Analysis of the Asymptotic Performance of Turbo Codes

by

Mohammad Hadi Baligh

A thesis
presented to the University of Waterloo
in fulfilment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Electrical and Computer Engineering

Waterloo, Ontario, Canada, 2006

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

Battail [1989] shows that an appropriate criterion for the design of long block codes is the closeness of the normalized weight distribution to a Gaussian distribution. A subsequent work shows that iterated product of single parity check codes satisfy this criterion [1994]. Motivated by these earlier works, in this thesis, we study the effect of the interleaver on the performance of turbo codes for large block lengths, $N \to \infty$. A parallel concatenated turbo code that consists of two or more component codes is considered. We demonstrate that for $N \to \infty$, the normalized weight of the systematic $\widehat{w_1} = \frac{w_1}{\sqrt{N}}$, and the parity check sequences $\widehat{w_2} = \frac{w_2}{\sqrt{N}}$ and $\widehat{w_3} = \frac{w_3}{\sqrt{N}}$ become a set of jointly Gaussian distributions for the typical values of $\widehat{w}_i, i = 1, 2, 3$, where the typical values of $\widehat{w}_i$ are defined as $\lim_{N\to\infty} \frac{\widehat{w}_i}{\sqrt{N}} \neq 0, 1$ for $i = 1, 2, 3$. To optimize the turbo code performance in the waterfall region which is dominated by high-weight codewords, it is desirable to reduce ρ_{ij}, $i, j = 1, 2, 3$ as much as possible, where ρ_{ij} is the correlation coefficient between $\widehat{w}_i$ and $\widehat{w_j}$. It is shown that: (i) $\rho_{ij} > 0$, $i, j = 1, 2, 3$, (ii) $\rho_{12}, \rho_{13} \to 0$ as $N \to \infty$, and (iii) $\rho_{23} \to 0$ as $N \to \infty$ for "almost" any random interleaver. This indicates that for $N \to \infty$, the optimization of the interleaver has a diminishing effect on the distribution of high-weight error events, and consequently, on the error performance in the waterfall region. We show that for the typical weights, this weight distribution approaches the average spectrum defined by Poltyrev [1994]. We also apply the tangential sphere bound (TSB) on the Gaussian distribution in AWGN channel with BPSK signalling and show that it performs very close to the capacity for code rates of interest. We also study the

statistical properties of the low-weight codeword structures. We prove that for large block lengths, the number of low-weight codewords of these structures are some Poisson random variables. These random variables can be used to evaluate the asymptotic probability mass function of the minimum distance of the turbo code among all the possible interleavers. We show that the number of indecomposable low-weight codewords of different types tend to a set of independent Poisson random variables. We find the mean and the variance of the union bound in the error floor region and study the effect of expurgating low-weight codewords on the performance. We show that the weight distribution in the transition region between Poisson and Gaussian follows a negative binomial distribution. We also calculate the interleaver gain for multi-component turbo codes based on these Poisson random variables. We show that the asymptotic error performance for multi-component codes in different weight regions converges to zero either exponentially (in the Gaussian region) or polynomially (in the Poisson and negative binomial regions) with respect to the block length, with the code-rate and energy values close to the channel capacity.

Acknowledgments

This thesis would not exist without the love for perfection and I praise the most perfect. This work was possible with help and support from many individuals and I wish to thank all of them.

I would like to thank my supervisor Professor Amir Keyvan Khandani whose valuable support, advice and comments made this work possible. His tireless effort, advice and guidance helped me learn valuable lessons regarding research, presentation, and life. The chance to work under Amir's supervision has been the most important opportunity in my educational life. His care, encouragement and support helped me in all aspects of my work. I would like to thank Amir for his bright and smart ideas throughout this work, his great personality and his editorial guidance to improve the quality of this thesis and the rest of my research works. I also wish to thank him for his close friendship at the same time as his great supervision.

I also wish to thank the members of the evaluation committee: Prof. Andrew J. Heunis, Prof. Murat Uysal, Prof. Giuseppe Tenti for their supportive efforts and constructive advices. It has been a great honor and pleasure to have Prof. Rudiger L. Ürbanke from EPFL, Switzerland as my external committee member. Special thanks for his insightful recommendations and suggestions which helped me a lot in improving this thesis.

I wish to thank all my friends from the Coding and Signal Transmission (CST) lab for all the fruitful discussions that we had on the topic of my thesis or any other subject. I have been lucky to have the opportunity to work with one of the most talented and brilliant

research groups on the face of this planet and I owe my success in part to my endowed colleagues from whom I learned very much. I wish that our friendship will continue forever.

I would like to thank (in the alphabetical order) CITO (OCE), the Government of Ontario, Nortel Networks, NSERC and the University of Waterloo for their direct financial support of my work in the form of scholarships and awards.

I would like to acknowledge everybody back home, my family and my friends, who encouraged me by their love and support from thousands of miles away.

My special thanks goes to Samira Bashiri Amid, my beloved wife, who has been my life glory and motivation. I would like to thank her for her support, help and patience in the process of this work.

Dedication

To my parents: Alireza and Badrolsadat

and

To my beloved wife: Samira

Contents

List of Tables

List of Figures

Chapter 1

Introduction

The advent of turbo codes [1] is one of the most important developments in coding theory in many years. These codes can achieve a near capacity error correcting performance with a relatively simple decoding method. Turbo codes consist of two or more recursive convolutional codes (RCCs) which are connected in parallel or serial via pseudo-random interleavers.

A typical error performance of a turbo code consists of two regions: the waterfall region and the error floor region. In the waterfall region, the error performance is determined by high-weight codewords, whereas in the error floor region, the performance is determined by low-weight codewords.

One of the tools to assess the performance of a binary linear block code with maximum likelihood (ML) decoding is its weight distribution[1]. While ML decoding is not feasible

[1]Number of codewords for different possible weight.

for turbo codes, it provides insight into the potential performance of these codes. Because of the existence of the interleaver, the analysis based on the actual weight distribution becomes very complicated. Benedetto and Montorsi introduce "uniform interleaving" technique and evaluate the "average weight distribution" of the code, which is defined as the average weight distribution among all codes generated with various possible interleavers [2].

Although turbo codes are not random coding schemes, with a randomly chosen interleaver, their pairwise distance spectrum is very similar to that of the random codes. In [3, 4], it is shown that turbo codes belong to the class of weakly random-like codes; although their frame error rate (FER) performance is poor, the bit error rate (BER) remains low up to the neighborhood of the channel capacity. In the class of weakly random-like codes, the normalized weight distribution has similarity with that of random coding measured by cross entropy [5]. Battail shows that an appropriate criterion for the design of long block codes is the closeness of the normalized weight distribution to Gaussian rather than large minimum distance [6]. Reference [7] provides techniques to apply the channel coding theorem and the resulting error exponent, which was originally derived for random block code ensembles, to the ensembles of codes with fewer restrictive randomness requirements.

Evaluating the performance of turbo codes is not feasible because of their complex structure. As a result, providing some bounds on the error performance is helpful to evaluate the potential performance of the code. Based on the weight distribution of turbo codes and by using Gallager's bounding techniques [8], some upper bounds on the performance of turbo codes are derived in [9–11].

It is known that using a pseudo-random interleaver in turbo codes guarantees an excellent BER performance, but a certain number of low-weight codewords are generated, resulting in a small minimum distance and the appearance of an error floor. The structure and the number of such low-weight codewords are studied in [12] and [13].

In this thesis, the weight distribution of turbo codes is addressed and it is proved that the weights of the systematic and parity streams for their typical values tend to a set of uncorrelated, and hence, independent, Gaussian random variables for a randomly chosen interleaver and for any nontrivial recursive convolutional code. We show that with probability one, in the waterfall region, a randomly chosen interleaver performs as well as the best interleaver. We also show that Gaussian weight spectrum is very close to the "average spectrum" [14]. The performance of a code with an average spectrum is very close to that of a capacity-achieving random code with binary phase shift keying (BPSK) signaling over an additive white Gaussian noise (AWGN) channel. We apply the tangential sphere bound (TSB) on the frame error rate of a code with asymptotically Gaussian weight distribution and find the region of rate and signal-to-noise ratio (SNR) where the error exponent is positive and hence, the error probability converges to zero as the block length increases. We show that the achievable rate is very close to the capacity for code rates of interest.

We also investigate the effect of the interleaver optimization on the error floor region. It is known that the low-weight codewords do not follow the Gaussian distribution and they are more important in determining the performance of the code in the error floor region

(at high SNR). Therefore, unlike in the waterfall region, the optimization of the component codes and the interleaver affect the performance in the error floor region. In [12], it is reported that as the block length increases, the low-weight codewords of a few special structures remain probable, and the expected number of low-weight codewords of each such structure remains finite as the block length tends to infinity. In this thesis, we show that the asymptotic probability mass function of the number of low-weight codewords of each structure is a Poisson random variable. We also show that indecomposable low-weight codewords constitute a set of independent Poisson random variables. We study the statistical properties of these codewords based on asymptotically possible low-weight codewords and derive the mean (and the variance) of the number of decomposable and indecomposable low-weight codewords. By means of these random variables, the probability mass function of the turbo code minimum distance, and the mean and the variance of the union bound in the error floor region, are evaluated.

The Gaussian approximation is valid for high-weight codewords and the Poisson distribution is valid for low-weight codewords. We show that the weight distribution in the transition region where the spectrum emerges from Poisson to Gaussian is negative binomial and we show that the effect of the codewords in this region on the error performance is negligible.

In [15], it is indicated that using $J > 2$ component codes improves the distance properties of turbo codes, resulting in a better performance when ML decoding is used. Here, we show that the Poisson distribution of low-weight codewords guarantees that for a turbo

code with J component codes and randomly chosen interleavers, the interleaver gain is $J-2$ which is the same as for the uniformly interleaved code reported in [16]. Our results show that the overall performance of multi-component turbo codes is very close to the capacity for BPSK signalling over an AWGN channel, because: (i) the error probability due to high-weight codewords exponentially tends to zero for SNR values close to the capacity, and (ii) the low-weight codewords result in an error floor which decreases polynomially as the block length increases. Finally, observing that the number of low-weight codewords is small, we discuss a method to expurgate the low-weight codewords following the method introduced in [17, 18], and show that the interleaver gain can be increased for multi-component turbo codes by expurgating low-weight codewords.

This thesis is organized as follows. In Chapter 2, we study the basic structure of turbo codes and the component codes and the typical performance of turbo codes and bounds on the performance of the code. In Chapter 3, we study the asymptotic weight distribution of the code for high-weight codewords and apply the TSB on the error performance of the code in the waterfall region. In Chapter 4, the statistical properties of the low weight codewords and their effect on the error floor is studied. In Chapter 5, the contributions of this thesis are summarized. Chapter 6 includes some future research directions.

Chapter 2

Basic Structure of Turbo Codes

2.1 Chapter Overview

Designing codes that achieve transmission rates close to the channel capacity defined in the Shannon's celebrated work [19] has been an attractive subject of research for decades. However, almost no near capacity coding schemes with practical encoding and decoding were known for about half a century. Turbo codes [1] presented in 1993 by Berrou achieve code rates very close to the capacity limit for a Gaussian channel over a wide range of signal-to-noise ratios with practical encoding and decoding algorithms.

The basic idea behind turbo codes is to make use of some recursive convolutional codes (RCC) connected through some interleavers. The resulting linear block code has a weight distribution which is very close to the distance spectrum of random codes [20].

The low-complexity suboptimal decoding algorithm introduced in [1] is based on an

iterative algorithm which employs a soft-output decoder for each of the constituent codes. In each iteration, the soft-input soft-output decoding improves the reliability values and eventually, under certain conditions, these reliability values converge to a valid codeword[1]. The complexity of this algorithm is proportional to the block length[2] and the number of iterations, while the complexity of the maximum likelihood decoding increases exponentially with the number of information bits.

The presence of the pseudo-random interleaver makes it difficult to evaluate the performance of turbo codes. However, the performance of turbo codes can be estimated by using bounding techniques. Some of these techniques use the weight distribution of the code to compute some upper bounds on the error performance.

The weight distribution of turbo codes is affected by the weight distribution of recursive convolutional codes and the interleaver structure. Although there are some analytical approaches to compute the weight distribution of RCCs, it is practically infeasible to compute the weight distribution of a turbo code because of the effect of the interleaver. The average weight distribution of the code among all possible interleavers known as the weight distribution under *uniform interleaving* is used to bound the performance of turbo codes.

[1] A valid vector of coded bits.

[2] Number of coded bits in each codeword.

2.2 Basic Structure of Turbo Codes

Conventional turbo codes consist of two (or more) convolutional codes connected in serial or in parallel via some pseudo-random interleavers. Other classes of turbo codes include bandwidth-efficient turbo codes [21] and turbo codes based on block constituent codes [22]. In this thesis, we focus on parallel concatenated turbo codes with recursive convolutional codes as their constituent codes.

Figure 2.1 presents a block diagram of an encoder of a systematic turbo code with a block length N that is composed of two recursive convolutional codes (RCC). The information bits are fed to the first RCC and after being interleaved are passed through the second constituent encoder. The resulting codeword consists of the systematic bits, $b_1(i)$, and two parity check streams, $b_2(i)$, $b_3(i)$, $i = 1, 2, \ldots, N$.

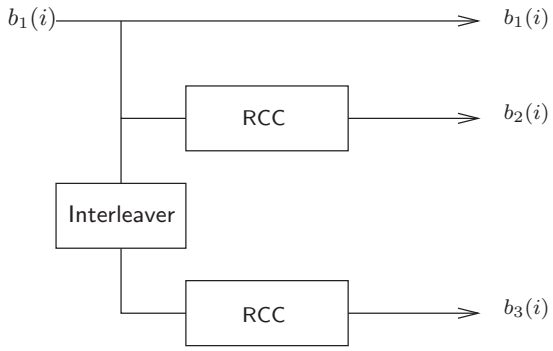

Figure 2.1: Basic structure of the turbo encoder.

The coding rate of this code is 1/3. Higher code rates are achievable by puncturing parity check bits (and even systematic bits). Using more constituent codes and/or constituent codes with lower code rates result in codes with rates lower than 1/3.

The Hamming weight of a codeword in a binary[3] block code is the number of ones in that codeword. For the turbo code presented in figure 2.1, the Hamming weight of the output codeword is equal to the sum of the weights of the b_1, b_2 and b_3 sequences over a block denoted by w_1, w_2, and w_3, respectively.

2.2.1 Linear Feedback Shift Registers

A turbo encoder employs two or more constituent recursive convolutional codes. Each code is a linear feedback shift register (LFSR) with an infinite impulse response (IIR). Here, we review the LFSR sequences and study the properties of the RCCs as the components of turbo codes. The properties of shift register sequences is studied by Golomb in [23].

A binary LFSR consists of some memory elements, each storing a binary variable, $b \in \{0, 1\}$. Figure 2.2.1 represents the basic structure of a binary LFSR with n memory elements.

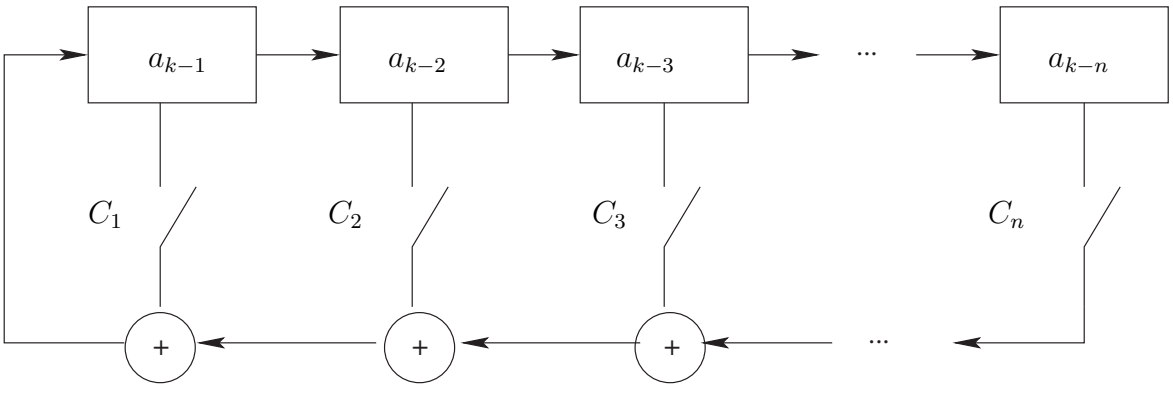

Figure 2.2: Binary linear feedback shift register.

[3]Only consisting of ones and zeroes

The binary values stored in all the memory elements is called the state of the LFSR. The output sequence is a function of the positions of the switches in figure 2.2.1 indicated by binary variables $C_i,\ i = 1, 2, \ldots, n$. A one indicates a closed switch, and a zero indicates an open switch. Note that in order to have n memory elements, C_n should be one. From the structure of an LFSR, it can be shown that the generated sequence $\{a_n\}$ satisfies the recursive equation

$$a_k = \sum_{i=1}^{n} C_i a_{k-i}. \tag{2.1}$$

The initial state of the LFSR is shown by $a_{-1}, a_{-2}, \ldots, a_{-n}$. The sequence generated by an LFSR is a function of its initial state, as well as the positions of the switches. The sequence $\{a_n\} = \{a_0, a_1, a_2, \ldots\}$ can be described by its generating function $G(x)$, defined by

$$G(x) = \sum_{i=0}^{\infty} a_i x^i = \frac{\sum_{i=1}^{n} C_i x^i \sum_{j=1}^{i} a_{-j} x^{-j}}{1 - \sum_{i=1}^{n} C_i x^i} = \frac{g(x)}{f(x)}. \tag{2.2}$$

The polynomial

$$f(x) = 1 - \sum_{i=1}^{r} C_i x^i \tag{2.3}$$

is called the *characteristic polynomial* of the shift register. Since $c_n = c_0 = 1$, $f(x)$ is a monic polynomial of degree n [23].

In an LFSR, the next entry in the sequence and the next state depend only on the current state. If a particular state occurs for the second time, the rest of the sequence will be periodic from that point on. Therefore, the maximum period of an LFSR sequence is $2^n - 1$, which corresponds to one cycle through each of the $2^n - 1$ non-zero states. A

sequence with a period of $2^n - 1$ is commonly known as a maximum length sequence (MLS) or an *m-sequence*. The period of an LFSR sequence with characteristic polynomial $f(x)$ is the smallest integer p such that $f(x)$ divides $1 - x^p$ (modulo 2 arithmetic) [23]. The integer p is also called the exponent of $f(x)$. A necessary, but not sufficient condition on $f(x)$ to produce an m-sequence is that $f(x)$ is irreducible. The number of polynomial of degree n with maximum exponent is given by $\phi(2^n - 1)/n$ where $\phi(\cdot)$ is the Euler ϕ-function [23].

2.3 Typical Performance of Turbo Codes

A typical error performance of a turbo code is illustrated in figure 2.3. The performance of the code is divided into two regions: the waterfall region and the error floor region. For signal to noise ratios close to the capacity, a small increase in the received bit-energy results in a considerable improvement in the error probability. This region of performance is called the waterfall region. In the error floor region, the performance does not improve significantly as the SNR increases and the error performance remains almost constant for a wide range of SNR values.

In the waterfall region, the error performance is determined by the codewords of high weights. As we will see in the following chapter, for large block turbo codes, the asymptotic weight distribution of high-weight codewords is Gaussian with parameters which are independent of the chosen component codes and the structure of the interleaver. As a result, the performance of long block turbo codes does not improve very much with interleaver

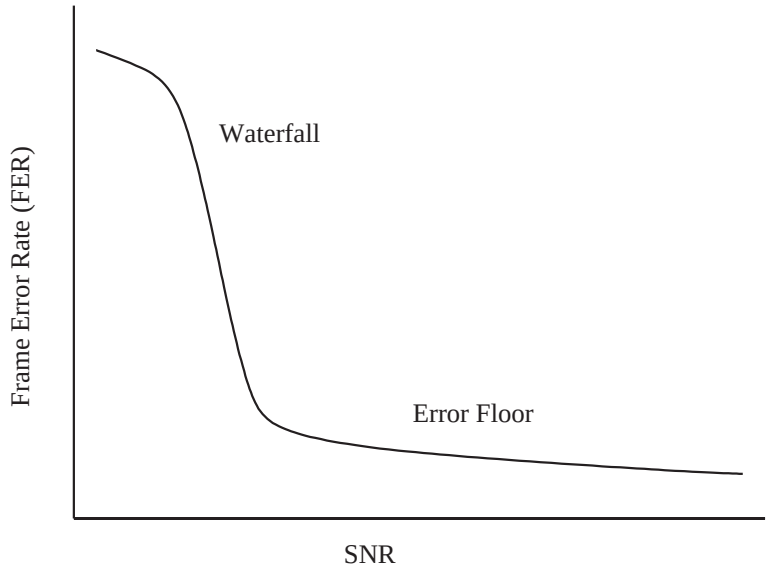

Figure 2.3: Typical error performance of a turbo code over an AWGN channel.

and RCC optimization.

In the error floor region, the performance is determined by low-weight codewords. The weight distribution of low-weight codewords is determined by the RCC and interleaver selection. As a result, the error floor can be lowered by optimizing the constituent codes and the interleaver. Turbo codes are powerful codes in part due to the fact that the number of their low-weight codewords remains small as the block length increases. This is unlike many other known block codes where the number of such codewords increases with the block length. A recursive convolutional encoder produces many nonzero parity bits from a low-weight systematic stream, unless for a small number of certain systematic patterns. The interleaver rearranges the bit positions in such streams and as a result, with a high probability, at least one of the parity streams will have a high weight. However, still a small

number of low-weight codewords may exist, as the interleaver may map a low-parity-weight pattern to another low-parity-weight pattern. Therefore, turbo codes have a relatively low minimum distance[4].

2.4 Weight Distribution of Linear Binary Block Codes

In a binary linear codebook[5], the binary addition of any two codewords is another codeword. As a result, the all-zero codeword is a valid codeword for any binary linear code. For a binary linear code, each codeword is located with the same set of distances from other codewords as the all-zero codeword is. This is called the distance invariance property. In this case, all codewords have the same error protection because the shape of all *Voronoi* regions are the same. The Voronoi region of a codeword is the region where the optimal decoder decodes that codeword when the received vector falls in that region. For AWGN channel and equiprobable codewords, each point belongs to the Voronoi region of the codeword with the shortest Euclidean distance. In this case, the Voronoi region of each codeword is surrounded by all the median planes between that codeword and its neighboring codewords.

The shape of the Voronoi region determines the error performance. An error occurs when the received vector is not in the Voronoi region of the actual transmitted codeword. Larger Euclidean distance between codewords results in a larger Voronoi region, and hence,

[4]The Hamming weight of the nearest codeword with respect to the all-zero codeword.

[5]Set of all possible codewords.

a better error performance. For a binary block code, larger Euclidean distance is equivalent to larger Hamming distance[6] between different codewords. The error protection is mainly affected not only by the distance of the nearby codewords, but also by the number of such codewords.

Weight distribution of a linear code is defined as the number of codewords of different weights. A useful tool to show the weight distribution of a code is its input-output weight enumerating function (IOWEF). IOWEF shows how the weight of the coded bits relates to the systematic weight. For the turbo code shown in figure 2.1, the IOWEF is given by

$$\mathbf{A}_{w_1,w_2,w_3}(W_1,W_2,W_3) = \sum_{w_1,w_2,w_3} A_{w_1,w_2,w_3} W_1^{w_1} W_2^{w_2} W_3^{w_3}, \tag{2.4}$$

where A_{w_1,w_2,w_3} indicates the number of codewords with the systematic weight of w_1 and the parity weights of w_2 and w_3. For this code, the total weight is the sum of the systematic and the parity weights ($w = w_1 + w_2 + w_3$) and so the weight enumerating function (WEF) of the code can be written as

$$\begin{aligned} \mathbf{A}_w(W) &= \mathbf{A}_{w_1,w_2,w_3}(W_1 = W, W_2 = W, W_3 = W) \\ &= \sum_{w_1,w_2,w_3} A_{w_1,w_2,w_3} W^{w_1} W^{w_2} W^{w_3} \\ &= \sum_{w} \sum_{w1+w_2+w_3=w} A_{w_1,w_2,w_3} W^{w} \\ &= \sum_{w} A_w W^w, \end{aligned} \tag{2.5}$$

where

$$A_w = \sum_{w1+w_2+w_3=w} A_{w_1,w_2,w_3}. \tag{2.6}$$

[6]Hamming weight of the binary addition of the two codewords.

Weight distribution can be used to evaluate some bounds on the error performance when maximum likelihood (ML) decoding is used. Although ML decoding is not practically feasible for turbo codes, this analysis provides insight into the performance of turbo codes. Reference [24] presents a method to determine the asymptotic weight distribution of various concatenated code ensembles. It also provides a method to derive lower bounds on the thresholds of these ensembles under maximum-likelihood (ML) decoding.

2.4.1 Uniform Interleaving

For parallel concatenated block codes, such as turbo codes, two linear systematic codes C_1 and C_2 are linked by an interleaver. In order to obtain the weight enumerating function of such a parallel code, the calculation must take into account each constituent code and the interleaver structure. Since this calculation becomes impractical even for small block lengths, Benedetto and Montorsi introduced an abstract interleaver which they called a uniform interleaver [2]. In [25], a simple approximation of the performance of parallel concatenated turbo codes with uniform interleaving based on the union bound is obtained.

A uniform interleaver of length N is a probabilistic device which maps a given input word of weight w_1 into all the distinct $\binom{N}{w_1}$ permutations with equal probability $\frac{1}{\binom{N}{w_1}}$. Suppose that there are A_1 different systematic patterns of weight w_1 which result in parity weight w_2 by the first RCC and there are A_2 different patterns of the same systematic weight resulting in a parity weight of w_3 in the second RCC. The definition of the uniform interleaver results in a weight enumerating function for the second code which is indepen-

dent of the first code and hence, the coefficients of the weight enumerating function of the code will be

$$A_{w_1,w_2,w_3} = \frac{A_1 A_2}{\binom{N}{w_1}}. \tag{2.7}$$

The term $\binom{N}{w_1}$ accounts for the number of different ways to interleave a systematic pattern of weight w_1, where N is the code block length.

Therefore, the input-output weight enumerating function for the parallel code can be calculated as follows:

$$\mathbf{A}_{w_1,w_2,w_3}(W_1, W_2, W_3) = \sum_{w_1,w_2,w_3} \frac{A^{(C_1)}_{w_1,w_2} A^{(C_2)}_{w_1,w_3}}{\binom{N}{w_1}} W_1^{w_1} W_2^{w_2} W_3^{w_3}, \tag{2.8}$$

where $A^{(C_1)}_{w_1,w_2}$ and $A^{(C_2)}_{w_1,w_3}$ are the coefficients of the input-output weight enumerating functions of the constituent codes.

2.5 Turbo Decoding

Like other coding schemes, the optimal decoder is a Maximum Likelihood (ML) decoder. But unlike conventional convolutional codes, the Viterbi algorithm [26] and other algorithms based on the trellis diagram are not practical as turbo codes do not have a simple trellis diagram. However, the constituent codes are convolutional and they have such simple trellis diagrams.

The suboptimal decoder is an iterative, modular decoder. A turbo decoder consists of two concatenated decoders, each using the received systematic stream and the correspond-

ing received parity stream. Each decoder provides a soft output of the transmitted bits by using the received data and the information provided by the other decoder. The soft output is the a posteriori probability (APP) and consists of three components: the intrinsic information which is a function of the received signal for the corresponding bit position, the a priori (AP) probability of that bit position and the extrinsic information which comes from the received signal for other bit positions and their a priori probabilities. In each iteration, the extrinsic information produced by the other constituent decoder is used to evaluate the a priori probabilities in that iteration. Repeating this procedure improves the estimation of the bit probability values and hence, reduces the probability of error.

One efficient algorithm for soft output decoding, based on the trellis diagram of the code known as the BCJR algorithm, is presented by Bahl *et al.* in [27]. Another efficient soft decoding algorithm is derived from the coset decomposition principle in [28]. Also, there are some special methods for soft decoding such as sectionalized trellis diagrams [29] and the use of the codewords of the dual code [30].

The suboptimal decoder introduced in [1] finds the extrinsic information on the transmitted bits by one of the constituent decoders and passes it to the other decoder through the interleaver. The decoder can decode the received vector only if the iterative decoding converges. Note that in the iterative decoding, it is assumed that the extrinsic data provided by the first constituent decoder is independent from the received vector corresponding to the second parity stream and the systematic stream and vice versa. Although this is not true for all bit positions, it is generally true for most bit positions.

References [31–33] introduce extrinsic information transfer (EXIT) chart to find the convergence criteria for turbo decoding. In this approach, the extrinsic information from constituent maximum a posteriori (MAP) decoders are assumed to be Gaussian random variables when the inputs to the decoders are Gaussian. Furthermore, it is assumed that after interleaving, the adjacent bits have independent extrinsic information as they have been far enough before interleaving. Under these assumptions, the iterative decoder converges to zero probability of error if the signal to noise ratio is higher than a certain threshold [32, 33]. This threshold predicts the SNR of the waterfall in the performance of the iterative decoder. The minimum SNR for which the iterative decoder converges depends on the constituent codes and hence, one can improve the performance of the iterative decoder by proper selection of the component codes by using the EXIT chart.

2.6 Low-Weight Codewords and Minimum Hamming Distance

The performance of turbo codes in the error floor region is determined by the low-weight codewords. It is known that using a randomly chosen interleaver guarantees an excellent BER performance, but a certain number of low-weight codewords are generated, resulting in the appearance of an error floor and a small minimum distance. For a parallel concatenated turbo code, a low-weight systematic stream which produces low-weight parity streams in both RCC encoders results in a low-weight codeword, and hence a low minimum distance.

Despite this fact, the power of turbo codes is in part due to the low number of such codewords in comparison to a conventional convolutional code.

The structure and the number of low-weight codewords are studied in [12, 13]. In [12], it is shown that for the turbo code shown in figure 2.1 and for $N \rightarrow \infty$, only certain low-weight codeword structures remain asymptotically probable. These codewords consist of a low weight systematic stream which produces one or more short error events in each parity stream. The number of these short error events in the two parity streams are the same. Furthermore, each short error event is caused by a systematic stream of weight two.

By using a combinatorial approach, an upper bound on the minimum distance of turbo codes as a function of the code rate, interleaver length and the structure of the constituent codes is derived and it is proved that the minimum Hamming distance of the turbo codes cannot asymptotically grow at a rate higher than the logarithm of the codeword length [34]. A method to design the interleaver is presented in [35] which achieves a minimum distance increasing with the logarithm of the block length.

Reference [36] introduces a systematic technique to find sequences which are primary candidates for obtaining the minimum distance of parallel concatenated turbo codes. This technique finds all the input sequences that are mapped to shifted versions of themselves. These streams satisfy the conditions in [12] to form an asymptotically possible low-weight codeword.

The algorithm presented in [37] is applied to calculate the minimum distance of the turbo codes. This algorithm is improved in [38] by using a tighter lower bound on the

minimum distance. The effect of the interleaver structure on the minimum distance of the code is studied in [39].

Reference [40] shows that for low density parity check (LDPC) code ensembles (which are closely related to turbo codes), the capacity achieving codes do not have a large minimum distance. Battail in [6] shows that an appropriate criterion for the design of long block codes is the closeness of the normalized weight distribution of the code to a Gaussian distribution. Biglieri [41] substantiates this by showing that iterated-product codes have a weight distribution that is approximately Gaussian. In [42], it is shown that for codes with rates approaching one, the weight distribution is asymptotically Gaussian as the block length increases. Reference [42] also shows that for codes with lower code rates, the cumulative weight distribution asymptotically tends to a Gaussian cumulative distribution (as the block length increases) when the minimum distance of the dual code tends to infinity. It provides a sufficient condition on the systematic parity-check matrix of the code in order to have a Gaussian distribution. This condition is rather restrictive and it cannot be applied to the turbo code structure shown in figure 2.1. This sufficient condition is satisfied by a special class of multi-component block codes based on a so-called parallelotope interleaver introduced in [42].

2.7 Improving the Performance of Turbo Codes

Although the asymptotic performance of turbo codes in the waterfall region is very close to the theoretical limit for coding rates of practical interest (low), their performance in the error floor region can be improved by optimizing the component codes and the interleaver structure.

By choosing a proper interleaver, one can increase the minimum distance of the code and/or reduce the number of low-weight codewords. The chosen interleaver also affects the weight distribution for high-weight codewords which affects the performance in the waterfall region for short turbo codes. The effect of the chosen interleaver on the weight distribution is studied in [13, 35, 43–52]. These references provide some methods to design interleavers in order to decrease the number of low-weight codewords and/or to increase their weight. These methods are more beneficial when the block length is relatively small. Reference [53] studies the design of nonsystematic turbo codes to achieve higher minimum distances.

In [54], a concatenation of a turbo code and a Reed-Solomon code, and in [55], a concatenation of a turbo code and a BCH code are deployed to improve the error floor performance.

References [56, 57] provide methods to design prunable interleavers. With these techniques, smaller interleavers are produced by pruning a larger interleaver, while maintaining the good performance of the original code.

The algorithm in [17] expurgates some low-weight codewords by injecting a zero in the

lower-protected bit positions, and then punctures the resulting code to compensate for the loss in the effective code rate.

In [58], the extrinsic information in the decoder is modified to exploit the source redundancy to enhance the system performance.

In [15], it is indicated that using more component codes improves the distance properties of the turbo codes, resulting in a better performance when ML decoding is used. However, the suboptimal iterative decoding does not perform very well for multiple component codes. In [16], it is shown that the bit and frame error rates for both serial and parallel concatenation with uniform interleaving under some mild conditions approaches zero, at least as fast as $N^{-\beta}$ where N is the block length and β is the interleaver gain. For the parallel concatenated turbo codes, β is $J-2$ and $J-1$ for the bit and frame error probabilities, respectively, where J is the number of component codes.

2.7.1 Bounds on the Performance of Turbo Codes

Exact performance evaluation of block codes is often infeasible. Several bounding techniques are proposed to find a tight upper bound on the error probability of block codes. Gallager bounding techniques provide some upper bounds on the performance of linear block codes based on their weight distribution. Fano [59] also used the same general bounding method as Gallager's first bounding technique (GFBT), and therefore, some authors refer to the GFBT as the Gallager-Fano bounding method [60].

In Gallager's first bounding technique (GFBT), the word (frame) error probability is

decomposed as in

$$\begin{aligned} P\{E\} &= P\{E, \mathbf{r} \in \mathfrak{R}\} + P\{E, \mathbf{r} \notin \mathfrak{R}\} \\ &= P\{E, \mathbf{r} \in \mathfrak{R}\} + P\{E|\mathbf{r} \notin \mathfrak{R}\}P\{\mathbf{r} \notin \mathfrak{R}\} \\ &\leq P\{E, \mathbf{r} \in \mathfrak{R}\} + P\{\mathbf{r} \notin \mathfrak{R}\}, \end{aligned} \tag{2.9}$$

where E is the frame error event, $\mathbf{r}$ is the received signal vector and $\mathfrak{R}$ is an appropriate region in R^n around the transmitted signal point. The above expression divides the total error probability into the sum of error probability in a region of few and a region of many errors, denoted by $\mathfrak{R}$ and $\mathfrak{R}^c$, respectively. The region of many errors is considered totally erroneous, and only the error events in the region of few errors are estimated or bounded. The choice of region $\mathfrak{R}$ is very important in this bounding method. Different choices of this region have resulted in various different tight bounds in different ranges of signal-to-noise ratio. Here, we briefly review some important bounds and bounding techniques based on the GFBT.

For the BPSK signalling, all codewords have the same energy nE_N, where n is the number of bit positions and E_N is the energy per channel use. In this scheme and other signalling schemes whose codewords have equal energy, the codewords constellation is located on the surface of a hyper-sphere centered at origin. The tangential bound (TB) of Berlekamp [61] results in a significantly tighter bound than the union bound in low SNRs. This bound uses Gallager's first bounding technique combined with union bound for sphere constellations. In this bounding technique, the radial and the tangential components of the Gaussian noise are separated with a half-space shown in figure 2.4 as the underlying

Gallager region. The location of this half-space Gallager region, $\mathfrak{R}$, is determined by the radial component of the noise. The Gallager region is defined as

$$\mathfrak{R} = \{\mathbf{r} | z < z_0\}. \tag{2.10}$$

If the transmitted signal point is $\mathbf{s}_0$, the radial component of noise is the noise component in the direction of the axis connecting $\mathbf{s}_0$ to the origin, referred to as the Z axis. In order to tighten the bound, z_0 should be optimized.

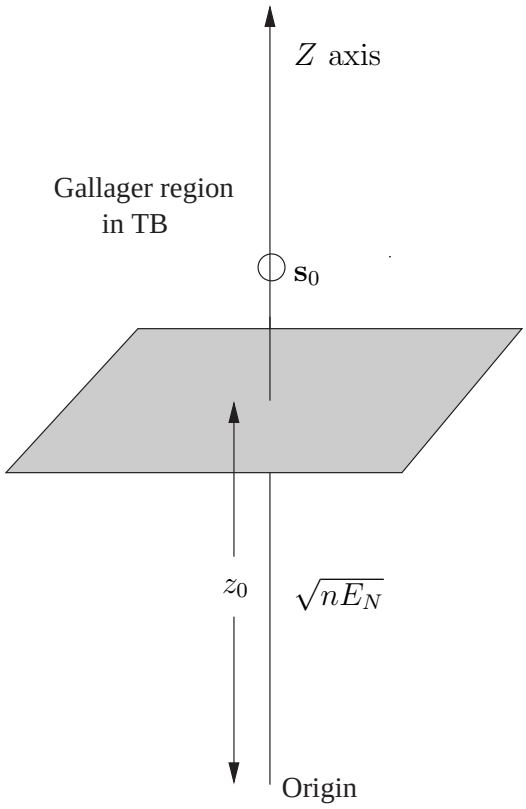

Figure 2.4: Gallager region in tangential boundB.

The Gallager region for the sphere bound (SB) of Herzberg and Poltyrev [62] is a sphere centered at the transmitted signal point as shown in figure 2.5, whose radius r is to be

optimized to tighten the bound, i.e.,

$$\mathfrak{R} = \{\mathbf{r} \mid \|\mathbf{r} - \mathbf{s}_0\| \leq r\}. \tag{2.11}$$

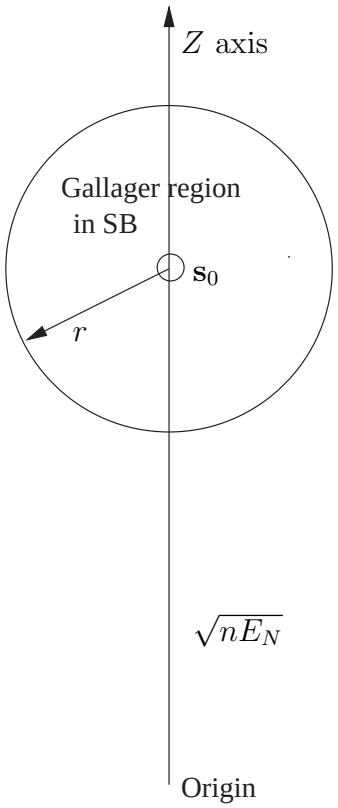

Figure 2.5: Gallager region in sphere bound.

The tangential sphere bound (TSB) is proposed by Poltyrev [63]. In TSB, the Gallager region $\mathfrak{R}$ is a hyper-cone whose apex is at the origin of the space and its main axis (referred to as the Z axis) is along the radial component of the noise as shown in figure 2.6 and

$$\mathfrak{R} = \{\mathbf{r} \mid r < z \tan\theta\}, \tag{2.12}$$

where $r = \sqrt{\|\mathbf{r}\|^2 - z^2}$ defines the boundary of the hyper-cone described above as a function

of its main axis variable. To tighten the bound, one should optimize the angle θ. This optimization is only a function of the weight distribution and does not depend on the noise variance. Reference [64] proves that the tangential bound is at least as tight as the union bound and is not tighter than the tangential sphere bound of Poltyrev. Reference [65] shows that the hyper cone used in the tangential sphere bound of Poltyrev for sphere codes is the optimum Gallager region for the Gallager's first bounding technique.

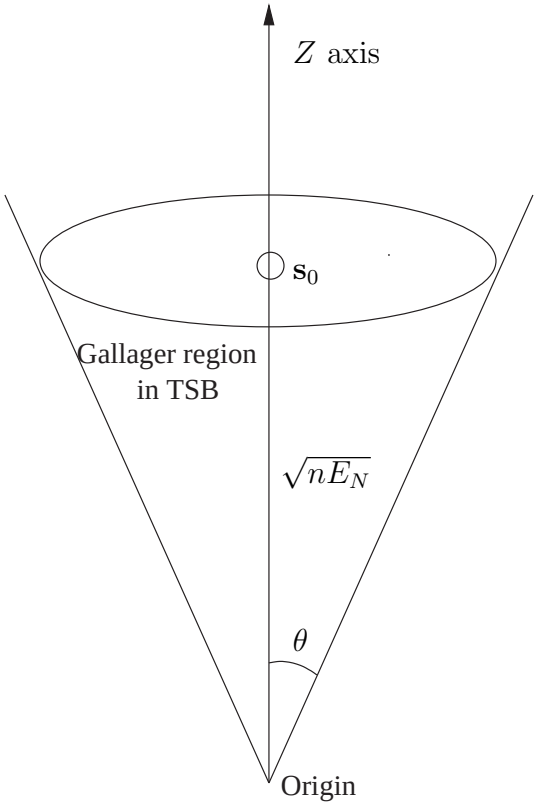

Figure 2.6: Gallager region in tangential sphere bound.

In [9], modified Gallager bounding technique is applied to some short block turbo codes to improve the union bound on the error performance for SNR values below the cutoff rate.

In [10, 11], the TSB is applied to short turbo codes to evaluate an upper bound which is tighter than the bound presented in [9].

2.8 Summary

In this chapter, the basic structure of turbo codes has been presented and an overview of the performance of turbo codes is provided. The literature on the asymptotic performance of turbo codes is reviewed. The weight distribution of the code based on the uniform interleaving is defined. The performance of the code and different solutions on how to improve it are provided. Finally, some bounds on the performance of turbo codes are reviewed.

Chapter 3

Performance Analysis in the Waterfall Region

3.1 Chapter Overview

In this chapter, the asymptotic weight distribution of turbo codes for high-weight codewords is studied and it is shown that the weight distribution is asymptotically Gaussian and its mean and variance are independent of the chosen interleaver. On the other hand, with a randomly chosen interleaver, its variance is equal to the best possible interleaver with probability one. As a result, interleaver optimization has little effect on the performance of the turbo codes in the waterfall region.

Based on the Gaussian distribution, the TSB is applied to the code and the achievable rate predicted by the TSB is compared to the capacity of BPSK signalling over an AWGN

channel.

3.2 Weight Distribution for Typical Weights

Consider the turbo code shown in figure 2.1 with N information bits. The joint probability distribution function of the systematic weight w_1 and each of the two parity weights $w_2,\ w_3$ is affected by the chosen recursive convolutional code and is not a function of the chosen interleaver. On the other hand, interleaver optimization can affect the conditional weight distributions of w_2 and w_3, when the other weight is known.

3.2.1 Probabilistic Properties of RCCs

It is assumed that the RCCs are generated by the transfer function namely $G(d) = N(d)/D(d)$. The impulse response of $G(d)$ is periodic with the period $P \leq 2^r - 1$, where r is the memory length of the code [23]. The main interest is in the group structure of the codebook, and also the periodicity property of the impulse response of $G(d)$. In this respect, we limit our attention to the structure of $D(d)$. This does not result in any loss of generality, because the group structure and also the periodicity property of the impulse response of $G(d)$ is not affected by the choice of $N(d)$.

In general, the desire is that the period of the impulse response of $G(d)$ is as large as possible. As mentioned earlier, the maximum period with r memory elements is equal to $2^r - 1$ for MLS sequences. For the rest of the paper, we assume that all the RCCs are

MLS. The rules to determine all the possible configurations of $D(d)$ to obtain a maximum length sequence of period $2^r - 1$ (for the given r) are provided in [23]. It can be shown that any MLS-sequence satisfies the three postulates of randomness [23]. One consequence of this property is that in any period of an MLS-sequence, the number of ones is equal to 2^{r-1}, and the number of zeros is $2^{r-1} - 1$.

If the impulse response of $D(d)$ is considered to be a periodic sequence (started at infinity in the past), we obtain $P = 2^r - 1$ non-zero sequences which are time shifts of each other. Each sequence corresponds to a specific positioning of the impulse within the period. These sequences are referred to as different phases of the periodic signal. We assume that the different phases are labeled by integer numbers, say $1, \ldots, P$, where the label of a phase corresponds to the relative position of the corresponding impulse within the period. It can be shown that the set of phases of an MLS-sequence (plus the all-zero sequence) constitutes a group under binary addition [23]. The order of each element in this group is equal to two, indicating that the sum of each phase with itself results in the all-zero sequence (denoted as the zero phase).

Using the group property of phases, we conclude that the function of the numerator of $G(d)$ is to replace each phase with a linear combination of some other phases. This function is equivalent to a permutation (relabeling) of phases and does not play a role in the following discussions.

For the bit position k, $(k = 1, \ldots, N)$ within the i'th output stream, we refer to the set of systematic bit positions $j \leq k$ for which an impulse at position j results in a 1 at

position k as $\mathcal{R}_i(k)$, $i = 1, 2, 3$.

For the systematic stream, it is easy to see that $\mathcal{R}_1(k) = \{k\}$. For the parity streams, if the bit position k is located in the L'th period, i.e., $L = \lceil k/P \rceil$, where $\lceil \, . \, \rceil$ denotes the ceiling function, then the number of positions belonging to $\mathcal{R}_i(k)$, $i = 2, 3$, within each of the periods $1, \ldots, L-1$ is equal to 2^{r-1} [23]. The number of positions within the L'th period (the period containing k itself) depends on the relative position of k within the L'th period and also on the numerator of $G(d)$.

We are mainly interested in the large values of L (parity bits far from the boundaries) for which the effect of the elements within the L'th period itself is negligible. Thus, $|\mathcal{R}_2(k)| = |\mathcal{R}_3(k)| \simeq \lceil k/P \rceil 2^{r-1}$, where $|\, . \,|$ denotes the cardinality of the corresponding set.

The notation $b_i(k)$, $i = 1, 2, 3$, $k = 1, \ldots, N$, is used to refer to the k'th bit within the i'th output stream. Since each bit is zero or one with an equal probability, then

$$\overline{b_i(k)} = \overline{b_i^2(k)} = 1/2. \tag{3.1}$$

3.2.2 Asymptotic Weight Distribution

To investigate the asymptotic weight distribution of turbo codes, we show that

$$\hat{w}_i = \frac{w_i}{\sqrt{N}}, \; i = 1, 2, 3, \tag{3.2}$$

referred to as the normalized weights, have a Gaussian distribution for their typical values when N is large. On the other hand, it is shown that the conditional weight distributions

are Gaussian. As a result, the three weights are jointly Gaussian distributed random variables.

It is easy to verify that each weight has a Gaussian distribution. All the 2^N possible combinations within the three steams are equiprobable, and consequently, the positions within each of the three output streams are independent and identically distributed (iid) binary random variables (where zero and one are equally probable). Using the Central Limit Theorem, we conclude that $\hat{w}_1$, $\hat{w}_2$ and $\hat{w}_3$, which are the normalized sum of N iid random variables, have a Gaussian distribution with mean $\sqrt{N}/2$ and variance $1/4$ for the large values of N.

In order to have a set of jointly Gaussian random variables, not only do the marginal weight distributions need to be Gaussian, but also the conditional distributions should be Gaussian. When the systematic weight w_1 is known, the parity bits are no longer independent of each other, because only $\binom{N}{w_1}$ out of 2^N codewords represent a systematic weight of w_1, and hence, remain probable. Under these circumstances, the parity bits in each stream tend to be an *m-dependent* sequence and the Central Limit Theorem can still be applied. In the following, using the properties of m-dependent random variables, we show that the conditional weight distributions of $\hat{w}_2$ and $\hat{w}_3$ given $\hat{w}_1$ are Gaussian for the typical values of $\hat{w}_1$. As a result, noting that the marginal distributions are Gaussian, we can conclude that $\hat{w}_1$, $\hat{w}_2$ and $\hat{w}_3$ are a set of jointly Gaussian random variables.

Definition: *m-dependent sequence* [66]

A sequence $X_1, X_2, \ldots$ of random variables is called m-dependent if and only if two subse-

quences $\{X_{a-r}, X_{a-r+1}, \ldots, X_a\}$ and $\{X_b, X_{b+1}, \ldots, X_{b+s}\}$ are independent sets of variables when $b - a > m$; that is, an m-dependent sequence is a sequence of dependent random variables for which the dependency lasts, at most, for m elements.

Theorem 3.1. Central Limit Theorem for the sum of dependent random variables *[66] If $X_1, X_2, \ldots$ is a sequence of m-dependent, uniformly bounded random variables and $S_n = X_1 + X_2 + \cdots + X_N$, with the standard deviation V_N. Then, if $\frac{V_N}{N^{1/3}} \to \infty$ as $N \to \infty$, $\overline{G}_N(x) \to \Phi(x)$ for all x, as $N \to \infty$, where $\overline{G}_N$ is the cumulative distribution function (cdf) of $\frac{S_N - E(S_N)}{V_N}$ and $\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} \exp(-\frac{t^2}{2})dt$.*

As indicated by the theorem, if the standard deviation of the sum of N consecutive elements of a stream of m-dependent random variables grows faster than the third root of N, the Central Limit Theorem can still be applied. In order to apply this theorem on the conditional weight distributions, we prove the following proposition.

Proposition 3.1. *Given that the systematic weight is w_1, each parity stream is an m-dependent sequence, and the variance of its weight is given by*

$$\sigma^2_{w_2|w_1} = \frac{N}{4}\left(1 + \frac{2(1 - \frac{2w_1}{N})^{(P+1)/2}}{1 - (1 - \frac{2w_1}{N})^{(P+1)/2}}\right). \tag{3.3}$$

To prove the proposition, we need the following lemma.

Lemma 3.1. *Suppose that we partition a stream of N bits consisting of w ones and $N - w$ zeros into K groups. Each group consists of N_k, $k = 1, \ldots, K$, $\sum_k N_k = N$ bits. We denote by O_k, the event in which the k'th group has an odd Hamming weight. For $N \to \infty$,*

if

$$\lim_{N\to\infty} \frac{N_k}{N} \neq 0,\ k = 1, \ldots, K, \tag{3.4}$$

then O_1, O_2, ..., O_{K-1} *tend to be independent events with probability* $1/2$ *as* N *goes to infinity (for the typical values of* w*).*

Proof. The Hamming weight of the k'th group is shown by W_k. Then, the probability mass function of W_k can be written as

$$P_{W_k}(w_k) = \frac{\binom{N-N_k}{w-w_k}\binom{N_k}{w_k}}{\binom{N}{w}},\ w_k = 0, 1, \ldots, N_k. \tag{3.5}$$

This probability mass function is an increasing function with respect to w_k for $0 < w_k < w_t$, where $w_t = \left\lfloor \frac{wN_k}{N} \right\rfloor$ is the typical value for the Hamming weight of the k'th subsequence, and is decreasing for $w_t < w_k < \min\{w, N_k\}$.

An integer random variable with a monotonic probability mass function is almost equally likely to be an even or an odd number. In fact, the difference between the two probabilities is less than the boundary probabilities. For example, suppose that X is a random variable with a monotonically increasing probability mass function defined for $2a < x < 2b,\ x, a, b \in \mathbb{Z}$. Then,

$$\begin{aligned} P\{X \text{ is even}\} &= \sum_{x=a}^{b} P\{X = 2x\} = \sum_{x=a}^{b-1} P\{X = 2x\} + P\{X = 2b\} \\ &\leq \sum_{x=a}^{b-1} P\{X = 2x+1\} + P\{X = 2b\} = P\{X \text{ is odd}\} + P\{X = 2b\}. \end{aligned} \tag{3.6}$$

The probability mass function that is described by (3.5) can be separated into two monotonic (one increasing and one decreasing) functions. For $N \to \infty$, the boundary

probabilities specified by (3.5) (i.e., the probabilities at $w = 0, N, w_t$) are 0, and so,

$$P\{W_k \text{ is odd}\} = P\{W_k \text{ is even}\} = \frac{1}{2}. \tag{3.7}$$

The same approach is valid for the k'th group ($k < K$) when the Hamming weight of the first $k-1$ groups are known, and hence, it is odd-weighted with probability 1/2. Obviously, the Hamming weight of the K'th group, given the Hamming weights of the other groups, is known. □

We are now ready to prove Proposition 3.1. Assuming the systematic weight is w_1, we show that each parity stream is an m-dependent sequence, and the variance of its weight is given by (3.3).

Proof. Consider two arbitrary parity bits (far from the boundaries) named pb_1 and pb_2 in a given parity stream. We show that these two bits are independent of each other, when the distance between them is large. The proof can be easily extended to two sets of parity bits. According to the distance between pb_1 and pb_2, two situations can occur.

Case I: The distance between these parity bits is not an integer multiple of the RCC impulse response period P. We divide the information bits into four subsets, depending on whether they trigger these two parity bits or not. We denote these four groups by $C_k, k = 0, 1, 2, 3$. C_0 is the set of systematic bits which do not trigger non of the parity bits. C_1 and C_2 are defined as the set of the systematic bits which trigger only the first parity bit and the second parity bit, respectively. Finally, C_3 consists of bits that trigger both parity bits. Similarly, we denote by O_i, the event that $C_i,\ i = 0, 1, 2, 3$ has an odd

weight. Systematic bits located after both parity bit position do not affect them and hence, they belong to set C_0. For any P information bits preceding the first parity bit, there is at least one bit in each of $C_i,\ i = 1, 2, 3$. Hence,

$$\frac{|C_i|}{N} \neq 0,\ i = 0, 1, 2, 3, \tag{3.8}$$

where $|\cdot|$ denotes the cardinality of a set. As a result, C_i's satisfy the conditions in Lemma 3.1. It is easy to see that

$$pb_1 = O_1 \oplus O_3, \quad pb_2 = O_2 \oplus O_3, \tag{3.9}$$

in which $\oplus$ is the binary addition (pb_1 is one if only one of O_1 and O_3 happens, and is zero, otherwise.) Since, O_1, O_2 and O_3 are equiprobable identical independent events, pb_1 and pb_2 are equiprobable independent bits.

Case II: The distance between the two parity bits is an integer multiple of impulse response period P, say kP. In this case, C_1 is empty, but C_0 and C_3 still satisfy the condition in the lemma 3.1. C_2 has only $k(P+1)/2$ elements since in each period P, only $(P+1)/2$ bits trigger a certain parity bit. However, as long as the distance between the two parity bits is large (when k is large which is true for almost any two typical bits), the conditions of the Lemma 3.1 are satisfied, and O_2 and O_3 become equiprobable independent and identically distributed events. As a result pb_1 and pb_2 are independent. Note that the dependency between parity bits last longer when the systematic weight is far from its typical values (around $N/2$) and as a result m will be larger.

To apply the Central Limit Theorem to the m-dependent sequence of the parity stream,

we have to find the variance of the conditional parity weight. This variance is a function of the cross correlation between the near parity bits that are separated by an integer multiple of P (all the other parity bit pairs are uncorrelated). To compute this correlation, we note that when the distance between the parity bits is kP (k is a relatively small integer), the elements of C_2 can be considered to be iid bits, and each of them is one with probability $\frac{w_1}{N}$. Then,

$$\text{cov}\left[b_2(i), b_2(i+kP)\right] = \frac{1}{4}\left(1 - \frac{2w_1}{N}\right)^{k(P+1)/2}, \tag{3.10}$$

because the probability of having an odd parity within these $k(P+1)/2$ bits is

$$P\{O_2 = 1\} = \frac{1 - \left(1 - \frac{2w_1}{N}\right)^{k(P+1)/2}}{2}. \tag{3.11}$$

The covariances of the other pairs are zero. Since, the parity weight is $w_2 = \sum_{i=0}^{N} b_2(i)$, then

$$\sigma^2_{w_2|w_1} = \sum_{i=1}^{N} \sigma^2_{b_2(i)} + 2 \sum_{1 \le i < j \le N} \text{cov}[b_2(i), b_2(j)]. \tag{3.12}$$

As a result,

$$\begin{aligned}
\sigma^2_{w_2|w_1} &= \sum_{i=1}^{N} \frac{1}{4} + 2\sum_{i=1}^{N} \sum_{k=1}^{\lfloor (N-i)/P \rfloor} \text{cov}[b_2(i), b_2(i+kP)] \\
&= \frac{N}{4} + 2\sum_{i=1}^{N} \sum_{k=1}^{\lfloor (N-i)/P \rfloor} \frac{1}{4}\left(1 - \frac{2w_1}{N}\right)^{k(P+1)/2} \\
&\simeq \frac{N}{4} + 2\sum_{i=1}^{N} \sum_{k=1}^{\infty} \frac{1}{4}\left(1 - \frac{2w_1}{N}\right)^{k(P+1)/2} \\
&= \frac{N}{4}\left(1 + \frac{2(1 - \frac{2w_1}{N})^{(P+1)/2}}{1 - (1 - \frac{2w_1}{N})^{(P+1)/2}}\right).
\end{aligned} \tag{3.13}$$

□

With this proposition and Theorem 3.1, the conditional parity weight distributions given the normalized systematic weight $\hat{w}_1$, asymptotically become Gaussian. A similar approach is valid for the conditional weight distribution of $\hat{w}_3$, given $\hat{w}_1$ and $\hat{w}_2$. As a result, $\hat{w}_1$, $\hat{w}_2$ and $\hat{w}_3$ are a set of jointly Gaussian random variables, since their marginal and conditional distributions are Gaussian.

A set of jointly Gaussian random variables can be completely described by their mean vector and covariance matrix. The mean and the marginal variance of $\hat{w}_1$, $\hat{w}_2$ and $\hat{w}_3$ are $\sqrt{N}/2$ and $1/4$, respectively. The correlation coefficients between $\hat{w}_i$ and $\hat{w}_j$ denoted by ρ_{ij}, $i, j = 1, 2, 3$, can be written as

$$\rho_{ij} = \frac{\overline{\hat{w}_i \hat{w}_j} - \overline{\hat{w}_i}\ \overline{\hat{w}_j}}{\sigma_{\hat{w}_i} \sigma_{\hat{w}_j}} = 4\left[\overline{\hat{w}_i \hat{w}_j} - \frac{N}{4}\right], \tag{3.14}$$

and

$$\overline{\hat{w}_i \hat{w}_j} = \frac{1}{N} \sum_m \sum_n \overline{b_i(m) b_j(n)}, \tag{3.15}$$

where the expectation is taken over all 2^N possible input combinations. The total normalized weight of the output sequence is equal to $\hat{w} = \hat{w}_1 + \hat{w}_2 + \hat{w}_3$ which has a Gaussian distribution with mean,

$$\mu_{\hat{w}} = 3\frac{\sqrt{N}}{2}, \tag{3.16}$$

and variance,

$$\sigma^2_{\hat{w}} = \frac{3 + 2\rho_{12} + 2\rho_{13} + 2\rho_{23}}{4}. \tag{3.17}$$

Noting that sequences with a smaller weight result in higher probabilities of error, we conclude that the main objective in the code design (as far as the waterfall region is

concerned) is to sharpen the peak of the pdf of the normalized Hamming weight $\hat{w}$ which is equivalent to minimizing the variance of the normalized weight. This is equivalent to minimizing the correlation coefficients ρ_{ij}. In the following, we first show that $\rho_{ij} \geq 0$; therefore, the minimum value for the correlation coefficient is zero. When the block length increases, $\rho_{1j},\ j = 2, 3$ become zero for any nontrivial RCC. Also, ρ_{23} tends to zero with probability one for a randomly chosen interleaver. Consequently, the asymptotic weight distribution by using a randomly chosen interleaver is optimum (in the waterfall region) with probability one.

In the following, we first show that the $\rho_{ij} \geq 0$; therefore, the minimum possible value for the correlation coefficients is zero.

Theorem 3.2. $\rho_{ij} \geq 0$ for $i, j = 1, 2, 3$.

Proof. Any of the pairs $b_i(m)$, $b_j(n)$ for $i, j = 1, 2, 3$ and $m, n = 1, \ldots, N$, can take four different values, $\{00, 01, 10, 11\}$. The set of the input sequences that result in the value of 00 form a sub-group of all the possible 2^N input combinations. This is a direct consequence of the linearity and the group property of the code. Due to the group property of the set of corresponding coset leaders, two situations can occur. There is either only one coset with the coset leader 11, or there are three cosets with the coset leaders 01, 10 and 11. The important point is that in both of these cases, the 00 sub-group and its cosets contain the same number of input sequences. Therefore, for the probability of the pair $b_i(m)$, $b_j(n)$, the following two cases exist:

Case I: $b_i(m), b_j(n)$ take the values $00, 11$, each with probability $1/2$, resulting in $\overline{b_i(m)b_j(n)} = 1/2$, so that

$$\overline{b_i(m)b_j(n)} - \overline{b_i(m)}\ \overline{b_j(n)} = \frac{1}{4}. \tag{3.18}$$

Case II: $b_i(m), b_j(n)$ take the values $00, 01, 10, 11$, each with probability $1/4$, resulting in $\overline{b_i(m)b_j(n)} = 1/4$, so that

$$\overline{b_i(m)b_j(n)} - \overline{b_i(m)}\ \overline{b_j(n)} = 0. \tag{3.19}$$

In both cases, we have

$$\overline{b_i(m)b_j(n)} - \overline{b_i(m)}\ \overline{b_j(n)} \geq 0. \tag{3.20}$$

This indicates that the correlation coefficients ρ_{ij}, $i, j = 1, 2, 3$ are always nonnegative. □

The following theorems show that when the block length increases, $\rho_{1j},\ j = 2, 3$ become zero for any nontrivial RCC. Also, ρ_{23} tends to zero with the probability of one for the randomly chosen interleavers. Consequently, the asymptotic weight distribution for the high-weight codewords with the probability of one is optimized when a randomly chosen interleaver is used.

Theorem 3.3. $\rho_{12}, \rho_{13} \to 0$ as $N \to \infty$.

Proof. For ρ_{12} and ρ_{13} (the interaction of the systematic stream with each of the parity checks), Case II in the previous two cases is valid, resulting in $\rho_{12}, \rho_{13} \to 0$ as $N \to \infty$. Note that $b_1(m)$ and $b_2(n)$ are independent of each other, if $b_1(m)$ is not mapped (through

interleaving) to a bit position within $\mathcal{R}_2(n)$, or if $\mathcal{R}_2(n)$ contains at least two elements. This is valid except for some trivial cases which have a vanishing effect on the overall result. □

Theorem 3.4. $\rho_{23} \to 0$ for $N \to \infty$ with the probability of one (for almost any random interleaver).

Proof. If $\mathcal{R}_2(m)$ differs from $\mathcal{R}_3(n)$, even by one bit position, then $b_2(m)$ and $b_3(n)$ are independent of each other. This results in $\overline{b_2(m)b_3(n)} = \overline{b_2(m)}\ \overline{b_3(n)} = 1/4$. This is the case, unless $|m - n| < P/2$, and the elements of $\mathcal{R}_2(m)$ and $\mathcal{R}_3(n)$ contain the same input bits (before and after interleaving). Consequently, the corresponding interleaver has a restriction on the mapping of the many bit positions. Obviously, the fraction of such interleavers tends to zero as $N \to \infty$. Therefore, for almost any random interleaver, $\rho_{23} \to 0$ as $N \to \infty$. □

As a result, the typical weight distribution of turbo codes is not a function of the chosen RCC and interleaver (for nontrivial RCCs and interleavers), and hence, the interleaver optimization has a diminishing effect on the asymptotic performance of the turbo code in its waterfall region.

This result is valid only for the waterfall region and when a maximum likelihood decoder is used. With the iterative decoder, the chosen RCCs affect the EXIT chart [31] and hence the RCC optimization can slightly change the SNR region in which the waterfall happens. On the other hand, as it is shown in the next chapter, the interleaver and RCCs affect

the weight distribution for low-weight codewords, and hence, the performance in the error floor region can be improved by the RCC and interleaver optimization.

The Gaussian weight distribution approximation is valid for the typical values of the Hamming weight. The number of low-weight codewords cannot be approximated by a continuous distribution, and as we will see in the next chapter, low-weight codewords appear only in certain structures and for each of these structures, their number is a Poisson random variable.

3.3 Cutoff Rate for large block Turbo Codes

In this section, in order to provide insight into the range of the SNR for which codewords of typical weights are dominant, we apply the union bound on the weight distribution to find the dominant weight in the error performance. Also, the cutoff rate which is based on applying the union bound on the weight distribution is calculated according to this assumption and compared to the random coding cutoff rate.

The Gaussian approximation of the turbo code weight distribution is similar to the weight distribution of random codes. This assumption remains valid when high-weight codewords dominate the performance. One of the tools to characterize random coding is the cutoff rate. The weight of the dominant codewords in computing the cutoff rate provides insight into the validity of the Gaussian approximation. We compute the cutoff rate using the Gaussian distribution, and compare it to the random coding cutoff rate;

namely,

$$R_0 = 1 - \log_2(1 + e^{-E_N/N_0}), \tag{3.21}$$

where E_N is the channel symbol energy, and N_0 is the one-sided Gaussian power spectrum of noise [67].

For a turbo code of rate R and block length N, the normalized weight distribution function can be modeled as a Gaussian distribution with the mean

$$\mu_{\hat{w}} = \frac{\sqrt{N}}{2R} \tag{3.22}$$

and variance

$$\sigma^2_{\hat{w}} = \frac{1}{4R}, \tag{3.23}$$

where the code rate R is achieved by employing a larger number of parallel concatenated RCCs and/or puncturing which does not affect the Gaussian assumption. The number of codewords of the normalized weight between $\hat{w}$ and $\hat{w} + \Delta\hat{w}$, under the Gaussian distribution, is

$$N_{\hat{w}} \simeq \frac{2^N \Delta\hat{w}}{\sqrt{\frac{\pi}{2R}}} \exp\left[-2R\left(\hat{w} - \frac{\sqrt{N}}{2R}\right)^2\right]. \tag{3.24}$$

The term in the union bound that corresponds to the probability of an error event of the normalized weight $\hat{w}$ (using the BPSK modulation) is

$$p_{\hat{w}} = Q\left(\sqrt{\frac{2\hat{w}\sqrt{N}E_N}{N_0}}\right). \tag{3.25}$$

The dominant codewords in the error probability are around the peak of $N_{\hat{w}} p_{\hat{w}}$, which occurs at

$$\hat{w_p} = \frac{\sqrt{N}}{2R}\left(1 - \frac{E_N}{2N_0}\right). \tag{3.26}$$

The Gaussian assumption is valid when

$$\lim_{N\to\infty} \frac{R\hat{w}_p}{\sqrt{N}} \neq 0, 1. \tag{3.27}$$

It is easy to see that

$$\frac{R\hat{w}_p}{\sqrt{N}} < \frac{1}{2}, \tag{3.28}$$

and consequently, we only require that

$$\frac{R\hat{w}_p}{\sqrt{N}} > 0, \tag{3.29}$$

resulting in

$$\frac{E_N}{N_0} < 2 \tag{3.30}$$

(which is equivalent to 3 dB). After the break point of $E_N/N_0 = 3$ dB is reached, the behavior of the turbo code cannot be modeled anymore by using the Gaussian distribution.

In practice, turbo codes are used in much lower ranges of signal to noise ratios than the break point. For example, the value $\frac{E_N}{N_0} = 3$ dB corresponds to the value of $\frac{E_b}{N_0} = 7.7$ dB (E_b stands for energy per information bit) for a code of the rate 1/3, or to $\frac{E_b}{N_0} = 6$ dB for a code of the rate 1/2. These values are substantially higher than those of the ranges of $\frac{E_b}{N_0}$ used in practical systems. In other words, the dominant codewords follow the Gaussian assumption for the SNRs of interest.

To find the cutoff rate under the Gaussian assumption, using the union bound, we have

$$P_e < \sum_{\hat{w}=0}^{\frac{\sqrt{N}}{R}} N_{\hat{w}} p_{\hat{w}}. \tag{3.31}$$

By using the inequality $Q(x) < \frac{1}{2}\exp\left(-\frac{x^2}{2}\right)$ and the Gaussian assumption, (3.31) can be rewritten as

$$P_e < \frac{2^N}{\sqrt{\frac{2\pi}{R}}} A \int_0^{\frac{\sqrt{N}}{R}} \exp\left(-2R\left[\hat{w} - \frac{\sqrt{N}}{2R}(1 - \frac{E_N}{2N_0})\right]^2\right) d\hat{w}, \tag{3.32}$$

where

$$A = \exp\left(-\frac{N}{2R}\left[1 - \left(1 - \frac{E_N}{2N_0}\right)^2\right]\right), \tag{3.33}$$

and hence,

$$P_e < 2^{N-1} AB, \tag{3.34}$$

where

$$B = Q\left[\sqrt{\frac{N}{R}}\left(\frac{E_N}{2N_0} - 1\right)\right] - Q\left[\sqrt{\frac{N}{R}}\left(\frac{E_N}{2N_0} + 1\right)\right]. \tag{3.35}$$

For $\frac{E_N}{N_0} < 2$ and $N \to \infty$,

$$\lim_{N\to\infty} Q\left[\sqrt{\frac{N}{R}}\left(\frac{E_N}{2N_0} - 1\right)\right] = 1, \tag{3.36}$$

and,

$$\lim_{N\to\infty} Q\left[\sqrt{\frac{N}{R}}\left(\frac{E_N}{2N_0} + 1\right)\right] = 0. \tag{3.37}$$

Hence, B can be approximated as 1.

Let us define

$$R_T = \frac{1}{2\ln(2)}\left[\frac{E_N}{N_0} - \frac{1}{4}\left(\frac{E_N}{N_0}\right)^2\right]. \tag{3.38}$$

We can see that if $R < R_T$, then the probability of error converges to 0 as $N \to \infty$. Figure 3.1 reflects the difference between R_0 and R_T around the break point of $\frac{E_N}{N_0} = 3$ dB ($\frac{E_b}{N_0} = 7.7$ dB for a code of the rate 1/3).

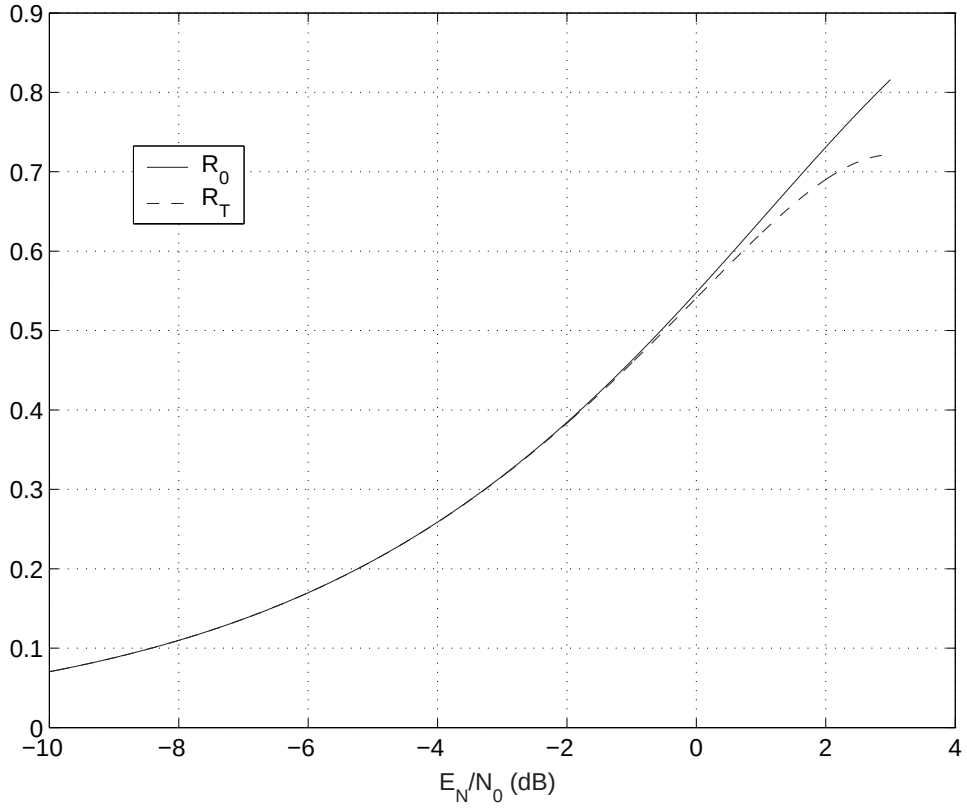

Figure 3.1: Comparison between R_0 and R_T versus $\frac{E_N}{N_0}$.

3.4 Tangential Sphere Bound on Average Spectrums

Gallager Bounding techniques [8] use weight enumerating function to give some upper bound on the error performance of linear codebooks. In [14], a very tight bound named tangential sphere bounding technique is presented.

Tangential sphere bound is very effective for binary codebooks over BPSK modulation. In these codebooks, all the codewords are located on the surface of a n-dimensional hyper-sphere of radius $\sqrt{nE_N}$ where n is the code length. The Voronoi region for these codebooks is a cone as all the median hyper-planes between codewords contain origin. In this technique, the Gallager region is a cone which mimics the Voronoi region. The cone's apex is at the origin and its main axes passes the corresponding codeword. In [65], it is shown that this Gallager region is optimum and it gives the tightest upper bound based on Gallager bounding technique.

Poltyrev shows that for codes with average spectrum, tangential sphere bounding provides an error exponent which is very close to that of capacity-achieving random coding schemes [14]. Average spectrum is defined as [14]

$$\overline{A}_w = \begin{cases} 2^{n(h(\omega)-h(\alpha))+o(n)} & w = \omega n \geq \alpha n \\ 0 & w = \omega n < \alpha n \end{cases}, \tag{3.39}$$

where $n = N/R$ is the code length and $0 < \alpha < \dfrac{1}{2}$ is the root to the following equation

$$R = 1 - h(\alpha) = 1 + \alpha \log_2(\alpha) + (1-\alpha)\log_2(1-\alpha). \tag{3.40}$$

The error exponent based on this bounding technique is

$$E_{ts}(R,\sigma^2) = \begin{cases} \dfrac{\alpha}{2\sigma^2}\log 2 & 0 < \sigma^2 \le \sigma_{ad}^2 \\ h(\alpha) - \log(1+e^{-1/2\sigma^2}) & \sigma_{ad}^2 < \sigma^2 \le \sigma_{st}^2 \\ \left(\dfrac{\eta_0^2}{2\sigma^2} + \dfrac{\rho_{\eta_0}}{2\sigma^2}\right)\log 2 - \dfrac{1}{2}\log\dfrac{e\rho_{\eta_0}}{\sigma^2} & \sigma_{ts}^2 < \sigma_{ad}^2 \le t_0 \end{cases}, \tag{3.41}$$

where

$$\rho_{\eta_0} = t_0(1-\eta_0)^2, \tag{3.42}$$

$$t_0 = \min_{\omega} \frac{\omega 2^{2r_\omega}}{(1-\omega)(2^{2r_\omega}-1)}, \tag{3.43}$$

$$r_\omega = \max\{0, (h(\omega) - h(\alpha))\}, \tag{3.44}$$

$$\eta_0 = \frac{1+2t_0 - \sqrt{1+4\sigma^2(1+t_0)}}{2(1+t_0)}, \tag{3.45}$$

$$\sigma_{ts}^2 = t_0(1-\omega_{ts})^2 - \omega_{ts}(1-\omega_{ts}), \tag{3.46}$$

and ω_{ts} is the root of the following equation:

$$2\omega(1-\omega)\log\frac{1-\omega}{\omega} = 2^{2(h(\omega)-h(\alpha))} - 1. \tag{3.47}$$

The Gaussian distribution, described by

$$A_\omega = \frac{2^{nR}}{\sqrt{\pi/2}}\exp\left[-2n\left(\omega - \frac{1}{2}\right)^2\right], \tag{3.48}$$

is slightly different from the average spectrum given by (3.39). The Gaussian weight distribution predicts a nonzero number of codewords in the region $0 < w < \alpha n$. Here, we apply the TSB on the error probability based on the Gaussian distribution to predict the performance of the code in the waterfall region where the dominant codewords are in the typical region.

The Gallager region in TSB is a cone whose apex is located at the origin and its axis denoted by the Z axis connects the origin to the all-zero codeword. We normalize the space by dividing each axes by $\sqrt{n}$. Note that with BPSK signalling, the all-zero codeword is located at $\left(\sqrt{\frac{E_N}{n}}, \sqrt{\frac{E_N}{n}}, \ldots, \sqrt{\frac{E_N}{n}}\right)$, where E_N is the energy per channel use. This cone is produced by rotating the line $r = z\tan\theta$ about the Z axis, where r is the distance to the Z axis in the polar coordinates. All codewords are on the surface of an n-dimensional sphere with radius $\sqrt{E_N}$. The Euclidean distance between two codewords in the normalized space is $2\sqrt{\frac{w_d E_N}{n}}$, where w_d is the Hamming distance between the two codewords.

To find the TSB error exponent on the Gaussian spectrum, we need the following lemma.

Lemma 3.2. *For large even integer n, if Y is a chi-squared random variable with mean n and n degrees of freedom, then for $y = \beta n > n$,*

$$P\{Y > y\} < \frac{n}{2}\frac{e^{-y/2}(y/2)^{n/2-1}}{(n/2)!} = O\left(\exp\left\{-\frac{n}{2}(\beta - 1 - \log\beta)\right\}\right). \tag{3.49}$$

Proof. For the chi-squared random variable Y with mean n and n degrees of freedom,

$$P\{Y > y\} = \frac{\Gamma\left(\frac{n}{2}, \frac{y}{2}\right)}{\Gamma\left(\frac{n}{2}\right)}, \tag{3.50}$$

where $\Gamma(\alpha, x)$ is the *incomplete gamma function* defined by

$$\Gamma(\alpha, x) = \int_x^\infty t^{\alpha-1}e^{-t}dt, \tag{3.51}$$

and the *gamma function*, $\Gamma(\alpha)$, is

$$\Gamma(\alpha) = \int_0^\infty t^{\alpha-1}e^{-t}dt. \tag{3.52}$$

For integer $\alpha = m$,

$$\Gamma(m) = (m-1)!. \tag{3.53}$$

and

$$\Gamma(m,x) = (m-1)!e^{-x}\sum_{k=0}^{m-1}\frac{x^k}{k!} = (m-1)!e^{-x}e_m(x), \tag{3.54}$$

where $e_m(x) = \sum_{k=0}^{m-1}\frac{x^k}{k!}$ is the *exponential sum function*. For $x > m$, $e_m(x)$ can be upper bounded by

$$e_m(x) = \sum_{k=0}^{m-1}\frac{x^k}{k!} < m\frac{x^{k_c}}{k_c!}, \tag{3.55}$$

where k_c is

$$k_c = \arg\max_{0\le k<m}\frac{x^k}{k!} = m-1. \tag{3.56}$$

and hence,

$$e_m(x) < m\frac{x^{m-1}}{(m-1)!}. \tag{3.57}$$

Then, by replacing $x = y/2$ and $m = n/2$, (3.50) is upper bounded by

$$P\{Y > y\} < \frac{n}{2}\frac{e^{-y/2}(y/2)^{n/2-1}}{(n/2)!}. \tag{3.58}$$

For odd n, we add an independent chi-squared random variable Y_1 with mean one and one degree of freedom to Y to form the random variable $Y' = Y + Y_1$ which will be a chi-squared random variable with mean $n+1$ and $n+1$ degrees of freedom. Since $Y_1 \ge 0$, for $y > n+1$,

$$P\{Y > y\} < P\{Y' = Y + Y_1 > y\} < \frac{n+1}{2}\frac{e^{-y/2}(y/2)^{(n+1)/2-1}}{((n+1)/2)!}. \tag{3.59}$$

For large even n and $y/n = \beta > 1$ and by using the Stirling's approximation,

$$P\{Y > \beta n\} < \frac{1}{2}\frac{e^{-\beta n/2}(\beta n/2)^{n/2-1}}{\sqrt{\pi n}(n/2e)^{n/2}} = \frac{n}{2}\frac{e^{-\beta n/2}(\beta e)^{n/2}}{\sqrt{\pi n}\beta/2} \quad \text{for } \beta > 1. \tag{3.60}$$

This indicates an exponent of $\frac{1}{2}(\beta - 1 - \log\beta)$ in the probability defined by (3.50). □

Theorem 3.5. *The probability of error for a code of rate R of length n and $N = nR$ information bits whose weight distribution is given by (3.48) approaches zero as $N \to \infty$, if*

$$E_2 = \min_{0<\omega<\frac{\sqrt{N_0/2}}{\sqrt{E_N}+\sqrt{N_0/2}}} \left\{ 2\left(\omega - \frac{1}{2}\right)^2 - \frac{1}{2}\log\left(1 - \frac{2\omega}{1-\omega}\frac{E_N}{N_0}\right)\right\} - R\log 2 > 0, \tag{3.61}$$

where E_N is the energy per channel use and N_0 is the one-sided noise spectrum.

Proof. Consider a thin disk of radius $c = \sqrt{E_N}\tan\theta$ and height $\epsilon \to 0$ around the all-zero codeword as shown in figure 3.2. Note that the surface of the disk is an $n-1$ dimensional sphere perpendicular to the Z axis. This disk is the portion of the cone which is confined by $\sqrt{E_N} - \frac{\epsilon}{2} < Z < \sqrt{E_N} + \frac{\epsilon}{2}$.

Each dimension is normalized by $\sqrt{n}$. Therefore, the noise component on each dimension is asymptotically zero. Hence, the probability that the received vector falls inside the disk given that the all-zero codeword is transmitted is equivalent to the probability that it falls inside the entire cone. In other words, the cone and the disk around the all-zero codeword are the same in $n-1$ dimensions and differ in only one dimension and the noise component along that dimension is zero with probability one.

If the thin disk is used as the Gallager region, assuming that the all-zero codeword is

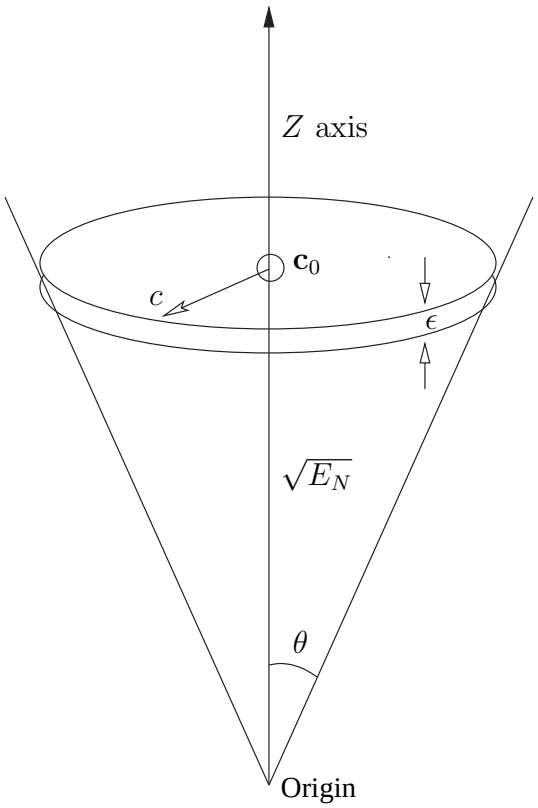

Figure 3.2: Gallager region used for TSB.

transmitted, the error probability is bounded by

$$P_e \leq P\{\mathbf{r} \notin \mathfrak{R}\} + \sum_{i \neq 0} P\{\mathbf{r} \in \mathfrak{R}, |\mathbf{r} - \mathbf{c}_i| < |\mathbf{r} - \mathbf{c}_0|\}, \tag{3.62}$$

where $\mathfrak{R}$ is the Gallager region (i.e. the thin disk), $\mathbf{c}_0$ is the all-zero codeword, $\mathbf{r} = \mathbf{c}_0 + \mathbf{n}$ is the received vector ($\mathbf{n}$ is the noise vector) and the summation is over all nonzero codewords $\mathbf{c}_i$ whose median planes with the all-zero codeword intersect with the Gallager region. In the following, we find an upper bound for each summand in the error probability bound in (3.62) and its associated error exponent. The probability of error converges to zero as $N \to \infty$ if all error exponents are positive.

The probability that the received vector is outside the Gallager region, given that the all-zero codeword is transmitted, is upper bounded by

$$\begin{aligned} P\{\mathbf{r} \notin \mathfrak{R}\} &\leq P\{|n_1| > \sqrt{n}\epsilon/2\} + P\left\{\frac{1}{n}\sum_{i=2}^{n} n_i^2 > c^2\right\} \\ &< P\{|n_1| > \sqrt{n}\epsilon/2\} + P\left\{\frac{1}{n}\sum_{i=1}^{n} n_i^2 > c^2\right\}, \end{aligned} \tag{3.63}$$

where $n_1 = n_Z$ is the noise component along the Z axis and $\sum_{i=1}^{n} n_i^2$ is the total noise energy in all n dimensions. The noise component along the Z axis is a zero mean Gaussian random variable with variance $N_0/2$, where N_0 is the one-sided noise power spectrum. Therefore,

$$P\{|n_1| > \sqrt{n}\epsilon/2\} = 2Q\left(\sqrt{\frac{n\epsilon^2}{2N_0}}\right) < \exp\left(-\frac{n\epsilon^2}{4N_0}\right), \quad \text{for } \epsilon > 0. \tag{3.64}$$

This indicates a positive exponent of $E_0 = \epsilon^2/4N_0$ for $\epsilon > 0$.

On the other hand, the total noise energy is a *chi-squared* random variable with n degrees of freedom and mean $nN_0/2$. Using Lemma 3.2, the error exponent for the second

summand in (3.63) can be obtained as

$$E_1 = \frac{1}{2}\left(\frac{c^2}{N_0/2} - 1 - \log\frac{c^2}{N_0/2}\right), \quad \text{for } c > \sqrt{N_0/2}. \tag{3.65}$$

Next, we find an upper bound on the second summand in (3.62), which is equal to

$$\sum_{i\neq 0} P\{\mathbf{r}\in\Re, |\mathbf{r}-\mathbf{c}_i| < |\mathbf{r}-\mathbf{c}_0|\} = \int_0^{\omega_{\max}} A_\omega P\{\mathbf{r}\in\Re, |\mathbf{r}-\mathbf{c}_\omega| < |\mathbf{r}-\mathbf{c}_0|\}d\omega, \tag{3.66}$$

where $\mathbf{c}_\omega$ is a codeword of weight $w = \omega n$ and $\omega_{\max} = \dfrac{c}{\sqrt{E_N}+c} = \dfrac{\tan\theta}{1+\tan\theta}$, because only the median planes between the all-zero codeword and codewords of weight $0 < w < n\omega_{\max} = n\dfrac{c}{\sqrt{E_N}+c}$ intersect with the Gallager region as shown in figures 3.3 and 3.4. Note that $P\{\mathbf{r}\in\Re, |\mathbf{r}-\mathbf{c}_\omega| < |\mathbf{r}-\mathbf{c}_0|\}$ represents the probability that the received vector falls in the shaded area in figure 3.3 given that the all-zero codeword is transmitted. Since the disc is very thin (i.e. $\epsilon \simeq 0$), the distance from the all-zero codeword to the intersection of the Gallager region and the median plane is $\sqrt{\dfrac{\omega E_N}{1-\omega}}$.

If c is chosen to be $\sqrt{N_0/2}+\epsilon,\ \epsilon\to 0$, the error exponent defined by (3.65) is positive for $\epsilon > 0$. On the other hand, if we omit the height of the thin disk and the noise component along with the Z axis, n_Z, the disk transforms to the $n-1$ dimensional noise sphere. For large n, the $n-1$ dimensional normalized white Gaussian noise is uniformly distributed within a sphere with radius $\sqrt{N_0/2}$ [68]. The volume of the noise sphere (noted by $\mathcal{S}_1$ in figure 3.4) is $\dfrac{\pi^{(n-1)/2}}{\Gamma\left(\frac{n+1}{2}\right)}(N_0/2)^{(n-1)/2}$. Note that for $\sqrt{N_0/2}+\epsilon,\ \epsilon\to 0$, the cone approaches the $n-1$ dimensional noise sphere and becomes tangent to it. The intersection of the median plane between the all-zero codeword and a codeword of weight $w = \omega n$ with the $n-1$ dimensional noise sphere confines a cap (the shaded area in figure 3.4 noted by

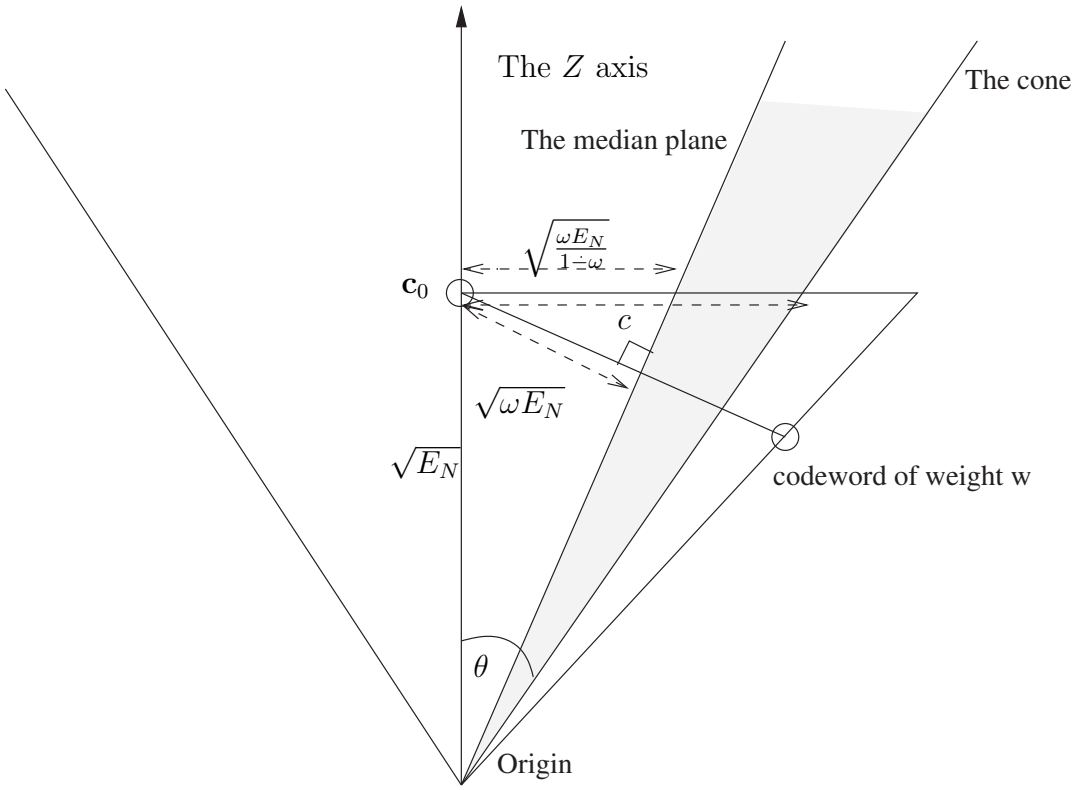

Figure 3.3: The median plane between the all-zero codeword and a codeword of weight $w = \omega n$ (side view).

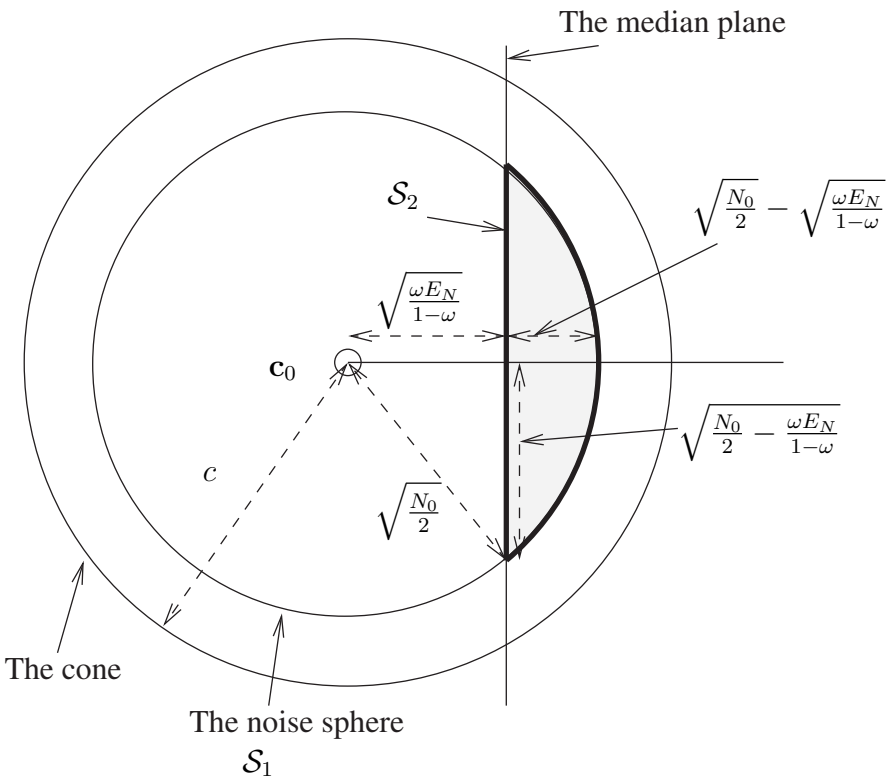

Figure 3.4: The intersection of the median plane and the Gallager region (top view).

$\mathcal{S}_2$) with radius $\sqrt{\frac{N_0}{2} - \frac{\omega}{1-\omega}E_N}$ and height $\sqrt{\frac{N_0}{2}} - \sqrt{\frac{\omega}{1-\omega}E_N}$. If the received vector given that the all-zero codeword is transmitted falls inside this cap, an error occurs. The volume of this cap is upper bounded by the volume of a $n-1$ dimensional sphere with radius $\sqrt{\frac{N_0}{2} - \frac{\omega}{1-\omega}E_N}$, which is $\frac{\pi^{(n-1)/2}}{\Gamma\left(\frac{n+1}{2}\right)}\left(\frac{N_0}{2} - \frac{\omega}{1-\omega}E_N\right)^{(n-1)/2}$. Note that this upper bound is tight, since the sphere cap and the sphere of radius $\sqrt{\frac{N_0}{2} - \frac{\omega}{1-\omega}E_N}$ differ in only one dimension. Comparing the volume of the sphere cap and the noise sphere, for a codeword $\mathbf{c}_i$ of weight $w = \omega n$,

$$\begin{aligned} P\{\mathbf{r} \in \mathfrak{R}, |\mathbf{r}-\mathbf{c}_i| < |\mathbf{r}-\mathbf{c}_0|\} &= \\ \frac{\text{vol}\{\mathcal{S}_2\}}{\text{vol}\{\mathcal{S}_1\}} &< \frac{\frac{\pi^{(n-1)/2}}{\Gamma\left(\frac{n+1}{2}\right)}\left(\frac{N_0}{2} - \frac{\omega}{1-\omega}E_N\right)^{(n-1)/2}}{\frac{\pi^{(n-1)/2}}{\Gamma\left(\frac{n+1}{2}\right)}(N_0/2)^{(n-1)/2}} \\ &= \left(1 - \frac{2\omega}{1-\omega}\frac{E_N}{N_0}\right)^{(n-1)/2}, \end{aligned} \tag{3.67}$$

whose exponent is

$$-\frac{1}{2}\log(1 - \frac{2\omega}{1-\omega}\frac{E_N}{N_0}). \tag{3.68}$$

Using the weight distribution described by (3.48) for codewords of weight $0 < w < n\frac{\sqrt{N_0/2}}{\sqrt{E_N}+\sqrt{N_0/2}}$, we finally arrive at the exponent for the second summand in (3.62)

$$E_2 = \min_{0<\omega<\frac{\sqrt{N_0/2}}{\sqrt{E_N}+\sqrt{N_0/2}}} \left\{2\left(\omega - \frac{1}{2}\right)^2 - \frac{1}{2}\log\left(1 - \frac{2\omega}{1-\omega}\frac{E_N}{N_0}\right)\right\} - R\log 2. \tag{3.69}$$

Since the error exponent in (3.65) is positive for $c = \sqrt{N_0/2} + \epsilon$, then the overall error exponent is positive if the error exponent in (3.69) is positive. □

Figure 3.5 shows the minimum signal to noise ratio for which the TSB error exponent for the Gaussian weight spectrum given in (3.69) is positive and compares it to that of the average spectrum. It also shows the capacity of BPSK signalling over an AWGN channel. The achievable rate predicted by the TSB for the Gaussian spectrum and for the average spectrum are very close to the BPSK capacity for code rates less than 1/2.

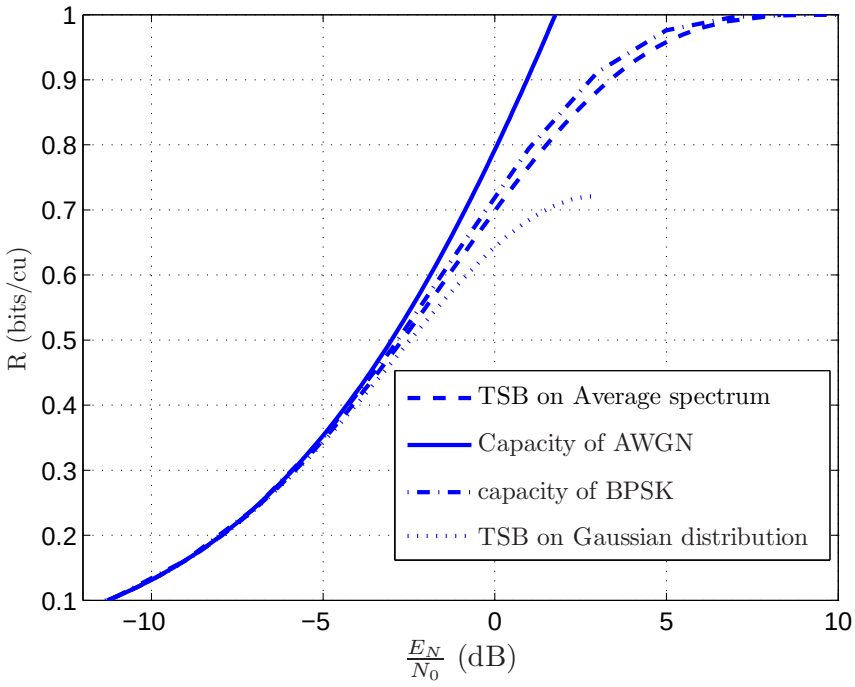

Figure 3.5: TSB bound vs capacity.

Figure 3.6 shows the dominant weight $\omega_c n$ on the error exponent in 3.69 for different SNR values:

$$\omega_c = \arg \min_{0<\omega<\frac{\sqrt{N_0/2}}{\sqrt{E_N}+\sqrt{N_0/2}}} \left\{ 2\left(\omega - \frac{1}{2}\right)^2 - \frac{1}{2}\log\left(1 - \frac{2\omega}{1-\omega}\frac{E_N}{N_0}\right)\right\}. \tag{3.70}$$

As the SNR increases, the error exponent is dominated by the error probability due to the codewords of lower weights. For $E_N/N_0 > 2 = 3$ dB, the dominant weight in (3.69) becomes zero as shown in figure 3.6 and the Gaussian approximation is no longer valid. This result matches the validity range derived by using the union bound to evaluate the cutoff rate for a Gaussian weight spectrum in section 3.3.

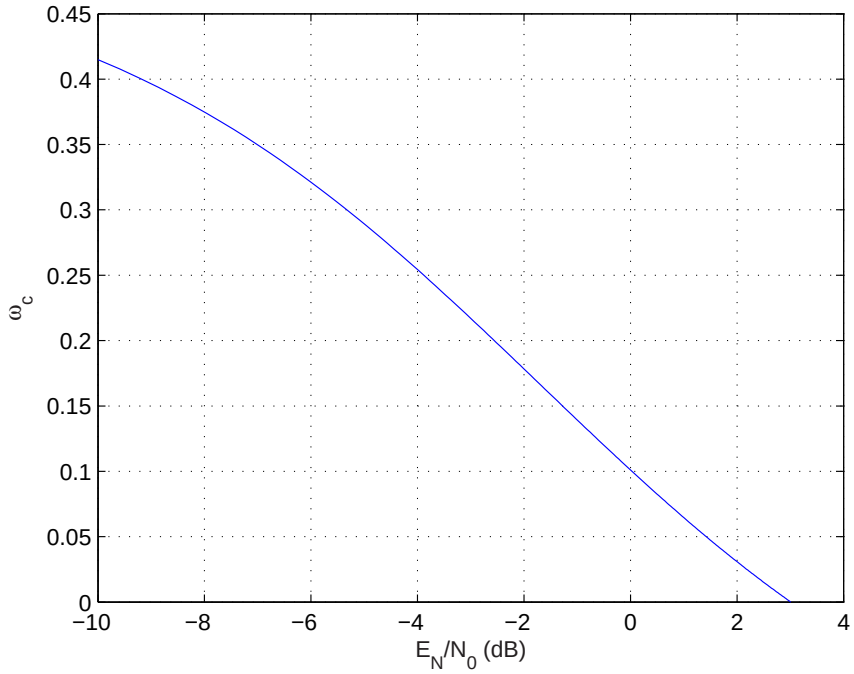

Figure 3.6: Dominant weight in the error exponent evaluation.

The derivations in this section remain valid for parallel concatenated turbo codes with $J > 2$ component codes as all the systematic and parity weights are Gaussian and each parity stream is an m-dependent sequence conditioned on the weight of the systematic and the other parity streams. If one punctures one or more of the parity streams to increase the

code rate, the central limit theorem is still applicable, if that puncturing leaves an infinite number of parity bits when $N \to \infty$.

The Gaussian weight distribution approximation is valid for the typical values of the Hamming weight. As the SNR increases, the error performance is determined by the codewords of lower weights. The number of low-weight codewords cannot be approximated by a continuous distribution. As we will see in Chapter 4, low-weight codewords appear only in certain structures. We will study the statistical properties of low weight codewords and their effect on the overall performance.

3.5 Summary

In this chapter, the asymptotic weight distribution of turbo codes is studied. It is shown that the weight distribution in its typical region is Gaussian. We also show that this weight distribution remains the same for almost any random interleaver and any nontrivial component codes. This weight distribution is compared with the "average spectrum" and the TSB is applied on the error performance of turbo code to find the region of signal to noise ratio and code rate values where the error probability converges to zero for a code with Gaussian distribution.

Chapter 4

Performance Analysis in the error floor region

4.1 Chapter Overview

As mentioned in the previous chapter, the weight distribution of the code for its typical weights is asymptotically Gaussian. However, the weight distribution for the low-weight codewords which affect the performance in the error floor region does not follow the Gaussian distribution. In this chapter, the statistical properties of the low-weight codewords will be evaluated and based on that the statistical properties of error floor and the minimum distance of the code will be derived. We also show that for multi-component codes, the error floor converges to zero as the block length increases. We present a method to expurgate the low-weight codewords.

4.2 Asymptotic Behavior of Low-weight Codewords

The error floor is caused by low-weight codewords. The number of low-weight codewords and their weights are determined by the RCC and the interleaver structure. To evaluate the statistical properties of the error floor among all the possible interleavers, the statistical properties of the low-weight codewords are required.

Consider the turbo code shown in figure 2.1 with two component codes. The probable low-weight codewords for large block lengths consist of some short single error events[1] with the systematic weight of two in both RCCs [12]. Each of these short error events is caused by two nonzero systematic bits that are separated by an integer multiple of the RCC impulse response period. In other words, an asymptotically probable codeword has an even systematic weight of $w_1 = 2M,\ M = 1, 2, \ldots$. Each RCC leaves the all-zero state M times and returns to it after an integer multiple of P transitions. This is equal to at least M repetitions of the RCC impulse response in each encoder. This phenomenon produces $\frac{K(P+1)}{2}$ nonzero parity bits, where $K \geq 2M$ is the number of RCC impulse response repetitions in the parity check sequences. Such a structure is denoted by type (M, K) where $K \geq 2M$. For a code consisting of J constituent codewords, the low-weight codeword of type (M, K) consists of M short error events in each parity stream. The systematic stream and all its $J-1$ interleaved versions contain M pairs of ones, each pair separated by an integer multiple of P as shown in figure 4.1.

To calculate the mean and the variance of the error floor, it is necessary to com-

[1] A single error event means leaving the zero-state and returning back to it for the first time.

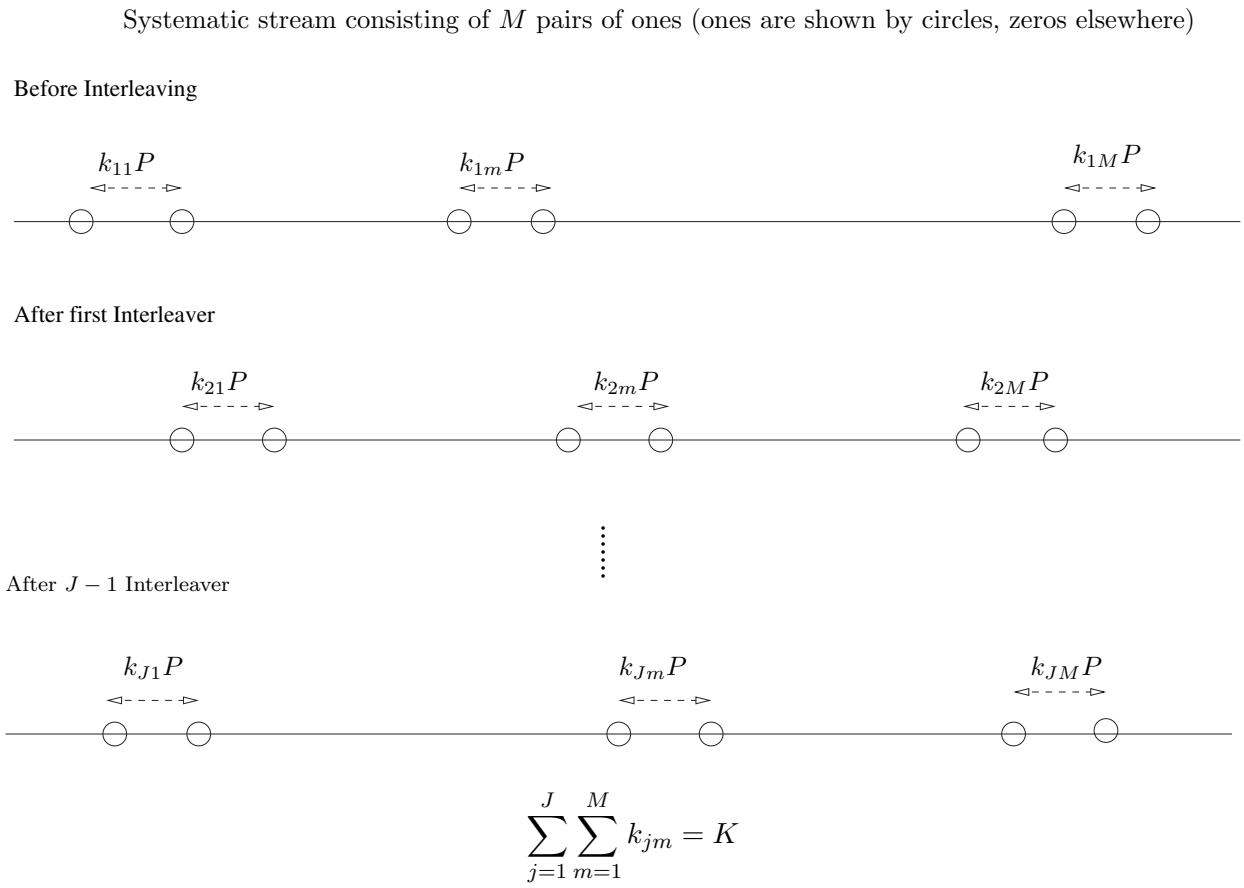

Figure 4.1: The structure of low-weight codewords of type (M, K).

pute the statistical properties of each low-weight structure. One can show that there are $\binom{K-1}{2M-1}$ ways to choose $2M$ positive integers whose sum is K. Equivalently, the structure of type (M, K) can be divided into $\binom{K-1}{2M-1}$ substructures. The number of codewords of each substructure has the same statistical properties as the number of codewords of type $(M, 2M)$.

Theorem 4.1. *The number of codewords of type* (M, K) *is a Poisson random variable with parameter* $\lambda_{M,K}^{(2)} = \binom{2M}{M}\binom{K-1}{2M-1}$, *where superscript* (2) *is used to denote the presence of two component codes.*

Proof. We first calculate the statistical properties of the number of codewords of type $(M, 2M)$ and then generalize the result to the other structures. There are $\binom{N}{M}$ systematic input combinations consisting of M pairs of ones, each pair with $P-1$ zeroes in between. This can be easily verified by determining the place of the first element of each pair. The overlapping pairs are neglected, because $N \gg M$. Such a structure generates $M(P+1)/2$ parity bits in the first convolutional encoder. There are the same number of parity bits in the second convolutional encoder, if the interleaver maps that systematic stream to another stream of the same structure. There are $\binom{N}{2M}$ ways to interleave a stream of length N and weight $2M$. However, among them, only $\binom{N}{M}$ result in M pairs of ones, each pair with $P-1$ zeroes in between; that is, the Bernoulli event that a low parity-weight generating stream changes to another one occurs with probability $\frac{\binom{N}{M}}{\binom{N}{2M}}$. The number of these Bernoulli events is $\binom{N}{M}$. These Bernoulli events are asymptotically independent because occupying a bit position in the interleaved stream by a certain information bit

does not asymptotically affect the probability for the other bits. As a result, the number of low-weight codewords of type $(M, 2M)$ is asymptotically a Poisson random variable with parameter

$$\lambda^{(2)}_{M,2M} = \frac{\binom{N}{M}}{\binom{N}{2M}} \binom{N}{M}. \tag{4.1}$$

For $N \to \infty$ and by using Stirling's approximation, (4.1) converges to

$$\lambda^{(2)}_{M,2M} = \binom{2M}{M}. \tag{4.2}$$

Finally, the Poisson parameter for the structure of type (M, K) is computed by multiplying $\lambda^{(2)}_{M,2M}$ by $\binom{K-1}{2M-1}$, which is

$$\lambda^{(2)}_{M,K} = \binom{2M}{M} \binom{K-1}{2M-1}. \tag{4.3}$$

□

As an example, there are approximately N codewords with the systematic weight of two, consisting of two nonzero bits separated by P. After interleaving, the distance between these two bits remains P with the probability of $2/N$. This occurs because these two bits can occupy about $N^2/2$ different places after interleaving, and only about N of the new places are separated by P. Then, the average number of low-weight codewords of this structure is two. On the other hand, there are four codewords with the systematic weight of 2 and parity weight of $3(P+1)/2$, averaged over all the possible interleavers, because there are two possible structures for this situation: distance P before interleaving and $2P$ after interleaving, and vice versa, and the Poisson parameter for each of two substructures

Table 4.1: Poisson parameters for different low-weight structures for P=7.

	K=2		K=3		K=4		K=5		K=6		K=7		K=8	
	λ	w	λ	w	λ	w	λ	w	λ	w	λ	w	λ	w
$M=1$	2	10	4	14	6	18	8	22	10	26	12	30	14	34
$M=2$					6	20	24	24	60	28	120	32	210	36
$M=3$									20	30	120	34	240	38
$M=4$													70	40

is two. Table 4.1 shows the Poisson parameter and the corresponding weights for some values of M and K.

4.2.1 Indecomposable Low-weight Codewords

In a linear binary codebook, the binary addition of two or more low-weight codewords results is another low-weight codeword. The new codeword is decomposable when the original low-weight codewords do not have common nonzero bit positions. Decomposable codewords can be easily ignored, because: (i) it easily follows that the decomposable codewords do not contribute to the walls of the Voronoi region of the all-zero codeword, and (ii) if each of the original low-weight codewords is expurgated, the decomposable codeword no longer exists.

The Poisson parameters calculated by (4.3) include both decomposable and indecomposable low-weight codewords. These Poisson random variables are not independent. For

example, the number of codewords of type $(M_1 + M_2, K_1 + K_2)$ depends on the number of codewords of types (M_1, K_1) and (M_2, K_2).

Theorem 4.2. *The number of indecomposable codewords of type* (M, K) *is a Poisson random variable with parameter* $\hat{\lambda}^{(2)}_{M,K} = \frac{2^{2M}}{2M}\binom{K-1}{2M-1}$.

Proof. Again, we begin with codewords of type $(M, 2M)$. A codeword of type $(M, 2M)$ consists of $2M$ systematic bits. There are $\binom{N}{2M}$ ways to choose $2M$ bits out of N systematic bits. Consider these bits as the $2M$ nodes of a graph. These bits form M pairs before and M pairs after interleaving. We denote each pair before interleaving by a red edge and each pair after interleaving by a blue edge. A graph with two edges for each node consists of one or more loops. Each loop represents an indecomposable codeword. We have one and only one indecomposable codeword of type $(M, 2M)$, if and only if there is only one loop in the graph. There are $(2M-1)!$ ways to form a loop with $2M$ nodes in such a way that each node has one blue edge and one red edge. For each edge in the graph, the probability that the corresponding systematic bits are separated by P trellis positions is $2/N$. Since the relative position of bits in different pairs are asymptotically independent, all pairs are separated by P before and after interleaving with probability $\left(\frac{2}{N}\right)^{2M}$. The number of low-weight codewords of type $(M, 2M)$ is the summation of many Bernoulli events with a low probability which is a Poisson random variable. Noting the above statements, the parameter of this random variable is

$$\hat{\lambda}^{(2)}_{M,2M} = \binom{N}{2M}(2M-1)!\left(\frac{2}{N}\right)^{2M}. \tag{4.4}$$

Table 4.2: Poisson parameters for different indecomposable low-weight structures.

	K=4		K=5		K=6		K=7		K=8		K=9		K=10	
	λ	$\hat{\lambda}$	λ	$\hat{\lambda}$	λ	$\hat{\lambda}$	λ	$\hat{\lambda}$	λ	$\hat{\lambda}$	λ	$\hat{\lambda}$	λ	$\hat{\lambda}$
$M=2$	6	4	24	16	60	40	120	80	210	140	336	224	504	336
$M=3$					20	$\frac{32}{3}$	120	64	420	224	1120	$\frac{1792}{3}$	2520	1344
$M=4$									70	32	560	256	2520	1152
$M=5$													252	$\frac{512}{5}$

For large N, (4.4) can be written as

$$\hat{\lambda}^{(2)}_{M,2M} = \frac{2^{2M}}{2M}. \tag{4.5}$$

The Poisson parameter for the number of indecomposable low-weight codewords of type (M, K) is the multiplication of $\hat{\lambda}^{(2)}_{M,2M}$ by $\binom{K-1}{2M-1}$:

$$\hat{\lambda}^{(2)}_{M,K} = \frac{2^{2M}}{2M}\binom{K-1}{2M-1}. \tag{4.6}$$

□

Table 4.2 compares the Poisson parameters of all low-weight codewords and indecomposable low-weight codewords for different structures. In this table, λ indicates the Poisson parameter of all low-weight codeword of one structure, while $\hat{\lambda}$ denotes the Poisson parameters corresponding to the indecomposable low-weight codewords.

Here, the Poisson parameter corresponding to the number of indecomposable low-wight codewords is evaluated based on an alternative approach based on the following lemma.

Lemma 4.1. *If X_m is a Poisson-distributed random variable of parameter $\hat{\lambda}_{m,2m}$, then*

$$E\left[\binom{X_m}{k_m}\right] = \frac{\hat{\lambda}_{m,2m}^{k_m}}{k_m!}. \tag{4.7}$$

Proof.

$$\begin{aligned} E\left[\binom{X_m}{k_m}\right] &= \sum_{k=k_m}^{\infty} \binom{k}{k_m} P\{X_m = k\} \\ &= \sum_{k=k_m}^{\infty} \frac{k!}{k_m!(k-k_m)!} e^{-\hat{\lambda}_{m,2m}} \frac{\hat{\lambda}_{m,2m}^{k}}{k!} \\ &= \frac{e^{-\hat{\lambda}_{m,2m}} \hat{\lambda}_{m,2m}^{k_m}}{k_m!} \sum_{k=k_m}^{\infty} \frac{\hat{\lambda}_{m,2m}^{k-k_m}}{(k-k_m)!} \\ &= \frac{e^{-\hat{\lambda}_{m,2m}} \hat{\lambda}_{m,2m}^{k_m}}{k_m!} \sum_{k=0}^{\infty} \frac{\hat{\lambda}_{m,2m}^{k}}{k!} \\ &= \frac{e^{-\hat{\lambda}_{m,2m}} \hat{\lambda}_{m,2m}^{k_m}}{k_m!} e^{\hat{\lambda}_{m,2m}} \\ &= \frac{\hat{\lambda}_{m,2m}^{k_m}}{k_m!}. \end{aligned} \tag{4.8}$$

□

In a large block turbo code, a decomposable low-weight codeword has a systematic weight of $2M \geq 4$, and consists of some smaller low-weight codewords which can be partitioned to k_m codewords of the systematic weight $2m$ for $m = 1, \ldots, M-1$. The k_m's are nonnegative integers that satisfy

$$\sum_{m=1}^{M-1} m k_m = M. \tag{4.9}$$

Again, we only consider codewords of type $(M, 2M)$. The same approach that was previously applied is still valid for the codewords of type (M, K), when $K > 2M$. The total

number of low-weight codewords of type $(M, 2M)$ is $\binom{2M}{M}$. Let us denote the average number (Poisson parameter) of indecomposable codewords of type $(M, 2M)$ by $\hat{\lambda}_{M,2M}$. If the number of indecomposable low-weight codewords of type $(m, 2m),\ m = 1, \ldots, M-1$, is X_m, then the number of decomposable codewords of type $(M, 2M)$ consisting of $k_m,\ m = 1, 2, \ldots, M-1$, codewords of types $(m, 2m)$, is

$$\prod_{m=1}^{M-1} \binom{X_m}{k_m}. \tag{4.10}$$

X_m is a Poisson random variable with parameter $\hat{\lambda}_{m,2m}$. Therefore, the average number of decomposable codewords is

$$\binom{2M}{M} - \hat{\lambda}_{M,2M} = \sum_{\sum_{m=1}^{M-1} m k_m = M} E\left[\prod_{m=1}^{M-1} \binom{X_m}{k_m}\right] = \sum_{\sum_{m=1}^{M-1} m k_m = M} \prod_{m=1}^{M-1} E\left[\binom{X_m}{k_m}\right]. \tag{4.11}$$

Equation (4.11) holds because Poisson random variables denoting the number of indecomposable codewords are asymptotically independent noting that generation of an indecomposable low-weight codeword in a large block turbo code does not affect the position occupied by other information bits.

Using Lemma 4.1, we can see that

$$\hat{\lambda}_{M,2M} = \binom{2M}{M} - \sum_{\sum_{m=1}^{M-1} m k_m = M} \prod_{m=1}^{M-1} \frac{\hat{\lambda}_{m,2m}^{k_m}}{k_m!} \quad , \quad \text{for } M > 1. \tag{4.12}$$

This recursive equation in conjunction with the fact that $\hat{\lambda}_{1,2} = 2$ yields another method to evaluate the Poisson parameter for the indecomposable, low-weight codewords of type $(M, 2M)$.

4.2.2 Minimum Distance of Turbo Codes

Using Poisson parameters in (4.6), we can evaluate the asymptotic probability mass function of the minimum distance over all possible interleavers. Note that the smallest low-weight structure with a nonzero number determines the minimum distance. In other words, if these structures are sorted in the ascending order of their weights (i.e., $w_i \leq w_{i+1},\ i = 1, 2, \ldots$) and Y_i is the number of low-weight codewords of the i'th structure, then the minimum distance of the code is w_i if

$$Y_j = 0, \quad j = 1, 2, \ldots, i-1 \quad \text{and} \quad Y_i \neq 0. \tag{4.13}$$

The probability of this event can be obtained by

$$P\{w_{\min} = w_i\} = P\{Y_1 = 0, Y_2 = 0, \ldots, Y_{i-1} = 0, Y_i \neq 0\}. \tag{4.14}$$

Since the number of indecomposable codewords of different types are independent Poisson random variables,

$$P\{w_{\min} = w_i\} = P\{Y_i \neq 0\} \prod_{j=1}^{i-1} P\{Y_j = 0\} = \exp\left\{-\sum_{j=1}^{i-1} \lambda_j\right\} \left(1 - \exp\{-\lambda_i\}\right), \tag{4.15}$$

where λ_i denotes the Poisson parameter of random variable Y_i. Figure 4.2 represents the pmf of the minimum distance of a large-block turbo code with $P = 3,\ 7,\ 15$.

4.3 Error Floor for Large Block Turbo Codes

In this section, the asymptotic behavior of the error floor will be studied. Using the results of the previous section, we can calculate the mean and the variance of the union bound

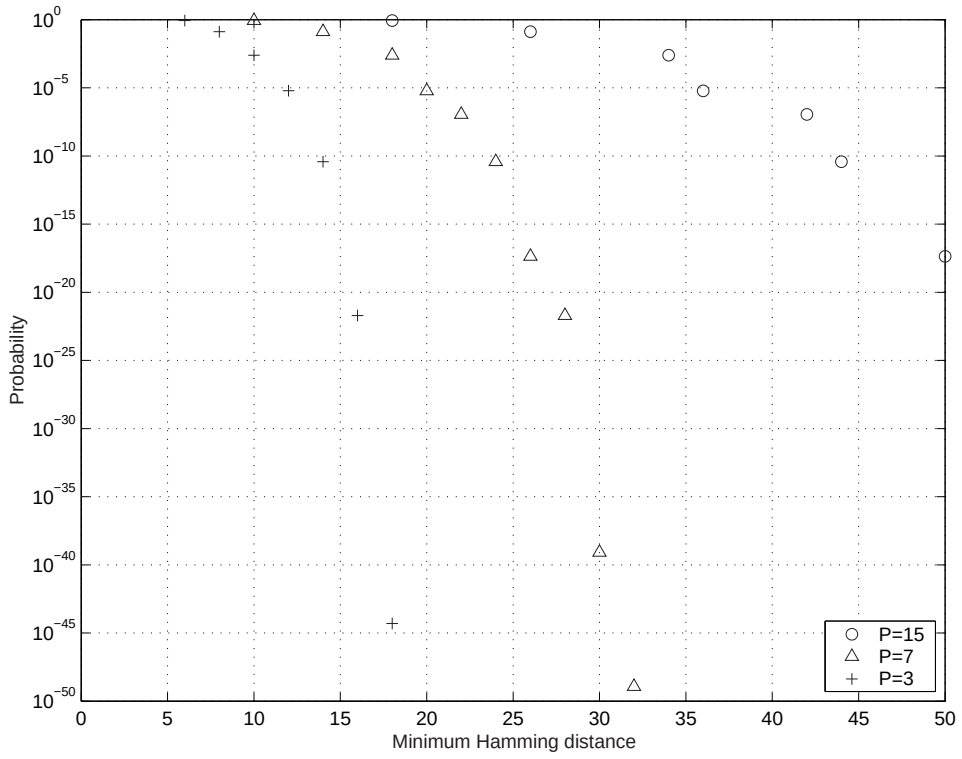

Figure 4.2: Asymptotic pmf of the turbo code minimum distance as $N \to \infty$.

on the error floor. Suppose that we sort the probable structures in the ascending order of their weights. Obviously, the minimum weight belongs to codewords of type $(1, 2)$. The weight of such codewords is $2+2(P+1)/2 = P+3$. Suppose that the number of codewords of the i'th structure is Y_i which is determined by a Poisson distribution with parameter λ_i. With the union bound, the error floor can be bounded as

$$P_e \leq P_u = \sum_i Y_i p_i, \tag{4.16}$$

where $p_i = Q\left(\sqrt{\frac{2E_N w_i}{N_0}}\right)$ is the corresponding error for any codeword of the i'th structure, where E_N is the energy per channel use. The mean of this upper bound can be determined by

$$E[P_u] = \sum_i \lambda_i p_i. \tag{4.17}$$

As mentioned earlier, the upper bound on the performance becomes tighter, if only indecomposable codewords are considered. On the other hand, since the Poisson random variables corresponding the number of indecomposable low-weight codewords are asymptotically independent, the variance of P_u can be evaluated by

$$\sigma^2_{P_u} = \sum_i \lambda_i p_i^2. \tag{4.18}$$

Figure 4.3 shows the mean and the standard deviation of the union bound that is applied on the error floor, by using the Poisson parameters in (4.3). As expected, both the mean and the standard deviation decrease when the SNR increases. As the SNR increases, the ratio between them converges to $\sqrt{2}$. This is because for this region of signal to noise

ratio values, only the codewords of the lowest weight structure, i.e., type (1,2), remain effective and the Poisson parameter for this type is two.

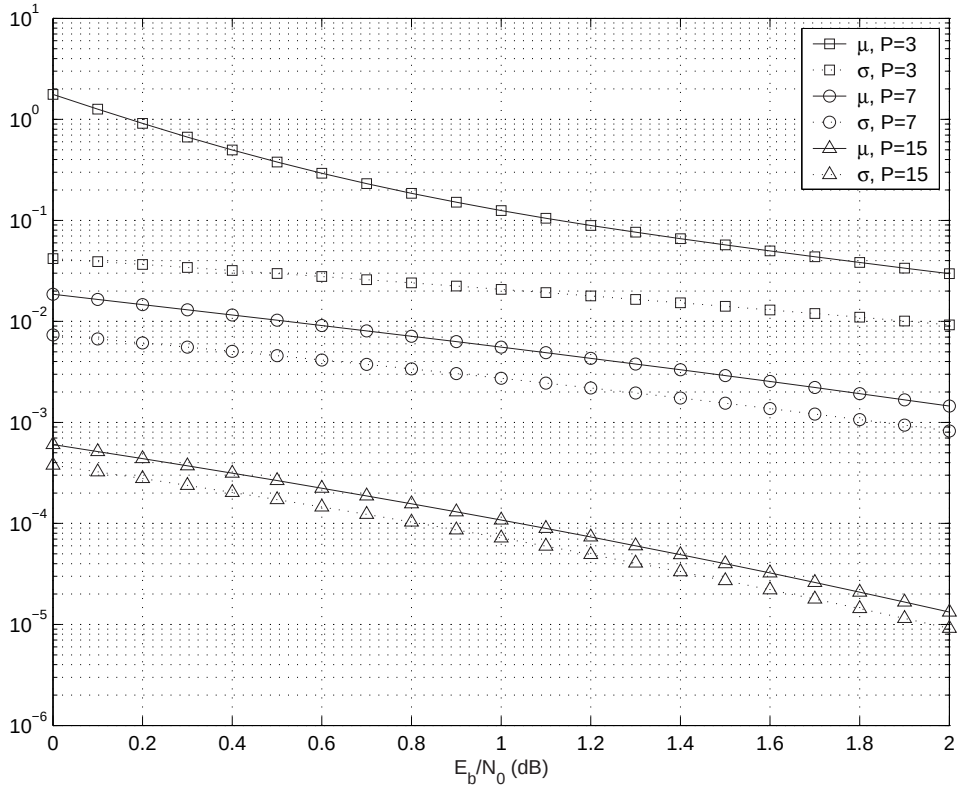

Figure 4.3: The mean and standard deviation of the union bound on the error floor for $P = 3, 7$ and 15.

In figure 4.4, the union bound on the average error floor using the Poisson distribution of the indecomposable low-weight codewords in (4.6) for a code with $P = 3$ is compared to the union bound evaluated by using Poisson parameters in (4.3). Since P is relatively small, the Poisson parameters of the indecomposable low-weight codewords result in a tighter bound than the Poisson parameters of the all low-weight codewords.

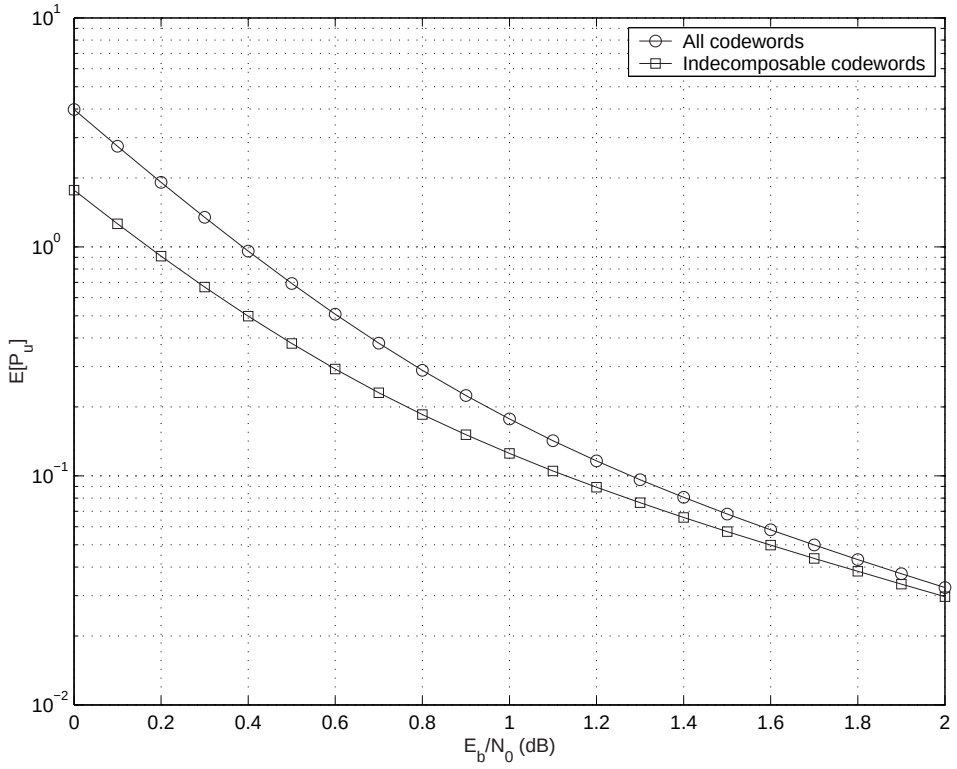

Figure 4.4: The error floor for a large-block turbo-code with $P = 3$.

4.4 Turbo Codes with Multiple Constituent Codes

In [15], it is shown that turbo codes with more component codes have a better performance when ML decoding is used. In [16], it is shown that for a randomly interleaved turbo code, the error floor decreases as $O(N^{-J+2+\epsilon})$ when J component codes are used. Next, we investigate the asymptotic behavior of multi-component turbo codes based on the Poisson distribution of low-weight codewords.

We focus on a parallel concatenated turbo code consisting of J component codes. These codes are concatenated via $J-1$ randomly chosen interleavers. The rate of this code is $\frac{1}{J+1}$. Higher code rates are achievable by puncturing and lower rates are achievable by using component codes of rate less than one.

Again, we concentrate on the codewords consisting of short error events due to two systematic bits. With a similar approach used in [12], the average number of codewords consisting of w_1 systematic bits and $A_j \leq \left\lfloor \frac{w_1}{2} \right\rfloor,\ j = 1, \ldots, J$ short error events in the jth encoder is $O(N^{-w_1(J-1)+\sum_j A_j})$. Within low weight codewords of systematic weight w_1, those codewords with $A_j = M = \lfloor \frac{w_1}{2} \rfloor,\ j = 1, \ldots, J$ are dominant. The parity weight of such codewords is $\frac{K(P+1)}{2}$, where $K \geq JM$.

We first study the statistical properties of codewords of weight $2M + \frac{JM(P+1)}{2}$ (i.e., $K = JM$). The number of low parity-weight patterns of systematic weight $2M$ is $\binom{N}{M}$. After each interleaver, only $\binom{N}{M}$ out of $\binom{N}{2M}$ possible outcomes will be a low-weight structure. As a result, the average number of low-weight codewords with the systematic

weight of $2M$ and parity weight of $\frac{JM(P+1)}{2}$ is

$$\lambda_{M,JM}^{(J)} = \frac{\binom{N}{M}^J}{\binom{N}{2M}^{J-1}} \simeq \frac{(2M!)^{J-1}}{(M!)^J} N^{-M(J-2)}. \tag{4.19}$$

For $J = 2$ component codes, $\lambda_{M,JM}^{(J)}$ is finite and non-zero. For $J > 2$, this average goes to zero as N increases. For structures with parity weight of $\frac{K(P+1)}{2}$, where $K \geq JM$, we have

$$\lambda_{M,K}^{(J)} = \lambda_{M,JM}^{(J)} \binom{K-1}{JM-1}. \tag{4.20}$$

The mean of the error floor based on the union bound can be bounded by

$$E\{P_e\} \leq \sum_M \sum_{K \geq JM} \lambda_{M,K}^{(J)} Q\left(\sqrt{\left(2M + \frac{K(P+1)}{2}\right)\frac{2E_N}{N_0}}\right). \tag{4.21}$$

Then,

$$E\{P_e\} \leq \sum_M \frac{(2M!)^{J-1}}{(M!)^J} N^{-M(J-2)} \sum_{K \geq JM} \binom{K-1}{JM-1} Q\left(\sqrt{\left(2M + \frac{K(P+1)}{2}\right)\frac{2E_N}{N_0}}\right). \tag{4.22}$$

The dominant term is the term corresponding to $M = 1$ as $N \to \infty$,

$$E\{P_e\} \leq 2^{J-1} N^{-J+2} \sum_{K \geq J} \binom{K-1}{J-1} Q\left(\sqrt{\left(2 + \frac{K(P+1)}{2}\right)\frac{2E_N}{N_0}}\right) = O(N^{-J+2}), \tag{4.23}$$

which indicates an interleaver gain of $J-2$. A similar result based on a different approach is reported in [16] for a turbo code with J component codes and uniform interleaving. This predicts a diminishing error floor for multi-component turbo codes with a performance improving inversely with the block length. This behavior is different from what we have seen in the waterfall region. Note that the performance of turbo code in the waterfall

region is determined by high-weight codewords and an error exponent and a cut-off rate could be defined for that region.

The BER for this code can be bounded by

$$E\{P_b\} \leq \sum_{M} \frac{(2M!)^{J-1}}{(M!)^{J}} N^{-M(J-2)-1} \sum_{K \geq JM} 2M \binom{K-1}{JM-1} P_{(M,K)}, \tag{4.24}$$

where

$$P_{(M,K)} = Q\left(\sqrt{\left(2M + \frac{K(P+1)}{2}\right)\frac{2E_N}{N_0}}\right).$$

This indicates an interleaver gain of $J-1$. As a result, although a turbo code consisting of two component codes has a nonzero asymptotic FER, its BER asymptotically tends to zero in the error floor region with almost any random interleaver.

If one punctures one or more of the parity streams to increase the code rate, the number of low-weight codewords remain unchanged but the weight of each codeword decreases. This increases the error floor in (4.22) and (4.24), but does not change the order of the error floor for bit and frame error probabilities, which are $O(N^{-J+2})$ and $O(N^{-J+1})$, respectively.

4.5 Transition Region

As discussed earlier, the asymptotic weight distribution of turbo code is Gaussian for typical values of weight, $w = O(N)$. In this region, the Central Limit Theorem applies to the systematic and parity weights. On the other hand, there are a few codewords with finite weight which consist of low systematic and parity weights and their number follows

a set of Poisson random variables. The two weight regions are separated by a transition region, with unbounded weight of $w = o(N)$ as $N \to \infty$, where the weight distribution emerges from a set of Poisson random variables to Gaussian.

Here, we find the average weight distribution of the code for the transition region among all possible interleavers. In this region, the systematic stream consists of many non-zero bits. However, the number of these bits is very small compared to the block length. As a result, two consecutive non-zero systematic bits (in the original systematic stream and all $J-1$ interleaved versions of it) are very far from each other. Such a systematic stream, can be modeled by N iid Bernoulli events with a very low success probability of $\frac{w_1}{N}$, where w_1 is the systematic weight. In this case, each parity stream is divided into w_1 segments. Each segment starts with a nonzero systematic bit and ends with the next nonzero systematic bit. If one RCC encoder is in the zero state before a nonzero systematic bit, it arrives to a non-zero state after it. If the previous state is a non-zero state, after a nonzero systematic bit, the RCC encoder arrives in one of $P-2$ non-zero states or the zero-state, depending on the current state. Since w_1 is very large, the law of the large number applies and in each RCC encoder, the encoder remains in the zero-state for about $\frac{w_1}{P+1}$ segments. Thus, with J component codes, the total number of non-zero segments is about $\left\lceil \frac{P}{P+1} J w_1 \right\rceil$. Considering that each systematic bit is one with probability $\frac{w_1}{N}$, the number of trellis transitions that each encoder corresponding to each segment is a geometric random variable with parameter $\frac{w_1}{N}$. As a result, the total number of trellis transitions of the non-zero segments in all J RCC encoders is a negative binomial random variable with

parameters $\left(\left\lceil \frac{P}{P+1}Jw_1 \right\rceil, \frac{w_1}{N}\right)$. Note that the overall parity weight is approximately $\left\lceil \frac{P+1}{2P}x \right\rceil$, where x is the overall length of the non-zero segments. Since there are $\binom{N}{w_1}$ different systematic inputs of weight w_1, the overall number of codewords of systematic weight $w_1 = o(N)$ and parity weight w_p is

$$A^{(J)}_{w_1,w_p} \simeq \binom{N}{w_1}\binom{x-1}{r-1}p^r(1-p)^{x-r}, \tag{4.25}$$

where $x = \left\lceil \frac{2P}{P+1}w_p \right\rceil$, $r = \left\lceil \frac{P}{P+1}Jw_1 \right\rceil$ is the total number of these segments and $p = \frac{w_1}{N}$. The weight enumerating function in (4.25) can be approximated by

$$A^{(J)}_{w_1,w_p} \simeq \frac{N^{w_1}}{w_1!}\frac{\left(\frac{2P}{P+1}w_p\right)^{\frac{P}{P+1}Jw_1}}{\left\lceil \frac{P}{P+1}Jw_1 \right\rceil!}\left(\frac{w_1}{N}\right)^{\frac{P}{P+1}Jw_1}. \tag{4.26}$$

The effect of this weight distribution on the error performance based on the union bound is

$$P_e < \sum_{w_1}\sum_{w_p} A^{(J)}_{w_1,w_p} Q\left(\sqrt{\frac{2E_N}{N_0}(w_1+w_p)}\right). \tag{4.27}$$

It is easy to see that for $w_1, w_p = o(N)$, the right hand-side of (4.27) converges to zero as $N \to \infty$. The error probability in (4.27) corresponds to codewords in the transition region for a code with a uniform interleaver. For a code with a pseudo-random interleaver, the effect of this region of weight on the overall performance depends on the order of the weight enumerating function. Note that with a randomly chosen interleaver, with probability one, the weight enumerating function has the same order as the weight distribution with the average interleaver in (4.25). As a result, with any randomly chosen interleaver, the effect of the transition region on the overall performance is negligible.

The three different weight regions and the corresponding conditional weight distributions are shown in figure 4.5, where $w = o(N)$ denotes weights where $\lim_{N\to\infty} w = \infty$ and $\lim_{N\to\infty} \frac{w}{N} = 0$, and $w = O(N)$ denotes weights linearly increasing with N.

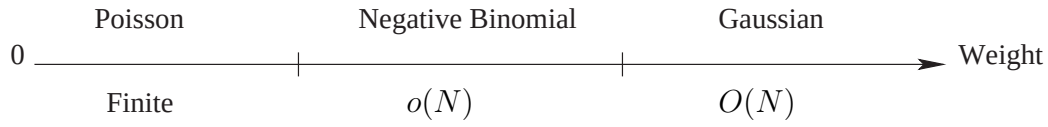

Figure 4.5: The weight distribution for different regions of weight.

4.6 Expurgating Low-weight Codewords

Low-weight codewords in turbo codes occur when a low-weight information stream results in a few parity bits in both recursive convolutional encoders. As mentioned before, the average number of low-weight codewords in which more than two nonzero systematic bits cause a short error event is zero for large block lengths. The important point is that the average number of such low-weight codewords does not increase with the block length N [12]. The number of low-weight codewords is a nonnegative integer with a finite average, and consequently, the probability of having an infinite number of such low-weight codewords approaches zero for large block lengths.

We can remove the effect of these low-weight codewords on the error floor region by expurgating them. Expurgating low-weight codewords decreases the dependency of the

turbo code performance on the RCCs and the interleaver structure, since the remaining codewords tend to the Gaussian weight distribution.

To expurgate these codewords, one way is to set one information bit in each low-weight codeword to zero as presented in [17]. However, no further puncturing is required to maintain the code rate, because when the block length is sufficiently large, the number of these bits is small in comparison with the block length, and consequently, the code rate is not affected.

In figure 4.6, the effect of expurgating low-weight codewords on the asymptotic mean of the error floor after expurgating codewords of the first low-weight structure (type (1,2), systematic weight 2 and parity weight $P+1$), and the second one (type (1,3), systematic weight 2 and parity weight $3(P+1)/2$) for a code of the rate 1/3 and $P=7$ is shown. On the average, there are two and four codewords of these two structures, respectively. The number of codewords in each of these two types does not exceed ten with probabilities 8×10^{-6} and 0.0028, respectively. Figure 4.7 presents the effect of the expurgation on a turbo code of the length 10000 and rate of 1/3 by using RCCs with three memory bits ($P=7$). The interleaver is chosen randomly. Simulation results show that by using this randomly chosen interleaver, three low-weight codewords with the systematic weight of two and parity weight of less than or equal to 12 (having the first or the second structure) exist.

In multi-component turbo codes, the Poisson parameters decrease with N. However, some low-weight codewords may still exist for large (but finite) block size turbo codes. As

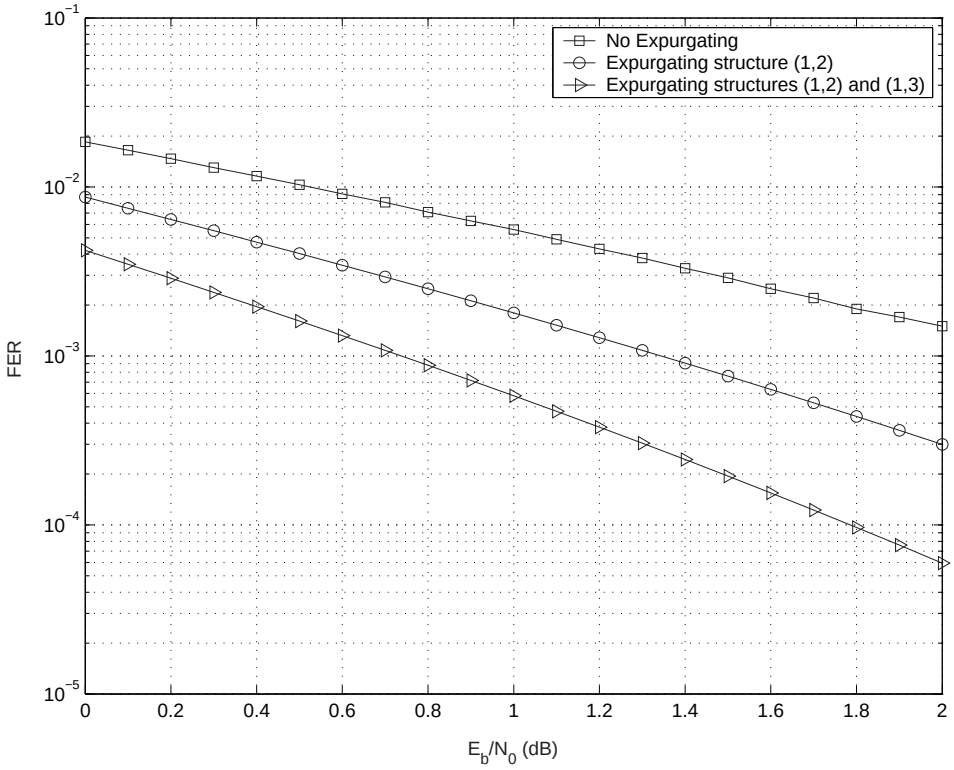

Figure 4.6: Asymptotic effect of expurgating two low-weight codeword structures

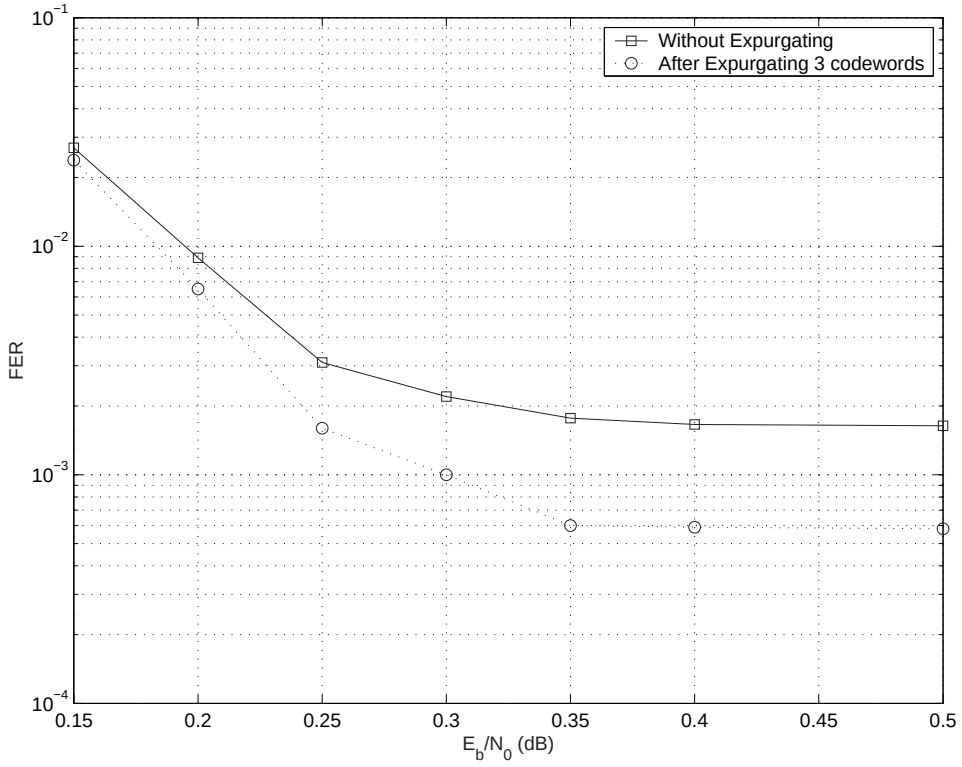

Figure 4.7: Effect of expurgating three low-weight codewords (N=10000)

the number of low-weight codewords is decreasing with the block length, it is possible to expurgate all codewords of weight less than a certain threshold. This threshold can be increased unbounded as the block length increases. This is because the total number of low-weight codewords with the systematic weight $2M$ and $K \leq K_{\max}$ is

$$\sum_{K=JM}^{K_{\max}} A_{M,K}^{(J)} = \frac{(2M!)^{J-1}}{(M!)^J} N^{-M(J-2)} \sum_{K=JM}^{K_{\max}} \binom{K-1}{JM-1} = O\left(N^{-M(J-2)} K_{\max}^{MJ}\right). \tag{4.28}$$

For $K_{\max} = o\left(N^{(J-2)/J}\right)$, the number of low-weight codewords in (4.28) is negligible compared to the block length and hence, expurgating those low-weight codewords does not change the code rate for $N \to \infty$. If all the low-weight codewords of systematic weight $1, 2, \ldots, 2M-1$ are expurgated, then the interleaver gain increases to $M(J-2)$, without affecting the code rate.

4.7 Summary

In this chapter, the performance of turbo codes in the error floor region is studied. It is shown that the weight distribution of the code for the low-weight codewords is a set of Poisson random variables. The statistical properties of these Poisson random variables are derived. Based on these random variables, the statistical properties of the error floor and the minimum distance of the code are evaluated. The interleaver gain for multi-component code is evaluated and a method to expurgate the low-weight codewords is presented. It is also shown that the weight distribution of the code in the transition from Gaussian to Poisson is negative binomial.

Chapter 5

Summary of Contributions

In this thesis, the asymptotic performance of turbo codes is studied. Our analysis is based on the code weight distribution. We show that for large block size turbo codes, the weight spectrum has three different regions: (i) the low-weight region where the weight spectrum is Poisson, (ii) the high weight region where the weight spectrum is Gaussian, and (iii) the transition region from Poisson to Gaussian where the weight spectrum is negative binomial. The performance of turbo codes in the waterfall region is mainly affected by the high-weight codewords. It is shown that for almost any random interleaver and any nontrivial recursive constituent code, the normalized weight distribution of turbo codes is asymptotically Gaussian and the code spectrum is very close to the average spectrum. A code with a randomly chosen interleaver performs the same as a code with the best interleaver with probability one and hence, interleaver optimization has little effect on the asymptotic performance of the code in the waterfall region. This Gaussian distribution

approaches the average spectrum defined in [14]. The TSB bound is applied on the Gaussian distribution and the region of code rate and SNR where the TSB error exponent is positive is evaluated. It is shown that the achievable rate is close to the capacity of BPSK signalling over AWGN channel. We also evaluated the weight of the dominant codewords in the performance of the code as a function of the signal to noise ratio. As the signal to noise ratio increases, the weight of the dominant codewords decreases and after a certain SNR, the Gaussian distribution is not valid for the dominant codewords.

In the error floor region (large SNR values), the performance of the code is affected by low-weight codewords and for a code with two RCCs, the number of these codewords remains finite as the block length increases. For large block lengths, only certain structures of these codewords remain possible. The number of indecomposable codewords of each structure is asymptotically characterized by a set of independent Poisson random variables. The frame error rate for these codes is bounded away from zero for a large block length. However, expurgating some low-weight codewords lowers the error floor. On the other hand, multi-component codes have a positive interleaver gain and the error floor disappears as the block length increases. The overall asymptotic error probability for these codes converges to zero either exponentially (in the Gaussian region) or polynomially (for Poisson and negative binomial regions).

Chapter 6

Future Research Directions

- In this work, the weight distribution of binary parallel concatenated turbo codes is used to evaluate the asymptotic performance of these codes over an AWGN channel with BPSK signalling. It is concluded that the weight distributions of these codes are Gaussian and the mean and the variance of the distribution remains the same for almost any component code and interleaver structure. These codes perform very close to the capacity for low values of spectral efficiency. It is known that when a higher spectral efficiency is desired, the performance of these codes with non-binary modulation schemes along with binary or non-binary codes is not very close to the capacity. It is desirable to analyze the pairwise distance spectrum of these codes and compare it to the distance spectrum of random coding. This will show the potential capability of these codes designed for higher spectral efficiencies.

- In non-binary modulation schemes, the coded bits are mapped to the modulation con-

stellation points through another interleaver. The asymptotic effect of the mapping interleaver on the performance of turbo codes for non-binary modulation is another extension to this study.

- In this thesis, we find the region of the signal to noise ratio and code rate values where the performance of the code due to high-weight codewords converges to zero. This helps us to evaluate the waterfall when ML decoding is used and compare it to the capacity. However, for a large (but finite) code length and signal to noise ratio values higher than the capacity, the value of the error exponent indicates how fast the performance improves as the block length increases. Using the same approach as provided in section 3.4, one can find the tightest error exponent on a code with Gaussian distribution using the tangential sphere bounding technique.

- There are some bounding methods [69] which give tighter bounds than the tangential sphere bound. It is desirable to apply these bounding techniques on the Gaussian weight distribution to improve the asymptotic achievable rate based on the Gaussian weight distribution.

- It is known that, the serial concatenated turbo codes and LDPCs perform better in the error floor region than parallel concatenated codes. Analyzing the weight distribution of these codes in their typical and low-weight codeword regions is another extension to the work presented in this thesis.

Bibliography

[1] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes (1)," in *IEEE International Conference on Communications*, Geneva, Switzerland, May 1993, pp. 1064–1070.

[2] S. Benedetto and G. Montorsi, "Unveiling turbo codes: Some results on parallel concatenated coding schemes," *IEEE Trans. on Inform. Theory*, vol. 42, pp. 409–428, March 1996.

[3] G. Battail, "A conceptual framework for understanding turbo codes," *IEEE Journal on Selected Areas in Communications*, vol. 16, pp. 245–254, Feb. 1998.

[4] G. Battail, C. Berrou, and A. Glavieux, "Psuedo-random recursive convolutional coding for near-capacity performance," in *IEEE Globecom Conference*, Houston, USA, Nov. 1993, pp. 23–27.

[5] G. Battail, "On random-like codes," in *4th Canadian Workshop Inform. Theory*, Lac Delage, PQ, Canada, May 1995, pp. 23–27.

[6] G. Battail, "Construction explicite do bons code longs," *Ann. Télécommun.*, vol. 44, pp. 392–404, July-Aug. 1989.

[7] N. Shulman, "Random coding techniques for nonrandom codes," *IEEE Trans. on Inform. Theory*, vol. 45, pp. 2101–2104, Sep. 1999.

[8] R. Gallager, "A simple derivation of the coding theorem and some applications," *IEEE Trans. on Inform. Theory*, vol. 11, pp. 3–18, Jan. 1965.

[9] T. M. Duman and M. Salehi, "New performance bounds for turbo codes," *IEEE Trans. on Communications*, vol. 46, pp. 717–723, June 1998.

[10] I. Sason and S. Shamai, "Improved upper bounds on the ML decoding error probability of parallel and serial concatenated turbo codes via their ensemble distance spectrum," *IEEE Trans. on Inform. Theory*, vol. 46, pp. 24–47, January 2000.

[11] I. Sason and S. Shamai, "Variations on the Gallager bounds, connections, and applications," *IEEE Trans. on Inform. Theory*, vol. 48, pp. 3029–3051, December 2002.

[12] L. C. Perez, J. Seghers, and D. J. Costello Jr., "A distance spectrum interpretation of turbo codes," *IEEE Trans. on Inform. Theory*, vol. 42, pp. 1698–1709, November 1996.

[13] P. Robertson, "Improving decoder and code structure of parallel concatenated recursive systematic (turbo) codes," in *IEEE Universal Personal Communications Conference*, San Diego, USA, September 1994, pp. 183–187.

[14] G. Poltyrev, "Bounds on the decoding error probability of binary linear codes via their spectra," *IEEE Trans. on Inform. Theory*, vol. 40, pp. 1284–1292, July 1994.

[15] C. Tanriover, B. Honary, J. Xu, and S. Lin, "Improving turbo code error performance by multifold coding," *IEEE Comm. Letters*, vol. 6, pp. 193–195, May 2002.

[16] H. Jin and R. J. McEliece, "Coding theorems for turbo code ensembles," *IEEE Trans. on Inform. Theory*, vol. 48, pp. 1451–1461, June 2002.

[17] F. Daneshgaran, M. Mondin, and P. Mulassano, "Turbo codes optimization via trace-bit injection and selective puncturing," in *IEEE International Conference on Communications*, April-May 2002, vol. 3, pp. 1706–1710.

[18] M. Öberg and P. H. Siegel, "Application of distance spectrum analysis to turbo code performance improvement," in 35^{th} *Annual Allerton Conference on Communication, Control, and Computing*, Allerton, IL, USA, Sep. 1997, pp. 701–710.

[19] C. E. Shannon, "A mathematical theory of communications," *Bell Syst. Tech. J.*, vol. 27, pp. 379–423 and 623–656, July and Oct. 1948.

[20] Robert G. Gallager, *Information Theory and Reliable Communication*, John Wiley & Sons, Inc., 1968.

[21] S. Le Goff, A. Glavieux, and C. Berrou, "Turbo-codes and high spectral efficiency modulation," in *ICC'94*, New Orleans, LA, USA, May 1994, pp. 645–649.

[22] S. Benedetto and G. Montorsi, "Average performance of parallel concatenated block codes," *Electronics Letters*, vol. 31, pp. 156–158, Feb. 1995.

[23] S. W. Golomb, *Shift Register Sequences*, San Francisco, Holden-Day, 1967.

[24] I. Sason, E. Teletar, and R. Urbanke, "On the asymptotic inputoutput weight distributions and thresholds of convolutional and turbo-like encoders," *IEEE Trans. on Inform. Theory*, vol. 48, pp. 3052–3061, December 2002.

[25] O. Y. Takeshita, M. P. C. Fossorier, and D. J. Costello, "A new technique for computing the weight spectrum of turbo codes," *IEEE Comm. Letters*, vol. 3, pp. 251–253, August 1999.

[26] G. D. Forney Jr., "The Viterbi algorithm," *proc. IEEE*, vol. 61, pp. 268–276, March 1973.

[27] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. on Inform. Theory*, vol. 20, pp. 284–287, March 1974.

[28] L. Ping and K.L. Yeung, "Symbol-by-symbol decoding of the golay code and iterative decoding of concatenated Golay codes," *IEEE Trans. on Inform. Theory*, vol. 45, pp. 2558–2562, November 1999.

[29] Ye Liu, Shu Lin, and M.P.C. Fossorier, "MAP algorithms for decoding linear block codes based on sectionalized trellis diagrams," *IEEE Trans. on Communications*, vol. 48, pp. 577–586, April 2000.

[30] S. Riedel, "Symbol-by-symbol MAP decoding algorithm for high-rate convolutional codes that use reciprocal dual codes," in *IEEE Journal on Selected Areas in Communications*, February 1998, vol. 16, pp. 175–185.

[31] S. ten Brink, "Convergence of iterative decoding," *IEEE Electronic Letters*, vol. 35, pp. 806–808, May 1999.

[32] H. El-Gamal and A. R. Hammons Jr, "Analyzing the turbo decoder using the Gaussian approximation," *IEEE Trans. on Inform. Theory*, vol. 47, pp. 671–686, February 2001.

[33] D. Divsalar, S. Dolinar, and F. Pollara, "Iterative turbo decoder analysis based on density evolution," *IEEE Journal on Selected Areas in Communications*, vol. 19, pp. 891–907, May 2001.

[34] M. Breiling, "A logarithmic upper bound on the minimum distance of turbo codes," *IEEE Trans. on Inform. Theory*, vol. 50, pp. 1692–1710, Aug. 2004.

[35] D. Truhachev, M. Lentmaier, O. Wintzell, and K. Sh. Zigangirov, "On the minimum distance of turbo codes," in *IEEE International Symp. on Inform. Theory*, Lausanne, Switzerland, July 2002, p. 84.

[36] F. Daneshgaran and M. Mondin, "Permutation fixed points with application to estimation of minimum distance of turbo codes," *IEEE Trans. on Inform. Theory*, vol. 46, pp. 2336–2349, Nov. 2000.

[37] R. Garello, P. Pierleoni, and S. Benedetto, "Computing the free distance of turbo codes and serially concatenated codes with interleavers: algorithms and applications," *IEEE Journal on Selected Areas in Communications*, vol. 19, pp. 800–812, 2001.

[38] E. Rosnes and Ø. Ytrehus, "On algorithms for determination of turbo code weight distribution," in *IEEE Int. Symp. on Inform. Theory*, Lausanne, Switzerland, July 2002, p. 82.

[39] K. Wu, H. Li, and Y. Wang, "Influence of interleaver on minimum turbo code distance," *IEEE Electronics Letters*, vol. 35, pp. 1456–1458, Aug. 1999.

[40] C. Di, R. Urbanke, and T. Richardson, "Weight distributions: How deviant can you be," in *IEEE Int. Symp. on Inform. Theory*, July 2001, p. 50.

[41] E. Biglieri and V. Volski, "Approximately gaussian weight distribution of the iterated product of single-parity-check codes," *IEE Electronics Letters*, vol. 30, pp. 923–924, June 1994.

[42] D. Yue and E. Yang, "Asymptotically gaussian weight distribution and performance of multicomponent turbo block codes and product codes," *IEEE Transactions on Communications*, vol. 52, pp. 728–736, May 2004.

[43] S. Dolinar and D. Divsalar, "Weight distributions for turbo codes using random and nonrandom permutations," *JPL TDA Progr. Rep.*, vol. 42-122, pp. 56–65, Aug. 1995.

[44] H.R. Sadjadpour, N.J.A. Sloane, M. Salehia, and G. Nebe, "Interleaver design for turbo codes," *IEEE Journal on Selected Areas in Communications*, vol. 19, pp. 831–837, May 2001.

[45] O. Y. Takeshita and D. J. Costello, "New deterministic interleaver designs for turbo codes," *IEEE Trans. on Inform. Theory*, vol. 46, pp. 1988–2006, September 2000.

[46] A. K. Khandani, "Optimization of the interleaver structure for turbo-codes," in *Canadian Workshop on Information Theory*, Kingston, Canada, June 1999, pp. 25–28.

[47] A. K. Khandani, "Design of the turbo-code interleaver using hungarian method," *IEE Electronics Letters*, vol. 34, pp. 63–65, January 1998.

[48] F. Daneshgaran and M. Mondin, "Optimized turbo codes for delay constrained applications," *IEEE Trans. on Inform. Theory*, vol. 48, pp. 293–305, Jan. 2002.

[49] F. Daneshgaran and M. Mondin, "Design of interleavers for turbo codes: iterative interleaver growth algorithms of polynomial complexity," *IEEE Trans. on Inform. Theory*, vol. 45, pp. 1845–1859, Sept 1999.

[50] W. Feng, J. Yuan, and B.S. Vucetic, "A code-matched interleaver design for turbo codes," *IEEE Trans. on Communications*, vol. 50, pp. 926–937, June 2002.

[51] J. Yuan, B. Vucetic, and W. Feng, "Combined turbo codes and interleaver design," *IEEE Transactions on Communications*, vol. 47, pp. 484–487, April 1999.

[52] J.A. Briffa and V. Buttigieg, "Interleaving and termination in unpunctured symmetric turbo codes," *IEE Proceedings on Communications*, vol. 149, pp. 6–12, Feb. 2002.

[53] F. Vatta, B. Scanavino, A. Banerjee, and D.J. Costello, "On the design of nonsystematic turbo codes," in *IEEE Int. Symp. on Inform. Theory*, Yokohama, Japan, June-July 2003, p. 320.

[54] C. Y. Liu, H. Tang S. Lin, and M. P. C. Fossorier, "An interactive concatenated turbo coding system," in *IEEE Vehicular Technology Conference*, Sept. 2002, vol. 51, pp. 998–1010.

[55] O. Y. Takeshita, O. M. Collins, P. C. Messay, and D. J. Costello, "On the frame-error rate of concatenated turbo codes," *IEEE Trans. on Communications*, vol. 49, pp. 602–608, April 2001.

[56] M. Ferrari, F. Scalise, and S. Bellini, "Prunable s-random interleavers," in *IEEE International Conference on Communications (ICC)*, New York, USA, April 2002, vol. 3, pp. 1711–1715.

[57] M. Eroz and A. R. Hammons, "On the design of prunable interleavers for turbo codes," in *Vehicular Technology Conference*. IEEE, May 1999, vol. 2, pp. 1669–1673.

[58] G. Zhu and F. Alajaji, "Turbo codes for nonuniform memoryless sources over noisy channels," *IEEE Communications Letters*, vol. 6, pp. 64–66, Feb. 2002.

[59] R. M. Fano, *Transmission of Information, a Statistical Theory of Communications*, jointly published by MIT Press and John Wiley & Sons, 1961.

[60] I. Shamai, S.; Sason, "Variations on the gallager bounds, connections, and applications," *IEEE Trans. on Inform. Theory*, vol. 48, pp. 3029–3051, December 2002.

[61] E. R. Berlekamp, "The technology of error-correcting codes," in *Proc. IEEE*, May 1980, pp. 564–593.

[62] H. Herzberg and G. Poltyrev, "Techniques of bounding the probability of decoding error for block-coded modulation structures," *IEEE Trans. on Inform. Theory*, vol. 40, pp. 903–911, May 1994.

[63] G. Poltyrev, "Bounds on the decoding error probability of binary linear codes via their spectra," *IEEE Trans. on Inform. Theory*, vol. 40, pp. 1284–1292, July 1994.

[64] H. Herzberg and G. Poltyrev, "The error probability of m-ary psk block-coded modulation schemes," *IEEE Trans. on Commun.*, vol. 44, pp. 427–433, April 1996.

[65] S. Yousefi and A.K. Khandani, "Generalized tangential sphere bound on the mldecoding error probability of linear binary block codes in awgn interference," *IEEE Trans. on Inform. Theory*, vol. 50, pp. 2810–2815, Nov. 2004.

[66] J. K. Patel and C. B. Read, *Handbook of the Normal Distribution*, Dekker Inc., second edition, 1996.

[67] John G. Proakis, *Digital Communication*, McGraw-Hill, fourth edition, 2001.

[68] J. M. Wozencraft and I. M. Jacobs, *Principles of Communication Engineering*, John Wiley, New York, 1965.

[69] S. Yousefi and A.K. Khandani, "A new upper bound on the ML decoding error probability of linear binary block codes in AWGN interference," *IEEE Trans. on Inform. Theory*, vol. 50, pp. 3026 – 3036, Dec. 2004.