by recursiveness of ϕ_i gives input weight 2 for each event. So the input

words we are counting can be written as $u(D) = \sum_{t=1}^{w/2} D^{b_t}(1 + D^{\delta^i a_t})$, with $b_t > \delta^i a_{t-1}$ (so that the error events have disjoint support). We also have the restriction $\mathrm{w_H}(\phi_i(D)u(D)) \leq d$, but we can obtain an upper bound on the number of such words by imposing a weaker condition: notice that $\mathrm{w_H}\left(\phi_i(D)\sum_{t=1}^{w/2} D^{b_t}(1 + D^{\delta^i a_t})\right) = \sum_{t=1}^{w/2} \mathrm{w_H}\left(\phi_i(D)(1 + D^{\delta^i a_t})\right) \geq I_i \sum_{t=1}^{w/2} a_t$ so we will ask only $I_i \sum_{t=1}^{w/2} a_t \leq d$.

There are $\binom{\lfloor d/I_i \rfloor}{w/2}$ choices for $a_1, \ldots, a_{w/2}$ satisfying $a_t \geq 1$ for all t and $\sum_{t=1}^{w/2} a_t \leq \lfloor d/I_i \rfloor$. Then, there are at most $\binom{M_N}{\frac{w}{2}}$ choices for the beginnings $b_1, \ldots, b_{w/2}$ of the error events, so finally

$$R^{i,N}_{w,\leq d,w/2} \leq \binom{\lfloor d/I_i \rfloor}{\frac{w}{2}} \binom{M_N}{\frac{w}{2}} \leq \frac{1}{w^w}(2M_N)^{w/2} \lfloor d/I_i \rfloor^{w/2} e^w$$

Remember that $M_N = \frac{r}{s}(N + \nu_o) \leq (r+1)N$ for all $N \geq \nu_o$.

Then we have to estimate $\sum_{n=1}^{w/2-1} R^{i,N}_{w,\leq d,n}$.

$$R^{i,N}_{w,\leq d,n} = \sum_{\substack{\boldsymbol{w}=(w_1,\ldots,w_n):\\ w_j \geq 2,\ \sum w_j = w}} \sum_{\substack{\boldsymbol{b}=(b_1,\ldots,b_n):\\ 0 \leq b_1 \leq \cdots \leq b_n \leq M_N - 1}} R^{i,N}_{\boldsymbol{w},\boldsymbol{b},\leq d,n}$$

where $R^{i,N}_{\boldsymbol{w},\boldsymbol{b},\leq d,n}$ is the number of codewords of weight $\leq d$ that are the concatenation of n error events, with input weights $w_1, \ldots, w_n$ and beginning at time $b_1, \ldots, b_n$ respectively. The constraint $w_j \geq 2$ for all j comes from the recursiveness of ϕ_i.

Claim:

$$R^{i,N}_{\boldsymbol{w},\boldsymbol{b},\leq d,n} \leq \binom{d\eta_i}{w-n} \tag{4.5}$$

Proof of the claim:
$R^{i,N}_{\boldsymbol{w},\boldsymbol{b},\leq d,n}$ is smaller than the number of binary words of length $d\eta_i$ with exactly $w - n$ ones, because it is possible to exhibit an injective map from the words we want to count and such words. Given an input word (of length M_N) producing n error events having input weights $w_1, \ldots, w_n$ and fixed beginnings $b_1, \ldots, b_n$, and total output weight $\leq d$, map it in a word of length $d\eta_i$ in the following way: remove all the zeros corresponding to zero state-zero state transitions on the trellis of ϕ_i, and furthermore remove the bit corresponding to the first zero state-other state

transition of each error event (which is surely a one, because a zero would give a zero state-zero state transition). The word obtained in such a way has surely length $< d\eta_i$, then add dummy zeros at the end to get a word of length $d\eta_i$; the number of ones is $w-n$, having removed many zeros and n ones from a word of weight w. Clearly this map is injective (remember that the beginning time of the error events is fixed and known).

Now, using the claim (4.5), we estimate

$$R^{i,N}_{w,\le d,n} = \sum_{\substack{\boldsymbol{w}=(w_1,\dots,w_n):\\ w_j\ge 2,\ \sum w_j = w}} \sum_{\substack{\boldsymbol{b}=(b_1,\dots,b_n):\\ 0\le b_1\le\cdots\le b_n\le M_N-1}} R^{i,N}_{\boldsymbol{w},\boldsymbol{b},\le d,n} \le \binom{w-n-1}{n-1}\binom{M_N}{n}\binom{d\eta_i}{w-n}$$

so that

$$\begin{aligned}
\sum_{n=1}^{w/2-1} R^{i,N}_{w,\le d,n} &\le \sum_{n=1}^{w/2-1} \binom{w-n-1}{n-1}\binom{M_N}{n}\binom{d\eta_i}{w-n} \\
&\le \sum_{n=1}^{w/2-1} e^{w+n-1}\frac{(eM_N)^n}{n^n}\frac{(d\eta_i)^{w-n}}{(w-n)^{w-n}} && \text{by (4.1) and (4.2)} \\
&\le \frac{e^{5w/2}\eta_i^{w/2}}{(w/2)^w}\sum_{n=1}^{w/2-1} M_N^n d^{w-n} && \text{by (4.3)} \\
&\le \frac{e^{5w/2}\eta_i^{w/2}}{(w/2)^w}\frac{w}{2}[(l+1)M_N]^{\frac{w}{2}-1} d^{\frac{w}{2}+1} && \text{as } d\le (l+1)M_N \\
&\le \frac{C^w}{w^w} M_N^{\frac{w}{2}-1} d^{\frac{w}{2}+1} \\
&= o\left(\frac{C^w}{w^w} M_N^{\frac{w}{2}} d^{\frac{w}{2}}\right) && \text{if } N\to\infty \text{ and } d/N\to 0
\end{aligned}$$

Finally, we have to consider the case of terminating events:

$$T^{i,N}_{w,\le d} = \sum_{n=1}^{w/2} T^{i,N}_{w,\le d,n} = \sum_{n=1}^{w/2} \sum_{\substack{\boldsymbol{w}=(w_1,\dots,w_n):\\ \sum w_j = w\\ w_j\ge 2\,\forall j<n,\, w_n\ge 1}} \sum_{\substack{\boldsymbol{b}=(b_1,\dots,b_n):\\ 0\le b_1\le\cdots\le b_n\le M_N-1\\ b_n\ge M_N - d\eta_i}} T^{i,N}_{\boldsymbol{w},\boldsymbol{b},\le d,n}$$

where $T^{i,N}_{w,\le d,n}$ is the number of words of $\mathbb{Z}_2^{L_N}$ with weight w, producing codewords of ϕ_i^N of weight $\le d$ made by exactly $n-1$ regular error events and one terminating event (the events possibly spaced with zeros in between) and $T^{i,N}_{\boldsymbol{w},\boldsymbol{b},\le d,n}$ is the same with the constraint that the error events have input weights $w_1,\dots,w_n$ and beginning

times $b_1, \ldots, b_n$ respectively. Everything is similar to the regular case, except the additional condition $b_n \geq M_N - d\eta_i$ which comes from the remark that the terminating event has clearly output weight $< d$ and so length $< d\eta_i$, and being terminating it cannot start before $M_N - d\eta_i$. Moreover, the recursiveness imposes $w_j \geq 2$ for the regular events, while for the terminating event only $w_n \geq 1$ is required.

With the same proof as for the estimation (4.5) of $R^{i,N}_{\boldsymbol{w},\boldsymbol{b},\leq d,n}$, we have also

$$T^{i,N}_{\boldsymbol{w},\boldsymbol{b},\leq d,n} \leq \binom{d\eta_i}{w-n} \tag{4.6}$$

so that

$$\begin{aligned}
T^{i,N}_{w,\leq d} &\leq \sum_{n=1}^{w/2} \sum_{\substack{\boldsymbol{w}=(w_1,\ldots,w_n):\\ \sum w_j = w\\ w_j \geq 2\, \forall j<n,\, w_n \geq 1}} \sum_{\substack{\boldsymbol{b}=(b_1,\ldots,b_n):\\ 0\leq b_1 \leq \cdots \leq b_n \leq M_N-1\\ b_n \geq M_N - d\eta_i}} \binom{d\eta_i}{w-n} \\
&\leq \sum_{n=1}^{w/2} \binom{w-n}{n-1}\binom{M_N}{n-1} d\eta_i \binom{d\eta_i}{w-n} \\
&\leq e^{5w/2}\eta_i^{w/2}\, d \sum_{n=1}^{w/2} \frac{M_N^{n-1} d^{w-n}}{(n-1)^{(n-1)}(w-n)^{(w-n)}} && \text{by (4.1) and (4.2)} \\
&\leq e^{5w/2}\eta_i^{w/2}\, \frac{d}{M_N} \sum_{n=1}^{w/2} \frac{M_N^{n} d^{w-n}}{n^{n}(w-n)^{(w-n)}}\, en && \text{by (4.4)} \\
&\leq \frac{e^{5w/2+1}\eta_i^{w/2}}{(w/2)^w}\, \frac{d}{M_N} \sum_{n=1}^{w/2} [(l+1)M_N]^n d^{w-n}\, n && \text{by (4.3)} \\
&\leq w \frac{e^{5w/2+1}[(l+1)\eta_i]^{w/2}}{(w/2)^w}\, \frac{d}{M_N} (M_N d)^{w/2} && \text{by (4.3) and } d \leq (l+1)M_N \\
&= o\left(\frac{C^w}{w^w} M_N^{\frac{w}{2}} d^{\frac{w}{2}} \right) && \text{if } N \to \infty \text{ and } d/N \to 0
\end{aligned}$$

Proof when w is odd: As in the even case, we use

$$A^{i,N}_{w,\leq d} = R^{i,N}_{w,\leq d} + T^{i,N}_{w,\leq d}$$

We start with the concatenations of regular events

$$R^{i,N}_{w,\leq d} = \sum_{n=1}^{\lfloor w/2 \rfloor} R^{i,N}_{w,\leq d,n}$$

and we separate the term $R^{i,N}_{w,\le d,\lfloor w/2\rfloor}$, which now is made of $w/2-1$ events with input weight 2 and one event with input weight 3, i.e. the input has the form $u(D)=\sum_{t=1}^{\lfloor w/2\rfloor-1} D^{b_t}(1+D^{\delta^i a_t})+D^b(1+D^a+D^{a'})$. All the error events have disjoint support, which implies the weaker condition that $b_1<\dots b_{\lfloor w/2\rfloor-1}$ and $b\neq b_1,\dots,b_{\lfloor w/2\rfloor-1}$. The overall output weight is $\le d$, and this implies the weaker condition $I_i\sum_{t=1}^{\lfloor w/2\rfloor-1} a_t\le d$ and $a<a'<\mu_i d$. There are $\binom{\mu_i d}{2}$ choices for such a,a', $\binom{\lfloor d/I_i\rfloor}{\lfloor w/2\rfloor-1}$ choices for $a_1,\dots,a_{\lfloor w/2\rfloor-1}$, less then $\lfloor w/2\rfloor\binom{M_N}{\lfloor w/2\rfloor}$ choices for $b_1,\dots,b_{\lfloor w/2\rfloor-1}$,$b$, where the factor $\lfloor w/2\rfloor$ comes from the choice of the position where to put the error event of weight 3 in between the other events. Summarizing:

$$\begin{aligned}R^{i,N}_{w,\le d,\lfloor w/2\rfloor}&\le\left\lfloor\frac{w}{2}\right\rfloor\binom{M_N}{\lfloor w/2\rfloor}\binom{\mu_i d}{2}\binom{\lfloor d/I_i\rfloor}{\lfloor w/2\rfloor-1}\\&\le\frac{\mu_i^2}{16\sqrt{e}}\,\frac{\sqrt{e}^w}{w^w}\,M_N^{\lfloor w/2\rfloor}\,d^2\left\lfloor\frac{d}{I_i}\right\rfloor^{\lfloor\frac{w}{2}\rfloor-1}\end{aligned}$$

Then the terms with $n<\lfloor w/2\rfloor$ regular error events are estimated exactly as in the case when w is even:

$$\begin{aligned}\sum_{n=1}^{\lfloor w/2\rfloor-1}R^{i,N}_{w,\le d,n}&=\sum_{n=1}^{\lfloor w/2\rfloor-1}\sum_{\substack{\boldsymbol{w}=(w_1,\dots,w_n):\\ w_j\ge 2,\ \sum w_j=w}}\ \sum_{\substack{\boldsymbol{b}=(b_1,\dots,b_n):\\ 0\le b_1\le\dots\le b_n\le M_N-1}}R^{i,N}_{\boldsymbol{w},\boldsymbol{b},\le d,n}\\&\le\sum_{n=1}^{\lfloor w/2\rfloor-1}\binom{w-n-1}{n-1}\binom{M_N}{n}\binom{d\eta_i}{w-n}\\&\le\sum_{n=1}^{\lfloor w/2\rfloor-1}e^{w+n-1}\frac{(eM_N)^n}{n^n}\frac{(d\eta_i)^{w-n}}{(w-n)^{w-n}}&&\text{by (4.1) and (4.2)}\\&\le\frac{e^{5w/2}\eta_i^{w/2}}{(w/2)^w}\sum_{n=1}^{\lfloor w/2\rfloor-1}M_N^n d^{w-n}&&\text{by (4.3)}\\&\le\frac{e^{5w/2}\eta_i^{w/2}}{(w/2)^w}\frac{w}{2}[(l+1)M_N]^{\lfloor\frac{w}{2}\rfloor-1}d^{\lceil\frac{w}{2}\rceil+1}&&\text{as } d\le(l+1)M_N\\&\le\frac{(2e^3\sqrt{(l+1)\eta_i})^w}{w^w}M_N^{\lfloor\frac{w}{2}\rfloor-1}d^{\lceil\frac{w}{2}\rceil+1}\end{aligned}$$

Now differently from the even case, when estimating $T^{i,N}_{w,\le d}$ we have to separate the main term $T^{i,N}_{w,\le d,\lceil w/2\rceil}$, which will not be $o\left(R^{i,N}_{w,\le d,\lfloor w/2\rfloor}\right)$. We have to count inputs producing $\lfloor w/2\rfloor$ regular error events each with input weight 2 and one terminating

event with input weight 1, with overall output weight $\leq d$. We count inputs of the kind $u(D) = \sum_{t=1}^{\lfloor w/2 \rfloor} D^{b_t}(1 + D^{\delta^i a_t}) + D^{M_N - l}$ satisfying the weaker conditions: $0 \leq b_1 < \cdots < b_{\lfloor w/2 \rfloor} < M_N$, $l \leq \eta_i d$, $I_i \sum_t a_t \leq d$. We get:

$$T^{i,N}_{w,\leq d,\lceil w/2 \rceil} \leq \binom{M_N}{\lfloor w/2 \rfloor} d\eta_i \binom{\lfloor d/I_i \rfloor}{\lfloor w/2 \rfloor} \leq \eta_i \left(\frac{2e}{w-1}\right)^{w-1} M_N^{\lfloor w/2 \rfloor}\, d \left\lfloor \frac{d}{I_i} \right\rfloor^{\lfloor w/2 \rfloor} \tag{4.7}$$

Finally, the same as in the even case,

$$\begin{aligned}
T^{i,N}_{w,\leq d} &\leq \sum_{n=1}^{\lfloor w/2 \rfloor} \sum_{\substack{\boldsymbol{w}=(w_1,\ldots,w_n):\\ \sum w_j = w \\ w_j \geq 2\, \forall j<n,\, w_n \geq 1}} \sum_{\substack{\boldsymbol{b}=(b_1,\ldots,b_n):\\ 0 \leq b_1 \leq \cdots \leq b_n \leq M_N - 1 \\ b_n \geq M_N - d\eta_i}} \binom{d\eta_i}{w-n} \\
&\leq \sum_{n=1}^{\lfloor w/2 \rfloor} \binom{w-n}{n-1} \binom{M_N}{n-1} d\eta_i \binom{d\eta_i}{w-n} \\
&\leq e^{5w/2} \eta_i^{\lfloor w/2 \rfloor}\, d \sum_{n=1}^{\lfloor w/2 \rfloor} \frac{M_N^{n-1} d^{w-n}}{(n-1)^{(n-1)}(w-n)^{(w-n)}} && \text{by (4.1) and (4.2)} \\
&\leq e^{5w/2} \eta_i^{w/2} \frac{d}{M_N} \sum_{n=1}^{\lfloor w/2 \rfloor} \frac{M_N^n d^{w-n}}{n^n (w-n)^{(w-n)}}\, en && \text{by (4.4)} \\
&\leq \frac{e^{5w/2+1} \eta_i^{w/2}}{(w/2)^w} \frac{d}{M_N} \sum_{n=1}^{\lfloor w/2 \rfloor} [(l+1)M_N]^n d^{w-n}\, n && \text{by (4.3)} \\
&\leq w \frac{e^{5w/2+1} [(l+1)\eta_i]^{w/2}}{(w/2)^w} \frac{d}{M_N} M_N^{\lfloor w/2 \rfloor} d^{\lceil w/2 \rceil} && \text{by (4.3) and } d \leq (l+1)M_N
\end{aligned}$$

This ends the proof of Lemma 4.4 ■

Proof of Lemma 4.5 ([41], Lemma 2):
We count only some words consisting of $w/2$ regular error events (clearly each of input weight 2): we count input words of the form

$$\sum_{t=1}^{w/2} D^{i_t + h_{t-1}\delta^i} + D^{i_t + h_t \delta^i}$$

with:

1. $0 \leq i_1 < i_2 < \cdots < i_{w/2} < M_N - \delta^i \lfloor d/d_2^i \rfloor$;
2. $h_0 = 0$ and $1 \leq h_1 < h_2 < \cdots < h_{w/2} \leq \lfloor d/d_2^i \rfloor$.

Note that these input words are defined in such a way that:

- they are all distinct (and so, by injectivity of ϕ_i, they give different codewords)
- they have weight w;
- they are inputs of $w/2$ disjoint error events;
- they produce output weight $\leq d$. In fact: the t-th error event has input $D^{i_t+h_{t-1}}(1+D^{\delta^i(h_t-h_{t-1})})$ (fixing the notation $h_0=0$), so that the output has weight $\mathrm{w_H}\big(\phi_i((1+D^{\delta^i(h_t-h_{t-1})}))\big) \leq d_2^i\,(h_t-h_{t-1})$, and then the total output weight of the $w/2$ events is less then $d_2^i \sum_{t=1}^{w/2}(h_t-h_{t-1}) = d_2^i h_{w/2} \leq d$.

How many such input words are there? There are $\binom{M_N-\delta^i\lfloor d/d_2^i\rfloor}{w/2}$ choices for the indexes $i_1,\ldots,i_{w/2}$ and $\binom{\lfloor d/d_2^i\rfloor}{w/2}$ choices for $h_1,\ldots,h_{w/2}$, so finally:

$$R^{i,N}_{w,\leq d,w/2} \geq \binom{M_N-\delta^i\lfloor d/d_2^i\rfloor}{w/2}\binom{\lfloor d/d_2^i\rfloor}{w/2} \geq \left[\frac{M_N-\delta^i\lfloor d/d_2^i\rfloor}{w/2}\,\frac{\lfloor d/d_2^i\rfloor}{w/2}\right]^{w/2}$$

The last remark is that $M_N-\delta^i\lfloor d/d_2^i\rfloor \geq \frac{M_N}{2}$ as $d \leq \frac{d_2^i M_N}{2\delta^i}$ by assumption. ■

4.3 Minimum distance

In this section we state and prove our results on the minimum distance: an estimation of the left tail of its distribution, based on techniques from [41], and a deterministic upper bound based on ideas from [2].

Define (d_f^o even):

$$\alpha := 1-\frac{4}{d_f^o}\,, \qquad \beta := 1-\frac{2}{d_f^o}\,. \tag{4.8}$$

Notice that both α and β are increasing functions of d_f^o. If $d_f^o \geq 4$, we have $0 \leq \alpha < \beta < 1$, and $d_f^o \geq 6$ implies also $\alpha > 0$.

4.3.1 Left tail of the minimum distance distribution

The upper bound for the left tail of the distribution of $\mathrm{d}_N^{\min}$ has been obtained in [41], as follows. The precise statement given in [41], Theorem 2.a is here the last item in Corollary 4.1.

Lemma 4.6 ([41], Lemma 6) For all $d \leq K_N$,

$$\mathbb{P}(\mathrm{d}_N^{\min} \leq d) \leq \sum_{w=d_f^o}^{\mu_i d} \frac{1}{\binom{L_N}{w}}\, A_w^{N,o} A^{i,N}_{w,\leq d}$$

□

Proof: Simply notice that

$$\begin{aligned}\{\mathrm{d}_N^{\min} \leq d\} &= \left\{\exists \boldsymbol{x} \in \phi_o^N(\mathbb{Z}_2^k N) : \mathrm{w_H}(\phi_i^N \circ \Pi_N(\boldsymbol{x})) \leq d\right\} \\ &= \bigcup_w \bigcup_{\substack{\boldsymbol{x}\in\phi_o^N(\mathbb{Z}_2^k N):\\ \mathrm{w_H}(\boldsymbol{x})=w}} \left\{\mathrm{w_H}(\phi_i^N \circ \Pi_N(\boldsymbol{x})) \leq d\right\}\end{aligned}$$

so that, by the union bound,

$$\mathbb{P}(\mathrm{d}_N^{\min} \leq d) \leq \sum_w \sum_{\substack{\boldsymbol{x}\in\phi_o^N(\mathbb{Z}_2^k N):\\ \mathrm{w_H}(\boldsymbol{x})=w}} \mathbb{P}\left(\mathrm{w_H}(\phi_i^N \circ \Pi_N(\boldsymbol{x})) \leq d\right) = \sum_w A_w^{o,N} \frac{A_{w,\leq d}^{i,N}}{\binom{L_N}{w}}$$

Finally notice that $A_w^{o,N} = 0$ if $w < d_f^o$ and $A_{w,\leq d}^{i,N} = 0$ if $w > \mu_i d$. ■

Theorem 4.1 ([41], Theorem 2.a) For $N \to \infty$, if $d = o(N^\beta)$, then

$$\mathbb{P}(\mathrm{d}_N^{\min} \leq d) \leq m_f^o \left(\frac{2e}{\sqrt{r}}\right)^{d_f^o} N^{1-d_f^o/2} \left\lfloor \frac{d}{d_2^i} \right\rfloor^{d_f^o/2} + o\left(N^{1-d_f^o/2} d^{d_f^o/2}\right)$$

□

Proof: It follows immediately from Lemma 4.6, estimating the enumerating coefficients of the constituent encoders with Lemmas 4.2 and 4.4, so that you get:

$$\mathbb{P}(\mathrm{d}_N^{\min} \leq d) \leq m_f^o \left(\frac{2e}{\sqrt{r}}\right)^{d_f^o} N^{1-d_f^o/2} \left\lfloor \frac{d}{d_2^i} \right\rfloor^{d_f^o/2} + \sum_{w=d_f^o+1}^{\mu_i d} C^w N^{\lfloor w/d_f^o \rfloor - \lceil w/2 \rceil} d^{\lceil w/2 \rceil}$$

for some $C > 0$ depending on ϕ_o and ϕ_i but not growing with N and d. The conclusion comes from separating odd and even w:

$$\sum_{\substack{w>d_f^o\\ w \text{ odd}}} C^w N^{\lfloor w/d_f^o \rfloor - \lceil w/2 \rceil} d^{\lceil w/2 \rceil} \leq \left(\frac{d}{N}\right)^{1/2} \sum_{w \geq d_f^o+1} \left[CN^{1/d_f^o} \left(\frac{d}{N}\right)^{1/2}\right]^w$$

and

$$\sum_{\substack{w>d_f^o\\ w \text{ even}}} C^w N^{\lfloor w/d_f^o \rfloor - \lceil w/2 \rceil} d^{\lceil w/2 \rceil} \leq \sum_{w \geq d_f^o+2} \left[CN^{1/d_f^o} \left(\frac{d}{N}\right)^{1/2}\right]^w$$

and finally noticing that if $d = o(N^\beta)$ then $CN^{1/d_f^o} \left(\frac{d}{N}\right)^{1/2} \to 0$, so that the sums are convergent; also notice that, being dominated by their first term, they are both

$o\left(N^{1-d_f^o/2}d^{d_f^o/2}\right)$. ■

It is possible to obtain also a lower bound for the left tail of the minimum distance distribution, showing that asymptotically the upper bound in Thm. 4.1 is tight. This lower bound is new; its proof uses techniques from the proof of Thm. 2b in [41] and the inclusion-exclusion principle.

First of all, fix some particular outer codewords. Let $\boldsymbol{c}^*$ be a word of the outer code which has $\mathrm{w_H}(\boldsymbol{c}^*) = d_f^o$ and is one error event, starting at time 0 and ending after T trellis steps for some constant T. Note that $2 \leq T \leq d_f^o \eta_o$.

Consider $N > T$. Define $\boldsymbol{c}_j^*$ as the shift to the right of $\boldsymbol{c}^*$ for j trellis steps; clearly, if $|i-j| \geq T$, then $\boldsymbol{c}_i^*$ and $\boldsymbol{c}_j^*$ have non-overlapping supports. Define the set of indexes $J := \{d_f^o \eta_o \, i \, , \, i \in \mathbb{Z}^+\} \cap \{0,1,\ldots,N-1-d_f^o\eta_o\}$, so that $i,j \in J$ clearly ensures $|i-j| \geq d_f^o \eta_o \geq T$. For $j \in \{0,1,\ldots,N-1-d_f^o\eta_o\}$ and $d \in \mathbb{N}$, define the events

$$E_j^*(d) := \left\{\mathrm{w_H}(\phi_i^N(\Pi_N(\boldsymbol{c}_j^*))) \leq d\right\} \cap \{\phi_i^N(\Pi_N(\boldsymbol{c}_j^*)) \, \text{has} \, d_f^o/2 \, \text{regular error events}\}$$

Clearly, for any j, $E_j^*(d)$ implies $\mathrm{d}_N^{\min} \leq d$, so that

$$\mathbb{P}(\mathrm{d}_N^{\min} \leq d) \geq \mathbb{P}\left(\bigcup_{j \in J} E_j^*(d)\right)$$

We will get our lower bound by estimating the probability of this union with the union-intersection bound. The following lemma, whose proof follows part of the proof of Thm. 2.b in [41], gives us the expression for $\mathbb{P}\big(E_j^*(d)\big)$ and shows that asymptotically these events are almost pairwise independent.

Lemma 4.7

- for all $j \in [0,\ldots,N-T-1]$, $\mathbb{P}(E_j^*(d)) = \dfrac{R^{i,N}_{d_f^o,\leq d,d_f^o/2}}{\binom{L_N}{d_f^o}}$.
- if i and j are such that $|i-j| \geq T$, $i \neq j$,

$$\mathbb{P}(E_i^*(d) \cap E_j^*(d)) \leq \frac{\binom{L_N}{d_f^o}}{\binom{L_N - d_f^o}{d_f^o}} \mathbb{P}(E_i^*(d)) \mathbb{P}(E_j^*(d))$$

□

Note that $1 \leq \frac{\binom{L_N}{d_f^o}}{\binom{L_N-d_f^o}{d_f^o}} \leq \left(1+\frac{d_f^o}{L_N-2d_f^o+1}\right)^{d_f^o}$, so that $\lim_{N\to\infty} \frac{\binom{L_N}{d_f^o}}{\binom{L_N-d_f^o}{d_f^o}} = 1$; also note that $\frac{\binom{L_N}{d_f^o}}{\binom{L_N-d_f^o}{d_f^o}}$ is decreasing with L_N, and so also with N.

Proof of Lemma 4.7: The first statement is immediate, let's prove the second one. Let $\boldsymbol{c}_i^* = \sum_{m=1}^{d_f^o} D^{t_m}$. Given d_f^o indexes $\tau_1,\ldots,\tau_{d_f^o}$ each in $[L_N] := \{0,\ldots,L_N-1\}$, define the event $E_{\tau_1,\ldots,\tau_{d_f^o}} = \{\Pi(D^{t_h}) = D^{\tau_h} \forall h = 1,\ldots,d_f^o\}$. Clearly

$$\mathbb{P}\big(E_i^*(d)\cap E_j^*(d)\big) = \sum_{\tau_1,\ldots,\tau_{d_f^o}\in[L_N]} \mathbb{P}\big(E_i^*(d)\cap E_{\tau_1,\ldots,\tau_{d_f^o}}\big)\mathbb{P}\big(E_j^*(d)\big|E_i^*(d)\cap E_{\tau_1,\ldots,\tau_{d_f^o}}\big)$$

Then note that

$$\mathbb{P}\big(E_j^*(d)\big|E_i^*(d)\cap E_{\tau_1,\ldots,\tau_{d_f^o}}\big) = \mathbb{P}\big(E_j^*(d)\big|E_{\tau_1,\ldots,\tau_{d_f^o}}\big) \leq \frac{R^{i,N}_{d_f^o,\leq d,d_f^o/2}}{\binom{L_N-d_f^o}{d_f^o}} = \mathbb{P}\big(E_j^*(d)\big)\frac{\binom{L_N}{d_f^o}}{\binom{L_N-d_f^o}{d_f^o}}$$

so that

$$\mathbb{P}\big(E_i^*(d)\cap E_j^*(d)\big) \leq \sum_{\tau_1,\ldots,\tau_{d_f^o}\in[L_N]} \mathbb{P}\big(E_i^*(d)\cap E_{\tau_1,\ldots,\tau_{d_f^o}}\big)\mathbb{P}\big(E_j^*(d)\big)\frac{\binom{L_N}{d_f^o}}{\binom{L_N-d_f^o}{d_f^o}}$$

which ends the proof, as $\sum_{\tau_1,\ldots,\tau_{d_f^o}\in[L_N]} \mathbb{P}\big(E_i^*(d)\cap E_{\tau_1,\ldots,\tau_{d_f^o}}\big) = \mathbb{P}\big(E_i^*(d)\big)$. ■

Theorem 4.2 For all $N \geq d_f^o\eta_o$ and $d \geq \frac{1}{2}d_f^o d_2^i$,

$$\mathbb{P}(\mathrm{d}_N^{\min} \leq d) \geq C_1\, N\, M_N^{-\frac{d_f^o}{2}} \left\lfloor \frac{d}{d_2^i} \right\rfloor^{\frac{d_f^o}{2}} \left[1 - C_2\, N\, M_N^{-\frac{d_f^o}{2}} \left\lfloor \frac{d}{d_2^i} \right\rfloor^{\frac{d_f^o}{2}}\right]$$

where $C_1 = \frac{2^{d_f^o/2}}{e^{\frac{d_f^o}{4}} d_f^o \eta_o}$ and $C_2 = \frac{(2e)^{d_f^o}}{2d_f^o\eta_o}$

Proof: We use the inclusion-exclusion principle:

$$\mathbb{P}(\mathrm{d}_N^{\min} \leq d) \geq \mathbb{P}\left(\bigcup_{j\in J} E_j^*(d)\right) \geq \sum_{j\in J}\mathbb{P}(E_i^*(d)) - \sum_{\substack{i,j\in J\\ i<j}}\mathbb{P}(E_i^*(d)\cap E_j^*(d))$$

From this, we use Lemma 4.7 and then we estimate $R^{i,N}_{d_f^o,\leq d,d_f^o/2}$ with Lemmas 4.4 and 4.5. Also remember that $|J| = \lfloor N/(d_f^o \eta_o) \rfloor$. We get:

$$\mathbb{P}\left(\bigcup_{j\in J} E_j^*(d)\right) \geq |J| \frac{R^{i,N}_{d_f^o,\leq d,d_f^o/2}}{\binom{L_N}{d_f^o}} - \binom{|J|}{2} \frac{\binom{L_N}{d_f^o}}{\binom{L_N - d_f^o}{d_f^o}} \left[\frac{R^{i,N}_{d_f^o,\leq d,d_f^o/2}}{\binom{L_N}{d_f^o}}\right]^2$$
$$\geq \frac{N}{d_f^o \eta_o} \frac{2^{d_f^o/2}}{e^{d_f^o}} M_N^{-d_f^o/2} \left\lfloor \frac{d}{d_2^i} \right\rfloor^{d_f^o/2} \left[1 - \frac{1}{2} \frac{N}{d_f^o \eta_o} (2e)^{d_f^o} M_N^{-d_f^o/2} \left\lfloor \frac{d}{d_2^i} \right\rfloor^{d_f^o/2}\right]$$

■

We highlight here an immediate consequence of Theorems 4.1 and 4.2 that we will need later.

Corollary 4.1 For $N \to \infty$, if $d = o(N^\beta)$, then there exist two positive constants C_1 and C_2 (depending on the constituent encoders, but neither on N nor on d) such that

$$C_1 N^{1-d_f^o/2} d^{d_f^o/2} \leq \mathbb{P}(\mathrm{d}_N^{\min} \leq d_N) \leq C_2 N^{1-d_f^o/2} d^{d_f^o/2}$$

and so, in particular, $\mathbb{P}(\mathrm{d}_N^{\min} \leq d_N) \to 0$. ■

4.3.2 Deterministic upper bound

The picture of $\mathrm{d}_N^{\min}$ given in Corollary 4.1 was completed in [41] by proving that if $d/N^\beta \to \infty$, then $\mathbb{P}(\mathrm{d}_N^{\min} \leq d) \to 1$ ([41], Thm. 2.b). Their proof, based on a second-order method, did not underline how fast was the convergence. However a much stronger result holds true: deterministically (i.e. for any given permutation π as interleaver), $\mathrm{d}_N^{\min}$ cannot grow more than $C \log N \, N^\beta$ for some constant C. This deterministic upper bound was obtained by Bazzi, Mahdian and Spielman for Repeat-Convolute codes ([2], Thm. 2), but it is easy to generalize it to our setting including a general outer encoder. Actually Bazzi et al. also study serial turbo codes, not only Repeat-Convolute, but they do so in an even more general setting allowing growing memory and obtain a result ([2], Thm. 4) which specialized to the constant memory case gives a less tight bound. In addition to considering a general outer encoder, we also underline the role that d_2^i plays in the bound, by slightly modifying the proof.

The result we get is the following.

Theorem 4.3 For all $N \geq \max(d_f^o \eta_o, \frac{1}{2} d_f^o \delta)$

$$\mathrm{d}_N^{\min} \leq \frac{1}{2} (d_f^o)^2 d_2^i \log b \left\lfloor \frac{N}{b} \right\rfloor \quad \text{where } b = \left(\frac{1}{4} \left\lceil \frac{1}{\delta^{d_f^o}} \left\lfloor \frac{N}{d_f^o \eta_o} \right\rfloor \right\rceil\right)^{2/d_f^o}$$

This also implies that, for sufficiently big N,

$$\mathrm{d}_N^{\min} \leq 2d_2^i d_f^o \delta^2 \big(4(d_f^o \eta_o + 1)\big)^{2/d_f^o} N^\beta \log N$$

□

We give now the proof of this theorem, which requires some preliminary lemmas.

Let $\boldsymbol{c}^*$, J, c_j^* be defined as in Section 4.3.1. The aim is to show that, for any interleaver, it is possible to find a suitable subset of the $\boldsymbol{c}_j^*$s, with cardinality growing at most as $c \log N$, such that the corresponding output has weight smaller than $KN^\beta \log N$.

Define $\boldsymbol{\sigma} : J \to \mathbb{Z}_\delta^{d_f^o}$ by associating to an index $j \in J$ a vector $(\sigma_1(j), \ldots, \sigma_{d_f^o}(j))$ in the following way: if $\boldsymbol{c}_j^* = \sum_{m=1}^{d_f^o} D^{t_m}$ and $\pi(c_j^*) = \sum_{m=1}^{d_f^o} D^{\tau_m}$ then $\sigma_m(j) = \tau_m \mod \delta$, i.e. $\boldsymbol{\sigma}(j)$ shows the positions where the ones of the codeword $\boldsymbol{c}_j^*$ end up after the permutation, only considered $\mod \delta$. By the pigeonhole principle, clearly there exists $U \subseteq J$ with $|U| \geq \left\lceil \frac{|J|}{\delta^{d_f^o}} \right\rceil$ such that $\boldsymbol{\sigma}(i) = \boldsymbol{\sigma}(j)$ for all $i,j \in U$.

From now on, we will consider only $\boldsymbol{c}_j^*$ with $j \in U$. The idea is that as all the ones in these words are permuted to positions at a distance multiple of δ, when applying ϕ_i any pair of them gives a bounded output weight. Now we need to bound the distance within these pairs, for a subset of indexes $S \subseteq U$, in order to bound the output weight (see Lemma 4.3).

Now look at $[M_N] = \{0, \ldots, M_N - 1\}$ and consider it as divided in b intervals $I_1, \ldots, I_b$, each of length $\lfloor M_N/b \rfloor$ (except a possibly longer one at the end); b is a parameter depending on N that will be properly chosen later.

Define an hypergraph $H = (V,E)$ in the following way. Take a d_f^o-partite vertex set V being the union of d_f^o disjoint copies of $W = \{I_1, \ldots, I_b\}$. The set of hyperedges E has cardinality $|U|$ and is d_f^o-regular in the sense that $E \subseteq W^{d_f^o}$, i.e. every hyperedge contains exactly one vertex from each of the d_f^o copies of W. Any edge in E corresponds to an index $j \in U$, and is defined as $e = (I_{h_1}, \ldots, I_{h_{d_f^o}}) \in W^{d_f^o}$ where, denoting $\boldsymbol{c}_j^* = \sum_{m=1}^{d_f^o} D^{t_m}$, h_m is the index in $[b]$ such that $\pi(D^{t_m}) \in I_{h_m}$.

Define the degree of a vertex in the hypergraph as the number of hyperedges that contain that vertex. The following lemma holds true:

Lemma 4.8 ([2], Lemma 3) If $4b^{d_f^o/2} \leq \left\lceil \frac{1}{\delta^{d_f^o}} \left\lfloor \frac{N}{d_f^o \eta_o} \right\rfloor \right\rceil$, then there exists a non-empty subset $S \subset E$, with $|S| \leq d_f^o \log b$ and such that in the induced subhypergraph (V,S) every vertex has even degree (possibly zero).

We first clarify how this lemma implies Theorem 4.3, then we will prove it using a few intermediate results. As $S \subset E$, there is a bijection from S to a subset $\tilde{S} \subset U$:

any $s \in S$ corresponds to a codeword $\boldsymbol{c}_j^*$, $j \in \tilde{S}$. If you define $\boldsymbol{c} = \sum_{j\in\tilde{S}} \boldsymbol{c}_j^*$ it is clearly a feasible outer codeword so that $\phi_{i,N}(\pi(\boldsymbol{c}))$ will be a codeword of the serial scheme. Now notice that every vertex having even degree in (V,S) means that $\pi(\boldsymbol{c})$ has an even number of ones in any interval $I_1, \dots I_b$. This also implies that $\pi(\boldsymbol{c})$ is made of pairs of ones falling in the same interval and so having a distance at most the length N/b; more precisely, if you let $\pi(\boldsymbol{c}) = \sum_{m=1}^{|S|d_f^o} D^{t_j}$, for any even m $t_{m+1} - t_m \leq N/b$. As $\tilde{S} \subset U$ we also know that $t_{m+1} - t_m$ is a multiple of δ, so:

$$\mathrm{w_H}\big(\phi_i^N(\pi(\boldsymbol{c}))\big) \leq \sum_{m=1}^{|S|d_f^o/2} \mathrm{w_H}\big(\phi_i^N(D^{t_{2m-1}} + D^{t_{2m}})\big) \leq d_2^i \sum_{m=1}^{|S|d_f^o/2} (t_{2m} - t_{2m-1}) \leq d_2^i |S| \frac{d_f^o}{2} \frac{N}{b}$$

Finally use the bound on $|S|$ which is the most important part of Lemma 4.8: $|S| \leq d_f^o \log b$, which ends the proof because of the upper bound on b.

Now we proceed to prove Lemma 4.8. From the hypergraph $H = (V,E)$, construct a bipartite graph $G = (V',E')$ in the following way. Let V' be the union of two disjoint copies (say W_1', W_2') of $W^{d_f^o/2}$, and then put and edge connecting vertices $(v_1, \dots, v_{d_f^o/2}) \in W_1'$ and $(w_1, \dots, w_{d_f^o/2}) \in W_2'$ if and only if there is an hyperedge $(v_1, \dots, v_{d_f^o/2}, w_1, \dots, w_{d_f^o/2}) \in E$. Note that $|E'| = |E| = |U|$ while $|V'| = 2|W|^{d_f^o/2} = 2b^{d_f^o/2}$. The next step is to apply the following lemma to prove that there is a 'small' cycle in G (i.e. a cycle with length growing at most logarithmically in N). Then we will conclude showing how this cycle in G gives the subhypergraph of H promised in Lemma 4.8.

Lemma 4.9 ([2], Lemma 4) Let G be a graph with n vertices and m edges. If $m \geq 2n$, then G has a cycle of length at most $2 \log_2 n$. □

Proof: First notice that surely G is not a tree, as $m > n - 1$. Let l be the length of the smallest cycle: clearly $l \geq 3$ (there are no self-loops and no parallel edges), now we want to prove that $l \leq 2 \log n$. The first part of the proof is for the case when $\deg v \geq 3$ for all vertex v. In this case, notice that if you look at any vertex, take its neighbours, then their neighbours and so on you can proceed for at least $\lceil l/2 \rceil - 1$ steps without touching again a previously visited vertex, and thanks to the assumption $\deg v \geq 3$, you have built a tree, subgraph of G, with at least $1 + 3\sum_{i=1}^{\lceil l/2 \rceil - 1} 2^{i-1}$ vertices. This implies that $n \geq 1 + 3\sum_{i=1}^{\lceil l/2 \rceil - 1} 2^{i-1} = 1 + 3(2^{\lceil l/2 \rceil - 1} - 1) = 2^{\lceil l/2 \rceil} + 2^{\lceil l/2 \rceil - 1} - 2 \geq 2^{l/2}$ (as $l \geq 3$). This gives $l \leq 2 \log_2 n$.

Now the proof for general graphs (i.e. removing the previous assumption on the degrees) is by induction on n. As initial step, you can take $n = 5$ where the only graph satisfying the assumption is the complete graph, which has cycles of length $3 \leq 2 \log_2 5$ (or even more simply you can notice that for $n < 5$, no graph with

$m \geq 2n$ exists so the statement of this Lemma trivially holds true). For the induction step, assume the statement of this Lemma is true for graphs with $n-1$ vertices. When you look at a graph G with n vertices, either every vertex has degree ≥ 3, and so the statement has already been proved true, or there is a vertex, say w, with degree $\deg w < 3$, so that if you look at G' subgraph of G obtained by removing w (and removing edges connected to w), you have G' with $n-1$ vertices and a number of edges $\geq m-2 \geq 2n-2 = 2(n-1)$. So G' has a cycle of length at most $2\log_2(n-1) \leq 2\log_2 n$ which is clearly also a cycle in G. ■

Now apply Lemma 4.9 to the graph G obtained from the hypergraph H, which has $2b^{d_f^o/2}$ vertices and $|U|$ edges, where $|U| \geq \lceil |J|/\delta^{d_f^o} \rceil$ and $|J| = \lfloor N/(d_f^o \eta_o) \rfloor$. So if b is such that $\lfloor N/(d_f^o \eta_o) \rfloor \geq 4b^{d_f^o/2}$ the graph G has a cycle of length $l \leq 2\log b$. Note that G could also have some parallel edges: in this case, we cannot use Lemma 4.9 but clearly G has a cycle of length 2. Now let's see what a cycle C of length l in G means on the hypergraph H: if you take in H exactly the hyperedges corresponding to edges of C, you have that every hyperedge has in common with the previous hyperedge half of the vertices, alternatively on the left or right side, so that in the end every vertex is touched an even number of times. So this construction gives the subset of hyperedges S claimed in Lemma 4.8 and ends the proof of the deterministic upper bound.

4.4 Probabilistic consequences

In this section we derive probabilistic results for the sequence of minimum distances based on the estimations of the previous section. Roughly speaking, we show that minimum distances almost grow as N to some positive exponent which is less then one and converges in a weak way to β, while in a strong way the sequence densely covers the whole interval $[\alpha,\beta]$, α and β being defined in (4.8). Finally we show how these results can be transferred to ML word error probabilities. We show that typically $P(e|\Pi_N)$ is subexponentially decreasing to zero, again with a speed densely covering the interval $[\alpha,\beta]$ with probability one and weakly converging to β.

Remember that our probabilistic space is the serial turbo ensemble generated by a sequence of independent r.v.s $(\Pi_N)_{N\in\mathbb{N}}$, with each Π_N uniformly distributed over S_{L_N}. The main probabilistic tool we will use in our derivation is the Borel-Cantelli lemma (see e.g. [12] Thm. 1.4.2) which states that, for every sequence of events $(A_n)_{n\in\mathbb{N}}$

(i) if $\sum_{n\in\mathbb{N}} \mathbb{P}(A_n) < \infty$, then $\mathbb{P}(\{A_n \text{ i.o.}\}) = 0$;

(ii) if the A_n's are independent and $\sum_{n\in\mathbb{N}} \mathbb{P}(A_n) = \infty$, then $\mathbb{P}(\{A_n \text{ i.o.}\}) = 1$;

where the event $\{A_n \text{ i.o.}\}$ ('A_n occurs infinitely often') is defined as

$$\{A_n \text{ i.o.}\} := \bigcap_{n\in\mathbb{N}} \Big(\bigcup_{m\geq n} A_m \Big) .$$

We define, for every $N \in \mathbb{N}$ and $x \in [0,1]$,

$$E_N^x := \{ \mathrm{d}_N^{\min} \leq K_N^x \} ,$$

$$\theta(x) := 1 + \tfrac{d_f^o}{2}(x-1) .$$

Observe that $\theta(x)$ is an increasing function of x, and that $\theta(\alpha) = -1$, $\theta(\beta) = 0$. From Corollary 4.1 it follows that, for $0 \leq x < \beta$, two positive constants C' and C'' exist such that

$$C' N^{\theta(x)} \leq \mathbb{P}(E_N^x) \leq C'' N^{\theta(x)} . \tag{4.9}$$

4.4.1 Minimum distances

Usually, asymptotics of the minimum distance of ensembles of codes are studied by defining the relative minimum distance $\delta_N = \mathrm{d}_N^{\min}/K_N$. In our case Theorem 4.3 directly implies that deterministically $\delta_N \overset{N\to\infty}{\longrightarrow} 0$ for any sequence of serial turbo codes. For this reason we propose the following non linear rescaling

$$X_N := \frac{\log(\mathrm{d}_N^{\min})}{\log(K_N)} .$$

With this rescaling, $(X_N)_N$ is a sequence of independent random variables taking values in $[0,1]$, since $1 \leq \mathrm{d}_N^{\min} \leq K_N$. The meaning of X_N is to capture the exponent of the sublinear asymptotic behaviour of $\mathrm{d}_N^{\min}$. Notice that

$$E_N^x = \{ X_N \leq x \} .$$

Our main results about $(X_N)_{N\in\mathbb{N}}$ are the two following theorems.

Theorem 4.4 With probability one:

(a) $(X_N)_{N\in\mathbb{N}}$ densely covers $[\alpha,\beta]$;

(b) $\liminf_N X_N = \alpha$;

(c) $\limsup_N X_N = \beta$. □

Proof:

(a) We define for any t,$n \in \mathbb{N}$, and $s = 1, \ldots, 2^t$,

$$B_t^{s,N} := \left\{ X_N \in \left[\alpha + \tfrac{s-1}{2^t}(\beta - \alpha), \alpha + \tfrac{s}{2^t}(\beta - \alpha) \right] \right\},$$

$$B_t^s := \left\{ B_t^{s,N} \text{ i.o.} \right\}, \quad B_t := \textstyle\bigcap_{s=1}^{2^t} B_t^s .$$

From (4.9), we have that

$$\begin{aligned} \mathbb{P}(B_t^{s,N}) &\geq C' N^{\theta\left(\alpha + \frac{s}{2^t}(\beta-\alpha)\right)} - C'' N^{\theta\left(\alpha + \frac{s-1}{2^t}(\beta-\alpha)\right)} \\ &= C' N^{\theta\left(\alpha + \frac{s}{2^t}(\beta-\alpha)\right)} \left(1 - \tfrac{C''}{C'} N^{-\frac{\beta-\alpha}{2^t}} \right) , \end{aligned}$$

so that, since $\theta\left(\alpha + \frac{s}{2^t}(\beta - \alpha)\right) \geq -1$,

$$\sum_{N \in \mathbb{N}} \mathbb{P}\left(B_t^{s,N} \right) = \infty .$$

Thus, part (ii) of the Borel-Cantelli lemma lets us conclude that $\mathbb{P}(B_t^s) = 1$ for any $s = 1, \ldots, 2^t$, and so

$$\mathbb{P}(B_t) = \mathbb{P}\left(\textstyle\bigcap_{s=1}^{2^t} B_t^s \right) = 1 , \quad \forall t \in \mathbb{N} .$$

But then

$$\begin{aligned} \mathbb{P}\left(\{ (X_N)_N \text{ densely covers } [\alpha,\beta] \} \right) &= \mathbb{P}\left(\textstyle\bigcap_{t \in \mathbb{N}} B_t \right) \\ &= \lim_{t \to \infty} \mathbb{P}(B_t) = 1 . \end{aligned}$$

(b) By (4.9) we have that, for every $\varepsilon > 0$

$$\sum_{N \in \mathbb{N}} \mathbb{P}(E_N^{\alpha-\varepsilon}) \leq \sum_{N \in \mathbb{N}} C N^{\theta(\alpha-\varepsilon)} < \infty ,$$

so that part (i) of the Borel-Cantelli lemma implies

$$\mathbb{P}\left(\left\{ E_N^{\alpha-\varepsilon} \text{ i.o.} \right\} \right) = 0 .$$

Denoting by A^c the complement of an event A, we have

$$\left\{ E_N^{\alpha-\varepsilon} \text{ i.o.} \right\}^c \subseteq \left\{ \liminf_{N \in \mathbb{N}} X_N \geq \alpha - \varepsilon \right\}$$

so that

$$\begin{aligned}\mathbb{P}\left(\liminf_{N\in\mathbb{N}} X_N \geq \alpha\right) &= \mathbb{P}\left(\bigcap_{k\in\mathbb{N}} \left\{\liminf_N X_N \geq \alpha - \tfrac{1}{k}\right\}\right) \\ &= \lim_{k\to\infty} \mathbb{P}\left(\left\{\liminf_N X_N \geq \alpha - \tfrac{1}{k}\right\}\right) \\ &\geq \lim_{k\to\infty} \mathbb{P}\left(\left\{E_N^{\alpha-1/k} \text{ i.o.}\right\}^c\right) \\ &= 1\ .\end{aligned}$$

Since by point (a) we have $\mathbb{P}\left(\liminf_N X_N \leq \alpha\right) = 1$, point (b) follows.

(c) Theorem 4.3 directly implies that $\limsup_N X_N \leq \beta$. Since point (a) implies that $\mathbb{P}\left(\limsup_N X_N \geq \beta\right) = 1$, point (c) follows. ■

Although Theorem 4.4 tells us that with probability one a random sequence of codes from the serial turbo ensemble has minimum distance exhibiting a chaotic behaviour, a weak form of convergence for the sequence of r.v.s $(X_N)_N$ can still be observed. Formally, we have to consider the sequence of probability measures instead of the probability space of sequences. We will denote by $X_N \xrightarrow{\mathbb{P}} X$ the convergence in probability (see [12] for definitions and properties). The following result is a restating of [41]'s Theorem 2 in our setting.

Theorem 4.5 $\quad X_N \xrightarrow{\mathbb{P}} \beta$. □

Proof: For every $\varepsilon > 0$, Corollary 4.1 and Theorem 4.3 guarantee that

$$\mathbb{P}(|X_N - \beta| < \varepsilon) \geq 1 - C\, N^{-\frac{d_f^o}{2}\varepsilon} \xrightarrow{N\to\infty} 1\ .$$

■

4.4.2 ML Error probabilities

In order to transfer our results about minimum distances to ML word error probabilities we use a classical tool of coding theory known as expurgation (see [33]). We estimate the averaged error probability conditioned on the complement events $(E_N^x)^c$ for some proper $x \in [0,\beta)$. By combining these estimations with (4.9) we derive strong probabilistic results about the asymptotic behaviour of $P(e|\Pi_N)$.

We define the following random variable

$$Y_N := \frac{\log(-\log P(e|\Pi_N))}{\log N}\ ;$$

the idea is that Y_N should capture the speed of the subexponential asymptotic decrease of $P(e|\Pi_N)$.

Proposition 4.1 If the channel is sufficiently good, for all $x \in [0,\beta)$,

$$\mathbb{E}[P(e|\Pi_N)\,|(E_N^x)^c] \leq \exp(-K_x N^x)$$

for some positive constant K_x. □

Proof: We use the Union-Bhattacharyya bound, remembering that $(E_N^x)^c = \{\mathrm{d}_N^{\min} > K_N^x\}$ and then, denoting by $\mathbb{1}_E$ the characteristic function of some event E:

$$\begin{aligned}\mathbb{E}[P(e|\Pi_N)|(E_N^x)^c] &= \frac{1}{\mathbb{P}((E_N^x)^c)}\mathbb{E}[P(e|\Pi_N)\cdot \mathbb{1}_{(E_N^x)^c}] \\ &\leq \frac{1}{\mathbb{P}((E_N^x)^c)}\sum_{h=K_N^x}^{K_N}\sum_{w=d_f^o}^{\mu_i h}\sum_{l=1}^{\mu_o w}\frac{A_{l,w}^{o,N}A_{w,h}^{i,N}}{\binom{N_N}{w}}\gamma^h\,.\end{aligned}$$

By Corollary 4.1, $\mathbb{P}((E_N^x)^c) \overset{N\to\infty}{\longrightarrow} 1$. So, for some $c \geq 1$,

$$\frac{1}{\mathbb{P}((E_N^x)^c)} \leq c\,.$$

We estimate $A_{w,h}^{i,N} \leq A_{w,\leq h}^{i,N}$ by Lemma 4.4 and $\sum_{l=1}^{\mu_o w} A_{l,w}^{o,N}$ by Lemma 4.2, so we can find a positive C such that:

$$\mathbb{E}[P(e|\Pi_N)|(E_N^x)^c] \leq c\sum_{h=K_N^x}^{K_N}\sum_{w=d_f^o}^{\mu_i h} C^w\Big(\frac{h}{w}\Big)^{\frac{w}{2}}\Big(\frac{w}{N}\Big)^{\frac{w}{2}-\frac{w}{d_f^o}}\gamma^h.$$

Then we remark that the function $g(z) := (a/z)^z$ has maximum value $g(a/e) = e^{a/e}$ and hence

$$(h/w)^{w/2} \leq e^{h/(2e)}\,.$$

Moreover, $w \leq L_N \leq \tilde{c}N$ for some $\tilde{c} \geq 1$, so $(w/N)^{\frac{w}{2}-\frac{w}{d_f^o}} \leq \tilde{c}^{(\frac{1}{2}-\frac{1}{d_f^o})w}$. Hence, as $w \leq \mu_i h$, we can find a constant $\bar{C} \geq 1$ such that:

$$\mathbb{E}[P(e|\Pi_N)|(E_N^x)^c] \leq \sum_{h=K_N^x}^{K_N}(\bar{C}\gamma)^h \leq \bar{c}(\bar{C}\gamma)^{K_N^x}$$

where the last inequality holds true, for some $\bar{c} > 0$, if $\gamma < 1/\bar{C}$. Notice that $\bar{C}\gamma < 1$ also implies that $\bar{c}(\bar{C}\gamma)^{K_N^x} \leq \exp(-K_x N^x)$ for some positive K_x. ■

Lemma 4.10 There exists a constant K such that, deterministically, $P(e|\Pi_N) \geq \exp(-KN^\beta \log N)$. □

Proof: We use the inequality $P(e|\Pi_N) \geq p^{\mathrm{d}_N^{\min}}$, where p is the equivocation probability of the channel (see [23]; e.g. $p = 1/2\,\mathrm{erfc}(\sqrt{E_s/N_0})$ for the BIAWGNC). This, together with Theorem 4.3, gives the result. ■

Lemma 4.11 For any $x \in [0,\beta)$, there exist two positive constants K and C, depending on x but not on N, such that

$$\mathbb{P}\Big(P(e|\Pi_N) \geq \exp(-KN^x)\Big) \geq CN^{\theta(x)}\,.$$

□

Proof: Since $P(e|\Pi_N) \geq p^{\mathrm{d}_N^{\min}}$, by (4.9) we get

$$\begin{aligned}\mathbb{P}\left(P(e|\Pi_N) \geq p^{K_N^x}\right) &\geq \mathbb{P}\left(\mathrm{d}_N^{\min} \leq K_N^x\right)\\ &= \mathbb{P}\left(E_N^x\right)\\ &\geq CN^{\theta(x)}\,.\end{aligned}$$

■

Lemma 4.12 For a sufficiently good channel, for any $x \in [0,\beta)$, there exist two positive constants K and K', depending on x but not on N, such that

$$\mathbb{P}\Big(P(e|\Pi_N) \geq \exp(-KN^x)\Big) \leq K'N^{\theta(x)}\,.$$

□

Proof: By Proposition 4.1 we have, for some $K_x > 0$

$$\mathbb{E}\left[P(e|\Pi_N)\,|(E_N^x)^c\right] \leq \exp(-K_xN^x)\,,$$

so that, by Markov inequality, we get

$$\begin{aligned}&\mathbb{P}\left(P(e|\Pi_N) \geq \exp(-\tfrac{K_x}{2}N^x)\Big|\,(E_N^x)^c\right)\\ &\leq \mathbb{P}\left(P(e|\Pi_N) \geq \frac{\mathbb{E}\left[P(e|\Pi_N)|\,(E_N^x)^c\right]}{\exp(-\frac{K_x}{2}N^x)}\,\middle|\,(E_N^x)^c\right)\\ &\leq \exp(-\tfrac{K_x}{2}N^x)\,.\end{aligned}$$

Thus, by (4.9) we get

$$\begin{aligned}&\mathbb{P}\left(P(e|\Pi_N) \geq \exp(-\tfrac{K_x}{2}N^x)\right)\\ &= \mathbb{P}\left(P(e|\Pi_N) \geq \exp(-\tfrac{K_x}{2}N^x)\Big|\,E_N^x\right)\mathbb{P}(E_N^x)+\\ &\quad+\mathbb{P}\left(P(e|\Pi_N) \geq \exp(-\tfrac{K_x}{2}N^x)\Big|\,(E_N^x)^c\right)\mathbb{P}((E_N^x)^c)\\ &\leq \mathbb{P}(E_N^x)+\mathbb{P}\left(P(e|\Pi_N) \geq \exp(-\tfrac{K_x}{2}N^x)\Big|\,(E_N^x)^c\right)\\ &\leq CN^{\theta(x)}+\exp(-\tfrac{K_x}{2}N^x)\end{aligned}$$

and the claim immediately follows with $K = K_x/2$, and for some $K' \geq C$. ■

Theorem 4.6 For a sufficiently good channel, with probability one it holds true:

(a) $(Y_N)_{N\in\mathbb{N}}$ densely covers $[\alpha,\beta]$;

(b) $\liminf_N Y_N = \alpha$;

(c) $\limsup_N Y_N = \beta$. □

Proof:

(a) The proof is rather technical and we omit it, but the main ideas are similar to those of the proof of Thm. 4.4 (a)

(b) For every $\varepsilon > 0$, by Lemma 4.12 we get

$$\sum_{N\in\mathbb{N}} \mathbb{P}\Big(P(e|\Pi_N) \geq \exp(-KN^{\alpha-\varepsilon})\Big) \leq \sum_{N\in\mathbb{N}} K' N^{\theta(\alpha-\varepsilon)} < \infty$$

Then point (i) of the Borel-Cantelli lemma implies

$$\mathbb{P}\left(\left\{P(e|\Pi_N) \geq \exp(-KN^{\alpha-\varepsilon})\right\} \text{ i.o.}\right) = 0$$

so that

$$\begin{aligned}&\mathbb{P}\left(\liminf_N Y_N \geq \alpha - \varepsilon\right)\\ &\geq \mathbb{P}\left(\left\{\left\{P(e|\Pi_N) \geq \exp(-KN^{\alpha-\varepsilon})\right\} \text{ i.o.}\right\}^c\right) = 1\,,\end{aligned}$$

and

$$\begin{aligned}&\mathbb{P}\left(\liminf_N Y_N \geq \alpha\right)\\ &= \mathbb{P}\left(\bigcap_{k\in\mathbb{N}} \left\{\liminf_N Y_N \geq \alpha - 1/k\right\}\right)\\ &= \lim_{k\to\infty} \mathbb{P}\left(\liminf_N Y_N \geq \alpha - 1/k\right) = 1\,.\end{aligned} \tag{4.10}$$

Moreover, by Lemma 4.11

$$\sum_{N\in\mathbb{N}} \mathbb{P}\left(P(e|\Pi_N) \geq \exp(-KN^{\alpha})\right) \geq \sum_{N\in\mathbb{N}} CN^{\theta(\alpha)} = \infty$$

and thus, by point (ii) of the Borel-Cantelli lemma:

$$\begin{aligned}&\mathbb{P}\left(\liminf_N Y_N \leq \alpha\right)\\ &\geq \mathbb{P}\Big(\left\{P(e|\Pi_N) \geq \exp(-KN^{\alpha})\right\} \text{i.o.}\Big) = 1\end{aligned}$$

(c) Lemma 4.10 implies that, deterministically

$$\limsup_N Y_N \leq \beta\,.$$

Moreover, for every $\varepsilon > 0$, by Lemma 4.12 we have

$$\mathbb{P}\Big(P(e|\Pi_N) \geq \exp(-KN^{\beta-\varepsilon})\Big) \leq CN^{\theta(\beta-\varepsilon)} \overset{N\to\infty}{\longrightarrow} 0\,.$$

Thus a subsequence $(\Pi_{N_k})_{k\in\mathbb{N}}$ exists such that

$$\sum_{k\in\mathbb{N}} \mathbb{P}\Big(P(e|\Pi_{N_k}) \geq \exp(-KN_k^{\beta-\varepsilon})\Big) < \infty\,,$$

so that part (i) of the Borel-Cantelli lemma implies

$$\begin{aligned}
&\mathbb{P}\left(\limsup_N Y_N \geq \beta - \varepsilon\right) \\
&\geq \mathbb{P}\Big(\big\{P(e|\Pi_N) \geq \exp(-KN^{\beta-\varepsilon})\big\}\ \text{i.o.}\Big) \\
&\geq \mathbb{P}\Big(\big\{P(e|\Pi_{N_k}) \geq \exp(-KN_k{}^{\beta-\varepsilon})\big\}\ \text{i.o.}\Big) = 1\,.
\end{aligned}$$

By essentially the same derivation as in (4.10), we get $\mathbb{P}\left(\limsup_N Y_N \geq \beta\right) = 1$. ■

Theorem 4.7 For a sufficiently good channel

$$Y_N \overset{\mathbb{P}}{\to} \beta\,.$$

□

Proof: This follows from Lemmas 4.10 and 4.12. ■

4.5 Generalizations

In this section, we discuss how the simplifying assumptions we made can be removed. To keep this section readable, we will address these issues one at the time, not explicitly address how to remove them altogether, e.g. taking odd d_f^o and ϕ_i with non-scalar input and non-proper, even though this is clearly possible.

4.5.1 $d_f^o = 2$

Throughout Sections 4.3 and 4.4 we have assumed $d_f^o > 2$. Now, for the case $d_f^o = 2$, let's look more in detail at which results are still true.

The deterministic upper bound (Theorem 4.3) holds true also for $d_f^o = 2$, where it gives a logarithmic upper bound on the minimum distance.

The upper bound on the left tail of the minimum distance (Theorem 4.1) in this case is not true, and following the same steps as the proof of Theorem 4.1 we get

$$\mathbb{P}(\mathrm{d}_N^{\min} \leq d) \leq m_f^o \left(\frac{2e}{\sqrt{r}}\right)^{d_f^o} N^{1-d_f^o/2} \left\lfloor \frac{d}{d_2^i} \right\rfloor^{d_f^o/2} + \sum_{w=d_f^o+1}^{\mu_i d} C^w N^{\lfloor w/d_f^o \rfloor - \lceil w/2 \rceil} d^{\lceil w/2 \rceil}$$

where now every term in the summation having even w does not decrease in N, and so the bound grows to infinity and becomes trivial.

The lower bound on the left tail of the minimum distance (Theorem 4.2) is still true, however it could become trivial with a right side getting negative. A more careful choice of the indexes set J gives the more interesting result that $\mathbb{P}(\mathrm{d}_N^{\min} \leq d_2^i)$ is bounded away from zero. In fact, if you replace $J := \{d_f^o \eta_o \, i \,,\ i \in \mathbb{Z}^+\} \cap \{0,1,\ldots,N-1-d_f^o\eta_o\}$ with $J := \{C\, i \,,\ i \in \mathbb{Z}^+\} \cap \{0,1,\ldots,N-1-C\}$ where $C \geq d_f^o \eta_o$ is a constant chosen large enough to ensure that $1 - \frac{2e^2}{rC} > 0$, following the proof of Theorem 4.2

$$\mathbb{P}(\mathrm{d}_N^{\min} \leq d_2^i) \geq \frac{1}{re^2 C}\left(1 - \frac{2e^2}{rC}\right) > 0$$

Clearly this means that Theorems 4.5 and 4.7 hold true (with $\beta = 0$) and, even more, $X_N \to 0$ and $Y_N \to 0$ deterministically. More precisely, the fact that $\mathbb{P}(\mathrm{d}_N^{\min} \leq d_2^i) \geq c$ for some positive constant c implies that also $\mathbb{P}(P_w(e) \geq p^{d_2^i}) \geq c > 0$ and that $\mathbb{E}P_w(e) \geq cp^{d_2^i} > 0$, where p is the equivocation probability of the channel. Finally note that the logarithmic upper bound implies that deterministically $P_w(e) \geq \tilde{c}/N$ for some positive constant $\tilde{c}$.

4.5.2 Non-scalar ϕ_i

When the inner encoder has non-scalar input ($s \neq 1$), we need to modify the results in Section 4.2.3. Let's start by looking at how Lemma 4.3 generalized to and encoder $\phi \in \mathbb{Z}_2^{l\times s}((D))$. Clearly Lemma 4.3 is still true for the s scalar-input encoders corresponding to columns of ϕ, so we know that also non-scalar recursive encoders admit input words of weight 2 producing finite-weight output (thus ensuring $d_2 < \infty$). But we can say more about input-weight-2 codewords of ϕ. Let's define:

- δ_{ij} the value of $\bar{\delta}$ obtained applying Lemma 4.3 to ϕ_{ij};
- δ_j the value of $\bar{\delta}$ obtained applying Lemma 4.3 to the j-th column of ϕ;
- $\mathcal{D}_{ij} = \left\{\delta : \mathrm{w_H}(\phi(e_i + D^\delta e_j)) < \infty\right\}$

Note that $\delta_j = \mathrm{lcm}\{\delta_{ij}, i : \phi_{ij} \text{ is recursive}\}$. Also note that $\mathcal{D}_{ij}$ could be empty or non-empty, depending on the encoder ϕ. The following result characterizes non-empty $\mathcal{D}_{ij}$'s. Suppose you write $\phi_{ij} = p_{ij}(D)/q_{ij}(D)$ with $\gcd(p_{ij},q_{ij}) = 1$ and with $q_{ij}(0) = 1$ (clearly you can always do so); then the following property holds.

Proposition 4.1 If $\mathcal{D}_{ij}$ is not empty, then:

1. $\delta_i = \delta_j$

2. $\mathcal{D}_{ij} = \tilde{\delta}_{ij} + \delta_i \mathbb{N}^*$, where $\tilde{\delta}_{ij} := \min \mathcal{D}_{ij}$;

3. $q_{mi} = q_{mj}$ for all $m = 1, \ldots, l$;

4. $\tilde{\delta}_{ij} < \delta_i$;

5. $\mathcal{D}_{ji} \neq \emptyset$ and, more precisely, $\mathcal{D}_{ji} = \tilde{\delta}_{ji} + \delta_i \mathbb{N}^*$ with $\tilde{\delta}_{ji} = \delta_i - \delta_{ij}$.

Proof: Assume ϕ is recursive, i.e. every column of ϕ has at least one entry which is a recursive scalar input - scalar output encoder. If not so, the proposition is trivially true.

To prove statement 1., say that $\frac{p_{ij}(D)}{q_{ij}(D)} + \frac{D^\delta p_{i'j'}(D)}{q_{i'j'}(D)} = r(D) \in \mathbb{Z}_2[D]$. This implies $p_{ij}(D)q_{i'j'}(D) + D^\delta p_{i'j'}(D)q_{ij}(D) = r(D)q_{ij}(D)q_{i'j'}(D)$, which in turn implies that $q_{ij}(D)$ both divides and is divided by $q_{i'j'}(D)$; in $\mathbb{Z}_2[D]$ this gives $q_{ij}(D) = q_{i'j'}(D)$.

To prove statements 2. and 3. we use two steps. The first claim we prove is that $a \in \mathcal{D}_{ij}$ implies $a + n_1\delta_i + n_2\delta_j \in \mathcal{D}_{ij}$ for any n_1,$n_2 \in \mathbb{N}^*$. To see this, write: $\phi\big(e_i + D^{a+n_1\delta_i+n_2\delta_j} e_j\big) = \phi\big(e_i + D^{n_1\delta_i} e_i\big) + \phi\big(D^{n_1\delta_i}(e_i + D^a e_j)\big) + \phi\big(D^{a+n_1\delta_i}(e_j + D^{n_2\delta_j} e_j)\big) \in \mathbb{Z}_2[D]$.

If $\mathcal{D}_{ij} \neq \emptyset$, this claim gives us

$$\tilde{\delta}_{ij} + \delta_i \mathbb{N}^* + \delta_j \mathbb{N}^* \subseteq \mathcal{D}_{ij} \tag{4.11}$$

The second claim is that

$$\mathcal{D}_{ij} \subseteq \tilde{\delta}_{ij} + \delta_j \mathbb{N}^* \tag{4.12}$$

and

$$\mathcal{D}_{ij} \subseteq \tilde{\delta}_{ij} + \delta_i \mathbb{N}^* \,. \tag{4.13}$$

To prove the first inclusion, note that, if $a \in \mathcal{D}_{ij}$, then $\phi\big(D^{\tilde{\delta}_{ij}}(1 + D^{a-\tilde{\delta}_{ij}})e_j\big) = \phi\big(e_i + D^a e_j\big) + \phi\big(e_i + D^{\tilde{\delta}_{ij}} e_j\big) \in \mathbb{Z}_2^l[D]$ which implies that $a - \tilde{\delta}_{ij} \in \delta_k \mathbb{N}^*$. To prove the second inclusion, note that if $a \in \mathcal{D}_{ij}$, then $\phi\big((1 + D^{a-\tilde{\delta}_{ij}})e_i\big) = \phi\big(e_i + D^a e_j\big) + \phi\big(D^a(1 + D^{a-\tilde{\delta}_{ij}})e_j\big) \in \mathbb{Z}_2^l[D]$ which implies that $a - \tilde{\delta}_{ij} \in \delta_i \mathbb{N}^*$.

Finally, Equations (4.11), (4.12) and (4.13) altogether prove statements 2. and 3.

To prove statement 4., use a simple argument by contradiction: if it was $\tilde{\delta}_{ij} \geq \delta_i$, then we would have also $\tilde{\delta}_{ij} - \delta_i \in \mathcal{D}_{ij}$, because $D^{\delta_i}\phi(e_i + D^{\tilde{\delta}_{ij}-\delta_i}) = \phi(e_i + D^{\delta_i}) + \phi(e_i + D^{\tilde{\delta}_{ij}} e_j) \in \mathbb{Z}_2[D]$.

To prove the last statement, we start noticing that $\delta_i - \tilde{\delta}_i \in \mathcal{D}_{ji}$. In fact, $D^{\tilde{\delta}_{ij}}\phi(e_j + D^{\delta_i - \tilde{\delta}_{ij}}) = \phi(e_i + D^{\tilde{\delta}_{ij}} e_j) + \phi(e_i + D^{\delta_i} e_i) \in \mathbb{Z}_2[D]$. Then, by statement 2. $\mathcal{D}_{ji} = \tilde{\delta}_{ji} + \delta_i \mathbb{N}^*$ for some $\tilde{\delta}_{ji} \leq \delta_i - \tilde{\delta}_{ij}$. The equality $\tilde{\delta}_{ji} = \delta_i - \tilde{\delta}_{ij}$ is true, because $\phi(e_j + D^{\tilde{\delta}_{ij}+\tilde{\delta}ji} e_j) = \phi(e_j + D^{\tilde{\delta}ji} e_i) + D^{\tilde{\delta}ji}\phi(e_i + D^{\tilde{\delta}ij} e_j) \in \mathbb{Z}_2[D]$ and so $\tilde{\delta}_{ij} + \tilde{\delta}ji \in \delta_i \mathbb{N}^*$. ■

Clearly $d_2 = \min\{\min_i \mathrm{w_H}(\phi(e_i + D^{\delta_i} e_i)), \min_{i,j:\mathcal{D}_{ij}\neq\emptyset} \mathrm{w_H}(\phi(e_i + D^{\tilde{\delta}_{ij}} e_j))\}$. When computing the average error probability of the serial uniform interleaver ensemble, this is the relevant parameter to consider (see e.g. [3], and see also Chapter 3). However, when dealing with typical instead of worse case, the most relevant parameters are all $d_{2,i} := \mathrm{w_H}(\phi(e_i + D^{\delta_i} e_i))$. In fact, for all $a \in \mathbb{N}^*$, we have $\mathrm{w_H}(e_i + D^{a\delta_i} e_i) \leq a\mathrm{w_H}(\phi(e_i + D^{\delta_i} e_i))$, with equality if the i-th column of ϕ has only proper fractions, and also $\mathrm{w_H}(e_i + D^{\tilde{\delta}_{ij} + a\delta_i} e_j) \leq \mathrm{w_H}(e_i + D^{\tilde{\delta}_{ij}}) + a \min\left(\mathrm{w_H}(\phi(e_i + D^{\delta_i} e_i)), \mathrm{w_H}(\phi(e_j + D^{\delta_i} e_j))\right)$ If ϕ is proper, i.e. $\deg p_{ij} < \deg q_{ij}$ for all i and j, it's straightforward to prove the following property.

Proposition 4.2 If $\mathcal{D}_{ij} \neq \emptyset$, define $s_i(D) = \phi(e_i + D^{\delta_i} e_i)$, $s_{ij}(D) = \phi(e_i + D^{\tilde{\delta}_{ij}} e_j)$. If ϕ is proper rational, then:

1. $\deg s_{ij} \leq \tilde{\delta}_{ij} < \delta_i$;
2. $\mathrm{w_H}(s_i) = \mathrm{w_H}(s_j) = \mathrm{w_H}(s_{ij}) + \mathrm{w_H}(s_{ji})$;
3. $\mathrm{w_H}(\phi(e_i + D^{\tilde{\delta}_{ij} + a\delta_i} e_j)) = \mathrm{w_H}(s_{ij}) + a\mathrm{w_H}(s_i) = (a+1)\mathrm{w_H}(s_{ij}) + a\mathrm{w_H}(s_i)$.

■

Now we have the tools to see how the estimations are modified for non-scalar-input inner encoder.

Consider the upper bounds of the inner weight enumerating coefficients given in Lemma 4.4. The modified version is:

Lemma 4.13 (Lemma 4.4 for non-scalar ϕ_i) Let $1 \leq w \leq \mu_i d$, $1 \leq d \leq K_N$.

- If w is even,
$$R^{i,N}_{w,\leq d,w/2} \leq s^w \frac{(2e)^w}{w^w} M_N^{w/2} \left\lfloor \frac{d}{d_2^i} \right\rfloor^{w/2}$$
- If w is even, for $N \to \infty$, if $d = o(N)$,
$$A^{i,N}_{w,\leq d} = R^{i,N}_{w,\leq d,w/2} + o\left(\frac{C^w}{w^w} (sM_N)^{w/2} \left\lfloor \frac{d}{d_2^i} \right\rfloor^{w/2} \right)$$
where $C = 2e^3 s^2 \sqrt{e(l+1)\eta_i}$.
- If w is odd, for $N \to \infty$, if $d = o(N)$,
$$A^{i,N}_{w,\leq d} = O\left(s^{w/2} \frac{C^w}{w^w} N^{\lfloor w/2 \rfloor} \left\lfloor \frac{d}{d_2^i} \right\rfloor^{\lfloor w/2 \rfloor - 1} d^2 \right)$$
where $C = \max\left(2e^3 s^2 \sqrt{(l+1)\eta_i}, s\mu_i^2 \sqrt{e}, 2se^2\eta_i\right)$ □

The bound for $R_{w,\le d,w/2}$ in Lemma 4.4 is generalized by counting words made of $w/2$ pairs of ones, each of this pairs being of the form $D^{b_t}\big(e_j(1+D^{a_t\delta_j})\big)$ or $D^{b_t}\big(e_i+D^{\tilde{\delta}_{ij}+a_t\delta_i}e_j\big)$ for some i,j such that $\mathcal{D}_{ij}\neq\emptyset$. By imposing $\sum_{t=1}^{w/2}a_t\le\frac{d}{d_2^i}$ we ensure that the total output weight is $\le d$ (overcounting; a tighter but very cumbersome bound could be obtained involving all $d_2^i(i)$'s and $d_2^i(ij)$'s). The number of possible times to begin the $w/2$ events is bounded by $\binom{M_N}{w/2}$ and finally the term s^w takes into account the choice of the component where the w ones are (if $\mathcal{D}_{ij}=\emptyset$ for all $i\neq j$, we can replace s^w by $s^{w/2}$).

For the remaining terms $R^{i,N}_{w,\le d,n}$ and $T^{i,N}_{w,\le d,n}$, follow the proof of Lemma 4.4 being careful to distinguish M_N from $L_N=sM_N$. The only tricky point is the estimation (4.5) and its analogous (4.6): it is essential to get $\binom{d\eta_i}{w-n}$ and not just $\binom{d\eta_i}{w}$, and for scalar-input ϕ_i this is obtained by noting that surely the first bit of each error event cannot be zero, so it is known to be one. Now, the first vector (element of $\mathbb{Z}_2^s$) of an error event is surely not all-zero, but it can still have 2^s-1 different values. However, with some careful estimation, we can replace (4.5) by the following:

$$R^{i,N}_{\boldsymbol{w},\boldsymbol{b},\le d,n}\le s^{2w}\left[\frac{ed\eta_i}{w-n}\right]^{w-n} \tag{4.14}$$

which is tight enough to prove Lemma 4.13. To prove (4.14), let's define $R^{i,N}_{\boldsymbol{w},\boldsymbol{b},\boldsymbol{v},\le d,n}$ to be the number of inner codewords with output weight $\le d$, made of n regular error events, where $\boldsymbol{w}=w_1,\dots,w_n$ are their input weights, $\boldsymbol{b}=b_1,\dots,b_n$ are their beginning time ($b_t\in[M_N]$), and $\boldsymbol{v}=v_1,\dots,v_n$ is the weight of their first vector (i.e. v_t is the weight of the vector, element of $\mathbb{Z}_2^s$ which is the input at time b_t). Clearly, for all t, $1\le v_t\le s$ and also $v_t\le w_t$ so that $\sum_t v_t\le\sum_t w_t=w$. Now we estimate $R^{i,N}_{\boldsymbol{w},\boldsymbol{b},\boldsymbol{v},\le d,n}\le\binom{s}{v_1}\dots\binom{s}{v_n}\binom{sd\eta_i}{w-\sum_{t=1}^n w_t}$. Note that $\binom{s}{v_1}\dots\binom{s}{v_n}\le s^{\sum_{t=1}^n w_t}\le s^w$. For the remaining term, we estimate $\binom{sd\eta_i}{w-\sum_{t=1}^n w_t}\le\left[\frac{esd\eta_i}{w-\sum_{t=1}^n w_t}\right]^{w-\sum_{t=1}^n w_t}$. Now note that the function $g(z):=(a/z)^z$ is an increasing function of z for $z\le a/e$, and here we have it with $a=esd\eta_i$ and $z=w-\sum_{t=1}^n w_t\le w-n<w\le sd\eta_i=a/e$, so that $\left[\frac{esd\eta_i}{w-\sum_{t=1}^n w_t}\right]^{w-\sum_{t=1}^n w_t}\le\left[\frac{esd\eta_i}{w-n}\right]^{w-n}$. Putting everything together:

$$R^{i,N}_{\boldsymbol{w},\boldsymbol{b},\le d,n}=\sum_{\substack{v_1,\dots,v_n\\1\le v_t\le s}}R^{i,N}_{\boldsymbol{w},\boldsymbol{b},\boldsymbol{v},\le d,n}\le s^n\,s^w\left[\frac{esd\eta_i}{w-n}\right]^{w-n}$$

The same bound can be proved also for $T^{i,N}_{\boldsymbol{w},\boldsymbol{b},\le d,n}$. ■

The lower bound on the enumerating coefficients given in Lemma 4.5 is easily extended to non-scalar input ϕ_i.

Lemma 4.14 (Lemma 4.5 for non-scalar-input ϕ_i) If w is even, $2 \leq w \leq \frac{N}{\eta_o}$ and $\frac{d_2^i w}{2} \leq d \leq \frac{d_2^i M_N}{2\delta^i}$

$$R^{i,N}_{w,\leq d,w/2} \geq \sum_{j=1}^{s} \binom{M_N - \delta_j^i \lfloor d/d_2^i(j) \rfloor}{w/2} \binom{\lfloor d/d_2^i(j) \rfloor}{w/2}$$

which implies also

$$R^{i,N}_{w,\leq d,w/2} \geq s \frac{2^{w/2}}{w^w} M_N^{w/2} \left\lfloor \frac{d}{d_2^i} \right\rfloor^{w/2}$$

□

Proof: Simply apply the proof of Lemma 4.5 to every column of ϕ_i separately.■

A slightly tighter bound could be obtained by counting also words with error events in different components, similarly to what is done in the upper bound.

These two modified lemmas allow to obtain easily the results in Theorems 4.2 and 4.1, only with suitably modified constants.

Also the deterministic upper bound (Theorem 4.3) generalizes to non-scalar-input ϕ_i:

Theorem 4.8 (Theorem 4.3 for non-scalar-input ϕ_i) If $M_N \geq \max(d_f^o \eta_o, \frac{1}{2} d_f^o (\max_j \delta_j^i)$

$$\mathrm{d}_N^{\min} \leq \frac{1}{2}(d_f^o)^2 (\max_j d_2^i(j)) \log b \left\lfloor \frac{N}{b} \right\rfloor \quad \text{where } b = \left(\frac{1}{4} \left\lceil \frac{1}{(\prod_{j=1}^s \delta_j^i)^{d_f^o}} \left\lfloor \frac{N}{d_f^o \eta_o} \right\rfloor \right\rceil \right)^{2/d_f^o}$$

This also implies that, for sufficiently big N,

$$\mathrm{d}_N^{\min} \leq 2(\max_j d_2^i(j)) d_f^o (\prod_{j=1}^{s} \delta_j^i)^2 \left(4(d_f^o \eta_o + 1)\right)^{2/d_f^o} N^{\beta} \log N$$

□

Proof: In the proof of Theorem 4.3, change the definition of $\boldsymbol{\sigma}$, letting $\boldsymbol{\sigma} : J \to (\mathbb{Z}_{\delta_1} \times \mathbb{Z}_{\delta_2} \times \cdots \times \mathbb{Z}_{\delta_s})^{d_f^o}$ be the function that associates to the d_f^o ones of the outer codeword $\boldsymbol{c}_j^*$ the position where they end up after the permutation, in the following way: if a one goes to component k at time t (we look at $\mathbb{Z}_2^{sM_N}$ as $(\mathbb{Z}_2^s)^{M_N}$, M_N being the time on the trellis) we put a 0 in all components except the k-th, where we put $t \mod \delta_k$. In this way, $\boldsymbol{\sigma}(i) = \boldsymbol{\sigma}(j)$ means that for all $k = 1, \ldots, d_f^o$, the k-th one of the codewords $\boldsymbol{c}_i^*$ and $\boldsymbol{c}_j^*$ are permuted to the same component, say m_k, and at distance multiple of δ_{m_k}, say $a_k \delta_{m_k}$ so each pair produces an output with weight upper bounded by $a_k d_2^i(k) < (a_k \delta_{m_k})(\max_k d_2^i(k))$.

By the pigeonhole principle, we can find $U \subseteq J$ such that $\boldsymbol{\sigma}(i) = \boldsymbol{\sigma}(j)$ for all $i,j \in U$, with $|U| \geq \frac{|J|}{(\delta_1^i \delta_2^i \ldots \delta_s^i)^{d_f^o}}$. With this new U you can apply all the proof of Theorem 4.3, simply replacing δ^i by $\prod_{j=1}^s \delta_s^i$ and replacing d_2^i with $\max_{j=1}^s d_2^i(j)$. ■

4.5.3 ϕ_i not proper rational

Now we want to relax the assumption that the degree of the numerator of ϕ_i is strictly smaller than the degree of the denominator, which was used to obtain the equality $\mathrm{w_H}(\phi(1+D^{a\delta_i}) = ad_2^i$ (Lemma 4.3). For simplicity we focus again on the scalar-input case.

Note that if both input and output are scalar, than non-catastrophicity and recursiveness together imply that $\phi_i(D) = 1/q(D)$, so there is no need to study non-proper rational encoders. However this is no longer the case when the output is non-scalar.

Clearly we have $\mathrm{w_H}(\phi(1+D^{a\delta_i}) \leq ad_2^i$ and, by definition of I_i, $\mathrm{w_H}(\phi(1+D^{a\delta_i}) \geq aI_i$, so Lemma 4.5 and Theorem 4.3 are still true, and Lemma 4.4 is true if we use I_i instead of d_2^i.

In the following we want to show a parameter which can appear both in the upper and in the lower bounds, at least for the case when the degree of the numerator is not too big. As a notation, say that $\phi_i(D) = \frac{1}{q(D)[p_1(D),\ldots,p_l(D)]}$ with $\gcd(q,p_1,\ldots,p_l) = 1$ and let $\frac{p_j(D)}{q(D)} = s_j(D) + \frac{r_j(D)}{q(D)}$ with $\deg r_j < \deg q$, s_j possibly zero. Denote $s(D) := [s_1(D),\ldots,s_l(D)]^T$. Define

$$L_i := \mathrm{w_H}\left(\frac{1}{q(D)}[r_1(D),\ldots,r_l(D)]^T(1+D^{\delta^i})\right)$$

i.e. L_i is the d_2 of the proper rational encoder $\frac{1}{q(D)}[r_1(D),\ldots,r_l(D)]^T$.

Lemma 4.15 If $\deg s < m\delta$, then:

$$L_i \leq \mathrm{w_H}(\phi(1+D^{a\delta})) \leq aL_i + 2\mathrm{w_H}(s(D))\,.$$

Moreover,

$$L_i = \lim_{a\to\infty} \frac{\mathrm{w_H}(\phi(1+D^{a\delta}))}{a}$$

□

Proof: Consider separately each of the scalar encoders $s_j(D) + \frac{r_j(D)}{q_j(D)}$. Note that $(s_j(D) + \frac{r_j(D)}{q_j(D)})(1+D^{a\delta}) = s_j(D) + \sum_{k=0}^{a-1} D^{k\delta}(1+D^{\delta})\frac{r_j(D)}{q(D)} + D^a s_j(D)$. Clearly $(1+D^{\delta})\frac{r_j(D)}{q(D)}$,$D^{\delta}(1+D^{\delta})\frac{r_j(D)}{q(D)}$,$\ldots$,$D^{(a-1)\delta}(1+D^{\delta})\frac{r_j(D)}{q(D)}$ and also $D^{a\delta}s_j(D)$ all have disjoint supports thanks to the assumption $\deg s < m\delta$, so the only part that overlaps and could cancel some bits in the summation is $s_j(D)$. Thus:

$$\mathrm{w_H}((s_j(D) + \frac{r_j(D)}{q_j(D)})(1+D^{a\delta})) \leq 2\mathrm{w_H}(s_j(D)) + a\mathrm{w_H}((1+D^{\delta})\frac{r_j(D)}{q(D)})$$

and

$$\mathrm{w_H}((s_j(D)+\frac{r_j(D)}{q_j(D)})(1+D^{a\delta})) \geq (\mathrm{w_H}(s_j(D)) + a\mathrm{w_H}((1+D^{\delta})\frac{r_j(D)}{q(D)})) - \mathrm{w_H}(s_j(D))$$

This applied for all j gives the bound. The limit follows immediately. ■

Note that in Lemma 4.15 the assumption $\deg s < m\delta$ is needed only for the lower bound, not for the upper. Also note that if $\deg s < \delta$, the bound in Lemma 4.15 is true for all values of m, and so $I_i = L_i$, which allows to have L_i appear in Lemma 4.4 without any further calculation. Moreover, in this case you can notice that $s_j(D)$ overlaps only with the first term in the summation $\sum_{k=0}^{a-1} D^{k\delta}(1+D^{\delta})\frac{r_j(D)}{q(D)}$, so that another interesting upper bound can be derived from the proof of Lemma 4.15:

$$\mathrm{w_H}(\phi(1+D^{a\delta_i})) \leq d_2^i + (a-1)L_i\,.$$

Lemma 4.5 and Theorem 4.3 are easily modified using the upper bound in Lemma 4.15. Lemma 4.5 now gives

$$R^{i,N}_{w,\leq d,w/2} \geq \binom{M_N - \delta^i\lfloor d/d_2^i\rfloor}{w/2}\binom{\lfloor\frac{d-w\mathrm{w_H}(s)}{L_i}\rfloor}{w/2}$$

which makes sense with the additional assumption that $w = o(d)$ for $N \to \infty$, which is clearly verified for the constant $w = d_f^o$ (this is the only use we make of this lemma). To prove it, simply modify the proof of Lemma 4.4 by choosing

$$1 \leq h_1 < h_2 < \cdots < h_{w/2} \leq \lfloor\frac{d - w\mathrm{w_H}(s)}{L_i}\rfloor$$

because $\mathrm{w_H}\big(\phi_i((1+D^{\delta^i(h_t-h_{t-1})}))\big) \leq L^i\,(h_t - h_{t-1}) + \frac{w}{2}2\mathrm{w_H}(s)$, and then the total output weight of the $w/2$ events is less then $L^i\sum_{t=1}^{w/2}(h_t - h_{t-1}) = L^i h_{w/2} + w\mathrm{w_H}(s(D)) \leq d$. The deterministic upper bound can also be modified, keeping all the proof the same, except that after the codeword construction the bound on the weight uses Lemma 4.15: $\mathrm{w_H}\big(\phi_i^N(\pi(c))\big) \leq \sum_{m=1}^{|S|d_f^o/2} \mathrm{w_H}\big(\phi_i^N(D^{t_{2m-1}} + D^{t_{2m}})\big) \leq$ $L_i\sum_{m=1}^{|S|d_f^o/2}(t_{2m}-t_{2m-1}) + 2\frac{|S|d_f^o}{2}\mathrm{w_H}(s(D)) \leq L_i|S|\frac{d_f^o}{2}\frac{N}{b} + 2\frac{|S|d_f^o}{2}\mathrm{w_H}(s(D))$ Theorem 4.3 now becomes:

$$\mathrm{d}_N^{\min} \leq \frac{1}{2}(d_f^o)^2 L_i\,\log b\left\lfloor\frac{N}{b}\right\rfloor \quad + d_f^o\mathrm{w_H}(s(D))\log b$$

where $b = \left(\frac{1}{4}\left\lceil\frac{1}{\delta^{d_f^o}}\left\lfloor\frac{N}{d_f^o\eta_o}\right\rfloor\right\rceil\right)^{2/d_f^o}$, also implying that, for sufficiently big N,

$$\mathrm{d}_N^{\min} \leq 2d_2^i d_f^o\delta^2\big(4(d_f^o\eta_o+1)\big)^{2/d_f^o}N^{\beta}\log N + 2\mathrm{w_H}(s(D))\log N$$

4.5.4 Odd d_f^o

The relevant parameters now are:

- $\alpha := 1 - \frac{2}{\lceil d_f^o/2 \rceil} = 1 - \frac{4}{d_f^o+1}$;
- $\beta := 1 - \frac{2}{d_f^o}$;
- $\tilde{\beta} := 1 - \frac{1}{\lceil d_f^o/2 \rceil} = 1 - \frac{2}{d_f^o+1}$.

Clearly $0 \leq \alpha < \beta < \tilde{\beta} < 1$, with $\alpha = 0$ if and only if $d_f^o = 3$.

Here Lemmas 4.2 and 4.4 hold true without any modification. Only note that a slightly more tight upper bound for $A^{i,N}_{w,\leq d}$ could be obtained in Lemma 4.4, by carefully evaluating the constants.

We need the following version of Lemma 4.5 for odd w, to be used with $w = d_f^o$. In addition to the parameter d_2^i, we also need here

$$d_1^i := \sup_{t=1,\ldots,M_N} \frac{1}{t} \mathrm{w_H}\big(\phi^{i,N}(D^{M_N-t})\big)$$

where we consider the terminated encoder $\phi^{i,N}$, not the convolutional encoder ϕ^i which would always give $\mathrm{w_H}\big(\phi_i(D^{M_N-t})\big) = +\infty$ by recursiveness. Note that we have the trivial bound $1 \leq d_1^i \leq l$.

Lemma 4.16 (Lemma 4.5 for odd w) If w is odd, for any choice of ϵ_1,$\epsilon_2 \in (0,1)$, if $\epsilon_2 d \leq l\epsilon_1 M_N$,

$$T^{i,N}_{w,\leq d,\lceil w/2 \rceil} \geq \binom{(1-\epsilon_1)M_N - \delta^i \left\lfloor \frac{(1-\epsilon_2)d}{d_2^i} \right\rfloor}{\lfloor w/2 \rfloor} \binom{\left\lfloor \frac{(1-\epsilon_2)d}{d_2^i} \right\rfloor}{\lfloor w/2 \rfloor} \left\lfloor \frac{\epsilon_2 d}{d_2^i} \right\rfloor$$

In particular, this implies that for all $d \leq M_N/(2l)$:

$$A^{i,N}_{w,\leq d} \geq T^{i,N}_{w,\leq d,\lceil w/2 \rceil} \geq \frac{1}{(8w)^w} M_N^{\lfloor w/2 \rfloor} \left\lfloor \frac{d}{d_2^i} \right\rfloor^{\lfloor w/2 \rfloor} \frac{d}{d_1^i}$$

Proof: We count some particular codewords which have a first part with support in $[0, \ldots (1-\epsilon_1)M_N - 1]$, consisting of $(w-1)/2$ error events each of input weight 2 and of total weight $\leq (1-\epsilon_2 d)$ (use Lemma 4.5 to count them); a second part is one single terminating event of input weight 1 and output weight $\leq \epsilon_2 d$. ■

Now we have the tools to see how Theorems 4.1 and 4.2 generalize.

Theorem 4.9 (Theorem 4.1, odd d_f^o) For $N \to \infty$, if $d = o(N^\beta)$, then

$$\mathbb{P}(\mathrm{d}_N^{\min} \leq d) = O\left(N^{1-\lceil d_f^o/2 \rceil} d^{\lceil d_f^o/2 \rceil}\right) + O\left(N^{2-d_f^o} d^{d_f^o}\right)$$

□

Proof: From Lemma 4.6, estimating the enumerating coefficients of the constituent encoders with Lemmas 4.2 and 4.4, you get:

$$\mathbb{P}(\mathrm{d}_N^{\min} \leq d) \leq \sum_{w=d_f^o}^{\mu_i d} C^w N^{\lfloor w/d_f^o \rfloor - \lceil w/2 \rceil} d^{\lceil w/2 \rceil} \tag{4.15}$$

for some $C > 0$ depending on ϕ_o and ϕ_i but not growing with N and d. Now we need to separate different terms. If you write $w = ad_f^o + b$, with integers $a \geq 1$, $0 \leq b < d_f^o$, you can see that

$$N^{\lfloor w/d_f^o \rfloor - \lceil w/2 \rceil} d^{\lceil w/2 \rceil} = \begin{cases} \left(\frac{d}{N}\right)^{b/2} \left(N^{1-d_f^o/2} d^{d_f^o/2}\right)^a & \text{if } a+b \text{ is even} \\ \left(\frac{d}{N}\right)^{\frac{b+1}{2}} \left(N^{1-d_f^o/2} d^{d_f^o/2}\right)^a & \text{if } a+b \text{ is odd} \end{cases}$$

For $N \to \infty$, if $d = o(N^\beta)$ then $N^{1-d_f^o/2} d^{d_f^o/2} \to 0$ and so $\sum_a N^{1-d_f^o/2} d^{d_f^o/2}$ converges. So, you need to split the summation in (4.15) in the following four terms (with the notation $[d_f^o] = \{0,1,\ldots,d_f^o - 1\}$):

- $\displaystyle\sum_{\substack{b \in [d_f^o] \\ b \text{ even}}} \left(\frac{d}{N}\right)^{b/2} \sum_{\substack{a \in \mathbb{N}^* \\ a \text{ even}}} \left(N^{1-d_f^o/2} d^{d_f^o/2}\right)^a \leq c_1 N^{2-d_f^o} d^{d_f^o}$
- $\displaystyle\sum_{\substack{b \in [d_f^o] \\ b \text{ odd}}} \left(\frac{d}{N}\right)^{b/2} \sum_{\substack{a \in \mathbb{N}^* \\ a \text{ odd}}} \left(N^{1-d_f^o/2} d^{d_f^o/2}\right)^a \leq c_2 \left(\frac{d}{N}\right)^{1/2} N^{1-d_f^o/2} d^{d_f^o/2}$
- $\displaystyle\sum_{\substack{b \in [d_f^o] \\ b \text{ even}}} \left(\frac{d}{N}\right)^{\frac{b+1}{2}} \sum_{\substack{a \in \mathbb{N}^* \\ a \text{ odd}}} \left(N^{1-d_f^o/2} d^{d_f^o/2}\right)^a \leq c_3 \frac{d}{N} N^{2-d_f^o} d^{d_f^o}$
- $\displaystyle\sum_{\substack{b \in [d_f^o] \\ b \text{ odd}}} \left(\frac{d}{N}\right)^{\frac{b+1}{2}} \sum_{\substack{a \in \mathbb{N}^* \\ a \text{ even}}} \left(N^{1-d_f^o/2} d^{d_f^o/2}\right)^a \leq c_4 \left(\frac{d}{N}\right)^{1/2} N^{1-d_f^o/2} d^{d_f^o/2}$

for some constants c_1,c_2,c_3,$c_4 > 0$.
Finally note that $\frac{d}{N} N^{2-d_f^o} d^{d_f^o} = o\left(N^{2-d_f^o} d^{d_f^o}\right)$. ■

The reason why we keep two terms in the right-hand side of Theorem 4.9 is that, depending on how fast d grows with N, either the first or the second term will be dominating. More precisely, define $\kappa := 1 - \frac{2}{d_f^o - 1}$ and note that for $N \to \infty$, if $d/N^\kappa \to 0$ then $N^{1-\lceil d_f^o/2 \rceil} d^{\lceil d_f^o/2 \rceil} = o\left(N^{2-d_f^o} d^{d_f^o}\right)$ and vice-versa if $d/N^\kappa \to \infty$ then $N^{2-d_f^o} d^{d_f^o} = o\left(N^{1-\lceil d_f^o/2 \rceil} d^{\lceil d_f^o/2 \rceil}\right)$. Also note that $\alpha < \kappa < \beta$, except for $d_f^o = 3$ which gives $0 = \alpha = \kappa < \beta = 1/3$.

A closer look to the proof shows that the term $N^{2-d_f^o} d^{d_f^o}$ corresponds to input weight $2d_f^o$, while the term $N^{1-\lceil d_f^o/2 \rceil} d^{\lceil d_f^o/2 \rceil}$ comes from input weight d_f^o and also from input weight $d_f^o + 1$ (if there is any outer codeword with such weight). This suggests which words we have to count in the different regimes (d below or above N^κ) in order to obtain a tight lower bound for $\mathbb{P}(\mathrm{d}_N^{\min} \leq d)$.

We give here a bound, analogous to Theorem 4.2, which is tight for d below N^κ and will allow us to show the role of α.

Theorem 4.10 (Theorem 4.2 for odd d_f^o) For all $N \geq d_f^o \eta_o$ and $d \geq \frac{d_f^o - 1}{2} d_2^i + d_1^i$

$$\mathbb{P}(\mathrm{d}_N^{\min} \leq d) \geq C_1 \, N \, M_N^{-\left\lceil \frac{d_f^o}{2} \right\rceil} \left\lfloor \frac{d}{d_2^i} \right\rfloor^{\left\lfloor \frac{d_f^o}{2} \right\rfloor} \frac{d}{d_1^i} \left[1 - C_2 \, N \, M_N^{-\left\lceil \frac{d_f^o}{2} \right\rceil} \left\lfloor \frac{d}{d_2^i} \right\rfloor^{\left\lfloor \frac{d_f^o}{2} \right\rfloor} d \right]$$

where $C_1 = \frac{1}{d_f^o \eta_o (8e)^{d_f^o}}$ and $C_2 = \frac{\eta_i}{4\eta_o} (4e)^{d_f^o}$. □

Proof: Modify the definition of the events $E_j^*(d)$ as follows:
$E_j^*(d) := \left\{ \mathrm{w_H}(\phi_i^N(\Pi_N(\boldsymbol{c}_j^*))) \leq d \right\} \cap \{ \phi_i^N(\Pi_N(\boldsymbol{c}_j^*)) \text{ has } \lfloor d_f^o/2 \rfloor \text{ regular and one terminating events } \}$
Clearly $E_j^*(d)$ implies $\mathrm{d}_N^{\min} \leq d$.

A slight modification of Lemma 4.7 gives:

- for all $j \in [0, \ldots, N-T-1]$, $\mathbb{P}(E_j^*(d)) = \dfrac{T^{i,N}_{d_f^o, \leq d, \lceil d_f^o/2 \rceil}}{\binom{L_N}{d_f^o}}$.
- if i and j are such that $|i-j| \geq T$, $i \neq j$,

$$\mathbb{P}(E_i^*(d) \cap E_j^*(d)) \leq \frac{\binom{L_N}{d_f^o}}{\binom{L_N - d_f^o}{d_f^o}} \mathbb{P}(E_i^*(d)) \mathbb{P}(E_j^*(d))$$

Then we estimate $T^{i,N}_{d_f^o,\leq d,\lceil d_f^o/2\rceil}$ from above using Eq. (4.7) from the proof of Lemma 4.4

$$T^{i,N}_{d_f^o,\leq d,\lceil d_f^o/2\rceil} \leq \eta_i \left(\frac{2e}{d_f^o-1}\right)^{d_f^o-1} M_N^{\lfloor d_f^o/2\rfloor} d \lfloor d/d_2^i \rfloor^{\lfloor d_f^o/2\rfloor}$$

while for the lower estimation we use Lemma 4.16. We conclude estimating $\mathbb{P}(\mathrm{d}_N^{\min} \leq d) \geq \sum_{j\in J} \mathbb{P}(E_i^*(d)) - \sum_{\substack{i,j\in J \\ i<j}} \mathbb{P}(E_i^*(d) \cap E_j^*(d))$ as in Theorem 4.2. ■

From Theorems 4.9 and 4.10 we understand the role of α:

$$\sum_N \mathbb{P}(\mathrm{d}_N^{\min} \leq d) < +\infty \text{ if and only if } d = o(N^\alpha)\,.$$

From Theorem 4.9 it is also clear that $d = o(N^\beta)$ implies $\mathbb{P}(\mathrm{d}_N^{\min} \leq d) \to 0$ when $N \to \infty$. However, to generalize Theorems 4.4, 4.5, 4.6 and 4.7 we still need the deterministic upper bound of Theorem 4.3. What we can get is the following.

Theorem 4.11 (Theorem 4.3 for odd d_f^o) For all $N \geq \max(d_f^o\eta_o, \frac{1}{2}d_f^o\delta)$

$$\mathrm{d}_N^{\min} \leq \frac{1}{2}(d_f^o)^2 d_2^i \log b \left\lfloor \frac{N}{b} \right\rfloor \quad \text{where } b = \left\lfloor \frac{1}{4} \left\lceil \frac{1}{\delta^{d_f^o}} \left\lfloor \frac{N}{d_f^o\eta_o} \right\rfloor \right\rceil \right\rfloor^{\frac{1}{\lceil d_f^o/2\rceil}}$$

This also implies that, for sufficiently big N,

$$\mathrm{d}_N^{\min} \leq 2 d_2^i d_f^o \delta^2 \big(4(d_f^o\eta_o+1)\big)^{\frac{2}{d_f^o+1}} N^\beta \log N$$

□

Proof: The proof is the same as for even d_f^o except the way you construct the bipartite graph G from the hypergraph H: now you let $V' = V_1 \times V_2$ with $V_1 = W^{\lceil d_f^o/2\rceil}$ and $V_2 = W^{\lfloor d_f^o/2\rfloor}$ and you put an edge connecting vertices $(v_1,\ldots,v_{\lceil d_f^o/2\rceil}) \in V_1$ and $(w_1,\ldots,w_{\lfloor d_f^o/2\rfloor}) \in V_2$ if and only if there is an edge $(v_1,\ldots,v_{\lceil d_f^o/2\rceil},w_1,\ldots,w_{\lfloor d_f^o/2\rfloor}) \in E$. Note that $|V'| = b^{\lceil d_f^o/2\rceil} + b^{\lfloor d_f^o/2\rfloor} < 2b^{\lceil d_f^o/2\rceil}$ while $|E'|$ is the same as for even d_f^o and satisfies $|E'| \geq \left\lceil \frac{1}{\delta^{d_f^o}} \left\lfloor \frac{N}{d_f^o\eta_o} \right\rfloor \right\rceil$ so you need to choose b satisfying $4b^{\lceil d_f^o/2\rceil} \leq \left\lceil \frac{1}{\delta^{d_f^o}} \left\lfloor \frac{N}{d_f^o\eta_o} \right\rfloor \right\rceil$ in order to apply Lemma 4.9 and conclude the proof. ■

Unfortunately, the exponent $\tilde{\beta}$ in this bound does not match β as it does when d_f^o is even. It is still possible to prove Theorems 4.5 and 4.7 (with the exponent β) by a second-order method used in [41] for even d_f^o. Following the suggestion in [41], for odd d_f^o it is necessary to look at events somehow similar to $E_j^*(d)$ but involving a pair of outer error events $\boldsymbol{c}_i^*,\boldsymbol{c}_j^*$ instead of just one of them. Roughly, what we want

to estimate is the probability that $\mathrm{w_H}\left(\phi_i^N \circ \Pi(\boldsymbol{c}_i^* + \boldsymbol{c}_j^*)\right) \leq d$, but we will choose slightly smaller events in order to simplify the study of the intersections. We define:

$$E_{ij}^*(d) := \Bigg\{ \Pi(\boldsymbol{c}_i^*) = \sum_{t=1}^{d_f^o} D^{b_t} \text{ and } \Pi(\boldsymbol{c}_j^*) = \sum_{t=1}^{d_f^o} D^{b_t + l_t \delta^i} \\ \text{for some } 0 \leq b_1 < \cdots < b_{d_f^o} \leq M_N,\ l_t \geq 1,\ \sum_{t=1}^{d_f^o} l_t \leq \left\lfloor \frac{d}{d_2^i} \right\rfloor \Bigg\}$$

Now define the random variable

$$Z := \sum_{i,j \in I,\, i \neq j} \mathbb{1}_{E_{ij}^*(d)}$$

Clearly

$$\mathbb{P}(\mathrm{d}_N^{\min} \leq d) \geq \mathbb{P}\Big(\bigcup_{i,j \in I,\, i \neq j} E_{ij}^*(d) \Big) = 1 - \mathbb{P}(Z = 0)$$

so all we need is to estimate $\mathbb{P}(Z = 0)$ using the following well-known trick

Lemma 4.17 ([41], Lemma 5) If Z is a r.v. with finite mean and finite variance,

$$\mathbb{P}(Z = 0) \leq \frac{\mathbb{E}(Z^2)}{[\mathbb{E}(Z)]^2} - 1$$

□

Proof: $\mathbb{P}(Z = 0) \leq \mathbb{P}\big(|Z - \mathbb{E}(Z)| \geq \mathbb{E}(Z)\big) \leq \frac{\mathbb{E}[(Z - \mathbb{E}(Z))^2]}{[\mathbb{E}(Z)]^2}$ by Chebyshev inequality.■

From this Lemma and from the definition of our Z, we get

$$\mathbb{P}(\mathrm{d}_N^{\min} \leq d) \geq 2 - \frac{\mathbb{E}(Z^2)}{[\mathbb{E}(Z)]^2} = 2 - \frac{\sum_{\substack{i,j,k,l \in I \\ i \neq j, k \neq l}} \mathbb{P}\big(E_{ij}^*(d) \cap E_{kl}^*(d)\big)}{\left[\sum_{\substack{i,j \in I \\ i \neq j}} \mathbb{P}\big(E_{ij}^*(d)\big)\right]^2} \tag{4.16}$$

The aim now is to prove that, for $N \to \infty$, if $d/N^\beta \to \infty$, then the right side of the above inequality tends to one and so also $\mathbb{P}(\mathrm{d}_N^{\min} \leq d) \to 1$. We do so by using the following estimations:

- a look at the proof of Lemma 4.5 (with $w = 2d_f^o$), gives:

$$\mathbb{P}\big(E_{ij}^*(d)\big) \geq \frac{1}{\binom{L_N}{2d_f^o}} \frac{2^{d_f^o}}{(d_f^o)^{2d_f^o}} L_N^{d_f^o} \left\lfloor \frac{d}{d_2^i} \right\rfloor^{d_f^o}$$

- with a similar proof to Lemma 4.7 (i.e. using the same conditioning trick) you find that, if i,j,k,l are all distinct:

$$\mathbb{P}\big(E^*_{ij}(d) \cap E^*_{kl}(d)\big) \leq \frac{\binom{L_N}{2d^o_f}}{\binom{L_N - 2d^o_f}{2d^o_f}} \mathbb{P}\big(E^*_{ij}(d)\big)\mathbb{P}\big(E^*_{kl}(d)\big)$$

- simple counting gives that, if i,j,k are all distinct,

$$\mathbb{P}\big(E^*_{ij}(d) \cap E^*_{ik}(d)\big) \leq \frac{1}{\binom{L_N}{3d^o_f}} \binom{L_N}{d^o_f} \binom{\lfloor d/d^i_2 \rfloor}{d^o_f}^2$$

  and the same bound holds for $\mathbb{P}\big(E^*_{ij}(d) \cap E^*_{kj}(d)\big)$

so that we can split the summation in Eq. (4.16) in the following terms:

- $\dfrac{\sum_{\substack{i,j,k,l \in I \\ i,j,k,l \text{ distinct}}} \mathbb{P}\big(E^*_{ij}(d) \cap E^*_{kl}(d)\big)}{\left[\sum_{\substack{i,j \in I \\ i \neq j}} \mathbb{P}\big(E^*_{ij}(d)\big)\right]^2} \overset{N \to \infty}{\longrightarrow} 1$
- $\dfrac{\sum_{\substack{i,j,k \in I \\ i,j,k \text{ distinct}}} \left[\mathbb{P}\big(E^*_{ij}(d) \cap E^*_{ik}(d)\big) + \mathbb{P}\big(E^*_{ij}(d) \cap E^*_{kj}(d)\big)\right]}{\left[\sum_{\substack{i,j \in I \\ i \neq j}} \mathbb{P}\big(E^*_{ij}(d)\big)\right]^2} \leq c_1 \dfrac{1}{N}$

  for some constant $c_1 > 0$ and so it tends to zero when $N \to \infty$;
- $\dfrac{\sum_{\substack{i,j \in I \\ i \neq j}} \mathbb{P}\big(E^*_{ij}(d)\big)}{\left[\sum_{\substack{i,j \in I \\ i \neq j}} \mathbb{P}\big(E^*_{ij}(d)\big)\right]^2} \leq c_2 \dfrac{1}{N^{2-d^o_f} d^{d^o_f}} = c_2 \left(\dfrac{N^\beta}{d}\right)^{d^o_f}$

  for some constant $c_2 > 0$ and so, if $d = o(N^\beta)$, it tends to zero when $N \to \infty$.

As a conclusion, for $N \to \infty$:

- if $d/N^\beta \to \infty$, then $\mathbb{P}(\mathrm{d}^{\min}_N \leq d) \to 1$;
- if $d/N^\beta \to 0$, then $\mathbb{P}(\mathrm{d}^{\min}_N \leq d) \to 0$.

This proves that Theorems 4.5 and 4.7 are true also for odd d^o_f. Unfortunately, the second moment technique used here does not tell anything about the speed of convergence: even for $d/N^\beta \to \infty$, when the we know that $\mathrm{d}^{\min} \leq d$ deterministically, the upper bound obtained with the second moment technique goes to zero as slow as $1/N$. Thus we cannot obtain a full strong result analogous to Theorems 4.4 and 4.6. However, it is clear that we have at least the following, by the same techniques used to prove Theorems 4.4 and 4.6: with probability one,

(a) $(X_N)_{N\in\mathbb{N}}$ and $(Y_N)_{N\in\mathbb{N}}$ densely cover $[\alpha,\beta]$;

(b) $\liminf_N X_N = \liminf_N Y_N = \alpha$;

(c) $\beta \le \limsup_N X_N = \limsup_N Y_N \le \tilde{\beta}$.

4.5.5 Other generalizations and open questions

One straightforward generalization is a relaxation of the assumption that both encoders are non-catastrophic, using instead the assumption that the family of serial encoding schemes is concatenatedly non-catastrophic (Definition 3.2). With this relaxed assumption, all results of this paper still hold true, without any modification in the proofs.

An interesting generalization is the study of other ensembles. From the same fixed component encoders ϕ^o and ϕ_i, it is possible to construct different ensembles, introducing other probabilistic structures for the interleaver sequence. For instance, instead of a sequence of independent interleavers $(\Pi_N)_{N\in\mathbb{N}}$ with Π_N uniformly distributed over S_{L_N} as in our serial turbo ensemble, we can consider a sequence of interleavers $(\Pi'_N)_{N\in\mathbb{N}}$ such that each Π'_N is still uniformly distributed over S_{L_N}, but possibly dependent on $\{\Pi'_i, i = 1,\dots,N-1\}$.

A close look at our proofs shows that independence among the Π_N's is required only when using point (ii) of the Borel-Cantelli lemma. Hence, for the new ensemble based on $(\Pi'_N)_{N\in\mathbb{N}}$ we can state that (for even d_f^o), with probability one:

- $\liminf_N X'_N \ge \alpha$; $\limsup_N X'_N = \beta$,
- $\liminf_N Y'_N \ge \alpha$; $\limsup_N Y'_N = \beta$,

while $X'_N \xrightarrow{\mathbb{P}} \beta$ and $Y'_N \xrightarrow{\mathbb{P}} \beta$.

This means that introducing some dependence among the uniform interleavers cannot make performance worse while it could possibly improve it. It would be interesting to develop an analysis for these hierarchical structures.

Finally, it would be very interesting to obtain results also for the non-binary case, for example in the same setting considered in Chapter 3. However, the proof techniques used to obtain the tight bounds on enumerating coefficients (Sections 4.2 and 4.2.4) are specific for the binary case: they don't generalize easily, not even for the case of non-binary codes still maintaining the rich algebraic structure of vector spaces over some finite field. We leave as a completely open problem the search for new proof techniques to address this issue.

Chapter 5

A family of structured linear-time encodable LDPC codes

In this chapter, we develop the study of a family of codes which are a generalization of Repeat-Accumulate codes, and which can be seen both as serial turbo schemes and as structured LDPC codes. We already introduced these schemes in Section 3.1.3, as an example of our very general serial turbo scheme, so that we could compute the average error probability (Section 3.4.3).

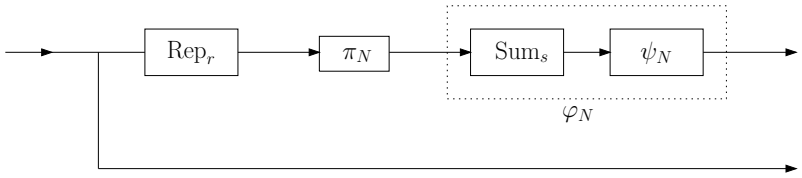
Now we want to focus on the binary case, for which it is easier to get a deeper understanding. First, we will propose the study of the average error probability of a smaller ensemble, which is quite natural in analogy with results from LDPC literature, and which allows to find a design parameter for the inner encoder which was not visible in the bigger ensemble. In this smaller ensemble, the interleaver is uniformly distributed on a subset of permutations which is not a subgroup and thus the study requires some other proof techniques in addition to what is done in Chapter 3.

Simulations using usual LDPC decoding show some dependence on the proposed design parameter, but not as much as it could be expected. An explanation to this is that in many cases the presence of a large amount (linear in the blocklength) of small cycles in the structured part of the Tanner graph deteriorates the performance of the iterative decoder, thus giving poor performance even for some good codes. For an important class of inner encoders, we introduce a modified decoding, which corresponds to suitably grouping together the nodes in the Tanner graph, obtaining a modified graph without cycles in the structured part. Under this modified algorithm, simulations show a good matching with the proposed design parameter: in the medium-high SNR, the hierarchy given by the inner encoder's effective free distance is respected. Moreover, it is possible to analyze the behaviour of codes under this decoding at small SNR, by density evolution. This was not possible for most codes of our structured family with usual LDPC decoding, because the tree-like assumption

was false even after very few iterations due to small structured cycles.

5.1 Encoder description and parity check matrix

We consider the following encoding scheme:



where:

- $\mathrm{Rep}_r : \mathbb{Z}_2^N \to \mathbb{Z}_2^{rN}$ we denote the repetition code with rate $1/r$;
- $\mathrm{Sum}_s : \mathbb{Z}_2^{rN} \to \mathbb{Z}_2^{rN/s}$ is defined by

$$\mathrm{Sum}_s(\boldsymbol{x}) = (x_1 + \ldots + x_s, x_{s+1} + \ldots + x_{2s}, \ldots)$$

  i.e. it gives the modulo-2 sum of every block of s bits (s is called the grouping factor in the Irregular Repeat Accumulate codes literature).

- $\psi(D) : \mathbb{Z}_2^k((D)) \to \mathbb{Z}_2^k((D))$ is a rate-1 non-catastrophic and recursive convolutional encoder, and $\psi_N : \mathbb{Z}_2^{rN/s} \to \mathbb{Z}_2^{rN/s}$ is the truncated encoder obtained by using the trellis of $\psi(D)$ for $rN/(sk)$ time steps (here we consider truncation instead of termination of the convolutional encoder, for simplicity, but the terminated encoder could be considered as well)

- $\pi_N \in S_{rN}$.

We will always assume that rN is a multiple of sk, so that the above construction can be properly made (this will be implicitly assumed also when taking limits for $N \to \infty$).

This scheme generalizes Repeat-Accumulate and Repeat-Convolute codes, which are the particular case when $s = 1$ and $k = 1$ (and $\psi(D) = 1/(1 + D)$ for Repeat-Accumulate codes). On the contrary, Irregular Repeat-Accumulate (see [38]) introduced for the first time the grouping factor s, and were more general than our scheme in the outer repetition, which was irregular, i.e. time-variant, while they were less general in the inner encoder, which was fixed to be the accumulator.

To make this scheme fit in the general setting presented in Chapter 3, you need to include the systematic branch in both the outer and the inner encoder and to consider interleavers (π_N,$\tilde{\pi}_N$) where $\tilde{\pi}_N$ permutes only the systematic bits, and so

does not change the performance of the code. This was the description of these schemes given in Section 3.1.3. Other two small differences from the scheme in Section 3.1.3 are that we are allowing ψ to be non-scalar, even if with rate 1, and that we are taking a truncated instead of a terminated version, i.e. we are not enforcing the return to all-zero state. However, this does not affect the results on the average-based analysis.

The decoding can be performed exploiting the fact that these same codes can also be seen as LDPC codes: a parity check matrix can be constructed in the following way. Notice that a pair $(\boldsymbol{u},\boldsymbol{c}) \in \mathbb{Z}_2^N \times \mathbb{Z}_2^{rN/s}$ belongs to our code if and only if $\boldsymbol{c} = \psi_N \circ \mathrm{Sum}_s \circ \pi_N \circ \mathrm{Rep}_r(\boldsymbol{u})$, which is equivalent to $\mathrm{Sum}_s \circ \pi_N \circ \mathrm{Rep}_r(\boldsymbol{u}) + \psi_N^{-1}(\boldsymbol{c}) = \boldsymbol{0}$ and can be represented with matrices as $[H_N\, K_N]\left[\begin{smallmatrix}\boldsymbol{u}\\ \boldsymbol{c}\end{smallmatrix}\right] = \boldsymbol{0}$.

Notice that H_N is a low-density matrix depending only on r, s and on the permutation π_N, and has at most s ones per row and r ones per column, while K_N is a matrix depending on the choice of ψ, and is also low density, having a number of ones per row bounded by the degree of the polynomial $\psi^{-1}(D)$ and a number of ones per column bounded by k times the degree of $\psi^{-1}(D)$.

For example, if $k = 1$ and $\psi(D)$ is the accumulator $\psi(D) = 1/(1+D)$, we have the so-called 'staircase' LDPC codes: K_N has ones on the diagonal and on the lower diagonal, and zeros everywhere else.

As another example, the scalar encoder $\psi(D) = \frac{1}{1+D+D^3}$ is associated to a matrix

$$K_N = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & \dots \\ 1 & 1 & 0 & 0 & 0 & 0 & \dots \\ 0 & 1 & 1 & 0 & 0 & 0 & \dots \\ 1 & 0 & 1 & 1 & 0 & 0 & \dots \\ 0 & 1 & 0 & 1 & 1 & 0 & \dots \\ 0 & 0 & 1 & 0 & 1 & 1 & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \end{bmatrix}$$

5.2 Error floor region analysis

In this section, we look for design parameters for the medium-high SNR region.

5.2.1 Uniform interleaver

It is clear from Section 3.4.3, that Theorem 3.1 applies to the family of codes we are analyzing. Theorem 3.1, together with the computations in Section 3.4, gives the following result.

If $r \geq 2$, there exist positive constants γ_0, C_1 and C_2 (depending only on the ensemble, i.e. on r,s,$\psi(D)$) such that, for any BIOS channel with Bhattacharyya noise parameter $\gamma \leq \gamma_0$:

- $C_1 p^{d^*} N^{-\mu} \leq \overline{P_b(e)} \leq C_2 \gamma^{d^*} N^{-\mu} + O(N^{-\mu-1})$ for $N \to \infty$
- $C_1 p^{d^*} N^{-\mu+1} \leq \overline{P_w(e)} \leq C_2 \gamma^{d^*} N^{-\mu+1} + O(N^{-\mu})$ for $N \to \infty$

where p is the equivocation probability of the channel and $\mu = \lfloor (r+1)/2 \rfloor$. For $s = 1$, i.e. for traditional Repeat-Convolute codes,

$$d^* = \begin{cases} 1 + d_2^\psi & \text{if } r \text{ is even} \\ 1 + \frac{r-3}{2} d_2^\psi + \min\left\{ d_2^\psi + d_{1,\mathrm{tr}}^\psi, d_{f,3}^\psi \right\} & \text{if } r \text{ is odd} \end{cases}$$

and for all $s \geq 2$,

$$d^* = \begin{cases} 1 & \text{if } r \text{ is even} \\ 2 & \text{if } r = 3 \\ 1 + \min\left\{ d_{1,\mathrm{tr}}^\psi, d_{f,3}^\psi \right\} & \text{if } r \text{ is odd, } r \geq 5 \end{cases}$$

where d_2^ψ and d_3^ψ are the smallest output weight of error events of $\psi(D)$ with input weight 2 and 3 respectively, while $d_{1,\mathrm{tr}}^\psi$ is the smallest output weight of a truncated error event of ψ_N. If $k = 1$, clearly $d_{1,\mathrm{tr}}^\psi = 1$ and so $d^* = 2$ for all odd r and all $s \geq 2$.

5.2.2 A better smaller ensemble and a design parameter

In the result given in previous section, notice that when $s \geq 2$ there is essentially no dependence of the exponents μ and d^* on the choice of the encoder ψ. Looking at traditional serial turbo codes [3], we see that it is natural that μ depends only on the free distance of the outer encoder, but we expect a dependence of the effective free distance d^* on the inner encoder too. What happens with our schemes is that pairs of ones which are repetition of a same information bit can be permuted by some interleaver in such a way that they are summed up by Sum_s, producing a zero output. The value of d^* is given by this 'worse case' scenario.

This remark suggests to consider a smaller family of interleavers, enforcing that ones coming from the same error event of Rep_r cannot end up in positions where they would be summed up by Sum_s. More precisely, we define the set

$$R_{r,s}^N := \left\{ \pi \in S_{rN} : \lfloor i/r \rfloor = \lfloor j/r \rfloor \Rightarrow \lfloor \pi(i)/s \rfloor \neq \lfloor \pi(j)/s \rfloor \right\}$$

What we want to consider is an ensemble of encoders constructed as in Section 5.1, except that now the permutation is uniformly distributed on $R_{r,s}^N$ instead of all S_{rN}. Additionally to the motivation of finding a more interesting effective free distance, this ensemble turns out to be a natural choice in analogy with classical results for regular LDPC codes: restricting the permutation to $R_{r,s}^N$ is the same as enforcing that the Tanner graph corresponding to the regular part of the matrix, H_N, does not have cycles of length two. This new ensemble is also equivalent to

pick H_N uniformly at random in the set of $N \times N$ binary matrices with exactly s ones per row and r ones per column.

As $R_{r,s}^N$ is not a group, we cannot directly apply results from Chapter 3. However, we can slightly modify our techniques for estimating $\mathbb{E}(P_b(e)|R_{r,s}^N)$, where $\mathbb{E}$ is taken in the ensemble with Π_N uniformly distributed in S_{rN}; notice that $\mathbb{E}(P_b(e)|R_{r,s}^N)$ is equal to the average $P_b(e)$ when Π_N is uniformly distributed in $R_{r,s}^N$ which is what we would like to estimate; we will also denote it $\overline{P_b(e)}_{\text{exp}}$.

The key remark is that the probability that a permutation uniformly extracted from S_{rN} belongs to $R_{r,s}^N$ is non-vanishing: $\mathbb{P}(R_{r,s}^N) \to e^{-(r-1)(s-1)/2}$ when $N \to \infty$ (see e.g. [11] Exercise 2.12 p. 59).

Notice that $\mathbb{P}(R_{r,s}^N)$ tends to a constant which is strictly smaller than one, so even though the techniques we use are the same usually known as expurgation, the result we will get is not the typical behavior of the ensemble introduced in Sect. 5.2.1: we will find the average behavior of a subensemble which is neither vanishing nor typical, but is well characterized.

Define $\mu = \lfloor (r+1)/2 \rfloor$ and

$$d_{\text{exp}}^* = \begin{cases} 2 & \text{if } r = 2 \text{ and } s \geq 2 \\ 1 + d_2^\psi & \text{if } r = 2 \text{ and } s = 1 \\ 2 & \text{if } r = 3 \text{ and } s \geq 2 \\ 1 + \min\left\{ d_2^\psi + d_{1,\text{tr}}^\psi, d_3^\psi \right\} & \text{if } r = 3 \text{ and } s = 1 \\ 1 + \frac{r}{2} d_2^\psi & \text{if } r \text{ is even, } r \geq 4 \\ 1 + \frac{r-3}{2} d_2^\psi + \min\left\{ d_2^\psi + d_{1,\text{tr}}^\psi, d_3^\psi \right\} & \text{if } r \text{ is odd, } r \geq 5 \end{cases}$$

Our main result is the following.

Theorem 5.1 If $s \geq 2$. there exist positive constants γ_0, c_1 and c_2 (depending only on the ensemble, i.e. on r,s,$\psi(D)$) such that, for any BIOS channel with Bhatthacharyya noise parameter $\gamma \leq \gamma_0$, and with equivocation probability p,

- $c_1 p^{d_{\text{exp}}^*} N^{-\mu} \leq \overline{P_b(e)}_{\text{exp}} \leq c_2 \gamma^{d_{\text{exp}}^*} N^{-\mu} + O(N^{-\mu-1})$
- $c_1 p^{d_{\text{exp}}^*} N^{-\mu+1} \leq \overline{P_w(e)}_{\text{exp}} \leq c_2 \gamma^{d_{\text{exp}}^*} N^{-\mu+1} + O(N^{-\mu})$ □

In the remainder of this section, we will prove this theorem, by following similar steps to the proofs in Sections 3.3.1 and 3.3.2 together with expurgation.

We start with the upper bound. By the union-Bhattacharyya bound:

$$\overline{P_b(e)}_{\text{exp}} \leq \sum_{w=1}^{N} \sum_{d=w}^{(r+s)N/s} \frac{w}{N} \mathbb{E}(A_{w,d}^N(\Pi_N)|R_{r,s}^N) \gamma^d \tag{5.1}$$

where $A^N_{w,d}(\pi)$ is the number of codewords of the concatenated scheme with input Hamming weight w and output Hamming weight d for a given permutation $\pi \in S_{rN}$.

For most of the terms, we will use the estimation

$$\mathbb{E}(A^N_{w,d}(\Pi_N)|R^N_{r,s}) \leq \frac{\mathbb{E}\big(A^N_{w,d}(\Pi_N)\big)}{\mathbb{P}(R^N_{r,s})} = \frac{\overline{A_{w,d}}^N}{\mathbb{P}(R^N_{r,s})} \tag{5.2}$$

and the fact that $\mathbb{P}(R^N_{r,s})$ is bounded away from zero, so that we can exploit all what we know about $\sum_w \sum_d \overline{A_{w,d}}^N \gamma^d$ from Section 3.3.1.

We consider separately the terms with $w \in \mathcal{W}$ where $\mathcal{W} := \{1\}$ for $r \geq 4$, $\mathcal{W} = \{1,2\}$ if $r = 3$ and $\mathcal{W} = \mathbb{N}^*$ if $r = 2$. The reason is that, with the notation from Section 3.3.1, because of Prop. 3.11, $\mathcal{H} = \{(w,rw) : w \in \mathcal{W}\}$, so these are the main terms in the estimation. Also note that $\mathrm{d}^*_{\mathrm{exp}}$ is the minimum output weight of φ_N if the input weight is restricted to $\{rw, w \in \mathcal{W}\}$ and the permutation is enforced to belong to $R^N_{r,s}$.

Now define $V^{\varphi_N}_{h,k}$ as the set of codewords of $\varphi_N(\mathbb{Z}_2^{rN})$ with input weight h and output weight k and note that

$$\mathbb{E}\big(A^N_{w,d}(\Pi_N)|R^N_{r,s}\big) = \sum_{\boldsymbol{u}\in\mathbb{Z}_2^N : \mathrm{w_H}(u)=w} \sum_{\boldsymbol{v}\in V^{\varphi_N}_{r,d-w}} \mathbb{P}\big(\Pi_N(\mathrm{Rep}_r(\boldsymbol{u})) = \boldsymbol{v}|R^N_{r,s}\big)$$

For the term with $w = 1$, define also $S^N_s = \big\{\boldsymbol{v} \text{ s.t. } \lfloor i/s \rfloor \neq \lfloor j/s \rfloor \, \forall i \neq j : v_i = v_j = 1\big\}$ and notice that $\mathrm{w_H}(\boldsymbol{u}) = 1$ and $\boldsymbol{v} \notin S^N_s$ give $\mathbb{P}\big(\{\Pi_N(\mathrm{Rep}_r(\boldsymbol{u})) = \boldsymbol{v}\} \cap R^N_{r,s}\big) = 0$, so that

$$\begin{aligned}\mathbb{E}\big(A^N_{w,d}(\Pi_N)|R^N_{r,s}\big) &= \sum_{\substack{\boldsymbol{u}\in\mathbb{Z}_2^N \\ \mathrm{w_H}(u)=w}} \sum_{\boldsymbol{v}\in V^{\varphi_N}_{r,d-w}\cap S^N_s} \mathbb{P}\big(\Pi_N(\mathrm{Rep}_r(\boldsymbol{u})) = \boldsymbol{v}|R^N_{r,s}\big) \\ &\leq \binom{wN}{w} |V^{\varphi_N}_{r,d-w} \cap S^N_s| \, \frac{1}{\binom{rwN}{rw}\mathbb{P}(R^N_{r,s})}\end{aligned}$$

Then, $|V^{\varphi_N}_{r,d-w} \cap S^N_s| = \sum_{n=0}^{n_{\max}} |V^{\varphi_N}_{r,d-w,n} \cap S^N_s|$, where $V^{\varphi_N}_{h,k,n}$ denotes the set of codewords of $\varphi_N(\mathbb{Z}_2^{rN})$ with input weight h and output weight k made by exactly n error events, plus possibly a final truncated error event not counted by n.

The recursiveness of φ_N ensures $n_{\max} \leq \lfloor r/2 \rfloor$, but also notice that if $\mathrm{w_H}(\boldsymbol{v}) \in \mathcal{W}$ and $\boldsymbol{v} \in V^{\varphi_N}_{r,d-w,n} \cap S^N_s$, then $\mathrm{w_H}(\varphi_N(\boldsymbol{v})) \geq d^*_{\mathrm{exp}}$, so that for $d < d^*_{\mathrm{exp}}$ we have the tighter bound $n_{\max} \leq \lfloor r/2 \rfloor - 1$. Finally, we estimate $|V^{\varphi_N}_{r,d-w,n} \cap S^N_s| \leq |V^{\varphi_N}_{r,d-w,n}|$.

For the terms $w \in \mathcal{W}$ such that $w \geq 2$, which exist only for $r = 2$ and $r = 3$, note that

$$\mathbb{E}\big(A^N_{w,d}(\Pi_N)|R^N_{r,s}\big) = \sum_{\substack{\boldsymbol{u}\in\mathbb{Z}_2^N \\ \mathrm{w_H}(u)=w}} \sum_{\boldsymbol{v}\in V^{\varphi_N}_{r,d-w}} \mathbb{P}\big(\Pi_N(\mathrm{Rep}_r(\boldsymbol{u})) = \boldsymbol{v}|R^N_{r,s}\big) \leq \binom{wN}{w} \frac{|V^{\varphi_N}_{r,d-w}|}{\binom{rwN}{rw}\mathbb{P}(R^N_{r,s})}$$

and $|V^{\varphi_N}_{r,d-w} = \sum_{n=0}^{n_{\max}} |V^{\varphi_N}_{r,d-w,n}$, with $n_{\max} = \lfloor r/2 \rfloor$ for $d \geq 2 = d^*_{\exp}$ while $n_{\max} \leq \lfloor r/2 \rfloor - 1$ for $d < 2 = d^*_{\exp}$.

We can now put the above estimations in the union-Bhatthacharyya bound (5.1):

$$\begin{aligned}\overline{P_b(e)}_{\exp} \leq & \frac{1}{\mathbb{P}(R^N_{r,s})} \sum_{w\in\mathcal{W}, w\leq d^*} \frac{w}{N}\binom{wN}{w}\frac{1}{\binom{rwN}{rw}} \sum_{n=0}^{\lfloor rw/2\rfloor} |V^{\varphi_N}_{r,d^*-w,n}|\gamma^{d^*} \\ & + \frac{1}{\mathbb{P}(R^N_{r,s})} \sum_{d=d^*+1}^{(r+s)N/s} \sum_{w\in\mathcal{W}, w\leq d} \frac{w}{N}\binom{wN}{w}\frac{1}{\binom{rwN}{rw}} \sum_{n=0}^{\lfloor rw/2\rfloor-1} |V^{\varphi_N}_{r,d-w,n}|\gamma^{d} \\ & + \frac{1}{\mathbb{P}(R^N_{r,s})} \sum_{d=d^*+1}^{(r+s)N/s} \sum_{w\notin\mathcal{W}, w\leq d} \frac{w}{N}\binom{wN}{w}\frac{1}{\binom{rwN}{rw}} \sum_{n=0}^{\lfloor rw/2\rfloor} |V^{\varphi_N}_{r,d-w,n}|\gamma^{d}\end{aligned}$$

Then the proof is ended by the same techniques used in Section 3.3.1, in particular Prop. 2.5, showing an analogous of Prop. 3.8 and Prop. 3.9, i.e. that the first summation is bounded by $c\gamma^{d^*}\frac{1}{N^\mu}$ while the second and third are bounded by $c(\gamma)\frac{1}{N^{\mu+1}}$. Here the remark that $\mathbb{P}(R^N_{r,s})$ remains bounded away from zero when $N \to \infty$ is essential.

Now we adapt the proof of the lower bound in Section 3.3.2 to our setting. We use the same technique:

$$\overline{P_w(e)}_{\exp} \geq p^{d^*_{\exp}} \mathbb{P}(\mathrm{d}^{\min}_N \leq d^*_{\exp} | R^N_{r,s})$$

and then

$$\mathbb{P}(\mathrm{d}^{\min}_N \leq d^*_{\exp} | R^N_{r,s}) \geq \mathbb{P}\Big(\bigcup_{\boldsymbol{a},\boldsymbol{b}} E_{\boldsymbol{a},\boldsymbol{b}} \big| R^N_{r,s}\Big)$$

for some suitably defined events $E_{\boldsymbol{a},\boldsymbol{b}}$, here slightly different from those in Sect. 3.3.2.

First of all, note that if $s = 1$ we don't have anything to prove, as $R^N_{r,s} = S_{rN}$ and the ensemble is the same as in previous section. When $s \geq 2$, the output weight $d^*_{\exp}$ is obtained with input weight $w = 1$, which simplifies the description of words $\boldsymbol{c}^*_{\boldsymbol{a}}$: now a is simply scalar, $a \in \mathcal{A} := \{0, \ldots, N-1\}$ and $\boldsymbol{c}^*_a := \mathrm{Rep}_r(D^a)$. Only for the case $r = 2$ we need the same trick used in Section 3.3.2 and we restrict $\mathcal{A}$ to be $\mathcal{A} = \{0, T, 2T, \ldots, T\lfloor N/T \rfloor - 1\}$ for a suitable fixed $T > 0$. Then fix $\boldsymbol{v}^*$ an input word for ψ^i_N, made of:

- if r is even: $r/2$ regular error events $\boldsymbol{v}^*_1, \ldots, \boldsymbol{v}^*_{r/2}$ of input weight 2 and output weight d^ψ_2, say λ is the length of such events;

- if r is odd and $d_3^\psi < d_2^\psi + d_{1,\mathrm{tr}}^\psi$: $(r-3)/2$ regular error events $\boldsymbol{v}_1^*, \ldots, \boldsymbol{v}_{(r-3)/2}^*$ of input weight 2 and output weight d_2^ψ, with length λ_2, and one regular event $\boldsymbol{v}_{(r-1)/2}^*$, of input weight 3, output weight d_3^ψ and length λ_3; let $\lambda = \max\{\lambda_1, \lambda_2\}$;
- if r is odd and $d_2^\psi + d_{1,\mathrm{tr}}^\psi \leq d_3^\psi$: $(r-1)/2$ regular error events $\boldsymbol{v}_1^*, \ldots, \boldsymbol{v}_{(r-1)/2}^*$ of input weight 2 and output weight d_2^ψ, with length λ, and a truncated event $\boldsymbol{v}_{\mathrm{tr}}^*$, of input weight 1, output weight $d_{1,\mathrm{tr}}^\psi$ and length 1.

Define $\mathcal{B} = \{0,1,\ldots,\lfloor \frac{2N}{sk} \rfloor - 2\}$. For all $\boldsymbol{b} \in \mathcal{B}^{\lfloor r/2 \rfloor}$ define the word $\boldsymbol{v}_{\boldsymbol{b}}^* \in (\mathbb{Z}_2^k)^{rN/(sk)}$ made of the same error events of $\boldsymbol{v}^*$, with its j-th error event starting at time $(j-1)\big(\lfloor \frac{2N}{sk} \rfloor - 1\big) + b_j$, plus possibly the truncated event starting at time $\lfloor \frac{2N}{sk} \rfloor - 1$. Finally define the word $\boldsymbol{u}_{\boldsymbol{b}}^* \in \mathbb{Z}_2^{rN}$ obtained from $\boldsymbol{v}^*$ by first identifying $(\mathbb{Z}_2^k)^{rN/(sk)}$ with $\mathbb{Z}_2^{rN/s}$ and then spreading the ones s apart, i.e. transforming $\sum_{i=1}^r D^{t_i}$ in $\sum_{i=1}^r D^{st_i}$. It is clear that, if $\pi(\boldsymbol{c}_a^*) = \boldsymbol{u}_{\boldsymbol{b}}^*$ for some $a \in \mathcal{A}$ and $\boldsymbol{b} \in \mathcal{B}^{\lfloor r/2 \rfloor}$, then $\psi_N \circ \mathrm{Sum}_s(\boldsymbol{u}_{\boldsymbol{b}}^*) = \psi_N(\boldsymbol{v}_{\boldsymbol{b}}^*)$ so that the final output weight corresponding to input D^a, including one systematic bit, is exactly d_{exp}^*. For $a \in \mathcal{A}$ and $\boldsymbol{b} \in \mathcal{B}^{\lfloor r/2 \rfloor}$, define $E_{a,\boldsymbol{b}} := \{\Pi_N(\boldsymbol{c}_a^*) = \boldsymbol{u}_{\boldsymbol{b}}^*\}$ so that

$$\mathbb{P}(\mathrm{d}_N^{\min} \leq d_{\mathrm{exp}}^* | R_{r,s}^N) \geq \mathbb{P}\Big(\bigcup_{\boldsymbol{a} \in \mathcal{A}, \boldsymbol{b} \in \mathcal{B}^{\lfloor r/2 \rfloor}} E_{\boldsymbol{a},\boldsymbol{b}} \big| R_{r,s}^N \Big)$$

Now we proceed to estimate this union of events with inclusion-exclusion, similarly to what done in Section 3.3.2. Here, it's easier to deal with intersections, as a is simply scalar, while we need to deal carefully with the conditioning on $R_{r,s}^N$. First of all, notice that $\mathbb{P}(E_{a,\boldsymbol{b}} \cap R_{r,s}^N)$ does not depend on a and $\boldsymbol{b}$ and is non-zero. Then, to find a lower bound for $\sum_{a,\boldsymbol{b}} \mathbb{P}(E_{a,\boldsymbol{b}} | R_{r,s}^N)$, for $\boldsymbol{l} = l_0, \ldots, l_{r-1}$ and $\boldsymbol{i} = i_0, \ldots, i_{r-1}$, define the events

$$F_{a,\boldsymbol{l},\boldsymbol{i}}^N = \Big\{ \Pi_N(\boldsymbol{c}_a^*) = \sum_{j=0}^{r-1} D^{sl_j + i_j} \Big\}$$

and notice that, for any $a \in \mathcal{A}$, $\boldsymbol{b} \in \mathcal{B}^{\lfloor r/2 \rfloor}$, we have

$$\begin{aligned} \mathbb{P}(R_{r,s}^N) &= \sum_{\substack{l_0 < \cdots < l_{r-1} \\ 0 \leq l_j \leq \frac{rN}{s} - 1}} \sum_{\substack{i_0, \ldots, i_{r-1} \\ 0 \leq i_j \leq s-1}} \mathbb{P}\big(R_{r,s}^N \cap F_{a,\boldsymbol{l},\boldsymbol{i}}^N\big) \\ &= \binom{rN/s}{r} s^r \, \mathbb{P}(R_{r,s}^N \cap E_{a,\boldsymbol{b}}) \end{aligned}$$

so that $\mathbb{P}(E_{a,\boldsymbol{b}} | R_{r,s}^N) = \dfrac{\mathbb{P}(E_{a,\boldsymbol{b}} \cap R_{r,s}^N)}{\mathbb{P}(R_{r,s}^N)} = \dfrac{1}{\binom{rN/s}{r} s^r}$ and finally

$$\sum_{a \in \mathcal{A}} \sum_{\boldsymbol{b} \in \mathcal{B}^{\lfloor r/2 \rfloor}} \mathbb{P}(E_{a,\boldsymbol{b}} | R_{r,s}^N) = |\mathcal{A}| \, |\mathcal{B}|^{\lfloor r/2 \rfloor} \frac{1}{\binom{rN}{r} s^r} \geq c N^{-\mu+1} \, .$$

For the term with intersections, we use the simple bound

$$\mathbb{P}(E_{a,\boldsymbol{b}} \cap E_{a',\boldsymbol{b'}} | R^N_{r,s}) = \frac{\mathbb{P}(E_{a,\boldsymbol{b}} \cap E_{a',\boldsymbol{b'}} \cap R^N_{r,s})}{\mathbb{P}(R^N_{r,s})} \leq \frac{\mathbb{P}(E_{a,\boldsymbol{b}} \cap E_{a',\boldsymbol{b'}})}{\mathbb{P}(R^N_{r,s})}$$

Then we exploit the fact that $\mathbb{P}(R^N_{r,s})$ is bounded away from zero and we estimate $\mathbb{P}(E_{a,\boldsymbol{b}} \cap E_{a',\boldsymbol{b'}})$ as follows. For $(a,\boldsymbol{b}) \neq (a',\boldsymbol{b'})$, $\mathbb{P}(E_{a,\boldsymbol{b}} \cap E_{a',\boldsymbol{b'}})$ can be non-zero only if $a \neq a'$ and $b_j \neq b'_j$ for all j and there is no final truncated error event in $\boldsymbol{u}^*_{\boldsymbol{b}}$. Under these assumptions, simply

$$\mathbb{P}(E_{a,\boldsymbol{b}} \cap E_{a',\boldsymbol{b'}}) \leq \mathbb{P}(\mathit{\Pi}_N(\boldsymbol{c}_a + \boldsymbol{c}_{a'} = \boldsymbol{u}_{\boldsymbol{b}} + \boldsymbol{u}_{\boldsymbol{b'}}) = \frac{1}{\binom{rN}{2r}}$$

so finally

$$\sum_{(a,\boldsymbol{b}) \neq (a',\boldsymbol{b'})} \mathbb{P}(E_{a,\boldsymbol{b}} \cap E_{a',\boldsymbol{b'}} | R^N_{r,s}) < \frac{1}{\mathbb{P}(R^N_{r,s})} |\mathcal{A}|^2 |\mathcal{B}|^{2\lfloor r/2 \rfloor} \frac{1}{\binom{rN}{2r}} \leq C N^{-2\mu+2} .$$

For $r \geq 3$, which ensures $\mu \geq 2$, this ends the proof:

$$\overline{P_w(e)}_{\text{exp}} \geq p^{d^*_{\text{exp}}} \mathbb{P}(\mathrm{d}^{\min}_N \leq d^*_{\text{exp}} | R^N_{r,s}) \geq p^{d^*_{\text{exp}}} \mathbb{P}\Big(\bigcup_{a,\boldsymbol{b}} E_{\boldsymbol{a},\boldsymbol{b}} \big| R^N_{r,s} \Big) \geq p^{d^*_{\text{exp}}} \big[c N^{-\mu+1} - C N^{-2\mu+2} \big]$$

For $r = 2$, a suitable choice of the constant T in the definition of $\mathcal{A}$ ensures that the constant c is bigger than C, thus showing that $\overline{P_w(e)}_{\text{exp}}$ is bounded away from zero.

5.2.3 ML predictions vs. standard BP simulations

We simulated the coding schemes with two simple examples of $\psi(D)$: the accumulator and $\psi(D) = 1/(1 + D + D^3)$, using the standard belief propagation algorithm over the Tanner graph associated to the low density matrix $[H_N K_N]$. Although this approach is satisfactory for $\psi(D) = 1/(1 + D)$, this is not the case for the encoder $1/(1 + D + D^3)$.

In fact, Monte-Carlo simulations reported in Fig. 5.4 are in contrast with the results of Theorem 5.1: the coding scheme based on the simple accumulator $\psi(D) = 1/(1 + D)$, having $d^\psi_2 = 1$, performs much better than the one using $\psi(D) = 1/(1 + D + D^3)$ as inner encoder, even if the latter has $d^\psi_2 = 4$.

A close look to the structure of the Tanner graphs suggests a possible explanation for such a disappointing behaviour. Indeed, a large number of 6-cycles appear in the structured part of the graph (see Fig. 5.2). More precisely there are $N - 2$ of such cycles and they are concatenated in a very particular way. The belief-propagation algorithm is known to be exact on cycle-free graphs [78] and has been shown to be highly performing on random graphs which with high probability do not contain

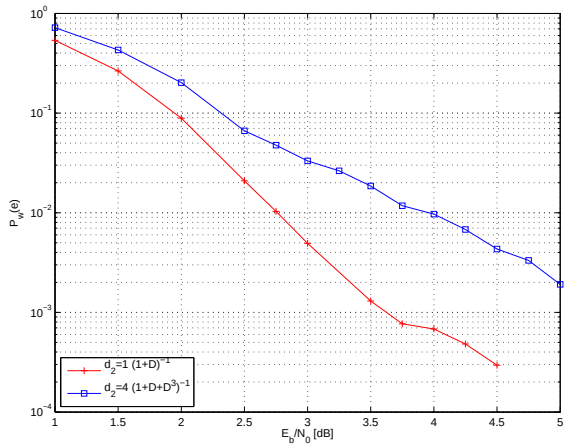
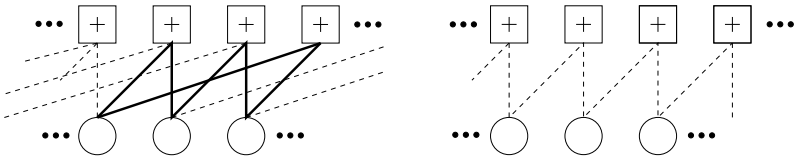


Figure 5.1. Simulation with BP: $1/(1+D)$, $1/(1+D+D^3)$ (blocklength = 300)



(a) $1/(1+D+D^3)$ (b) $1/(1+D)$

Figure 5.2. Structured part of the Tanner graph

small cycles [60]. Thus, the presence of such a big and structured collection of 6-cycles seems to be a possible explanation why the algorithm fails to converge. Notice that the Tanner graph of the Repeat-Accumulate does not contain any cycle in its structured part.

This remark suggests to focus the attention on codes without cycles in the structured part on the Tanner graph, but this is very restrictive. A closer look to the matrix K_N associated with $\psi(D) = 1/(1+D+D^3)$ suggests a different approach. In fact if you look at this matrix gathering together blocks of three bits, a staircase structure emerges:

$$K_N = \begin{bmatrix} \begin{smallmatrix} 1&0&0\\1&1&0\\0&1&1 \end{smallmatrix} & & & \\ & \begin{smallmatrix} 1&0&1\\0&1&0\\0&0&1 \end{smallmatrix} & \begin{smallmatrix} 1&0&0\\1&1&0\\0&1&1 \end{smallmatrix} & \\ & & \begin{smallmatrix} 1&0&1\\0&1&0\\0&0&1 \end{smallmatrix} & \begin{smallmatrix} 1&0&0\\1&1&0\\0&1&1 \end{smallmatrix} \\ & & & \ddots \end{bmatrix}$$

This suggests to focus on matrices with such block-wise staircase structure and to associate to them a modified Tanner graph, where k-tuples of bits are aggregated

together. This will be done in the next section.

5.3 Non-binary decoding of block-wise staircase LDPC codes

5.3.1 Encoder structure

We consider the encoding scheme described in Section 5.1, in the particular case when $\psi(D) = (A + BD)^{-1}$ for some matrices A,$B \in \mathbb{Z}_2^k$, so that the structured part of the matrix will have the form

$$K_N = \begin{bmatrix} A & 0 & 0 & \dots & 0 \\ B & A & 0 & \dots & 0 \\ 0 & B & A & \dots & 0 \\ \vdots & & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & B & A \end{bmatrix} \tag{5.3}$$

Another possibility is to consider an upper staircase structure, with an encoder of the form $\psi = (AD^{-1} + B)^{-1}$, which gives:

$$K_N = \begin{bmatrix} B & A & 0 & \dots & 0 \\ 0 & B & \ddots & & 0 \\ 0 & 0 & \ddots & A & 0 \\ \vdots & & \ddots & B & A \\ 0 & 0 & \dots & 0 & B \end{bmatrix} \tag{5.4}$$

In these schemes, the matrices A and B must be carefully chosen in order to ensure that the inverse matrix we have written in the definition of $\psi(D)$ indeed exists, and that $\psi(D)$ has all the good properties we need: causality, non-catastrophicity, recursiveness (see Section 2.3 for definition of these properties). In this section, we will show that the 'safe' choice, that ensures all these properties, is the lower-staircase structure with A and B both invertible. We will also discuss other possible choices.

We start by deriving an explicit expression for the minimal state-space realization of such convolutional codes: this is helpful for actually implementing the encoder, but it will also enlighten the theoretical properties of $\psi(D)$.

Let's focus at first on $\psi(D) = (A + BD)^{-1}$, and let's assume that A is invertible. This assumption guarantees that the inverse $(A+BD)^{-1}$ exists, as we can write $(A+BD)^{-1} = A^{-1}(I + BA^{-1}D)^{-1} = A^{-1}\sum_{t\geq 0}(BA^{-1})^t D^t$. Under this assumption, it is

also easy to find a realization with 2^m states where $m = \operatorname{rank} B$, and to prove that it is indeed minimal. In fact, you can note that $y(D) = \psi(D)u(D) \Leftrightarrow \psi^{-1}(D)y(D) = u(D) \Leftrightarrow (A + BD)y(D) = u(D) \Leftrightarrow \forall t, Ay_t + By_{t-1} = u_t \Leftrightarrow \forall t, y_t = A^{-1}u_t + A^{-1}By_{t-1}$. Now, if you define the state $x_t = By_{t-1}$, you have the state space $X = B(\mathbb{Z}_2^k)$, which is a subspace of $\mathbb{Z}_2^k$ having dimension $\operatorname{rank} B$, and you have the following realization:

$$y(D) = \psi(D)u(D) \;\Leftrightarrow\; \begin{cases} y_t = A^{-1}u_t + A^{-1}x_t \\ x_{t+1} = By_t = BA^{-1}u_t + BA^{-1}x_t. \end{cases} \tag{5.5}$$

To prove minimality, we need to prove controllability, i.e. that all states can be reached from the zero-state, and observability, i.e. for all sequence $(u_t,y_t,x_t)_{t\in\mathbb{N}}$ satisfying the system (5.5) if $y_t = u_t = 0 \forall t \geq 0$ then also $x_0 = 0$. Controllability is clearly true. Also observability is true: even more, as (5.5) gives $y_0 = A^{-1}u_0 + A^{-1}x_0$, simply $u_0 = y_0 = 0$ is enough to ensure $x_0 = 0$.

Now, what can we say about the properties of $\psi(D)$? Non-catastrophicity immediately follows from the fact that $A + BD$ is polynomial and is the inverse of $\psi(D)$. The state realization we just described clearly shows that $\psi(D)$ is causal. We also need recursiveness. If also B is invertible, it is clear from the state realization that $\psi(D)$ is recursive. For general B, we have the following effective test for recursiveness.

Proposition 5.1 A convolutional encoder $\psi(D) = (A+BD)^{-1}$ with A invertible, is recursive if and only if the matrix $(BA^{-1})^{2^{k-1}}$ has non-zero weight on each column. □

Proof: For simplicity of notation, define $M := (BA^{-1})$.

Recursiveness means that any input $u(D)$ with Hamming weight 1 produces infinite-weight output; on the minimal state realization, this translates as: an input u_0 of weight $\mathrm{w_H}(u_0) = 1$ followed by $u_t = 0$ for all $t > 0$ produces a state sequence never returning to zero. With the realization (5.5), this condition reads: for all $t > 0$, $x_t = M^t u_0 \neq 0$ for all u_0 with $\mathrm{w_H}(u_0) = 1$, i.e. all columns of M^t have non-zero weight.

Then we can note that there is no need to test this condition for infinitely many values of t. In fact, by Cayley-Hamilton theorem, if you denote by $p(z)$ the characteristic polynomial of M, $p(M) = 0$. Now, $p(z)$ has degree k, with leading coefficient 1. If M is not invertible, we also know that $p(z)$ has trailing coefficient equal to zero (in the case M is invertible, we don't have anything to prove). These remarks on $p(z)$ ensure that M^k is a linear combination of $M,M^2,\ldots,M^{k-1}$, and this also implies that all M^t with $t \geq k$ are linear combinations of $M,M^2,\ldots,M^{k-1}$. There are only 2^{k-1} choices for the coefficients of these combinations, including the choice with only one non-zero coefficient which simply gives again one of the matrices $M,M^2,\ldots,M^{k-1}$. So, finally, after having considered $M,M^2,\ldots,M^{2^{k-1}}$ you will surely never encounter a new different matrix.

The final remark is that testing $M, M^2, \ldots, M^{2^{k-1}}$ is equivalent to testing only $M^{2^{k-1}}$: in fact, if some M^t has zero-weight on some column, then also $M^{t'}$ will have a zero column in the same position, for all $t' > t$. ■

Now we switch to the case when we do not enforce invertibility of A, and we assume the invertibility of B. This assumption ensures existence of the inverse $(A + BD)^{-1}$. However, it does not guarantee that the encoder obtained in such a way is causal; we will prove (Prop. 5.2) that it is causal if and only if also A is invertible. This seems to suggest to avoid non-invertible A's; however, we can prove that for some choices of non-invertible A, the shifted encoder $D(A + BD)^{-1}$, corresponding to an upper-staircase parity matrix as in (5.4), is indeed causal.

Proposition 5.2 If $\psi(D) = (A + BD)^{-1}$ with B invertible, then:

- $\psi(D)$ is causal if and only if also A is invertible;
- $D\psi(D)$ is causal if and only if there exists some $m \in \mathbb{N}^*$ such that $N^{m+1} = N$, where $N := AB^{-1}$.

Moreover, if $N^{m+1} = N$, then

$$D\psi(D) = B^{-1} + B^{-1}N^m(I + N^{2m-1}D)^{-1}\,. \tag{5.6}$$

□

Proof: We start by explicitly constructing the Laurent series of $\psi(D)$. First, $(A + BD)^{-1} = D^{-1}B^{-1}(I + AB^{-1}D^{-1})^{-1} = D^{-1}B^{-1}\sum_{t\geq 0}(AB^{-1})^t D^{-t}$. Now, we use the remark that surely the sequence $(N^t)_{t\in\mathbb{N}}$ is periodical, i.e. there exist $a \in \mathbb{N}$ and $b \in \mathbb{N}^*$ such that $N^{a+b} = N^a$; moreover, $a = 0$ if and only if N is invertible. So:

$$\begin{aligned}
BD\psi(D) &= \sum_{t\geq 0} N^t D^{-t} \\
&= \sum_{t=0}^{a-1} N^t D^{-t} + N^a \sum_{h\geq 0} D^{-(a+bh)} + N^{a+1} \sum_{h\geq 0} D^{-(a+1+bh)} + \cdots + N^{a+b-1} \sum_{h\geq 0} D^{-(a+b-1+bh)} \\
&= \sum_{t=0}^{a-1} N^t D^{-t} + N^a \tfrac{D^{-a}}{1+D^{-b}} + N^{a+1} \tfrac{D^{-(a+1)}}{1+D^{-b}} + \cdots + N^{a+b-1} \tfrac{D^{-(a+b-1)}}{1+D^{-b}} \\
&= \sum_{t=0}^{a-1} N^t D^{-t} + N^a \tfrac{D^{-a+b}}{1+D^{b}} + N^{a+1} \tfrac{D^{-(a+1)+b}}{1+D^{b}} + \cdots + N^{a+b-1} \tfrac{D^{-(a+b-1)+b}}{1+D^{b}} \\
&= \sum_{t=0}^{a-1} N^t D^{-t} + \left(N^{b+a-1}D^{-a+1} + \cdots + N^{a+1}D^{-a+b-1} + N^a D^{-a+b}\right)\left(\sum_{t\geq 0} D^{bt}\right).
\end{aligned} \tag{5.7}$$

Looking at the last line, we see that the first part has terms with exponent increasing from $-a+1$ to 0, while the second has terms from $-a+1$ to $+\infty$: $D\psi(D)$ is causal when all pairs of terms with negative exponent cancel each other. In particular, the terms with exponent -1 cancel, i.e. $N = N^{m+1}$ for some $m+1 \in \{a, \ldots, a+b-1\}$. If $\psi(D)$ itself is causal, then also the term with exponent 0 is zero, i.e. $I = N^m$ for the same m as above.

Viceversa, let's assume that there exists $m \in \mathbb{N}^*$ such that $N^{m+1} = N$, say m is the smallest integer satisfying this relation. If moreover $N^m = I$, i.e. N is invertible, Eq. (5.6) is trivially verified. If $N^m \neq I$, with the above notation we have $a = 1$ and $b = m$, so that Eq. (5.7) simplifies to

$$BD\psi(D) = I + \left(N^m D^0 + N^{m-1} D + \cdots + N^2 D^{m-2} + N D^{m-1}\right) \left(\sum_{t \geq 0} D^{mt}\right).$$

which immediately shows that $D\psi(D)$ is causal. From this, simple calculation gives

$$(BD\psi(D) + I)(I + N^{2m-1} D) = N^m$$

which proves Eq. (5.6). ■

Remark 5.1 By the same argument as in the proof of Prop. 5.1 (i.e. using Cayley-Hamilton Theorem), you can prove that $N^{a+b} = N^a$ with $a + b \leq 2^{k-1}$. So, to check wether there exists m such that $N^{m+1} = N$, you need to compute and compare with N only powers up to $N^{2^{k-1}}$. ■

We can also exploit Eq. (5.6) in order to construct a minimal linear realization of $\tilde{\psi}(D) := D\psi(D)$, with state space $X = N^{2m-1}(\mathbb{Z}_2^k)$. Note that, under the assumption $N^{m+1} = N$, $\operatorname{rank} N^t = \operatorname{rank} N$ for all $t \geq 1$, so $\dim X = \operatorname{rank} N = \operatorname{rank} A$. Also note that $N^{2m-1} = N^{m-1}$ if $m \geq 2$ and $N^{2m-1} = N$ if $m = 1$.

We start finding a realization for the encoder $\varphi(D) := N^m (I + N^{2m-1} D)^{-1}$.

$$v(D) = \varphi(D) u(D) \;\Leftrightarrow\; \begin{cases} v_t = N^m u_t + x_t \\ x_{t+1} = N^{2m-1} v_t = N^{2m-1} u_t + N^{2m-1} x_t. \end{cases} \tag{5.8}$$

From Eq. (5.8), we can find a realization for $\tilde{\psi}(D)$ with the same state space:

$$y(D) = \tilde{\psi}(D) u(D) \;\Leftrightarrow\; \begin{cases} y_t = B^{-1} u_t + B^{-1} v_t = B^{-1}(I + N^m) u_t + B^{-1} x_t \\ x_{t+1} = N^{2m-1} u_t + N^{2m-1} x_t. \end{cases} \tag{5.9}$$

It is easy to prove that this realization is minimal, by the same technique we used for the realization (5.5): both controllability and observability are trivially true.

With the same technique we used for encoders with invertible A, we can use the state realization in order to characterize recursiveness.

Proposition 5.3 Given a causal encoder $\psi(D) = D(A+BD)^{-1}$ with B invertible, $\psi(D)$ is recursive if and only if all columns of N are non-zero, where $N = AB^{-1}$. □

Proof: Recursiveness means that if $\mathrm{w_H}(u_0) = 1$ and $u_t = 0\,\forall t \geq 1$, then $x_t \neq 0\,\forall t \geq 1$. The assumption on the inputs, used in the realization (5.9), gives for $m \geq 2$ the state sequence $x_0 = 0$, $x_1 = N^{m-1}u_0$ $x_2 = N^{2m-2}u_0$, …, $x_m = N$, $x_{m+1} = N^m$, $x_{m+2} = N^{m-1}$ and so on, periodically. For $m = 1$, simply $x_0 = 0$, $x_t = Nu_0$ for all $g \geq 1$. So the recursiveness request translates in asking that no column of the following matrices is zero: N,N^2,…,N^m,$N^{m+1} = N$; this is the same as simply asking that all the columns of N are non-zero. ■

We show here an example of a causal encoder with non-invertible A and invertible B, thus clarifying that the class of encoders characterized in Prop. 5.2 is not empty.

Example 5.1 If you choose matrices

$$A = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix}, \qquad B = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}$$

you get the encoder

$$\psi(D) := D(A+BD)^{-1} = \begin{bmatrix} 1+D^{-1} & D^{-1} & 0 \\ 0 & 1 & D^{-1} \\ 1+D^{-1} & D^{-1} & 1+D^{-1} \end{bmatrix}^{-1} = \begin{bmatrix} 1+\frac{D}{1+D^2} & \frac{1}{1+D} & \frac{1}{1+D^2} \\ \frac{1}{1+D} & 1 & \frac{1}{1+D} \\ \frac{D}{1+D} & 0 & \frac{D}{1+D} \end{bmatrix}$$

From the last expression, you can understand that $\psi(D)$ is causal and recursive, but you can see it directly from the matrices A and B, by applying Propositions 5.2 and 5.3: compute $N = AB^{-1} = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}$, $N^2 = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$, $N^3 = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix} = N$. $N^3 = N$ ensures a causal encoder, and N with no zero-column gives recursiveness. In this example, Eq. (5.9) gives a state realization with 4 states:

$$y(D) = \psi(D)u(D) \;\Leftrightarrow\; \begin{cases} y_t = B^{-1}(I+N^2)u_t + B^{-1}x_t \\ x_{t+1} = Nu_t + Nx_t. \end{cases}$$

□

5.3.2 Decoding algorithm

Motivated by the remarks in Section 5.2.3 about cycles in the Tanner graph, we propose the following modified version of the BP algorithm for block-wise staircase encoders ($k > 1$).

Associate to the parity matrix $[H_N K_N]$ a labeled factor graph with vertex set given by $\mathcal{V}_i \cup \mathcal{V}_p \cup \mathcal{V}_c$ (see Fig.5.3), where:

- $\mathcal{V}_i = \{i_1, \ldots, i_N\}$ is a set of N information nodes, each corresponding to an information bit (recall the codes are systematic);

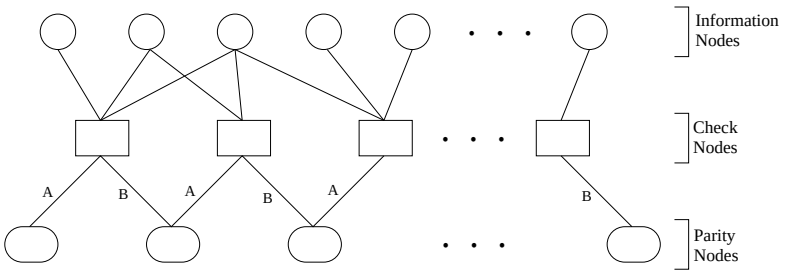


Figure 5.3. Tanner graph of the hybrid non-binary algorithm

- $\mathcal{V}_p = \{p_1, \ldots, p_{\frac{r}{ks}N}\}$ is a set of $\frac{r}{ks}N$ parity nodes, each corresponding to a group of k consecutive parity bits;
- $\mathcal{V}_c = \{c_1, \ldots, c_{\frac{r}{ks}N}\}$ is a set of $\frac{r}{ks}N$ check nodes each corresponding to a group of k consecutive rows of the matrix.

For every $1 \leq j \leq \frac{r}{ks}N$, the parity node p_j is connected only to the check node c_j with an edge labeled by $\lambda_{i_j,c_j} = A$, and to the check node c_{j+1} with an edge labeled by $\lambda_{i_j,c_{j+1}} = B$. There is an edge between a check node c_l in $\mathcal{V}_c$ and an information node i_j in $\mathcal{V}_i$ whenever the $k \times 1$ block $(H_N)_{[k(l-1)+1,kl],j}$ is nonzero; such an edge is labeled by the $k \times 1$ block $\lambda_{c_l,p_j} = (H_N)_{[k(l-1)+1,kl],j}$ itself.

We use a sum-product belief propagation algorithm over this graph. Messages exchanged between information nodes and check nodes consist in probability distributions over $\mathbb{Z}_2$, while messages exchanged between parity nodes and check nodes consist in probability distributions over $\mathbb{Z}_2^k$. For every parity or information node v denote the a posteriori probability distributions given by the channel output by $\boldsymbol{z}_v$. Denote the message sent from node v to node v' at the t-th iteration by $\boldsymbol{m}^t_{v\to v'}$. For every adjacent parity node v and check node c initialize $\boldsymbol{m}^0_{c\to v}$ as the uniform distribution over $\mathbb{Z}_2^k$ and similarly for every adjacent information node v and check node c let $\boldsymbol{m}^0_{c\to v}$ be the uniform distribution over $\mathbb{Z}_2$. Then for every time step $t \geq 1$

- the message sent from a node v in $\mathcal{V}_i \cup \mathcal{V}_p$ to an adjacent check node c, $\boldsymbol{m}^t_{v\to c}$ is the normalized pointwise product of $\boldsymbol{z}_v$ and of messages $\boldsymbol{m}^{t-1}_{c'\to v}$ received by the node v from all its neighbors c' but c;
- the message sent from a check node c to an adjacent information or parity node v is given by

$$\boldsymbol{m}^t_{c\to v}(x) = \mathbb{P}^t_{c\to v}\big(\sum_{\substack{v'\sim c\\ v'\neq v}} \lambda_{c,v'} X_{v'} = \lambda_{cv} x\big)$$

  where the probability $\mathbb{P}^t_{c\to v}$ is evaluated by considering the random variables $X_{v'}$ mutually independent, each distributed accordingly to $\boldsymbol{m}_{v'\to c}$.

The labels A and B on edges clearly affect the messages. If the label is an invertible matrix, it just gives a permutation of the messages. If not, for outgoing messages from the parity node, you need to map a message (which is a probability measure) to the corresponding image measure with respect to the label matrix; in the opposite direction, the probability of some vector is split evenly among all elements of its preimage with respect to the label matrix.

This algorithm falls within the large class of generalized belief propagation algorithms described in [79].

Notice that the complexity of this non-binary BP algorithm (with an efficient implementation of the check-nodes updates), scales with k and N as $2^{2k+1}\frac{r}{ks}N$ operations per iteration, compared to $\frac{4r(k+1)}{s}N$ for the standard BP algorithm.

5.3.3 Simulation results

We have focused our attention on codes with invertible A and B, so that all good properties of $\psi(D)$ were guaranteed.

All the examples we simulated have $r = 4$ and $s = 4$, consequently the overall rate R is $1/2$. A maximum of 50 iterations has been considered.

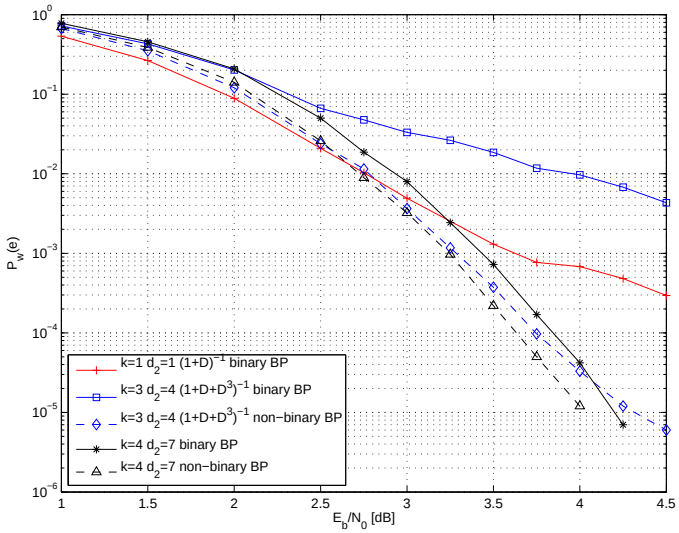


Figure 5.4. Binary vs. non-binary BP decoding (blocklength = 300)

Fig. 5.4 shows how the use of the non-binary algorithm may lead to a dramatic improvement with respect to the standard BP algorithm, in two examples: $1/(1 + D + D^3)$, considered with $k = 3$, and the following encoder $\psi(D) = (A + BD)^{-1}$

with $k = 4$ and $d_2^{\psi} = 7$:

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}, \qquad B = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{bmatrix}$$

In both cases, we conjecture that the improvement is strictly related to the large number of small cycles in the structured part of the Tanner graph. In fact, for other codes such that $1/(1 + D^3 + D^4)$, where the structured cycles have length at least 8, there is almost no difference in between binary and non-binary decoding with $k = 4$. However, we don't have yet a precise theory on how performance is affected by the number of small cycles, by their length and even by the way they are intertwined. For example, we don't have a satisfactory explanation of the much bigger improvement for $1/(1 + D + D^3)$, which has cycles of length 6 with respect to the other $\psi(D)$ reported in Fig. 5.4, which has also cycles of length 4.

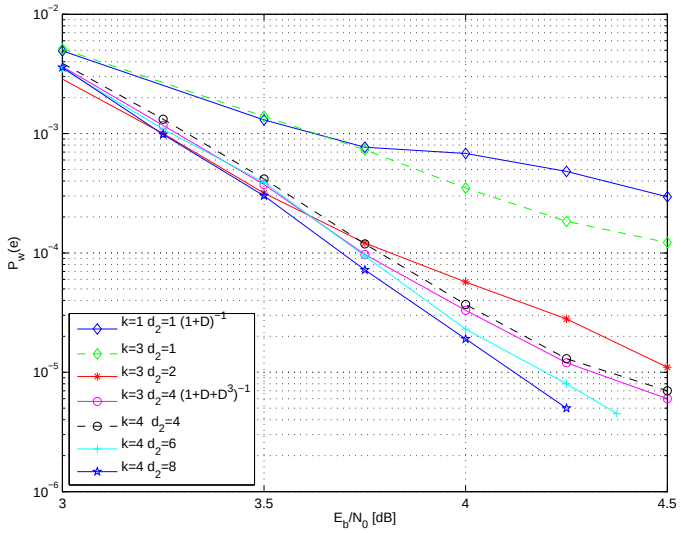


Figure 5.5. Dependence on d_2^{ψ} for different values of k, block length = 300

Figures 5.5 and 5.6 show the role of d_2^{ψ}, comparing different encoders all decoded with the non-binary algorithm. The hierarchy given by this parameter is clearly respected in the error-floor region, as predicted by the theoretical results. At low SNR, we see that the hierarchy is inverted (see Fig. 5.6), so that we have cross points among curves. The codes in Figures 5.5 and 5.6 have blocklength 300 and 600 respectively; at higher lengths it is more difficult to get simulation results in the error floor region, which has very low $P_w(e)$.

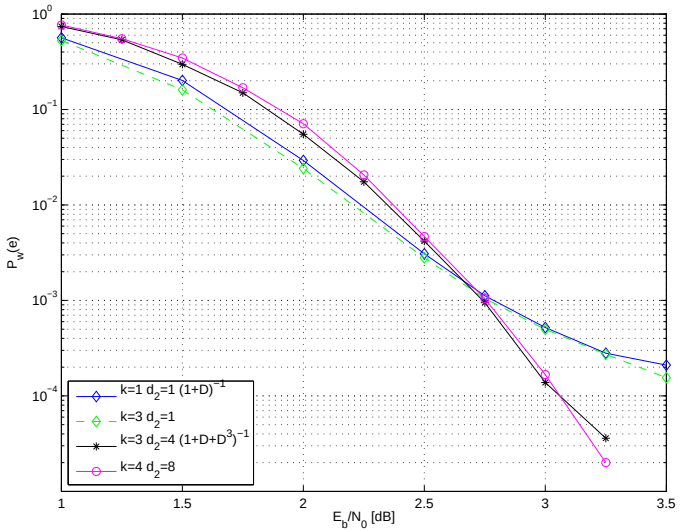


Figure 5.6. Dependence on d_2^{ψ} for different values of k, block length = 600

5.4 Density evolution analysis of the non-binary decoding algorithm

Density evolution is a useful tool introduced in [60] to analyze the convergence of the message-passing decoding of ensembles of LDPC codes.

This tool has been extended to the structured ensemble of IRA in [66], but it could not be extended to our geralized repeat-accumulate codes under usual message-passing, because of the structural presence of cycles in the Tanner graph. In fact, density evolution describes how the distribution probability of the messages exchanged evolve during the decoding algorithm, under the assumption that up to that time no cycle has been encountered. It has been shown [60] that for random regular and irregular LDPC ensembles, if you let the blocklength grow, the probability of not having cycles up to any fixed number of iterations grows to one. On the contrary, most Repeat-Sum-Convolute concatenations give rise to many small cycles, which not only may deteriorate the performance of the usual LDPC decoder, but also prevent the use of density evolution as a tool to analyze its performance.

With our modified algorithm, there are no cycles in the structured part of the graph, while to the remaining random part the same results of [60] apply. The only drawback is that the non-binary algorithm is more difficult to analyze and gives rise to a bigger number of variables, as was already noted in [58] who generalized density evolution to different non-binary decoding.

5.4.1 Density evolution equations

In order to keep the number of variables finite, we will focus on the case of transmission over Binary Erasure Channel (BEC). This choice is quite common in the density evolution literature, because it leads to the study of a finite-dimensional dynamical system.

Thanks to the code linearity and the symmetry of the channel, we can perform the analysis under the assumption that the all-zero codeword has been sent, exactly as in the classical case [60]

On the BEC and supposing transmission of all-zero codewords, the only possible messages sent by the decoding algorithm are:

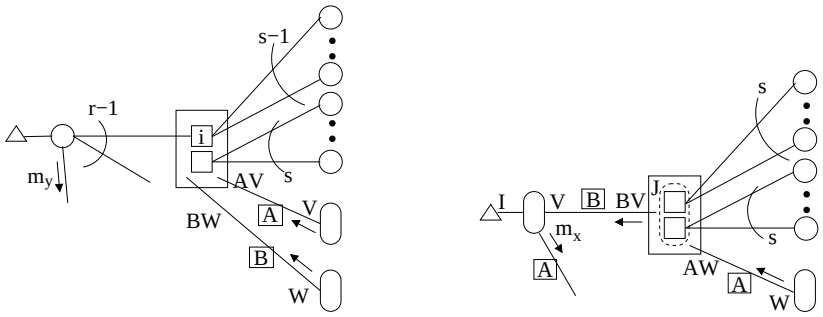
- from and to an information node: either 0 or 'erased'. (i.e. the uniform probability on the set $\{0\}$ and $\{0,1\}$ respectively);
- from and to a parity node: the uniform distribution on some vector space, subset of $\mathbb{Z}_2^k$, possibly $\{0\}$ or $\mathbb{Z}_2^k$ itself. If the message comes from the channel, not all subspaces are possible, only those corresponding exactly to the restriction of $\mathbb{Z}_2^k$ to some of its components, i.e. the spaces having as a basis a subset of the canonical basis of $\mathbb{Z}_2^k$.

Clearly, one can take as message the subspace itself, instead of the uniform distribution on that subspace, so that the set of possible messages from and to the parity nodes becomes $G := \{$ subspaces of $\mathbb{Z}_2^k \}$.

In the density evolution, clearly we want to keep track of the fraction of information bits erased, which we would like to see converging to zero. Then, we need to keep track of the probability that the parity nodes outputs each of its possible messages. Differently from [58], for our parity nodes we need to keep separate the distribution of messages on edges with label A and B, because we have fixed matrices A, B and we cannot use the simplification given by averaging. We can exploit the averaging effect only for information nodes, which have random labels on their output edges. So, the density evolution system will have the following variables:

- $y_t \in [0,1]$ = fraction of information bits erased at time t;
- $\boldsymbol{x}_t^A \in \mathcal{P}(G)$ defined by $x_t^A(V) =$ fraction of output messages from parity symbols (on edge with label A), which at time t that are equal to V;
- analogous definition for $\boldsymbol{x}_t^B$ on edges with label B.

By $\mathcal{P}(G)$ we denote the set of probability vectors of length $|G|$, with the convention that components of these vectors will be labeled by elements $V \in G$ instead of by numbers $1, \ldots, |G|$.



(a) Information Node (b) Parity Node

Figure 5.7. Portions of Tanner graph, with messages exchanged

We will now give the equations describing the density evolution. We will use the short-hand notations $[k] := \{1,2,\ldots,k\}$ and, for $I \subseteq [k]$, $\mathbb{Z}_2^I := \mathrm{span}\{\boldsymbol{e}_i, i \in I\}$. We will denote by $\pi_i(V)$ the restriction of a vector space $V \in G$ to its i-th component, i.e. $\pi_i(V)$ is $\{0\}$ if all vectors in V have their i-th component equal to zero, and is $\{0,1\}$ otherwise.

The update equations are:

$$y_{t+1} = \varepsilon \left[1 - \frac{(1-y_t)^{s-1}}{k} \sum_{i=1}^{k} \sum_{\substack{V \in G: \\ \pi_i(AV)=\{0\}}} x_t^A(V) \sum_{\substack{W \in G: \\ \pi_i(BW)=\{0\}}} x_t^B(W) \right]^{r-1} \tag{5.10}$$

and, for any $U \in G$,

$$x_{t+1}^A(U) = \sum_{I \subseteq [k]} \sum_{J \subseteq [k]} \sum_{W \in G} \varepsilon^{|I|} (1-\varepsilon)^{k-|I|} x_t^A(W)\, p_J(y_t) n_{U,W,I,J}^A \tag{5.11}$$

where:

- $p_J(y_t) = \left(1 - (1-y_t)^s\right)^{|J|} (1-y_t)^{s(k-|J|)}$
- $n_{U,W,I,J}^A = \#\left\{V \in G : BV = AW + \mathbb{F}_2^J, U = V \cap \mathbb{F}_2^I\right\}$.

The equation for $\boldsymbol{x}_{t+1}^B$ is the same, simply exchanging the role of A and B everywhere.

Note that, if B is invertible, $n_{U,W,I,J}^A$ becomes simply 1 if $U = B^{-1}(AW + \mathbb{F}_2^J) \cap \mathbb{F}_2^I$ and 0 otherwise.

These update equations describe the evolution of y, x^A and x^B at one step of the decoding algorithm, under the assumption that no loops have been created up to that time (as in classical density evolution) and under the additional assumption

that labels on edges connected to information nodes all have weight one (this is true with high probability).

Fig. 5.7 helps to understand the meaning of equations (5.10) and (5.11). It shows the portion of Tanner graph corresponding to one step in the iterative decoding, from the perspective of an information and a parity node. The triangles denote the output from the channel. The check nodes can be thought as the aggregation of k bit-wise check nodes, where the i-th bit-wise check is connected to the information nodes having label $\boldsymbol{e}_i$.

Referring to Fig. 5.7(a), y_{t+1} is the probability that $m_y = \{0,1\}$. This happens only if both the message from the channel and all the $r-1$ incoming messages from check nodes give an erasure. The channel sends an erasure with probability ε. For each of the $r-1$ edges, we will now compute the probability that the message is $\{0\}$, i.e. not erased, assuming that the label is $\boldsymbol{e}_i$: the averaging on i then comes from the fact that the labels are uniformly random. Looking at the check node, we see that it sends $\{0\}$ for the i-th component when all the other $s-1$ edges incoming with label $\boldsymbol{e}_i$ carry a $\{0\}$ and both messages AV and BW from parity nodes give $\{0\}$ when restricted to the i-th component. This happens with probability $(1-y_t)^{s-1} \sum_{V:\pi_i(AV)=\{0\}} x_t^A(V) \sum_{W:\pi_i(BW)=\{0\}} x_t^B(W)$.

$\boldsymbol{x}_{t+1}^A$ is the distribution of the messages m_x in Fig. 5.7(b); let's compute the probability that $m_x = U$. Note that U is the intersection of the message received from the channel and the one coming from the check node. The channel can send any of the spaces $\mathbb{Z}_2^I$, $I \subseteq [k]$ (i.e. an erasure exactly in the components listed in the indexes set I), each with probability $\varepsilon^{|I|}(1-\varepsilon)^{k-|I|}$. The check node computes the sum of the vector spaces it receives. The combination of the ks messages from the information nodes is $\mathbb{Z}_2^J$, $J \subseteq [k]$, i.e. an erasure exactly in the components listed in J, with probability $p_J(y_t)$. In fact, a bit-wise check node j is erased when at least one of the s information nodes with label $\boldsymbol{e}_j$ carries an erasure. The check node has to sum $\mathbb{Z}_2^J$ and the message it receives on the edge labeled with A, which is AW with probability $x_t^A(W)$, for any $W \in G$. In conclusion, any triple $I \subseteq [k]$, $J \subseteq [k]$, and $W \in G$ appears with probability $\varepsilon^{|I|}(1-\varepsilon)^{k-|I|} p_J(y_t) x_t^A(W)$ and contributes to $x_{t+1}^A(U)$ $n_{U,W,I,J}^A$ times.

5.4.2 Convergence threshold and stability condition

The evolution equations (5.10) and (5.11) describe a dynamical system, with variable $\boldsymbol{z} = (y, \boldsymbol{x}^A, \boldsymbol{x}^B) \in [0,1] \times \mathcal{P}(G) \times \mathcal{P}(G)$. It is clear that if we denote by δ_V a vector in $\mathcal{P}(G)$ with a one in position V and zeros everywhere else, $\boldsymbol{z}^* := (0, \delta_{\{0\}}, \delta_{\{0\}})$ is a fixed point of the system. Since $y_t \to 0$ represents successful decoding, finding the threshold means finding up to what value of ε the system converges to $\boldsymbol{z}_0$ from the initial condition $\boldsymbol{z}_0 = (1, \delta_{\mathbb{Z}_2^k}, \delta_{\mathbb{Z}_2^k})$. This choice of $\boldsymbol{z}_0$ corresponds to the initialization

of the decoding algorithm.

Fig. 5.8 shows an example of the threshold behaviour.

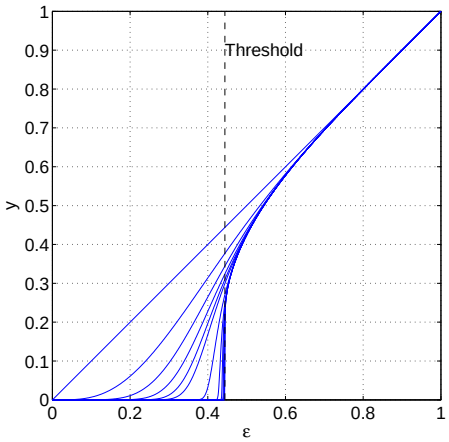


Figure 5.8. Fraction of erased info bits (y) vs. erasure probability of the BEC (ε), at iterations from 1 to 100 of the density evolution

Numerical computation of the threshold for different values of A and B can guide the choice of the inner encoder, as is discussed in Section 5.4.3.

An interesting theoretical question which is often considered in the Density Evolution literature is the stability condition: you look for conditions ensuring that the fixed point to which you wish convergence (in our case, $\boldsymbol{z}^*$) is asymptotically stable for all ε, i.e. for all ε, there exists a neighbourhood of $\boldsymbol{z}^*$ such that starting from any initial condition in that neighbourhood the system will converge to $\boldsymbol{z}^*$. This is clearly a necessary condition for convergence from the given initial condition $\boldsymbol{z}_0$, and it can provide interesting design guidelines, as it happens for the degree distributions of the irregular binary random LDPC ensemble.

In our setting, it turns out that $\boldsymbol{z}^*$ is asymptotically stable, for all ε, for any choice of A and B, provided that $r \geq 3$. This generalizes the well-known result that for the regular LDPC ensemble, with left degree at least three, the asymptotic stability of the fixed point 0 is always true. However, the proof in our setting is less trivial. You need at first to linearize the system, i.e. to compute the Jacobian matrix in $\boldsymbol{z}^*$, $J(\boldsymbol{z}^*)$. $r \geq 3$ ensures that the first line of $J(\boldsymbol{z}^*)$ is all-zero. Then you can note that $\boldsymbol{x}^A_{t+1}$ does not depend on $\boldsymbol{x}^B_t$ and depends linearly on $\boldsymbol{x}^A_t$; denote by M_A the $|G| \times |G|$ matrix describing this linear map in the case when $y_t = 0$. Analogously define M_B. Now note that the eigenvalues of $J(z^*)$ are: 0 and then the eigenvalues of M_A and M_B. Now, instead of explicitly computing the eigenvalues of A and B, which are hard to express in closed form, we prove that the linear systems on $\mathcal{P}(G)$ associated with M_A and M_B have a unique asymptotically stable fixed

point in $\delta_{\{0\}}$, by using a Lyapunov technique (see e.g. [43]): we define the function $\eta(x) = \sum_{U \in G}(\dim U)x(U)$, which can be interpreted as the average dimension of the subspaces of $\mathbb{Z}_2^k$ with respect to the probability distribution $\boldsymbol{x}$. We note that η is a linear function, $\eta(\boldsymbol{x}) \geq 0$ for all $\boldsymbol{x} \in \mathcal{P}(G)$, $\eta(\boldsymbol{x}) = 0$ if and only if $\boldsymbol{x} = \delta_{\{0\}}$, and we prove that η is strictly decreasing along the trajectories, i.e. $\eta(M_A\boldsymbol{x}) < \eta(\boldsymbol{x})$ and $\eta(M_B\boldsymbol{x}) < \eta(\boldsymbol{x})$ for all $\boldsymbol{x} \neq \delta_{\{0\}}$.

5.4.3 Simulation results

Our analysis is validated by simulation results in which at low SNR the hierarchy given by the threshold is respected. The threshold has been obtained numerically, iteratively calculating message densities and considering a maximum of 250 iterations.

All the examples simulated have $r = 4$ and $s = 4$, so that the overall rate R is $1/2$. Simulations differ for k and for the choice of $\psi(D)$, which influences both the threshold and the parameter d_2^{ψ}.

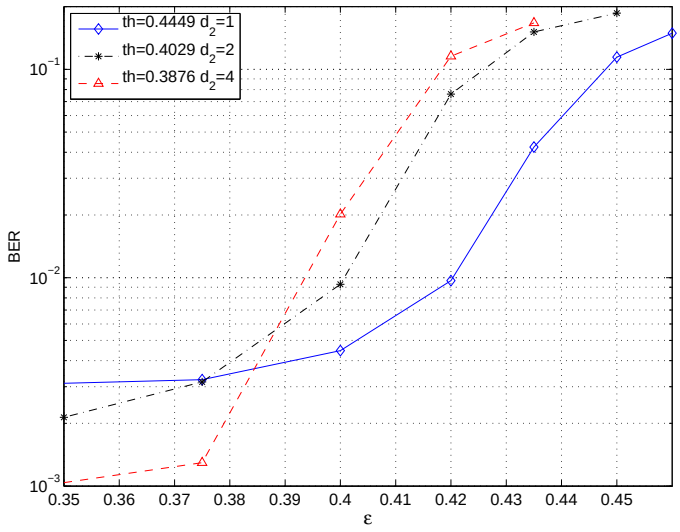


Figure 5.9. Results on BEC channel, $k = 3$, block length 2400, rate $1/2$

Fig. 5.9 shows the behaviour of the non-binary decoding algorithm with $k = 3$ for three encoders, on the BEC channel. For the low SNR region the predicted hierarchies are respected, they can be read on the graph in the BER region between 10^{-1} and 10^{-2}. Hierarchies are reversed at higher SNR, as predicted by the parameter d_2^{ψ}.

Fig. 5.10 shows analogous simulations for some codes with $k = 4$. For all the curves the matrix B has been kept fixed equal to the identity while A, starting from

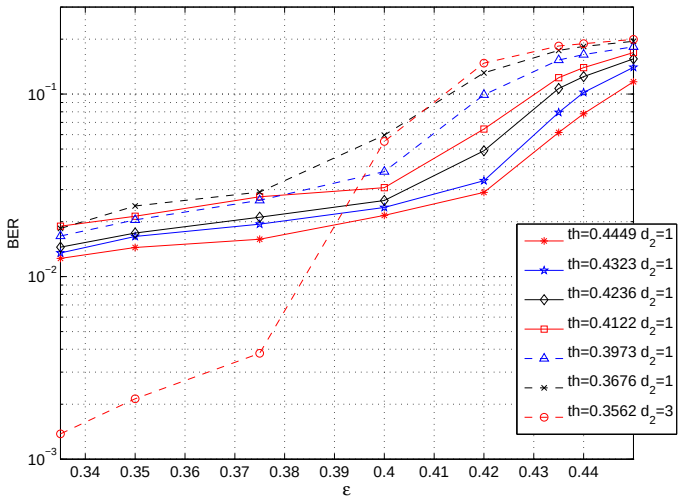


Figure 5.10. Results on BEC channel, $k = 4$, block length 2000, rate 1/2

the identity matrix, has been filled up with more and more ones: this leads to a decreasing threshold and a d_2^{ψ} which is very low when A is sparse; this suggests some relation between our design parameters and sparseness of the matrices. Simulation results are again perfectly matching with the predictions.

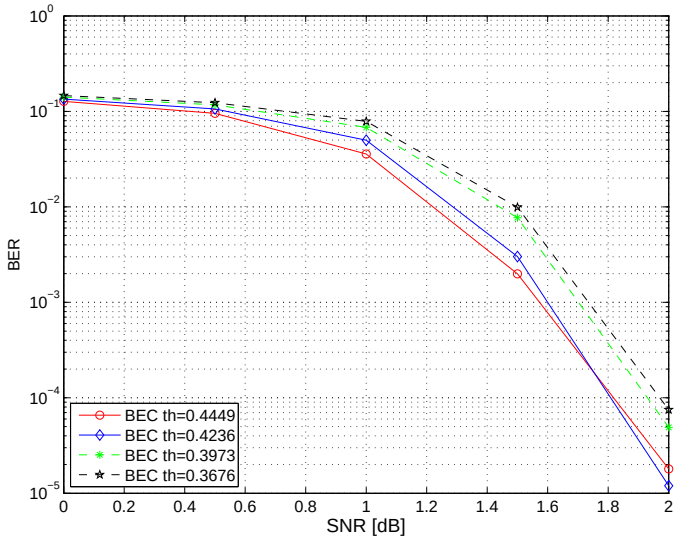


Figure 5.11. Results on AWGN channel, $k = 4$, block length 2000, rate 1/2

Fig. 5.10 shows performance of some codes on the AWGN channel. A look at

the thresholds for these codes on the BEC shows that the hierarchy is respected. This suggests that density evolution on the BEC can also provide some insight in the behaviour on other channels.

In Sections 5.2.2 and 5.3.3, we have underlined the role of the parameter d_2^{ψ}: its maximization improves performance at high SNR. Now density evolution provides an optimization criterion for low SNR: maximizing the threshold. It is well-known that these two optimizations are often in contrast, so that a compromise is necessary if both SNR regions are targeted. We want to investigate if this happens also in our setting.

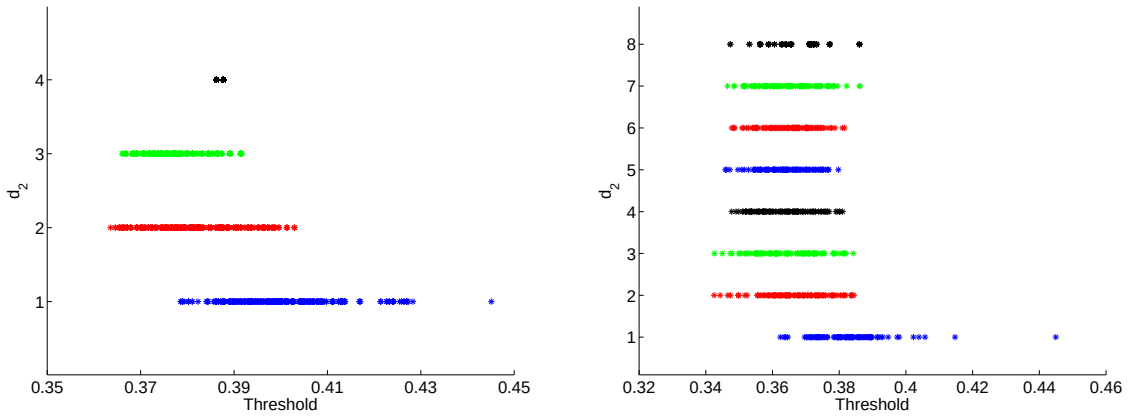


Figure 5.12. Distribution of the threshold as function of d_2^{ψ} with $k = 3, 4$

Figure 5.12 reports the threshold vs. d_2^{ψ} for a large number of choices of the pairs A, B.

These numerical results show that the best threshold corresponds to A and B being both permutation matrices; unfortunately, it is easy to prove that this condition implies $d_2^{\psi} = 1$, the same as with the simple accumulator on which we wanted to improve. For $k = 3$ we see that the values of the threshold are quite dispersed, even if there is some dependence on d_2^{ψ}, in that the lower d_2^{ψ}, the higher the maximum threshold. For $k = 4$, apart from the special case $d_2^{\psi} = 1$, the thresholds don't exhibit any apparent dependence on d_2^{ψ}. This suggests, especially for $k = 4$, the following simple design criterion: take A and B with the maximum threshold among those having the maximum d_2^{ψ}.

Chapter 6

Conclusion

In this thesis, we have considered very general serial turbo coding ensembles, where convolutional encoders on a finite Abelian group are concatenated through an interleaver which can be a permutation, or some more general homomorphic transformation. This setting includes as special cases usual binary serial and parallel turbo codes, as well as turbo trellis-coded modulation for AWGN channels with geometrically uniform input constellation, e.g. m-PSK.

We have proved in this general setting an upper bound on the average error probability which generalizes the interleaver gain result for binary serial turbo codes given in [3]. The tightness of this bound is ensured by a lower bound, which is new also for the binary case. Together, the upper and the lower bound prove that, under mild assumptions on the constituent encoders, the average error probability is vanishing when the blocklength goes to infinity, with a polynomial decay. We have also characterized both the speed of decay and the dependence of the average error probability on the channel's signal-to-noise ratio as the solution of two combinatorial optimization problems, in general involving both constituent encoders.

In the particular case of classical binary serial turbo codes, we have shown that there is no concentration of the error probability around its average, in the sense that both average and typical error probability are decreasing to zero when the blocklength grows to infinity, but the first one decreases polynomially while the latter one decreases sub-exponentially fast. The exponent of the sub-exponential decay, as well as some multiplicative constants appearing in the bounds, give design parameter for the constituents encoders of the scheme, and are perfectly matching with the ones suggested by the average-based analysis and well-known from simulations (e.g. [3]). Our typical-case analysis is based on tight bounds for the average enumerating coefficients, which use techniques specific for binary codes ([41, 2]). We conjecture that also non-binary serial concatenations have a typical error probability decreasing faster than the average, but we leave as an open problem to find some suitable tools to obtain a proof.

We have considered also another binary ensemble fitting in the general scheme discussed above. It is a generalization of Repeat-Accumulate codes, and it can also be interpreted as a family of structured linear-time encodable and decodable LDPC codes, generalizing staircase LDPC codes. The inner encoder is itself the composition of two maps, and in order to find a design criterion for its inner part we presented a modified average-based analysis, where we considered a sub-ensemble using expurgation techniques. Simulation results with usual LDPC decoders showed poor performance and poor correspondence with theoretical predictions of ML error probability. This, due to the presence of cycles in the structured part of the graph associated with the parity-check matrix. We have proposed a different decoding algorithm, where groups of nodes in the Tanner graph were associated to form a single super-node, thus breaking cycles. This new decoding algorithm allowed both an improvement of performance (at least for some codes) and the use of density evolution analysis of convergence, which was not possible in the presence of many small cycles. Simulation results show hierarchies well matching the theoretical predictions given by the density evolution threshold in the low SNR (waterfall) region and by the average-based ML error probability analysis in the medium-high SNR (error floor) region, and this allows us to give guidelines on the design of the code. There are still many open questions in the study of this family of codes. A straightforward step is to turn from regular degrees to irregular degrees in the random part of the matrix, i.e. time-varying outer repetition and inner summator codes. Density evolution equations can be easily generalized to this setting, and even though their theoretical analysis does not look simple, it is clearly possible to use them to compute numerically the threshold and thus guide the design of both the inner code and the degree distributions. Another interesting open problem is to generalize to this decoding algorithm the finite-length analysis techniques proposed in [1], which provide more refined predictions of the convergence.

A broader open problem, which looks more like a whole research area, includes the search for results on serial turbo-like codes performance under actual iterative decoding algorithms. We believe that our techniques, based on the study of weight enumerators and distances, could be used, together with new tools, to study the weight of turbo-stopping-sets, a new measure of the performance of a binary turbo decoder on the BEC introduced for parallel turbo codes in [65].

Bibliography

[1] A. Amraoui, A. Montanari, T. Richardson, and R. Urbanke, *Finite-Length Scaling for Iteratively Decoded LDPC Ensembles*, submitted to IEEE Trans. Inform. Theory (2004), available online: `http://arxiv.org/abs/cs.IT/0406050`

[2] L. Bazzi, M. Mahdian, and D. Spielman, *The minimum distance of turbo-like codes*, submitted to IEEE Trans. Inform. Theory (2003), available online: `http://basilo.kaist.ac.kr/papers/MIT/Spielman/s_12.pdf`

[3] S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollara, *Serial concatenation of interleaved codes: performance analysis, design and iterative decoding*, IEEE Trans. Inform. Theory, 44 (1998), pp. 909–926.

[4] S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollara, *Analysis, design, and iterative decoding of double serially concatenated codes with interleavers*, IEEE J. Sel. Areas Communications, 16 (1998), pp. 231–244.

[5] S. Benedetto, R. Garello, M. Mondin, and G. Montorsi, *Geometrically uniform TCM codes based on L×MPSK constellations*, IEEE Trans. Inf. Theory, 40 (1994), pp.137–152.

[6] S. Benedetto and G. Montorsi, *Unveiling turbo codes: some results on parallel concatenated coding schemes*, IEEE Trans. Inform. Theory, 42 (1996), pp. 409–428.

[7] S. Benedetto and G. Montorsi, *Design of parallel concatenated convolutional codes*, IEEE Trans. Communications, 44 (1996), pp. 591–600.

[8] A. Bennatan and D. Burshetein, *On The Application of LDPC Codes to Arbitrary Discrete Memoryless Channels*, IEEE Trans. Inf. Theory, 50 (2004), pp. 417-438.

[9] C. Berrou, A. Glavieux, and P. Thitimajshima, *Near Shannon Limit Error-Correctiong Coding and Decoding: Turbo Codes*, Proc. IEEE Int. Conf. Communications (1993), pp. 1064–1070.

[10] C. Berrou and A. Glavieux, *Near optimum error correcting coding and decoding: turbo-codes*, IEEE Trans. Communications, 44 (1996), pp. 1261–1271.

[11] B. Bollobás, *Random Graphs*, 2nd edition, Cambridge University Press, 2001.

[12] V. Borkar, *Probability Theory*, NewYork: Springer-Verlag, 1995.

[13] F. Brännström, L. K. Rasmussen, A. J. Grant, *Convergence Analysis and Optimal Scheduling for Multiple Concatenated Codes*, IEEE Trans. Inform. Theory, 51 (2005), pp. 3354–3364.

[14] M. Breiling *A logarithmic upper bound on the minimum distance of turbo codes*, IEEE Trans. Inform. Theory, 50 (2004), pp. 1692–1710.

[15] G. Como and F. Fagnani, *The capacity of Abelian group codes over symmetric channels*, submitted to IEEE Trans. Inform. Theory (2005), available online: `http://calvino.polito.it/ricerca/2005/pdf/33_2005/art_33_2005.pdf`

[16] M. C. Davey and D. J. C. MacKay, *Low density parity check codes over GF(q)*, IEEE Communications Letters, 2 (1998), pp. 159-166.

[17] D. Divsalar, *A simple tight bound on error probability of block codes with application to turbo codes*, JPL TDA Progress Report, 42-139 (1999), pp. 1–35.

[18] D. Divsalar, S. Dolinar, and F. Pollara, *Iterative turbo decoder analysis based on density evolution*, IEEE J. Sel. Areas Communications, 19 (2001), pp. 891–907.

[19] T. M. Duman and M. Salehi, *New performance bounds of turbo codes*, IEEE Trans. Communications, 46 (1998), pp. 717–723.

[20] S. Eilenberg, *Automata, machines, and languages. Vol A.*, Academic Press, 1974.

[21] H. El-Gamal and A. R. Hammons, *Analyzing the turbo decoder using the Gaussian approximation*, IEEE Trans. Inform. Theory, 47 (2001), pp. 671–686.

[22] U. Erez and G. Miller, *The ML Decoding Performance of LDPC Ensembles Over* $\mathbb{Z}_q$, IEEE Trans. Inform. Theory, 51 (2005), pp. 1871–1879.

[23] F. Fagnani, *Performance of parallel concatenated coding schemes*, accepted for publication, IEEE Trans. Inform. Theory (2008). Pre-print available on-line: `http://calvino.polito.it/ricerca/2004/pdf/31_2004/art_31_2004.pdf`

[24] F. Fagnani, R. Garello, B. Scanavino, and S. Zampieri, *Geometrically uniform parallel concatenated coded modulation schemes*, in preparation.

[25] F. Fagnani and S. Zampieri, *Convolutional codes over finite Abelian groups: some basic results*, in Codes, systems and graphical models, B. Marcus and J. Rosenthal, eds, IMA Volumes in Mathematics and its applications, vol. 123, pp. 327–346, 2001.

[26] F. Fagnani and S. Zampieri, *System-theoretic properties of convolutional codes over rings*, IEEE Trans. Inform. Theory, 47 (2001), pp. 2256–2274.

[27] F. Fagnani and S. Zampieri, *Minimal and systematic convolutional codes over finite Abelian groups*, Linear Algebra Appl., 378 (2004), pp. 31–59.

[28] G. D. Forney, Jr., *Concatenated codes*, Cambridge, MA: MIT Press, 1966.

[29] G. D. Forney, Jr., *Geometrically uniform codes*, IEEE Trans. Inform. Theory, 37 (1991), pp. 1241–1260.

[30] G. D. Forney, Jr. and M. D. Trott, *The dynamics of group codes: state spaces, trellis diagrams and canonical encoders*, IEEE Trans. Inform. Theory, 39 (1993), pp. 1491–1513.

[31] G. D. Forney, Jr. and M. D. Trott, *The dynamics of group codes: Dual Abelian Group Codes and Systems*, IEEE Trans. Inform. Theory, 50 (2004), pp. 2935–2965.

[32] C. Fragouli, R.D. Wesel, *Turbo-Encoder design for symbol-interleaved parallel concatenated Trellis-Coded Modulation*, IEEE Transactions on Communications, 49 (2001), pp. 425–435.

[33] R. G. Gallager, *Low Density Parity Check Codes*, Cambridge, MA: MIT Press, 1963.

[34] R. Garello, G. Montorsi, S. Benedetto, D. Divsalar, and F. Pollara, *Labelings and encoders with the uniform bit error property with applications to serially concatenated trellis codes*, IEEE Trans. Inform. Theory, 48 (2002), pp. 123–136.

[35] A. Graell i Amat, G. Montorsi, and F. Vatta, *Analysis and design of rate compatible serial concatenated convolutional codes*, Proc. Int. Symp. Inform. Theory (2005), pp. 607-611.

[36] T. W. Hungerford, *Algebra*, Springer-Verlag, 1974.

[37] I. Ingemarsson, *Commutative Group Codes for the Gaussian Channel*, IEEE Trans. Inform. Theory, 19 (1973), pp. 215–219.

[38] H. Jin, A. Khandekar, and R. J. McEliece, *Irregular Repeat-Accumulate Codes*, Proc. Intern. Symposium Turbo Codes (2000).

[39] H. Jin and R. J. McEliece, *Coding theorems for turbo code ensembles*, IEEE Trans. Inform. Theory, 48 (2002), pp. 1451–1461.

[40] R. Johannesson, Z.-X. Wan, and E. Wittenmark, *Some structural properties of convolutional codes over rings*, IEEE Trans. Inform. Theory, 44 (1998), pp. 839-845.

[41] N. Kahale and R. Urbanke, *On the minimum distance of parallel and serially concatenated codes*, submitted to IEEE Trans. Inform. Theory (1997), available online: `http://lthcwww.epfl.ch/papers/KaU.ps`

[42] R. E. Kalman, P. L. Falb, and M. A. Arbib, *Topics in mathematical system theory*, McGraw Hill, 1969.

[43] J. P. LaSalle, *The Stability and Control of Discrete Processes*, Applied Mathematical Sciences, 62, SpringerVerlag, 1986.

[44] S. Le Goff, A. Glavieux, and C. Berrou, *Turbo-codes and high spectral efficiency modulation*, Proc. IEEE Int. Conf. Communications (1994), pp. 645–649.

[45] H.-A. Loeliger, *Signal sets matched to groups*, IEEE Trans. Inform. Theory, 37 (1991), pp. 1675–1682.

[46] H.-A. Loeliger and T. Mittelholzer, *Convolutional Codes Over Groups*, IEEE Trans. Inform. Theory, 42 (1996), pp. 1660–1686.

[47] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, D. A. Spielman *Improved low-density parity-check codes using irregular graphs* IEEE Trans. Inform. Theory, 47 (2001), pp. 585–598.

[48] D. J. C. MacKay, *Good Error Correcting Codes Based On Very Sparse Matrices*, IEEE Trans. Inform. Theory, 45 (1999), pp. 399–431.

[49] D. J. C. MacKay and R. M. Neal, *Good Codes Based on Very Sparse Matrices*, Cryptography and Coding. 5th IMA Conf. (Cirencester, UK), LNCS 1025: 100-111. Berlin: Springer, 1995.

[50] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes.* Amsterdam: North-Holland, 1977.

[51] R. J. McEliece, D. J. C. MacKay, and J.-F. Cheng, *Turbo Decoding as an Instance of Pearl's "Belief Propagation" Algorithm*, IEEE J. Sel. Areas Communications, 16 (1998), pp. 140–152.

[52] G. Miller and D. Burshetein, *Bounds on the Maximum Likelihood Decoding Error Probability of Low–Density Parity–Check Codes*, IEEE Trans. Inform. Theory, 47 (2001), pp. 2696–2710.

[53] H. Ogiwara, A. Mizutome, and K. Koike, *Performance evaluation of parallel concatenated Trellis-Coded Modulation*, IEICE Trans. Fundamentals, E84-A (2001), pp. 2410–2417.

[54] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, San Mateo, CA: Morgan Kaufmann, 1988.

[55] A. Perotti and S. Benedetto, *An Upper Bound on the Minimum Distance of Serially Concatenated Convolutional Codes*, IEEE Trans. Inform. Theory, 52 (2006), pp. 5501–5509.

[56] H. D. Pfister and P. H. Siegel, *The Serial Concatenation of Rate-1 Codes Trough Uniform Random Interleavers*, IEEE Trans. Inform. Theory, 49 (2003), pp. 1425–1438.

[57] H. D. Pfister, I. Sason and R. Urbanke, *Capacity-Achieving Ensembles for the Binary Erasure Channel With Bounded Complexity*, IEEE Trans. Inform. Theory, 51 (2005), pp. 2352–2379.

[58] V. Rathi and R. Urbanke, *Density evolution, thresholds and the stability condition for non-binary LDPC codes*, IEE Proc. on Communications, 152 (2005), pp. 1069-1074.

[59] T. J. Richardson, *The geometry of turbo-decoding dynamics*, IEEE Trans. Inform. Theory, 46 (2000), pp. 9–23.

[60] T. J. Richardson and R. Urbanke, *The Capacity of Low-Density Parity-Check Codes Under Message-Passing Decoding*, IEEE Trans. Inform. Theory, 47 (2001), pp. 599-618.

[61] T. J. Richardson, M. A. Shokrollahi, and R. Urbanke, *Design of Capacity-Approaching Irregular Low-Density Parity-Check Codes*, IEEE Trans. Inform. Theory, 47 (2001), pp. 619–637.

[62] T. Richardson and R. Urbanke, *Efficient Encoding of Low-Density Parity-Check Codes*, IEEE Trans. Inform. Theory, 47 (2001), pp. 638–656.

[63] T. Richardson and R. Urbanke, *Modern Coding Theory*. Online: `http://lthcwww.epfl.ch/mct`

[64] P. Robertson and T. Wörz, *Novel bandwidth efficient coding scheme employing turbo-codes*, Proc. IEEE Int. Conf. Communications (1996), pp. 962–967.

[65] E. Rosnes, Ø. Ytrehus, *Turbo Decoding on the Binary Erasure Channel: Finite-Length Analysis and Turbo Stopping Sets*, submitted to IEEE Trans. Inform. Theory (2006), available online: `http://arxiv.org/PS_cache/cs/pdf/0602/0602072v1.pdf`

[66] A. Roumy, S. Guemghar, G. Caire, and S. Verdú, *Design Methods for Irregular Repeat-Accumulate Codes*, IEEE Trans. Inform. Theory, 50 (2004), pp. 1711–1727.

[67] I. Sason and S. Shamai (Shitz) *Improved upper bounds on the ML decoding error probability of parallel and serially concatenated turbo codes via their ensemble distance spectrum*, IEEE Trans. Inform. Theory, 46 (2000), pp. 24–47.

[68] I. Sason, E. Telatar, and R. Urbanke, *The asymptotic input-output weight distributions and thresholds of convolutional and turbo-like encoders*, IEEE Trans. Inform. Theory, 48 (2002), pp. 3052–3061.

[69] S. Shamai (Shitz) and I. Sason, *Variations on the Gallager bounds, connections and applications*, IEEE Trans. Inform. Theory, 48 (2002), pp. 3029–3051.

[70] C. E. Shannon, *A Mathematical Theory of Communication*, Bell System Technical Journal, 27 (1948), pp. 379–423 and pp. 623–656.

[71] D. Slepian, *Group codes for the Gaussian channel*, Bell System Technical Journal, 47 (1968), pp. 575–602.

[72] D. Slepian, *On Neighbor Distances and Symmetry in Group Codes*, IEEE Trans. Inform. Theory, 17 (1971), pp. 630–632.

[73] *Special Issue on Codes on Graphs and Iterative Algorithms*, B. Frey, R. Koetter, G. D. Forney, Jr., F. R. Kschischang, R. J. McEliece, and D. A. Spielman, Editors, IEEE Trans. Inform. Theory, vol. 47 n. 2, 2001.

[74] D. Sridhara and T. E. Fuja, *LDPC Codes Over Rings for PSK Modulation*, IEEE Trans. Inform. Theory, 51 (2005), pp. 3209–3220.

[75] E. Telatar and R. Urbanke, *On the ensemble performance of turbo codes*, Proc. IEEE Int. Symp. Inform. Theory (1997), pp. 105–105.

[76] S. ten Brink, *Convergence of iterative decoding*, Electronics Letters, 35 (1999), pp. 806–808.

[77] G. Ungerboeck, *Channel Coding with Multilevel/Phase Signals*, IEEE Trans. Inform. Theory, 28 (1982), pp. 55–67.

[78] N. Wiberg, *Codes and Decoding on General Graphs*, Ph.D. thesis, Linköping University, Linköping, Sweden, 1996.

[79] J. S. Yedidia, W. T. Freeman, and Y. Weiss, *Constructing Free-Energy Approximations and Generalized Belief Propagation Algorithms*, IEEE Trans. Inform. Theory, 51 (2005), pp. 2282–2312.