PERFORMANCE ANALYSIS IN TERMS OF BIT ENERGY TO NOISE RATIO SPECTRAL DENSITY ON DECODING ALGORITHMS OF TURBO CODES

NOORLIN RAWAIDA BINTI MOHD ISA

This report is submitted in partial fulfillment of the requirements for the award of the Bachelor of Electronic Engineering (Telecommunication Electronic) with Honours

Faculty of Electronic and Computer Engineering
Universiti Teknikal Malaysia Melaka

April 2010

UNIVERSTI TEKNIKAL MALAYSIA MELAKA
FAKULTI KEJURUTERAAN ELEKTRONIK DAN KEJURUTERAAN KOMPUTER

BORANG PENGESAHAN STATUS LAPORAN
PROJEK SARJANA MUDA II

Tajuk Projek	:	Performance Analysis in terms of Bit Energy to Noise Ratio Spectral Density on Decoding Algorithms of Turbo Codes
Sesi Pengajian	:	0 9 / 1 0

Saya NOORLIN RAWAIDA BINTI MOHD ISA
(HURUF BESAR)

mengaku membenarkan Laporan Projek Sarjana Muda ini disimpan di Perpustakaan dengan syarat-syarat kegunaan seperti berikut:

1. Laporan adalah hakmilik Universiti Teknikal Malaysia Melaka.
2. Perpustakaan dibenarkan membuat salinan untuk tujuan pengajian sahaja.
3. Perpustakaan dibenarkan membuat salinan laporan ini sebagai bahan pertukaran antara institusi pengajian tinggi.
4. Sila tandakan (√) :

☐	**SULIT***	*(Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972)
☐	**TERHAD****	**(Mengandungi maklumat terhad yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan)
☐	**TIDAK TERHAD**	

Disahkan oleh:

(TANDATANGAN PENULIS)

(COP DAN TANDATANGAN PENYELIA)

Tarikh: ………………………..

Tarikh: ………………………..

I hereby declare that this report entitled “Performance analysis in term of bit energy to noise ratio spectral density on decoding algorithms of Turbo Codes” is the result of my own research except as cited in the references.

Signature	:	
Author	:	Noorlin Rawaida Binti Mohd Isa
Date	:	APRIL 2010

"I hereby declare that I have read this report and in my opinion this report is sufficient in terms of the scope and quality for award of Bachelor of Electronic Engineering (Telecommunication Electronics). with honours"

Signature	:	
Supervisor's Name	:	Pn Noor Shahida Binti Mohd Kasim
Date	:	APRIL 2010

DEDICATION

To my mother, father and beloved family, thank you for spiritual and supportive

ACKNOWLEDGEMENT

Alhamdulillah, thankful to ALLAH because with HIS grace and blessing give me an opportunity to recite this project. I would like to express my gratitude and appreciation to my supervisors Puan Noor Shahida Bt Mohd Kasim for her guidance, effort and useful, for keeping me on my toes, and for her kind understanding throughout this project from the beginning stage until the final stage. I am especially grateful for all the help she provided and resources she made available without which the project would not have reached its current stage.

My gratitude appreciation also goes to my beloved parent and family that giving me the strength and much needed moral that resulted in my project progression. Special thanks to my colleagues and friends that always give support and cooperation.

Last but not least, I like to take this opportunity to express my appreciation to all the individuals for giving me the courage, perseverance and advice during the progression of this project.

ABSTRACT

Turbo Codes are a class of high-performance error correction codes. It used to achieve maximal information transfer over a limited-bandwidth communication link in the presence of data-corrupting noise. Turbo codes come closest to approaching the Shannon limit, the theoretical limit of maximum information transfer rate over a noisy channel. This project is about performance analysis in terms of bit energy to noise ratio spectral density (Eb/No) on decoding algorithms of turbo codes. This project proposes a method to improve performance between Soft Output Viterbi Algoritms (SOVA) and Log-Maximum A Posteriori Probability (Log-MAP), and simulated by MATLAB. SOVA and Log-MAP can be used in an iterative decoder and share some common operation to reduced consumption power. The calculation of Log-Map algorithm is obviously more complex than Sova algorithm. The final simulation results will shows the best performance between Log-MAP and SOVA which methods can be improves the bit energy to noise ratio spectral density (Eb/No) on decoding algorithms of turbo by using MATLAB over AWGN channel. Besides that, the effect of a range of system parameters is investigated in a systematic fashion, in order to gauge Turbo Codes performance.

Keywords: Turbo Codes, SOVA, Log-MAP, MATLAB

ABSTRAK

"*Kod Turbo*" adalah kelas kod pembetulan kesalahan yang berprestasi tinggi. Ini digunakan untuk memindahkan maklumat yang maksimum melalui lebar jalur perhubungan komunikasi yang terhad. dalam kehadiran maklumat yang dirosakkan oleh hingar. "*Kod Turbo*" menghampiri *"had shanon"*, iaitu teori mengenai maksimum maklumat dapat dihantar secara limit melalui sebuah saluran hingar. Projek ini adalah berkaitan dengan analisis prestasi dalam istilah 'tenaga bit per nisbah ketumpatan spektrum hingar (Eb / No) pada algoritma penyahkodan pada turbo kod. Projek ini mencadangkan suatu kaedah untuk memperbaiki prestasi antara "*Soft Output Viterbi Algorithms*" (SOVA) dan "*Log-Maximum a posteriori Probability*" (Log-MAP), projek ini disimulasikan dengan menggunakan "*MATLAB*". "*SOVA* dan *Log-MAP"* boleh digunakan sebagai penyahkod yg berulang-ulang dan berkongsi operasi yang sama untuk mengurangkan penggunaan kuasa. Pengiraan bagi "*Log-MAP"* algoritma adalah lebih kompleks berbanding dengan "*SOVA"* algorithma. Keputusan akhir simulasi akan menunjukkan prestasi yang paling terbaik di antara "*Log-MAP* dan *SOVA"* di mana kaedah-kaedah ini boleh meningkatkan tenaga bit per nisbah ketumpatan spektrum hingar (Eb / No) pada algoritma penyahkodan "*kod-kod turbo"* dengan meggunakan "*MATLAB"* simulasi melalui saluran "*AWGN"*. Selain itu, kesan ke atas sistem parameter juga disiasat secara sistematik untuk mengukur prestasi "*Kod Turbo*".

Kata kunci: Kod Turbo, *SOVA, Log-MAP, MATLAB.*

TABLE OF CONTENTS

CONTENTS		PAGE
TITLE OF PROJECT		**i**
REPORT STATUS VERIFICATION FORM		**ii**
DECLARATION		**iii**
APPROVAL		**iv**
DEDICATION		**v**
ACKNOWLEDGEMENT		**vi**
ABSTRACT		**vii**
TABLE OF CONTENTS		**ix**
LIST OF TABLES		**xii**
LIST OF FIGURES		**xiv**
LIST OF ABBREVAITION		**xvi**
CHAPTER 1: INTRODUCTION		
1.1	Project Background	1
1.2	Problem Statement	2
1.3	Project Objectives	3
1.4	Scope of Works	3
1.5	Important of the Project	3
1.6	Report Structure	4

CHAPTER 2: LITERATURE REVIEW

2.1 Channel Coding 5
2.2 Turbo Codes 6
2.3 Fundamentals of Turbo Codes 9
2.3.1 Introduction 9
2.3.2 Log-Likelihood Ratio 10
2.4 Bit Energy to Noise Ratio Spectral Density (Eb/No) 11
2.5 Decoding Principles 13
2.5.1 Decoding Algorithms 14
2.6 Introduction to SOVA for Turbo Codes 15
2.6.1 A Basic Description of the SOVA 15
2.6.2 Soft Output Viterbi Algorithms (SOVA) 16
2.7 Log-MAP Algorithms 17
2.8 Puncturing 19
2.9 Interleaver 20
2.10 Performance Factors and Tradeoffs 22
2.11 Introduction to MATLAB 22
2.11.1 The MATLAB System 23
2.12 Advantages and Disadvantages 24
2.12.1 Advantages 24
2.1.2.2 Disadvantages 25
2.13 Principles and Application 25

CHAPTER 3: METHODOLOGY

3.1 Methodology 26
3.2 Project Flowchart 27
3.3 Identify the Problem and Study Scope of the Project 28
3.4 Project Planning 28
3.5 Data Collection 28
3.6 Flow Chart for Simulation 29
3.7 Simulation 30
3.7.1 MATLAB 30

3.7.2 Adds White Gaussion Noise (AWGN) Channel 32
3.8 Testing and Evaluation 32

CHAPTER 4: RESULTS AND DISCUSSION

4.1 Performance Analysis of Turbo Codes 33
4.2 Simulation Result 33
4.2.1 Effect Number of Iteration Used 33
4.2.2 Effect the Number of Iteration for Different Frame Size 38
4.2.3 Effect the Number of Iteration for Different Code Generator 45
4.3 Discussion 52

CHAPTER 5: CONCLUSION AND FUTURE WORK

5.1 Conclusion 54
5.2 Future Work 54

REFERENCES 56

APPENDICES 58

LIST OF TABLES

NO	TITLE	PAGE
3.7.1	Format conversion specifiers for fprintf	31
3.7.2	Format flag	31
4.2.1	Minimum value of BER for Log-MAP algorithms for 8 iteration	34
4.2.2	Minimum value of BER for SOVA algorithms for 8 iteration	35
4.2.3	Minimum value of BER for Log-MAP algorithms for 5 iteration	36
4.2.4	Minimum value of BER for SOVA algorithms for 5 iteration	37
4.2.5	Minimum value of BER for Log-MAP algorithms for FS = 40	39
4.2.6	Minimum value of BER for SOVA algorithms for FS = 40	40
4.2.7	Minimum value of BER for Log-MAP algorithms for FS = 200	41
4.2.8	Minimum value of BER for SOVA algorithms for FS = 200	42
4.2.9	Minimum value of BER for Log-MAP algorithms for FS = 500	44
4.2.10	Minimum value of BER for SOVA algorithms for FS = 500	45
4.2.11	Minimum value of BER for Log-MAP algorithms for g = 7, 5	46
4.2.12	Minimum value of BER for SOVA algorithms	

	for g = 7, 5	47
4.2.13	Minimum value of BER for Log-MAP algorithms for g = 5, 7	49
4.2.14	Minimum value of BER for SOVA algorithms for g = 5, 7	49
4.2.15	Minimum value of BER for Log-MAP algorithms for g = 15, 17	51
4.2.16	Minimum value of BER for SOVA algorithms for g = 15, 17	51

LIST OF FIGURES

NO	TITLE	PAGE
2.2.1	Block Diagram of Turbo Codes	8
2.4.1	BER vs Eb/No	12
2.5.1	Turbo Decoder	14
2.6.1	System model for SOVA Derivation	16
2.6.2	SOVA component decoder used in Turbo Codes	17
2.9.1	Operation of Interleaver and de-interleaver	21
3.7.1	The Component of Typical Format Specifier	30
4.2.1	Iterative turbo decoding using the Log-MAP Algorithm with 8 iteration and 15 terminate frame error	34
4.2.2	Iterative turbo decoding using the SOVA Algorithm with 8 iteration and 15 terminate frame error	35
4.2.3	Iterative turbo decoding using the Log-MAP Algorithm with 5 iteration and 15 terminate frame error	36
4.2.4	Iterative turbo decoding using the SOVA Algorithm with 5 iteration and 15 terminate frame error	37
4.2.5	Iterative turbo decoding using the Log-MAP Algorithm with 10 terminate frame error with frame size is 40	38
4.2.6	Iterative turbo decoding using the SOVA Algorithm with 10 terminate frame error with frame size is 40	39
4.2.7	Iterative turbo decoding using the Log-MAP Algorithm with 10 terminate frame error with frame size is 200	41
4.2.8	Iterative turbo decoding using the SOVA Algorithm with 10 terminate frame error with frame size is 200	42
4.2.9	Iterative turbo decoding using the Log-MAP Algorithm with 10 terminate frame error with frame size is 500	43

4.2.10	Iterative turbo decoding using the SOVA Algorithm with 10 terminate frame error with frame size is 500	44
4.2.11	Iterative turbo decoding using the Log-MAP Algorithm with 5 iteration and code generator, g = [1 1 1, 1 0 1]	46
4.2.12	Iterative turbo decoding using the SOVA Algorithm with 5 iteration and code generator, g = [1 1 1, 1 0 1]	47
4.2.13	Iterative turbo decoding using the Log-MAP Algorithm with 5 iteration and code generator, g = [1 0 1, 1 1 1]	48
4.2.14	Iterative turbo decoding using the SOVA Algorithm with 5 iteration and code generator, g = [1 0 1, 1 1 1]	49
4.2.15	Iterative turbo decoding using the Log-MAP Algorithm with 5 iteration and code generator, g = [1 1 0 1, 1 1 1 1]	50
4.2.16	Iterative turbo decoding using the SOVA Algorithm with 5 iteration and code generator, g = [1 1 0 1, 1 1 1 1]	51

LIST OF ABBREVAITION

AWGN	-	Additive White Gaussion Noise
BER	-	Bit Error Rate
Eb/No	-	Bit Energy to Noise Ratio Spectral Density
LLR	-	Log-Likelihood Ratio
Log-MAP	-	Log-Maximum A Posteriori Probability
MATLAB	-	Matrix Laboratory
ML	-	Maximum Likelihood
RSC	-	Recursive Systematic Convolutional
SNR	-	Signal to Noise Ratio
SOVA	-	Soft Output Viterbi Algoritms
TC	-	Turbo Codes

CHAPTER 1

INTRODUCTION

1.1 Project Background

Over the years, there has been a tremendous growth in digital communications especially in the fields of cellular, satellite, and computer communication. In these digital communication systems, the information is represented as a sequence of binary bits. The binary bits are then modulated into analog signal waveforms and transmitted over a communication channel. Usually, the communication channel introduces noise and interference to corrupt the transmitted signal. At the receiver, the channel corrupted transmitted signal is demodulated to binary bits. The received binary information is an estimate of the transmitted binary information. Bit errors may result due to the transmission and the number of bit errors depends on the amount of noise and interference in the communication channel. Channel coding is often used in digital communication systems to protect the digital information from noise and interference and reduce the number of bit errors.

Channel coding is mostly accomplished by selectively introducing redundant bits to the transmitted information stream. These additional bits will allow detection and correction of bit errors in the received data stream and provide more reliable information transmission. The cost of using channel coding to protect the information is a reduction in data rate or an expansion in bandwidth.

In electrical engineering and digital communications, turbo codes are a class of high-performance error correction codes developed in 1993 which are finding use in deep space satellite communications and other applications where it can be achieve to maximal information transfer over a limited-bandwidth communication link in the presence of data-corrupting noise.

For this project, there are two methods of turbo codes that will use: Soft Output Viterbi Algoritms (SOVA) and Log-Maximum A Posteriori (Log-MAP) to investigate the performance in terms of bit energy to noise ratio spectral density (Eb/No). E_b/N_o is classically defined as the ratio of Energy per Bit (Eb) to the Spectral Noise Density (No). If this definition leaves you with a empty, glassy-eyed feeling, you're not alone. The definition does not give you any insight into how to measure E_b/N_o or what it's used for. Eb/No is the measure of signal to noise ratio for a digital communication system. It is measured at the input to the receiver and is used as the basic measure of how strong the signal is. Energy per bit-to-noise power density ratio is simply the ratio of the energy of a single bit to the noise power present in 1 Hz of bandwidth.

1.2 Problem Statement

In channel coding, redundancy is introduced in the information sequence in order to increase its reliability. The channel coding theorem states that even at relatively low Eb/No reliable communication can still be maintained [1]. However, the theorem cannot do anything about how to design the code that achieves such performance. All what it says is that the code should appear random. Unfortunately random codes are very difficult to decode. It needs to have some structure in the code to make the decoding feasible. These two seemingly needs to resolve: structure and randomness. A Turbo code has solved the dilemma of structure and randomness by allowing structure through concatenation and randomness through interleaving. The introduction of turbo codes has increased the interest in the coding area since these codes give most of the gain promised by the channel-coding theorem. Turbo codes make it possible to increase data rate without increasing the power of a transmission, or they can be used to decrease the amount of power used to transmit at a certain data rate.

1.3 Project Objectives

i. To investigate the performance analysis on decoding algorithms of Turbo Codes.
ii. To compare the best performance analysis using two different decoding algorithms, Log-Maximum A Posteriori (Log-MAP) and Soft Output Viterbi Algorithms (SOVA).
iii. To investigate the effect performance based on system parameters (code rate, frame size, number of iteration, frame error, punctured and unpunctured).
iv. To evaluate the performance of turbo between theoretical analysis and computer simulation by using MATLAB

1.4 Scope of Works

In this project, this Turbo Codes is designed by using Matlab software. The simulation is carried out until the best performance analysis obtained the required specifications. Finally, the comparison between two methods of Turbo Codes (Log-MAP and SOVA) is investigated.

1.5 Important of the Project

- The introduction of turbo codes has increased the interest in the coding area since these codes give most of the gain promised by the channel-coding theorem
- Turbo codes come closest to approaching the Shannon limit, the theoretical limit of maximum information transfer rate over a noisy channel.
- It possible to increase data rate without increasing the power of a transmission
- Can be used to decrease the amount of power used to transmit at a certain data rate.
- Turbo codes also can be solved the problem of stucture and randomness.

1.6 Report Structure

This report is organized in five chapters. The first chapter gives the brief introduction of Turbo Codes.

Chapter 2: Covers literature review

Chapter 3: Covers methodology

Chapter 4: Covers result and discussion

Chapter 5: Covers conclusion and future work

CHAPTER 2

LITERATURE REVIEW

This chapter will explained in detail about Turbo Codes, Bit Energy to Noise Ratio Spectral Density (Eb/No), Log-Maximum A Posteriori (Log-MAP) and Soft Output Viterbi Algorithms (SOVA).

2.1 Channel Coding

Forward-error-correcting (FEC) channel codes are commonly used to improve the energy efficiency of wireless communication systems. On the transmitter side, an FEC encoder adds redundancy to the data in the form of parity information. Then at the receiver, a FEC decoder is able to exploit the redundancy in such a way that a reasonable number of channel errors can be corrected. Because more channel errors can be tolerated with than without an FEC code, coded systems can afford to operate with a lower transmit power, transmit over longer distances, tolerate more interference, use smaller antennas, and transmit at a higher data rate [8].

A binary FEC encoder takes in k bits at a time and produces an output (or code word)of n bits, where n > k.While there are 2n possible sequences of n bits, only a small subset of them, 2k to be exact, will be valid code words. The ratio k/n is called the code rate and is denoted by r. Lower rate codes, characterized by small values of r, can generally correct more channel errors than higher rate codes and are thus more energy efficient. However, higher rate codes are more bandwidth efficient than lower rate codes because the amount of overhead (in the form of parity bits) is lower. Thus the selection of the code rate involves a tradeoff between energy efficiency and bandwidth efficiency. For every combination of code rate (r), codeword length (n), modulation format, channel type, and received noise power, there is a theoretical lower limit on the amount of energy that must be expended to convey one bit of information. This limit is called the channel capacity or Shannon capacity, named after Claude Shannon, whose 1948 derivation of channel capacity is considered to have started the applied mathematical field that has come to be known as information theory [8].

Since the dawn of information theory, engineers and mathematician shave tried to construct codes that achieve performance close to Shannon capacity. Although each new generation of FEC code would perform incrementally closer to the Shannon capacity than the previous generation, as recently as the early 1990s the gap between theory and practice for binary modulation was still about 3 dB in the most benign channels, those dominated by additive white Gaussian noise (AWGN). In other words, the practical codes found in cell phones, satellite systems, and other applications required about twice as much energy (for example: 3 dB more) as the theoretical minimum amount predicted by information theory [8].

2.2 Turbo Codes

Turbo Codes are generated by using the parallel concatenation of two simple convolutional codes, with one coder preceded by an interleaver. The interleaver ensures that the error-prone words received for one corresponds to error-resistant words received for the other code [13].

It is theoretically possible to approach the Shannon limit by using a block code with large block length or a convolutional code with a large constraint length. The processing power required to decode such long codes makes this approach impractical. Turbo codes overcome this limitation by using recursive coders and iterative soft decoders. The recursive coder makes convolutional codes with short constraint length appear to be block codes with a large block length, and the iterative soft decoder progressively improves the estimate of the received message [4].

According to Shannon, the ultimate code would be one where a message is sent infinite times, each time shuffled randomly. The receiver has infinite versions of the message albeit corrupted randomly. From these copies, the decoder would be able to decode with near error-free probability the message sent. This is the theory of an ultimate code, the one that can correct all errors for a virtually signal [4].

The concept behind turbo decoding is to pass soft information from the output of one decoder to the input of the succeeding one, and to iterate this process several times to produce better decisions. Turbo codes are still in the process of standardization but future applications will include mobile communication systems, deep space communications, telemetry and multimedia.

The decoder works in an iterative way. Figure 2.2.1 shows a block diagram of a turbo decoder. The iteration stage is shown with doted lines to differentiate it from the initialization stage. Only one loop is performed at a time. In practice the number of iterations does not exceed 18, and in many cases 6 iterations can provide satisfactory performance. Actually, the term turbo codes are given for this iterative decoder scheme with reference to the turbo engine principle. The firs decoder will decode the sequence and then pass the hard decision together with a reliability estimate of this decision to the next decoder. Now, the second decoder will have extra information for the decoding; a priori value together with the sequence the interleaver in-between is responsible for making the two decisions uncorrelated and the channel between the two decoders will seem to be memory less due to interleaving. The exact procedure in what information to pass to the next decoder or next iteration stage is a subject of research. In the next section, we describe a widely accepted decoding algorithm, which is the modified soft output Viterbi algorithm.

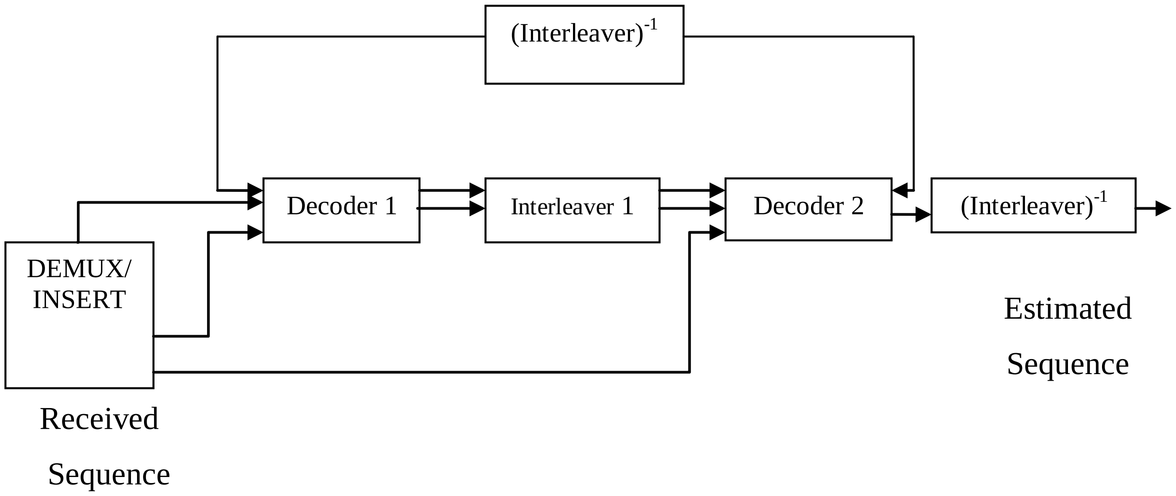

Figure 2.2.1: Block Diagram for Turbo Codes [1]

The decoder works in an iterative way. The first decoder will decode the sequence and pass the hard decision together with a reliability estimate of this decision to the next decoder after proper interleaving. The second decoder will utilize the reliability estimates produced by the first encoder, and thus will have extra information for decoding. After a certain number of iterations is executed, a hard decision is made and the estimated sequence is delivered to the user. Several decoding algorithms have bee n proposed. In this work, we opt for the Normalized SOVA (**S**oft **O**utput **V**iterbi **A**lgorithm) decoder of [10]. It was shown that SOVA approaches maximum likelihood (ML) decoding for large number of decoding iterations [3]. The interleaver is the most critical part in the design of a turbo code. he original paper on turbo codes [7] assumed a random interleaver.

Conventionally, interleaving is used to spread out the errors occurring in bursts like those exhibited in fading channels [9]. In this regard, matrix interleaving, where bits are fed in a matrix row by row and read out column by column, is usually implemented. For turbo codes, the interleaver does other functions. At the encoder, the interleaver is used to feed the encoders with different permutations of the information sequence so that the generated parity sequences can be assumed independent. At the decoder, both the interleaver and the deinterleaver are used to randomize the symbols after every decoding step, thus making the iterative decoding more efficient.