

INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.

**ProQuest Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
800-521-0600**

UMI[®]



Université d'Ottawa • University of Ottawa

TURBO CODED MODULATION IN ADSL DMT SYSTEMS

by

Liang Zhang

**Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of
Doctor of Philosophy**

Ottawa-Carleton Institute for Electrical and Computer Engineering

School of Information Technology and Engineering

University of Ottawa

© Liang Zhang, Ottawa, Canada, 2002



**National Library
of Canada**

**Acquisitions and
Bibliographic Services**

**395 Wellington Street
Ottawa ON K1A 0N4
Canada**

**Bibliothèque nationale
du Canada**

**Acquisitions et
services bibliographiques**

**395, rue Wellington
Ottawa ON K1A 0N4
Canada**

Your file Votre référence

Our file Notre référence

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-76471-0

Canada

Abstract

Asymmetric digital subscriber line (ADSL) technology provides transport of high-bit-rate digital information over digital subscriber lines (i.e., common telephone lines), which were initially engineered to carry a single voice signal with a 3.4 kHz bandwidth. Sophisticated digital transmission techniques have to be utilized to overcome digital subscriber line transmission impairments due to the high signal attenuation, inter-symbol interference (ISI), crosstalk noise (XT) from the signals present on adjacent wires, signal reflections, radio-frequency noise, and impulse noise. In current ADSL standard, a multicarrier modulation technique named discrete multitone (DMT) is used to deal with the impairments caused by heavy ISI and signal reflections. A concatenated channel coding structure is employed to efficiently increase the transmission capability in the presence of strong additive crosstalk noise. This channel coding scheme consists of an inner 4-Dimensional Trellis Coded Modulation (4D-TCM) and an outer Reed-Solomon (RS) code separated by an interleaver. Even with this powerful concatenated channel coding structure, there is still a gap of several dB between the transmission performance and the channel capacity. Before the introduction of turbo codes, reducing this gap required the employment of much more complex channel coding schemes, leading to impractical decoding complexity. Turbo codes, introduced in 1993, were shown to

provide transmission performance approaching channel capacity with practical decoding complexity. In this thesis, we investigated the potential transmission performance improvement by using Turbo code in ADSL DMT systems. To achieve this, we first proposed a turbo coding structure that fits well in the current ADSL DMT system architecture. This structure is very similar to the existing concatenated channel coding structure, except a bandwidth efficient Turbo Trellis Coded Modulation (TTCM) is used instead of the 4D-TCM. To obtain high bandwidth efficiency, only part of the bits are turbo coded and the uncoded bits are protected by a constellation mapping similar to Ungerboeck's set partitioning in TCM. The proposed TTCM was shown to provide extremely good performance when a Symbol Maximum a posteriori (SMAP) decoding algorithm is employed. In spite of the good performance of this SMAP algorithm, we proposed to use a simplified TTCM decoding structure to avoid the excessive decoding complexity introduced by the possibly large number of uncoded bits. The proposed turbo coded ADSL DMT system was simulated to evaluate the achievable transmission performance in a stationary noise environment and its performance was compared to that obtained by using the 4D-TCM. Simulation results showed significant performance improvements by using turbo codes. Furthermore, only negligible performance degradation was observed when using our simplified decoding structure instead of the optimal decoding with SMAP algorithm.

Next, the performance of turbo coding against non-stationary impulse noise was investigated. We demonstrated, by simulation, that even very short impulses could cause disastrous effect in a turbo-coded communication system. This is mainly caused by the very random reliability information of the information bits and associated parity bits hit by impulse noise. Erasure turbo decoding was proposed to improve the turbo coded system performance in the presence of impulse noise. With perfect knowledge of the impulse locations, i.e., erasure locations, simulation results showed that the decoding performance is significantly improved. Furthermore, unlike a random turbo decoding error, where erroneous bits are spread all over the whole turbo block, erasure turbo decoding is very likely to result in short error bursts, where all decoding errors fall into only the erasure area. In a concatenated channel coding structure where an RS code is used as the outer code, this results in a significant decrease of the interleave depth

required by the RS code and therefore the end-to-end transmission delay. Applications of the erasure turbo decoding to single carrier binary phase shift keying (BPSK) modulated and quadrature amplitude modulated (QAM) transmissions showed very impressive advantages in both coding gain performance and system delay reduction. We then applied this erasure turbo decoding to turbo coded ADSL DMT systems. Although insignificant improvement was observed in terms of the coding gain performance by employing erasure turbo decoding, a significant reduction in the system delay was obtained.

Acknowledgements

First of all, I would like to express my sincere appreciation to my thesis supervisor, Dr. Abbas Yongacoglu, for his constant guidance, support and encouragement throughout the course of this project. His patience and rich knowledge make it a pleasure to have discussions with him. Besides being a very good supervisor, he has always been a good friend and has helped a lot in areas other than research.

I would also like to thank Dr. Jean-Yves Chouinard, Dr. Tet Hin Yeap, Dr. Ian D. Marsland (from Carleton University) and Dr. Weihua Zhuang (from University of Waterloo) for their valuable inputs as part of my supervisory committee.

I am grateful to my colleague, Richard Boudreau, in the Advanced Audio Systems group at the Communications Research Center, for his generous help on the technical writing of this thesis, spending so much time in reviewing this thesis and the countless valuable suggestions.

Finally I would like to thank my wife for her consistent support and encouragement during my Ph.D. program.

To my wife Yanhui and my parents.

Table of Contents

ABSTRACT	II
ACKNOWLEDGEMENTS	V
TABLE OF CONTENTS	VII
LIST OF FIGURES.....	X
LIST OF TABLES.....	XIII
ABBREVIATIONS.....	XV
LIST OF SYMBOLS.....	XVIII
CHAPTER 1.....	1
INTRODUCTION	1
1.1. INTRODUCTION.....	1
1.2. THESIS OUTLINE.....	4
1.3. RESEARCH MOTIVATIONS	5
CHAPTER 2.....	7
ADSL DMT SYSTEMS.....	7
2.1. ADSL DMT SYSTEMS	7
2.2. CHANNEL MODELS AND IMPAIRMENTS IN ADSL TRANSMISSIONS	12
2.2.1. <i>Subscriber Loop Model</i>	12
2.2.2. <i>Stationary additive noises</i>	13
2.2.3. <i>Impulse Noise</i>	14
2.3. SUMMARY	14

CHAPTER 3.....	16
BANDWIDTH-EFFICIENT TURBO CODED MODULATION	16
3.1. SHANNON LIMIT	16
3.2. PARALLEL CONCATENATED CODES – TURBO CODES.....	19
3.2.1. Parallel Concatenated Codes	20
3.2.2. Iterative Decoding	22
3.3. BANDWIDTH EFFICIENT TURBO CODED MODULATION	24
3.3.1. Benedetto's TTCM encoding structure	27
3.3.2. TTCM decoding structure.....	29
3.4. SUMMARY	29
CHAPTER 4.....	32
TURBO CODES IN ADSL DMT SYSTEMS	32
4.1. TTCM IN ADSL DMT SYSTEMS	33
4.1.1. TTCM encoding structure.....	33
4.1.2. TTCM decoding structure.....	37
4.1.3. Generalized pragmatic TTCM	46
4.2. PERFORMANCE ANALYSIS OF THE PROPOSED TTCM.....	47
4.2.1. Gray mapping and the ESBS distribution.....	47
4.2.2. BER upper bound with an ML decoder.....	53
4.2.3. Performance estimation of TTCM based on binary turbo code performance.....	61
4.2.4. Performance improvement with a specific interleaver	64
4.2.5. Performance balancing between turbo-coded bits and uncoded bits	65
4.2.6. Design of Multilevel Turbo Coded Modulations.....	68
4.3. SIMULATION RESULTS OF TTCM IN QAM TRANSMISSIONS AND DMT SYSTEMS.....	77
4.3.1. TTCM with Symbol-MAP algorithm in AWGN channels.....	78
4.3.2. Performance of using concatenated coding structure	81
4.3.3. RS code requirements in concatenated coding scheme.....	89
4.3.4. Employing erasure RS decoding technique	93
4.4. PERFORMANCE EVALUATION IN PRACTICAL ADSL DMT SYSTEMS	93
4.4.1. Turbo coded ADSL DMT system with crosstalk noises	94
4.4.2. An ideal case of TTCM	96
4.5. FURTHER PERFORMANCE IMPROVEMENT	98
4.6. SUMMARY	100
CHAPTER 5.....	104
TURBO CODES AGAINST IMPULSE NOISE.....	104
5.1. IMPULSE NOISE IN SUBSCRIBER LOOPS	105
5.2. IMPULSE NOISE MODELING	106

5.2.1.	<i>Peak voltage distribution and sample amplitude distribution</i>	<i>107</i>
5.2.2.	<i>Inter-arrival duration</i>	<i>108</i>
5.2.3.	<i>Length of impulse noise</i>	<i>109</i>
5.2.4.	<i>Simulation of impulse noise</i>	<i>109</i>
5.3.	EFFECT OF IMPULSE NOISE TO TURBO DECODING.....	115
5.3.1.	<i>Effect of impulse noise in turbo coded single carrier BPSK systems.....</i>	<i>116</i>
5.3.2.	<i>Effect of impulse noise in turbo coded single carrier QAM systems.....</i>	<i>119</i>
5.3.3.	<i>Effect of impulse noise in turbo coded ADSL DMT systems</i>	<i>120</i>
5.4.	SIMULATIONS ON EFFECT OF IMPULSE NOISE.....	125
5.4.1.	<i>Turbo-coded BPSK transmissions in impulse noise environment.....</i>	<i>125</i>
5.4.2.	<i>Erasur turbo decoding against impulse noise.....</i>	<i>131</i>
5.4.3.	<i>LLR patterns</i>	<i>133</i>
5.4.4.	<i>Error pattern from binary turbo decoder using erasure turbo decoding.....</i>	<i>135</i>
5.5.	IMPULSE NOISE IN HIGH SPEED QAM TRANSMISSION WITH TTCM.....	140
5.5.1.	<i>Performance of TTCM in impulse noise environment</i>	<i>141</i>
5.5.2.	<i>Performance improvement by employing an additional interleaver.....</i>	<i>142</i>
5.5.3.	<i>Utilizing erasure turbo decoding</i>	<i>143</i>
5.5.4.	<i>RS-TTCM concatenated coding</i>	<i>149</i>
5.5.5.	<i>Simulation results using the Markov impulse noise model</i>	<i>151</i>
5.6.	TURBO CODED ADSL TRANSMISSIONS.....	153
5.6.1.	<i>Effect of impulse noise on single carrier ADSL transmissions</i>	<i>155</i>
5.6.2.	<i>Effect of impulse noise in ADSL DMT systems</i>	<i>157</i>
5.6.3.	<i>Discussion.....</i>	<i>163</i>
5.7.	SUMMARY	164
	CHAPTER 6.....	166
	CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE WORK	166
	APPENDIX A. TURBO CODE COMPONENTS.....	173
	APPENDIX B. SISO ALGORITHMS.....	178
	REFERENCES	186

List Of Figures

FIGURE 2-1 – A GENERIC MULTICARRIER TRANSMITTER.....	8
FIGURE 2-2 – ADSL DMT SYSTEM MODEL	10
FIGURE 2-3 – PROPAGATION LOSS MEASUREMENT OF 5 CSA TEST LOOPS	13
FIGURE 2-4 – NEAR-END CROSSTALK NOISE AND FAR-END CROSSTALK NOISE	14
FIGURE 3-1 – CHANNEL CAPACITY VS. MINIMUM γ_b OF BAND-LIMITED AWGN CHANNELS	18
FIGURE 3-2 – CHANNEL CAPACITY FOR DIFFERENT QAM TRANSMISSIONS	19
FIGURE 3-3 – A STANDARD TURBO ENCODER WITH TWO IDENTICAL RSCCs WITH AN INTERLEAVER.....	21
FIGURE 3-4 – A CONVENTIONAL ITERATIVE DECODING SCHEME FOR A TURBO CODE WITH TWO CCs.....	23
FIGURE 3-5 – THE ENCODER OF THE ROBERTSON'S TTCM.....	26
FIGURE 3-6 – BENEDETTO'S TURBO TRELLIS CODED MODULATION ENCODING STRUCTURE.....	28
FIGURE 3-7 – ITERATIVE (TURBO) DECODER STRUCTURE FOR THE TTCM SHOWN IN FIGURE 3-6.....	30
FIGURE 4-1 – ADSL DMT SYSTEM CHANNEL CODING CONFIGURATION	34
FIGURE 4-2 – BITS-TO-SYMBOL MAPPING OF THE PROPOSED TTCM IN ADSL DMT SYSTEMS	35
FIGURE 4-3 – CONSTELLATION MAPPING OF 64QAM IN THE PROPOSED TTCM	36
FIGURE 4-4– SIMPLIFIED TTCM DECODING STRUCTURE	41
FIGURE 4-5 – BIT MAPPING OF THE IN-PHASE COMPONENT OF A 64QAM CONSTELLATION	42
FIGURE 4-6 – MAPPING OF BITS	44
FIGURE 4-7 – EXAMPLE CONSTELLATION MAPPING OF A PRAGMATIC TCM FOR 16PAM	46
FIGURE 4-8 – RATE $\frac{1}{2}$ TURBO ENCODER WITH PUNCTURING.....	47
FIGURE 4-9 – PDF OF ESBS OF U_I , TTCM WITH 64QAM	48

FIGURE 4-10 – PDF OF THE ESBS OF U_i FOR VERY LARGE QAM CONSTELLATIONS	51
FIGURE 4-11 – 3-BIT GRAY MAPPING OF 256QAM TTCM ALONG ONE DIMENSION	52
FIGURE 4-12 – BER UPPER BOUND OF THE TURBO-CODED BITS IN A TTCM	56
FIGURE 4-13 – BER UPPER BOUND OF THE TURBO-CODED BITS IN A 64QAM TTCM.....	57
FIGURE 4-14 – BER UPPER BOUND OF THE TURBO-CODED BITS IN A TTCM.....	59
FIGURE 4-15 – BER UPPER BOUND OF THE TURBO-CODED BITS IN A TTCM.....	60
FIGURE 4-16 – WORD ERROR PROBABILITY OF ML DECODING WITH ESBSS	64
FIGURE 4-17 – COMPREHENSIVE ENCODING STRUCTURE FOR AN ADSL DMT SYSTEM	68
FIGURE 4-18 – CAPACITIES OF 64QAM AND THE LOWER LEVEL CODED BITS	72
FIGURE 4-19 – 3-BIT GRAY MAPPING OF IN-PHASE OR QUADRATURE COMPONENT OF A 256QAM	75
FIGURE 4-20 – LLR DISTRIBUTION OF THE FIRST BIT IN 3-BIT GRAY MAPPING IN THE 256QAM AS SHOWN IN FIGURE 4-19.....	76
FIGURE 4-21 – BER OF 64QAM TTCM WITH SYMBOL-MAP DECODING ALOGRITHM	79
FIGURE 4-22 – BER OF TTCM WITH SIMPLIFIED TTCM DECODING	80
FIGURE 4-23 – RS-TTCM BER PERFORMANCE WITHOUT CONSIDERING SNR PENALTIES	83
FIGURE 4-24 – RS-TTCM BER PERFORMANCE CONSIDERING SNR PENALTIES	84
FIGURE 4-25 – RS-TTCM BER PERFORMANCE WITHOUT CONSIDERING SNR PENALTIES	85
FIGURE 4-26 – RS-TTCM BER PERFORMANCE CONSIDERING SNR PENALTIES	85
FIGURE 4-27 – BER OF TTCM WITH SIMPLIFIED DECODING, 400-BIT INTERLEAVERS.....	86
FIGURE 4-28 – BER OF TTCM WITH SIMPLIFIED DECODING, 400-BIT INTERLEAVERS.....	87
FIGURE 4-29 – BER COMPARISON BETWEEN CHANNEL CODING SCHEMES IN ADSL DMT SYSTEMS.....	88
FIGURE 4-30 – SERIAL-PARALLEL-CONCATENATED-RS-TTCM CODING SCHEME	91
FIGURE 4-31 – BER PERFORMANCE OF LOWER BRANCH OF SPC-RS-TTCM.....	92
FIGURE 4-32 – EFFECT OF USING BLOCK SWITCHING TO IMPROVE THE SYSTEM PERFORMANCE.....	96
FIGURE 4-33 – PERFORMANCE OF BENEDETTO'S 16QAM TTCM WITH A RATE $\frac{1}{2}$ TURBO CODE.....	97
FIGURE 4-34 – PERFORMANCE OF 64QAM TTCM WITH RATE $\frac{1}{3}$ TURBO CODE.....	100
FIGURE 5-1 – TWO-STATE MARKOV CHANNEL MODEL	111
FIGURE 5-2 – FOUR CASES OF N SEPARATED IMPULSES IN ONE DMT BLOCK.....	112
FIGURE 5-3 – IMPULSE LENGTH DISTRIBUTION, $P_A=10^{-7}$ WITH $P_B=0.01$, AND $P_A=10^{-4}$ WITH $P_B=0.1$, $N_c=512$	115
FIGURE 5-4 – CROSS-CHANNEL-CORRELATION OF IMPULSE NOISE CONTRIBUTIONS IN DMT SYSTEMS	123
FIGURE 5-5– PERFORMANCE OF TURBO CODES AGAINST IMPULSE NOISE VARIANCE	127
FIGURE 5-6 – TURBO CODE PERFORMANCE IN THE PRESENCE OF IMPULSES VS. BACKGROUND SNR.....	129
FIGURE 5-7 – LLR PATTERN RESULTED FROM A SUCCESSFUL TURBO DECODING.....	134
FIGURE 5-8 – TYPICAL LLR PATTERN RESULTED FROM AN ERROR TURBO DECODING	134
FIGURE 5-9 – TYPICAL LLR PATTERN RESULTED FROM AN ERROR TURBO DECODING	135

FIGURE 5-10 – TYPICAL LLR PATTERN AND ERROR PATTERN RESULTED FROM ERASURE TURBO DECODING	136
FIGURE 5-11 – TYPICAL LLR PATTERN AND ERROR PATTERN OF ERASURE TURBO DECODING $L_{imp}=200$	137
FIGURE 5-12 – NUMBER OF ERROR BYTES RESULTED FROM ERROR TURBO DECODING	138
FIGURE 5-13 – PROBABILITY DISTRIBUTION OF THE NUMBER OF ERROR BYTES PER TURBO BLOCK	139
FIGURE 5-14 – PROBABILITY DISTRIBUTION OF ERROR BYTE NUMBER PER TURBO BLOCK	139
FIGURE 5-15 – BER PERFORMANCE OF 64QAM TTCM IN IMPULSE NOISE ENVIRONMENTS	141
FIGURE 5-16 – TYPICAL ERROR PATTERN OF QAM TTCM USING ERASURE TURBO DECODING	147
FIGURE 5-17 – CUMULATIVE DISTRIBUTION OF THE NUMBER OF ERROR BYTES IN ONE TURBO BLOCK	148
FIGURE 5-18 – PROBABILITY DISTRIBUTION OF THE NUMBER OF ERROR BYTES IN ONE TURBO BLOCK	149
FIGURE 5-19 -- IMPULSE LENGTH DISTRIBUTION, ONE IMPULSE PER 10 SECONDS	153
FIGURE 5-20 – PROBABILITY DENSITY FUNCTION OF THE DISTRIBUTION OF $VAR\{I_k\}$	161
FIGURE 5-21 – CUMULATIVE DISTRIBUTION FUNCTION OF $VAR\{I_k\}$	161

List of Tables

TABLE 4-1 – PROTECTION STATES OF THE TURBO CODED BITS $[U_1, P_1]$ WITH THE CONSTELLATION MAPPING SHOWN IN FIGURE 4-5	50
TABLE 4-2 – PROTECTION STATES OF THE TURBO-CODED BITS $[C_1, C_2, C_3]$ IN EACH DIMENSION OF 256QAM, WITH THE MAPPING OF EACH DIMENSION AS SHOWN IN FIGURE 4-11	53
TABLE 4-3 – DISTANCE SPECTRUM PERFORMANCE OF THE BEST CCs	58
TABLE 4-4 – MINIMUM CODING GAINS OF TURBO-CODED BITS IN VARIOUS QAM TTCM	62
TABLE 4-5 – MINIMUM CODING GAINS OF TURBO-CODED BITS IN VARIOUS QAM TTCM	62
TABLE 4-6 – MINIMUM CODING GAINS OF TURBO-CODED BITS IN VARIOUS QAM TTCM	62
TABLE 4-7 – WEIGHT DISTRIBUTION OF A RATE-1/2 TURBO CODE, $K=100$	65
TABLE 4-8 – DEFAULT SIMULATION ASSUMPTIONS	78
TABLE 4-9 – SNR PENALTY RESULTED FROM DIFFERENT RS CODE REDUNDANCIES IN DMT SYSTEMS	83
TABLE 4-10 – INTERLEAVE DEPTH REQUIREMENTS AND SYSTEM DELAYS IN RS-TTCM	90
TABLE 4-11 – DECISION ERROR PROBABILITIES OF UNCODED BITS IN 64QAM TTCM WITH ASSUMED TRANSMITTED SIGNALS AND TURBO DECODING RESULTS	102
TABLE 5-1 – BER OF IMPULSE NOISE CORRUPTED BINARY TURBO CODE	127
TABLE 5-2 – BER PERFORMANCE WITH IMPULSE NOISE STARTING FROM DIFFERENT POSITIONS	128
TABLE 5-3 – BER PERFORMANCE OF TURBO CODE AGAINST IMPULSE NOISE VS. BACKGROUND NOISE	128
TABLE 5-4 – BER OF TURBO DECODING WITH AND WITHOUT ERASURE	132
TABLE 5-5 – BER OF TURBO DECODING WITH (W.) AND WITHOUT (W/O) ERASURE	132
TABLE 5-6 – BER PERFORMANCE OF ERASURE TURBO DECODING WITH CSI	133

TABLE 5-7 – DEFAULT SIMULATION ASSUMPTIONS	140
TABLE 5-8 – EFFECT OF ADDING AN ADDITIONAL INTERLEAVER IN 64QAM TTCM	142
TABLE 5-9 – PERFORMANCE OF 64QAM TTCM WITH ERASURE DECODING AGAINST IMPULSE NOISE	143
TABLE 5-10 – EFFECT OF ADDING AN ADDITIONAL INTERLEAVER IN 64QAM TTCM WITH ERASURE TURBO DECODING, UNDER DIFFERENT BACKGROUND NOISE CONDITIONS	144
TABLE 5-11 – PERFORMANCE OF 64QAM TTCM USING ERASURE TURBO DECODING.....	145
TABLE 5-12 – NUMBER OF ERROR BYTES PER TURBO BLOCK WITH AND W/O ERASURES	150
TABLE 5-13 – BER OF 64QAM TTCM WITH MARKOV CHANNEL MODEL	152
TABLE 5-14 – BER OF 64QAM TTCM WITH MARKOV CHANNEL MODEL	152
TABLE 5-15 – DISTRIBUTION OF IMPULSE VARIANCE, σ_i^2	154
TABLE 5-16 – TWO ADSL TRANSMISSION TEST CASES, 64QAM TTCM.....	155
TABLE 5-17 – EFFECT OF IMPULSE NOISE IN SINGLE CARRIER ADSL TRANSMISSIONS WITH TTCM.....	156
TABLE 5-18 – MARKOV MODEL PARAMETERS.....	156
TABLE 5-19 – BER PERFORMANCE ESTIMATION WITH MARKOV IMPULSE MODEL.....	157
TABLE 5-20 – EQUIVALENT IMPULSE VARIANCES	159
TABLE 5-21 – BER PERFORMANCE OF ADSL DMT TRANSMISSIONS WITH TTCM	159
TABLE 5-22 – DISTRIBUTION OF THE EQUIVALENT IMPULSE NOISE VARIANCE.....	162
TABLE 5-23 – BER PERFORMANCE OF ADSL DMT SYSTEM WITH MARKOV CHANNEL MODEL	162
TABLE 5-24 – PERFORMANCE OF 64QAM TTCM IN IMPULSE NOISE ENVIRONMENTS WITH AND WITHOUT ERASURE TURBO DECODING.....	163

Abbreviations

4D-TCM	4-Dimensional Trellis Coded Modulation
A/D	Analog-to-Digital Converter
ADSL	Asymmetric Digital Subscriber Line
ASIC	Application Specific Integrated Circuit
AWGN	Additive White Gaussian Noise
BCJR	Bahl-Cocke-Jelinek-Raviv
BE-TTCM	Bandwidth Efficient Turbo Trellis Coded Modulation
BER	Bit Error Rate
BPSK	Binary Phase Shift Keying
CAP	Carrierless Amplitude-Phase Modulation
CC	Constituent Codes
CE	Cross-Entropy
CIR	Channel Impulse Response
CSA	Carrier Service Area
CSI	Channel State Information
CO	Central Office
COFDM	Coded Orthogonal Frequency Division Multiplexing
CWEF	Conditional Weight Enumeration Function
CP	Cyclic Prefix
CSI	Channel State Information

DAB	Digital Audio Broadcasting
D/A	Digital-to-Analog Converter
DFE	Decision Feedback Equalizer
DFT	Discrete Fourier Transform
DMT	Discrete Multitone
DSL	Digital Subscriber Line
DSP	Digital Signal Processing
EFED	Effective Free Euclidean Distance
ESBS	Equivalent Soft Binary Symbol
ESD	Energy Spectrum Density
FEC	Forward Error Correction (Code)
FEXT	Far-End-Crosstalk
FFT	Fast Fourier Transform
<i>iff</i>	If and Only If
<i>i.i.d.</i>	Identical-Independently-Distributed
IN	Impulse Noise
ICI	Inter-Carrier-Interference
IDFT	Inverse Discrete Fourier Transform
IFFT	Inverse Fast Fourier Transform
ISI	Inter-Symbol-Interference
LLR	Log-Likelihood Ratio
MAP	Symbol-by-Symbol Maximum <i>a Posteriori</i>
MD-TCM	Multi-Dimensional Trellis Coded Modulation
ML	Maximum-Likelihood
MLC	Multi-Level-Coding
MSD	Multi-Stage-Decoding
Msp/s	Mega-sample-per-second
NCSO	No Channel State Information
NEXT	Near-End-Crosstalk
NRSC	Nonrecursive Systematic Convolutional Codes
PAM	Pulse-Amplitude-Modulation
PDF	Probability Density Function
PSD	Power Spectrum Density
QAM	Quadrature Amplitude Modulation
RS	Reed-Solomon
RSCC	Recursive Systematic Convolutional Code
SER	Symbol Error Rate

SISO	Soft-In/Soft-Out
SLLR	Symbol LLR
SMAP	Symbol-MAP
SNR	Signal-to-Noise Ratio
SOVA	Soft Output Viterbi Algorithm
SPC	Serial-Parallel-Concatenated
TCM	Trellis Coded Modulation
TTCM	Turbo Trellis Coded Modulation
UTP	Unshielded Twisted Pair
VA	Viterbi Algorithm
VLSI	Very Large Scale Integration
WEF	Weight Enumeration Function
W-CDMA	Wideband-Code-Division-Multiplex-Access
XT	Crosstalk Noise

List of Symbols

$a(f)$	Attenuation constant in a subscriber loop
b	Number of information bits transmitted by one multicarrier (DMT) symbol
b_{av}	Average number of bits per subchannel in an ADSL DMT system
b_k	Number of transmission bits carried by the k^{th} subchannel in a multicarrier (DMT) system
c_i	General representation of turbo coded bits
C	Normalized channel capacity, bits per channel symbol
d_{free}	Free Euclidean distance
$d_{free,eff}$	Effective free Euclidean distance
d_w	Hamming weight of convolutional codeword generated from a weight- w information word
e	Constant, 2.71828
f	Frequency
$f_{c,k}$	Center frequency of the k^{th} subchannel in a multicarrier system
E_b	Average energy per information bit
$g(t)$	Waveform filter at the transmitter
G_{neg}	SNR penalty resulted from RS code redundancy
$G(x)$	Gaussian distribution
$h(t), h(n)$	Continuous and discrete channel impulse response
I_k	Impulse noise samples in the k^{th} subchannel after DMT demodulation

$I(X;Y)$	Mutual information contained in one symbol in transmission with X as the input symbol set and Y as the output symbol set
k	Subchannel index
K	Number of information bits in one turbo block
K_c	Constant in calculating the equivalent binary BPSK symbol in simplified TTCM decoding
K_{RS}	Number of information bytes in one RS code word
K_{sc}	Number of input bits over each trellis branch of the CCs in TTCM
l	Number of impulse samples
l_m	Average number of impulse samples in one DMT symbol
$L(.)$	Log-likelihood ratio
$L_c(u_k)$	Channel information (systematic information) of bit d_k
$L_{apn}(u_k)$	<i>A priori</i> information of bit d_k
$L_{dB}(l,f)$	Propagation loss of a subscriber line of length l at frequency f
$L_e(u_k)$	<i>Extrinsic</i> information of bit d_k
L_{imp}	Length of impulse noise in number of samples
$\ln(x)$	Natural logarithm of x
M	Constellation size of channel input symbol set
n	Multicarrier symbol time axis index
$n(t)$	Continuous-time AWGN noise process
N	Length of a turbo code word in number of bits
N_c	Number of subchannels in ADSL DMT systems
N_0	Single-sided AWGN background noise power spectrum density
N_{fft}	Number of FFT points
N_{free}	Number of error paths with the free Euclidean distance
N_w	Multiplicity of the convolutional code word generated from a weight- w information word
N_{sc}	Number of output bits over each trellis branch of the CCs in TTCM
N_S	Length of a QAM symbol vector that contains a turbo code word
N_{RS}	Number of bytes in a RS code word
P_b	Bit error rate
p	Parity bits output from a turbo encoder
p_a	Transition probability from good state to bad state in a two-state Markov channel
p_b	Transition probability from bad state to good state in a two-state Markov channel
p_{apo}	<i>A posteriori</i> probability
p_{imp}	Probability of occurrence of impulse noise samples
$p(x)$	Probability density function of random variable x
P_M	Symbol error rate

P_{re}	Redundancy ratio of the information sequence introduced by RS code
$P(x)$	Distribution of random variable x
$Q(x)$	Marcum-Q function of x
$\bar{r}$	Received turbo code word, including systematic symbols and parity symbols
$r(k)$	Received bit in the k^{th} time interval
$r_s(k)$	k^{th} received systematic symbol into a soft-in/soft-out decoder
$r_p(k)$	k^{th} received parity symbol into a soft-in/soft-out decoder
$r(t), r(n)$	Continuous and discrete time domain signal with cyclic prefix (CP) at the receiver
R_b	Channel bit rate per second
R_c	Channel coding code rate
R_{DMT}	DMT symbol rate per second
R_s	Number of information bits per channel symbol
R_T	Code rate of turbo code
$R_x(n)$	A block of received bits in the n^{th} multicarrier symbol
$s(t), s(n)$	Continuous and discrete time domain signal with CP output from the transmitter
S_I	Starting point of impulse noise in one turbo coded bit/symbol block
T	Average sample duration of the ADSL DMT systems
$T_x(n)$	The block of transmission bits in the n^{th} multicarrier symbol
u	Information bit block in turbo codes
u'	Interleaved version of the information sequence fed into the 2 nd CC in a turbo encoder
$\bar{u}_l$	The systematic bit vector contained in the l^{th} QAM symbol
U	Duration of one multicarrier (DMT) symbol
v	Length of cyclic prefix
w	Hamming weight of a code word (or error word for linear channel coding)
w_{min}	Minimum hamming weight of all non-zero code words of an error correction code
W	Total available bandwidth in an DMT transmission
$x(t), x(n)$	Continuous and discrete time domain multicarrier signal at the transmitter without CP
x	In-phase or quadrature component of the transmitted symbol sequence from TTCM encoder
X	Multi-level channel input symbol
X_i	i^{th} possible channel input symbol in calculation of channel capacity
$\tilde{X}$	Set of channel input symbols in calculation of channel capacity
$X_k(n)$	Transmitted QAM symbol in the k^{th} subchannel, $k=1, \dots, N_c$ in the n^{th} multicarrier symbol interval
$X_{b,k}(n)$	Transmitted bit group allocated to the k^{th} subchannel, $k=1, \dots, N_c$ in the n^{th} multicarrier symbol interval

$\bar{X}(k)$	k^{th} complex number in the conjugate symmetric sequence that is input into the IFFT modulator
y	In-phase or quadrature component of the received QAM symbol sequence at the TTCM decoder
$y(t), y(n)$	Continuous and discrete time domain multicarrier signal at the receiver without CP
Y	Multi-level channel output symbol
$\tilde{Y}$	Set of channel output symbols in calculation of channel capacity
$Y_k(n)$	Received QAM symbol in the k^{th} subchannel, $k=1, \dots, N_c$ in the n^{th} multicarrier symbol interval
$Y_{b,k}(n)$	Recovered bit group from the k^{th} subchannel, $k=1, \dots, N_c$ in the n^{th} multicarrier symbol interval
z_{min}	Minimum weight of the parity check sequence generated by a CC in turbo codes
γ_b	Signal-to-noise ratio per information bit, defined as $\gamma_b = E_b/N_0$.
$\bar{\gamma}_b$	Minimum γ_b requirement for a certain channel capacity
$\lambda(r)$	Symbol-Log-Likelihood-Ratio of the received symbol r
π	Interleaver
π^{-1}	De-interleaver
σ^2	AWGN variance

Chapter 1.

Introduction

1.1. INTRODUCTION

Digital subscriber line (DSL) techniques have been pushed forward by the ever-increasing demands of high-speed data transmission to ordinary residential homes. Although fiber has become the preferred medium for transmission between local exchanges, the final few kilometers between the exchanges and residences consist of old-fashioned unshielded twisted-pair (UTP) copper lines¹, which lack the large bandwidth of fiber-based transmission links. However, because of the huge investment of replacing the twisted-pairs with fibers, the global telecommunication companies are seeking every way to provide high-speed data transmission over the already existing infrastructure. DSL techniques make this possible by taking advantage of advanced digital transmission techniques. A key idea in DSL techniques is that a lot of envisioned applications such as

¹ In this thesis, digital subscriber line, digital subscriber loop and UTP copper line are all referred as the same concept.

video on demand and Internet access require high-speed transmission only in the downstream direction, i.e., from central office (CO) to subscribers. Upstream transmission typically consists of short messages and control signaling and these transmissions require a much lower rate. This transmission concept together with advanced digital signal processing and very large scale integration (VLSI) technology gave birth to the Asymmetric Digital Subscriber Lines (ADSL).

ADSL is an access technology intended to provide up to 8 Mbps downstream digital transmission from CO to subscribers and up to 640 kbps upstream transmissions over the existing nonloaded subscriber loop plant. Such high-speed digital transmission over UTP requires sophisticated digital signal processing techniques in order to overcome the impairments caused by the heavily frequency-dependent attenuation. These include advanced modulation and channel coding techniques, time-domain equalization and echo cancellation, etc.

The two major modulation schemes proposed for ADSL transmission are carrierless amplitude and phase (CAP)² modulation and discrete multi-tone (DMT). Each technique has its own advantages and challenges. After a long time debate, DMT finally won the battle in the standardization process and therefore most of the current ADSL chipset implementations are based on DMT techniques. DMT is basically a multicarrier modulation technique implemented by Discrete-Fourier-Transform (DFT). Employing DMT in ADSL transmission system provides more efficient utilization of the available channel bandwidth, as well as a complexity reduction in the receiver equalizer. Furthermore, the availability of the Fast-Fourier-Transform (FFT) implementation of the DFT makes the modulation fast and very efficient in either an Application-Specific-Integration-Circuit (ASIC) design or utilizing an existing Digital-Signal-Processors (DSPs). Despite the various benefits, employing DMT also creates some challenges, particularly in the area of channel coding, such as,

- Coding for multiple channel
- Coding against long impulse noise

² CAP is a special way of implementing single-carrier QAM transmission in passband.

The key issue in an ADSL transmission is to maximize the transmission rate subject to the transmission power and bandwidth constraints. A suitable channel coding technique can be employed to increase the transmission rate given the transmission power and bit error rate (BER) requirements. Current ADSL standard is based on a concatenated channel coding scheme where an optional bandwidth efficient 4D-TCM is used as the inner code and an RS code is employed as the outer code. This concatenated channel coding structure can provide an up to 6 dB coding gain with very reasonable complexity.

Although the concatenation of an inner 4D-TCM and an outer RS code provides significant coding gain in ADSL transmissions, the system performance is still several dB from the channel capacity. This indicates that there is still potential to improve the ADSL transmission rate. There are two methods to reduce the gap between the current achievable performance and the channel capacity. The first method is to use more powerful TCM by increasing the constraint length of the convolutional code. However, this will result in impractical decoding complexity for any non-trivial coding gain improvement. Another method is to use a more efficient channel coded modulation, instead of the TCM, with feasible decoding complexity. Such a code had not been discovered until the introduction of turbo codes in 1993.

In 1993, a parallel-concatenated code named “turbo code” was introduced [1], [2], by Claude Berrou *et al.* A rate 1/2 turbo code was reported capable of providing performance within 0.7 dB from the Shannon limit at a BER of 10^{-5} . The significance of turbo code is that by using a sub-optimal iterative decoding, the near-capacity performance is achieved with feasible implementation. The near-Shannon limit performance was also achieved by combining turbo code with multi-level modulation schemes, which is hereafter termed turbo trellis coded modulation (TTCM), [3], [4]. This totally changed the previous idea of designing practical channel coding according to the cut-off rate³. People started investigating the potential gain by employing this powerful new code in various applications. Telecommunication industry is considering putting turbo codes into the international standards for some large communication systems such

³ It was shown in [5] that although information theory indicates that channel capacity is the ultimate limit on reliable throughput, pushing the rate beyond the cut-off rate proves practically difficult.

as CDMA2000, W-CDMA and ADSL transmissions. In this thesis, we investigate the potential of employing bandwidth efficient turbo codes in ADSL transmissions.

1.2. THESIS OUTLINE

In Chapter 2, we perform a detailed discussion on ADSL transmission systems. The discrete communication system model of an ADSL DMT system is presented. This is the basic system model we will be using in the simulations in this thesis. The discussion also identifies the crosstalk noise (XT) and impulse noise (IN) as the two major impairments in ADSL transmissions. A complete discussion of applying turbo code to ADSL transmissions should include the performance in the presence of both impairments.

Chapter 3 briefly presents the concepts of turbo code and TTCM. Typical turbo encoding and decoding structures and their major components are introduced. Since choosing a proper decoding algorithm is one of the most important issues in designing turbo-coded systems, various decoding algorithms are also briefly discussed in this chapter. Among the several TTCM schemes proposed in the literature [3], [4], we chose the one that Benedetto proposed in [3] and expanded it by adding uncoded bits.

In Chapter 4, the proposed bandwidth efficient TTCM scheme is incorporated into the ADSL DMT transmission structure. Performance of employing this TTCM in ADSL transmissions in stationary noise environment, i.e., AWGN and XT, was evaluated by simulations. We then proposed a simplified decoding algorithm based on employing simply a binary turbo decoder and a binary soft information calculation module. Our simulations showed very little performance degradation with a significant decoding complexity reduction by using this simplified decoding. Furthermore, since the decoding method is based on using a binary turbo decoder, various performance-improving techniques for binary turbo decoding could be applied to the proposed system. It was shown in the literature that TTCM usually provides an excellent coding gain only at a BER of 10^{-6} or higher, below which an error floor will occur. Therefore, a concatenated channel coding structure with TTCM as inner code and RS code as outer code is then considered as a better solution. Very promising coding gain performance is obtained in our simulations with this code concatenation. Some other possible implementations of concatenated coding structures are also discussed in this chapter.

The other major impairment in subscriber line transmissions, impulse noise, is considered in Chapter 5. First we gave a brief survey on major characteristics of impulse noise found in high-speed subscriber line transmissions. Based on this survey, two impulse noise models were proposed and used in our simulations. In this chapter, we proposed using erasure turbo decoding in a burst channel where the error performance is decided mainly by impulse noise. Not only does the erasure turbo decoding improve the BER performance, it also significantly reduces the interleaving requirements of the external RS code by generating much shorter error bursts. Performances of turbo coded ADSL transmissions with single-carrier and multicarrier modulation techniques in impulse noise environment are compared. With respect to the different natures of these two modulation techniques, different combinations of interleaving schemes and erasure decoding techniques are proposed to combat the performance degradation caused by impulse noise.

1.3. RESEARCH MOTIVATIONS

In this thesis, we tried to answer many questions related with the application of turbo codes to ADSL systems. Some of the major questions are summarized as follows:

- What are the existing turbo trellis coded modulation (TTCM) schemes? Which scheme is the most suitable one for ADSL applications, taking into account the performance, the implementation complexity and the flexibility of being applied to a multicarrier system?
- How can we efficiently incorporate the TTCM into the existing ADSL DMT systems? What are the major structure modifications that need to be made to the current ADSL DMT system architecture specified by the ADSL standard in order to utilize the TTCM?
- When using a TTCM (non-binary turbo coding) scheme, can we still use a standard binary turbo decoder? If so, what are the necessary modifications to the TTCM decoder structure? What is the performance penalty, if there is any?
- What is the theoretical performance (or performance bound) of the proposed TTCM scheme in ADSL transmissions? How can we estimate the asymptotic performance of the proposed TTCM scheme?

- What is the design criteria of the turbo codes used in the proposed TTCM scheme?
- What performance can be obtained by simulations?
- When the performance of using the TTCM cannot meet the system requirements, what are the alternatives? What are the associated performance and implementation issues?
- What are the major statistical characteristics of the impulse noise in subscriber line transmissions? For the purpose of performance evaluation, how do we model them in our simulations taking into account both the programming simplicity and the modeling accuracy?
- What is the theoretical performance (or performance bound) of turbo coded systems in the presence of impulse noise?
- What is the effect of impulse noise to turbo decoding?
- What is the effect of impulse noise to turbo coded ADSL DMT systems?
- What techniques can be used to deal with impulse noise in a turbo-coded transmission? What are the requirements and limitations of these techniques?
- What is the performance of a practical ADSL DMT system with our proposed TTCM in the presence of impulse noise? Will the proposed TTCM remove the effect of impulse noise totally?

The answers to these questions will be given throughout this thesis and the original contributions achieved in due process are summarized in Chapter 6.

Chapter 2.

ADSL DMT Systems

In this chapter, the basic structure of an ADSL DMT system is briefly described. It is shown that a high-speed ADSL DMT transmission can be modeled as a set of parallel independent low-speed frequency-division-multiplexed quadrature-amplitude-modulation (QAM) transmissions. This equivalent parallel channel model and the major impairments discussed in this chapter are the basic assumptions for the simulations in this thesis.

2.1. ADSL DMT SYSTEMS

A typical multicarrier system is shown in Figure 2-1. In such a system, the total available bandwidth, denoted as W , is divided into N_c subchannels, each having a bandwidth of W/N_c . These N_c subchannels will be independently modulated and frequency-multiplexed into one multicarrier symbol before they are transmitted into the channel. Since each subchannel has a bandwidth of W/N_c , the multicarrier symbol duration, denoted as U , will be close, if not equal, to N_c/W . Assuming each multicarrier

symbol carries a total of b bits, and let b_k be the number of bits in one symbol duration

transmitted by the k^{th} subcarrier centered at $f_{c,k}$, then, $b = \sum_{k=1}^{N_c} b_k$.

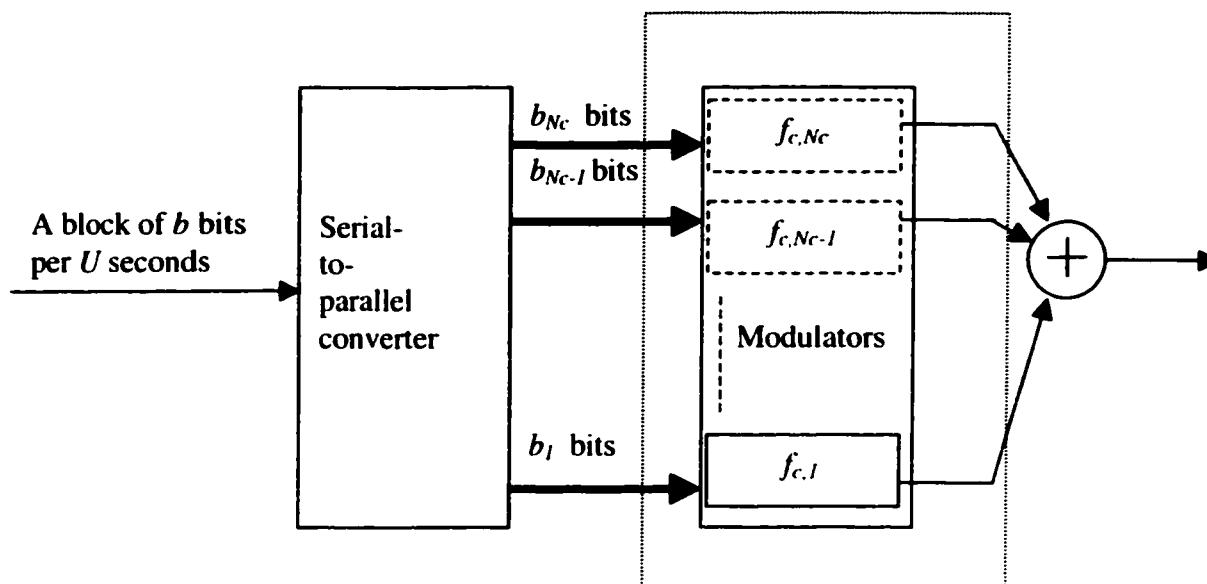


Figure 2-1 – A generic multicarrier transmitter

Compared to a single carrier system, the major advantages of a multicarrier system include [6], [7]:

- With relatively narrow-band subchannels, the problem of bandwidth optimization is greatly simplified, and this further leads to better utilization of the available bandwidth.
- In single carrier systems, linear equalization is employed to reduce the inter-symbol-interference (ISI). The equalizer causes noise and/or interference enhancement at null (or nearly null) points of the channel frequency response. A multicarrier system can avoid this problem simply by not using subchannels containing those null points.
- In a multicarrier system as shown in Figure 2-1, the total bandwidth W is divided into N_c subchannels, each with a bandwidth of W/N_c . For a single carrier transmission with this bandwidth, the symbol duration is around $1/W$, whereas the symbol duration in each subchannel in the multicarrier system is around $U = N_c/W$.

The longer symbol duration gives a much greater immunity to impulse noise and fast fades. This advantage becomes more apparent as the number of subchannels increases.

Traditional multicarrier systems perform subchannel modulations independently and separate the modulated signals by filters [6], where the low bandwidth efficiency and strict filtering requirement make such transceiver implementations impractical. In [8], Weinstein and Ebert first used Discrete Fourier Transform (DFT) to perform baseband modulation and demodulation. The efforts were then concentrated on efficient digital signal processing techniques, with which the requirement for expensive analog filter bank is eliminated. Later, Peled and Ruiz totally removed (theoretically) the inter-symbol-interference (ISI) and inter-channel-interference (ICI) by the introduction of a cyclic prefix (CP) [9]. A DMT system is basically a multicarrier system employing DFT to perform frequency-division-multiplexed multicarrier (QAM) modulation and using a CP to eliminate the ISI/ICI. One of the major advantages of DMT systems is that modulation and demodulation can be performed using Fast Fourier Transform (FFT) techniques to greatly reduce the transceiver complexity.

A basic ADSL DMT transmission system model is shown in Figure 2-2. Like the generic multicarrier system shown in Figure 2-1, a total of N_c subchannels are assumed. A thick arrow denotes a vector of N_c symbols.

As shown in Figure 2-2, the processing of the information bits is performed on a block-by-block basis. Let's still assume that there are b bits in each block. These b bits are allocated to the N_c sub-channels by a bit-allocation algorithm, which decides the optimal power and bit distribution among all the subchannels based on their signal-to-noise ratio (SNR) values. As shown in Figure 2-2, for the J^{th} block, the b bits are divided into N_c bit groups, $X'_b(n) = \{X'_{b,1}(n), X'_{b,2}(n), \dots, X'_{b,N_c}(n)\}$, where $X'_{b,k}(n)$ is the bit group allocated to the k^{th} subchannel. To optimize the channel utilization, the number of bits carried by different subchannels could be different. The Encoder and Mapper translates these bit groups into QAM symbols, $X = \{X(1), X(2), \dots, X(N_c)\}$. Since the signal processing on each block is the same, in the following discussions, we will ignore the block index J .

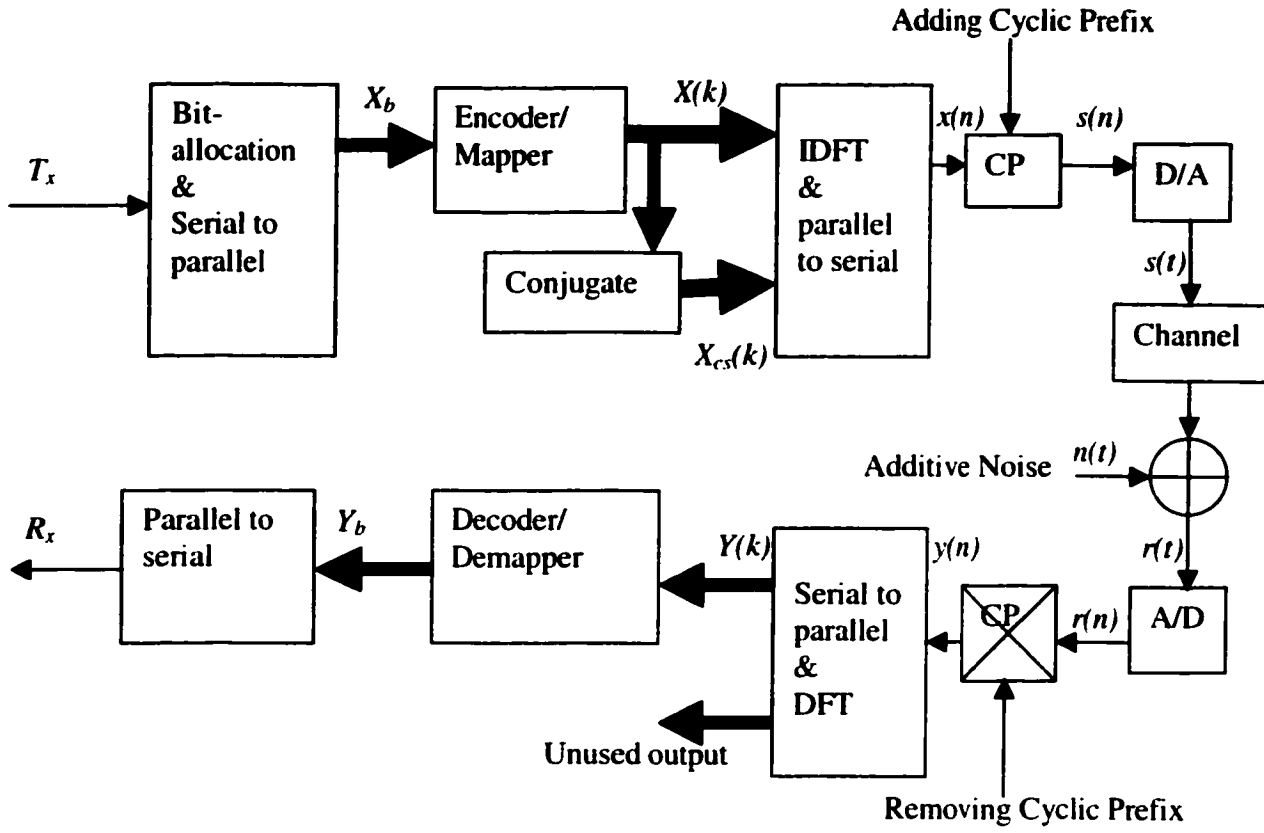


Figure 2-2 – ADSL DMT system model

The modulation is performed by an Inverse-Discrete-Fourier-Transform (IDFT) on this complex vector. Since ADSL DMT transmissions use the bandwidth from DC to a maximum frequency W , it is overall a baseband transmission. This requires the output of the IDFT operator be real values. This is guaranteed by generating a conjugate-symmetric complex sequence $\bar{X}$ of length $2N_c$, as:

$$\bar{X}(k) = \begin{cases} X(k), & 0 \leq k \leq N_c - 1 \\ X^*(2N_c - k), & N_c \leq k \leq 2N_c - 1 \end{cases}$$

Eq. 2-1

with $\bar{X}(0)$ and $\bar{X}(N_c)$ being real. The $X_{cs}(k)$ in Figure 2-2 is the additional conjugate-symmetric vector, where $X_{cs}(k) = X^*(N_c - k)$.

The conjugate symmetric sequence, $\bar{X}$, is then input into a length- $2N_c$ IDFT and the output sequence, after the parallel-to-serial conversion, $x(n)$, is given by,

$$x(n) = \frac{1}{N_c} \sum_{k=0}^{2N_c-1} \bar{X}(k) \cdot e^{j \frac{2\pi nk}{2N_c}} \quad (0 \leq n \leq 2N_c - 1)$$

Eq. 2-2

where n is the time-domain sample index in this DMT block.

A complete DMT signal block, $s(n)$, is then generated by adding a cyclic prefix of length v to $x(n)$.

$$s(n) = \begin{cases} x(2N_c + n) & -v \leq n \leq -1 \\ x(n) & 0 \leq n \leq 2N_c - 1 \end{cases}$$

Eq. 2-3

This digital signal, $s(n)$, is converted into an analog signal, $s(t)$, by a digital-to-analog converter (D/A) as,

$$s(t) = \sum_{n=-v}^{2N_c-1} s(n) \cdot g(t - nT)$$

Eq. 2-4

where $g(t)$ is the rectangular waveform filter at the transmitter, and T is a sampling interval. Such a signal block is called a DMT symbol.

The signal $r(t)$ arriving at the receiver contains the channel attenuated and distorted signal plus the additive noises, including AWGN, XT and IN.

$$r(t) = s(t) * h(t) + n(t)$$

Eq. 2-5

where $h(t)$ is the channel impulse response (CIR), $n(t)$ is the sum of the additive noises.

After an analog-to-digital converter (A/D), $r(t)$ is converted into a digital signal $r(n)$ with an equivalent discrete expression of

$$r(n) = s(n) * h(n) + n(n)$$

Eq. 2-6

where $h(n)$ is the equivalent discrete channel model.

After the CP is removed from $r(n)$, we obtain $y(n)$, which is then fed into an $2N_c$ -point DFT operator for demodulation.

$$\bar{Y}(k) = \sum_{n=0}^{2N_c-1} y(n) \cdot e^{-j \frac{2\pi nk}{2N_c}}$$

Eq. 2-7

As is known from the transmitter structure, only the first half of the output sequence, $\bar{Y}(k)$, $0 \leq k \leq N_c - 1$, is useful for recovering the data, because the second half is simply the conjugate of the first half and also the second half experiences much higher attenuation. After the DECODER and the DEMAPPER, this block of decoded QAM symbols is translated back into the received bit sequence.

It is convenient to think of a DMT system as consisting of N_c quadrature amplitude modulated (QAM) channels with certain ideal assumptions. For ideal AWGN channels, it has been proved in [6] that using IDFT and DFT to perform modulation and demodulation in transceivers will keep the orthogonality among the subchannels. However, the orthogonality is lost when the channel is nonideal. In [9], it is shown that using CP keeps the orthogonality between the subchannels even if the channel is nonideal, as long as the length of the cyclic prefix is longer than the channel impulse response.

2.2. CHANNEL MODELS AND IMPAIRMENTS IN ADSL TRANSMISSIONS

In this section, we briefly introduce the frequency response characteristics of subscriber loop and the main impairments in high-speed subscriber line transmissions.

2.2.1. Subscriber Loop Model

The most important characteristic of a subscriber loop is the transmission attenuation, which is also called the propagation loss. For a perfectly terminated⁴ loop with length l , the attenuation at frequency f can be approximated by [10]:

$$L_{dB}(l, f) \approx 8.686 \cdot l \cdot \alpha(f) \quad (\text{dB}) \quad \text{Eq. 2-8}$$

where the attenuation constant $\alpha(f)$ is roughly proportional to $\sqrt{f}$. (This is a good approximation with f less than 20 kHz or more than 200 kHz.) Figure 2-3 shows the propagation loss models of five carrier-service-area (CSA)⁵ test loops given in [11]. It is shown that the channel impulse response is heavily frequency dependent. Therefore, a

⁴ A loop is said to be perfectly terminated if it is terminated with its characteristic impedance.

⁵ CSA test loops are some typical short or medium length subscriber loops in current loop plant. These CSA loops are defined in the ADSL DMT standard [11] for ADSL transmission equipment test.

long equalizer will be required in a single-carrier transmission over UTPs. On the other hand, a DMT transmission divided the whole channel into many narrow-band subchannels, each having a nearly flat channel impulse response. Therefore, the equalization process is greatly simplified. Furthermore, instead of trying very hard to compensate some highly attenuated frequency areas in a single-carrier transmission, a DMT system simply doesn't use these areas by allocating zero bits to the corresponding subchannels.

2.2.2. Stationary additive noises

Electronic noise, including quantization noise from the A/D and D/A converters, and thermal noise in the analog portion of the receiver, can be modeled as additive white Gaussian noise (AWGN). Because of the very low power spectral density (PSD) of the AWGN, -140 dBm/Hz in [11], it is not one of the limiting impairments in ADSL DMT systems.

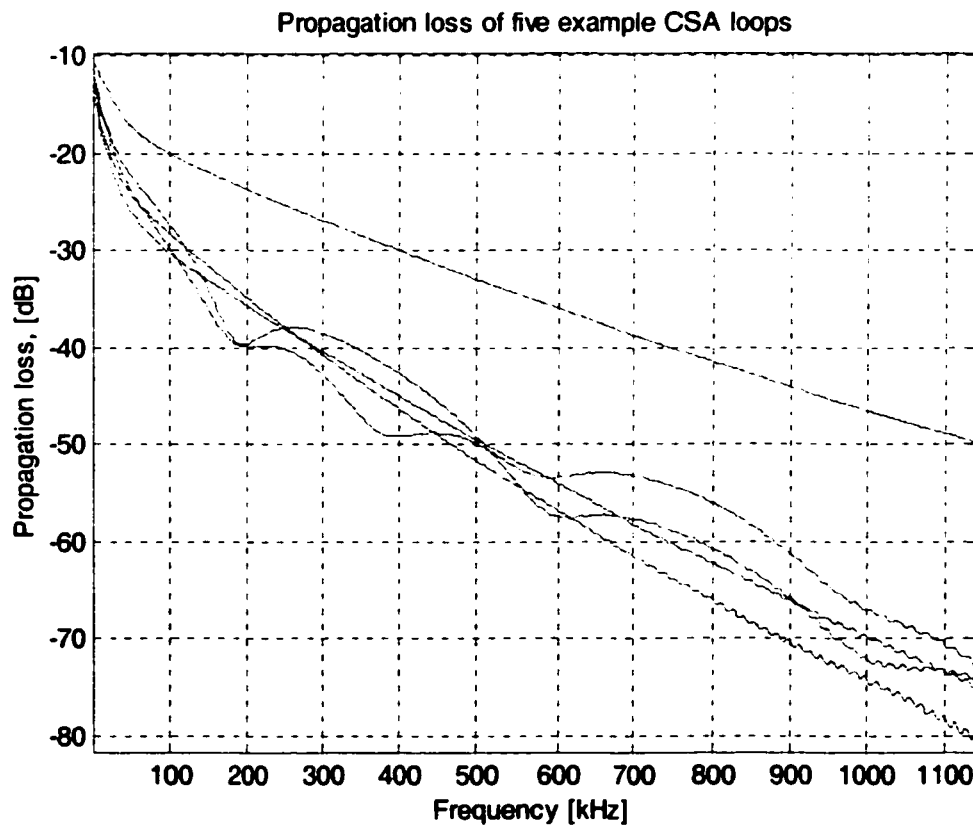


Figure 2-3 – Propagation loss measurement of 5 CSA test loops

Crosstalk noise generally refers to the interference between communication channels through some coupling path. Figure 2-4 shows the XT coupling between subscriber loop transmissions. Wire pair 1 and 2 represent two adjacent unshielded twisted pairs (UTP) in one pair bundle; near-end-crosstalk (NEXT) and far-end-crosstalk (FEXT) from wire pair 1 to wire pair 2 are as shown. It is easy to see that typically NEXT can be much more damaging than FEXT because it doesn't experience the attenuation of the loop, which greatly decreases the FEXT noise power.

Empirical studies show that crosstalk is one of the limiting impairments for high-speed data transmissions over unshielded-twisted-pairs (UTP) [12], since it usually has a much higher PSD than the background AWGN noise.

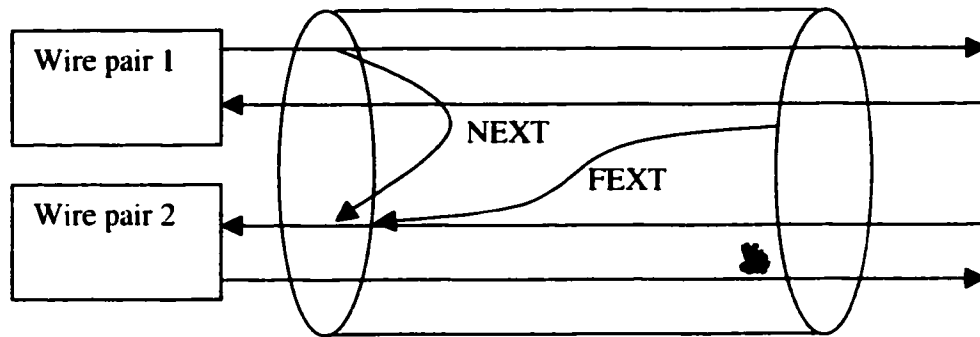


Figure 2-4 – Near-end crosstalk noise and Far-end crosstalk noise

2.2.3. Impulse Noise

Impulse noise in ADSL DMT systems is one of the major topics in this thesis. Detailed description of impulse noise will be presented in section 5.1.

2.3. SUMMARY

- The ADSL DMT transmission model is introduced. It is shown that a properly designed ADSL DMT system can be simplified to a set of ideal, independent subchannels that are quadrature amplitude modulated.

- The general channel model and major impairments are introduced. It was shown that ADSL channel response is heavily frequency dependent and DMT is a good solution to efficiently utilize the spectrum of such channels.
- The major additive impairments in ADSL transmissions are the crosstalk noises and impulse noise. A proper channel coding structure should provide satisfactory protection against both impairments.

Chapter 3.

Bandwidth-Efficient Turbo Coded Modulation

This chapter gives a brief introduction to turbo codes and the components of the turbo encoder and decoder. Shannon's noisy channel coding theorem is first reviewed since the performance of a turbo code is usually evaluated by the difference between the SNR requirement of a turbo coded system and the SNR lower bound calculated according to this theorem.

3.1. SHANNON LIMIT

Shannon's Noisy Channel Coding Theorem.

A channel coding scheme exists to make a communication system as reliable as possible if, and only if, the transmission rate, R_s , which is defined as the number of bits per channel use (channel symbol), is less than the channel capacity, C .

The Shannon limit hereafter is defined as the minimum signal-to-noise ratio (SNR) at which the required transmission rate in bits/symbol can be achieved.

A good approximation of each subchannel in ADSL DMT systems is a band limited ideal Gaussian channel. The normalized capacity of such a Gaussian channel, C in bits per symbol, can be expressed as [13]:

$$C = \log_2 \left(1 + \frac{R_s E_b}{N_0} \right) \quad \text{Eq. 3-1}$$

where E_b is the average transmission energy per information bit, R_s is the number information bits per symbol, and N_0 is the single-sided power spectrum density (PSD) of the background AWGN noise. The following inequality is obtained following Eq. 3-1:

$$\gamma_b \equiv \frac{E_b}{N_0} \geq \frac{2^{R_s} - 1}{R_s} \quad \text{Eq. 3-2}$$

Eq. 3-2 gives the normalized channel capacity as a function of SNR per bit, γ_b (defined as $\gamma_b = E_b/N_0$), with a bandwidth efficiency of R_s , which is plotted in Figure 3-1.

For QAM transmissions, channel capacity in bits per symbol is the maximum average mutual information contained in one symbol in a discrete-in-continuous-out channel, which can be calculated by [13]:

$$\begin{aligned} C &= \max_{p(\tilde{X})} I(\tilde{X}; \tilde{Y}) \\ &= \max_{p(\tilde{X})} \sum_i \int_{-\infty}^{\infty} p(Y | X_i) \cdot p(X_i) \cdot \log_2 \frac{p(Y | X_i)}{p(Y)} dY \end{aligned} \quad \text{Eq. 3-3}$$

where $I(\tilde{X}; \tilde{Y})$ is the mutual information per channel use for the symbol set of source data, $\tilde{X}$, and the symbol set of channel output, $\tilde{Y}$. $P(Y/X_i)$ is the probability of receiving Y when X_i is transmitted into the channel.

The maximization is performed over all possible distributions of channel input X . The maximum value is achieved when the data source is continuous and has a Gaussian distribution, which is calculated by Eq. 3-2. However, for an M -ary QAM transmission, the channel inputs are uniformly distributed from a symbol set of order M . Eq. 3-3 can then be expressed as:

$$C = \frac{1}{M} \sum_{i=0}^{M-1} \int_{-\infty}^{\infty} p(Y | X_i) \cdot \log_2 \frac{p(Y | X_i)}{p(Y)} dY$$

Eq. 3-4

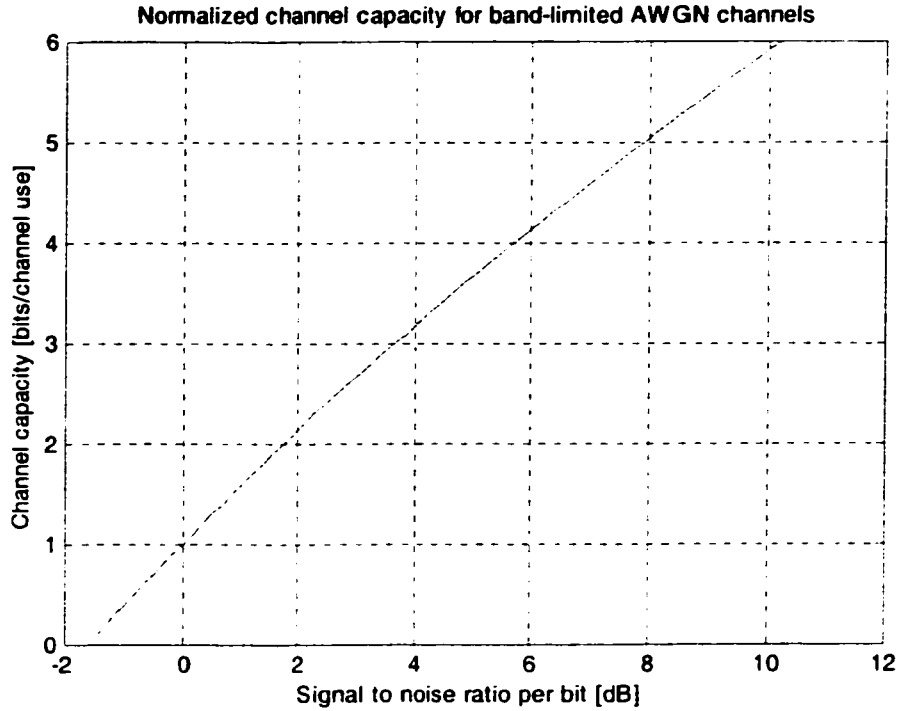


Figure 3-1 – Channel capacity vs. minimum γ_b of band-limited AWGN channels

Therefore, the channel capacity for two-dimensional constellations with equal probability channel input can be obtained as [5]:

$$C = \int \sum_{i=0}^{M-1} \frac{1}{M} p(Y | X_i) \log_2 \left[\frac{p(Y | X_i)}{\sum_k p(Y | X_k)} \right] dY$$

Eq. 3-5

The channel capacities calculated from (Eq. 3-5) for different QAM constellations are shown in Figure 3-2. These results will be used to evaluate how close the turbo coding scheme in QAM transmissions brings us to the Shannon limit. Please note that in Figure 3-2, no channel coding is assumed.

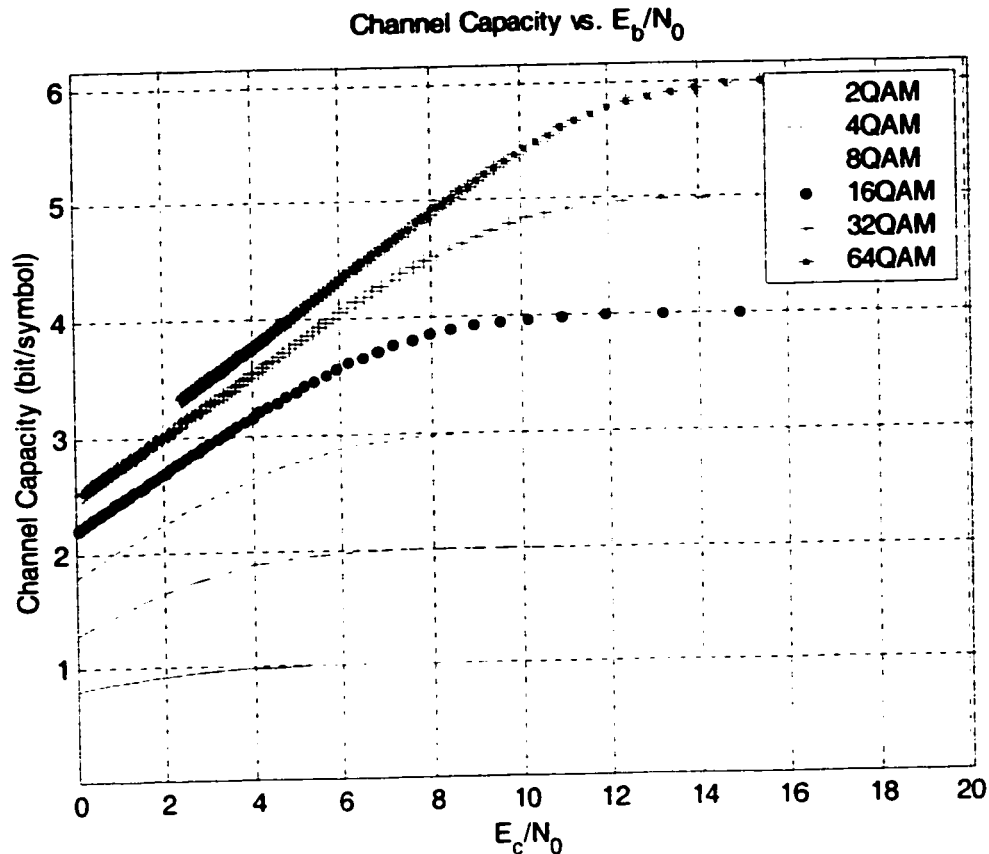


Figure 3-2 – Channel capacity for different QAM transmissions

3.2. PARALLEL CONCATENATED CODES – TURBO CODES

From Shannon's coding theorem, it is known that the performance of a given code can be improved by using longer code words. Unfortunately, the decoding complexity increases exponentially with the code word length, which makes using very long codes impractical in current communication systems. In order to obtain a high coding gain with moderate decoding complexity, concatenation has been shown as an attractive technique. Classic concatenated coding schemes consist of cascading several codes in serial, where the effective overall code length is much longer than its constituent codes (CCs). However the overall code rate is the product of the CCs, which can be very low. In general, a concatenated code is not as powerful as the best single-stage code with the same code rate and block length. The advantage of concatenated code lies in the multi-stage decoding. Compared to the exponentially increased decoding complexity of linear

block codes, using multi-stage decoding greatly reduces the decoding complexity, which is the sum of the decoding complexities of the component codes.

In 1993, parallel-concatenated channel coding was proposed, which was reported to achieve near Shannon limit performance [1]. In such a parallel concatenation structure, as shown in Figure 3-3, the two CCs are connected in parallel, separated by an interleaver, which plays a critical role for its outstanding performance. Decoding of parallel-concatenated codes is performed by an iterative decoding process. In iterative decoding, two soft-in/soft-out (SISO) decoders are used for the two CCs respectively. Unlike multi-stage decoding, the decoding procedures of the two CCs are no longer independent. The two decoders exchange reliability information about the information bits provided by the coding structure and parity bits of their own codes. With a sufficient number of iterations, a performance approaching that of maximum-likelihood (ML) decoding can be achieved with feasible complexity.

3.2.1. Parallel Concatenated Codes

The major components of a turbo code are 2 (or several) recursive systematic convolutional codes (RSCCs) separated by interleavers, denoted as π in this thesis. An example of a turbo encoder with two identical RSCCs is shown in Figure 3-3. The input bit sequence is processed in blocks of K bits. A block of bits is directly input into the first constituent RSCC, while an interleaved version of this block is input into the second constituent RSCC. The output of the turbo encoder consists of the systematic bits and the parity bits from both RSCCs. Note that the systematic bits in the second CC are not transmitted into the channel since they are only the interleaved repetitions of the systematic bits generated by the first CC. In this way, a rate-1/3 turbo code is formed using two rate-1/2 CCs. The puncturing mechanism shown in Figure 3-3 allows for higher code rate by puncturing the parity bits regularly from the two RSCCs.

Usually, rate 1/2 convolutional codes are used in turbo codes because of their implementation simplicity. However, rate $k/(k+1)$ codes can also be used in turbo codes to achieve higher code rates, at the cost of higher decoding complexity.

As stated in [1] and [14], the CCs are not necessarily identical. However, it was shown that the performance of turbo codes consisting of a strong constituent code and a

weak constituent code mainly follows the performance of the weak constituent code. Therefore, two (or more) identical CCs (or performance-comparable CCs) are used in most turbo code implementations.

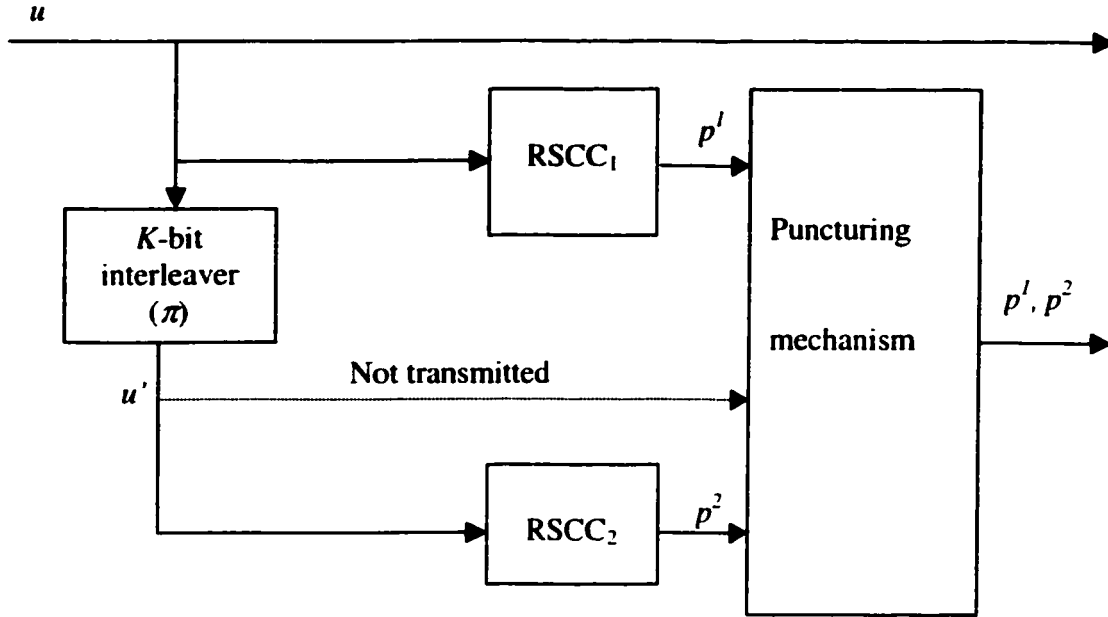


Figure 3-3 – A standard turbo encoder with two identical RSCCs with an interleaver

u is the information bit sequence, u' is the interleaved version of the information sequence, p^k is the parity bit sequence generated by the k^{th} RSCC, π is a K -bit interleaver.

In designing turbo codes, the goal is to choose the best CCs to maximize the effective free distance, $d_{free,eff}$, defined as the minimum Hamming weight of turbo code words with a weight-2 input sequence in [15], since these are the most likely low-weight code words. At large values of E_b/N_0 , this is tantamount to maximizing the minimum codeword weight. However, at low values of E_b/N_0 (the region of greatest interest) optimizing the weight distribution of the code words is as important as maximizing the minimum codeword weight [16]. The design issues of turbo codes will be discussed in details in Chapter 4 of this thesis.

3.2.2. Iterative Decoding

The Viterbi algorithm (VA) is known to offer ML sequence decoding performance with relatively low complexity. However, using a VA to decode turbo codes is far too complex because of the extraordinary complicated trellis introduced by the interleaver between the two encoders. The practical importance of turbo codes resides in the availability of the feasible iterative decoding algorithm. It is shown in [1] and [17] that with a sufficient number of iterations, the suboptimal iterative decoder can achieve performance very close to that provided by the ML decoder.

A conventional iterative decoding scheme for turbo codes is shown in Figure 3-4. The major components of the iterative decoder are the two SISO decoders. The output of each SISO decoder is the Log-Likelihood Ratio (LLR) of each information bit, L , which is defined as:

$$L(u_k) = \ln \frac{p(u_k = 1 | \bar{r})}{p(u_k = 0 | \bar{r})} \quad \text{Eq. 3-6}$$

where $\bar{r}$ is the received noisy code word and u_k is the k^{th} transmitted information bit in this code word.

The LLR output from conventional SISO algorithms (such as *maximum a posteriori algorithm*), can be separated into three terms,

$$L(u_k) = L_r(u_k) + L_{apri}(u_k) + L_e(u_k) \quad \text{Eq. 3-7}$$

where $L_r(u_k)$ is the channel LLR measurement of u_k , $L_{apri}(u_k)$ is the *a priori* information of u_k and $L_e(u_k)$ is the *extrinsic* information obtained from the coding structure and the other received symbols in the same code word. By using a good interleaver, $L_e(u_k)$ can be made almost independent of the other two terms. Therefore, this information can be used as the *a priori* information for the next SISO decoder.

By exchanging the *extrinsic* information between the component SISO decoders, better and better reliable soft outputs on the transmitted information bits can be obtained, until a point is reached where additional iteration gives negligible performance improvement.

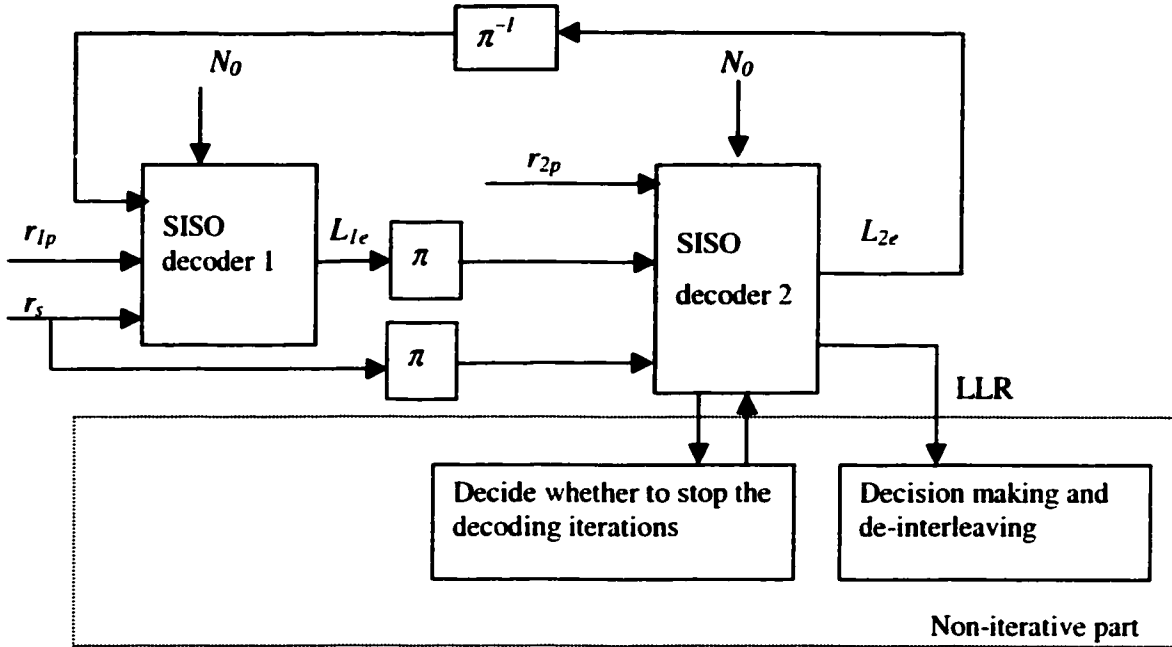


Figure 3-4 – A conventional iterative decoding scheme for a turbo code with two CCs

In the figure, $[r_s, r_{1p}, r_{2p}]$ are the received systematic and parity symbols. N_0 is the AWGN noise variance, π and π^{-1} are the K -bit interleaver and deinterleaver, L_{1e} and L_{2e} are the extrinsic information outputs from the 1st and 2nd SISO decoders, LLR is the final log-likelihood ratio calculated in the last iteration.

It was shown in [1], [14] and [18] that the performance differs significantly for the first several iterations. Simulations in [1] showed that using two iterations provides almost 2 dB additional coding gain over using one iteration where using three iterations will give a further improvement of 0.5 dB. There is a saturation value for the number of iterations beyond which no performance improvement can be achieved. This is because of the fact that more iterations result in more correlation between the *extrinsic* information and the other decoder inputs (received systematic and parity symbols) in the later decoding processes. The extrinsic information can improve the decoding performance only when it provides information in addition to the systematic information. Therefore, in later iterations, less and less additional information can be obtained from the *extrinsic* information and less improvement can be achieved.

In [18], a concept of cross-entropy (CE) is introduced in order to estimate the required number of iterations for a turbo decoding. The CE is a measure of the difference

between two distributions. If the CE between the outputs of the two SISO decoders is more than a given threshold, it means that the correlation between them is so large that further iterations will only give very little or no performance improvement. The iterative decoding can be stopped at this point with only a negligible performance loss. This criterion can be used to reduce the decoding delay in a turbo decoder. However, in practical systems, for the simplicity of implementation, the number of iterations are typically predetermined and kept constant during the decoding operations.

There are mainly two SISO algorithms that were proposed in the literature, the Bahl-Cocke-Jelinek-Raviv (BCJR) algorithm and the soft-output Viterbi algorithm (SOVA). The BCJR algorithm is a *symbol-by-symbol maximum a posteriori* (MAP) algorithm, and it is optimal in estimating the states of a Markov process observed in white noise [19]. However, it is too complicated to be implemented in practice because of the numerical representation of the probabilities, non-linear functions and the multiplications and additions of these values. Therefore, a suboptimal version of MAP, Max-log-MAP, was proposed to reduce the complexity, with the price of only a fraction of a dB performance loss [18], [20]. The Log-MAP algorithm introduced in [17] decreases the complexity by using a look-up table to implement the nonlinear log-computation, while avoiding any performance loss at the same time.

SOVA is another sub-optimal solution for SISO decoding algorithm. It is basically a traditional VA that generates soft outputs, which are derived from the metric differences between the surviving and the discarded paths entering one state [21]. The complexity of SOVA is much less than that of the MAP algorithm, while the performance reported is about 1 dB worse [17].

3.3. BANDWIDTH EFFICIENT TURBO CODED MODULATION

The good coding performance of turbo code is obtained at the cost of a bandwidth expansion hardly affordable in any bandwidth-limited applications, such as subscriber line transmissions. Therefore, bandwidth efficient turbo coded modulation schemes, termed turbo-trellis-coded-modulation (TTCM) in this thesis, have been investigated ever since turbo codes were initially proposed. Trellis-coded-modulation (TCM) combines convolutional codes with multi-level modulation techniques to provide excellent coding

gain without any bandwidth expansion [22]. It is straightforward to expect that turbo codes can also be incorporated with multi-level modulations, where the coding redundancy is absorbed by the constellation expansion.

In the literature, three different TTCM schemes were proposed by Le Goff [13], Robertson [23] and Benedetto [3]. Simulation results in these papers showed that near Shannon capacity transmission performance could be achieved by using bandwidth efficient turbo codes without sacrificing any bandwidth efficiency.

The TTCM scheme introduced in [13] is based on obtaining a rate- $k/(k+1)$ turbo code by puncturing parity bits. The resulted turbo coded bits are then mapped to multi-level symbols in a constellation of size 2^{k+1} . Therefore, the redundancy introduced by turbo code is absorbed by doubling the constellation size. Since most of the parity bits are punctured to obtain the high-rate turbo code, in the constellation mapping the parity bits are allocated to the best-protected positions for the optimal decoding performance. At the decoder, the LLRs of the turbo-coded bits are first extracted from the received channel symbols and then fed into a standard binary turbo decoder. The computation of the LLRs leads to relatively complicated expressions that depend on the signal-to-noise ratio (SNR) and channel characteristics. Nevertheless, it is possible to use linear approximations for the computation requirements. It is shown in [13] that very good decoding performance can be achieved using those linear approximations. The MAP decoding is not optimal because the LLRs calculated from the received symbols are no longer Gaussian distributed. Nevertheless, for large SNR values, the LLRs distributions are approximately Gaussian, and therefore the turbo decoding performance is still quite attractive.

This TTCM structure is shown to provide about 6 to 8 dB coding gain over uncoded systems and 2.5 dB coding gain over a 64-state TCM in AWGN channels, with a turbo block length of $K=4096$ bits [13].

The advantage of this TTCM scheme is that the receiver employs any conventional SISO algorithm (MAP, Max-Log-MAP, Log-MAP or SOVA) while no additional complexity is added because of the multi-dimensional modulation. The disadvantage of this method is that the coding performance becomes worse for larger constellations, since large number of bits contained in one symbol experience the same noise (strong correlation), which degrades the performance of the iterative decoding. The Le Goff's

TTCM scheme is used in [24] and [25] to improve the performance of multicarrier communications.

The second bandwidth efficient turbo coding structure was proposed by Robertson in [23]. This approach uses two TCM units instead of the constituent codes in conventional turbo codes. A symbol interleaver is used instead of a bit interleaver, as shown in Figure 3-5. The multi-level symbols generated by the two TCM units are punctured alternately at the output of the TTCM. The decoding of this TTCM structure is based on the Symbol-MAP (SMAP) algorithm, which is a natural extension of the MAP algorithm.

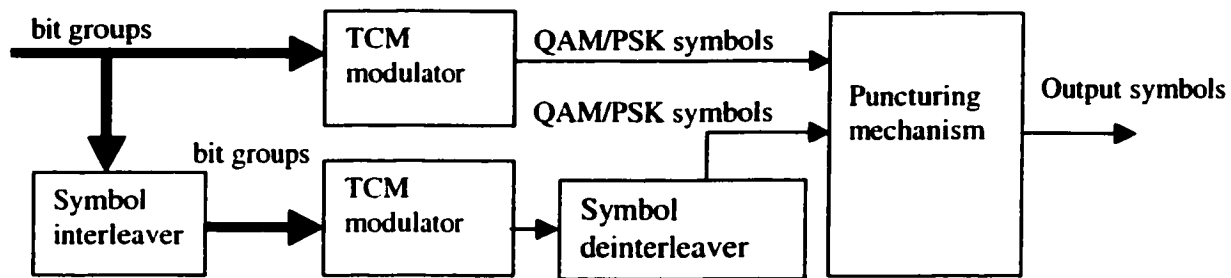


Figure 3-5 – The encoder of the Robertson's TTCM

For higher bandwidth efficiencies, uncoded bits can be added to each symbol using Ungerboeck's set partitioning. In the TTCM decoder, the uncoded bits are regarded as unknown ambiguities in the metric evaluation and they are decoded during the last decoding iteration. Waiting until the last iteration will ensure reliable and computational-efficient detections of these uncoded bits in case of successful turbo decoding. Please refer to [26] for details.

Simulation results in [23] show a coding gain of 1.7 dB over 64-state Ungerboeck TCM with 8PSK modulation at a BER of 10^{-4} . At this BER, the TTCM system has a 0.5 dB coding gain advantage over the first discussed TTCM scheme.

The advantage of Robertson's TTCM structure is that it can achieve good coding performance with a good flexibility of code rate. Simulation results in [23] and [26] showed near Shannon limit performance using this TTCM scheme. However, a BER floor occurs before 10^{-5} . The early BER floor is a result of the puncturing of one of the

two multi-dimensional symbols at the encoder. Such an early BER floor makes it a good candidate to be used in a concatenated code.

A second disadvantage of this TTCM is the complexity involved in the decoding when there are uncoded bits. The uncoded bits must be taken as unknowns during the SMAP decoding process. Since the decoder has to calculate LLRs for all combinations of the possible values of these uncoded bits, it introduces very large computational load when there are large number of uncoded bits.

A third disadvantage is the additional delay introduced by the deinterleaver in the encoder. This deinterleaver doubles the encoding delay during the encoding process.

The third TTCM scheme of interest was proposed by Benedetto in [3]. This TTCM structure uses high rate CCs and all the parity bits are transmitted. This TTCM performs within 1 dB from the Shannon limit at a BER of 10^{-7} , and outperforms all TTCMs introduced previously for the same throughput [3]. Simulation results also show no BER floor until 10^{-8} , which is adequate for most data applications. This advantage of a low BER floor makes it possible to use a single TTCM in applications with a BER requirement of as low as 10^{-7} , instead of using a concatenated coding structure.

Another advantage of Benedetto's TTCM scheme is that uncoded bits can be easily incorporated into the coding structure. The proposed turbo coded ADSL DMT transmission system in this thesis is an extension of Benedetto's TTCM scheme with some additional encoding and decoding simplifications.

3.3.1. Benedetto's TTCM encoding structure

An example of Benedetto's TTCM encoder is shown in Figure 3-6, (regenerated from [3]). In this TTCM scheme, two rate-2/3 CCs are used to implement a rate-1/2 turbo encoder. The incoming information sequence is divided into two subsequences, whose bits are identified as u_1 and u_2 . At each instant, a pair of bits, $[u_1, u_2]$, are input into the first CC from which one parity bit, denoted by p_1 , is generated. At the output of the constituent encoders, one of the systematic bits (u_2 in this case) is punctured. The interleaved versions of these two bit sequences, identified by $[u_1', u_2']$ are input into the second CC, where the second parity bit, p_2 , is generated and the u_1' is punctured at the output. Finally, these four bits, $[u_1, p_1, u_2', p_2]$, are mapped into a 16QAM symbol.

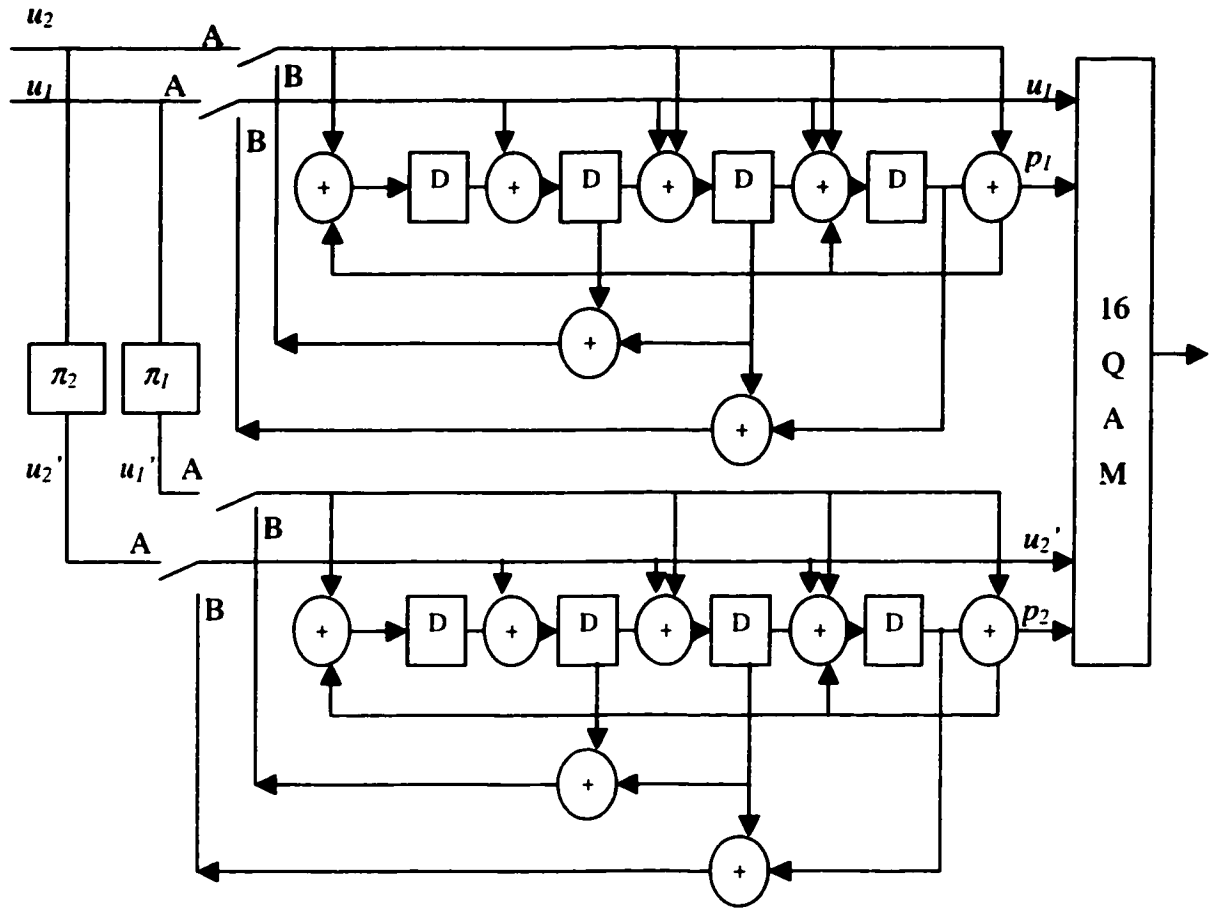


Figure 3-6 – Benedetto's Turbo Trellis Coded Modulation encoding structure

Two rate-2/3 CCs, two interleavers, 16QAM, 2 bits/sec/Hz, [3]

In the mapping from the 4-bit vector $[u_1, p_1, u_2', p_2]$ to a 16QAM symbol, $[u_1, p_1]$ are used to select the in-phase component and $[u_2', p_2]$ for the quadrature component. This effectively decouples the turbo-coded bits from the two CCs. This will simplify the decoding procedure since each SISO decoder only needs to operate on either the in-phase or the quadrature component of the received QAM symbols.

The switches at the input of each shift-register are initially in the 'A' position to receive the incoming bit sequence and are put in the 'B' position at the end of each turbo block to return both encoders to the zero state.

3.3.2. TTCM decoding structure

In the TTCM encoder shown in Figure 3-6, every state transition in each CC corresponds to a multi-level symbol. This symbol is either the in-phase or the quadrature component of the QAM symbol. The Symbol-LLR (SLLR) will then be the direct result of the SISO algorithm. The LLRs of the bits contained in the symbols are calculated from the SLLRs. The *extrinsic* information is obtained by subtracting the channel component and *a priori* component from the bit LLRs. It is then fed into the other SISO decoder as the *a priori* information. A generic decoding structure is shown in Figure 3-7. Note that the interleaver, π , and deinterleaver, π^{-1} , each consists of a pair of interleavers (π_1 and π_2), as shown in Figure 3-6, and their corresponding deinterleavers (π_1^{-1} and π_2^{-1}).

The critical modules in Figure 3-7 are the *Symbol-MAP decoders* and *bit-reliability calculation* module. Section 4.1.2 and [3] provide additional details on the operations of the *Symbol-MAP* and *bit reliability calculation*.

Higher QAM constellations can be supported by simply adding uncoded information bits. This is equivalent to adding parallel transitions to the trellis structure of the turbo code. However, to guarantee an overall satisfactory performance, the uncoded bits must be protected as well as the turbo coded bits. Set partitioning is used to give uncoded bits better protection, where they are used to select constellation points within a subset of the constellation having larger minimum distance. However, in the case of TTCM, the minimum distance in the subset must be very large because turbo codes can provide a coding gain of about 6 to 7 dB to the turbo coded bits. Detailed analysis of the protection balance between the turbo coded bits and uncoded bits is performed in section 4.2.5.

3.4. SUMMARY

- In the literature, performances of bandwidth efficient TTCM schemes were shown to approach the Shannon limit. In this chapter, we first presented a review on the Shannon limit in the form of the minimum γ_b requirement for a reliable transmission in a band limited AWGN channel. These values will be used later in this thesis to evaluate the performance of turbo-coded QAM and ADSL DMT transmissions.
- We discussed the basic encoding and decoding structure of a typical turbo code. A typical turbo encoder consists of two identical recursive systematic convolutional

constituent codes (RSCCs). An interleaver is used to separate these two CCs. The transmitted sequence is the multiplex of the systematic bit sequence and the two parity bit sequences generated by the CCs. Higher rate CCs can be used to generate a high rate turbo code. Puncturing is also an effective way to generate high-rate turbo codes.

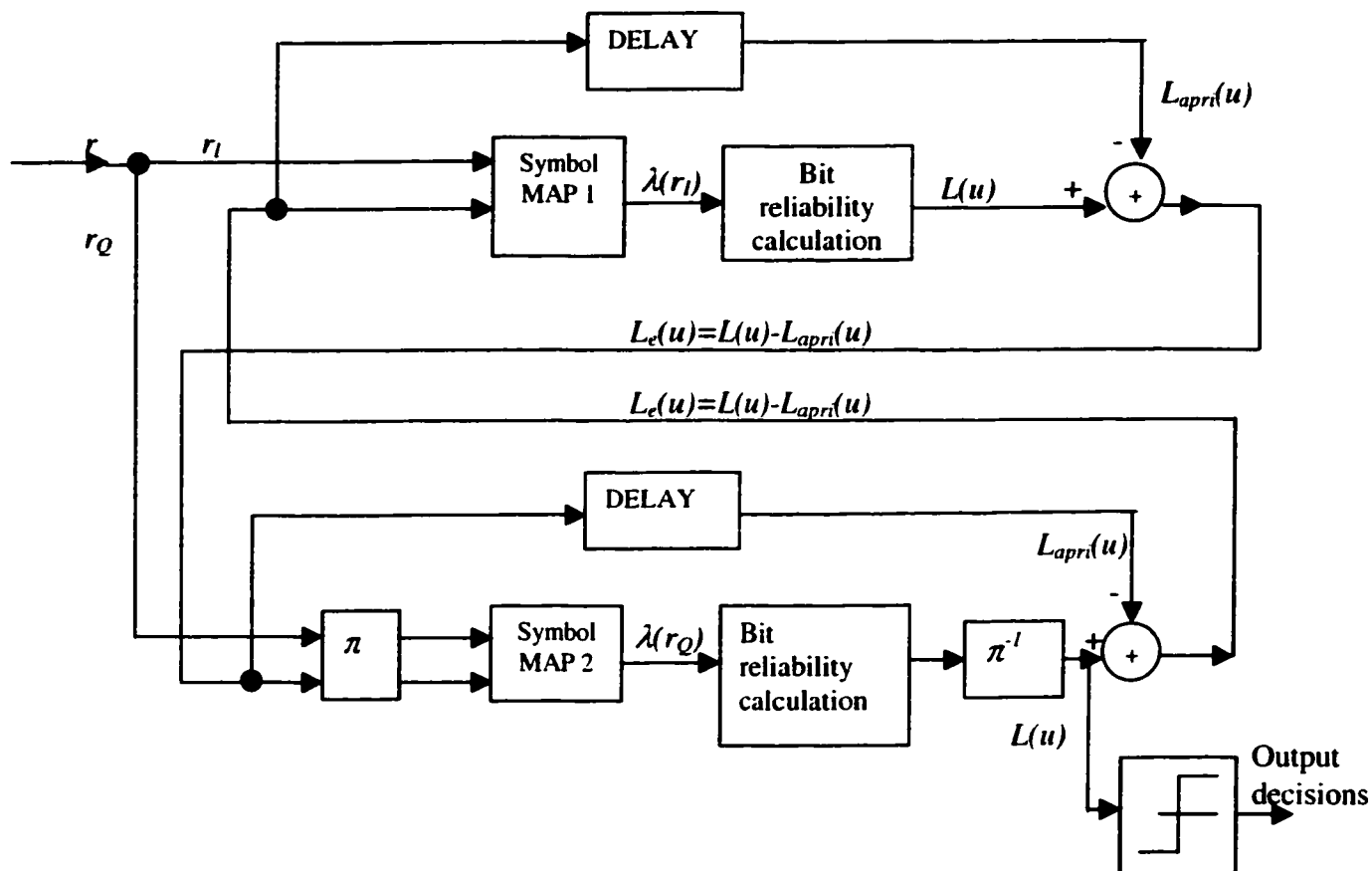


Figure 3-7 – Iterative (Turbo) Decoder Structure for the TTCM shown in Figure 3-6

r is the received two-dimensional symbols, $[r_I, r_Q]$ are the in-phase and quadrature components, λ is the Symbol-LLR of the in-phase or the quadrature component calculated from the Symbol-MAP algorithm, $L(u)$ is the LLR of the information bits derived from λ , $L_e(u)$ is the extrinsic information exchanged between the two SISO decoders, π and π^{-1} are the interleaver and deinterleaver.

- We presented a survey of three TTCM structures along with their advantages and disadvantages. All of the three TTCM structures were shown to provide excellent coding gain performance when the constellation size is small.
- Benedetto's TTCM structure introduced in [3] provides excellent coding gains and a low BER floor. It also provides the easiest way to increase the bandwidth efficiency by adding uncoded bits. We identified this TTCM structure as the most suitable candidate for our application.

Chapter 4.

Turbo Codes in ADSL DMT Systems

In this chapter, we propose a multilevel TTCM structure based on Benedetto's TTCM structure described in section 3.3.1. The encoding is performed pragmatically, where a single binary turbo encoder is employed to implement turbo codes with various rates. Different numbers of uncoded bits could be incorporated in the constellation mapping without changing the turbo encoding process. The proposed TTCM tries to keep the coding gain constant for any size of QAM constellation; a property which is particularly useful for ADSL DMT systems. At the decoder, an equivalent soft binary symbol (ESBS) of each coded bit carried in the received QAM symbol is calculated and a standard binary turbo decoding is performed over these ESBSs. To achieve better performance from the binary turbo decoding, Gray coding is employed in the constellation mapping to provide extra protection to some of the coded bits. Better protection is usually applied to parity bits since error paths in turbo codes contain more parity bits than the systematic bits.

The asymptotic coding gain of the turbo coded QAM transmissions is predicted based on the simulation results obtained from the literature and through our simulations. Simulation results of QAM transmissions and ADSL DMT transmissions employing the

proposed TTCM are presented and performance discussions are performed based on the observations from the simulations. To meet the BER requirement of 10^{-7} in ADSL transmissions, concatenated coding schemes were used to lower the error floor resulted from the turbo decoding, at the expense of additional complexity and latency. At last, coding gain expectations of applying this TTCM to a practical ADSL DMT system is predicted based on the performance of turbo coded QAM transmissions.

4.1. TTCM IN ADSL DMT SYSTEMS

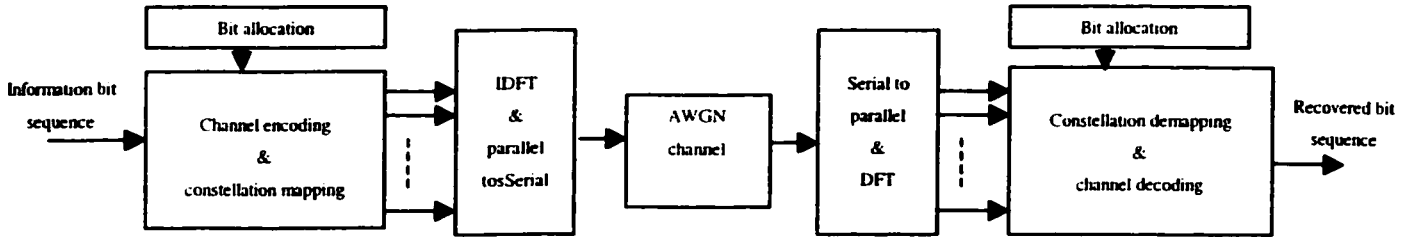
4.1.1. TTCM encoding structure

A generic channel coding and modulation structure in ADSL DMT systems is shown in Figure 4-1.a. A block of information bits is fed into a forward-error-correction (FEC) encoder. The coded bits are then divided into bit groups according to the transmission capabilities of different subchannels. A bit allocation algorithm is used to generate the optimal number of bits and transmission power carried in each subchannel. These bit groups are then mapped into QAM symbols from different sizes of constellations. The modulation of the subchannels by these QAM symbols is then performed by an IDFT operation. The output is a time-domain sample sequence of the N_c frequency-multiplexed QAM transmissions, where N_c is the number of subchannels.

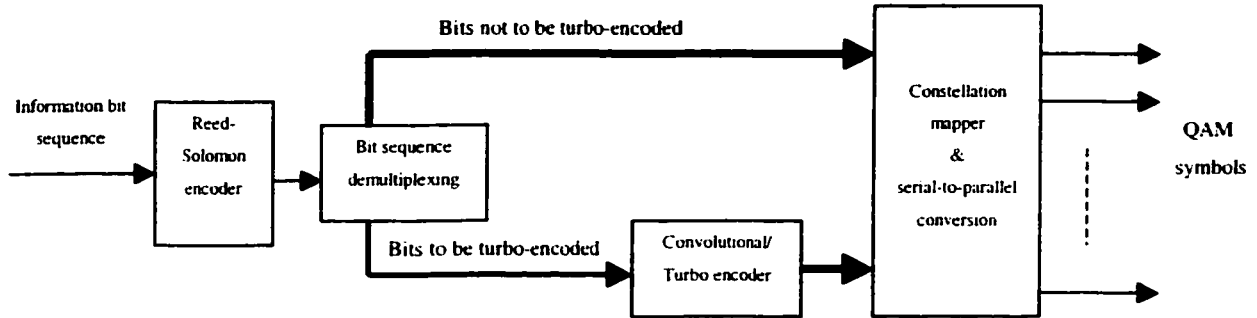
The structure of the channel encoder and the constellation MAPPER is shown in Figure 4-1.b. This is a concatenated channel coding structure with an RS code as the outer code and a 4-dimensional trellis coded modulation (4D-TCM) as the inner code, as specified in the current ADSL DMT standard [28]. The 4D-TCM operates across all subchannels. An RS code is chosen as the outer code due to its strong burst error correction capability. Usually, to further improve the error correction capability of the RS code, an interleaver is used right after the RS encoder in the transmitter and the corresponding deinterleaver is used in the receiver before the RS decoder. Theoretically, with an unlimited interleave depth, any length of burst error can be corrected. In practice, the transmission system delay requirement puts determine the maximum value of the interleave depth.

The proposed turbo coded ADSL DMT system has the similar channel coding structure, except for the substitution of the 4D-TCM (the combination of the

convolutional encoder shown in Figure 4-1.b and a specific constellation mapper) with a TTCM.



(a) ADSL DMT channel coding and modulation structure



(b) Channel encoding and constellation mapping

Figure 4-1 – ADSL DMT system channel coding configuration

Although any turbo encoder could be employed, we will always use the rate-1/2 turbo encoder, shown in Figure 3-6, in the following discussions.

An optimum bit allocation algorithm usually allocates more bits to lower-frequency subchannels where the channel attenuation is low, as shown in Figure 2-3. This results in larger QAM constellations in lower frequency subchannels. In our proposed TTCM, to allow the variable size of QAM constellations in different subchannels, each subchannel carries a fixed number of turbo-coded bits and a variable number of un-turbo-coded bits⁶. The number of un-turbo-coded bits allocated to each subchannel is calculated as the

⁶ Un-turbo-coded-bits could be either information bit sequence or RS code word bits if an outer RS code is employed.

difference between the subchannel transmission capability and the number of assigned turbo-coded bits.

The constellation mapping of the k^{th} subchannel is performed as shown in Figure 4-2, assuming a transmission capability of b_k bits. Among these b_k bits, there are 4 turbo coded bits: $[u_1, p_1, u'_2, p_2]$. Combined with (b_k-4) un-turbo-coded bits, $[u_3, \dots, u_{b_k-2}]$, a QAM symbol in a constellation of size 2^{b_k} is generated and used to modulate the k^{th} subcarrier.

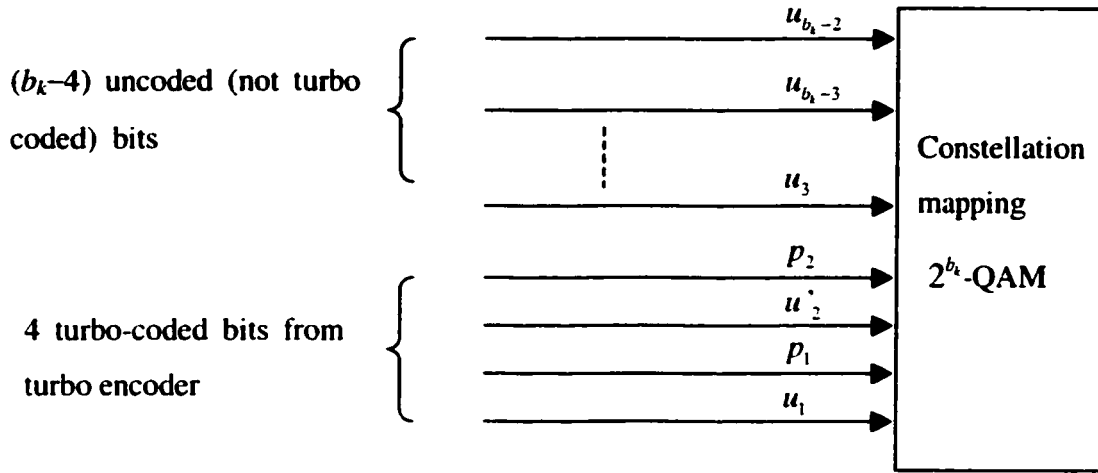


Figure 4-2 – Bits-to-symbol mapping of the proposed TTCM in ADSL DMT systems

Every b_k bits are mapped to one 2^{b_k} QAM symbol, These contain 4 turbo-coded bits and (b_k-4) uncoded (not turbo-coded) bits.

Compared to a conventional TCM, this TTCM structure expands the constellation size by four times since each QAM symbol contains two parity bits. Usually, expanding the constellation size by more than twice results in an inefficient coding gain in conventional TCM. However, simulation results given in [23] and [3] showed that near Shannon limit performance was obtained by TTCM structures even when the constellation was expanded by a factor of four.

Employing the mapping as shown in Figure 4-2 suggests that a minimum constellation of 16QAM be used. In ADSL DMT transmissions, the QAM constellation used in a subchannel is decided based on its propagation loss and the additive noise

variance. Some subchannels can only support 4QAM or 8QAM. Only the turbo-coded bits will be assigned to these subchannels.

As an example, when $b_k=6$, every 6-bit group is mapped to a signal point in a 64QAM constellation, as shown in Figure 4-3. The constellation consists of four quadrants. Among the six bits, the two un-turbo-coded bits will be used to select one of the four quadrants. The two pairs of turbo-coded bits, $[u_1, p_1]$ and $[u_2, p_2]$, are used to determine the in-phase and the quadrature values of the constellation point inside each quadrant. The mapping of each bit pair to the in-phase (quadrature) values follows the rule of Gray mapping. From a different perspective, the overall QAM constellation can be viewed as consisting of 16 subsets each containing four signal points. The four turbo-coded bits are used to select a subset, while the two un-turbo-coded bits are used to select a constellation point within a given subset. Let's assume that the minimum Euclidean distance between two adjacent constellation points is d . The minimum Euclidean distance between constellation points within each subset is then $4d$. Therefore, the un-turbo-coded bits are much better protected from additive noise than the turbo-coded bits.

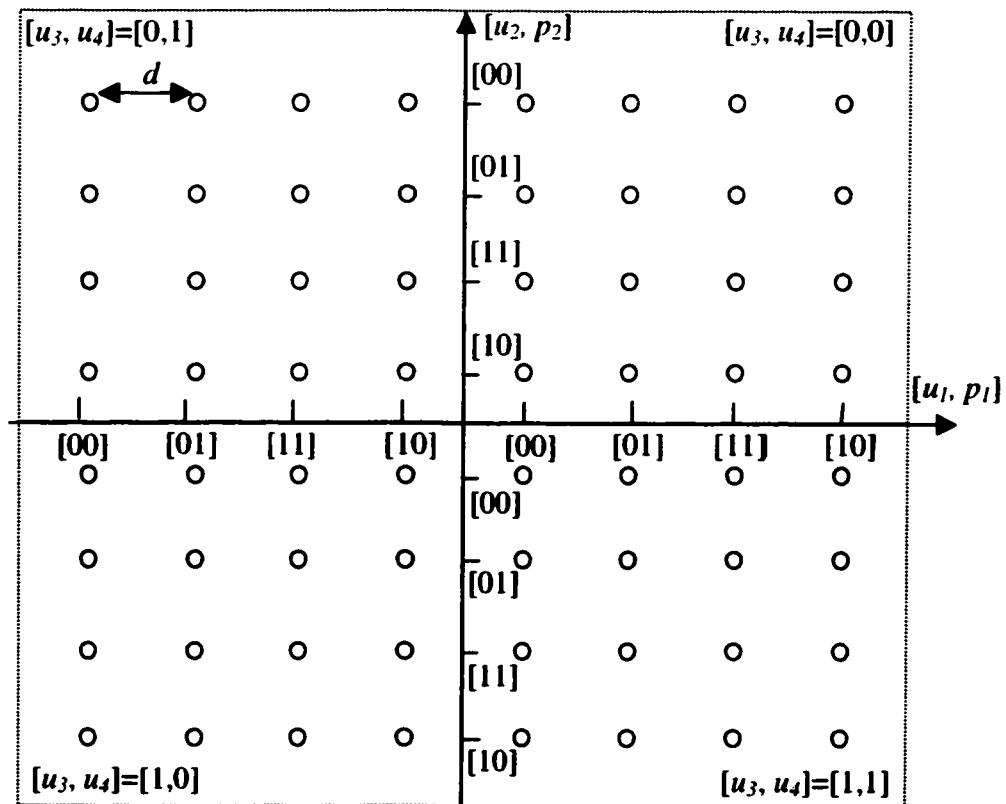


Figure 4-3 – Constellation mapping of 64QAM in the proposed TTCM

4.1.2. TTCM decoding structure

4.1.2.1. Optimal Symbol-MAP decoding algorithm

The TTCM decoding structure has been shown in Figure 3-7, corresponding to the encoder structure shown in Figure 3-6. When a $[N_{sc}, K_{sc}]^7$ convolutional code is used as the CCs, each transition in the convolutional encoder trellis is caused by a combination of K_{sc} input bits and N_{sc} coded bits are generated. When $K_{sc}=1$, the symbol-by-symbol maximum a posteriori (MAP) algorithm calculates the LLR of the transmitted bit by calculating the state transitions probabilities [19]. For $K_{sc}>1$, a natural extension of the MAP algorithm is to calculate the LLR of every K_{sc} combination through the calculation of the state transition probabilities. This LLR is known as Symbol⁸-LLR (SLLR) and the modified MAP algorithm to calculate the SLLR is called the Symbol-MAP (SMAP) algorithm.

Let's assume that a turbo encoder receives a block of K information bits and generates a code word of N bits. These N bits are then allocated to N_S QAM symbols. If we follow the same TTCM encoder structure as shown in Figure 3-6, the in-phase component of the QAM symbol contains only the output bits from the first CC, and the quadrature component contains only the bits from the second CC. The first SISO decoder then works only with the received in-phase components.

In the following, the in-phase (or quadrature) component in the l^{th} transmitted QAM symbol and the corresponding received symbol are denoted as x_l and y_l . Furthermore, we assume that the l^{th} QAM symbol consists of b_l bits, including the turbo-coded bits and the un-turbo-coded bits. $u_{l,j}$ is used to identify the j^{th} turbo-coded systematic bit in the l^{th} QAM symbol. The LLR of the j^{th} bit in the l^{th} QAM symbol is defined as:

$$L(u_{l,j}) = \ln \frac{p(u_{l,j} = 1 | y)}{p(u_{l,j} = 0 | y)}$$

Eq. 4-1

where y is the received QAM symbol vector of length N_S .

⁷ $N_{sc}=3, K_{sc}=2$ in the CC shown in Figure 3-6.

⁸ A 'symbol' means a particular combination of the K_{sc} input bits.

In the constellation mapping proposed in section 4.1.1, each QAM symbol contains a fixed number of turbo-coded bits, say N_b bits, among which K_b bits are the systematic bits. If we further assume that the in-phase (or quadrature) component of a QAM symbol contains the turbo-coded bits generated by one state transition, then we have $K_b = K_{sc}$.

The K_b -length systematic bit vector in the l^{th} QAM symbol, $[u_{l,1}, \dots, u_{l,K_b}]$, is named the l^{th} systematic symbol, identified as $\bar{u}_l$. The SLLR of $\bar{u}_l$, $\lambda(\bar{u}_l)$, is defined as,

$$\lambda(\bar{u}_l) = \log \frac{P(\bar{u}_l | \bar{y})}{P(\bar{0} | \bar{y})} \quad \text{Eq. 4-2}$$

where ' $\bar{0}$ ' is the all zero vector of length K_b .

The symbol *a posteriori* probability, $p_{apo}(\bar{u}_l | y)$, is calculated as,

$$\begin{aligned} p_{apo}(\bar{u}_l | y) &= \frac{\sum_{S_l=m} \sum_{S_{l-1}=m'} P(S_{l-1}=m' | y_1^{l-1}) \cdot P(S_l=m, x_l, y_l | S_{l-1}=m') \cdot P(y_{l+1}^N | S_l=m)}{P(y_1^N | y_1^{l-1})} \\ &= \sum_{S_l=m} \sum_{S_{l-1}=m'} \alpha_{l-1}(m') \cdot \gamma(m, m', x_l) \cdot \beta_l(m) \end{aligned} \quad \text{Eq. 4-3}$$

where y_1^N is the vector $[y_1, \dots, y_N]$, x_l is decided by $\bar{u}_l$ and the transition state S_{l-1} , and,

$$\alpha_l(m) = \frac{\sum_{m'} \sum_{\bar{u}_l} \gamma(m, m', x_l, y_l) \cdot \alpha_{l-1}(m')}{\sum_m \sum_{m'} \sum_{\bar{u}_l} \gamma(m, m', x_l, y_l) \cdot \alpha_{l-1}(m')} \quad \text{Eq. 4-4}$$

$$\beta_l(m') = \frac{\sum_m \sum_{\bar{u}_l} \gamma(m, m', x_{l+1}, y_{l+1}) \cdot \beta_{l+1}(m)}{\sum_m \sum_{m'} \sum_{\bar{u}_l} \gamma(m, m', x_{l+1}, y_{l+1}) \cdot \alpha_l(m)} \quad \text{Eq. 4-5}$$

which can be calculated recursively, and

$$\gamma(m, m', x_l, y_l) = p(S_l=m, x_l, y_l | S_{l-1}=m') \quad \text{Eq. 4-6}$$

Eq. 4-6 can be rewritten as

$$\gamma(m, m', x_l) = p(y_l | x_l, S_l=m, S_{l-1}=m') \cdot p(x_l | S_l=m, S_{l-1}=m') \cdot p(S_l=m | S_{l-1}=m') \quad \text{Eq. 4-7}$$

where the *a priori* information is included in the state transition probability $p(S_l = m \mid S_{l-1} = m')$.

Therefore, the SLLR can be expressed as:

$$L(\bar{u}_l) = \ln \frac{\sum_{S_l=m} \sum_{S_{l-1}=m'} \alpha_{l-1}(m') \cdot \gamma(m, m', \bar{u}_l) \cdot \beta_l(m)}{\sum_{S_l=m} \sum_{S_{l-1}=m'} \alpha_{l-1}(m') \cdot \gamma(m, m', \bar{0}) \cdot \beta_l(m)} \quad \text{Eq. 4-8}$$

and the bit LLR is calculated as:

$$L(u_{l,j}) = \ln \frac{\sum_{u_l, u_{l,j}=1} \sum_{S_l=m} \sum_{S_{l-1}=m'} \alpha_{l-1}(m') \cdot \gamma(m, m', x_l) \cdot \beta_l(m)}{\sum_{u_l, u_{l,j}=0} \sum_{S_l=m} \sum_{S_{l-1}=m'} \alpha_{l-1}(m') \cdot \gamma(m, m', x_l) \cdot \beta_l(m)} \quad \text{Eq. 4-9}$$

Once we have the SLLRs, the LLR of the j^{th} bit in the l^{th} symbol can also be calculated by,

$$L(u_{l,j}) = \ln \frac{\sum_{\bar{u}_l, u_{l,j}=1} p(\bar{u}_l \mid y)}{\sum_{\bar{u}_l, u_{l,j}=0} p(\bar{u}_l \mid y)} = \ln \frac{\sum_{\bar{u}_l, u_{l,j}=1} e^{L(\bar{u}_l)}}{\sum_{\bar{u}_l, u_{l,j}=0} e^{L(\bar{u}_l)}} \quad \text{Eq. 4-10}$$

This LLR will be sent to the other SISO decoder as *a priori* information. However, as shown in Eq. 4-7, a symbol *a priori* information, $L_{\text{apri}}(\bar{u}_l)$, is required since each transition in the trellis is caused by an input bit vector $\bar{u}_l$. This symbol *a priori* probability can be calculated as:

$$P[u = (u_1, \dots, u_{b_l})] = \prod_{j=1}^{b_l} \frac{e^{u_j L(u_j)}}{1 + e^{L(u_j)}} \quad \text{Eq. 4-11}$$

when the bit reliabilities, $L(u_{l,j})$, are assumed independent of each other, which is a fair assumption if a proper interleaver is used.

As mentioned earlier, in the turbo encoder shown in Figure 3-6, the in-phase component contains only output bits from the first CC, $[u_l, p_l]$ and the quadrature component contains the output bit pair $[u_2', p_2]$ from the second CC. This decoupling of the two CCs is necessary for the iterative decoding process shown above. The output of the first SISO decoder is the LLR values of both $[u_l, u_2]$. However, the in-phase

component does not directly contain the systematic bit sequence u_2 , and there is no channel LLR measurement available from this decoder. Therefore, the LLR value calculated from Eq. 4-9 can be expressed as:

$$L(u_1) = L_c(u_1) + L_{ex}(u_1)$$

$$L(u_2) = L_{ex}(u_2)$$

Eq. 4-12

where it contains the channel LLR information and *extrinsic* information of u_1 and only the *extrinsic* information of u_2

For the second SISO decoder, the quadrature component does not contain the channel values from the systematic bit sequence u_1 . To obtain the best decoding performance, it needs the channel value of systematic bits, u_1 , and the *extrinsic* information calculated from the first SISO decoder for both u_1 and u_2 . These are exactly the outputs from the first SISO decoder. A similar analysis applies to the reliability exchange from the second SISO decoder to the first SISO decoder in the later iterations.

In the SMAP algorithm, the uncoded bits are considered as unknowns. Let's assume that the in-phase (quadrature) component of the l^{th} QAM symbol contains an uncoded bit vector I_l of length $b_{l,t}$. In the calculation of the branch metric in Eq. 4-7, the first item is expressed as:

$$\begin{aligned} p(y_l | \bar{u}_l) &= p(y_l | u_{l,1}, u_{l,2}, p_l) \\ &= \sum_{I_l} p(y_l, I_l | u_{l,1}, u_{l,2}, p_l) \\ &= \sum_{I_l} p(y_l | u_{l,1}, u_{l,2}, p_l, I_l) \cdot p(I_l | u_{l,1}, u_{l,2}, p_l) \\ &= \sum_{I_l} p(y_l | u_{l,1}, u_{l,2}, p_l, I_l) \cdot p(I_l) \\ &= \frac{1}{2^{b_{l,t}}} \sum_{I_l} p(y_l | u_{l,1}, u_{l,2}, p_l, I_l) \end{aligned}$$

Eq. 4-13

where the last step is based on the fact that the uncoded information bits are assumed to be ± 1 with equal probability.

Therefore, applying the SMAP decoding to the proposed TTCM consists simply in replacing the first item in Eq. 4-7 with Eq. 4-13.

In a convolutional code, uncoded bits cause parallel transitions between two states in the trellis. To calculate the SLLRs, a SMAP decoder merges the parallel transitions by adding the probabilities of all possible parallel transitions. There is one such sum for one particular combination of the uncoded bit groups. The SMAP decoder then calculates and passes on only the likelihood's of the coded bits. During the last decoding stage, decisions (and if desired, reliabilities) on the uncoded bits are made based on the decisions on the turbo-coded bits. It should be noted that finding the maximum likelihood path in the turbo code trellis is still the criterion to make the optimal decisions on uncoded bits. However, a suboptimal but much simpler method is to take into account only those transitions along the most likely path decided by the turbo decoding decisions.

4.1.2.2. Simplified TTCM decoding

The optimal SMAP decoding algorithm involves high computational complexity, because the decoder has to go through all combinations of the uncoded information bits to calculate the SLLRs. It would be more practical if a traditional binary turbo decoder were employed. With this in mind, we propose the decoding structure shown in Figure 4-4 for the TTCM proposed in section 4.1.1 with a constellation mapping shown in Figure 4-2.

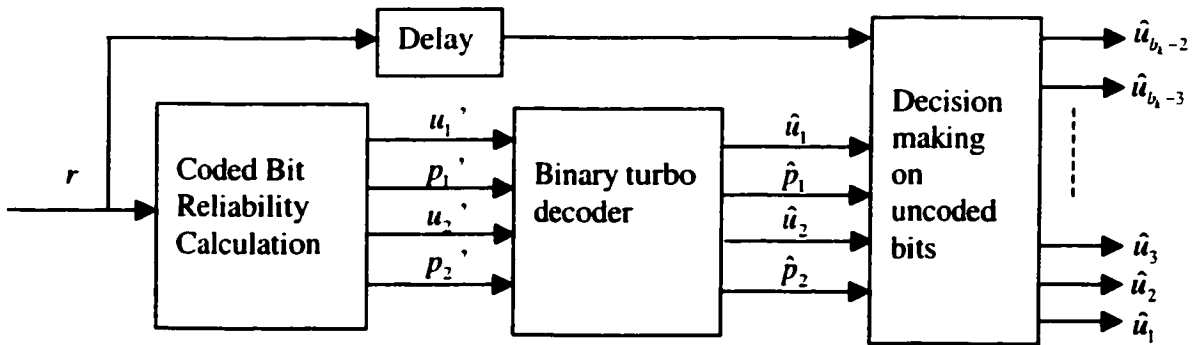


Figure 4-4– Simplified TTCM decoding structure

In Figure 4-4, the reliabilities of the turbo-coded bits (systematic and parity bits) are first calculated based on the received QAM symbols. With the reliability information as the input, a traditional binary MAP decoder is employed to perform the turbo decoding. The turbo coded bit reliability calculation is required only in the first iteration of the

decoding and the calculated reliability values can be stored and reused in the following iterations. The uncoded information bits can either be recovered at the last iteration of the turbo decoding, which guarantees that the decoder makes the most reliable decisions; or be recovered after a re-encoding of the decoded information bits, which not only results in suboptimal performance but also introduces extra decoding delay.

In the proposed TTCM encoding structure shown in Figure 4-2, four turbo-coded bits (either systematic or parity bits) are used to select the subsets of a QAM constellation. Since the in-phase and quadrature components are independent of each other, the reliabilities of the turbo-coded bits allocated to the in-phase component ($[u_1, p_1]$ as shown in Figure 4-3) and those allocated to the quadrature component ($[u_2, p_2]$ as shown in Figure 4-3) are also independent. Therefore, in the following, we will only discuss the 2-bit group in the in-phase component, $[u_1, p_1]$. The LLR of u_1 conditioned on the in-phase component, y_1 , can be expressed as:

$$LLR(u_1) = \ln \frac{p(u_1 = 1 | y_1)}{p(u_1 = 0 | y_1)} = \ln \frac{\sum_{u_1=1, x_1} p(y_1, x_1)}{\sum_{u_1=0, x_1} p(y_1, x_1)} = \ln \frac{\sum_{u_1=1, x_1} p(y_1 | x_1)}{\sum_{u_1=0, x_1} p(y_1 | x_1)} \quad \text{Eq. 4-14}$$

where the last step is based on the assumption of equal *a priori* probability of all possible transmitted in-phase components, x_1 .

Let's use the 64QAM constellation shown in Figure 4-3 as an example, where each dimension consists of an 8PAM constellation. Mapping of the two coded bits to the signal points is shown in Figure 4-5, where the solid circle denoted as y_1 represents the received symbol.

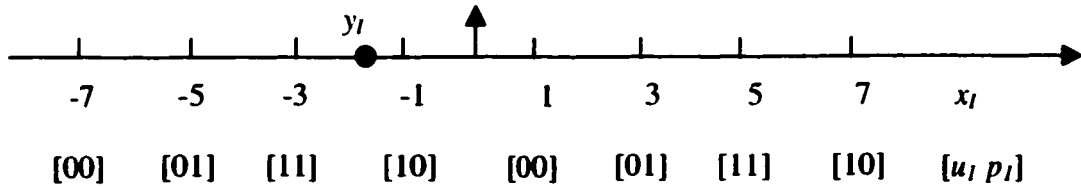


Figure 4-5 – Bit mapping of the in-phase component of a 64QAM constellation

Calculation of the ESBS with Eq. 4-14 involves large computation complexity such as exponential and logarithmic operations. In [17], the following approximation is given to reduce the complexity.

$$\begin{aligned}\ln(e^x + e^y) &= \max(x, y) + \ln(1 + e^{-|x-y|}) \\ &= \max(x, y) + f_c(|x-y|)\end{aligned}$$

Eq. 4-15

where $f_c(.)$ is implemented by a simple table lookup. It was shown that an 8-entry table is enough for negligible performance degradation. Therefore, Eq. 4-5 can be used to simplify the computation in Eq. 4-14. Details can be found in [17].

A further simplification can be made by observing that when an AWGN channel is assumed and the noise variance is not too large, only the four closest in-phase constellation points give the main contribution in the reliability calculation. In the example shown in Figure 4-5, these 4 points are [-5, -3, -1, 1]. Therefore, only four probabilities need to be calculated. This is based on the fact that the Gaussian distribution, $G(x)$ ⁹, decreases rapidly as x increases. The LLR of bit u_1 is therefore calculated as,

$$LLR(u_1) \cong \ln \frac{p(y_1 | -3) + p(y_1 | -1)}{p(y_1 | -5) + p(y_1 | +1)}$$

Eq. 4-16

We define an equivalent soft binary symbol (ESBS) of u_1 , $u_{1,s}$, as a received BPSK symbol in a Gaussian channel, where the same minimum constellation distance is the same as that of the QAM constellation and it produces the same LLR value as u_1 . With a single-sided AWGN noise variance of $N_{0,s}$, the LLR of the received binary symbol $u_{1,s}$ in a BPSK transmission is calculated as:

$$\begin{aligned}LLR(u_{1,s}) &= \ln \frac{p(u_{1,s} | +1)}{p(u_{1,s} | -1)} \\ &= \frac{4 \cdot u_{1,s}}{N_{0,s}}\end{aligned}$$

Eq. 4-17

Therefore, with the assumed $N_{0,s}$, the LLR of u_1 calculated from Eq. 4-16, $u_{1,s}$ can be calculated as:

$$u_{1,s} = \frac{N_{0,s}}{4} \cdot LLR(u_1)$$

Eq. 4-18

⁹ $G(x)$ represents a Gaussian distribution function with zero mean.

The ESBS of the other turbo-coded bits, (p_1, u_2, p_2) , are calculated with the same equation. These ESBSs will be fed into a conventional binary turbo decoder with the assumed AWGN noise variance, $N_{0,s}$. In this thesis, we will always use a $N_{0,s}=N_0$ and a minimum constellation distance of $d_{min}=2$.

A closer look at the mapping in Figure 4-5 is shown in Figure 4-6. The in-phase axis can be divided into nine parts. In the figure, the variable d_{eu} is defined as the Euclidean distance of the received signal point to the nearest constellation point on its left side; except in section A, where it is the distance between the received symbol and the nearest constellation symbol to its right, since there is no signal points to the left of section A.

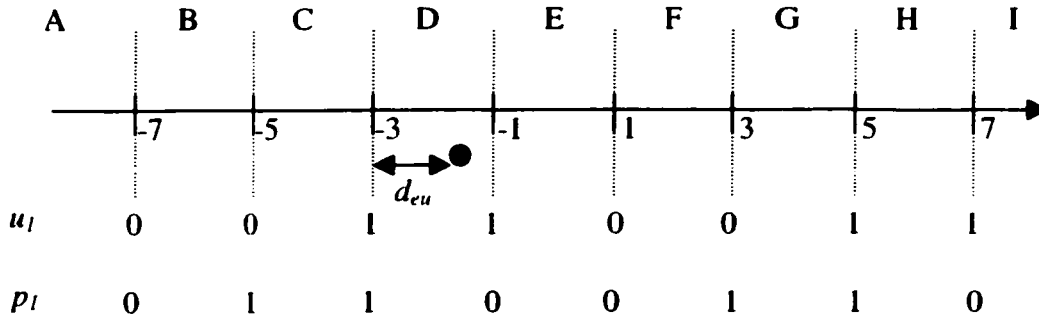


Figure 4-6 – Mapping of bits

It is observed that the ESBSs can be well approximated with some linear functions of d_{eu} . The linear equations for calculation of $u_{1,s}$ are listed in Eq. 4-19 for the nine regions shown in Figure 4-6, with an assumed AWGN noise variance of $N_0=1$.

K_c is a constant obtained from the simulation. Its value varies from 0 to 0.06 corresponding to N_0 from 0.5 to 1, which is the noise variance of our interest. The smaller is the noise variance, the better the linear approximations are. In TTCM systems, the N_0 values of interest to us are usually less than 0.8, which guarantees a good linear approximation.

It is observed from Eq. 4-19 that only four linear functions of the distance d_{eu} are needed for the calculation of the soft information, which in the above example are for the regions A, B, C, & D. These results can be easily extended to larger PAM constellations.

$$\begin{aligned}
 A: \quad u_{1,s} &= -(4 + K_c) - (2 - K_c) \cdot d_{eu} \\
 B: \quad u_{1,s} &= -(4 + K_c) + 1.5 \cdot d_{eu} \\
 C: \quad u_{1,s} &= -(1 + K_c) \cdot (1 - d_{eu}) \\
 D: \quad u_{1,s} &= \begin{cases} ((1 + K_c) + (1 - K_c) \cdot d_{eu}) & 0 \leq d_{eu} < 1 \\ (2 - (1 - K_c) \cdot (d_{eu} - 1)) & 1 \leq d_{eu} < 2 \end{cases} \\
 E: \quad u_{1,s} &= (1 + K_c) \cdot (1 - d_{eu}) \\
 F: \quad u_{1,s} &= \begin{cases} -((1 + K_c) + (1 - K_c) \cdot d_{eu}) & 0 \leq d_{eu} < 1 \\ -(2 - (1 - K_c) \cdot (d_{eu} - 1)) & 1 \leq d_{eu} < 2 \end{cases} \\
 G: \quad u_{1,s} &= -(1 + K_c) \cdot (1 - d_{eu}) \\
 H: \quad u_{1,s} &= (4 + K_c) - 1.5 \cdot d_{eu} \\
 I: \quad u_{1,s} &= (4 + K_c) + (2 - K_c) \cdot d_{eu}
 \end{aligned}$$

Eq. 4-19

Since the reliability calculations are all based on the linear equations shown in Eq. 4-19, as long as the minimum distance of the QAM constellation (normalized by the AWGN noise power) is a constant, the calculated reliability distribution is the same for different sizes of QAM constellations. This implies that various sizes of QAM constellations have similar decoding performance.

In ADSL DMT systems, during the system initialization process, the system needs to determine the optimal power allocation of the available bandwidth using a bit allocation algorithm. Traditional bit allocation algorithms assume a constant channel coding gain on all subchannels. This assumption is reasonable since a 4D-TCM used in traditional ADSL DMT systems results in nearly constant coding gains over the subchannels. However, this is not true for the TTCM schemes proposed in the literature, where TTCM with larger constellations results in much lower coding gains. In this case, the bit allocation algorithm has to know the coding gains of all subchannels in order to optimize the channel utilization. However, any traditional bit allocation algorithms could be employed in our turbo coded ADSL DMT systems because of the equal coding gain property.

4.1.3. Generalized pragmatic TTCM

A pragmatic approach to TCM is introduced in [27]. In this approach, a single rate-1/2 convolutional code is used and various bandwidth efficiencies are obtained through a mapping of two convolutional coded bits (corresponding to the output of one trellis transition during the convolutional encoding) and different number of uncoded bits into one multi-level transmission symbol. The coding and mapping structure is exactly the same as that shown in Figure 4-2, except there are only two coded bits from a rate-1/2 convolutional encoder $[w_1, w_2]$ instead of the four turbo coded bits $[u_1, p_1, u_2, p_2]$. If we assume that (b_k-2) uncoded bits are combined with two coded bits, the signal constellation of size 2^{b_k} is divided into 2^{b_k-2} sectors, each consisting of four adjacent constellation points. The (b_k-2) uncoded bits select the sector lexicographically (i.e., the binary vector whose decimal equivalent is j ($0 \leq j \leq 2^{b_k-2}-1$) selects the $(j+1)^{st}$ sector). An example of the mapping from a 4-bit group to a 16PAM symbol is shown in Figure 4-7, with coded bit pair $[c_1, c_2]$ (generated by u_1 from the rate-1/2 convolutional encoder) and uncoded bit pair $[u_2, u_3]$.

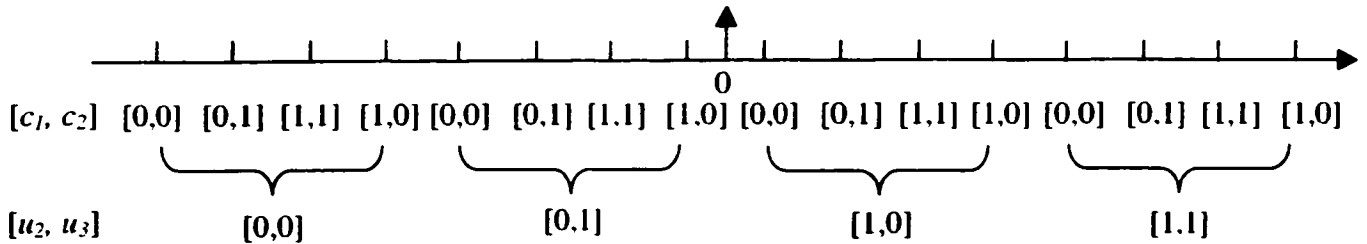


Figure 4-7 – Example constellation mapping of a pragmatic TCM for 16PAM

The constellation is divided into 4 4-point sectors, convolutional coded bit pair $[c_1, c_2]$ is used to determine the signal points inside each sector, uncoded bit pair $[u_2, u_3]$ is used to determine the sector

This pragmatic approach permits the use of a single basic convolutional encoder and Viterbi decoder to achieve reasonable coding gains for bandwidth efficiencies from 1 bit/sec/Hz to 6 bits/sec/Hz.

It can be readily seen that our proposed TTCM is a natural extension of the pragmatic TCM to a pragmatic TTCM. In the proposed TTCM, two rate-2/3 convolutional codes are used as CCs. Higher bandwidth efficiency is obtained by simply adding uncoded bits.

A single basic rate-2/3 convolutional encoder and binary MAP decoder could be employed.

This TTCM structure can be further generalized to use a rate-1/3 turbo code with rate-1/2 convolutional CCs. The higher rate turbo code can be obtained by puncturing, as shown in Figure 4-8. The advantage of this TTCM structure is its flexibility. Different TTCM schemes could be obtained by using various puncturing patterns. It should be noted that by puncturing the parity bits from the two CCs, it becomes a rate-1/2 turbo code with two rate-2/3 CCs. With the simplified TTCM decoding structure, a single rate-1/2 MAP decoder can be used for all the variations of this TTCM.

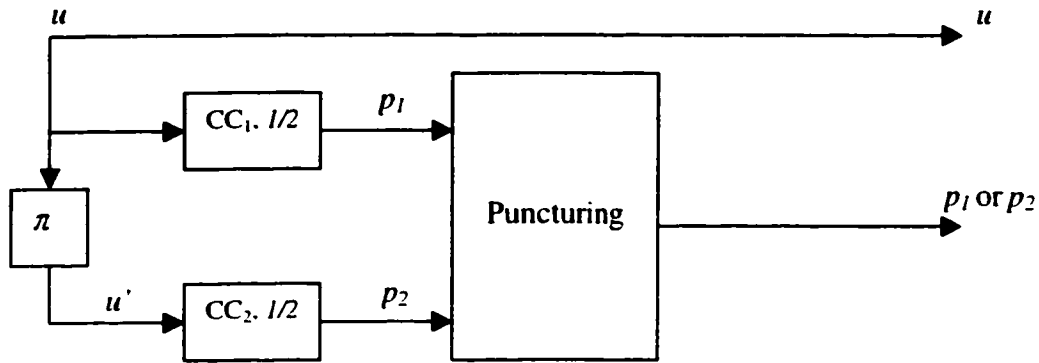


Figure 4-8 – Rate 1/2 turbo encoder with puncturing

One possible disadvantage of this TTCM is that the punctured rate-2/3 CCs could perform worse than the best available rate-2/3 CCs. Details of the performance differences can be found in [31].

4.2. PERFORMANCE ANALYSIS OF THE PROPOSED TTCM

In section 4.1, we proposed a TTCM structure for ADSL DMT transmissions. For low complexity implementations, the proposed TTCM structure utilizes a standard binary turbo decoder. The performance of this TTCM can therefore be estimated based on the binary turbo code performance over BPSK transmissions presented in the literature.

4.2.1. Gray mapping and the ESBS distribution

In the proposed TTCM, the ESBSs of the turbo-coded bits are obtained first from the noisy QAM symbols and then fed into a standard binary turbo decoder. However, it will

be shown in the following that these ESBSs no longer have a Gaussian distribution. Furthermore, as a result of the Gray mapping, different turbo-coded bits will have different distributions. Therefore, in order to estimate the TTCM decoding performance, we first need to know the distribution of the ESBS and its impact on the performance of a binary turbo decoder.

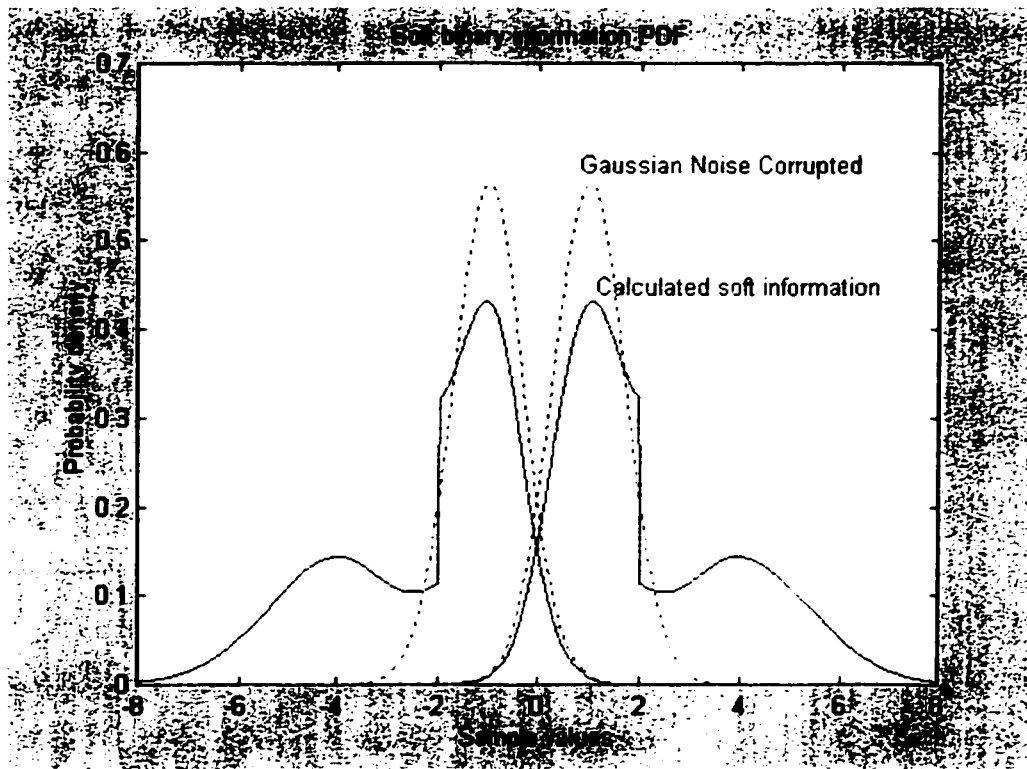


Figure 4-9 – PDF of ESBS of u_l , TTCM with 64QAM

The MAP algorithm is optimal in the sense of minimizing the symbol error probability in AWGN channels. Using the linear approximation in Eq. 4-19, the probability density function (PDF) of the ESBS of u_l is plotted in Figure 4-9, along with the Gaussian distribution of a received BPSK symbol in an AWGN channel. In this figure, a 64QAM modulation is assumed with an AWGN noise of variance 0.5. It is observed that the ESBS still has high probabilities around ± 1 . However, there are two other peaks around ± 4 . These two peaks are caused mainly by the edge constellation points (-7 and +7 as shown in Figure 4-6) on the two sides of the constellation. This

results in a larger mean value of ± 2.74 instead of ± 1 , and a larger variance. Distribution of bit p_l is also calculated, which is very close to the Gaussian distribution shown in Figure 4-9.

When a maximum likelihood (ML) decoder is used, it tries to find the code word closest to the received code sequence. For soft decision decoding, the Euclidean distance is the metric used to measure the distance between the received sequence and a known code word. For BPSK transmissions over an AWGN channel, an error code word of Hamming weight w gives a Euclidean distance of $2w^{10}$. However, when decoding with ESBSs calculated in Eq. 4-18, the average Euclidean distance for the same error sequence becomes,

$$\begin{aligned}
 D(w) &= \sum_{i=1}^w \text{mean}\{d_i\} \\
 &= \sum_{i=1}^w (p(u_i) \text{mean}\{d_i | u\} + p(p_i) \text{mean}\{d_i | p\}) \\
 &= \frac{1}{2} \cdot \sum_{i=1}^w (\text{mean}\{d_i | u_i\} + \text{mean}\{d_i | p_i\}) \\
 &= \frac{1}{2} \cdot \sum_{i=1}^w (5.48 + 2) \\
 &= 3.74w
 \end{aligned}$$

Eq. 4-20

where $p(u_i)$ and $p(p_i)$ are the probabilities of the error location falling on the systematic bits (u_i bits) and parity bits (p_i bits).

Therefore, we conclude that, when an ML decoder is used, the performance of decoding with the ESBSs of the 64QAM transmission should be at least better than that of a turbo coded BPSK transmission having the same constellation minimum distance, d_{min} over the same AWGN channel.

For illustration purpose, let us use the example shown in Figure 4-6. It is assumed that all constellation points have an equal probability of occurrence. When the symbol X is transmitted, the received symbol is $Y = X + n_0$, where n_0 is the AWGN noise component. When $X = -7$, the ESBS of u_l is calculated according to Eq. 4-18 as,

¹⁰ Here, we are assuming the minimum distance of 2 for both BPSK transmission and the 64QAM transmission.

$$u_{1,s} = \frac{N_0}{4} \log \left(\frac{e^{-\frac{(Y+3)^2}{N_0}}}{e^{-\frac{(Y+7)^2}{N_0}} + e^{-\frac{(Y+5)^2}{N_0}}} \right) \approx \begin{cases} n_0 - 3, & 1 \ll n_0 < 3 \\ 2n_0 - 4, & n_0 \ll 1 \end{cases}$$

Eq. 4-21

In the case when $n_0 \ll 1$, $u_{1,s}$ has a high probability of having a large negative value. This case can be viewed as a BPSK transmission with a minimum distance of 8, and an AWGN variance of $4N_0$. This, in turn, is equivalent to a BPSK transmission with a $d_{min}=4$ and an AWGN variance of N_0 . Therefore, when the symbols $\{\pm 7\}$ are transmitted, the bit u_1 is better protected because its ESBS symbol has at least a doubled minimum distance. In Table 4-1, two possible states of bit u_1 are shown along with their corresponding minimum distances. In state-0, the equivalent BPSK signal is better protected because the minimum distance is doubled. However, in state-1, the ESBS has only a minimum distance of 2, which is the same as the minimum distance of the QAM constellation. For bit p_1 shown in Figure 4-6, only one state exists corresponding to the state-1 of u_1 .¹¹

Table 4-1 – Protection states of the turbo coded bits $[u_1, p_1]$ with the constellation mapping shown in Figure 4-5

State	Const. Points	Occurrence prob.	d_{min}	Noise variance
Bit I_1				
0	-7, 7	$\frac{1}{4}$	4	σ_0^2
1	-5, -3, -1, 1, 3, 5	$\frac{3}{4}$	2	σ_0^2
Bit I_2				
0	All	1	2	σ_0^2

With larger constellations where uncoded bits are added, such as that shown in Figure 4-7, the advantage of the better protection of bit u_1 vanishes. This is because the better protection is obtained only when the edge symbols are transmitted and the possibility of transmitting the edge symbols is inverse proportional to the constellation size. The distribution converges to one that is very similar to a Gaussian distribution, as shown in

¹¹ Note that, for the second state of bit u_1 , the performance should be slightly worse than that of BPSK transmission, since there can be negative decision area on both side of the transmitted symbol. The effect is neglected here since the negative decision area is usually far from the transmitted signal and the received signal has a small probability to fall into these areas.

Figure 4-10. It is therefore concluded that, the performance of the proposed TTCM system using Gray mapping can be estimated (or upper bounded) by the corresponding performance of turbo coded BPSK transmissions with the same Gaussian noise.

Iterative decoding is employed in turbo decoding to approach the ML decoding performance with much smaller complexity. As shown in Figure 4-9, the ESBS of $u_I, u_{I,S}$, has a greater probability to take higher values. As a consequence, the SISO decoder should give larger likelihood ratios (absolute values) because it still considers the soft information to be Gaussian distributed. Therefore, the performance should also be at least better than that of a turbo-coded BPSK transmission with the same minimum constellation distance and over the same AWGN channel.

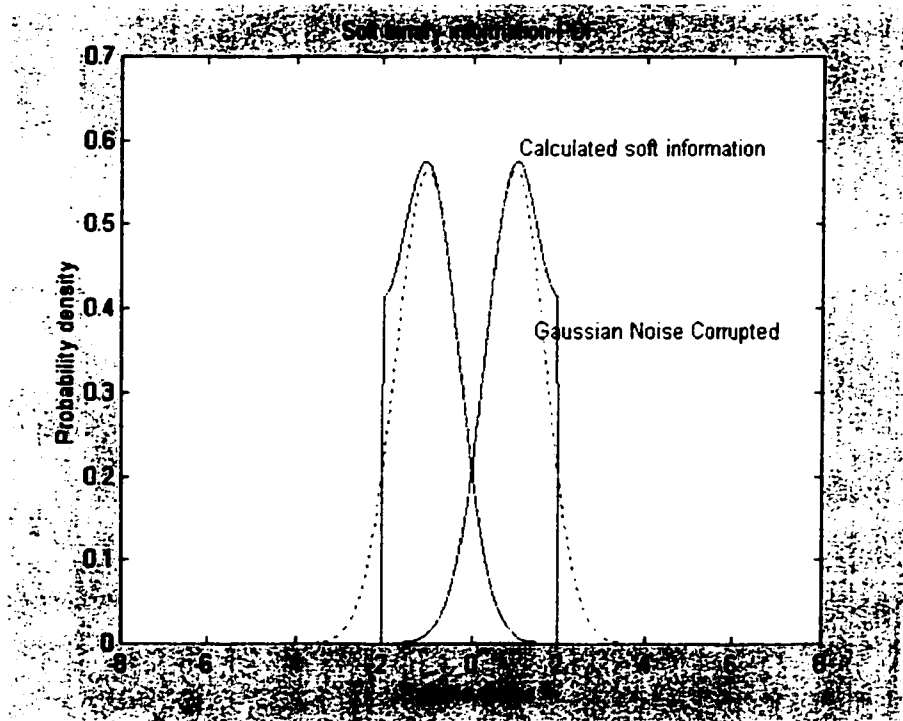


Figure 4-10 – PDF of the ESBS of u_I for very large QAM constellations

The above analysis is based on using 2-bit Gray mapping, where the Gray mapping is applied to each subsection of 16 constellation points, as shown in Figure 4-3. The analysis can be further extended to TTCM schemes using larger subsections. For example, we could use subsections of 64 constellation points, where each dimension of a

subsection becomes an 8PAM. To keep the number of redundant bits per symbol constant, a rate-2/3 turbo code should be used instead of the rate-1/2 turbo code. One dimension of a TTCM system with 256QAM constellation is shown in Figure 4-11. On each axis, the coordination of each signal point in each subsection is selected by the three coded bits $[c_1, c_2, c_3]$ from the rate-2/3 turbo encoder, generated from 2 information bits $[u_1, u_2]$. A third information bit, u_3 , is then used to select the subsection.

Obviously, among the three bits $[c_1, c_2, c_3]$, c_1 is the best protected, while c_2 also has a good chance of having better protection. The other bit, c_3 , has a similar protection as a BPSK transmission since there is a '0' and a '1' on either side, no matter what signal point is transmitted. The protection states of these turbo-coded bits are obtained in a similar way as for a 2-bit Gray mapping and the results are plotted in Table 4-2.

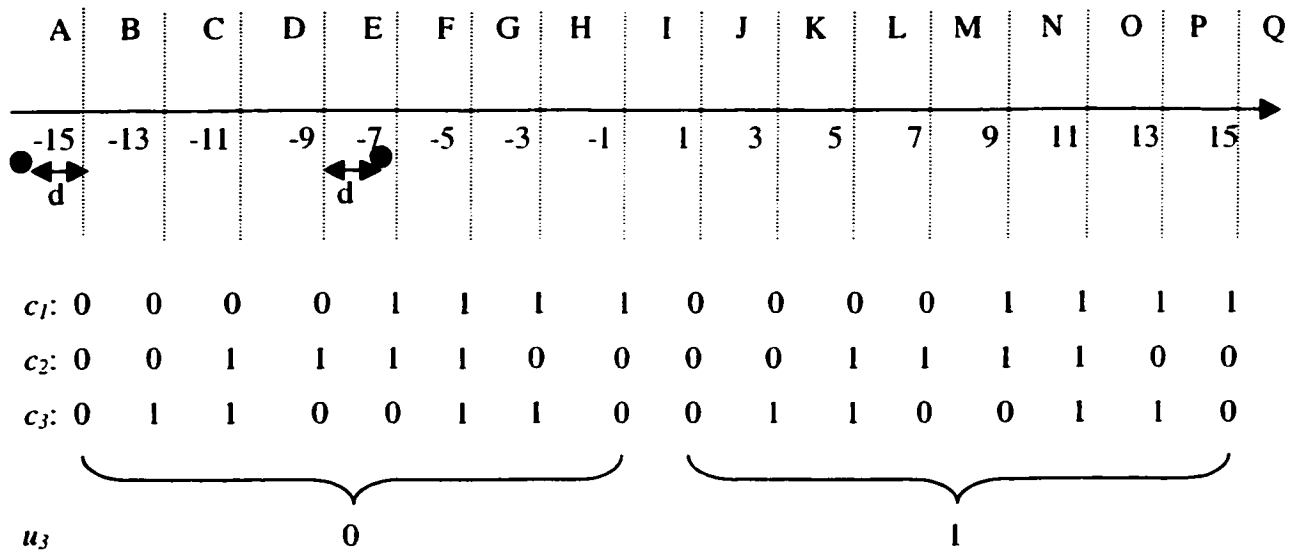


Figure 4-11 – 3-bit Gray mapping of 256QAM TTCM along one dimension

In-phase (quadrature) axis of a 256QAM, an overall bandwidth efficiency of $\frac{3}{4}$, the three turbo-coded bits are $[c_1, c_2, c_3]$ are used to select coordination in the two subsections according to 3-bit Gray mapping, the uncoded bit u_3 determines the subsection.

A rate-2/3 turbo code usually consists of two identical rate-4/5 CCs, where there is one parity bit for every four systematic bits generated by each CC. Since the reliability accumulation in iterative decoding depends heavily on the trellis structure and the parity bits, the parity bits will need better protection. Therefore, a straightforward method

would be to assign one parity bit and two systematic bits to each single-dimensional PAM symbol, where the parity bit is allocated to c_1 and the systematic bits to c_2 and c_3 .

Table 4-2 – Protection states of the turbo-coded bits $[c_1, c_2, c_3]$ in each dimension of 256QAM, with the mapping of each dimension as shown in Figure 4-11

State	Const. Points	Prob.	d_{min}	Noise variance
Bit c_1				
0	± 15	1/8	8	σ_0^2
1	± 13	1/8	6	σ_0^2
2	$\pm 11, \pm 5, \pm 3$	3/8	4	σ_0^2
3	$\pm 9, \pm 7, \pm 1$	3/8	2	σ_0^2
Bit c_2				
0	$\pm 15, \pm 9, \pm 7, \pm 1$	1/2	4	σ_0^2
1	$\pm 13, \pm 11, \pm 5, \pm 3$	1/2	2	σ_0^2
Bit c_3				
0	All	1	2	σ_0^2

4.2.2. BER upper bound with an ML decoder

In [14] and [29], Benedetto proposed using a *uniform interleaver* to obtain the BER upper bound of binary turbo codes with a maximum-likelihood (ML) decoder. A *uniform interleaver* is defined as a device that maps a given input sequence of length K and weight w into all possible distinct permutations with equal probability. The union bound derived based on this *uniform interleaver* provides an upper bound on the performance of all possible interleavers, which guarantees that at least one actual interleaver will perform better than the bound obtained by using the *uniform interleaver*.

When the encoders of both CCs of a turbo code start from the zero state and end into the zero state, the performance of this turbo code is equivalent to a block code having the same block length and the same code rate. In this case, the BER union bound of a binary turbo code can be expressed as:

$$P_b(e) \leq \sum_{w=w_{min}}^K \frac{w}{K} W^w A^{C_r}(w, Z) \Big|_{W=Z=e^{-R_r E_b / N_0}} \quad \text{Eq. 4-22}$$

where $A^{C_r}(w, Z)$ is the conditional weight enumeration function (CWEF) of all code words generated by weight- w input words, w_{min} denotes the minimum information weight in the error events of the CC and K is the length of the interleaver.

When a *uniform interleaver* is employed, the CWEF of the turbo code can be expressed as,

$$A^{C_r}(w, Z) = \frac{[A^C(w, Z)]^2}{\binom{K}{w}} \quad \text{Eq. 4-23}$$

where $A^C(w, Z)$ is the conditional weight enumeration function of each CC with the input word weight of w .

An asymptotic expression of the upper bound for a rate-1/3 turbo code with two identical rate-1/2 CCs was obtained in [29] as:

$$P_b(e) \leq \frac{1}{K} \sum_{k=1}^{\lfloor K/2 \rfloor} 2k \binom{2k}{k} W^{2k} \frac{Z^{2k z_{\min}}}{(1 - Z^{z_{\min} - 2})^{2k}} \bigg|_{W=e^{-R_c E_{bt}/N_0}, Z=e^{-R_c E_{tp}/N_0}} \quad \text{Eq. 4-24}$$

where it is assumed that the BER contribution from weight-(2k+1) information words are negligible compared to the BER contribution from weight-2k information words. $z_{\min}$ is the minimum weight of the parity check bits in error events generated by information sequences of weight 2 from one CC. Therefore, the minimum code word weight generated by a weight-2 information word is,

$$d_{free,eff} = 2 + 2 \cdot z_{\min} \quad \text{Eq. 4-25}$$

which is defined as the *effective free distance* ($d_{free,eff}$) of the turbo code.

In our TTCM systems with Gray mapping, Eq. 4-22 must be changed because the soft binary symbols of the bits assigned to different bit positions have different minimum Euclidean distances and the E_b/N_0 will not have a constant value over all the bits.

Assume an additional random interleaver is inserted between the turbo encode and the bit-to-symbol mapper, such that all the bits, systematic and parity bits, have the same probability to be assigned to any protection levels. As shown in previous section, these bits have a probability to be better protected depending on which QAM symbols are transmitted. Using the constellation mapping shown in Figure 4-5 as an example, the turbo-coded bit u_i has a probability of 1/4 to be better protected. With the assumption of this additional random interleaver between the turbo encoder and the constellation

mapper, the turbo-coded bit (u_i or p_i) will each have a probability of 1/8 of being better protected.

At the decoder, when an ML decoder is employed, the Euclidean distance of an error word of Hamming weight w is now a random variable having a distribution of,

$$p[d' = ((w - w_1) \cdot d_0 + w_1 \cdot d_1)] = \binom{w}{w_1} (1 - 1/8)^{w-w_1} \cdot (1/8)^{w_1} \quad \text{Eq. 4-26}$$

where w_1 is the number of better-protected bits, d_0 and d_1 are the distance contributions from a bit error falling on a normal protected bit and a better protected bit respectively.

As mentioned in the last section, there are reasons to assign parity bits to the better-protected positions. One reason is that to improve the iterative decoding performance, parity bits need better protection when a large number of them are punctured. Another reason is that most error words from a turbo decoder contain more parity bits than systematic bits. This is obvious when we look at the weight spectrums shown in Table 4-3 later in this section.

With the minimum Euclidean distance distribution shown in Eq. 4-26, the union bound becomes,

$$P_b(e) \leq \frac{1}{K} \left(\sum_{k=1}^{\lfloor K/2 \rfloor} 2k \binom{2k}{k} \cdot \sum_{w_1=0}^{2k(1+z_{min})} p(w_1) \cdot \frac{Z^{2k(1+z_{min})-w_1} \cdot Z^{w_1}}{(1 - Z^{z_{min}-2})^{2k}} \right) \bigg|_{Z=e^{-K_t E_{b0}/N_0}, Z_1=e^{-K_t E_{b1}/N_0}} \quad \text{Eq. 4-27}$$

where $p(w_1)$ is the probability of w_1 errors falling on the better protected positions.

The BER upper bounds calculated from Eq. 4-27 is shown in Figure 4-12, based on rate-1/3 turbo codes with rate-1/2 CCs. It shows the BER upper bounds of the turbo-coded bits in the 64QAM TTCM with 3 different turbo codes. These 3 turbo codes are differentiated by the z_{min} values of their CCs, which are 3, 6 and 10 respectively. For the purpose of comparison, the BER upper bound of turbo-coded BPSK transmissions with the same CCs are also shown in this figure.

In Figure 4-12, it is observed that the BER upper bound of the turbo-coded bits in a 64QAM TTCM with 2-bit Gray mapping (as shown in Figure 4-6) is always better than a corresponding BPSK transmission with a same minimum constellation distance over the same AWGN channel. However, the x -axis is plotted as the E_b/N_0 values of the BPSK transmission. The 3 curves for 64QAM TTCM with rate-1/3 turbo code are redrawn in

Figure 4-13 with the correct E_b/N_0 values for 64QAM constellations. The reader is advised that these curves are only valid for the turbo-coded bits that are mapped to the QAM symbols.

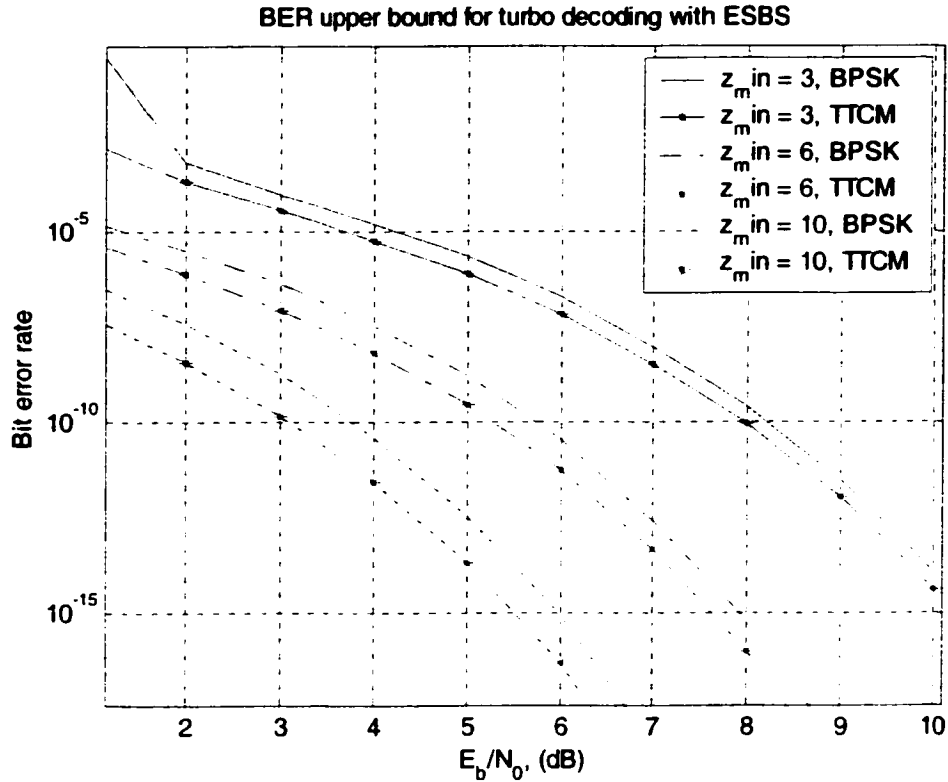


Figure 4-12 – BER upper bound of the turbo-coded bits in a TTCM

64QAM TTCM, each symbol contains 4 turbo-coded bits and 2 uncoded bits, the constellation mapping is carried out as shown in Figure 4-6, except that mapping of the systematic and parity bits to the symbols are totally random, a rate-1/3 turbo code is assumed instead of a rate-1/2 turbo code since it is very hard to obtain a closed-form expression of the BER upper bound for a punctured rate-2/3 CC, the E_b/N_0 values are calculated assuming BPSK transmission.

To obtain an upper bound for rate-1/2 turbo codes employing rate-2/3 CCs or punctured rate-1/2 CCs, the weight enumeration functions (WEF) of the CCs need to be derived first. The readers can refer to [30] for details of the WEF derivation of punctured convolutional codes. For simplicity, the derivation of WEF for high-rate convolutional codes obtained from punctured rate-1/2 convolutional codes will be skipped. In [31], a search for good CCs for various rate turbo codes was presented based on recursively

optimizing the (d_w, N_w) pairs, where d_w is the minimum Hamming weight of convolutional code words generated by an information bit sequence of weight w and N_w is the number of such sequences. As a result, various rate turbo codes with the best CCs were given, together with their (d_w, N_w) pairs for input information sequences having weight-2 to weight-6. This actually gives the first five items of the WEF of the CCs. The low-weight parts of the CWEF of turbo code assuming a *uniform interleaver* can be derived from Eq. 4-23. For moderate to high SNR values, the BER performance of turbo codes mainly depend on the first few items since they provide the lowest Hamming-weight code words.

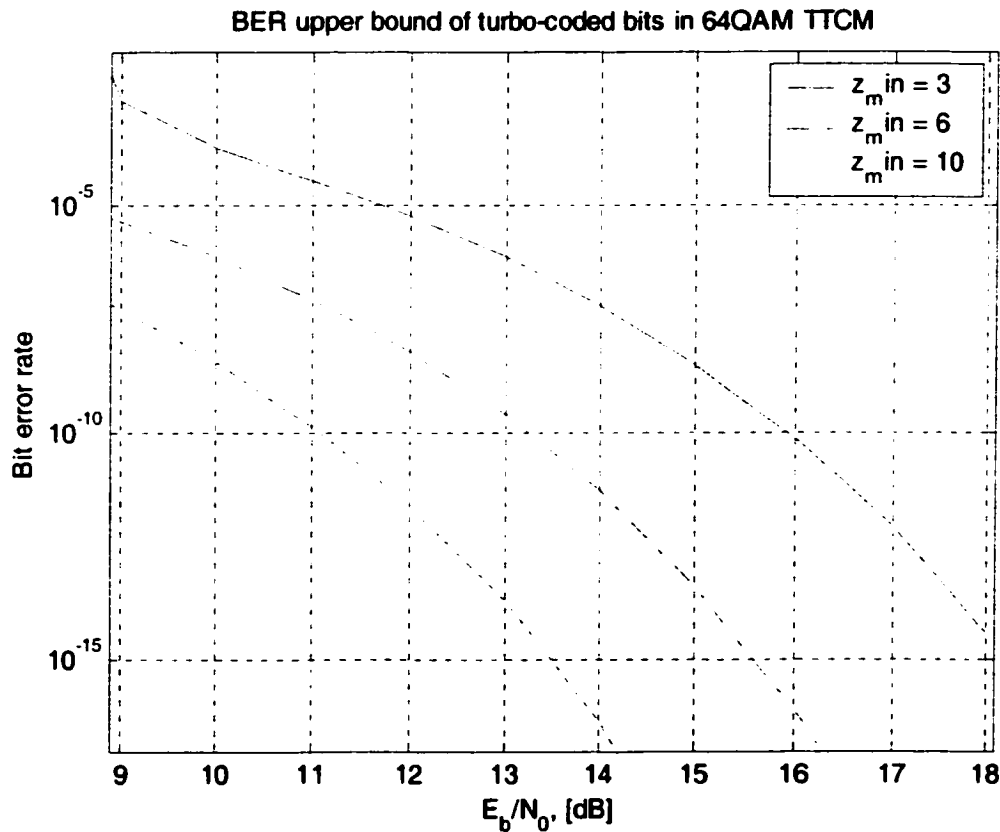


Figure 4-13 – BER upper bound of the turbo-coded bits in a 64QAM TTCM

64QAM TTCM, each symbol contains 4 turbo-coded bits and 2 uncoded bits, the constellation mapping is carried out as shown in Figure 4-6, except that mapping of the systematic and parity bits to the symbols are totally random, a rate-1/3 turbo code is assumed instead of a rate-1/2 turbo code since it is very hard to obtain a close expression of the BER upper bound for a punctured rate-2/3 CC.

Table 4-3 – Distance spectrum performance of the best CCs

	d_2, N_2	d_3, N_3	d_4, N_4	d_5, N_5	d_6, N_6
<i>1/2, 16-state</i>	12, 1	8, 3	6, 1	10, 14	8, 1
<i>2/3, 16-state</i>	8, 2	5, 3	6, 9	7, 29	6, 2
<i>4/5, 8-state</i>	4, 2	3, 1	5, 8	5, 4	6, 11

The first term of the WEF of the CCs with an input weight- w are shown in Table 4-3. We will assume that only this term will give most BER contribution. With a *uniform interleaver*, the corresponding term in the CWEF of the turbo code will be,

$$d_w^{TC} = \frac{K^2 \cdot N_w^2}{\binom{K}{w}}$$

Eq. 4-28

where there are a total of $C(K, w)$ combinations for the input word to the second CC. Note that by shifting the w -bit input pattern by m , i.e., the input sequence has the first “1” at position $(m+1)$, the same CWEF contribution is obtained. The maximum number of the shifting is $K-l$, where l is the length from the first “1” to the last “1”. Therefore, an additional multiplication by K is applied to obtain an upper bound. The CWEF is then obtained by applying Eq. 4-28 to all the entries in Table 4-3. Finally, the BER upper bound is then calculated from the obtained CWEF of the turbo codes, which is plotted in Figure 4-14.

It is observed that the BER upper bounds of the turbo-coded bits in the 64QAM TTCM transmissions are always better than the corresponding turbo-coded BPSK transmissions. The BER upper bounds for the 64QAM TTCM transmissions are re-plotted in Figure 4-15 with the correct E_b/N_0 values. It should be noted that the bandwidth efficiencies of the 64QAM TTCM with the three CCs shown in Table 4-3 vary from 3.3 bits/symbol to 4.6 bits/symbol.

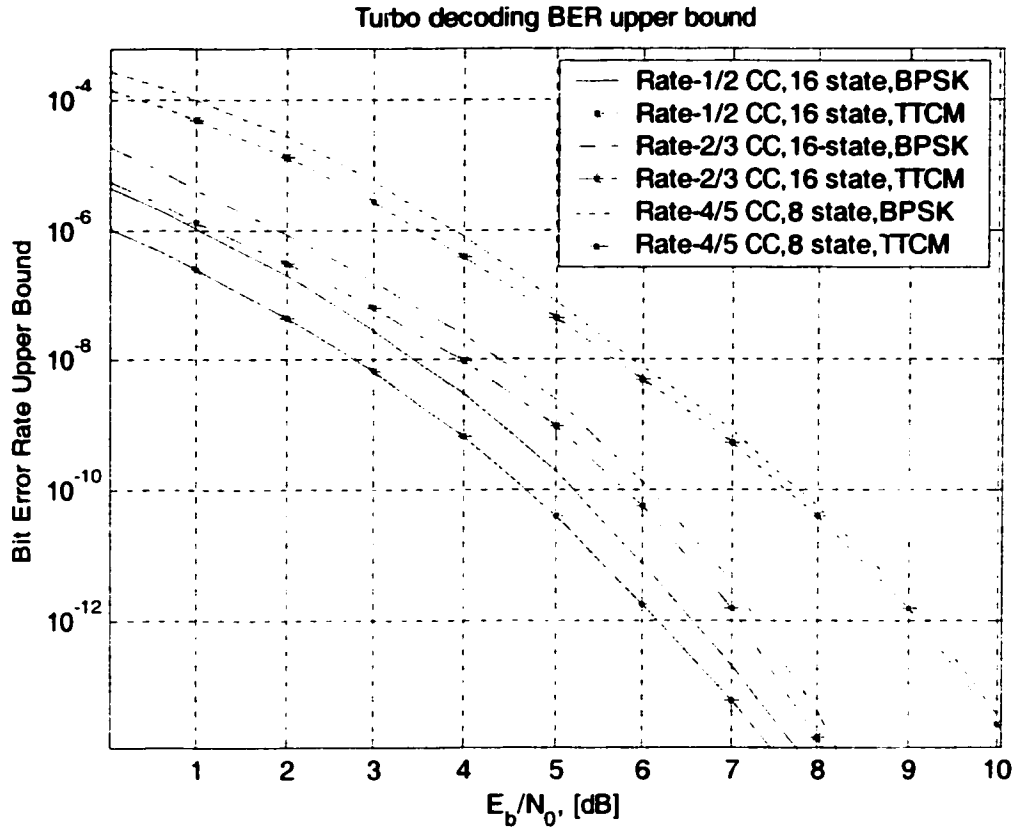


Figure 4-14 – BER upper bound of the turbo-coded bits in a TTCM

64QAM TTCM, each symbol contains 4 turbo-coded bits and 2 uncoded bits, the constellation mapping is carried out as shown in Figure 4-6, except that mapping of the systematic and parity bits to the symbols are totally random, the E_b/N_0 values are calculated assuming BPSK transmission, 1024-bit turbo code interleaver.

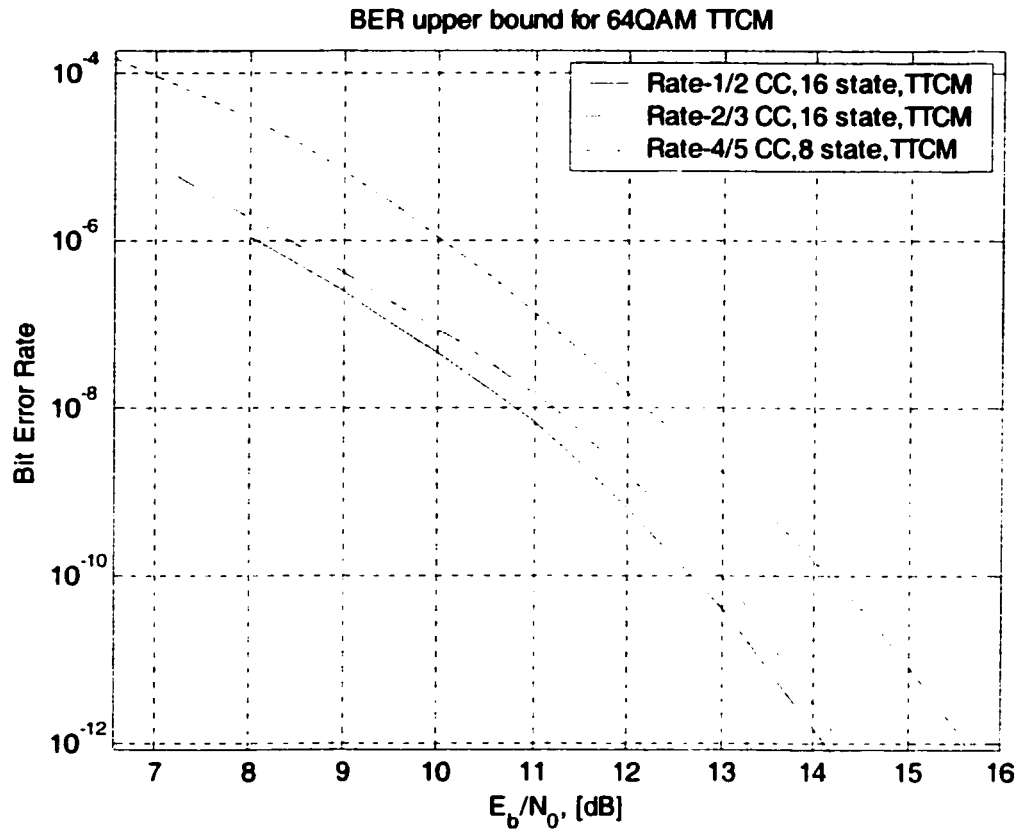


Figure 4-15 – BER upper bound of the turbo-coded bits in a TTCM

64QAM TTCM, each symbol contains 4 turbo-coded bits and 2 uncoded bits, the constellation mapping is carried out as shown in Figure 4-6, except that mapping of the systematic and parity bits to the symbols are totally random, 1024-bit turbo code interleaver.

4.2.3. Performance estimation of TTCM based on binary turbo code performance

In this section, the performance of our TTCM system is estimated based on the best-reported performances in the literature. We will assume that iterative decoding achieves performance very close to the ML decoding. These results will shed some light on the actual achievable coding gain by using TTCM with different QAM constellations.

For TTCM with 64QAM transmissions, it was shown above in Table 4-1 that, statistically, $1/4$ of b_I bits have an equivalent BPSK d_{min} of 4. Assuming a rate-1/2 turbo code and the constellation mapping shown in Figure 4-6 are used, half of the bits are u_I bits and the others are p_I bits; therefore, $1/8$ of the total bits have this doubled d_{min} . The pair-wise word error probability of an ML decoder can be expressed as

$$p(m) = \sum_{m_I=0}^m p(m_I) Q\left(\frac{m_I \cdot d_I + (m - m_I) \cdot d_0}{\sqrt{m \cdot N_0/2}}\right) \leq Q\left(\frac{m \cdot d_0}{\sqrt{m \cdot N_0/2}}\right) \quad \text{Eq. 4-29}$$

where m is the total number of error bits, m_I is the differing positions on the error path containing better protected bits. $p(m_I)$ is the probability of m_I out of m erroneous bits are better protected, which can be calculated according to Eq. 4-26. d_I and d_0 are the Euclidean distance contributions resulted from errors on better protected bits and normally protected bits.

The coding gain estimations, G , for different QAM transmissions employing the proposed TTCM with a rate-1/3 turbo code are shown in Table 4-4, based on simulation results of a binary rate-1/3 turbo code in BPSK transmissions. Our simulation showed that with an AWGN noise variance of 1.2, a BER of 10^{-5} could be achieved for a BPSK transmission using a rate-1/3 binary turbo code. We already know that, with this same AWGN noise, the BER of the turbo-coded bits of a 64QAM TTCM with 2-bit Gray mapping is as least as good as the binary turbo coded BPSK transmission. In Table 4-4, the 3rd row gives the γ_b values of various QAM constellations with an AWGN variance of 1.2. It is assumed that there are always 4 turbo-coded bits generated from a rate-1/3 turbo encoder allocated to each QAM symbol.

In [32], with a combined CC and interleaver design, a rate-1/2 turbo code with an interleaver of 1024 bits yields a BER of 10^{-5} with γ_b of 1.4 dB, where it also showed that without the CC-interleaver combination optimization, a γ_b of 1.6 dB is needed to obtain the same BER. Here, we will use 1.4 dB for optimal results. Performance bounds of a rate-1/2 turbo code are also obtained and summarized in Table 4-5.

Table 4-4 – Minimum coding gains of turbo-coded bits in various QAM TTCM

2-bit Gray mapping, rate-1/3 turbo code, $K=1024$, AWGN with $N_0=2.4$,
 G is the coding gain, the results are for turbo-coded bits only

N_s (bits)	6	7	8	9	10	11	12	13	14	15
K_s (bits)	3.3	4.3	5.3	6.3	7.3	8.3	9.3	10.3	11.3	12.3
E_b/N_0 (dB)	7.20	9.11	11.23	13.51	15.88	18.34	20.86	23.43	26.04	28.68
Uncoded E_b/N_0 requirement (dB)	12.23	14.22	16.35	18.57	20.92	23.31	25.79	28.30	30.88	33.46
G (dB)	5.03	5.11	5.12	5.06	5.04	4.97	4.93	4.87	4.84	4.78

Table 4-5 – Minimum coding gains of turbo-coded bits in various QAM TTCM

2-bit Gray mapping, rate-1/2 turbo code, $K=1024$, AWGN with $N_0=1.45$,
 G is the coding gain, the results are for turbo-coded bits only

N_s (bits)	6	7	8	9	10	11	12	13	14	15
K_s (bits)	4	5	6	7	8	9	10	11	12	13
E_b/N_0 (dB)	8.60	10.67	12.91	15.26	17.69	20.19	22.75	25.34	27.98	30.64
Uncoded E_b/N_0 requirement (dB)	13.54	15.62	17.84	20.14	22.5	24.98	27.46	30.04	32.6	35.23
G (dB)	4.94	4.95	4.93	4.88	4.81	4.79	4.71	4.70	4.62	4.59

The performance of a best rate-2/3 turbo code obtained from a punctured rate-1/3 turbo code is shown in [37], where the searching is over all the puncturing patterns. The coding gain lower bounds derived from the simulation results presented in [37] are shown in Table 4-6.

Table 4-6 – Minimum coding gains of turbo-coded bits in various QAM TTCM

3-bit Gray mapping, rate-2/3 turbo code, $K=10000$, AWGN with
 $N_0=0.97$, G is the coding gain, the results are for turbo-coded bits only

N_s (bits)	8	9	10	11	12	13	14	15
K_s (bits)	6	7	8	9	10	11	12	13
E_b/N_0 (dB)	14.66	17.00	19.44	21.94	24.49	27.09	29.72	32.38
Uncoded E_b/N_0 requirement (dB)	17.84	20.14	22.5	24.98	27.46	30.04	32.6	35.23
G (dB)	3.24	3.14	3.06	3.04	2.97	2.95	2.88	2.95

It should be noted that when a rate-2/3 turbo code is used, 3-bit Gray mapping could be employed as shown in Figure 4-11. The coding gain performance shown in Table 4-6 does not take into account the much better error protection on certain bits in the 3-bit Gray mapping. Therefore, these coding gain lower bounds are pretty loose.

It was shown in Table 4-2 that there are four levels of protection in a 3-bit Gray mapping. Let us still assume that a random interleaver is employed between the rate-2/3 turbo encoder and the constellation mapper, every turbo-coded bits has an equal probability of being allocated to any bit positions of any protection level. Therefore, the probability of making a decoding error with an error word of weight- w can be calculated as,

$$p(m) = \sum_{m_1=0}^m \sum_{m_2=0}^{m-m_1} \sum_{m_3=0}^{m-m_1-m_2} p(m_1, m_2, m_3) Q \left(\frac{m_1 \cdot d_1 + m_2 \cdot d_2 + m_3 \cdot d_3 + (m - m_1 - m_2 - m_3) \cdot d_0}{\sqrt{m \cdot N_0/2}} \right)$$

Eq. 4-30

where m is total number of error bits, m_1, m_2 and m_3 are the number of erroneous bits falling on the bit positions of protection states of 0, 1 and 2, as shown in Table 4-2, and $(m - m_1 - m_2 - m_3)$ is the number of error bits falling on bit positions of protection state of 4. $p(m_1, m_2, m_3)$ is the probability of this particular error event.

The word error probability of the code words of weight-3 to 6 are calculated and presented in Figure 4-16. It is observed that the ML decoding using ESBSs from 3-bit Gray mapping performs at least 0.5 dB better than using received BPSK symbols, for error words of weight-3. This gain improves to more than 1.0 dB for error words of weight-6 and more than 2 dB for error words of weight-10 (not shown in the figure). The rate-2/3 turbo code shown in Table 4-3 has only one code word of weight-3. Furthermore, for low E_b/N_0 values, the BER performance is also determined by moderate-weight error code words with large multiplicities. Therefore, it is expected that using ESBS from a 3-bit Gray mapping could provide coding gain advantage much higher than 0.5 dB.¹²

¹² Other the other hand though, this gain was achieved based on ML decoder. The assumption of ML decoding performance of an iterative decoder at low SNR values has not been validated.

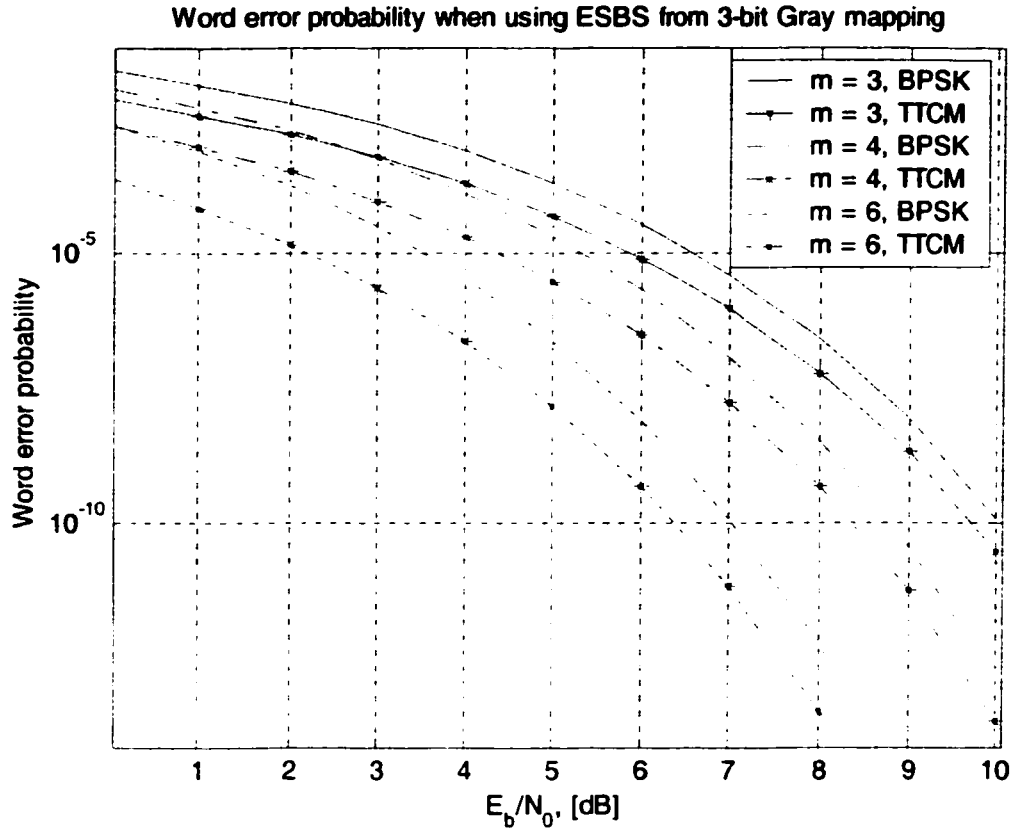


Figure 4-16 – Word error probability of ML decoding with ESBSs

3-bit Gray mapping, m is the weight of the word error, turbo-coded bits only, E_b/N_0 values are calculated as for BPSK transmissions

4.2.4. Performance improvement with a specific interleaver

In previous sections, it was assumed that the constellation mapping is performed randomly, i.e., all bits in one turbo block have the same probability of being assigned to bit positions with any protection levels. However, when the weight spectrum of a turbo code is known and the most vulnerable positions can be located, allocating better protected bit positions during the constellation mapping to these vulnerable positions could provide better performance than allocating the better protected bit positions randomly to any turbo coded bits.

As an example, for a rate-1/2 turbo code with a specific S-random interleaver, the weight spectrum of the code words generated from weight-2 and weight-3 information word is shown in Table 4-7.

Table 4-7 – Weight distribution of a rate-1/2 turbo code, $K=100$

$\begin{smallmatrix} d_w \\ w \end{smallmatrix}$	9	10	11	12	13	14	15	16	17	18	19	20
2	0	0	1	1	2	2	2	4	5	6	8	11
3	1	0	1	1	6	4	15	6	30	20	45	41

It was shown in [29] that the performance of a turbo code at moderate and high SNR is determined mainly by the low-weight code words generated from weight-2 (or low-weight) information words. Therefore, improving the protection on the bits contained in these low-weight code words will, therefore, improve the overall turbo decoding performance.

For example, in a 64QAM with a rate-1/2 turbo code with 2-bit Gray mapping, we can improve the turbo decoding performance by ensuring that the bits on the error positions in the low-weight error words shown in Table 4-7 are allocated to the better-protected bit position u_1 . We observe from Table 4-7 that for turbo code words of weight smaller than 15 generated from weight-2 and weight-3 information words, there are a maximum of 243 bits in one turbo block that need to be better protected. Both the transmitter and receiver have to know these positions to perform the constellation mapping and demapping. Note that this is only useful in moderate or high SNR situations. In the case of low SNR, the multiplicities of medium-weight error words also play a significant role in determining the BER performance. In this case, it is expected that randomly allocating the better-protected positions will provide a reasonable performance improvement.

4.2.5. Performance balancing between turbo-coded bits and uncoded bits

The performance analysis so far is for the turbo-coded bits. The uncoded bits are protected by the large intra-subset constellation distance provided by set partitioning.

The proposed QAM transmission with TTCM can be viewed as a two-level channel coding, where the lower level bits are turbo coded and no coding is applied to the higher-level bits. A properly designed QAM transmission with this two-level TTCM structure should meet the following criteria:

- The BER of the turbo-coded bits (lower-level) should meet the BER requirement.
- The BER of the uncoded bits should meet the BER requirements on condition of the possible lower-level decoding errors.

- The BER of the turbo-coded bits should not be too much better than the BER of the uncoded bits. When this is the case, it means the system is wasting some bandwidth or power in providing protection on the turbo coded bits. The overall performance is then decided by the uncoded bits.
- The BER of the uncoded bits, with the assumption of correctly decoded turbo-coded bits, should not be too much better than that of turbo coded bits. This happens when there are many turbo-coded bits allocated to one QAM symbol. In the constellation mapping, these turbo-coded bits divide the constellation into partitions. The more partitions there are, the smaller the partition size is, the larger the intra-subset constellation distance is. Since the uncoded bits are used to determine different constellation points in one subset, it is possible that the protection provided by the large intra-subset minimum distance is much higher than the protection to the turbo-coded bits by the turbo coding. When this happens, the overall performance is decided by the turbo coded bits and the over-protection on the uncoded bits will be totally useless since the receiver needs correct decisions on the turbo-coded bits to make decisions on these uncoded bits.

Therefore, in the proposed TTCM structure, the error performance from different levels should be balanced. In this section, we compare the performances of the uncoded bits and the turbo-coded bits in our proposed TTCM structure.

With a rate-1/2 turbo code and the constellation mapping as shown in Figure 3-6, each QAM symbol from a constellation of 2^b will carry $(b-2)$ information bits. For the required BER of P_b , let us assume $d_{1,min}$ is the required minimum distance in the QAM constellations for uncoded systems and $d_{2,min}$ is the required minimum distance for the turbo-coded system.

An information bit sequence $(u_1, u_2, \dots, u_{b-2})$ is turbo-encoded into $(u_1, p_1, u_2, p_2, \dots, u_{b-2})$, where the first two bits $\{u_1, u_2\}$ are encoded into $\{u_1, p_1, u_2, p_2\}$. With the mapping shown in Figure 4-3, the uncoded bits are separated by $4 \cdot d_{2,min}$ after the constellation mapping. To obtain the same P_b for uncoded bits in the TTCM as that in an uncoded system, we must have for uncoded bits:

$$d_2 \geq \frac{d_1}{4}$$

Eq. 4-31

Next, we look at the requirement on $d_{2,min}$ to obtain the required BER from turbo coded bits. For the uncoded QAM transmission, the signal-to-noise ratio per bit (γ_b) is calculated as,

$$\gamma_b^{un} = \frac{\frac{2}{3} d_{1,min}^2 (2^{b-2} - 1)}{b - 2} \quad \text{Eq. 4-32}$$

whereas for TTCM γ_b is calculated as,

$$\gamma_b^{co} = \frac{\frac{2}{3} d_{2,min}^2 (2^b - 1)}{b - 2} \quad \text{Eq. 4-33}$$

When a 6 dB coding gain can be obtained for the required BER, we have

$$\frac{\gamma_b^{un}}{\gamma_b^{co}} = 6 \text{ (dB)} \quad \text{Eq. 4-34}$$

Combining Eq. 4-32, Eq. 4-33 and Eq. 4-34, we obtain

$$d_{2,min} \geq \frac{d_1}{4} \quad \text{Eq. 4-35}$$

Looking at Eq. 4-31 and Eq. 4-35, it is apparent that the coded and uncoded bits have similar performance. Therefore, mapping 4 coded bits to each QAM symbol balances the BER performance of the uncoded bits and coded bits when a coding gain of 6 dB could be obtained from the turbo code on the turbo coded bits. The actual required coding gain for the turbo-coded bits should be slightly more than 6 dB since the estimated BER performance of the uncoded bits is based on an assumption of correct turbo decoding.

Simulation results in [3] showed more than 8 dB coding gain for 16QAM TTCM of 2 bits per symbol. This means that the turbo-coded bits are better protected than the uncoded bits. In this case, the overall performance is decided by the performance of the uncoded bits. A modified channel coding structure shown in Figure 4-17 is proposed to provide some additional protection to the uncoded bit.

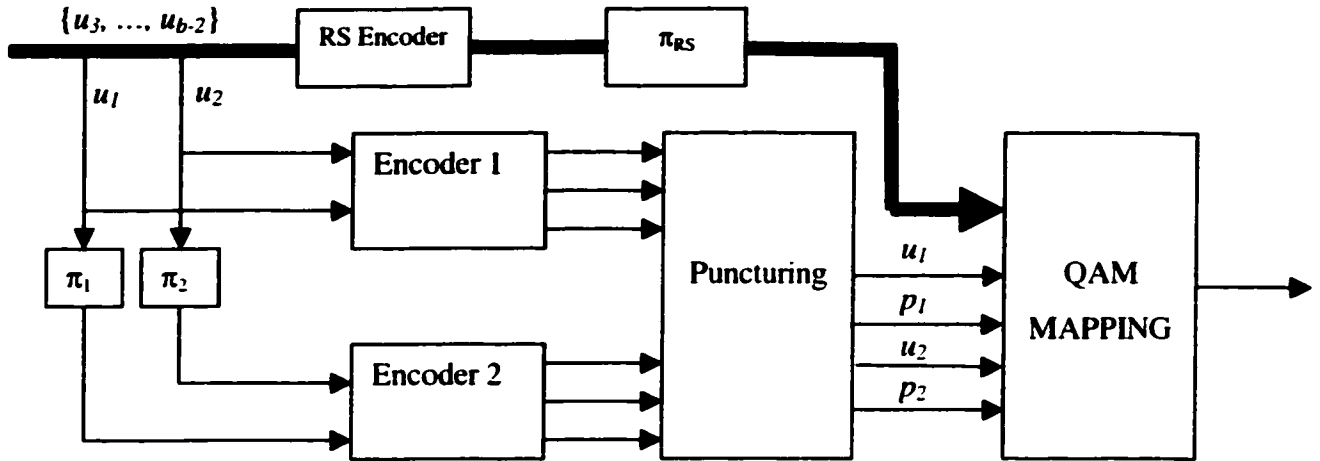


Figure 4-17 – Comprehensive encoding structure for an ADSL DMT system

An optional RS coding (or any other block coding) is performed on the uncoded bits only. This additional RS code can be of very high code rate since it is used to improve the BER performance from an already low value. The interleaver for the RS code, π_{RS} , is not necessary for an AWGN channel. However, it is necessary when impulse noise is a major consideration.

4.2.6. Design of Multilevel Turbo Coded Modulations

In this section, we discuss the design of the CCs employed in TTCMs and the mapping of the turbo coded bits to the constellation points.

4.2.6.1. Multilevel coding employing turbo codes

Three distinct multilevel TTCM structure is actually a special form of multilevel-coding (MLC) and multistage decoding (MSD), where the lowest four levels are coded with a turbo code and the upper levels are left uncoded. To achieve the optimum performance of this two level coded system, the code rate must be designed properly. An additional high rate block code, either employing soft decision or hard decision decoding, can be applied to the higher-level information bits to balance the error performance.

Details of MLC and MSD are discussed in [33]. Performance analysis and design issues of MLC/MSD using turbo codes are given in [34]. It is shown that the channel capacity can be approached by using MLC and the suboptimal MSD, if, and only if (iff)

an optimum code rate distribution across different coding levels is chosen. It has also been pointed out that decision on the parity bit error probability must be taken into account if turbo code is used in the low-levels of MLC/MSD systems [33]. Since the modulation scheme used in ADSL DMT systems is QAM, the following discussions will concentrate on QAM transmission systems. The discussion can be extended to other multidimensional modulation schemes in a straightforward manner.

In an MLC/MSD scheme, the bits in the two-dimensional QAM symbols are grouped into different levels, based on a certain bit-to-symbol mapping scheme. Usually, Ungerboeck's set partitioning is employed. Let's assume an MLC/MSD scheme with l_c levels, where b_i is used to indicate the bits in the i^{th} level, r_i is used to indicate the code rate in the i^{th} level, and code words in all levels have the same length of n_c . The overall transmission rate, R_b in bits per channel symbol, is the sum of the transmission rate of each level, where

$$R_b = \sum_{i=1}^{l_c} r_i \quad \text{Eq. 4-36}$$

It is shown in [33] and [34] that the overall channel can be viewed as l_c equivalent parallel binary channels, which are characterized by their capacities. Applying the well-known chain rule for mutual information, it has been shown in [50] that the sum of the mutual information of these equivalent binary channels yields the mutual information for the complete communication scheme. The following theorem on channel capacity for multilevel coded systems is therefore given as,

MLC/MSD Capacity Theorem

The capacity $C=C(A)$ of a 2^l -ary digital modulation scheme is equal to the sum of the capacity C' of the equivalent channels at the individual coding levels of a multilevel coding scheme which is based on a regular binary partitioning tree of the signal set:

$$C = \sum_{i=0}^{l-1} C' \quad \text{Eq. 4-37}$$

The capacity C can be achieved via MLC and MSD iff the individual rates R' at the different coding levels are chosen equal to the capacities of the equivalent channels, $R' = C'$.

This theorem tells that when using MLC and MSD, the channel capacity could be achieved even though the MSD is a suboptimum decoding process, if and only if the correct coding rate distribution is designed according to the subchannel capacities. The most common criterion of the rate design is to balance the minimum Euclidean distances for all levels. However, recent studies show that the performance obtained based on the balanced Euclidean distance design rule is far from the asymptotic coding gain. This is caused by the enormous increase in the number of nearest neighbor error events at lower levels due to the multiple representation of the binary symbol in level k by all elements of the signal subset, $A_{0...k}$, the subset decided by the bits from level-0 to level- k . Furthermore, turbo codes cannot be clearly characterized by their Hamming distance. It was shown in [34] that significant improvement (about 1.2 dB) could be achieved by using the capacity rate design rule. Applying the channel capacity design rule, it was found that using balanced Euclidean distance design rule often results in lower level design rates that are higher than their capacity. Therefore, no reliable transmission is possible in this level. (Please refer to [33] and [34] for detailed discussions on this issue.)

There are two levels in our proposed TTCM. The lower level consists of turbo-coded bits and the higher level of un-turbo-coded bits. The channel is therefore divided into two parallel equivalent QAM channels, h_0 and h_1 . The optimal code rates shall be designed according to the capacity criterion. The capacity of a modulation scheme with a signal constellation A of equiprobable signal points applied to a memoryless channel with output variables r out of the variable space R is defined by:

$$C(A) = \int_R \frac{1}{|A|} \sum_{a_m \in A} f_r(r|a_m) \cdot \log_2 \frac{f_r(r|a_m)}{\frac{1}{|A|} \sum_{a_k \in A} f_r(r|a_k)} dr$$

Eq. 4-38

Applying Eq. 4-38 to the turbo coded bits, the following equation is obtained,

$$C^0 = \frac{1}{2} \sum_{c_0 = \{0,0,0,0\}}^{\{1,1,1,1\}} \int_R f_r(r|c_0) \log_2 \frac{f_r(r|c_0)}{f_r(r)} dr$$

Eq. 4-39

where c_0 is the 4-bit vector at the turbo-coded level, C^0 is the low-level channel capacity.

It is easy to show that

$$C^0 = C(A) - \frac{1}{16} \sum_{c_0=[0.0.0.0]}^{[1.1.1.1]} C(A_{c_0}) \quad \text{Eq. 4-40}$$

where $C(A)$ is the total channel capacity with constellation A , and $C(A_{c_0})$ is the channel capacity of the constellation consisting of signal points with the last four bits set to a fixed vector C_0 .

With Ungerboeck's set partitioning, Eq. 4-38 could be expressed equivalently as,

$$C^0 = C(A) - C(A_{c_0}) \quad \text{Eq. 4-41}$$

The capacity of the equivalent channel at the lower level is simply the difference of the capacities of the entire signal constellation and that of its subsets addressed by the low-level bits. Calculation of the channel capacity of a QAM transmission has been shown in section 3.1. The capacity calculation of the subset decided by the low-level bits is similar to that for the entire constellation, the difference being the different number of symbols and the minimum constellation distance.

The channel capacities of the overall 64QAM transmission and the lower level equivalent channel are plotted in Figure 4-18. It is shown that for an overall capacity of 4 bits per symbol, the capacity of the lower level is 2 bits per symbol. According to the capacity criterion of rate distribution, this suggests that a rate $\frac{1}{2}$ is the highest rate that the lower level code can have. With a rate higher than $\frac{1}{2}$, reliable transmission is impossible. The higher-level bits do not need FEC or only need a very high rate code. Using high rate codes for low-level bits to achieve capacity is possible if more low levels are encoded together. For example, it is expected that using a rate $\frac{2}{3}$ code for a two-level coding, in which the lowest three bits are the turbo-coded bits including systematic and parity bits, can meet the rate distribution requirement given by the capacity design rule.

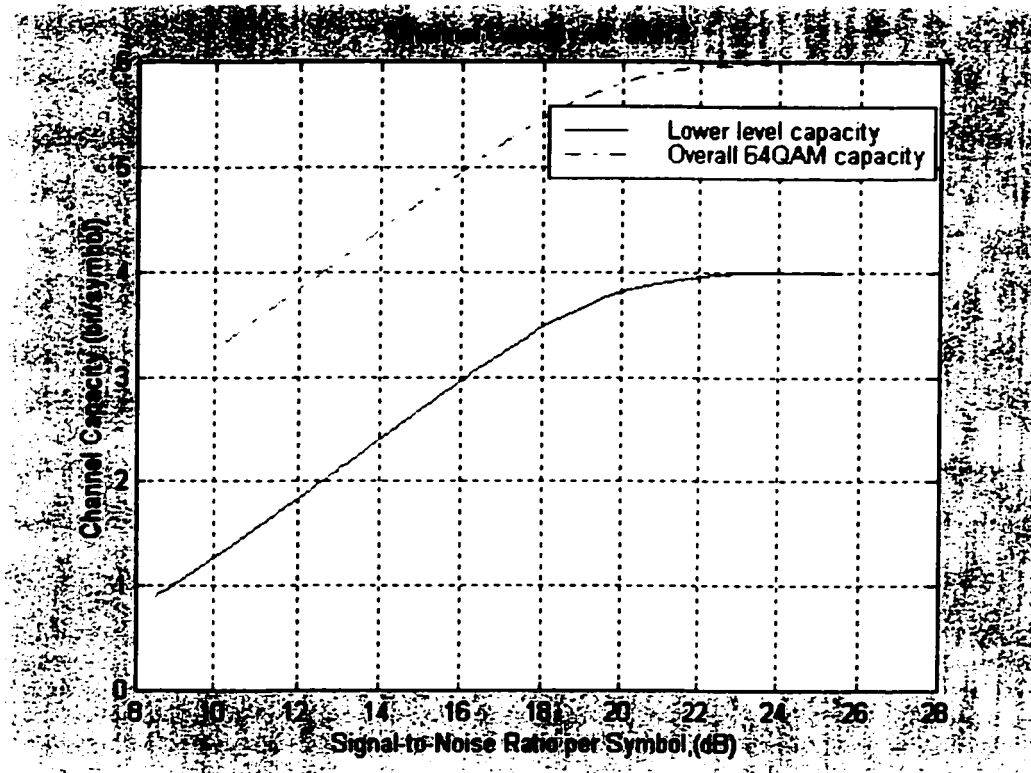


Figure 4-18 – Capacities of 64QAM and the lower level coded bits

The above capacity results are asymptotic results based on infinite block length. For finite-length codes, the following rate design rule is proposed in [34].

Rate design for multilevel coding systems with finite-length code words

For a maximum tolerable block error rate p_e and length n of component codes at all levels, choose the rates R_i of a multilevel coding scheme from the corresponding isoquants of the random coding exponents $E_i(R_i)$ for given total rate $R = \sum_{i=0}^{l-1} R_i$ or given noise variance σ^2 .

For moderate to high block lengths, such as 1000, and for low block error rates, such as p_e less than 10^{-4} , the rate designed according to the above rule gives little difference from the asymptotic rate distribution obtained based on unlimited block length. However,

for shorter codes, remarkable difference exists. It is reasonable to estimate that the TTCM system based on block length-1024 and a BER of 10^{-7} follows the capacity rate distribution rule with infinite block length assumption.

It is also shown in [33] that using maximum likelihood hard decision decoding for the higher levels in an MLC/MSD system will only introduce slight performance loss compared to using the soft decision decoding. This is because the higher levels always use high-rate FECs. Additional performance loss will be introduced by using bounded-distance hard decision decoding instead of the maximum likelihood hard decision decoding. However, for high rate codes, bounded-distance decoding performs close to the hard-decision ML decoding. It follows that using bounded-distance hard decision decoding in the high levels will induce only slight performance degradation. This validates the application of an RS code to the un-turbo-coded bits with hard-decision decoding or erasure decoding in the proposed TTCM systems.

4.2.6.2. TTCM with Gray mapping

It has been shown that with Gray coding in the bits-to-QAM symbol mapping, the binary soft information obtained from the received noise corrupted QAM symbols is no longer Gaussian distributed. The distribution of the best-protected bits is flattened out and it tends to have a larger mean value and variance. Therefore, different optimization criteria of turbo coded ADSL DMT systems should be expected.

The parity bit LLR distribution for 2-bit Gray mapped 64QAM signals is plotted in Figure 4-3. For systematic bits, the distribution is very close to a Gaussian distribution with the same noise variance. Therefore, we conclude that the soft information of parity bits has mean values of ± 2.76 , while the soft information of systematic bits has mean values of ± 1 .

With the help of the *uniform interleaver*, the following asymptotic bound for the bit error probability is obtained as shown in Eq. 4-24. It is apparent that the BER is inverse proportional to the interleaver length. Another important conclusion from Eq. 4-24 is that the most significant parameter through which the CC influences the turbo code performance, namely, z_{min} , is the lowest weight of the parity-check bits in error events of the CC generated by information sequences of length 2. The *effective free distance*, d_{free} , is then calculated as

$$d_{free,eff} = 2 + 2z_{min}$$

Eq. 4-42

Therefore, the CCs should be designed to maximize the effective free distance for turbo code design.

Interleaver designs have been discussed in several previous papers. The design of an interleaver consists of breaking the information patterns, which will produce low weight code words.

In Gray mapped TTCM systems, the criteria will be changed because, as shown previously in this chapter, the soft binary symbols on those bits assigned to different bit positions have different average minimum Euclidean distances. Therefore, the E_b in Eq. 4-24 will be different for different bits. Assigning systematic and parity bits to different positions will produce different performance. Assume now the systematic bits and parity bits have different energy per bit as E_{bs} and E_{bp} . The union bound is then approximated by:

$$P_b(e) \lesssim \sum_{k=1}^{\lfloor N/2 \rfloor} 2k \binom{2k}{k} W^{2k} \frac{Z^{2kz_{min}}}{(1 - Z^{z_{min}-2})^{2k}} \bigg|_{W=e^{-R_s E_{bs} / N_0}, Z=e^{-R_p E_{bp} / N_0}}$$

Eq. 4-43

If only the weight-2 input words are considered as the most significant part on the performance determination, in the design of the CCs and interleavers of the TTCM system with simplified decoding algorithm, the goal will be to maximize the effective free Euclidean distance (EFED), which is now defined as:

$$d_{free,eff}^e = \sum_{i=1}^{w+j} d_{i,min}$$

Eq. 4-44

where w is the number of differing systematic bits, j is the number of differing parity bits and $d_{i,min}$ is the minimum Euclidean distance for the i^{th} differing bit.

Furthermore, assuming that the number of parity bits in the low weight code word is much larger than the number of systematic bits, the bit positions with larger minimum Euclidean distances (MED) should be assigned to the parity bits. This is actually done in two steps. In the first step, the CCs are designed according to the effective free distance. In the second step, a proper bit assignment is performed after during the Gray mapping to improve the turbo code performance. The interleaver is designed according to the same criteria, i.e., to maximize the EFED.

An extension is to assign those most important bits, systematic or parity, to the best-protected bits in the Gray mapping. This will significantly improve the performance because the effective free distance can be much higher. Interleaver should be designed to achieve the same goal. The major problem is therefore to find the CCs, an interleaver and a bit-assignment algorithm to improve the turbo code performance.

Based on breaking the error patterns caused by weight-2 or weight-3 input sequences, punctured turbo codes have been shown to give near Shannon limit performance for a block length of 10000 bits. For long interleavers, in addition to designing the interleavers to break the error patterns, for the remaining unbreakable patterns, the most common position of the differing bits in one turbo block can be obtained through simulations. Performance improvement is therefore expected if those vulnerable positions are better protected by assigning proper Gray mapping positions.

For shorter interleavers, the effective free distance is small. Therefore, by breaking some low-weight error patterns and assigning some vulnerable positions by high-protected bits, the EFED can be increased significantly.

This analysis can be further extended to the use of higher rate turbo codes and 8-Gray mapping. An 8-Gray mapping for one dimension of the 256QAM transmission is shown in Figure 4-19.

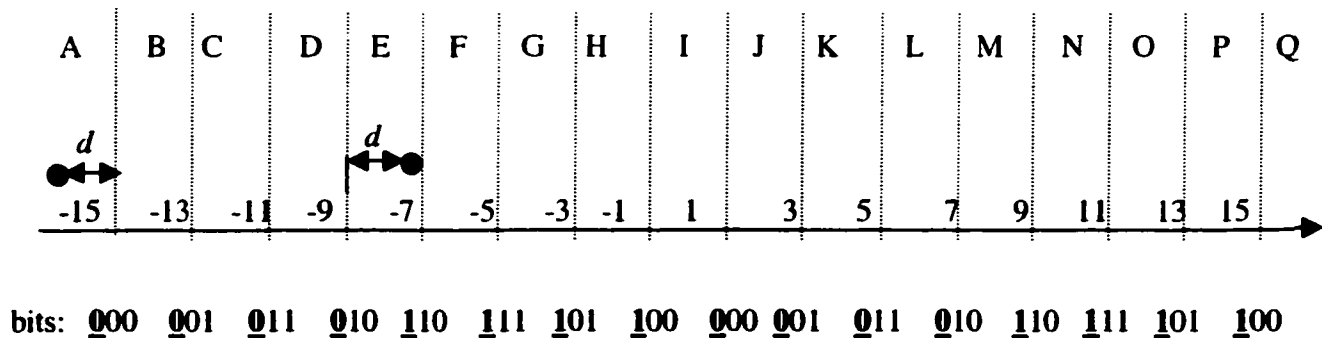


Figure 4-19 – 3-bit Gray mapping of in-phase or quadrature component of a 256QAM

It is also easy to show that the soft information can be well approximated with linear functions of d . The overall PDF of the bold bits in Figure 4-19 is shown in Figure 4-20.

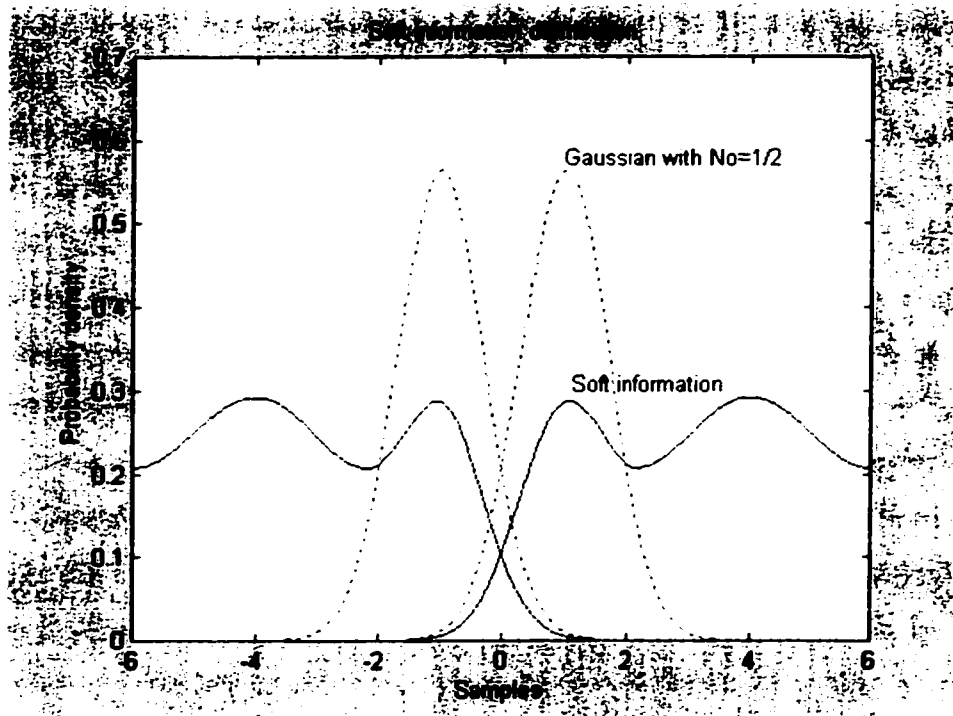


Figure 4-20 – LLR distribution of the first bit in 3-bit Gray mapping in the 256QAM as shown in Figure 4-19

The mean of the channel values for this bit position to be '1' and '0' are ± 4.5 , which is much larger than ± 1 in the standard BPSK with the same $N_0 = 1/2$. Through similar procedure, the second bits position is calculated to have means of ± 1.84 . The third bits position has a distribution very similar to a Gaussian distribution with means of ± 1 .

It is apparent that the soft information symbols have larger average mean values. This effect doesn't vanish with an increase in the size of the QAM constellation because the soft binary symbol distribution in Figure 4-20 considers only the inside bits. Therefore, the effective free Euclidean distance (EFED) of the TTCM with any size of QAM constellation can be improved. The two systematic bits are allocated to the two less protected bits and the parity bits are allocated to the best-protected bits.

However, to keep the number of parity bits in each QAM symbol unchanged, a rate-2/3 turbo code must be employed. While using large size Gray mapping can increase the EFED, the Shannon limit (in SNR per bit) for rate 2/3 FEC codes is about 0.5 dB worse than that of rate 1/2 codes. Therefore, to achieve a better performance by using higher

order Gray mapping, the performance improvement should be able to make up for this capacity loss. It is also shown in section 4.2.6.1 that for a multilevel TTCM with the lowest four-level bits coded, the optimum rate for the low level bits is $\frac{1}{2}$, whereas for structures with the lowest six-level bits coded, the optimum rate is $\frac{2}{3}$. Therefore, they both satisfy the optimum rate distribution obtained according to the capacity design rule for multilevel coding in AWGN channels with MSD suboptimal decoding.

4.3. SIMULATION RESULTS OF TTCM IN QAM TRANSMISSIONS AND DMT SYSTEMS

This section presents simulation results of the proposed TTCM in single-carrier QAM transmissions. It is further shown that these results can be directly applied to estimate the performance of ADSL DMT systems employing the proposed TTCM. The TTCM structure with the turbo encoder shown in Figure 3-6 and the pragmatic TTCM shown in Figure 4-8 are both simulated and the performances are compared. For the TTCM with turbo encoder shown in Figure 3-6, SMAP algorithm is employed for best performance. For the pragmatic TTCM, the simplified decoding algorithm is used. In the later case, there are two simplifications that make it sub-optimum. The first one consists in the using of the binary MAP decoder instead of the SMAP decoder. The second one uses the punctured rate-1/2 CC to implement a rate-2/3 CC. We will evaluate the performance degradation caused by these two simplifications. To avoid adding too much additional delay to ADSL DMT transmissions, we limit the interleaver length to 1024-bit. For an ADSL DMT system with 200 subchannels, using a 1024-bit interleaver results in an additional delay of at least 5 DMT symbols, i.e., 1.25 ms. There are applications where the delay requirement is more flexible. For these applications, longer interleavers can be used and better performance can be expected. Without special indication, the assumptions made in the simulations are listed in Table 4-8.

Previous published simulation results in the literature showed an error floor of binary turbo decoding at a BER of 10^{-5} . It is expected that the proposed TTCM using simplified TTCM decoder will encounter the same problem because a binary turbo decoder is employed. To lower the error floor to the required BER of 10^{-7} in ADSL transmissions, a concatenated channel coding structure would be a good option. The inner code is the TTCM we have discussed in previous sections. An RS code will be added as the outer

code to correct the burst errors output from the TTCM decoder. This concatenated coding structure is also simulated and the assumptions are also shown in Table 4-8.

Table 4-8 – Default simulation assumptions

Channel	AWGN channel, no ISI
Decoding algorithm	SMAP and binary MAP
Turbo code interleaver size	1024-bit and 400-bit
Number of iterations	10
Modulation	64QAM and 256QAM
Bandwidth efficiency	4 bits/symbol for 64QAM 6 bits/symbol for 256QAM
Number of redundant bits per symbol	2 parity bits per QAM symbol
RS codes	GF(256)
RS code interleaver	Byte interleaver

4.3.1. TTCM with Symbol-MAP algorithm in AWGN channels

The simulation was first performed with a 64QAM transmission with TTCM. The bandwidth efficiency is 4 information bits per symbol. The constellation mapping is performed according to Figure 4-3. Two uncoded bits are used to select one of the four quadrants in the 64QAM constellation. The two other bits are turbo encoded into 4 bits, and they are used to select a signal point in each subsection.

The simulation results are shown in Figure 4-21. Each point on the performance curve has been tested with over 10^7 bits. The highest signal-to-noise ratio per bit (γ_b), 8.45dB, which has been tested with over $4 \cdot 10^7$ bits, did not give any error on the turbo coded bits. However, no conclusion should be drawn that the coded bit error rate with this γ_b is below 10^{-7} , due to the lack of sufficient simulation data. It was observed from the simulations that for low SNR values, a decoding error most probably gives about 80 (information) bit errors. Therefore, with a simulation of $4 \cdot 10^7$ bits, once a single error event occurs, the BER is at the order of 10^{-6} .

It is shown in Figure 4-21 that a coding gain of more than 5.3 dB was achieved by using the proposed TTCM in the 64QAM transmission at a BER of 10^{-5} . It is observed that uncoded bits always have a higher BER than coded bits. No BER floor is observed for the turbo coded bits. However, it is observed that the uncoded bits have a “BER floor” at around 10^{-6} . Some explanations of the simulation results are:

- ❖ Although no BER floor is found for turbo coded bits, it is observed that the BER curve of the coded bits becomes flatter after $2 \cdot 10^{-5}$. Therefore, it is likely that the BER floor of the turbo-coded bits was not observed because of an insufficient simulation data.
- ❖ The uncoded bits were not protected as well as the turbo-coded bits. Therefore, in the range of E_b/N_0 larger than 8.2 dB, where no error in turbo-coded bits occurs, the overall error performance follows a Q-function. Compared to the water-fall region of the performance curve of turbo codes, the BER vs. SNR curve plotted according to a Q-function is much flatter. It results in a “BER floor” as shown in Figure 4-21.

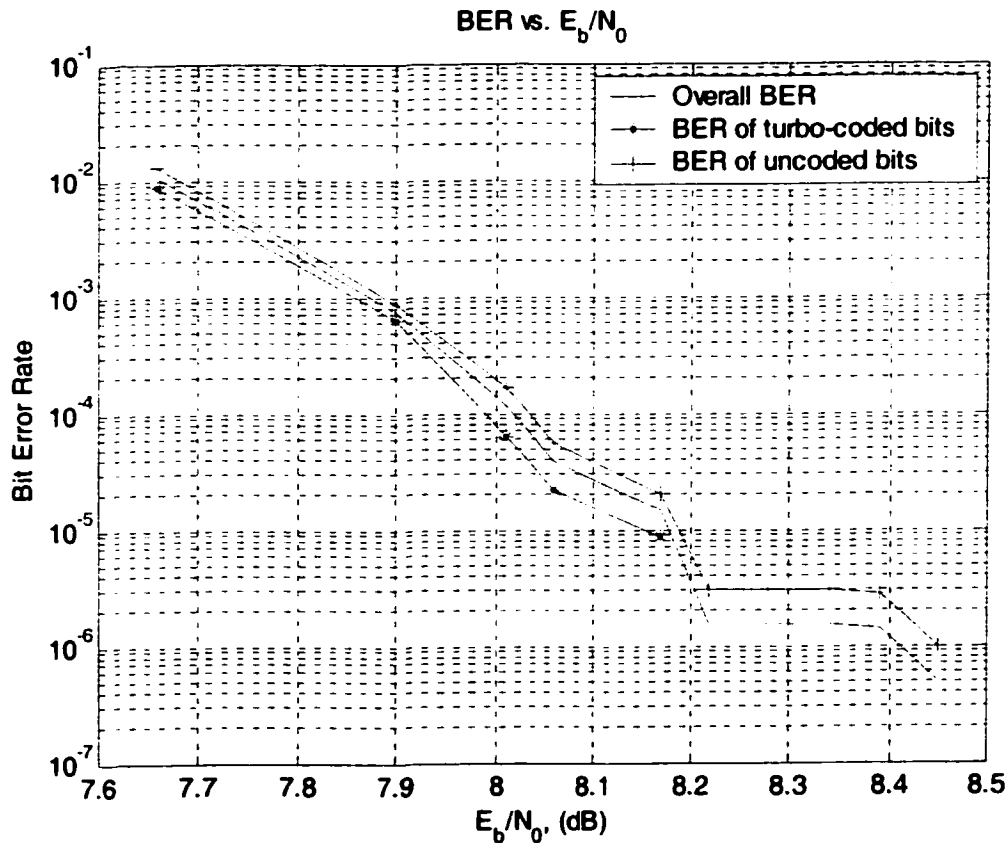


Figure 4-21 – BER of 64QAM TTCM with Symbol-MAP decoding algorithm

64QAM transmission, 4 information bits per symbol, rate-2/3 CCs, two
1024-bit interleavers, Symbol-MAP, 10 iterations

- ❖ The decisions of uncoded bits depend heavily on the lower-level turbo-coded bits. Whenever there are decoding errors, it is very likely that the receiver makes error decisions on the corresponding uncoded bits allocated to the same QAM symbols.

Using the mapping scheme shown in Figure 4-3, assuming that a pair of output systematic and parity bits, $[u_k, p_k]$, from turbo decoder during a decoding error is uniformly distributed over all possible combinations, the error probability of uncoded bits is more than $1/8^{13}$.

The performance of the pragmatic TTCM employing simplified decoding structure is shown in Figure 4-22. A coding gain of 5.2 dB can be achieved at a BER of 10^{-5} and this is less than 0.2 dB worse than that obtained by SMAP with two 1024-bit interleavers.

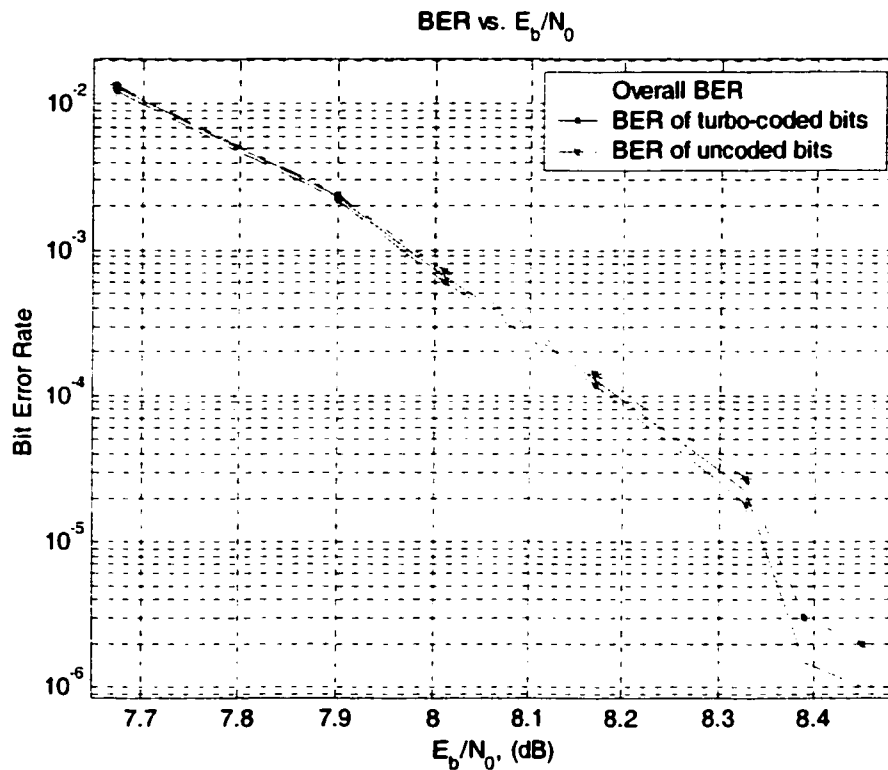


Figure 4-22 – BER of TTCM with simplified TTCM decoding

64QAM transmission, 4 information bits per symbol, binary-MAP, one
1024-bit interleaver, 10 iterations

It was observed from the above results that good coding gains could be obtained at a BER of 10^{-5} . However, the BER requirement in ADSL DMT system is 10^{-7} . We are then facing two situations here:

¹³ The derivation of this number is given at the end of this chapter.

- ❖ When there is no BER floor from the turbo decoder before 10^{-7} , a coding gain of at least 7 dB is expected. In this case, we would have to add additional protection to the uncoded bits.
- ❖ When there is a BER floor before 10^{-7} from the turbo decoder, we should consider using a concatenated coding structure to keep the good coding gain performance by lowering this BER floor with an outer RS code.

Although it has not been proved that the BER floor can be avoided in the ADSL DMT systems employing the TTCM, the following facts support the asymptotic performance:

- ❖ In [3], no BER floor was observed at 10^{-8} from a 16QAM TTCM scheme, with 2 length-16384 bit interleavers. For a 64QAM TTCM with 4 length-4096 interleavers, no BER floor was observed before 10^{-7} .
- ❖ In [32], it was shown that the BER floor of binary turbo coded BPSK transmission in AWGN can be lowered to $2 \cdot 10^{-7}$ with an interleaver size of 1024 bits. It is therefore expected that by using a longer and carefully designed interleaver for the specific CCs, a BER floor lower than 10^{-7} could be achieved.

In [3], a 64QAM system based on two interleavers with a length of 4096 bits achieves 10^{-7} BER at around 7.5 dB. This is about 1.25 dB better than our simulation performance shown in Figure 4-21. This difference can be explained by the fact that we are using a shorter interleaver length of 1024 bits. At a low BER, the minimum distance determines the performance of a turbo code. In [35], it is shown that the minimum distance of a turbo code with a long interleaver is expected to be much larger than that of a turbo code with a short interleaver. The difference between the S-parameters of the interleavers of length 1024 and 4096 also suggests this difference, as shown in Table A- 1.

In the following sections, we present simulation results when applying the concatenated coding structures.

4.3.2. Performance of using concatenated coding structure

An outer RS code can be employed to further improve the coding gain performance, as shown in Figure 4-1. The RS code is applied in combination with an interleaver to break the long error burst output from the inner code decoder (the turbo decoder in our case). The price to pay is the complexity of the RS encoder, the RS decoder and the delay

introduced by the RS encoding, RS decoding and the additional interleaving and deinterleaving processes. In the ADSL standard, an RS code in Galois Field GF(256) is used, which means an RS code word symbol is 8 bits, i.e., 1 byte. The interleaver and deinterleaver operate also on the byte basis for optimal performance.

4.3.2.1. TTCM with Symbol-MAP decoding

In this section, the concatenated coding scheme with an RS code as the outer code, and the TTCM as the inner code is evaluated via simulations. An RS(N_{rs}, K_{rs}) code generates a length N_{rs} -byte code word from K_{rs} information bytes. It could correct up to $(N_{rs}-K_{rs})/2$ error bytes in one RS code word. Therefore, with the same redundancy ratio, the longer is the RS code word, the better the burst error correction capability it has. The RS code in GF(256) can have code words of any length shorter than 256 bytes. In our simulation, we used $K_{rs}=50$ and $K_{rs}=200$ bytes and 2, 4 and 8 redundant bytes to test the performance.

In ADSL DMT systems, the redundancy introduced by the RS encoding is absorbed into the DMT symbols. This results in an increase in the SNR requirements of some subchannels because they are now carrying more bits. The SNR penalty, G_{neg} , caused by the redundant bits can be calculated as,

$$G_{neg} = \frac{2^{b_{av} \cdot (1+P_{re})} - 1}{2^{b_{av}} - 1} \quad \text{Eq. 4-45}$$

where b_{av} is the average number of bits per subchannel and P_{re} is the RS code redundancy ratio. The SNR penalties for various redundancy ratios are shown in Table 4-9, with the assumptions of 100 subchannels and an average of 4 bits per subchannel.

Without considering the SNR penalties, the BER performance of the concatenated RS-TTCM in a DMT system with 64QAM transmission in all subchannels are shown in Figure 4-23. The SNR penalties resulted from using redundancy of 2, 4, 6 and 8 bytes are 0.1283 dB, 0.2564 dB, 0.3843 dB and 0.5119 dB respectively. The actually coding gains are obtained by subtracting the SNR penalties from the coding gains shown in Figure 4-23, which are listed in Figure 4-24. It is shown that a coding gain of 7 dB could be obtained at a BER of 10^{-7} by employing the concatenated RS-TTCM.

Table 4-9 – SNR penalty resulted from different RS code redundancies in DMT systems
Assuming 100 subchannels, on average 4 bits/subchannel

P_{re} (%)	2%	4%	6%	8%	% 10	% 12	16%	20%
SNR penalty (dB)	0.26	0.5	0.7	1	1.2	1.5	2	2.5

It is observed from Figure 4-24 that for a BER of 10^{-7} , the best coding gain is obtained by using 2 redundant bytes or 4 redundant bytes. The SNR penalties of using 6 or 8 redundant bytes overwhelm their coding gain advantages. Therefore, using a more powerful RS code will not necessarily give better performance in DMT systems.

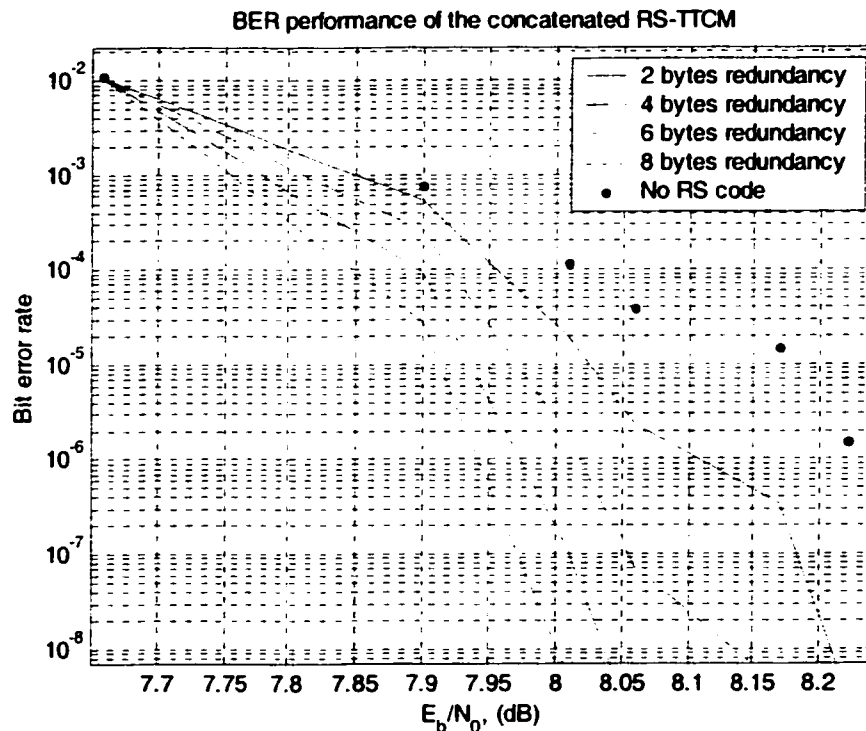


Figure 4-23 – RS-TTCM BER performance without considering SNR penalties

64QAM transmission, 4 information bits per symbol, symbol-MAP, two 1024-bit interleavers, 10 iterations, $RS(N_{rs}, K_{rs})$ in $GF(256)$, $K_{rs}=200$, SNR penalties of RS code redundancy are not considered.

Although an RS code with higher redundancy may not provide a higher coding gain, it can reduce the interleaving delay. The output error sequence of turbo code is always a long burst, which necessitates an RS code that can correct long burst errors. These can be

either RS codes with higher redundancy ratio or longer RS codes with the same redundancy percentage.

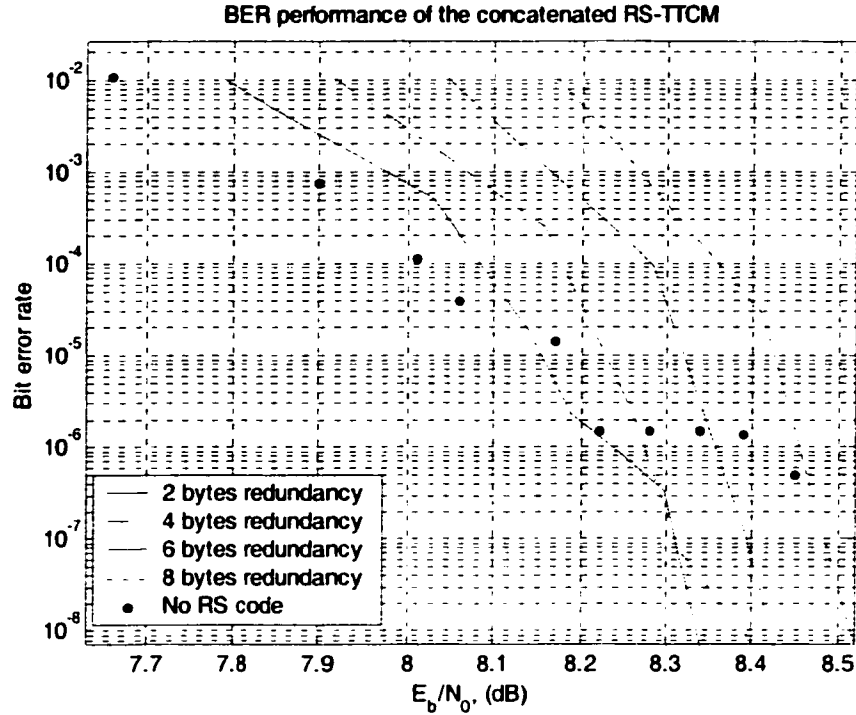


Figure 4-24 – RS-TTCM BER performance considering SNR penalties

64QAM transmission, 4 information bits per symbol, symbol-MAP, two 1024-bit interleavers, 10 iterations, $RS(N_{rs}, K_{rs})$ in $GF(256)$, $K_{rs}=200$, SNR penalties of RS code redundancy are included.

4.3.2.2. Pragmatic TTCM with simplified decoding structure

The performance of the concatenated coding structure with TTCM using simplified decoding structure is shown in Figure 4-25 and Figure 4-26 for an interleaver size of 1024 bits. A coding gain of 6.8 dB is obtained for a BER of 10^{-7} .

Similar performance of the concatenated coding scheme is observed as those in Figure 4-24. With more than 2 redundant bytes in each RS code word, the SNR penalty overwhelms the coding gain advantage. Therefore, 2 redundant bytes is the best choice to remove the BER floor of the binary turbo decoding while keeping the optimal coding gain performance.

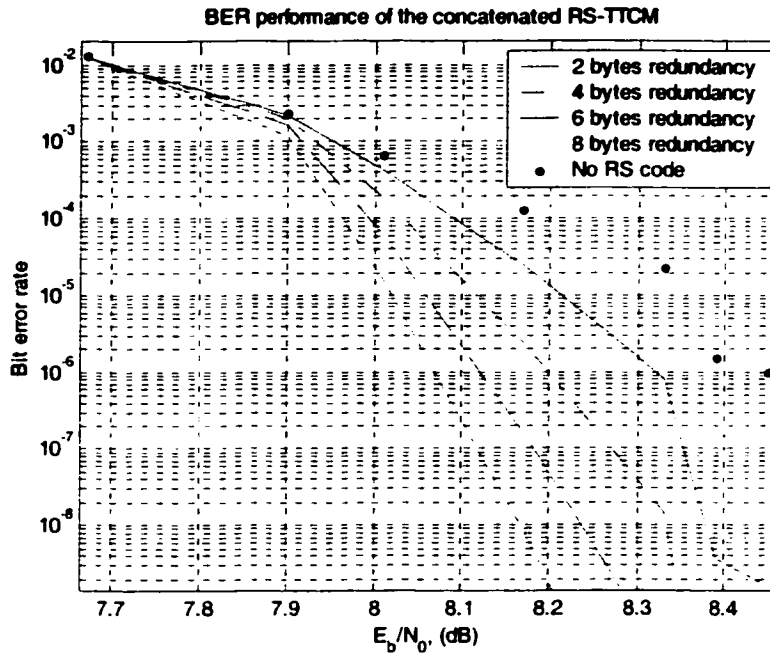


Figure 4-25 – RS-TTCM BER performance without considering SNR penalties

64QAM transmission, 4 information bits per symbol, symbol-MAP, two 1024-bit interleavers, 10 iterations, $RS(N_{rs}, K_{rs})$ in $GF(256)$, $K_{rs}=200$, SNR penalties of RS code redundancy are not considered.

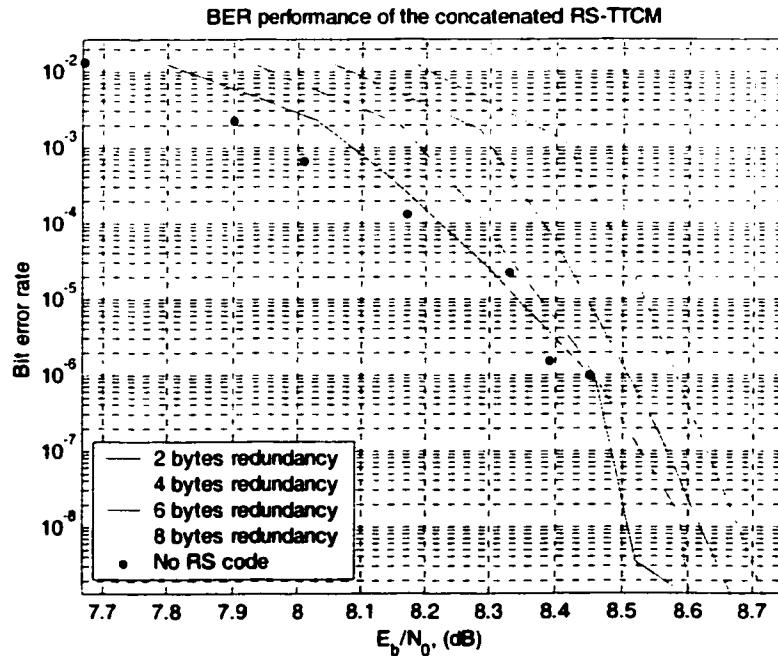


Figure 4-26 – RS-TTCM BER performance considering SNR penalties

64QAM transmission, 4 information bits per symbol, symbol-MAP, two 1024-bit interleavers, 10 iterations, $RS(N_{rs}, K_{rs})$ in $GF(256)$, $K_{rs}=200$, SNR penalties of RS code redundancy are included.

4.3.2.3. Employing shorter (400-bit) interleaver

Simulation results show that the overall error sequence of turbo codes at low SNR usually cause over 80 bit errors spreading over the 1024-bit turbo block, resulting in an error block of 128 bytes. To efficiently correct these error bytes, a large interleave depth is required, which results in a very large additional system latency.

The latency could be reduced by using short turbo blocks. For example, a single decoding error in a turbo code with a 400-bit interleaver introduces an error burst no longer than 50 bytes. The interleaving requirement is therefore reduced by more than twice.

Simulations of the TTCM with 400-bit interleavers were performed and the results are shown in Figure 4-27. Since we are more interested in the pragmatic TTCM with the simplified decoding structure, we only present the performance for this structure. It is observed that the coding gain performances are more than 0.4 dB worse than those obtained using the 1024-bit interleaver.

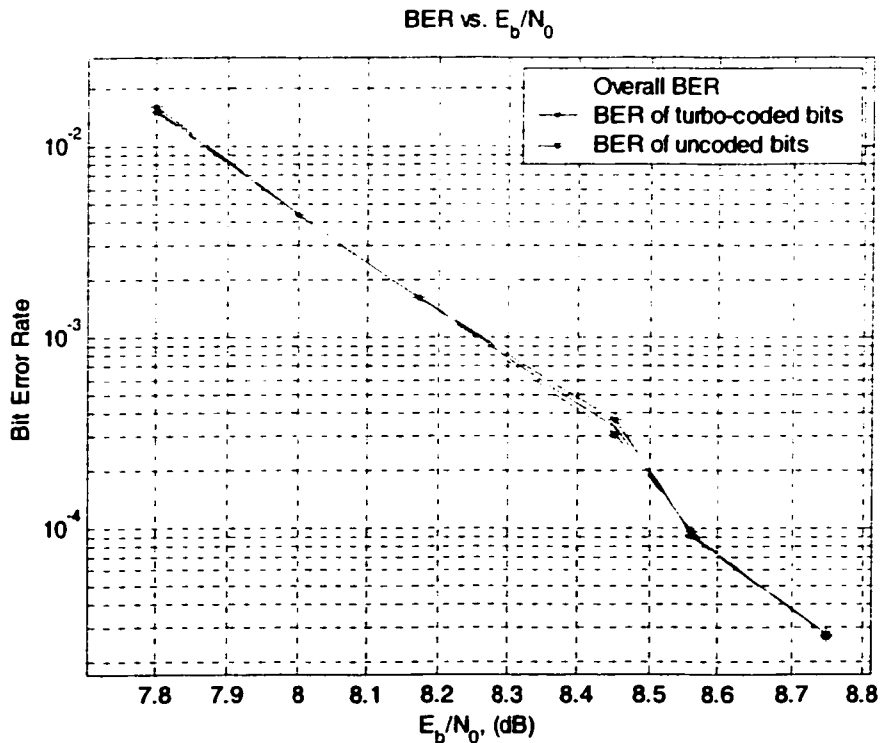


Figure 4-27 – BER of TTCM with simplified decoding, 400-bit interleavers

64QAM transmission, 4 information bits per symbol, punctured rate-1/2 CCs, one 400-bit interleaver, binary MAP decoding, 10 iterations, $RS(N_{rs}, K_{rs})$ in $GF(256)$, $K_{rs}=200$.

The BER performances of the concatenated RS-TTCM, where the TTCM uses 400-bit interleaver, are shown in Figure 4-28. Only 6.4 dB coding gain could be obtained, compared to the 6.8 dB coding gain obtained with the 1024-bit interleaver.

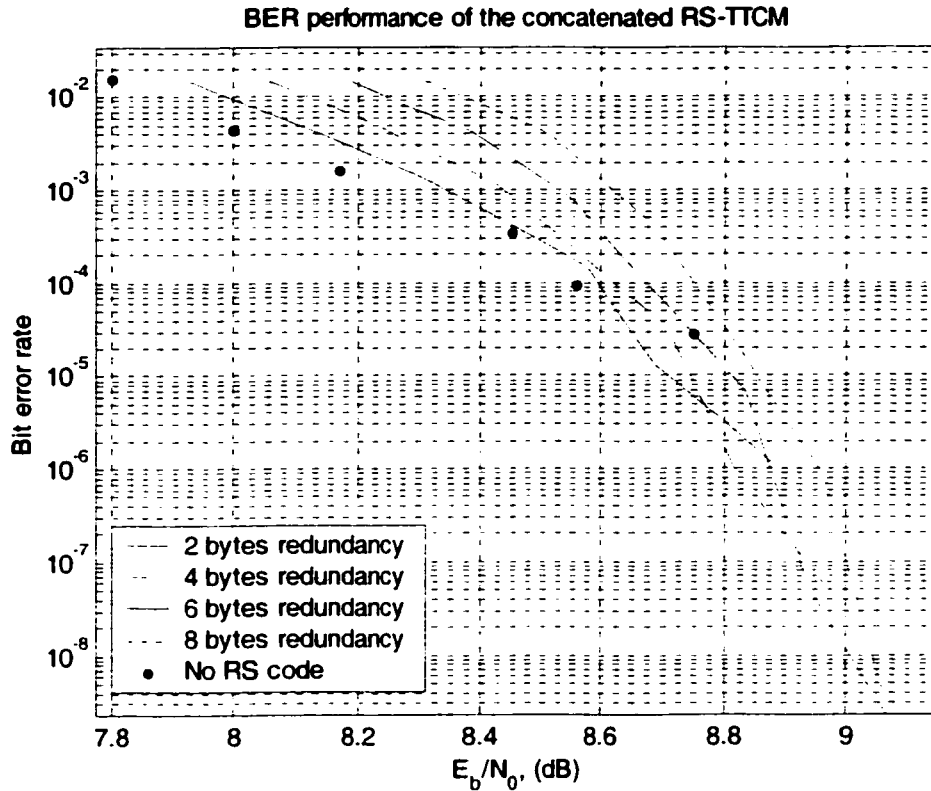


Figure 4-28 – BER of TTCM with simplified decoding, 400-bit interleavers

64QAM transmission, 4 information bits per symbol, punctured rate-1/2 CCs, one 400-bit interleaver, binary MAP decoding, 10 iterations, SNR penalties of RS code redundancy are included.

It is also observed that when using a short interleaver, the coding gain obtained by using the SMAP algorithm at a BER of 10^{-5} is more than 0.4 dB better than that obtained from using the simplified TTCM decoding.

4.3.2.4. Performance comparison of different channel coding schemes

The performances obtained with different coding schemes in an ADSL DMT system are shown in Figure 4-29. Each transmission has a bandwidth efficiency of close to 4 information bits per channel symbol. An RS (208, 200) code is used for RS-coded 16QAM transmission and RS-4DTCM transmission since it was shown in the literature

that ADSL DMT system achieves best results with a redundancy ratio of 4% to 6%, with or without the 4D-TCM. In RS-TTCM, the simulation results presented in previous sections show that 2% gives the best coding gain. Therefore, RS(204, 200) is used for performance comparison.

It was observed that using TTCM or RS-TTCM provides about 1.5 dB additional coding gain than that from using RS-4DTCM.

The channel capacity of 4 bits/symbol is 5.74 dB. It is observed that even with our RS-TTCM, the performance is still almost 3 dB from the channel capacity.

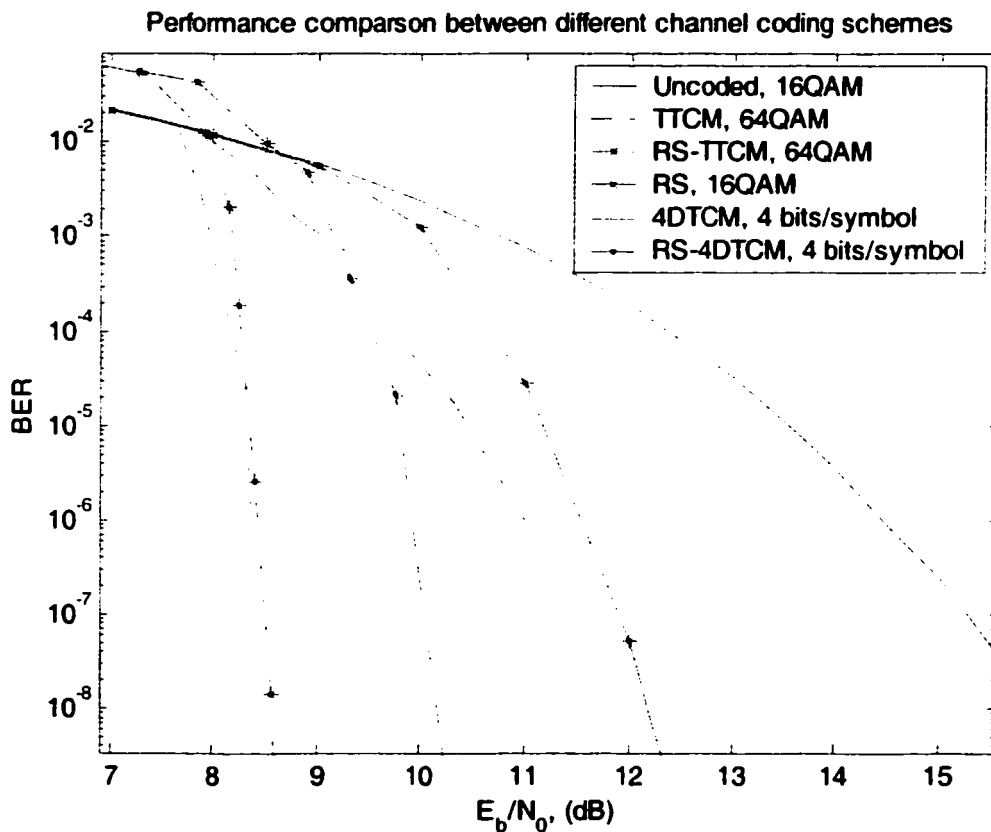


Figure 4-29 – BER comparison between channel coding schemes in ADSL DMT systems

Bandwidth efficiency of 4 information bits per symbol, 16QAM uncoded transmission, RS coded 16QAM transmission, 64QAM TTCM transmission, the TTCM employs simplified decoding, one 1024-bit interleaver, 10 iterations, RS(208, 200) code is used for RS 16QAM and RS-4DTCM and RS(204,200) code is used for RS-TTCM

4.3.3. RS code requirements in concatenated coding scheme

The total delay induced by the concatenated coding of RS-TTCM in ADSL DMT system is the sum of the delay introduced by the turbo code interleaver and deinterleaver and the delay resulted from the byte interleaver and deinterleaver to improve RS code's burst error correction capability.

The delay introduced by the turbo code interleaver depends on how many DMT symbols one turbo code word occupies, the less DMT symbols the shorter the delay is. However, less DMT symbols in one turbo code word means that high-rate turbo code needs to be used to keep two parity bits in each QAM symbol. Let us assume a DMT system with N_c subchannels and each subchannel can support at least 256QAM. It is assumed that a TTCM with an interleaver of size 1024 is used in this ADSL DMT system. When a rate-1/2 turbo code with 2-bit Gray mapping as shown in Figure 4-5 is used, one turbo code word will cover 512 QAM symbols, which is $512/N_c$ DMT symbols. However, when a rate-2/3 turbo code with 3-bit Gray mapping as shown in Figure 4-11 is used, one turbo code word will cover 384 QAM symbols, which is $384/N_c$ DMT symbols. Therefore, employing rate-2/3 turbo code induces only 2/3 of the delay resulted from using a rate-1/2 turbo code. However, even in this case, the delay improvement may not be as much as it appears because in a practical ADSL system, a lot of subchannels cannot support more than 6 bits/symbol transmissions.

Another significant delay is introduced by the interleaver and deinterleaver pairs usually employed in combination with the outer RS codes. In our simulation of 64QAM TTCM systems with 2 uncoded bits per symbol, each decoded turbo block contains 2048 (turbo coded bits and uncoded bit) bits output to the RS encoder, which gives us 256 bytes. It is observed from the simulations that there are, on the average, 180 byte errors in these 256 bytes whenever an error turbo decoding occurs. These 180 byte errors spread all over the 256-byte block. Therefore, the RS decoder is facing an error block of 256 bytes. The required interleaving depths are calculated and listed in Table 4-10. Corresponding results for 256QAM transmission with rate-2/3 turbo codes are also calculated and shown in Table 4-10. It is observed that using a rate-2/3 turbo code requires smaller interleave depth since each turbo code word has a shorter length. The cost is the coding gain performance degradation.

Table 4-10 – Interleave depth requirements and system delays in RS-TTCM

**64QAM TTCM transmission, 4 information bits/symbol, simplified
TTCM decoding, 10 iterations, RS code in GF(256)**

<i>r</i> (bytes)	2	4	6	8
64QAM, 1/2 TC, 1024-bit interleaver	256	128	86	64
256QAM, 2/3 TC, 1024-bit interleaver	192	96	64	48
64QAM, 400-bit interleaver	100	50	34	25

The interleave depths shown in Table 4-10 are fairly large for delay stringent applications, such as multimedia application over ADSL transmissions. It is therefore desirable to decrease the delay without sacrificing the coding gain performance.

The most straightforward way is to use a turbo code with a shorter interleaver, which both decreases the delay introduced by the turbo code interleaving and the RS code interleaving. With a 400-bit interleaver, the output from the TTCM decoder is an 800-bit block. It was shown in the simulations that there are about 90 bytes errors in this block at a γ_b of 8.17 dB. At higher γ_b , this number could be smaller, such as 68 bytes for 8.33 dB. However, since the error bytes are still spread all over the 100-byte block, the RS code still has to deal with an error burst of 100 bytes. The required interleaving depths are shown in Table 4-10.

It has been shown that the BER performance of turbo codes is inversely proportional to the interleaver length. Therefore, smaller coding gain is the price to pay for the shorter delay. Comparing the performances of RS-TTCM shown in Figure 4-25 and Figure 4-26, the coding gains of using a 400-bit interleaver is worse than those obtained by using a 1024-bit interleaver. The shorter delay is achieved with a coding gain loss of 0.4 dB.

Another method to reduce the system delay is to use the modified channel coding structure introduced in the following sections.

4.3.3.1. Serial-Parallel-Concatenated RS-TTCM

It has been shown in the simulations that at the output of the turbo decoder, most of the error decisions on the uncoded bits are caused by the corresponding error decisions on the turbo-coded bits. The number of errors in uncoded information bits is much higher than that of the coded information bits. This is because that all coded bits, including systematic bits and parity bits, cause error propagation to the higher-level uncoded bits,

where usually the number of parity bits in error is much larger than the number of systematic bits in error.

Furthermore, in our proposed TTCM, the uncoded bits are employed to select signal points in the constellation subsections, which have a much larger minimum Euclidean distance. With the assumption of correct turbo decoding, the error probability of uncoded bits should be very small and randomly distributed. Therefore, we do not need to provide similar protection to these uncoded bits as to the turbo-coded bits. Only a simple and high rate channel coding is needed to provide the protection required by the uncoded bits in addition to that provided by the set partitioning.

Keeping these considerations in mind, a serial-parallel-concatenated- (SPC) RS-TTCM structure as shown in Figure 4-30 is proposed to solve the problem of long delay and high SNR penalty of the RS-TTCM concatenated coding structure.

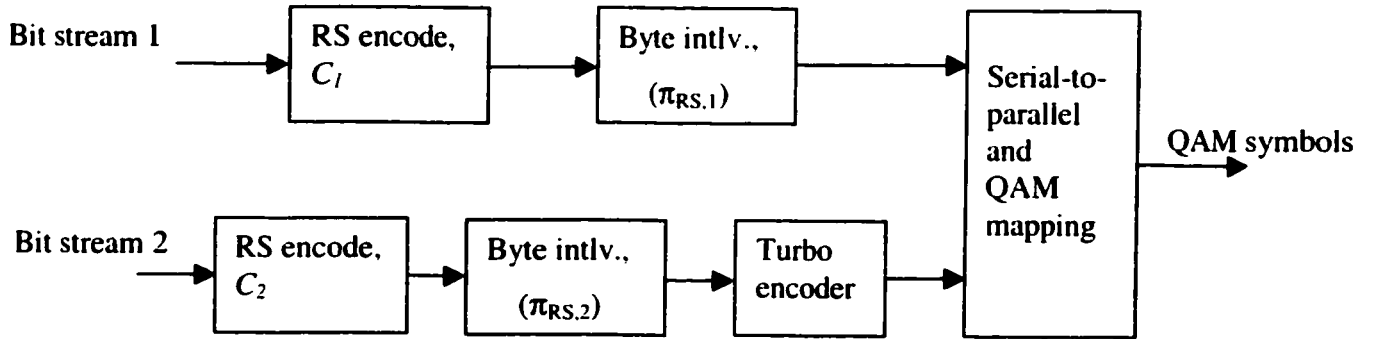


Figure 4-30 – Serial-Parallel-Concatenated-RS-TTCM coding scheme

In this SPC-RS-TTCM scheme, the first RS code, C_1 , is used to provide further protection to the un-turbo-coded bits. The second RS code, C_2 , is used to lower the BER floor of the turbo code. The redundancies of both RS codes can be adjusted to give satisfactory performance.

With this structure, the required interleave depth of the RS code to lower the BER floor of the turbo code is halved for the 64QAM TTCM with 4 information bits per symbol. For QAM transmissions with larger constellations, the delay reduction is greater because more than half of the bits are not turbo coded. Simulation results of the turbo-coded bits are shown in Figure 4-31.

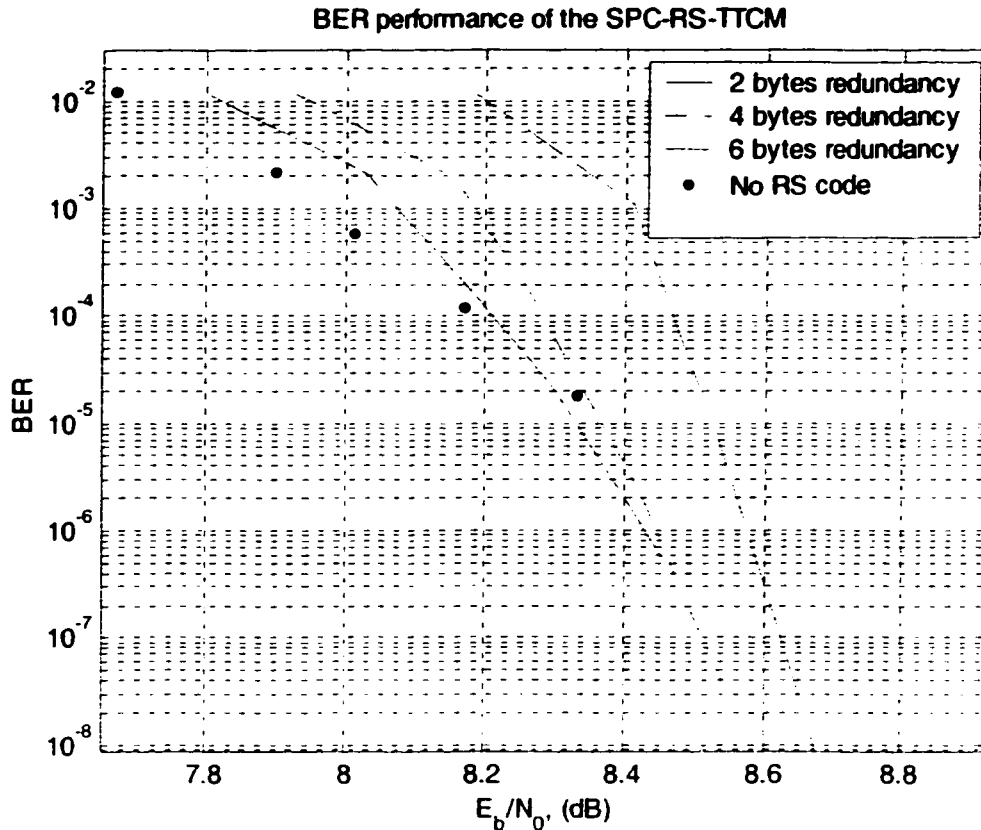


Figure 4-31 – BER performance of lower branch of SPC-RS-TTCM

64QAM, 4 bits/symbol, simplified decoding, one 1024-bit interleaver,
10 iterations, $RS(N_{rs}, K_{rs})$ in GF(256), $K=200$

The BER of the uncoded bits is shown to be in the range of 10^{-5} to 10^{-6} at γ_b of 8 dB based on correct decoding of the turbo coded bits. A very high rate RS code, C_I , such as RS(202,200) can be used here to lower the BER to a value much lower than 10^{-7} . Because the uncoded bits in error will distribute randomly in AWGN channels, no interleaver is required for C_I .

In the presence of impulse noise, while the concatenation of RS code C_2 and the TTCM can provide satisfactory performance, the uncoded bits have 6 dB additional protection against impulse noise. It is therefore expected that impulse noise will introduce short error bursts for the uncoded bits. In addition, with a delay of $D_{e,I}$ introduced by the concatenation of C_2 and TTCM, an interleaver with a delay smaller than $D_{e,I}$ can also be concatenated with C_I to improve its burst error correction capability.

4.3.4. Employing erasure RS decoding technique

Erasure RS decoding can be employed to significantly decrease the system latency. It was mentioned earlier that the error correction capability of an $RS(N_{rs}, K_{rs})$ code is $(N_{rs}-K_{rs})/2$ symbols¹⁴ out of the N_{rs} symbol contained in one RS code word. When the receiver has the exact knowledge of the unreliable symbol locations, it will declare these symbols as erasures that contain no information. Erasure RS decoding is then carried out to recover the information contained in these erasures. The erasure correction capability of an $RS(N_{rs}, K_{rs})$ code is $(N_{rs}-K_{rs})$ symbols out of the N_{rs} symbol contained in one RS code word. Therefore, with the knowledge of the unreliable symbol locations, erasure RS decoding can correct an error burst twice as long as that for a normal RS decoding. Since in the RS-TTCM (or SPC-RS-TTCM), the error sequence output from the turbo decoder is usually the whole turbo block, using erasure RS decoding would require half of the interleave depth compared to using conventional RS decoding. However, the difficulty of using erasure-decoding techniques lies in the difficulty in finding the erasure positions.

It is observed from the simulations that, in most of the times, error turbo decoding results in low absolute LLR values, compared to the LLRs generated by a successful decoding. Therefore, when the average absolute LLR value output from the turbo decoder is under a certain predefined threshold, the current turbo block could be erased. The efficiency of this erasure-declaration method depends on the convergence property of the iterative decoding. Usually, most error decodings are caused by non-convergence of the iterative decoding process, instead of an error convergence¹⁵. If the probability of the error convergence is small enough, this LLR-value-based erasure-declaration can be quite accurate.

4.4. PERFORMANCE EVALUATION IN PRACTICAL ADSL DMT SYSTEMS

In practice, a subscriber line is modeled as an ISI channel with crosstalk noises, where both the channel transfer function and the noise environment are very different

¹⁴ For an RS code in GF(256), each RS symbol contains 8 bits, i.e., 1 byte.

¹⁵ An error convergence is defined as the case where the turbo decoder converges to another valid code word sequence.

from an ideal AWGN channel. In this section, performance evaluation of applying TTCM to ADSL DMT systems is performed and the simulation results are presented.

4.4.1. Turbo coded ADSL DMT system with crosstalk noises

The simulation results shown in the previous sections are based on 64QAM for 4 bits per symbol. In a DMT system, we will have various QAM constellations in subchannels carrying 2 bits/symbol to 15 bits/symbol. The question here is whether the above simulation results can be extended to the DMT systems with different QAM constellations.

In the suggested turbo coded DMT system, the last 4 bits are always turbo-coded bits and the rest are left uncoded. Only turbo-coded bits are transmitted in the subchannels that carry no more than 4 bits/symbol. In the following, only the subchannels carrying at least 4 bits are considered. Subchannels carrying 4QAM or 8QAM should have at least similar performance.

The main difference between any two subchannels is the number of uncoded bits in one QAM symbol. In practice, an ADSL DMT system is always assumed to have a constant QAM symbol error rate (SER) from all the subchannels. In QAM transmission, the SER can be approximated by

$$P_s = 4Q\left(\sqrt{\frac{3E_s}{(2^k - 1) \cdot N_0}}\right) = 4Q\left(\frac{d_{min}}{2\sigma_0}\right) \quad \text{Eq. 4-46}$$

where E_s is the average energy of each QAM symbol and can be expressed as

$$E_s = \frac{1}{6} \cdot (2^k - 1) \cdot d_{min}^2 \quad \text{Eq. 4-47}$$

where d_{min} is the minimum distance between adjacent QAM constellation points and σ_0^2 is the AWGN noise variance and $Q(.)$ is the Marcum Q-Function.

If an AWGN noise environment is assumed, where σ^2 is a constant across all the subchannels, and an ideal bit allocation with infinite bit resolution is performed, QAM symbols with the same minimum distance would arrive at the receiver with the same SER in all the subchannels. If similar constellation mapping is applied, the error performance of turbo-coded bits in different size constellations with the same minimum distance is expected to be approximately the same as the 16QAM transmission without any uncoded

bits. The error performance of uncoded bits, among which the minimum distance is $4 \cdot d_{min}$, will also be the same for all the subchannels as long as the d_{min} is a constant.

It is known that the major stationary impairment in ADSL DMT systems is crosstalk noise (XT), including FEXT and NEXT. The PSD of the XT is usually not flat in the frequency domain. With sufficiently narrow subchannel bandwidth, the XT can be looked as AWGN noises with different variances in different subchannels. In this case, the bit allocation algorithm will give a bit distribution across the subchannels according to the SNRs in the subchannels at the receiver. Because of the subchannel dependent variances, the d_{min} is no longer a constant, unlike in an AWGN channel. However, all of the channels can still be modeled as AWGN channels. For an AWGN channel with a noise variance of σ_i^2 , the information loss in this channel with a transmission power, E_i , should be identical to the information loss caused by another AWGN channel with a noise variance of σ_0^2 and a transmission power of E_0 , when

$$E_0 = E_i \cdot \frac{\sigma_0^2}{\sigma_i^2} \quad \text{Eq. 4-48}$$

Since the bit allocation results in a constant E/σ_i^2 , all subchannels can be normalized to an AWGN channel with the same minimum constellation distance and the same AWGN noise variance. Therefore, it is expected that the TTCM structure can be applied to this ADSL DMT system with XT without any modifications on the decoding procedure. The performance should be similar in all the subchannels.

The correlation between the adjacent turbo-coded bits allocated to the same QAM symbol in the same dimension might induce some performance degradation since MAP decoding is optimal for binary transmission in an AWGN channel. To find out how much this correlation degrades the turbo decoding performance, a block switching method is employed. In the turbo block switching, the parity bits of two adjacent turbo blocks are switched before the constellation mapping. The constellation mapping is therefore using the systematic bits from the first turbo block and the parity bits from the next turbo block. At the decoder, the turbo decoder performs the decoding based on the systematic soft information obtained from current block of QAM symbols and parity soft information obtained from next block of QAM symbols. The correlation between the soft information of the systematic bits and parity bits is totally removed. The performance obtained using

block switching is simulated and shown in Figure 4-32. Less than 0.1 dB improvement is observed. Therefore, it is concluded that the correlation does not leads to severe performance degradation.

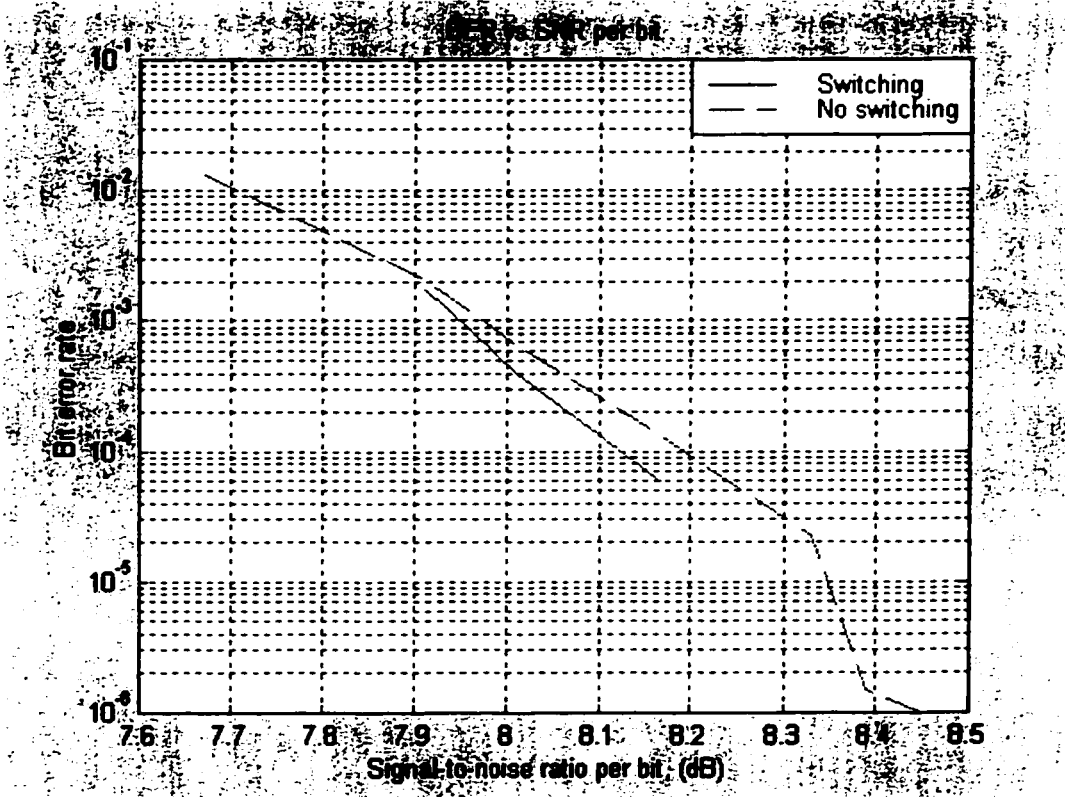


Figure 4-32 – Effect of using block switching to improve the system performance

4.4.2. An ideal case of TTCM

The performances obtained so far are all based on our own simulation results with a reasonable interleaver length. It was observed that using a concatenated RS-TTCM is necessary to lower the BER floor, at around 10^{-5} , caused by the turbo decoding. Adding the external RS code not only increases the receiver implementation complexity and the system delay, but also introduces degradation to the coding gain performance. Therefore, an ideal case is to avoid using the serial concatenated coding structure.

Our proposed TTCM structure is based on using the structure proposed by Benedetto in [3]. It was shown that no BER floor was observed at 10^{-8} for a 16QAM TTCM scheme

with two length-16384-bit interleavers. The coding gain of this TTCM at 10^{-7} is about 8 dB. This TTCM can be used in the parallel-concatenated structure as shown in Figure 4-17. Since this TTCM provides a very high coding gain, an additional RS code is necessary to provide more protection to the un-turbo-coded bits. As an ideal case, we assume that incorporating this TTCM into the structure shown in Figure 4-17 still provides the same coding gain performance to the turbo-coded bits, which is repeated in Figure 4-33. Let us now find out what kind of RS code is needed for the further protection to the uncoded bits and what is the achievable coding gain.

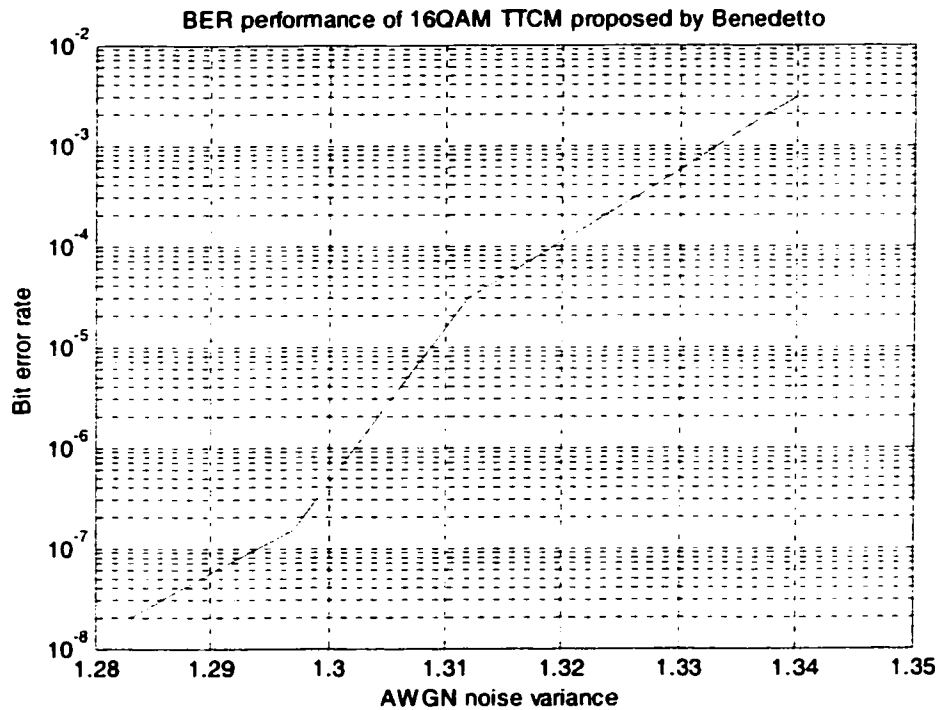


Figure 4-33 – Performance of Benedetto’s 16QAM TTCM with a rate $\frac{1}{2}$ turbo code
16QAM, 2 bits/symbol, SMAP decoding, two 16384-bit interleaver,
18 iterations

For a 64QAM with the same turbo encoder as proposed by Benedetto, with a noise variance of 1.283 (resulting in a BER of $2 \cdot 10^{-8}$), the uncoded bits have a BER of around $2 \cdot 10^{-4}$. Since the desired BER requirement is 10^{-7} , an additional RS code is required for the un-turbo-coded bits. It is shown that an $RS(N_{rs}, K_{rs})$ code with $K_{rs}=200$ requires at least 8 redundant bytes to provide an output BER of 10^{-7} with an input BER of $2 \cdot 10^{-4}$.

The delay required by this turbo code is mainly caused by the turbo code interleavers, which are of length 16384-bit. The length of a turbo block is 65536. The end-to-end total delay, D_b , in number of transmission bits is $65536 \cdot 2 = 131072$ bits.

In ADSL DMT systems, the maximum interleave depth for the RS code is defined as 64 and one RS codeword can contain as many as 2048 bits. Therefore, the maximum delay, in number of bit, is calculated as $2048 \cdot 64 = 131072$.

Therefore, it is observed that the ideal case TTCM just satisfies the delay requirement of the ADSL DMT systems.

4.5. FURTHER PERFORMANCE IMPROVEMENT

In a multilevel coding scheme, the decoding of the higher level bits depends on the low-level code words. In our proposed TTCM, the receiver needs both the systematic bit decisions and the parity bit decisions to determine the uncoded bits. Therefore, whenever a decoding error occurs, the uncoded information bits associated with the errors in both systematic and parity bits have a high probability of being in error. In designing a turbo code, the low weight error sequences usually have few (two, usually) systematic bits and a large number of parity bits [32]. In the high SNR range, these sequences are the major contributors to the BER performance. In the moderate or low SNR range, it was observed from our simulation that a decoding error usually causes much more erroneous parity bits than erroneous systematic bits. The number of QAM symbol errors associated with the erroneous parity bits is therefore much larger than those associated with erroneous systematic bits. As a result, the main contribution to the BER of uncoded information bits comes from the errors associated with erroneous decisions in parity bits.

Unfortunately, the MAP decoding algorithm is solely designed to minimize the bit error probability of the information bits, i.e., systematic bits. In our simulation, the code word required for recovering the uncoded bits are obtained through a re-encoding process. This results in a suboptimal performance in determining the uncoded bits. An optimal solution should employ a turbo decoder with SISOs that minimize the error probability of both systematic bits and parity bits. Such a SISO algorithm was introduced in [36] that tries to minimize the parity bits error rate by direct parity bit estimation during the MAP decoding process. It was shown in [34] that using this direct parity bit estimation lowers the BER of all coded bits by 10 times. It is therefore expected that

applying this SISO decoding algorithm in our TTCM decoder should result in better performance for the uncoded bits.

Since our proposed TTCM uses a conventional binary turbo decoder, the methods to improve the binary turbo decoding performance in the literature can be applied to further improve its performance.

In [37], a detailed design procedure is introduced for high rate punctured binary turbo codes. It is easier to achieve high-rate turbo codes by puncturing because it doesn't introduce much additional complexity in the encoding and decoding process. The design is based on searching for the best CCs, designing good interleavers that break most of the low-weight code words generated by weight-2 or weight-3 input words, followed by the searching for the best puncturing mechanism. The criteria for the design are the maximization of the minimum word weight of the turbo code, and the minimization of the multiplicities of low-weight code words. Simulation results based on a block size of 10000 bits were shown to be within 0.9 dB from Shannon limit, for rates from 2/3 to 16/17. It might be advantageous to combine their design of high-rate binary turbo codes with our proposed TTCM, using either simplified decoding or SMAP decoding.

In our simulation, it is observed that using the punctured rate-1/2 turbo codes achieve much poorer performance (about 2 dB) than using the original rate-1/3 turbo code. Therefore, rate-1/3 turbo codes can be employed instead of the rate-1/2 turbo code as shown in previous sections. However, as the redundancy increases, the throughput, i.e. the number of information bits per channel symbol, decreases. Using larger QAM constellations will make the transmission loss smaller. More importantly, as shown in [32], combined CC and interleaver designs can lower the BER floor of rate-1/3 binary turbo codes to a BER of almost 10^{-7} . If this technique can be employed in our TTCM, for a BER requirement of 10^{-7} in ADSL DMT systems, and without considering impulse noise, no concatenated coding schemes is required. The performance of TTCM using rate-1/3 turbo code is shown in Figure 4-34. This TTCM achieves 1.7 dB from the Shannon limit with a turbo block of 1024 bits, while the Shannon limit is 4.8 dB at a BER of 10^{-5} and the transmission rate is 3.3 bits per symbol.

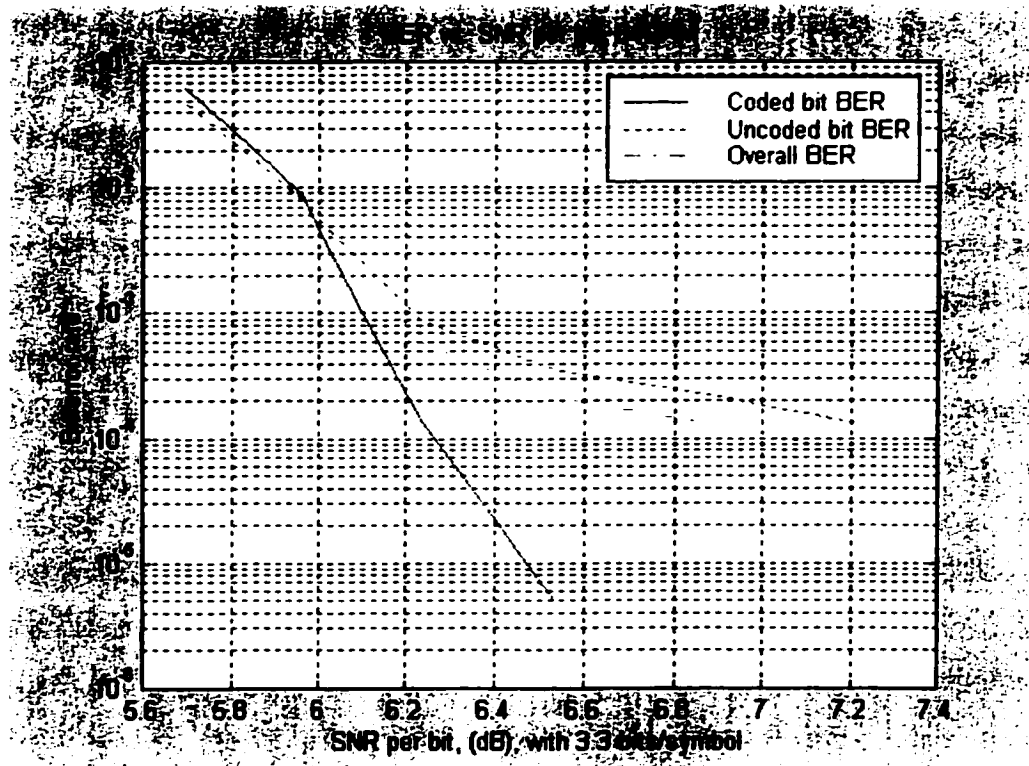


Figure 4-34 – Performance of 64QAM TTCM with rate 1/3 turbo code

64QAM, 3.3 bits/symbol, simplified decoding structure, 1024-bit interleaver, 10 iterations in decoding

4.6. SUMMARY

In this chapter, we investigated the performance of ADSL DMT systems with the proposed TTCM in a stationary noise environment.

- In section 4.1, we first proposed a flexible TTCM structure in ADSL DMT systems based on Benedetto's TTCM structure introduced in [3]. The TTCM operates across all the subchannels. The encoder of this TTCM consists of a turbo encoder and constellation mapper that maps the turbo-coded bits and un-turbo-coded bits into a single QAM symbol. For implementation simplicity, each QAM symbol contains a fixed number of turbo-coded bits and a variable number of un-turbo-coded bits to meet the various transmission capabilities of different subchannels.

- Decoding of the proposed TTCM can be performed by using SMAP decoding. To avoid the excessive complexity of SMAP algorithm, we proposed a simplified TTCM decoding structure, which employs a conventional binary turbo decoder.
- To lower the BER floor at the output of a typical binary turbo decoder, we proposed to use a concatenated RS-TTCM coding structure. The outer RS code is used to lower the BER floor at the output of the turbo decoder that is usually around 10^{-5} .
- In section 4.2, we presented a performance analysis of the proposed TTCM. To improve the performance, Gray coding is used in the constellation mapping. The effect of employing Gray mapping to the TTCM performance was investigated.
- Based on the discussion on the effect of using Gray mapping, a BER upper bound was derived and presented with the help of the *uniform interleaver*.
- It was shown that our TTCM have a performance being at least better than a turbo-coded BPSK transmission with the same binary turbo decoder over the same AWGN channel, when the BPSK and our QAM constellations have the same minimum distance. Performance estimation of the proposed TTCM was also derived based on the best-reported binary turbo decoding performance in the literature.
- Based on the observations in the performance estimations, a discussion on the design criteria of the turbo code to be used in the proposed TTCM structure was presented.
- In section 4.3, we presented the simulation results of the proposed TTCM structure in a 64QAM transmission. The performance of employing the concatenated RS-TTCM in QAM transmission was also presented.
- A discussion was given on the requirements to the outer RS code that was used in the concatenated RS-TTCM structure. Several methods of reducing the system delay were proposed.
- At the end, we proved that although the above simulation results are for QAM transmissions, they can be used directly for the performance evaluation of ADSL DMT systems with TTCM.
- In addition to the results obtained from analysis and simulation, we presented several possible options to further improve the TTCM performance. First of all, an SISO algorithm that tries to also minimize the parity bit error rate should be used. Secondly, since the simplified TTCM decoding uses a binary turbo decoder, various techniques proposed in the literature to improve the performance of binary turbo

code could be employed directly in our TTCM system for performance improvement.

Appendix – Error probability of uncoded bits when error turbo decoding occurs

Taking the 64QAM with TTCM proposed in this chapter as an example, the two uncoded are used to determine the different subsections. Since each uncoded bit is used independently along one dimension, the analysis here will be concentrate on the error rate of one of these two bits along the in-phase axis. We will show here that when an error decoding occurs, the error probability of uncoded bits is more than 1/8.

The constellation mapping is along the in-phase axis is shown in Figure 4-5. It is noted that the constellation points on both sides of the zero point are symmetric, therefore, only those on one side needs to be analyzed. The signals on the other side should give exactly the same results. In the following, we will only consider the signal points on the negative side of the zero point, i.e., an uncoded bit of '0' is assumed.

Assuming the signal points $[-7, -5, -3, -1]$ are transmitted with equal probability of 1/4. It is further assumed that an error turbo decoding generates totally random decisions, where a transmitted signal will be decoded on any of the four possibilities with equal probability of 1/4. The error probability of this uncoded bit is shown for different conditions as shown in Table 4-11.

Table 4-11 – Decision error probabilities of uncoded bits in 64QAM TTCM with assumed transmitted signals and turbo decoding results

Decoding results	Decoding threshold	Value of n_0 to make a mistake according to the transmitted symbol ($n_0 > \dots$ / probability)			
		$-7 + n_0$	$-5 + n_0$	$-3 + n_0$	$-1 + n_0$
-7	-3	4 / (0)	2 / (0)	0 / (1/2)	-2 / (1)
-5	-1	6 / (0)	4 / (0)	2 / (0)	0 / (1/2)
-3	1	8 / (0)	6 / (0)	4 / (0)	2 / (0)
-1	3	10 / (0)	8 / (0)	6 / (0)	4 / (0)

The total error probability is therefore calculated as

$$\begin{aligned}
 p_e &= p_{ir}(-3) \cdot p_{rc}(-7) \cdot p_e(-3, -7) + p_{ir}(-1) \cdot (p_{rc}(-7) \cdot p_e(-1, -7) + p_{rc}(-5) \cdot p_e(-1, -5)) \\
 &= \frac{1}{4} \cdot \frac{1}{4} \cdot \frac{1}{2} + \frac{1}{4} \cdot \left(\frac{1}{4} \cdot 1 + \frac{1}{4} \cdot \frac{1}{2} \right) \\
 &= \frac{1}{8}
 \end{aligned}$$

Eq. 4-49

It is therefore concluded that when an turbo decoding error occurs, the uncoded bits associated with the error-decoded bits (systematic and parity) have an error probability of at least 1/8.

Chapter 5.

Turbo Codes Against Impulse Noise

Non-stationary impulse noise is one of the major impairments in high-speed subscriber line transmissions. While the FEC structure used in an ADSL DMT system provides good coding gain against stationary noise (mainly crosstalk noises), it must also guarantee a satisfactory transmission performance in the presence of impulse noise.

In this chapter, we investigate the performance of the proposed TTCM in the presence of impulse noise. We start from standard turbo coded BPSK transmissions and a general single-carrier QAM transmission with the proposed TTCM. We observe that impulse noise could be devastating to the turbo decoding performance. Although an outer RS code can be employed to combat the impulse noise, it results in extremely long end-to-end transmission delay. In this work, we propose to use erasure turbo decoding in an impulsive channel to improve the TTCM decoding performance. It is further shown that when a concatenated RS-TTCM coding structure is employed, besides the BER performance improvement, erasure turbo decoding also significantly reduces the interleaving requirement of the outer RS code and therefore the end-to-end system delay.

Finally, a performance analysis and simulation results of QAM transmissions with TTCM are applied to estimate the performance of employing TTCM in ADSL transmissions with practical impulse noise models. It is shown that combating impulse noise in single-carrier ADSL transmissions is much easier than in multi-carrier ADSL transmissions.

In the following, we start with a discussion on general binary turbo codes against impulse noise and move to employing TTCM in fixed-size constellation QAM transmissions. At last, as a specific application, the performance of single-carrier ADSL transmissions and ADSL DMT systems with the proposed TTCM are evaluated and presented.

5.1. IMPULSE NOISE IN SUBSCRIBER LOOPS

Non-stationary impulse noise is characterized as infrequent, short-duration pulses with relatively high amplitudes; it is one of the major impairments in the digital subscriber-line transmissions. Impulse noise in the subscriber lines is generated by both internal sources and external sources. External sources include high-voltage devices, power plants and distribution, transportation means (railways, underground, ...), fluorescent tubes, lightning surges and electrostatic discharges. Internal sources include signaling in the forms of dialing pulses, busy signals, ringing and mechanical switch noise [38]. All these very different sources give impulse noise extremely complex statistical properties.

In previous research work, impulse noise was usually modeled as a sequence of high-amplitude pulses with infinitely short duration, because the length of an impulse is usually much smaller than symbol duration. The most important statistical parameters of this impulse noise model are the distributions of the sample amplitudes and the inter-arrival times, defined as the duration between the occurrences of two consecutive impulses. However, in high-speed digital subscriber line transmissions such as ADSL transmissions, impulse noise can no longer be modeled as having infinitely small duration because of the very short channel symbol duration. In this case, the impulse length and the impulse waveform become the other two important parameters in impulse noise description and modeling.

Since there are so many different factors in the occurrence of the impulse noise, the only way to derive analytical models for its statistical characterizations is to rely on extensive experimental measurements. A lot of work has been carried out in the literature in trying to get a proper impulse noise model, with which the performance of different transmission systems in the presence of impulse noise can be properly evaluated.

5.2. IMPULSE NOISE MODELING

Impulse noise in mobile communication environments and industrial noise environments in wide urban areas is well modeled by a stream of Poisson arriving impulses with tail areas distributed according to a “power” Rayleigh PDF. This impulse noise model closely models atmospheric and man-made noise [39]. The error performance analysis given in [39] shows that Fourier-Bessel analysis is a fast and accurate evaluation tool to assess the performance of impulse noise in QAM transmission systems. In [40], Ghosh showed that a much simpler impulse noise model, the Bernoulli-Gaussian impulse noise model in the discrete time domain, is equivalent to the continuous model discussed above and a closed mathematical expression can be derived easily based on this model. In this model, the occurrence of impulse noise is modeled as a Bernoulli process with a probability p_{imp} , and the amplitude of the impulse noise is assumed to have a Gaussian distribution. Therefore, impulse samples are modeled as a product of a real Bernoulli process and a complex Gaussian process as follows:

$$i_k = b_k \cdot g_k \tag{Eq. 5-1}$$

where b_k is a Bernoulli process, i.e., an identical-independently-distributed (*i.i.d.*) sequence of zeros and ones with $P_r(b_k=1)=p_{imp}$, and g_k is complex white Gaussian noise with a zero mean and a variance of $2\sigma_i^2$.

Unlike the case in wireless communications, the models of impulse noise in high-speed DSL communications are much harder to obtain. Various mathematical impulse noise models and practical impulse noise models based on on-site measurements have been obtained in both Europe and North America.

Since a typical impulse may last over several transmitted symbols in DSL transmissions, a good impulse noise model should represent the randomness of the waveform and its finite duration. In [41], Melbourne Barton introduced a method where

the impulse noise is modeled as an ensemble of random waveforms of finite duration. The peak impulse voltage distribution is modeled as Gaussian, Laplacian or hyperbolic. The occurrence of impulse noise is modeled as a Poisson process. It was shown later that although the Poisson process can give a good approximation to randomly occurring events in most cases, it is not favored to model impulse noise by the various surveys. Furthermore, the fixed impulse event durations and impulse waveforms are also simplifications of the practical impulses.

Other statistical impulse noise models are derived based on massive number of measurements and statistical analysis performed by several different research groups. By curve-fitting the survey results, relevant statistical impulse model parameters were obtained in Germany [42], Britain [43] and North America [44], [45], [46]. Furthermore, typical waveforms can be obtained from a fixed mean energy spectrum density (ESD) and a fixed mean phase. Note that a fixed mean ESD would be a relevant parameter since it doesn't limit the impulse noise waveform and duration at all.

5.2.1. Peak voltage distribution and sample amplitude distribution

Peak amplitude of impulse noise, V_R , is an identification flag of impulse noise. In any impulse noise measurement, whenever a sample with amplitude over a certain level is detected, it is said that an impulse occurs. In [47], based on 150,000 registered impulse events, the peak amplitude distribution is quite well expressed as

$$P_R(V_R) = \begin{cases} \left(\frac{V_R}{V_2}\right)^{-2} & \text{if } V_R \leq 40 \text{ mV, where } V_2 = 5 \text{ mV} \\ \left(\frac{V_R}{V_1}\right)^{-1} & \text{if } V_R \geq 40 \text{ mV, where } V_1 = 0.625 \text{ mV} \end{cases}$$

Eq. 5-2

$P_R(V_R) = P_R(V_R; V_{th})$ is the probability that the random variable V_R exceeds a given threshold, V_{th} . Furthermore, $P_R(V_R)$ is found to be independent of the observation period and the bandwidth of the receiver. Similar distribution curves for impulse peak amplitudes were obtained in the survey performed in Germany [42].

The peak voltage amplitude is usually used to estimate the occurrence frequency of impulse noise. However, it does not provide information on the noise amplitude

distribution during the impulse duration, which is necessary for the estimation of impulse noise impact on high-speed ADSL transmissions.

In [42], the PDF of the impulse noise voltage is reported to be well modeled by:

$$f_i(u) = \frac{1}{240u_0} e^{-\left|\frac{u}{u_0}\right|^{1/5}} \quad (u_0 > 0) \quad \text{Eq. 5-3}$$

where u_0 is a curve-fitting constant.

This distribution is a double exponential distribution with a power of 1/5 in the exponent and with the parameter $u_0 > 0$ V.

5.2.2. Inter-arrival duration

The occurrence of impulse noise can be roughly modeled by a Poisson process [47]. However, this model is not favored in [42], [43] and [48]. A generalized Poisson distribution introduced in [48] and a generalized exponential distribution in [42] and [43] are both found more suitable to describe practical impulse noise.

However, as long as the probability of the occurrence of two consecutive impulses is low enough, the most important parameter to characterize the inter-arrival duration is the mean inter-arrival duration, from which the average number of impulses per second can be estimated.

The relationship between the mean inter-arrival duration and the peak impulse amplitude is another important statistics of the impulse noise. Two surveys, from [43] and [47], give a similar relationship between the peak impulse amplitude and the mean inter-arrival time, $\bar{T}$, as,

$$\bar{T} \propto V_{th}^{2.5} \quad \text{Eq. 5-4}$$

where V_{th} is the peak amplitude threshold for impulse noise detection. This means that the average inter-arrival duration of impulse noise with peak amplitude higher than V_{th} is proportional to $V_{th}^{2.5}$. The higher is the threshold, the less frequently the impulse noise occurs.

5.2.3. Length of impulse noise

A typical distribution of the impulse noise length, $f(t)$, can be approximated by the sum of two log-normal densities [42]:

$$f_t(t) = B \frac{1}{\sqrt{2\pi s_1 t}} e^{-[\ln^2(t/t_1)]/(2s_1^2)} + (1-B) \frac{1}{\sqrt{2\pi s_2 t}} e^{-[\ln^2(t/t_2)]/(2s_2^2)} \quad \text{Eq. 5-5}$$

where t_1, t_2 are the median values and s_1, s_2 are the shape parameters of the log-normal densities, and B is a curve-fitting constant.

5.2.4. Simulation of impulse noise

To conduct simulations of transmission systems in an impulse noise environment, a proper, yet simple, mathematical impulse noise model is required. In this work, several simplifications were made for convenience of both analysis and simulation, which might not comply with those survey results introduced in the preceding sections.

5.2.4.1. Gaussian burst model

As mentioned above, the occurrence of impulse noise approximately follows the Poisson distribution. However, as long as there is little possibility of two impulses occurring at the same time, the impulse noise can simply be described as evenly occurring events with a mean inter-arrival duration. In ADSL DMT systems, the impulse noise usually has much shorter duration than one DMT symbol. The position of the impulse in one DMT symbol is assumed uniformly distributed, and the situation when one impulse having impacts on two adjacent DMT symbols is ignored [49].

The distribution of impulse sample amplitudes is necessary to estimate the performance of the transmission during the impulse duration. It is one of the key parameters of the impulse model. For analysis convenience, our simple model assumes that the impulse samples have a Gaussian distribution with a zero mean and a constant variance.

In a practical model, the sample variance should have a certain statistical distribution. This could be obtained in our model by adjusting the noise variance. The peak amplitude distribution can be obtained from the impulse noise surveys in the literature. In our model, the peak amplitude of the impulse, V_p , can be expressed as an amplitude level

over which the impulse noise sample amplitudes, V_{imp} , exceed with a large enough probability, p_{th} . Given this p_{th} , the noise variance is easily obtained as the value which satisfies,

$$p(|V_{imp}| > V_p) = 2 \cdot Q(V_p / \sigma_i) = p_{th} \quad \text{Eq. 5-6}$$

where σ_i is the standard deviation of the impulse samples.

For the length of the impulse noise, the distribution proposed in [42] can be incorporated into the simulation and analytical work quite straightforwardly. The only concern is that the distribution of the impulse length is quite different from the survey results presented from those in North America. Therefore, applying this distribution to obtain a common result is questionable. Instead, in our simulations, we use a fixed impulse length. Simulation results were obtained with different impulse lengths. These results can then be used for either worst-case performance estimation or an average performance evaluation.

Non-stationary impulse noise is not confined to certain waveforms, since it is not generated from fixed sources. All the surveys give symmetric distribution of impulse samples with zero mean, i.e., no DC component, since impulses come from a combination of various unknown sources, most of which give no or little DC component. Therefore, the waveforms of the impulse are assumed randomly distributed, consisting of *i.i.d.* Gaussian distributed impulse samples. This does not totally comply with the previous survey results, where the impulse samples have a much flatter distribution, such as the double exponential distribution. Later in this chapter, we will show that in DMT systems, as long as the mean and variance of the impulse samples are known, it is not necessary to know the exact distribution of the impulse noise samples to evaluate the system performance.

In summary, the impulse noise is modeled as a fixed-length vector of *i.i.d.* zero-mean Gaussian random variables with a constant variance. The impulses come every T seconds, where T is the mean inter-arrival duration. The variance of the impulse noise has a distribution determined by the peak amplitude distribution and the pre-defined threshold-exceeding probability, p_{th} . The amplitude distribution can be further modeled by the double exponential distribution, with parameters taken from [42].

5.2.4.2. Markov burst channel model

A more realistic model of a subscriber loop with impulse noise for high-speed data transmission is the two-state Markov burst channel model. In this channel, the noise is generated according to a two-state hidden Markov model (HMM), as shown in Figure 5-1. The two states are the 'Good state' (G-state), when only normal background noise¹⁶ presents and the 'bad state' (B-state), with both background noise and impulse noise. When the channel is in one state, it tends to stay in the same state and will switch into the other state with only a small probability. Usually, the probability of transition from the B-state to G-state is much higher than the transition from the G-state to the B-state. This corresponds to a burst channel with much longer Good state period than Bad state period.

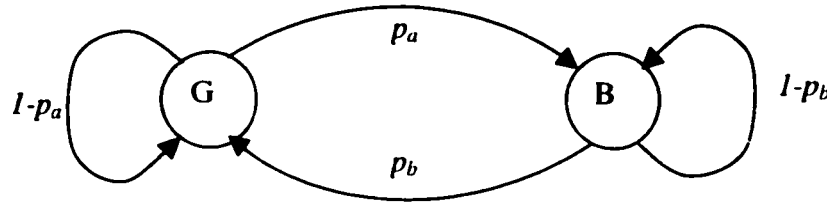


Figure 5-1 – Two-state Markov channel model

As shown in Figure 5-1, the channel is modeled as a Markov process where the current state depends only on the previous state. When the previous state is the G-state, the channel has a probability of $(1-p_a)$ to stay in the G-state, and a probability of p_a to switch into the B-state. It is the same if the last state is the B-state, only with a different transition probability of p_b . Since impulse noise is a relatively rare phenomenon in subscriber loop channels, p_a tends to be very small. The average occurrence rate, λ , of the impulsive noise is given by,

$$\lambda = \frac{p_a}{p_a + p_b}$$

Eq. 5-7

¹⁶ The normal background noises are thermal noise and crosstalk noises in most modern subscriber line transmission systems.

where the average burst length is

$$l_{ave} = \frac{1}{p_b}$$

Eq. 5-8

To simplify the analysis, impulse samples are still assumed to be Gaussian distributed and independent of each other. This is in accordance with the randomly distributed impulse samples. Again, a Gaussian distribution doesn't comply with the previous survey results.

In turbo coded QAM transmissions, the transmitted symbol sequence is divided into turbo blocks, where each block contains one turbo code word. Assuming that each turbo blocks covers N_s QAM symbols, there could be various number of impulses with various lengths within this turbo block. To evaluate the performance of QAM transmissions with TTCM in an impulse noise environment modeled as a Markov burst channel, it is necessary to know what impulse patterns give the most significant error contribution.

We start by finding the probability of l impulse samples falling in one turbo block. Let's assume that n impulse events occur in this block contributing totally l impulse samples, with starting points $\{i_1, i_2, \dots, i_n\}$ and lengths $\{l_1, l_2, \dots, l_n\}$. The two adjacent impulses are separated by at least one sample of good state. For any turbo block, four situations exist as shown in Figure 5-2 and the probabilities of any sample sequences of these four conditions are given in Eq. 5-9.

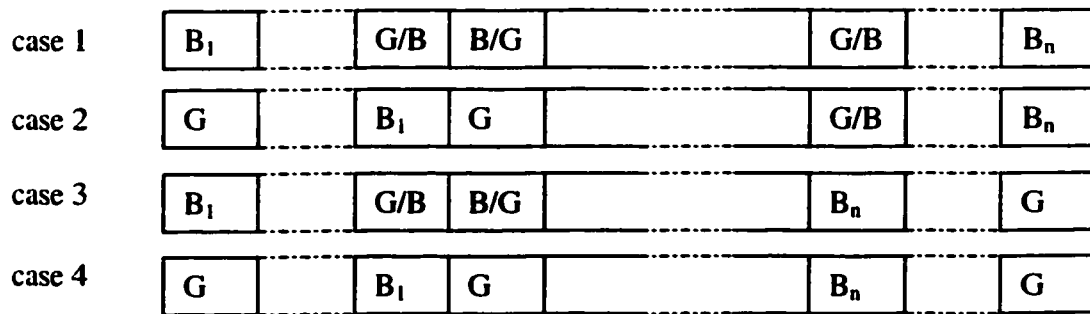


Figure 5-2 – Four cases of n separated impulses in one DMT block

$$\begin{aligned}
 p(\text{case } 1) &= P_B \cdot (p_a \cdot p_b)^{n-1} \cdot (1-p_b)^{l-n} \cdot (1-p_a)^{(N_s-l)-(n-1)} \\
 p(\text{case } 2) &= P_G \cdot p_a \cdot (p_a \cdot p_b)^{n-1} \cdot (1-p_b)^{l-n} \cdot (1-p_a)^{(N_s-l)-n} \\
 p(\text{case } 3) &= P_B \cdot p_b \cdot (p_a \cdot p_b)^{n-1} \cdot (1-p_b)^{l-n} \cdot (1-p_a)^{(N_s-l)-n} \\
 p(\text{case } 4) &= P_G \cdot (p_a \cdot p_b)^n \cdot (1-p_b)^{l-n} \cdot (1-p_a)^{(N_s-l)-(n+1)}
 \end{aligned}$$

Eq. 5-9

where P_B is defined as the probability that the first sample of this turbo block is in B-state, and P_G for G-state, where $P_B=\lambda$ as shown in Eq. 5-7 and $P_G=1-\lambda$.

Denoting the number of impulse events in case i as N_n^i , the probability of one turbo block containing n impulse events contributing to a total number of l impulse samples is,

$$P_n(l) = \sum_{i=1}^4 N_n^i \cdot p(\text{case } i)$$

Eq. 5-10

The probability of a turbo block containing l impulse sample can thus be calculated as

$$P(l) = \sum_{n=1}^{N_s/2} P_n(l)$$

Eq. 5-11

where $N_s/2$ is the maximum number of impulse events in one turbo block.

The derivation of $P_n(l)$ shows that

$$P_n(l) \propto (p_a \cdot p_b)^n \cdot (N_s - l)^n \cdot (l - n)^n$$

Eq. 5-12

Assuming an N_s of 512, a symbol rate of 10^6 symbols per second and an impulse occurrence every 10 seconds on the average, with a length shorter than 100 samples in most of the cases, the Markov model parameters can be calculated as $p_a=10^{-7}$, $p_b=1/100=0.01$. Using Eq. 5-12, it is shown that the probability of two impulse events occurring in one turbo block contributing l impulse samples is much smaller than that of a single impulse event.

It is therefore concluded that one turbo block has a very small probability of having two or more impulse events.

$$\begin{aligned}
 \frac{P_2(l)}{P_1(l)} &= p_a \cdot p_b \cdot \left[(N_s - l) \cdot \frac{(l-2)^2}{l-1} \right] \\
 &< p_a \cdot p_b \cdot [(N_s - l) \cdot l] \\
 &< p_a \cdot p_b \cdot \frac{N_s^2}{4} \\
 &= 6.6 \cdot 10^{-5}
 \end{aligned}$$

Eq. 5-13

With the approximation of only one impulse event in the current turbo block, the occurrence probability of an impulse with length l is easily calculated as:

$$\begin{aligned}
 P(l) &\approx \sum_{i=1}^4 N_i' \cdot p(\text{case } i) \\
 &= \left\{ p_a \cdot p_b \cdot [(1-p_a) \cdot (1-p_b)]^{-1} \cdot (1-p_a)^{N_s-1} / (p_a + p_b) \right\} \\
 &\quad \cdot \left[(1+p_a)^2 + (1+p_a) + p_b \cdot (N_s - l - 1) \right] \cdot \left(\frac{1-p_b}{1-p_a} \right)^l
 \end{aligned}$$

Eq. 5-14

Note that Eq. 5-14 is the actual probability normalized by the average impulse occurrence probability in the current DMT symbol, which is calculated as:

$$P_{ave} = 1 - P_G \cdot (1 - p_a)^{N_s-1}$$

Eq. 5-15

The impulse length distribution in one turbo block for an impulse occurrence rate of one impulse every 10 seconds is computed and shown in Figure 5-3.

5.2.4.3. Discussions

The simplest impulse model of a Gaussian distributed sample sequence is easily applied to simulation and analysis, because the impulse length, impulse occurrence probability and impulse variance can all be independently set. Since impulses in subscriber lines are usually much shorter than a DMT symbol (0.25 ms), for analysis simplicity, the situation of an impulse crossing two consecutive DMT symbols will not be considered.

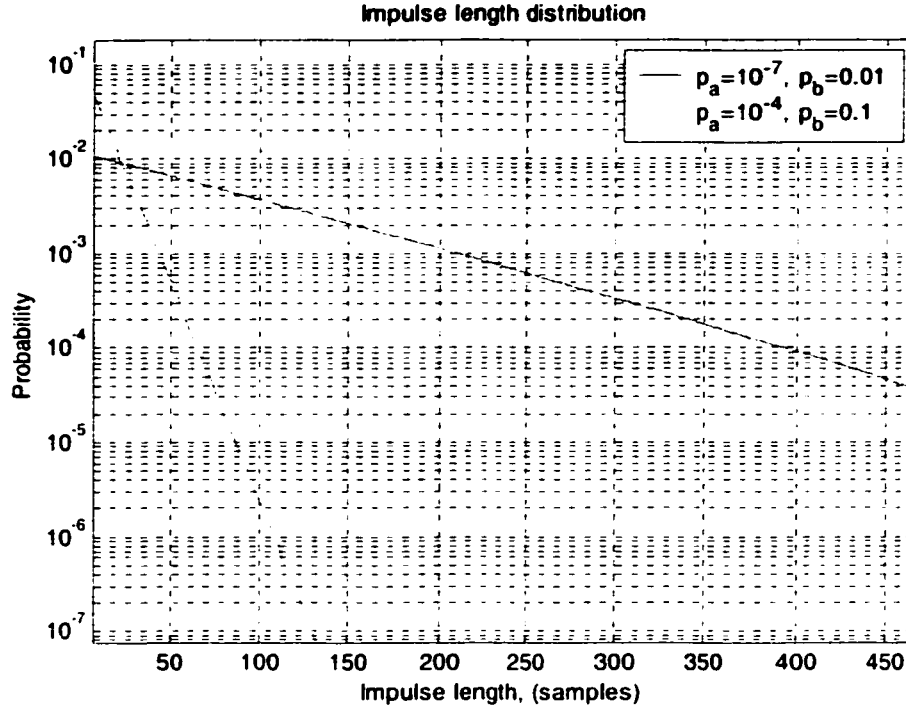


Figure 5-3 – Impulse length distribution, $p_a=10^{-7}$ with $p_b=0.01$, and $p_a=10^{-4}$ with $p_b=0.1$, $N_c=512$

The two models we proposed above do not completely comply with the measurements obtained in previous surveys. Nevertheless, we will still use these two models because they offer the following advantages:

- They are not limited to a specific on-site measurement.
- They simplify the analysis and simulation because of the simple distributions.
- It is easy to adjust the parameters to approximate any on-site measurements, or at least to provide a worse case condition.

5.3. EFFECT OF IMPULSE NOISE TO TURBO DECODING

In this section, we perform an analysis on the effect of impulse noise to turbo-coded systems. To simplify the analysis, an overall interleaver is applied to the turbo-coded block, including systematic bits and parity bits. This interleaver randomizes the impulse noise samples in one turbo block. In the situation where a turbo block is hit by an impulse of length l_{imp} and variance σ_i^2 , after this interleaver, the impulse samples can be modeled as a transmission with Bernoulli-Gaussian sample sequence, where the impulse sample

occurrence probability $p_{imp} = l_{imp}/N_s$, and impulse variance is σ_i^2 . Note that, for binary transmissions, N_s is equal to the number of bits in one turbo block, N .

5.3.1. Effect of impulse noise in turbo coded single carrier BPSK systems

With the Bernoulli-Gaussian model of impulse noise and assuming a turbo code with a information word length of K bits and a codeword length of N bits, the probability of l impulse samples occurring in one turbo block is

$$p(l) = \binom{N}{l} p^l \cdot (1-p)^{N-l} \quad \text{Eq. 5-16}$$

For an error path of weight $d = w + j$, where w is the number of errors in systematic bits and j is the number of errors in parity bits, the code word error probability with the l impulse sample in the current turbo block, $p_e(l)$, can be expressed as

$$p_e(l) = \sum_{i=1}^d p(N_l = i) \cdot p(M_0 < M_i) \quad \text{Eq. 5-17}$$

where N_l is the number of impulse samples located in the d error positions and M_i is the metric calculated by the maximum likelihood decoder.

The probability that i out of the l impulse samples fall into the d differing positions can be calculated as,

$$p(N_l = i) = \frac{\binom{d}{i} \binom{N-d}{l-i}}{\binom{N}{l}} \quad \text{Eq. 5-18}$$

The metrics can be expressed as

$$M_0 = \sum_{k=1}^d (-\sqrt{E_c} + n_k) \cdot (-1) \quad \text{Eq. 5-19}$$

and

$$M_i = \sum_{k=1}^d (-\sqrt{E_c} + n_k) \quad \text{Eq. 5-20}$$

where n_k is the noise sample on the k^{th} error position and E_c is the average channel symbol energy.

The decoder will make an error decision if $M_0 < M_1$, the word error probability is therefore calculated as,

$$\begin{aligned}
 p(M_0 < M_1) &= p\left(\sum_{k=1}^d n_k > d \cdot \sqrt{E_c}\right) \\
 &= p\left(\sum_{k=1}^i n_k(0, N_i + N_0) + \sum_{k=1}^{d-i} n_k(0, N_0) > d \cdot \sqrt{E_c}\right) \\
 &= Q\left(\frac{d \cdot \sqrt{E_c}}{\sqrt{\frac{d \cdot N_0 + i \cdot N_i}{2}}}\right) \\
 &= Q\left(\sqrt{\frac{2 \cdot d \cdot R_c \cdot \gamma_b}{1 + R_i \cdot R_N}}\right) \\
 &= Q\left(\sqrt{2 \cdot d \cdot R_c \cdot \bar{\gamma}_b}\right)
 \end{aligned}$$

Eq. 5-21

where N_0 and N_i are the single-sided PSD of the impulse noise, R_c is the channel coding code rate, $R_i = i/d$, $R_N = N_i/N_0$, $\gamma_b = E_b/N_0$ and $\bar{\gamma}_b = \gamma_b/(1 + R_i R_N)$.

The error probability between these two code words with a total of l impulse samples in the current turbo block is therefore expressed as

$$P_e(l) = \sum_{i=0}^d p(N_l = i) \cdot Q\left(\sqrt{2 \cdot d \cdot R_c \cdot \bar{\gamma}_b(d, i)}\right)$$

Eq. 5-22

and the bit error probability is

$$P_b(l) = \sum_{i=0}^d \frac{\omega}{K} p(N_l = i) \cdot Q\left(\sqrt{2 \cdot d \cdot R_c \cdot \bar{\gamma}_b(d, i)}\right)$$

Eq. 5-23

where ω is the number of erroneous systematic bits.

The overall bit error rate is the mean of the $P_b(l)$ over all possible values of l .

$$P_b = \sum_{l=0}^{N'} p(l) \cdot P_b(l)$$

Eq. 5-24

Using the approximation

$$Q(\sqrt{2 \cdot \gamma_b \cdot R_c \cdot d}) \leq e^{-\gamma_b \cdot R_c \cdot d} = D^d \Big|_{D=e^{-\gamma_b R_c}} \quad \text{Eq. 5-25}$$

the bit error probability can be expressed as

$$P_b(l) \leq \sum_{i=0}^d \frac{\omega}{K} \cdot p(N_l = i) \cdot D^d \Big|_{D=e^{-\gamma_b R_c}} \quad \text{Eq. 5-26}$$

The weight enumeration function of a turbo code is defined as

$$A^{C_r}(W, Z) = \sum_{\omega, j} A_{\omega, j} \cdot W^{\omega} \cdot Z^j \quad \text{Eq. 5-27}$$

and the union bound of BER has the following expression as

$$P_b(e) \leq \frac{W}{K} \cdot \frac{\partial A^{C_r}(W, Z)}{\partial W} \Big|_{W=Z=e^{-R_c \cdot l_b / N_0}} \quad \text{Eq. 5-28}$$

For a fixed impulse length l and a fixed number of impulse samples in the weight-differential positions of i , the union bound of BER can be expressed as

$$\begin{aligned} P_b(l, i) &\leq \frac{W}{K} \cdot \frac{\partial A^{C_r}(W, Z)}{\partial W} \Big|_{W=Z=e^{-R_c \cdot l_b}} \\ &= \sum_{\omega=\omega_{\min}}^K \frac{\omega}{K} \cdot \sum_j A_{\omega, j} \cdot W^{\omega} \cdot Z^j \Big|_{W=Z=e^{-R_c \cdot l_b}} \\ &= \sum_{\omega=1}^K \frac{\omega}{N} \cdot W^{\omega} \cdot Z^j \cdot A^{C_r}(\omega, Z) \Big|_{W=Z=e^{-R_c \cdot l_b}} \end{aligned} \quad \text{Eq. 5-29}$$

where $A^{C_r}(\omega, Z)$ is the conditional weight enumeration function(CWEF) with input weight parameter ω for the turbo code.

Assuming a *uniform interleaver* is used, the CWEF can be calculated as

$$A^{C_r}(\omega, Z) = \frac{[A^C(\omega, Z)]^2}{\binom{N}{\omega}} \quad \text{Eq. 5-30}$$

where $A^C(\omega, Z)$ is the CWEF of the CCs.

The average BER with a fixed impulse length is therefore calculated as,

$$P_b(l) \leq \sum_{w=w_{\min}}^K \frac{w}{K} \cdot \sum_j A_{w,j} \cdot \sum_{i=0}^{w+j} p(N_l = i) \cdot W^w \cdot Z^j \Big|_{W=Z=\epsilon^{-R_c \tilde{r}_b}} \quad \text{Eq. 5-31}$$

and the average BER is calculated as:

$$P_b \leq \sum_{w=w_{\min}}^K \frac{w}{K} \cdot \sum_j A_{w,j} \cdot \sum_{l=0}^N p(l) \cdot \sum_{i=0}^{w+j} p(N_l = i) \cdot W^w \cdot Z^j \Big|_{W=Z=\epsilon^{-R_c \tilde{r}_b}} \quad \text{Eq. 5-32}$$

Applying Eq. 5-16 and Eq. 5-18 into Eq. 5-32, we obtain

$$P_b \leq \sum_{l=0}^N p^l \cdot (1-p)^{N-l} \cdot \left\{ \sum_{w=w_{\min}}^K \frac{w}{K} \cdot A_{w,j} \cdot \sum_{i=0}^{w+j} \binom{w+j}{i} \binom{N-w-j}{l-i} \cdot W^w \cdot Z^j \Big|_{W=Z=\epsilon^{-R_c \tilde{r}_b}} \right\} \quad \text{Eq. 5-33}$$

Eq. 5-30 and Eq. 5-33 give the BER union bound for turbo coded BPSK systems with Bernoulli-Gaussian modeled impulse noise, assuming a *uniform interleaver* is used in the turbo code. This guarantees that at least one interleaver could be found to give a better performance.

Without the additional interleaver, a better model is a consecutive block of impulse noise samples having a Gaussian distribution with a very high variance. With such an impulse noise model, it is difficult to give an explicit expression for the performance of the turbo-coded systems.

5.3.2. Effect of impulse noise in turbo coded single carrier QAM systems

Direct analysis of the effect of impulse noise in turbo-coded single carrier QAM system is difficult, because no explicit relationship exists between the received QAM symbol samples and the binary turbo decoding performance. However, with the help of Gray mapping and bit reliability calculation, a similar analysis can be carried out as presented in last section. However, two differences exist:

- ❖ Firstly, the soft binary information is no longer Gaussian distributed, as shown in the previous chapter. Performance of the QAM transmissions with TTCM, using simplified TTCM decoding based on the equivalent soft binary symbols (ESBS), is at least as good as the performance of the same turbo decoding on a BPSK transmission with the same minimum constellation distance, over the same AWGN channel.

- ❖ Secondly, impulse noise will not give a very high binary information values for large QAM constellations except when the sample amplitude is much higher than the received symbols. Even if the signal-to-impulse-noise ratio could be very low when the impulse noise samples have a large variance, in a QAM transmission with a large constellation, a large impulse sample would likely cause the received symbols to still fall into the constellation. The calculated ESBS will not show a noise value as large as the real impulse noise amplitude. Therefore, the effect of a large amplitude impulse noise is reduced.

Therefore, over the same AWGN channel and with the same impulse noise, a QAM transmission with TTCM using simplified TTCM decoding based on the ESBS should have better performance than that of a turbo coded BPSK transmission with the same minimum constellation distance.

5.3.3. Effect of impulse noise in turbo coded ADSL DMT systems

In ADSL DMT systems, the power of the impulse noise is spread into every subchannel in the current DMT symbol. For impulses with moderate power and short duration, this property gives DMT systems inherent impulse noise robustness. For impulses with very high power, DMT will give worse performance because the information in all subchannels in the current DMT symbol can be lost. However, impulses with very high power have a very small probability of occurrence.

The effect is much easier to analyze if the Bernoulli-Gaussian model is assumed. The output of the DFT demodulation at the receiver is actually a weighted sum of the input signals. If *i.i.d.* impulse samples are assumed, the sum will have a normal distribution, according to the law of large-numbers. Assuming a perfect synchronization and an ideal channel, the received symbol is given by

$$r_k = s_k + w_k + i_k, \quad k = 0, 1, \dots, 2N_c - 1$$

Eq. 5-34

where w_k is the AWGN noise sample, i_k is the impulse noise sample and s_k is the time domain samples sent into the channel and is obtained by performing an IDFT operation over the QAM symbols,

$$s_k = \frac{1}{\sqrt{2N_c}} \sum_{n=0}^{2N_c-1} a_n \cdot e^{j \frac{2\pi nk}{2N_c}}, \quad k = 0, 1, \dots, 2N_c - 1$$

Eq. 5-35

where, a_n are the QAM symbols and N_c is the number of carriers used in the modulation schemes. The actual QAM sequence is only of length N_c . The length- $2N_c$ sequence is generated by concatenating the actual QAM sequence with its conjugate symmetric sequence.

At the receiver, the demodulation is performed by a DFT operation.

$$\begin{aligned} R_k &= \frac{1}{\sqrt{2N_c}} \sum_{n=0}^{2N_c-1} r_n \cdot e^{-j \frac{2\pi nk}{2N_c}} \\ &= DFT\{\bar{s}\}_k + DFT\{\bar{w}\}_k + DFT\{\bar{i}\}_k \\ &= a_k + W_k + I_k, \quad k = 0, 1, \dots, 2N_c - 1 \end{aligned}$$

Eq. 5-36

where W_k is an AWGN with variance $2\sigma_w^2$ and I_k is the DFT output resulted from the impulse samples:

$$I_k = \frac{1}{\sqrt{2N_c}} \sum_{n=0}^{2N_c-1} i_n \cdot e^{j \frac{2\pi nk}{2N_c}}, \quad k = 0, 1, \dots, 2N_c - 1$$

Eq. 5-37

The total noise in the k^{th} subchannel is $N_k = W_k + I_k$. The distribution of W_k is still Gaussian distributed and has a zero mean and a variance equal to that of w_k . The distribution of I_k is now another Gaussian variable, which has the following statistics:

$$E\{I_k\} = \frac{1}{\sqrt{2N_c}} \sum_{n=0}^{2N_c-1} E\{i_n\} \cdot e^{j \frac{2\pi nk}{2N_c}}, \quad k = 0, 1, \dots, 2N_c - 1$$

Eq. 5-38

and

$$Var\{I_k\} = \frac{1}{2N_c} \sum_{n=0}^{2N_c-1} Var\{i_n\}, \quad k = 0, 1, \dots, 2N_c - 1$$

Eq. 5-39

where the mean and variance of the noise components in all the subchannels are the same.

It is easy to prove that the AWGN noise, W_k , in the k^{th} subchannel is independent of W_j in the j^{th} subchannel, because of the orthogonality property of the frequency coefficients. However, this is not the case for I_k . If a zero-mean is assumed for the impulse noise in the channel, then I_k also has a mean of zero. The variance of I_k , however, becomes $(N_l/2N_c) \cdot \text{Var}\{i_n\}$, which is determined by the variance and the length of the impulse noise, σ_i^2 and N_l . The smaller is the impulse noise variance, the shorter is the impulse, the smaller the $\text{Var}\{I_k\}$ is. Because impulse noise is characterized by short-length and high-amplitude, the DMT system therefore averages its energy over all the subchannels.

A difficulty in analyzing the impact of the impulse noise in turbo-coded systems is the correlation between the impulse contributions in different subchannels. The correlation between the impulse contributions in subchannel k and subchannel i can be calculated as,

$$\begin{aligned} E\{I_k \cdot I_i^*\} &= \frac{1}{2N_c} E \left\{ \sum_{n=0}^{2N_c-1} i_n e^{\frac{-j2\pi nk}{2N_c}} \cdot \sum_{m=0}^{2N_c-1} i_m^* e^{\frac{j2\pi im}{2N_c}} \right\} \\ &= \frac{1}{2N_c} \sum_{n=0}^{2N_c-1} \sum_{m=0}^{2N_c-1} E\{i_n \cdot i_m^*\} e^{\frac{-j2\pi(nk-im)}{2N_c}} \\ &= \frac{\sigma_i^2}{2N_c} \sum_{n, i_k \neq 0} e^{\frac{-j2\pi n(k-i)}{2N_c}} \end{aligned}$$

Eq. 5-40

which is not zero in most of the cases. Apparently, the shorter is the impulse, the smaller is the the noise variance, the smaller the correlation between the impulse contributions in any two subchannels is. The correlation coefficient also depends on the parameter $(k-i)$ and the positions of the impulse samples in the DMT symbol. In subscriber line transmissions, an impulse event gives a consecutive sequence of impulse samples. Assuming an impulse noise with length l which starts from the k^{th} sample in one DMT symbol, from Eq. 5-40, the correlation coefficient, ρ , between impulse contributions in two channels spaced d_{ch} subchannels from each other can be calculated as shown in Eq. 5-41.

$$\begin{aligned}
 \rho &= abs \left\{ \frac{E \{ I_k \cdot I_{k+d_{ch}}^* \}}{\sigma_i^2} \right\} = abs \left\{ \frac{1}{2N_c} \sum_{n=k}^{k+l-1} e^{\frac{-j 2\pi n d_{ch} l}{2N_c}} \right\} \\
 &= abs \left\{ \frac{1}{2N_c} \cdot \frac{e^{\frac{-j 2\pi k d_{ch}}{2N_c}} \cdot \left(1 - e^{\frac{-j 2\pi d_{ch} l}{2N_c}} \right)}{1 - e^{\frac{-j 2\pi d_{ch}}{2N_c}}} \right\} \\
 &= \frac{1}{2N_c} \cdot \frac{\sin \left(\frac{\pi d_{ch} l}{2N_c} \right)}{\sin \left(\frac{\pi d_{ch}}{2N_c} \right)}
 \end{aligned}$$

Eq. 5-41

The correlation coefficients calculated using Eq. 5-41 are plotted in Figure 5-4 as a function of channel spacing. It is shown that the adjacent channels are strongly correlated. As the distance between the two subchannels increases, the correlation decreases very fast. Beyond 10 samples, the correlation coefficients are lower than 0.025.

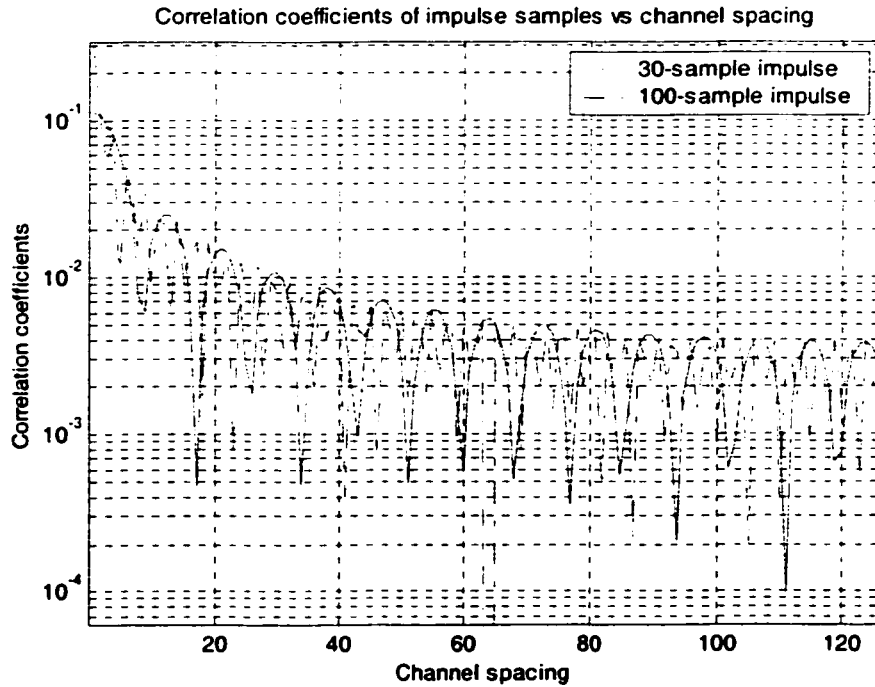


Figure 5-4 – Cross-channel-correlation of impulse noise contributions in DMT systems
DMT system with 128 subchannels, 256-point FFT and IFFT

In the turbo coded ADSL DMT systems proposed in section 4.1, the Turbo encoder and decoder operate across all subchannels, i.e., each turbo block contains bits transmitted in all the subchannels. The SISO decoding of the CCs depends on independent adjacent symbols to produce the extrinsic information, which will be fed into the next decoding stage. The correlation between the symbols will apparently have a negative impact on the decoding performance.

It is observed from Figure 5-4 that the correlation between the small-spaced subchannels is fairly high, which might give significant performance degradation. Employing a block interleaver can help to decrease the correlation between the received symbols in adjacent subchannels. It is stated in [1] that the larger the distance between the correlated bits is, the less the effect to the decoding performance will be. As long as the highly correlated bits are separated far from each other, the performance degradation can be reduced to a satisfactorily small value. However, this will introduce extra delays for the encoding and decoding processes of the turbo code. Another solution to mitigate this correlation problem is to incorporate it into the bit allocation procedure. Instead of allocating the turbo-coded bits sequentially to the subchannels, an interleaved order can be used, which also achieves the goal of the channel interleaving.

The cross-channel-correlation of the impulse samples has been shown to be small and can be further decreased. To simplify the analysis, in this section and all the following sections, the assumption of no correlation will be made in ADSL DMT systems.

With this assumption, an analysis of the TTCM in ADSL DMT systems in impulse noise environment becomes straightforward. First, the soft LLR values of the turbo-coded bits are obtained from the received QAM symbols. The effect of impulse noise is then equivalent to an additional AWGN noise to every QAM symbol, which is in turn translated into an AWGN noise in every binary symbol (ESBS). A conventional binary turbo decoding is performed for the decoding.

The analysis performed in last section can be applied to estimate the performance of a turbo coded ADSL DMT system. It has been shown above that the impulse samples are added to the time-domain DMT symbol samples. The received samples are demodulated with a DFT operation. After the DFT demodulation, a short duration impulse is transformed into an equivalent impulse with a fixed duration (one DMT symbol duration)

and various impulse noise variances. The variance of the equivalent impulse is proportional to the length and variance of the actual impulse. If the random interleaver used in section 5.3.1 at the output of the turbo encoder is still assumed, the turbo decoder is looking at a channel with Bernoulli-Gaussian modeled impulse noise. Ignoring the case where two impulses occur in one turbo block, for each turbo block containing impulse noise, the impulse samples has a fixed occurrence probability. The equivalent impulse noise has a variant variance depending on the distribution of the length of the actual impulse noise. The distribution of impulse length can be obtained from the surveys in the literature. The performance of turbo coded ADSL DMT systems can therefore be estimated as

$$P_b = \sum_{l_{imp}} p(l_{imp}) \cdot P_b(l_{imp})$$

Eq. 5-42

where $p(l_{imp})$ is the probability the impulse noise has a length of l_{imp} , $P_b(l_{imp})$ is the average BER of the turbo code containing an impulse noise of length l_{imp} , Eq. 5-31.

5.4. SIMULATIONS ON EFFECT OF IMPULSE NOISE

This section presents the simulation results on the performance of turbo-coded single-carrier and multi-carrier transmission systems in the presence of impulse noise. Based on the simulation observations, four techniques are proposed to as the techniques to improve the impulse noise combating capability: erasure turbo decoding, erasure RS decoding, adding an external interleaver and concatenated RS-TTCM. Significant performance improvement is observed by employing erasure techniques.

The following assumptions are made in the simulations:

- Gaussian burst model is used for impulse noise simulation.
- Rate-1/2 or rate-1/3 turbo codes are simulated.
- Length 1024-bit turbo code interleaver is used in the simulation.

5.4.1. Turbo-coded BPSK transmissions in impulse noise environment

This section gives the performance of turbo coded BPSK transmission in impulse noise environment.

5.4.1.1. Simulation results

Simulations were performed with two impulses of 50-symbol duration ($L_{imp}=50$) and 100-symbol duration ($L_{imp}=100$), where the impulse noise starts from the 400th sample in the turbo code block. The simulation results are presented in Table 5-1 and Figure 5-5.

An impulse will cause a consecutive unreliable sequence of systematic and parity bits in the first turbo decoder. The systematic information during the impulse noise duration (provided by the received systematic symbols) and the channel coding information of the first CC (provided by the received parity symbols generated from the first CC) are therefore highly unreliable. The channel coding information of the second CC, however, is preserved since the parity bits from the second CC are generated according to an interleaved version of the systematic sequence. Therefore, they are not all hit by impulse samples. These channel coding information are then employed to recover the information bits. However, since the systematic symbols and parity symbols from the first CC are very unreliable, the decoding error probability of these information bits is large. Adding an additional block interleaver to the whole turbo block could efficiently spread the impulse samples over the turbo block evenly instead of consecutively. The systematic or parity bits hit by the impulse samples still have low reliabilities. However, since the adjacent systematic or parity bits are no longer affected by impulse samples, they could have high reliabilities. These high reliable symbols can be used by the decoder to recover the less reliable symbols. However, the opposite is also true. When the impulse samples are highly unreliable, i.e., have wrong reliabilities with large amplitudes, they could degrade the decoding performance of the adjacent reliable symbols. These effects will be shown in the following simulation results.

The performances of the system with an impulse noise of length 50 and 100 samples are presented in Table 5-1 and Figure 5-5. The performance of adding the additional interleaver to separate the impulse samples is also shown.

It is observed in Figure 5-5 that both impulse sample variance and impulse length have a significant effect in the performance of turbo coded BPSK transmissions. Spreading the impulse over the whole turbo code block improves the performance when the impulse noise has a low to moderate variance. When σ_i^2 is high, more than 18 in this

case, spreading the impulse samples results in a worse performance. For shorter impulses, spreading gives much better performance.

Table 5-1 – BER of impulse noise corrupted binary turbo code

Rate-1/3 turbo code, $K=1024$, BPSK transmission, $N_0=1.0$, consecutive impulse noise starts at the 400th sample in each turbo block, block interleaver separates two adjacent impulse samples by 10 samples, MAP decoding, 10 iterations

σ_i^2	5	7	10	15	20	25
50 samples	0	0.000134	0.00045	0.00242	0.0115	0.0123
100 samples	0.000137	0.0019	0.00645	0.02328	0.029	0.039
100 samples with an additional interleaver	0	$3.55 \cdot 10^{-5}$	0.00079	0.01417	0.0398	0.0595

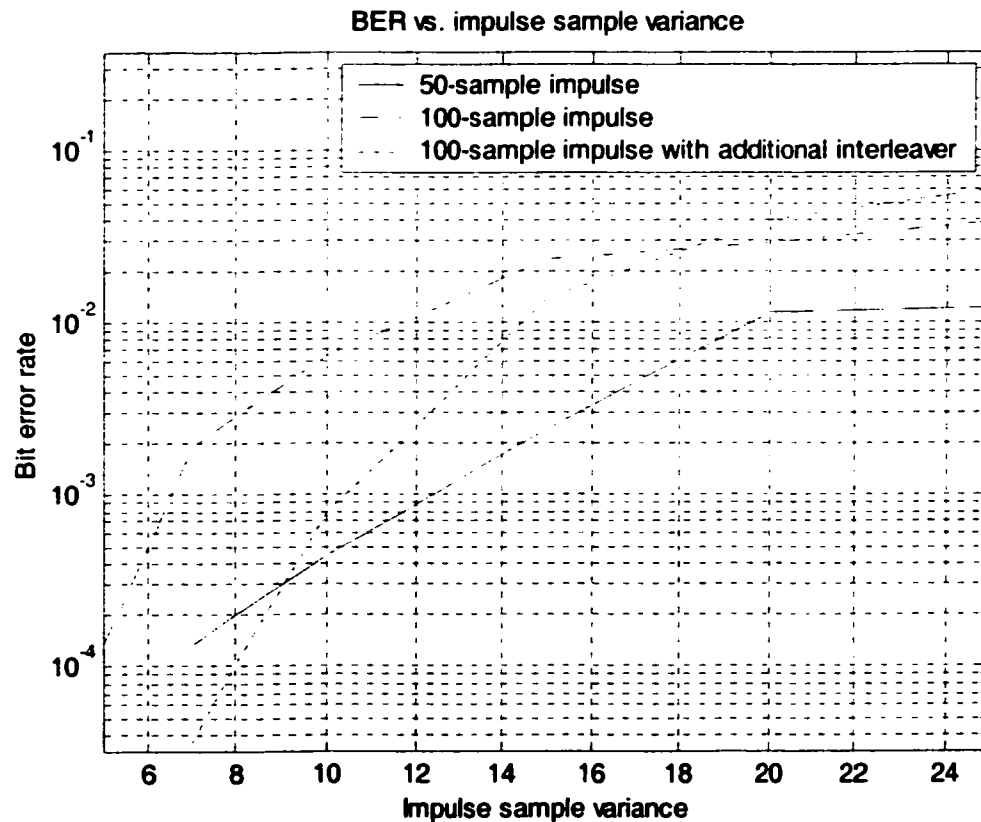


Figure 5-5– Performance of turbo codes against impulse noise variance

Rate-1/3 turbo code, $K=1024$, BPSK transmission, $N_0=1.0$, consecutive impulse noise starts at the 400th sample in each turbo block, block interleaver separates two adjacent impulse samples by 10, MAP decoding, 10 iterations

In Table 5-2, the BER performance of a turbo code against impulse noise with an impulse sample variance of 10, versus impulse starting positions, S_i , in one turbo block, are shown.

Table 5-2 – BER performance with impulse noise starting from different positions

Rate-1/3 turbo code, $K=1024$, BPSK transmission, $N_0=1.0$, $N_i=20$,
impulse length of 50 samples, MAP decoding, 10 iterations, S_i is the
starting position of the impulse noise

S_i	1	101	201	301	401	501	601	701	801	901
BER	0.003	0.007	0.01	0.007	0.011	0.008	0.006	0.011	0.005	0.011

The performance is the average over all possible positions. However, it is shown in Table 5-2 that impulses starting from different positions give performances that are not far from each other.

Investigation of turbo codes against impulse noise was performed in the environment of AWGN plus impulse noise. The degradation resulting from impulse noise lies in the fact that impulse noise samples cause an error path in the trellis. Therefore, the larger the background noise is, the more likely that an impulse will cause a decision on an error path. Effects of background noise variance on the impulse noise performance of turbo codes are shown in Table 5-3 and Figure 5-6.

Table 5-3 – BER performance of turbo code against impulse noise vs. background noise

Rate-1/3 turbo code, $K=1024$, BPSK transmission, $N_0=1.0$, impulse
length of 100 samples, impulse starting from the 400th samples, MAP
decoding, 10 iterations

γ_b (dB)		0.9794	1.7712	2.7403	3.9897	5.7506
BER	$N_i=10$	0.019	0.00784	0.00623	0.00464	0.004
	$N_i=20$	0.0538	0.034	0.0222	0.018	0.017

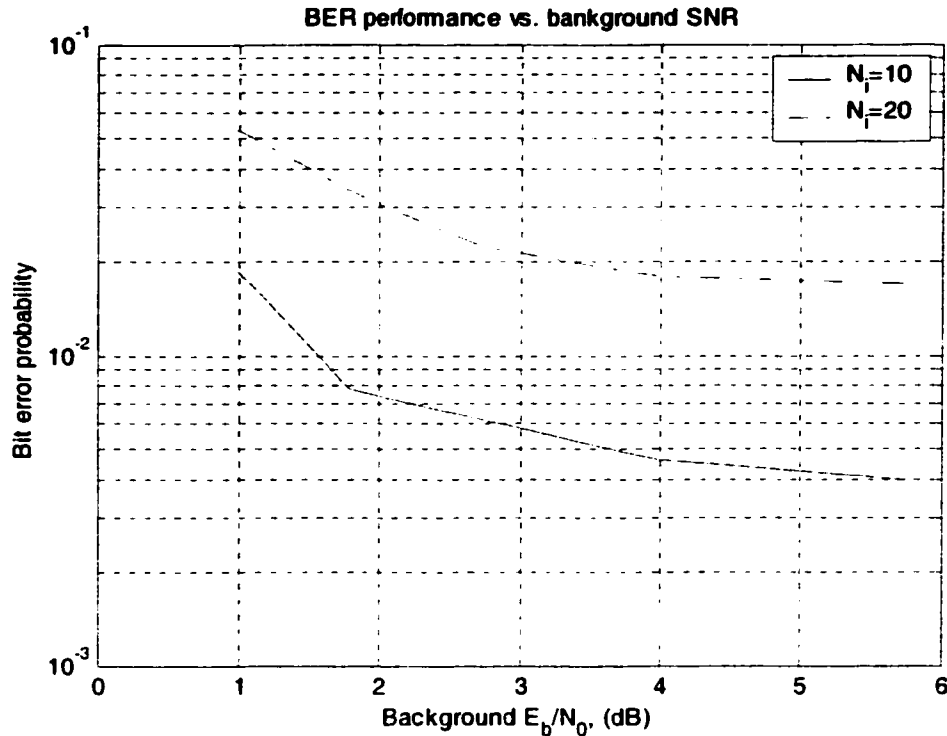


Figure 5-6 – Turbo code performance in the presence of impulses vs. background SNR

Rate-1/3 turbo code, $K=1024$, BPSK transmission, $N_0=1.0$, impulse length of 100 samples, impulse starting from the 400th samples in one turbo block, MAP decoding, 10 iterations

5.4.1.2. Discussions on turbo codes in impulse noise environment

1) Sensitivity to the impulse length

By simulations, it was shown that the impulse length has a significant effect on the performance of turbo decoding. Applying an additional block interleaver over the whole turbo code block could provide better performance against impulses of short lengths. For long impulses, the opposite effect was observed.

2) Sensitivity to the impulse noise variance

It was shown in Figure 5-5 that, with large impulse noise variances, even short impulse noise can be disastrous to turbo codes with block lengths 10 to 20 times longer than the impulse length. At low to moderate impulse noise variances, the performance in the presence of short impulses is much better than that with long

impulses; while with large impulse noise variances, the difference becomes less significant.

3) Sensitivity to the position of impulse noise in one turbo block

From the simulation, we found that the impulses starting from different positions do not result in significantly different performances. Therefore, from this simple simulation, we conclude that the position of the impulse noise doesn't make a major difference.

4) Effects of the background noise

The significance of turbo codes lies in the excellent performance at low SNR values. For binary turbo decoding, the BER vs. SNR is a "water-fall" curve before BER of 10^{-6} and switches to a much flatter BER floor in the lower BER region. In the "water-fall" region, a slightly higher noise variance will significantly degrade the BER performance. Therefore, it is expected that the impulse noise will significantly degrade the performance of a turbo-coded system operating in the "water-fall" region. However, the performance penalty caused by impulse noise to turbo-coded system in the "BER floor" region will be much smaller. This means that the price to pay for the turbo code to combat impulse noise is some coding gain performance degradation against the stationary background noise. The effect of background noise is shown later in Table 5-5. It will be shown that by increasing the signal-to-background-noise ratio slightly, the performance of the turbo-coded system against impulse noise can be significantly improved when erasure turbo decoding is employed.

5) LLR patterns and error patterns caused by the impulse noise

The output of a turbo decoder is the soft information of each information bit, followed by a decision device which produces a '1' for positive LLRs and a '0' for negative LLRs. The LLRs depend on the transmitted symbols as well as the. Usually, it is hard to predict the output LLRs if the decoding procedure doesn't converge to the correct path in the trellis, because the LLRs have almost a Gaussian distribution. The output error pattern will be error bits spread all over the block because of the interleaver. However, when a decoding error is caused by an impulse noise, it will be shown that the LLRs no longer follow the same distributions.

5.4.2. Erasure turbo decoding against impulse noise

Erasure techniques can be employed to combat burst errors in RS coded transmission systems, as shown in [49], because the erasure correction capability of an RS code is twice that of its error correction capability. In this section, we investigate the effect of employing erasure techniques in turbo-coded systems against impulse noise.

Locating the erasures is the key in employing erasure-decoding techniques. Usually the receiver tries to find the highly unreliable symbols in the received symbol sequence and identifies them as erasures. The receiver interprets these erasures as null symbols (zeros) in the channel decoding procedure. However, accurately locating the unreliable symbols is not a simple task. Furthermore, reliable symbols marked as erasures can also significantly degrade the performance of the turbo code against background noise.

There are two situations we might encounter in a practical transmission system that experience burst errors, with channel state information (CSI) or without channel state information (NCSI). The CSI provides the instantaneous state of a time varying channel, including channel impulse response and the channel noise condition. Apparently, CSI gives the receiver more reliability information of the received symbols, which enables a better detection performance. On the other hand, in the case of NCSI, the receiver has to estimate the channel states based on the received symbols. In this situation, the erasure locations are decided by the estimated channel state and this is called blind erasure.

In a BPSK transmission, because impulse noise usually has a much larger amplitude than the background AWGN noise and the received signal, blind erasure is likely to locate the impulse sample locations accurately. For implementation simplicity, in a BPSK transmission, the received samples with amplitudes larger than a pre-defined threshold, V_{th} , are considered as erasures. Simulation results of employing erasure turbo decoding techniques are shown in Table 5-4.

It is observed that using blind-erasure gives much better performance when the impulse noise variance is high, i.e., where the reliability of identifying the erasure locations by monitoring the received signal amplitude is accurate. For low impulse noise variances, using blind erasure gives little or no performance improvement. However, erasure decoding with exact CSI gives significant performance improvement.

Table 5-4 – BER of turbo decoding with and without erasure

Rate-1/3 turbo code, $K=1024$, BPSK transmission, $N_0=1.0$, impulse length of 100 samples, impulse starting from the 400th samples in one turbo block, erasure detection threshold $E_{th}=4$, MAP decoding, 10 iterations

Variance	5	10	15	20	25
No erasure	0.000137	0.006452	0.02328	0.029	0.039
Blind erasure	0.000124	0.000931	0.002884	0.005612	0.006406
Erasure with CSI	0 (after 1000 blocks simulation) $\leq 10^{-4}$				

Simulation results of using erasure turbo decoding with CSI for impulses of different lengths at different background noise levels are shown in Table 5-5. It is shown that the BER increases roughly 10 times when the impulse length is doubled.

Table 5-5 – BER of turbo decoding with (w.) and without (w/o) erasure

Rate-1/3 turbo code, $K=1024$, MAP decoding, 10 iterations, erasure decoding with CSI, BPSK transmission

L_{imp} (samples)		150	200	400	800
$N_0=1.2$	w/o int.	0.000417	0.0012	0.0179	0.1286
	w. int.	0.001	0.00456	0.0635	0.21
$N_0=1.0$	w/o int.	0	0.00079	0.05165	0.04695
	w. int.	0	0	0	0.1217

An additional interleaver is applied to the overall encoded bits to see if any performance improvement can be achieved when erasure turbo decoding is performed. The motivation of adding this interleaver is to spread a long erasure sequence over the turbo block, thus creating many short erasure sequences separated by non-erased segments. When these erasure sequences are short enough, they can be corrected with the help of the adjacent non-erased symbols. However, in very noisy channel conditions, i.e., at the margin of the ‘waterfall’ region of the turbo code performance, this could make the performance worse because the adjacent bits are also not very reliable. However, if an additional 1 dB margin is given, the performance improvement achieved by adding this interleaver is obvious. The performances of using erasure turbo decoding against impulse noise with (w. int.) or without (w/o int.) the additional interleaver are shown in Table 5-5.

Similar simulations were performed with a rate-1/2 turbo code obtained from puncturing the rate-1/3 turbo code. The simulation results are shown in Table 5-6.

Table 5-6 – BER performance of erasure turbo decoding with CSI

Rate-1/2 turbo code, $K=1024$, MAP decoding, 10 iterations, BPSK transmission

Impulse length		100	200	300	400
N_0 is 0.75	w/o int.	0.011	0.0438	0.0923	0.135
	w. int.	0.024	0.0886	0.1426	0.1736
N_0 is 0.7	w/o int.	0.00132	0.013	0.0445	0.096
	w. int.	0.00295	0.034247	0.1136	0.155
N_0 is 0.6	w/o int.	0	0.00536	0.0129	0.029
	w. int.	0	0	0.0117	0.087

5.4.3. LLR patterns

In this section, we look into the LLR patterns output from an error turbo decoding caused by impulse noise. In a successful decoding, the reliability outputs from the turbo decoder for BER around 10^{-5} are usually very large. Typical LLRs resulted from a successful turbo decoding is plotted in Figure 5-7. The absolute values of all the LLRs are around 47, which is the maximum limit we set in the simulation. Almost all the LLRs have this high value of reliability. The output of the same turbo decoder for a decoding error caused by an impulse noise, starting from the 401th sample and ending at the 800th sample, with a large variance ($\sigma_i^2=20$), is shown in Figure 5-8.

The two short vertical lines at 133 and 267 on the x -axis in Figure 5-8 identify the starting position and ending position of the impulse. It is apparent that the LLR values are relatively small compared to those generated from a successful decoding. In particular, the LLR values within the impulse noise duration have a high probability of getting very large absolute values, i.e., a large variance. It is therefore fairly easy to identify the range of the impulse noise. However, the large difference between the LLR values within the impulse duration and LLR values outside the impulse duration disappears when the variance of impulse samples becomes smaller. Figure 5-9 shows the *LLR* pattern for the same binary turbo decoder containing an impulse noise with a variance of 5. The impulse range is no longer so obvious as that shown in Figure 5-8.

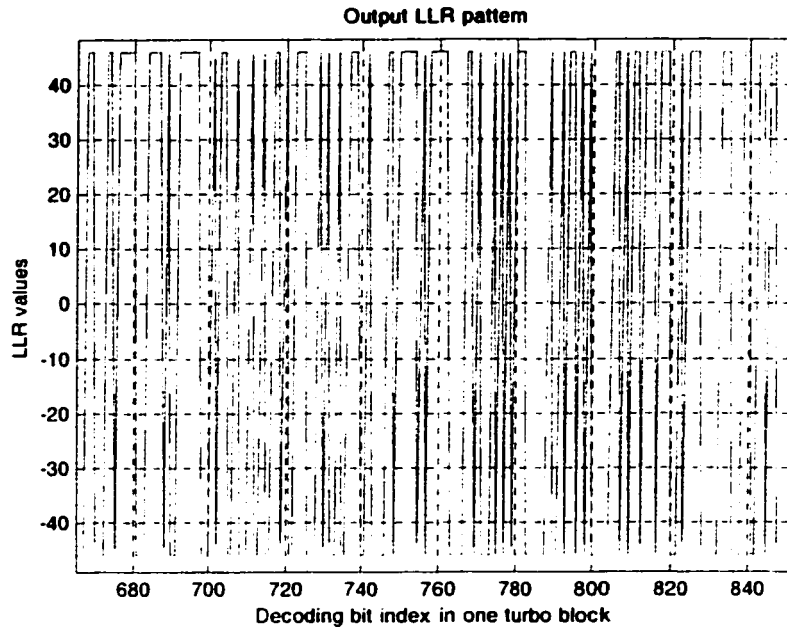


Figure 5-7 – LLR pattern resulted from a successful turbo decoding

Rate-1/3 turbo code, $K=1024$, BPSK transmission, MAP decoding, 10 iterations

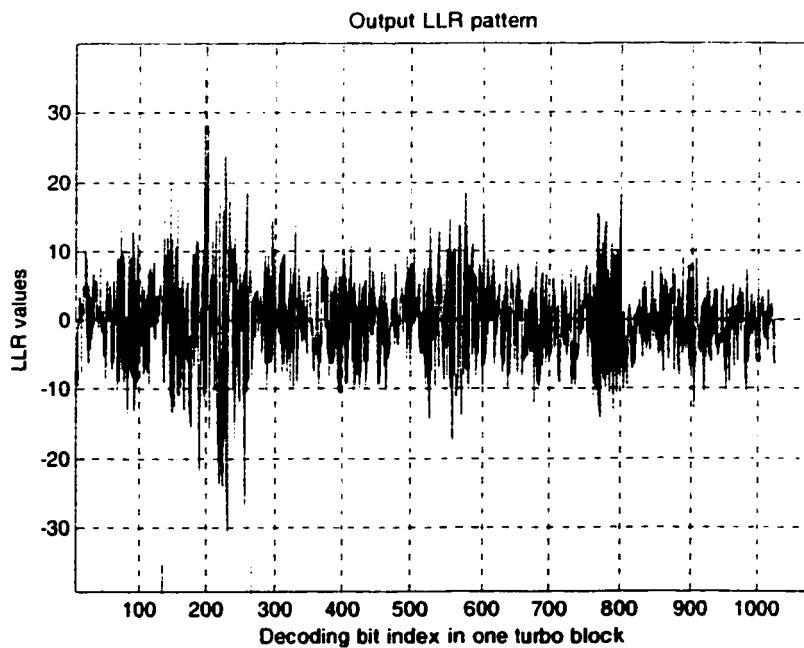


Figure 5-8 – Typical LLR pattern resulted from an error turbo decoding

Rate-1/3 turbo code, $K=1024$, BPSK transmission, impulse length of 400 samples, impulse starting from the 401th sample and ending at 800th sample, $N_0=1.0$, $N_t=20$, MAP decoding, 10 iterations.

Nevertheless, we can still observe that there is an impulse noise in the current turbo block and an error event happened in the turbo decoding. Backup methods can then be employed in communication systems. A retransmission might be performed, or else, an approximate impulse range can be estimated and the received symbols in this range can be declared as erasures. Consequently, another turbo decoding can be applied to the symbol block with erasures, which, as shown in section 5.4.2, can give a much better performance.

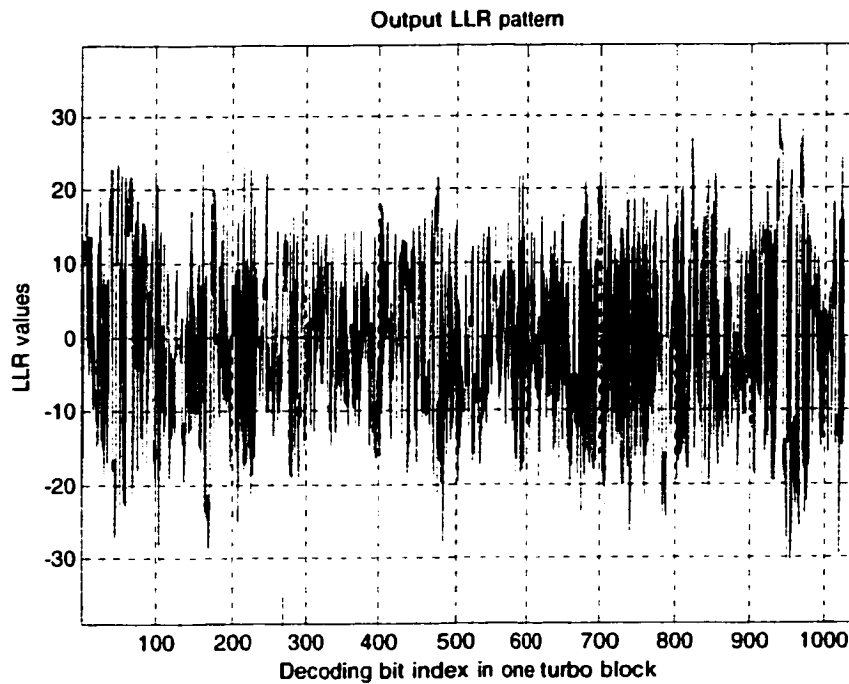


Figure 5-9 – Typical LLR pattern resulted from an error turbo decoding

Rate-1/3 turbo code, $K=1024$, BPSK transmission, impulse length of 400 samples, impulse starting from the 400th sample and ending at 800th sample, $N_0=1.0$, $N_i=5$, MAP decoding, 10 iterations.

5.4.4. Error pattern from binary turbo decoder using erasure turbo decoding

In this section, another advantage of using erasure turbo decoding is introduced. It is observed that erasure turbo decoding generates a short error pattern when the erasure sequence is not very long. For a rate-1/3 turbo code, the LLR pattern and error pattern

given by the turbo decoder using erasure decoding for a turbo block with an impulse noise of 400-bit long are shown in Figure 5-10.

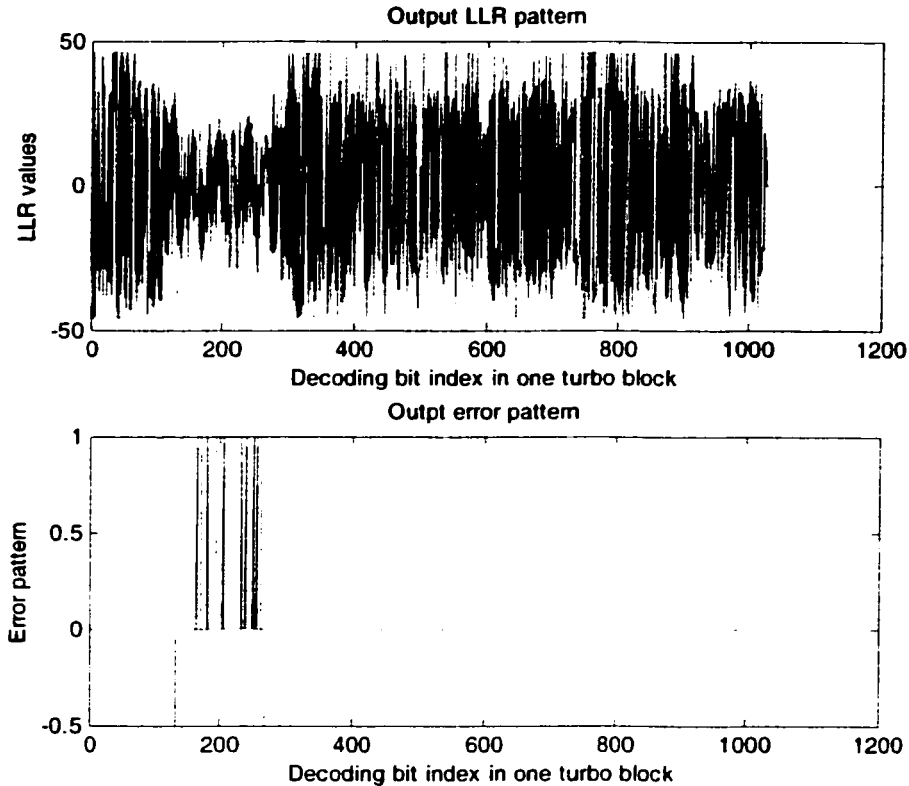


Figure 5-10 – Typical LLR pattern and error pattern resulted from erasure turbo decoding
 Rate-1/3 turbo code, $K=1024$, BPSK transmission, impulse length of 400 samples, impulse starting from the 400th sample and ending at 800th sample, $N_0=1.0$, erasure turbo decoding with CSI, MAP decoding, 10 iterations.

Apparently, the LLR values within the erasure area are relatively small compared to the non-erasure section. Furthermore, the decoding errors occur mainly inside the erasure area, which results in a much shorter error burst.

With this error pattern, if an outer RS code is used to correct the burst error output from the turbo decoder, the requirements on the outer RS code and the associated interleave depth is much smaller.

This advantage disappears for longer erasures. It is shown by simulation that an erasure longer than 800-bit results in an error pattern randomly distributed all over the

turbo code block. It is actually an error burst of a length equal to the length of the turbo block.

Similar simulations over rate-1/2 turbo codes give the results shown in Figure 5-11. This time, however, to keep a low probability of long error burst, the erasure length has to be less than 200-bit, which is only half the length for rate-1/3 turbo code.

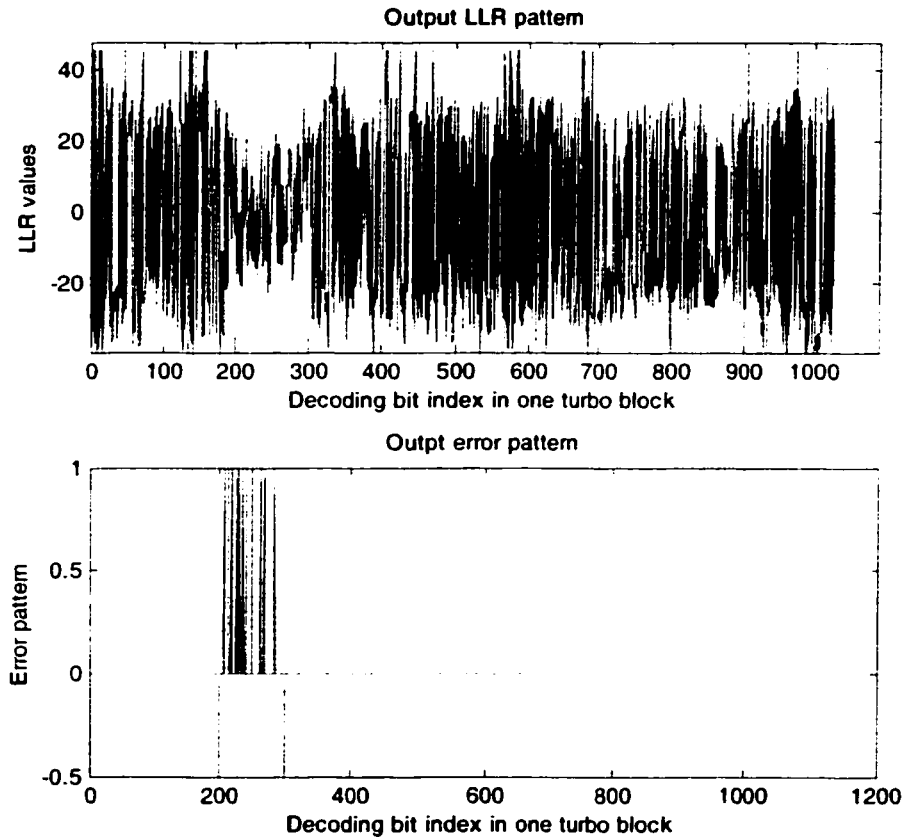


Figure 5-11 – Typical LLR pattern and error pattern of erasure turbo decoding $L_{imp}=200$
 Rate-1/2 turbo code, $K=1024$, BPSK transmission, impulse length of 200 samples, impulse starting from the 401th sample and ending at 600th sample, $N_0=0.75$, erasure turbo decoding with CSI, MAP decoding, 10 iterations

To give a clearer picture of the output error pattern, the number of error bytes in each error-decoded turbo code word is obtained from the simulations and presented in Figure 5-12. It is apparent that most of the erroneously decoded turbo blocks have less than 10

error bytes, while only about 1/10 have more than 10 error bytes and few blocks have more than 30 bytes in error. Compared to the cases without erasure, where on average 90 bytes are in error in a turbo block, the number of output bytes is reduced significantly. An RS code with a smaller interleave depth or smaller redundancy could be used to decrease either the system delay or the bandwidth redundancy.

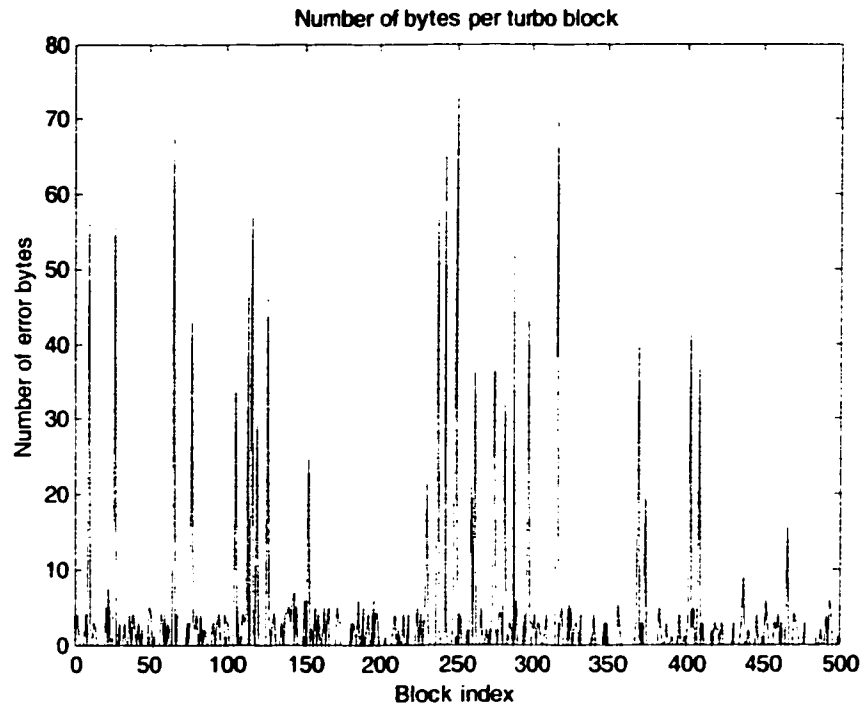


Figure 5-12 – Number of error bytes resulted from error turbo decoding

Rate-1/2 turbo code, $K=1024$, BPSK transmission, impulse length of 150 samples, $N_0=0.70$, erasure turbo decoding with CSI, MAP decoding, 10 iterations

The probability density function of the number of bytes in error is calculated and shown in Figure 5-13. Apparently, erasure turbo decoding gives less than 10 error bytes in most of the cases.

This advantage will disappear when the length of the erasure sequence increases. Figure 5-14 gives the probability distribution of error byte number for an erasure of length 300-bit.

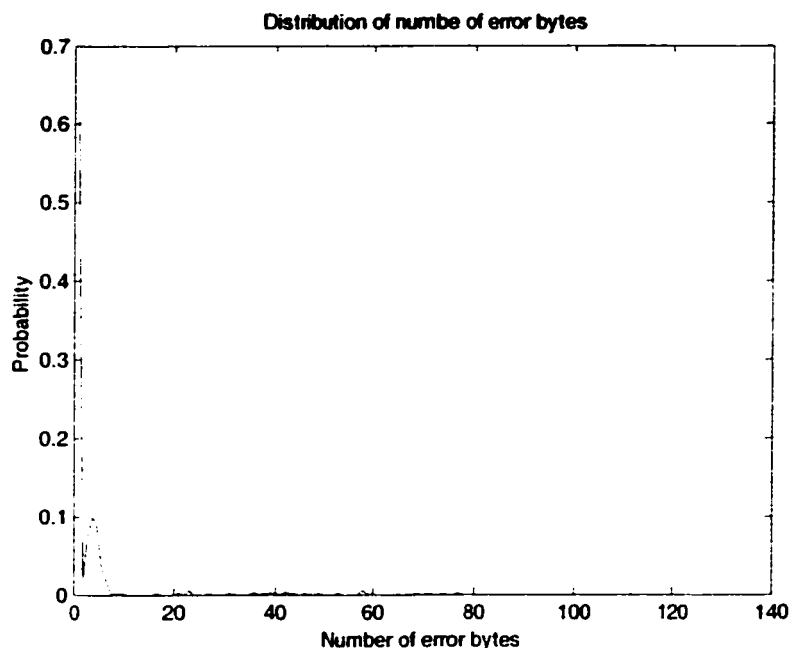


Figure 5-13 – Probability distribution of the number of error bytes per turbo block

Rate-1/2 turbo code, $K=1024$, BPSK transmission, impulse length of 150 samples, $N_0=0.7$, erasure turbo decoding with CSI, MAP decoding, 10 iterations

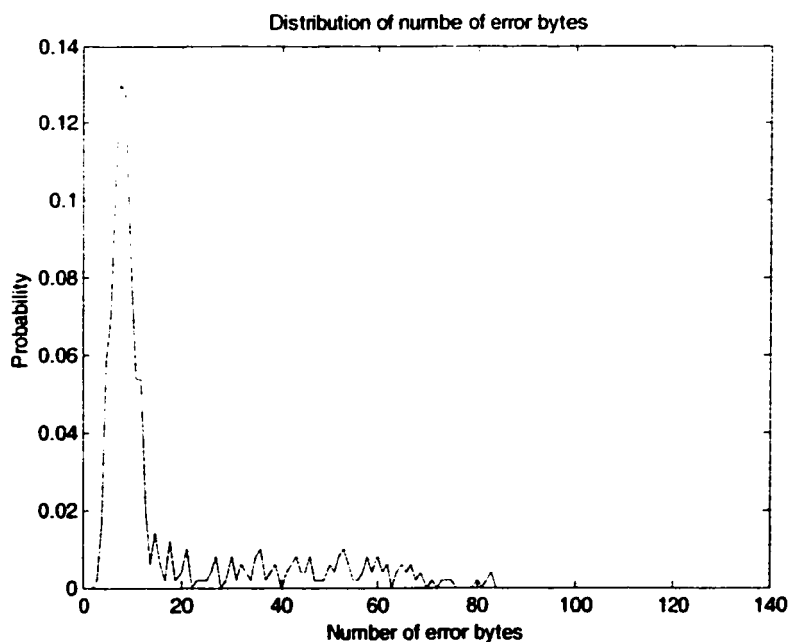


Figure 5-14 – Probability distribution of error byte number per turbo block

Rate-1/2 turbo code, $K=1024$, BPSK transmission, impulse length of 300 samples, $N_0=0.75$, erasure turbo decoding with CSI, MAP decoding, 10 iterations

The background noise plays an important role here. Simulation results showed that the smaller the background noise is, the more likely the error sequence will be limited within the impulse duration. However, this shorter error sequence is achieved at the cost of an increase in the signal-to-background noise ratio.

5.5. IMPULSE NOISE IN HIGH SPEED QAM TRANSMISSION WITH TTCM

In this section, we present the simulation results of QAM transmissions with TTCM, which can be applied to estimate the performance of turbo coded ADSL DMT systems. The impulse noise model employed in this section is the generalized Gaussian burst model. The simulation results presented below can therefore be used to evaluate the effect of impulse noise on TTCM systems in various applications, with impulses having various characteristics.

We will focus on the error performance of the turbo coded bits in the TTCM structure, since uncoded bits have a much better protection against impulse noise compared to the turbo coded bits. Further protection can be provided by applying an additional high-rate RS code to the uncoded bits. Since impulses cause relatively short error bursts on the uncoded bits if the turbo-coded bits are decoded correctly, there should be small burst error correction capability requirement on this RS code. In a multicarrier system, since the receiver transforms the short high-peak impulse into a long low-peak impulse, the protection provided by the constellation mapping will hopefully be enough to combat the impulse noise.

Since impulse noise comes in addition to the background AWGN noise, our simulations of turbo-coded systems are performed with a reasonable background AWGN variance. This AWGN variance guarantees a BER of 10^{-5} when there is no impulse noise.

Table 5-7 – Default simulation assumptions

Channel	AWGN channel, no ISI
Turbo code	Rate-1/2 turbo code
Turbo code interleaver	1024-bit
Decoding algorithm	Binary MAP, 10 iterations
Modulation	64QAM, 4 information bits per symbol
Impulse noise model	Burst Gaussian model

5.5.1. Performance of TTCM in impulse noise environment

Performance of TTCM in an impulse noise environment with different impulse noise variances and different impulse lengths are shown in Figure 5-15.

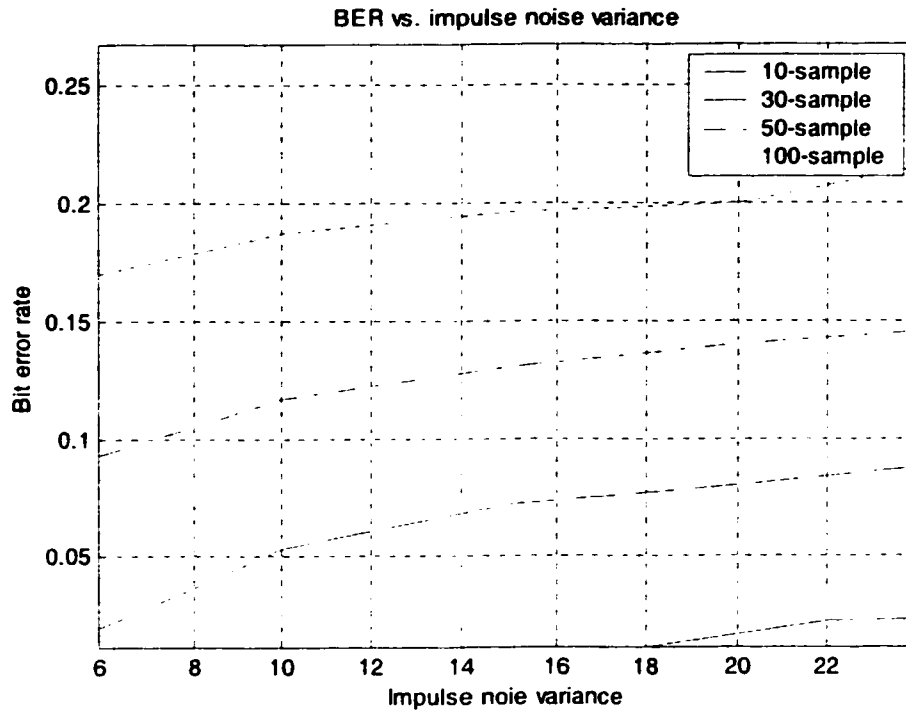


Figure 5-15 – BER performance of 64QAM TTCM in impulse noise environments

Rate-1/2 turbo code, $K=1024$, 64QAM transmission, 4 information bits per symbol, $N_0=0.75$, binary MAP decoding, 10 iterations, turbo-coded bits only.

It is observed that impulses, even the very short ones, cause devastating effect to the TTCM decoding. The impulse length has a significant effect on the turbo decoding results. The impulse noise variance also has a major effect on the performance. The background noise variance of 0.75 in the simulation is on the edge of providing a satisfactory performance of 10^{-5} for the turbo code without impulse noise. This gives a γ_b value close to the end of “water-fall” region of the turbo code, which is the desired high-coding gain area.

Three methods can be employed to improve the performance of TTCM against impulse noise: using an additional overall interleaving, performing erasure turbo decoding and adding an outer RS code.

5.5.2. Performance improvement by employing an additional interleaver

Employing an additional interleaver to spread the consecutive impulse samples over one turbo code word could possibly improve the performance, since an error path is more easily chosen by the decoder if the received code word contains a sequence of consecutive unreliable symbols. If the unreliable symbols are apart from each other, the reliable symbols and the coding structure could possibly make up for the effect of the unreliable symbols. In [13], an interleaver is used at the output of the turbo encoder to improve the performance of bandwidth efficient turbo coding in fading channels. Since such an interleaver is used to separate the impulse samples as far from each other as possible, a regular interleaver (block interleaver, convolutional interleaver) is sufficient. Table 5-8 shows the BER performance of 64QAM with TTCM using an additional block interleaver. Performance improvement could only be obtained for very short impulses. When the impulse is moderate or long, adding this interleaver will only degrade the performance. The cost of the performance improvement (for short impulses) is an additional delay resulted from the additional interleaving and deinterleaving.

Table 5-8 – Effect of adding an additional interleaver in 64QAM TTCM

Rate-1/2 turbo code, $K=1024$, 64QAM transmission, 4 information bits per symbol, $N_0=0.75$, MAP decoding, 10 iterations, turbo-coded bits only

σ_i^2		5	10	15	20
L_{imp}					
10	w/o int	$2 \cdot 10^{-4}$	10^{-3}	$6 \cdot 10^{-3}$	$1.1 \cdot 10^{-2}$
	w. int.	$6.4 \cdot 10^{-5}$	10^{-3}	$4.5 \cdot 10^{-3}$	$1.3 \cdot 10^{-2}$
30	w/o int	0.016	0.0525	0.0714	0.0811
	w. int.	0.03	0.07	0.1	0.11
50	w/o int	0.086	0.1177	0.1289	0.1392
	w. int.	0.11	0.13	0.14	0.16

5.5.3. Utilizing erasure turbo decoding

It was shown in previous sections that employing erasure turbo decoding could improve the performance of a turbo coded BPSK transmission. Since our proposed TTCM with simplified decoding utilizes a binary turbo decoder, this technique can be applied to the TTCM directly to obtain the same performance improvement.

5.5.3.1. Performance improvement by utilizing erasure turbo decoding

Since impulse noise samples have a very large noise variance, they might provide negative information to the turbo code decoder. However, if the receiver knows exactly which samples containing the impulse noise, instead of trying to perform the turbo decoding based on these very unreliable symbols, it could mark them as erasures (zeros that provide no information).

In multilevel transmissions with TTCM, each channel symbol usually contains several turbo-coded bits. Therefore, a short impulse could result in a long erasure bit sequence. In the proposed QAM transmissions with TTCM, each QAM symbol contains at most four coded bits. The actual erasure bit sequence is therefore shorter than four times the impulse length in samples. Simulation results of turbo coded QAM transmissions with erasure turbo decoding are shown in Table 5-9.

Table 5-9 – Performance of 64QAM TTCM with erasure decoding against impulse noise
Rate-1/2 turbo code, $K=1024$. 64QAM transmission, 4 information bits per symbol, erasure turbo decoding with CSI. $N_0=0.75$, MAP decoding, 10 iterations, turbo-coded bits only.

σ_i^2		5	10	15	20	25
BER No erasure	100	0.17	0.188	0.197	0.2	0.22
	50	0.086	0.1177	0.1289	0.1392	0.1443
	30	0.016	0.0525	0.0714	0.0811	0.0889
BER Erasure with CSI	100	0.0437				
	50	$5.67 \cdot 10^{-3}$				
	30	$7.6 \cdot 10^{-3}$				

Two observations are made from Table 5-9. First, impulses as short as 30 samples result in BERs higher than 10^{-2} . Second, erasure turbo decoding with CSI produces much better performance. However, perfect CSI in impulse noise environment is very hard to obtain, especially for QAM transmissions with high peak-to-average power ratio.

Again, a block interleaver can be applied for further performance improvement when the impulse is short. If there are too many erasures, adding this interleaver will degrade the performance because of the large multiplicities of mid-weight error words in the turbo code. In Table 5-10, simulation results on the effect of adding this additional interleaver under different background noise variances are shown.

Table 5-10 – Effect of adding an additional interleaver in 64QAM TTCM with erasure turbo decoding, under different background noise conditions

Rate-1/2 turbo code, $K=1024$, 64QAM transmission, 4 information bits per symbol, erasure turbo decoding with CSI, MAP decoding, 10 iterations, turbo-coded bits only

L_{imp}		25	50	75	100
N_0 is 0.75	w/o int.	$5 \cdot 10^{-6}$	$5.67 \cdot 10^{-3}$	$1.62 \cdot 10^{-2}$	$5.07 \cdot 10^{-2}$
	w. int.	$6.2 \cdot 10^{-4}$	$4 \cdot 10^{-3}$	$6 \cdot 10^{-2}$	0.1349
N_0 is 0.7	w/o int.	$3 \cdot 10^{-6}$	$5 \cdot 10^{-3}$	$1.18 \cdot 10^{-2}$	$2.73 \cdot 10^{-2}$
	w. int.	0 ¹⁷	$5.48 \cdot 10^{-4}$	$1.6 \cdot 10^{-2}$	0.105
N_0 is 0.6	w/o int.	0	$4.845 \cdot 10^{-3}$	$1.12 \cdot 10^{-2}$	$2.07 \cdot 10^{-2}$
	w. int.	0	0	$3.42 \cdot 10^{-3}$	$1.2 \cdot 10^{-2}$

It is observed that without the additional external interleaver, the background noise variance has little influence on the BER performance. However, the background noise variance plays an important role when the additional interleaver is employed. Improving the performance by adding this interleaver depends on recovering the lost information in the erasures by the trellis structure and the more reliable symbols corrupted only by background noise. The smaller is the background noise variance, the more reliable the unerased symbols are, and the easier it is to recover the erasures.

For large background noise, it is shown that the performance improvement with the additional interleaver is not significant. Degradation occurs for moderate impulse noise lengths of 75 samples or 100 samples. However, when the background noise is small, the performance improvement by adding this interleaver becomes significant. For example, with a background noise variance of 0.75, the performance improvement for impulse length of 50 samples is only marginal: from $5.67 \cdot 10^{-3}$ to $4 \cdot 10^{-3}$. For the same impulse

¹⁷ "0" means that no errors were detected in the simulation. However, it is only safe to say that the BER is lower than $5 \cdot 10^{-6}$.

length, for variance of 0.7, the performance is improved from $5 \cdot 10^{-3}$ to $5.5 \cdot 10^{-4}$. Further decreasing the background noise variance will give more performance improvements for short to moderate length impulses. However, decreasing the background noise variance is equivalent to increasing the transmission power. It is known that the coding gain of turbo code in BER floor region is not as good as in the water-fall region. Increasing the transmission power, i.e. the γ_b , will easily put the system out of the water-fall region. The coding gain of the TTCM against stationary noises will then be less efficient. Therefore, a compromise exists and a good operating point of SNR must be chosen to provide satisfactory performance against both stationary noise and impulse noise.

The BER performances of TTCM employing rate-1/3 turbo code are shown in Table 5-11. It is observed that for rate-1/3 turbo code, employing the overall interleaver can significantly improve the performance of the turbo decoding in the presence of impulse noise.

Table 5-11 – Performance of 64QAM TTCM using erasure turbo decoding

Rate-1/3 turbo code, $K=1024$, 64QAM transmission, on average 3.3
information bits per symbol, erasure turbo decoding with CSI, $N_0=1.0$.
MAP decoding, 10 iterations, turbo-coded bits only.

Impulse length	50	100	200
w/o int	$9.5 \cdot 10^{-3}$	$8.4 \cdot 10^{-3}$	$3.6 \cdot 10^{-2}$
w. int.	0	0	$3.1 \cdot 10^{-3}$

For rate-1/2 turbo codes, however, the performance is much worse for long erasure with length of 100 or 200. This is mainly because rate-1/2 turbo codes are obtained from punctured rate-1/3 turbo codes. Long erasures have a greater chance to make the received symbol sequence close to an erroneous code word. In the case of erasing 100 samples, systems employing rate-1/2 turbo codes showed much worse performance as expected. While even for 200-sample length erasure, TTCM with rate-1/3 turbo code can still achieve a much better BER by employing the external interleaver.

5.5.3.2. LLR distribution and output error pattern

In conventional turbo decoding with two SISO decoders, the first SISO decoder performs the decoding based on the received systematic symbols and the parity symbols

generated from the first CC. The second SISO decoder performs decoding based on the interleaved version of the received systematic symbols and the parity symbols from the second CC. There exists a consecutive set of branches in the decoding trellis for the first SISO where no information of both the systematic bits and parity bits is available. However, due to the interleaving of the systematic symbols before the second CC, there is little possibility that an interleaved systematic symbol and its corresponding parity symbol are erased at the same time.

Therefore, the first SISO decoder will provide unreliable information mainly for the bits contained in the erasure region. For the second SISO decoder, the information lost in the parity erasures is made up by the interleaved non-erased systematic symbols. The information lost in the erased systematic symbols is made up by the non-erased parity symbols. Furthermore, for non-erased systematic symbols, the corresponding parity symbols from the first CC are also non-erased. The *extrinsic* information output from the first SISO decoder helps to recover the information contained in the erased parity bits generated by the second CC. The shorter the erasure length, the better the SISO decoder can make up for the erasures.

In case of short erasures, the receiver has the information of non-erased systematic bits from the received systematic symbols, the parity symbols generated by the first CC and some non-erased parity bits generated by the second CC. For the erased systematic bits, the receiver only has the information from some non-erased parity symbols generated in the second CC. Therefore, the wrong decisions are most probably made on these erased systematic bits and result in a short burst error pattern.

When the erasure sequence is long, there will be some erased systematic symbols also falling into the parity symbol erasure region of the input sequence to the second decoder. The second decoder therefore has a similar weak decision region as that of the first SISO decoder. Because of the random interleaver between the two decoders, the resulted error words could have errors all over the turbo code word.

A typical output error pattern resulted from erasure turbo decoding is shown in Figure 5-16. This is at a γ_b of 8.45 dB (against background noise) at which the TTCM system in stationary noise environment achieves a BER of 10^{-5} . Impulse samples are located

between the two short vertical lines with negative values. It is noted that all of the errors fall into the impulse duration.

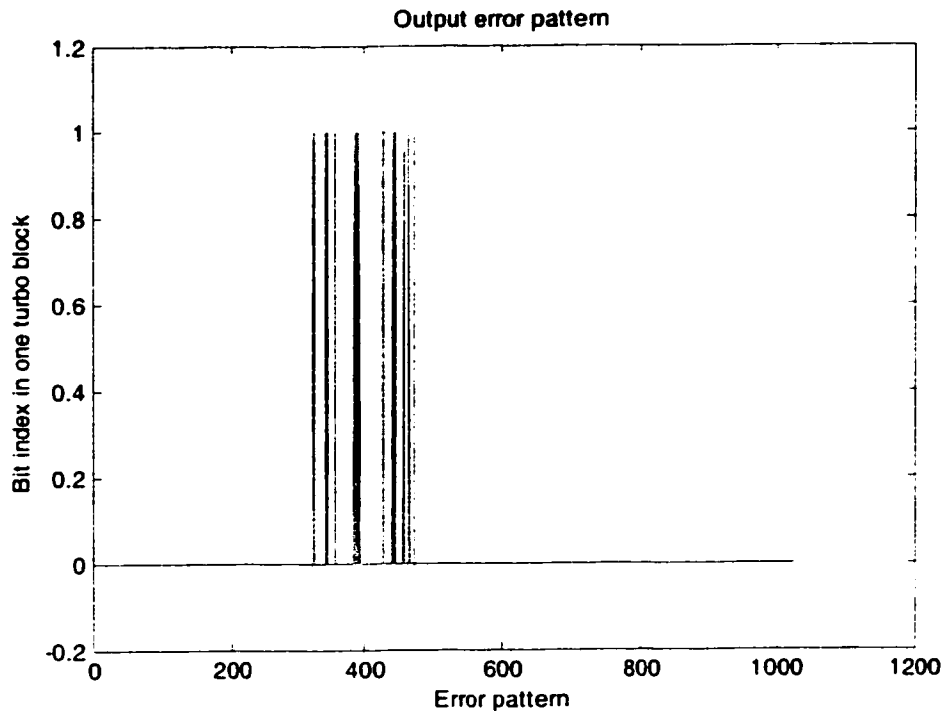


Figure 5-16 – Typical error pattern of QAM TTCM using erasure turbo decoding
 Rate-1/2 turbo code, $K=1024$, 64QAM transmission, 4 information bits per symbol, impulse of length 75-sample, erasure turbo decoding with CSI, $N_0=0.75$, MAP decoding, 10 iterations, turbo-coded bits only.

It is observed by simulation that more than 90% of the error-decoded blocks give less than 13 erroneous bytes concentrated in the impulse duration. Therefore, an RS outer code capable of correcting 13 erroneous bytes (with interleaving) will successfully remove 90% of the impulses. On the other hand, without the erasure decoding, the average number of byte errors caused by a decoding error is 80 bytes. It has been shown in Table 5-9 that employing erasure turbo decoding significantly improves the BER performance. The second advantage of erasure turbo decoding is that the much shorter error pattern is generated from a decoding error. This results in much fewer error bytes when there is a decoding error and consequently a much lower interleave depth requirement when a concatenated RS-TTCM coding structure is employed.

The statistics of the number of byte errors resulted from a decoding error with various values of background noise variance and an impulse noise of 75 samples are shown in Figure 5-17 and Figure 5-18.

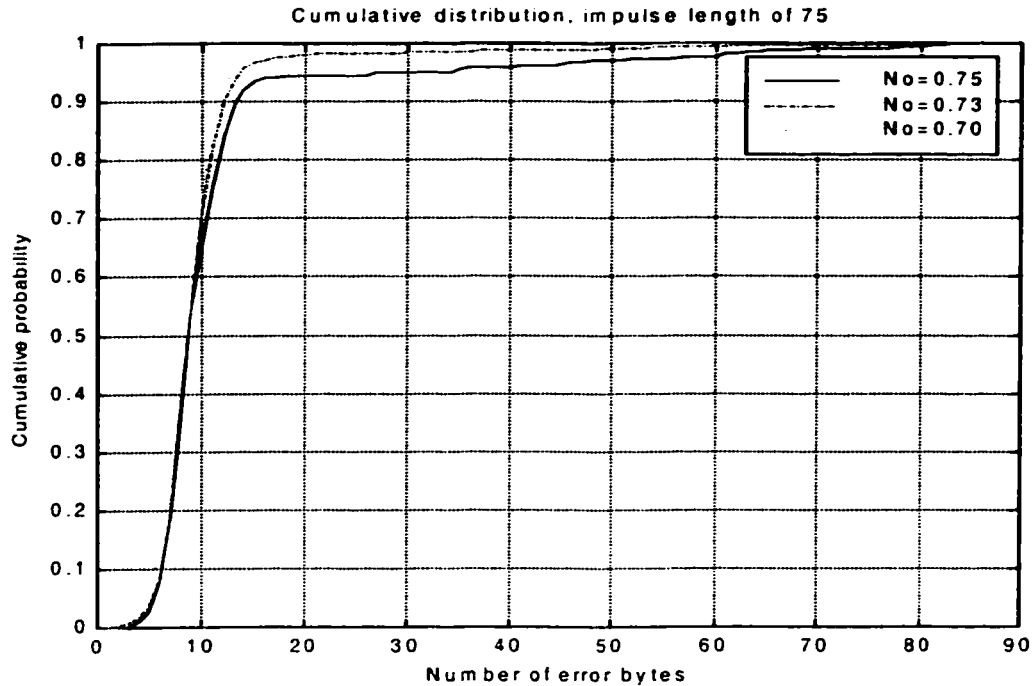


Figure 5-17 – Cumulative distribution of the number of error bytes in one turbo block

Rate-1/2 turbo code, $K=1024$, 64QAM transmission, 4 information bits per symbol, impulse of length 75-sample, erasure turbo decoding with CSI, $N_0=0.75$, MAP decoding, 10 iterations, turbo-coded bits only

Longer erasures will cause more byte errors in one turbo block. For example, only 70% of error blocks have less than 30 byte errors when an impulse of 100 samples occurs for the same turbo code.

Increasing the transmission signal power, however, can significantly help concentrate the error bits inside the impulse duration and therefore reduce the number of byte errors. This is easily explained by the fact that more reliable information symbols are received, which help to recover the erased information.

It is observed in Figure 5-17 that when the noise variance is decreased from 0.75 to 0.7, i.e. an SNR increase of 0.3 dB, more than 98% of the error blocks have less than 13 error bytes that are concentrated in the impulse noise duration. For impulses of 100

samples, this 0.3 dB decrease in noise variance results in more than 90% decoding errors containing less than 25 erroneous bytes concentrated in the impulse duration.

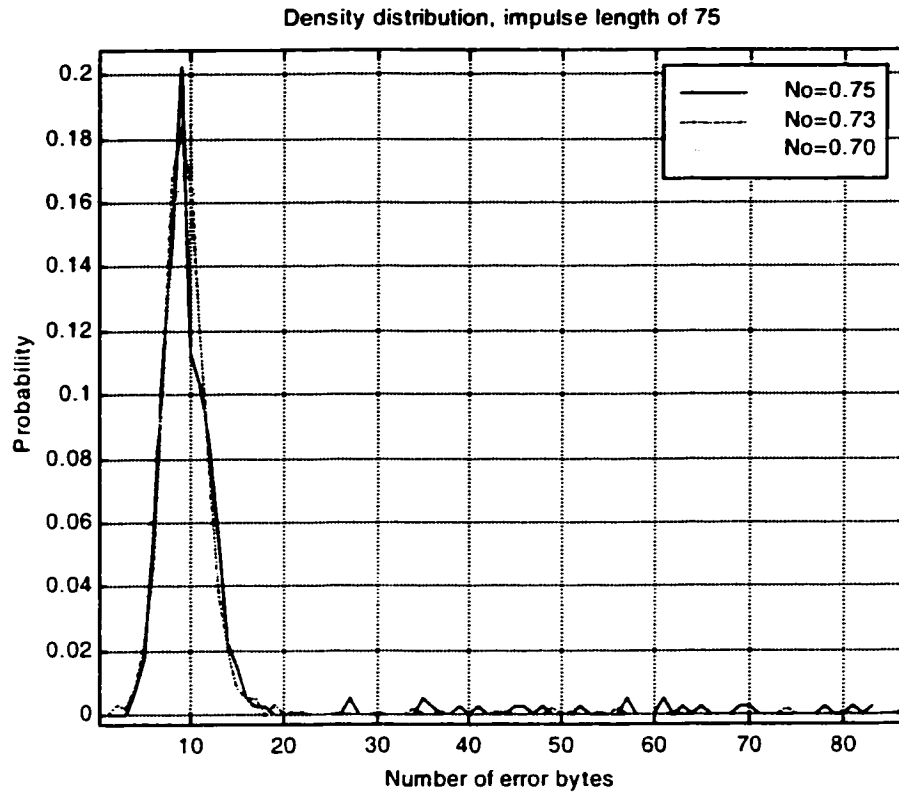


Figure 5-18 – Probability distribution of the number of error bytes in one turbo block

Rate-1/2 turbo code, $K=1024$, 64QAM transmission, 4 information bits per symbol, impulse of length 75-sample, erasure turbo decoding with CSI, $N_0=0.75$, MAP decoding, 10 iterations, turbo-coded bits only.

5.5.4. RS-TTCM concatenated coding

A conventional method to deal with impulse noise is to use an outer RS code to remove the error burst generated from the turbo decoder. RS codes in Galois field $GF(256)$ are used in our analysis since they are recommended in the ADSL standards. An $RS(N_{rs}, K_{rs})$ code in $GF(256)$ can correct t byte errors when $t \leq (N_{rs} - K_{rs})/2$, where K_{rs} is the number of information bytes in one RS code word and N_{rs} is the total block length in bytes. To combat long error bursts, usually a pair of interleaver and deinterleaver is employed between the RS encoder and decoder to efficiently reduce the number of byte

errors in each RS code word. A t -byte error correction RS code with an interleave depth of D can correct up to $t \cdot D$ erroneous bytes. The statistics of byte errors for QAM transmission using TTCM are listed in Table 5-12. The entries are in the format of “average number of byte errors / more than 90% have byte errors less than”

Table 5-12 – Number of error bytes per turbo block with and w/o erasures

Rate-1/2 turbo code, $K=1024$, 64QAM transmission, 4 information bits per symbol, $N_0=0.75$, $N_i=5$, MAP decoding, 10 iterations, turbo-coded bits only

N_0 L_{imp}		0.75	0.73	0.70
75	Non-erasure	79/88	77/89	76/88
	Erasure with CSI	12/13	11/12	10/12
100	Non-erasure	86/95	85/94	84/93
	Erasure with CSI	30/66	24/52 (25 for 80%)	19/23

It is observed in Table 5-12 that for impulses with length shorter than or equal to 75-sample, erasure turbo decoding generates short bursts of length 13 bytes. For conventional turbo decoding, there are about 80 bytes spreading all over the turbo block, which is actually a burst error of length 128 bytes. The interleave depth is therefore decreased by at least 9 times, which means 1/9 of the previous delay and 9 times lower memory requirement.

For impulse noise with length of 100-sample, it is shown that erasure turbo decoding will decrease the interleave depth by less than twice for a background SNR per bit (γ_b) of 8.45 dB. However, sacrificing a background γ_b by 0.3 dB (a noise variance of 0.7) can decrease the interleave depth by at least 4 times.

If the RS code operates on both coded bits and uncoded bits instead of coded bits only, erasure turbo decoding helps in a similar way since the uncoded bit during the impulse noise duration are also very likely to be in error.

Erasure RS decoding can be applied to the turbo coded QAM transmissions to further lower the required interleave depth. In the example shown in Table 5-12, without erasure turbo decoding, an error turbo decoding causes 128 error bytes. Therefore, an interleave depth $128/t$ is needed for correction of these errors, where t is the error correction

capability of the RS code. With erasure RS decoding, the interleave depth is reduced to $64/t$.

To apply erasure RS decoding, the receiver has to know if an turbo decoding error occurs. A decoding error can be detected quite easily since it usually generates small and non-convergent LLR values. Extensive simulations of the 64QAM transmission with TTCM, with and without erasure, showed no error event in which the resulted absolute LLR values have a mean larger than a quarter of the maximum value, which is $46/4=11.5$ in our simulation. These simulation results also showed that when a decoding error occurs, there are no more than 10 out of the 1024 output LLRs having absolute values larger than 10. On the other hand, for correct turbo decoding, the average absolute LLR values are always larger than 40 and there are always at least 900 LLRs having absolute value larger than 40. Therefore, both the average absolute LLR value and the number of LLRs larger than a certain threshold are good indications of error-decoding detection. Using these error-decoding indications, the receiver will erase the whole turbo block whenever a decoding error is detected. An erasure RS decoding could be performed subsequently.

5.5.5. Simulation results using the Markov impulse noise model

Simulations using two-state Markov impulse model were also performed and the following simulation results were shown in Table 5-13. The other assumptions are the same as those shown in Table 5-7.

For an ADSL DMT system with 256 subchannels of 4 kHz bandwidth, the sample rate in the channel is about $2 \cdot 10^6$ samples/second. With an assumption of one impulse every 10 seconds, p_a is calculated as $5 \cdot 10^{-8}$. Therefore a p_a of $5 \cdot 10^{-8}$ is employed in the simulation. The parameter p_b determines the average length of impulse noise. Since impulses are usually shorter than 100 μ s, which covers less than 200 samples in such an ADSL DMT transmission system, values of 0.005 are used for p_b in the simulations.

It is observed that the BER performance depends heavily on the value of p_b , which directly decides the average impulse length. A similar conclusion is that the impulse length plays the most important role in determining the performance.

Table 5-13 – BER of 64QAM TTCM with Markov channel model

Rate-1/2 turbo code, $K=1024$, 64QAM transmission, 4 information bits per symbol, $N_0=0.75$, MAP decoding, 10 iterations, turbo-coded bits only, Markov channel model, $p_a=5 \cdot 10^{-8}$

N_0	$N_i \backslash p_b$	5	10	15	20	25
0.75	0.005	$6.1 \cdot 10^{-6}$	$7.2 \cdot 10^{-6}$	$7.6 \cdot 10^{-6}$	$7.8 \cdot 10^{-6}$	$8.3 \cdot 10^{-6}$
	0.01	$3.1 \cdot 10^{-6}$	$4.0 \cdot 10^{-6}$	$4.3 \cdot 10^{-6}$	$5.0 \cdot 10^{-6}$	$5.3 \cdot 10^{-6}$
	0.02	$1.7 \cdot 10^{-6}$	$2.2 \cdot 10^{-6}$	$2.6 \cdot 10^{-6}$	$2.8 \cdot 10^{-6}$	$3.1 \cdot 10^{-6}$
	0.05	$4.9 \cdot 10^{-7}$	$8.1 \cdot 10^{-7}$	$1.0 \cdot 10^{-6}$	$1.2 \cdot 10^{-6}$	$1.4 \cdot 10^{-6}$
0.70	0.005	$5.4 \cdot 10^{-6}$	$6.5 \cdot 10^{-6}$	$5.8 \cdot 10^{-6}$	$7.6 \cdot 10^{-6}$	$8.2 \cdot 10^{-6}$
	0.01	$3.4 \cdot 10^{-6}$	$3.7 \cdot 10^{-6}$	$4.3 \cdot 10^{-6}$	$4.4 \cdot 10^{-6}$	$4.5 \cdot 10^{-6}$
	0.02	$1.7 \cdot 10^{-6}$	$2.1 \cdot 10^{-6}$	$2.2 \cdot 10^{-6}$	$2.5 \cdot 10^{-6}$	$2.8 \cdot 10^{-6}$
	0.05	$3.5 \cdot 10^{-7}$	$5.9 \cdot 10^{-7}$	$8.1 \cdot 10^{-7}$	$1.1 \cdot 10^{-6}$	$1.4 \cdot 10^{-6}$

With the two-state Markov channel model, simulation results showed similar BER results and output error patterns as shown in the last section, when erasure turbo decoding is applied. However, the distribution of the number of byte errors is different. It is observed that erasure turbo decoding significantly lowers the BER, 10 times for short impulses. For longer impulses, the improvement by employing erasure turbo decoding is not so significant. Since in Markov channel model, the length of the impulses have a flat distribution as shown in Figure 5-19, and are not limited, the overall performance improvement by using erasure turbo decoding is not that significant.

Table 5-14 – BER of 64QAM TTCM with Markov channel model

Rate-1/2 turbo code, $K=1024$, 64QAM transmission, 4 information bits per symbol, $N_0=0.75$, erasure turbo decoding with CSI, MAP decoding, 10 iterations, turbo-coded bits only, Markov channel model, $p_a=5 \cdot 10^{-8}$

N_0		0.005	0.01	0.02	0.05
0.75	w/o int	$4.8 \cdot 10^{-6}$	$2.1 \cdot 10^{-6}$	$5.6 \cdot 10^{-7}$	$4.6 \cdot 10^{-8}$
	w. int.	$5.0 \cdot 10^{-6}$	$2.7 \cdot 10^{-6}$	$9.4 \cdot 10^{-7}$	$8.1 \cdot 10^{-8}$
0.70	w/o int	$4.0 \cdot 10^{-6}$	$1.67 \cdot 10^{-6}$	$4.5 \cdot 10^{-7}$	$2.7 \cdot 10^{-8}$
	w. int.	$5.4 \cdot 10^{-6}$	$2.3 \cdot 10^{-6}$	$7.0 \cdot 10^{-7}$	$4.3 \cdot 10^{-8}$

The effect of applying an additional external interleaver is also shown in Table 5-14. Performance degradation is observed. This is also the result of the flat impulse length distribution.

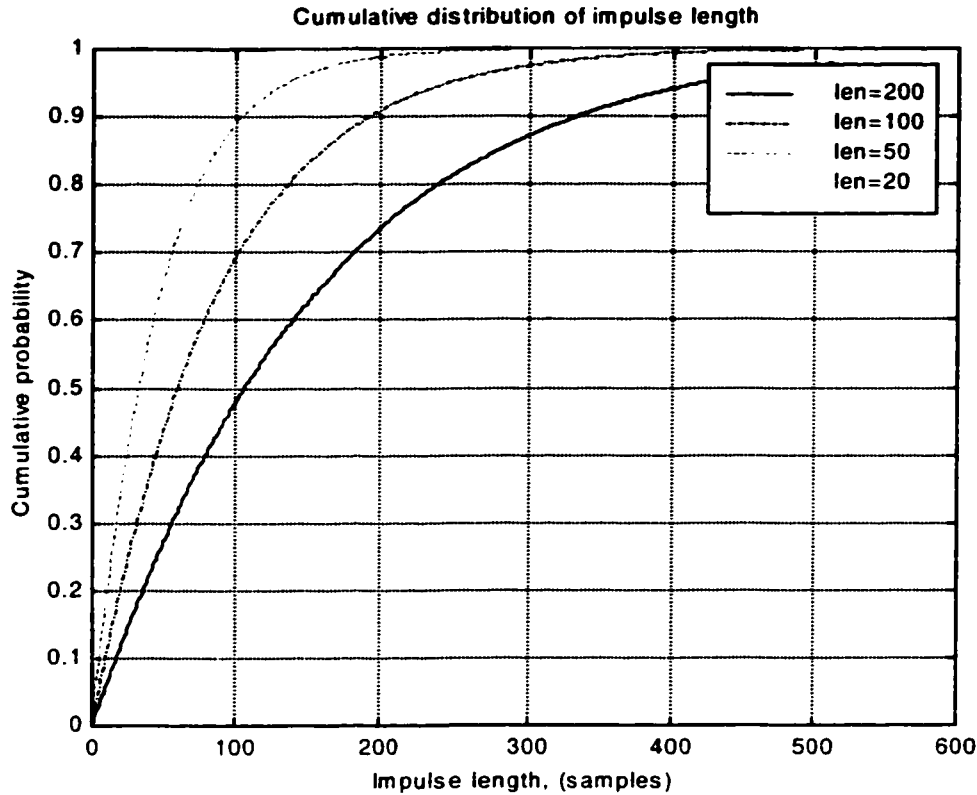


Figure 5-19 -- Impulse length distribution, one impulse per 10 seconds

5.6. TURBO CODED ADSL TRANSMISSIONS

The discussions and simulation results presented so far are for conventional QAM transmissions with a generalized impulse noise model. To obtain a good performance evaluation of ADSL DMT transmissions in an impulse noise environment, the impulse noise model should have parameters as close as possible to the previous survey results. The major characteristics of the impulse noise in subscriber lines shown in various surveys include:

- Impulse samples have high amplitudes with relatively flat distribution, compared to a Gaussian distribution [42].
- Impulses have short impulse durations with an approximately double exponential distribution. Impulse length is usually less than 100 μ s [42].
- The occurrence of impulses is unpredictable. Impulses with higher peak-values occur with a lower probability [42], [43], [47].

Combining these characteristics in the Gaussian noise model and Markov noise model will give close performance estimation of ADSL transmissions with TTCM in impulse noise environment. In the Gaussian noise model, the impulse length and impulse occurrence probability are easily adapted since these parameters are defined independently. In the two-state Markov channel model, the impulse occurrence probability is related to the impulse length through the two parameters p_a and p_b . It shows an inherent compliance to the statistical characteristics of the impulse length and impulse occurrence shown in the literature. Therefore, both of them are used in our simulation.

The following assumptions are made about the impulse noise in subscriber lines for the performance estimation of ADSL transmissions with TTCM:

- Impulses are shorter than 100 μ s.
- Impulse noise observed from all previous surveys have lengths shorter than 100 μ s. Furthermore, the impulses for testing ADSL systems defined in [11] also have lengths shorter than 100 μ s.
- Impulse samples in one impulse event have the same variance.
- Impulse samples have a variance of 5, 10, 15, 20 and 25 with a well-defined distribution.

In [45] and [47], the occurrence rate of impulse noise is shown to be approximately reverse proportional to the square of the peak impulse sample amplitude. Using as a further approximation that the square of the peak impulse voltage is directly proportional to the impulse sample variance, the occurrence rate of impulse noise is therefore inversely proportional to the impulse sample variance. This distribution is shown in Table 5-15 and will be incorporated into the performance analysis in this section. Furthermore, we will use a uniform distribution of the impulse occurrence probability as a worst-case scenario. Corresponding to the uniform distribution, we call the first distribution the “reverse variance distribution” in the following sections.

Table 5-15 – Distribution of impulse variance, σ_i^2

σ_i^2	5	10	15	20	25
Reverse variance distribution	0.44	0.22	0.15	0.11	0.08
Uniform distribution	0.2	0.2	0.2	0.2	0.2

- An impulse comes every 10, 30 or 50 seconds.

The impulse occurrence frequency depends heavily on the site of measurement and very different values have been obtained in various surveys. For simplicity, we only use the three values: 10, 30 and 50 seconds. For the impulse noise inter-arrival duration, we are using one impulse every 10 seconds as the worst case and one impulse every 50 seconds as the best case.¹⁸

5.6.1. Effect of impulse noise on single carrier ADSL transmissions

In this simulation, we still use the 64QAM-TTCM as the one used in section 5.5. In this section, we only consider the performance of the turbo-coded bits. This is because of the fact that with correct turbo decoding, the uncoded bits are protected by the set partitioning and additional non-turbo coding techniques.

The performance estimation was performed based on the two ADSL transmissions with the transmission rates shown in Table 5-16.

Table 5-16 – Two ADSL transmission test cases, 64QAM TTCM

Test cases	Transmission rate	Sample rate per second	Sample duration, (μ s)
1	2.0 Mbps	500 kbps	2
2	4.0 Mbps	1 Mbps	1

An impulse of duration 100 μ s covers 50 samples in the Test case 1 and less than 100 samples in Test case 2. The BERs caused by the impulse noise in these test cases are calculated according to the simulation results based on the Gaussian burst model and are shown in Table 5-17. It is shown that the impulse noise causes a BER of around 10^{-5} in the case of one impulse every 10 seconds. Employing erasure turbo decoding could lower the BER to 10^{-6} . Even with erasure turbo decoding, the performance still does not satisfy the BER requirement of 10^{-7} of the ADSL transmissions.

With a lower background noise variance and for impulses shorter than 50 samples, employing an external overall interleaver in combination with erasure turbo decoding could further improve the performance. It was shown in Table 5-10 that, for a N_0 of 0.7 and impulses of 50 samples, erasure decoding with the additional interleaver gives a BER

¹⁸ An impulse inter-arrival time of 50 seconds is defined in the system testing part of ADSL standard [11].

of $5.5 \cdot 10^{-4}$ for each error decoded turbo code word. This can be translated into a BER of $5.6 \cdot 10^{-8}$, $1.84 \cdot 10^{-8}$ and $1.12 \cdot 10^{-8}$ for impulse inter-arrival durations of 10, 30 and 50 seconds respectively, which meet the ADSL's BER requirement.

Table 5-17 – Effect of impulse noise in single carrier ADSL transmissions with TTCM
Rate-1/2 turbo code, $K=1024$, 64QAM transmission, 4 information bits per symbol, $N_0=0.75$, MAP decoding, 10 iterations, turbo-coded bits only, Burst Gaussian impulse model, $p_a=5 \cdot 10^{-8}$

Inter-arrival time (seconds)		10	30	50
BER (Test case 1)	Conventional decoding	$1.28 \cdot 10^{-5}$	$4.24 \cdot 10^{-6}$	$2.56 \cdot 10^{-6}$
	Erasure with CSI	$5.8 \cdot 10^{-7}$	$1.92 \cdot 10^{-7}$	$1.2 \cdot 10^{-7}$
BER (Test case 2)	Conventional decoding	$1.0 \cdot 10^{-5}$	$3.36 \cdot 10^{-6}$	$2.0 \cdot 10^{-6}$
	Erasure with CSI	$2.24 \cdot 10^{-6}$	$7.44 \cdot 10^{-7}$	$4.5 \cdot 10^{-7}$

If the Markov burst channel model is used for performance estimation, the parameters are calculated as shown in Table 5-18 and the BER estimations are shown in Table 5-19.

Table 5-18 – Markov model parameters

	p_a	p_b	Average impulse length	Impulse length reliability
Case 1	$2 \cdot 10^{-7}$	0.05	20	$L_{imp} < 50$ with 90%
Case 1.1	$2 \cdot 10^{-7}$	0.02	50	$L_{imp} < 100$ with 90%
Case 2	10^{-7}	0.02	50	$L_{imp} < 100$ with 90%
Case 2.1	10^{-7}	0.01	100	$L_{imp} < 200$ with 90%

In Table 5-18, for the Test case 1 (Test case 2), the impulses are chosen to have an average impulse length of 20-sample (50-sample) where impulses have a possibility of 90% to be shorter than the 50-sample (100-sample) in the worst case. Test case 1.1 is a more conservative case, where the impulse average length is set to the 50-sample (100-sample) in the worst case. In this case, impulses have a probability of more than 30% to be longer than the worst case.

Test case 1 and 2 give better performance than those obtained using the Gaussian burst model. This is because the impulses have a larger probability of having shorter length. For the more conservative test cases (Test case 1.1 and 2.1) the performance is worse than those obtained by using the Gaussian burst model because impulses have large probability of being longer than the worst cases used in the Gaussian burst model simulations. The performance estimations are somewhat pessimistic.

Table 5-19 – BER performance estimation with Markov impulse model

Rate-1/2 turbo code, $K=1024$, 64QAM transmission, 4 information bits per symbol, $N_0=0.75$, MAP decoding, 10 iterations, turbo-coded bits only, Markov channel model

Inter-arrival time (seconds)		10	30	50
BER Test case 1	Conventional decoding	$3.67 \cdot 10^{-6}$	$1.22 \cdot 10^{-6}$	$7.34 \cdot 10^{-7}$
	Erasure with CSI	$1.8 \cdot 10^{-7}$	$6.0 \cdot 10^{-8}$	$4.5 \cdot 10^{-8}$
BER Test case 1.1	Conventional decoding	10^{-5}	$3.3 \cdot 10^{-6}$	$2 \cdot 10^{-6}$
	Erasure with CSI	$2.24 \cdot 10^{-6}$	$7.5 \cdot 10^{-7}$	$4.5 \cdot 10^{-7}$
BER Test case 2	Conventional decoding	$8.55 \cdot 10^{-6}$	$1.67 \cdot 10^{-6}$	10^{-6}
	Erasure with CSI	$1.12 \cdot 10^{-6}$	$3.73 \cdot 10^{-7}$	$2.24 \cdot 10^{-7}$
BER Test case 2.1	Conventional decoding	$4.15 \cdot 10^{-6}$	$1.38 \cdot 10^{-6}$	$8.3 \cdot 10^{-7}$
	Erasure with CSI	$8.55 \cdot 10^{-6}$	$2.85 \cdot 10^{-6}$	$1.71 \cdot 10^{-6}$

For a general estimation of ADSL transmission performance in impulse noise environment, the Markov model is preferred because it inherently provides a good impulse length distribution compliant to those reported in the literature. Furthermore, it takes into account the cases where one impulse affects two turbo blocks, whereas this case is simply ignored in the Gaussian burst model.

However, Markov model doesn't provide a hard limit on the impulse length. In the Gaussian noise model, there is usually a hard limit on the maximum impulse noise length, which comply with previous survey results. Impulses with lengths longer than the worst case will not be considered in the Gaussian noise model.

In the cases where some specific impulse lengths and impulse occurrence distributions are required, the Gaussian burst model is obviously a better choice since it provides the maximum flexibility of parameter selection.

5.6.2. Effect of impulse noise in ADSL DMT systems

It has been shown that applying the analysis and simulation results obtained for QAM transmissions with TTCM to single-carrier ADSL transmissions is straightforward. However, in DMT transmissions, the received symbols are input into a DFT for demodulation, which averages the impulse noise energy into all subchannels in the affected DMT symbol, even though the impulse might only have an effect on a few channel symbols. Therefore, a high-power short-duration impulse gives the effect of a equivalent low-power long-duration impulse having the length of one DMT symbol. The variance of this equivalent impulse is calculated as shown in Eq. 5-39.

With zero-mean impulse samples, i_k , from the channel, the equivalent impulse noise samples in the subchannels, I_k , also have a zero mean. The variance of I_k , however, becomes $(l/N_{\text{fft}}) \cdot \text{Var}\{i_n\}$, which depends mainly on the impulse variance and the impulse length, l . The smaller is the impulse noise variance, the shorter is the impulse, the smaller the $\text{Var}\{I_k\}$ is. Because impulse noise is characterized by short-length and high-amplitude, the DMT system therefore averages its impact over all the subchannels. Furthermore, it has been shown in section 5.3.3 that the impulse contributions in the subchannels can be viewed as independent Gaussian samples.

Therefore, an evaluation of ADSL DMT transmission in impulse noise environment needs only the impulse occurrence probability, impulse sample variance and the distribution of the number of impulse samples in one DMT symbol. It is a major advantage of DMT systems that the performance estimation only needs the impulse sample variance. The exact distribution of time-domain impulse samples is no longer important for the performance estimation.

The following assumptions are made to evaluate the effect of impulse noise to DMT transmissions.

- The DMT system has 256 subchannel and $N_{\text{fft}}=512$
- Each subchannel has a bandwidth of 4.0 kHz

The two examples shown in Table 5-16, with 2 Mbps and 4 Mbps information rate, are used again in this section. Each DMT symbol should therefore carry 500 bits and 1000 bits. A subchannel could carry a constellation of any size from 4QAM to as large as 2^{15} QAM. Let us assume that, on the average, the parity bits use 1/3 of total transmitted bits (same as the 64QAM with TCM for single-carrier ADSL transmissions), the DMT symbols are therefore required to carry 750 bits and 1500 bits in these two examples. Reasonable estimations of the employed subchannels in these two cases can be made as 100 subchannels and 200 subchannels for these two test cases. The number of useable subchannels could be smaller than the estimated values because lower-frequency subchannels usually carry very large QAM constellations. Therefore, 75 and 150 will be used as the general cases.

5.6.2.1. Performance estimation with Gaussian burst model

In ADSL DMT systems with TTCM, a direct result of the bit allocation operation is a specific number of QAM symbols contained in each DMT symbol. In the proposed TTCM structure, each QAM symbol contains 4 coded bits, with a 1024-bit interleaver, one turbo block contains 512 QAM symbols. For a DMT transmission where 75 subchannels are actually carrying data, there are 7 DMT symbols covered by one turbo block, where for another DMT transmission with 200 usable subchannels each turbo block covers only 2.5 DMT symbols. The result is a different delay in the turbo encoder and decoder. The more DMT symbols contained in one turbo block, the longer the delay required by the turbo encoding and decoding processes.

BER performance of ADSL DMT systems in impulse noise environment can be obtained based on the analysis shown in the last section and the simulation results shown in section 5.5. After the DFT demodulation in the receiver, each impulse is transformed into an equivalent impulse noise with a length of one DMT symbol and a variance calculated as in Eq. 5-39 and shown in Table 5-20. The BER performances are shown in Table 5-21.

Table 5-20 – Equivalent impulse variances

σ_i^2 L_{imp}	5	10	15	20	25
75	0.73	1.46	2.2	2.92	3.65
100	0.98	1.96	2.97	3.92	4.9
150	1.46	2.92	4.38	5.84	7.3
200	1.95	3.90	5.85	7.8	9.75

Table 5-21 – BER performance of ADSL DMT transmissions with TTCM

Rate-1/2 turbo code, $K=1024$, 64QAM transmission, 4 information bits per symbol, $N_0=0.75$, MAP decoding, 10 iterations, turbo-coded bits only, Gaussian burst model, one impulse per 10 seconds

σ_i^2 distribution L_{imp}	Uniform distribution	Reverse variance distribution	Erasure decoding with CSI
75	$7 \cdot 10^{-6}$	$4.13 \cdot 10^{-6}$	$1.66 \cdot 10^{-6}$
100	$1.27 \cdot 10^{-5}$	$9.2 \cdot 10^{-6}$	$5.2 \cdot 10^{-6}$
150	10^{-5}	$9.0 \cdot 10^{-6}$	$8.3 \cdot 10^{-6}$
200	$1.2 \cdot 10^{-5}$	$1.14 \cdot 10^{-5}$	$1.17 \cdot 10^{-5}$

5.6.2.2. Performance estimation with Markov burst model

As discussed before, in ADSL DMT systems, the short-duration high-amplitude impulses are transformed into long-duration low-amplitude impulses with a fixed impulse length. Let's assume that N_{ch} subchannels are used in each DMT symbol, where $N_{ch} \in [75, 100, 150, 200]$. If an impulse occurs every T seconds, the equivalent impulse also occurs every T seconds, with a fixed length of N_{ch} -sample.

The most important difference between Markov burst model and Gaussian burst model is that the Markov burst model provides an impulse length distribution. In ADSL DMT systems, this impulse length distribution is translated into an equivalent impulse variance distribution after the DFT demodulation in the receiver.

In the ADSL DMT transmission, one DMT block contains 512 samples with a sample rate of 2.048 Msps. In an impulsive channel described by a Markov burst model with the parameters p_a and p_b , the probability of an impulse of length l occurring in one DMT block is calculated by Eq. 5-14 and Eq. 5-15. The noise variance of the equivalent impulse after the DFT operation is calculated by Eq. 4-31 with $N_{fft}=512$. In the case of $p_a=5 \cdot 10^{-8}$ (one impulse per 10 seconds) and $p_b=0.01$ (an average length of 100-sample), the average $Var\{I_k\}$ according to the uniform impulse variance distribution and the reverse variance distribution shown in Table 5-15 are plotted in Figure 5-20. The cumulative distribution of the equivalent impulse variance is plotted in Figure 5-21.

According to the distribution of the impulse lengths shown in Figure 5-20, the distribution of the equivalent impulse variance is calculated and shown in Table 5-22. To simplify the simulation, we divide the equivalent impulse noise variance into four ranges, each having its probability. The simulations were performed with these four variances. The final BER performance is the sum of the BERs obtained with these four equivalent impulse variances weighted by their probabilities. The BER performances are calculated and presented in Table 5-23.

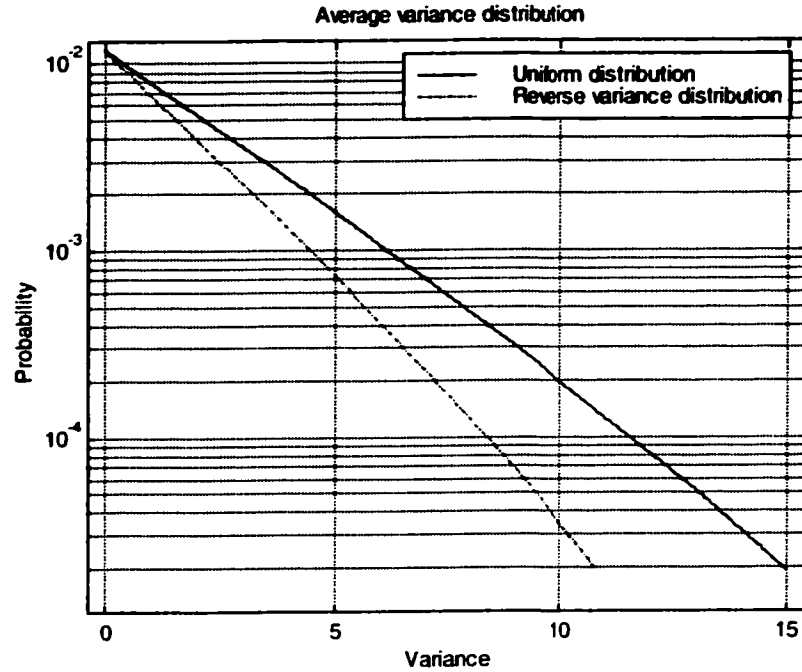


Figure 5-20 – Probability density function of the distribution of $Var\{I_k\}$
 DMT system with 256 subchannels. 512-point IFFT and FFT is performed, 2-state Markov channel model, $p_a=5 \cdot 10^{-8}$, $p_b=0.01$

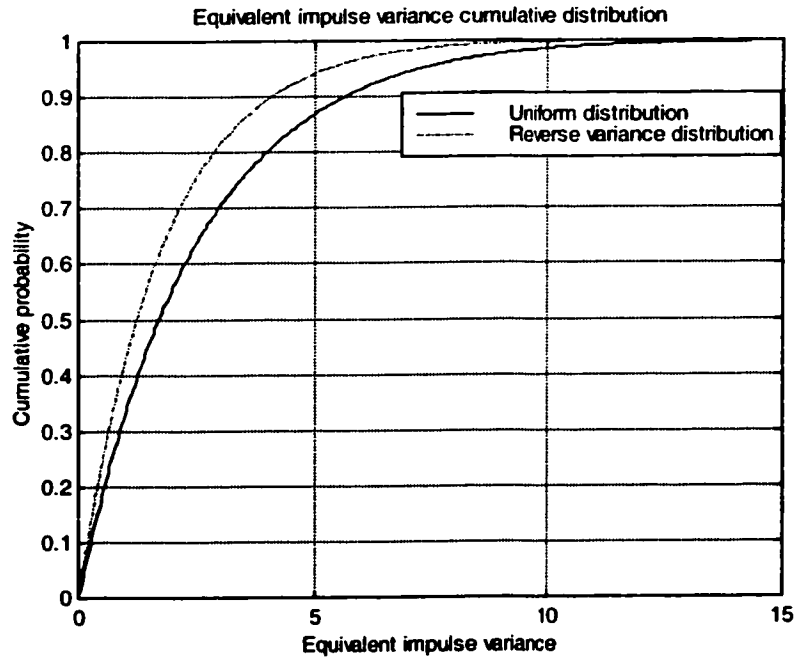


Figure 5-21 – Cumulative distribution function of $Var\{I_k\}$
 DMT system with 256 subchannels, 512-point IFFT and FFT is performed, 2-state Markov channel model, $p_a=5 \cdot 10^{-8}$, $p_b=0.01$

Table 5-22 – Distribution of the equivalent impulse noise variance

DMT system with 256 subchannels, 512-point IFFT and FFT is performed, 2-state Markov channel model, $p_a=5 \cdot 10^{-8}$, $p_b=0.01$

	Equivalent Variance	2	4	6	8
$P(\sigma_t^2 < \text{Var})$	Uniform distribution	0.55	0.25	0.12	0.08
	Reverse variance distribution	0.67	0.23	0.07	0.03

Table 5-23 – BER performance of ADSL DMT system with Markov channel model

DMT system with 256 subchannels, 512-point IFFT and FFT is performed, 2-state Markov channel model, $p_a=5 \cdot 10^{-8}$, one impulse per 10 seconds

p_b	1/75	1/100	1/150	1/200
Uniform distribution	$9.7 \cdot 10^{-6}$	$1.45 \cdot 10^{-5}$	$9.4 \cdot 10^{-5}$	$1.1 \cdot 10^{-4}$
Reverse variance distribution	$8.7 \cdot 10^{-6}$	$1.4 \cdot 10^{-5}$	$9.1 \cdot 10^{-5}$	$1.06 \cdot 10^{-4}$
Erasure decoding with CSI	$1.66 \cdot 10^{-6}$	$5.2 \cdot 10^{-6}$	$8.3 \cdot 10^{-6}$	$1.17 \cdot 10^{-5}$

It is observed that the performance of an ADSL DMT system with TTCM in the presence of impulses longer than 75 samples is usually in the range of 10^{-4} to 10^{-5} . This is mainly due to the fact that a short impulse is converted into a fixed-length long-duration impulse. Since impulse length plays a significant role in the turbo decoding performance, it is very hard to improve the performance by employing the erasure turbo decoding. However, increasing the size of the turbo code block is expected to improve the BER performance significantly, especially in the case when erasure turbo decoding is used. A slight sacrifice in the signal-to-background-noise ratio can also significantly improve the erasure turbo decoding performance. The price is the additional delay introduced by the longer turbo code interleaver.

Performance of 64QAM transmission with TTCM using a 4096-bit interleaver is shown in Table 5-24. For 100-sample erasures, it was observed by simulation that the BER is lowered by almost 40 times, down to 0.0012, for turbo coded bits in an ADSL DMT system if a turbo code interleaver of length 4096-bit is used instead of the one of length 1024-bit. This suggests that turbo codes with interleaver length of 4096-bit should be employed to efficiently combat impulse noise in turbo coded ADSL DMT systems.

Table 5-24 – Performance of 64QAM TTCM in impulse noise environments with and without erasure turbo decoding

Rate-1/2 turbo code, $K=4096$, 64QAM transmission, 4 information bits per symbol, erasure turbo decoding with CSI, $N_0=0.75$, simplified TTCM decoding, MAP decoding, 10 iterations, turbo-coded bits only

N_i	1	4	7	10	15	Erasure with CSI
100 samples	0	0.0006	0.026	0.0359	0.0404	0.0012
150 samples	0	0.0175	0.0572	0.0698	0.0884	0.0028
200 samples	0	0.0622	0.1004	0.1159	0.1279	0.0051
250 samples	0	0.1008	0.1313	0.1418	0.1542	0.0084

5.6.3. Discussion

It has been shown that erasure turbo decoding significantly improves the QAM transmission performance. However, it is not a simple task for the receiver to detect the exact position of the impulse samples and perform the erasing. In a single-carrier ADSL transmission, locations of impulse noise samples could be obtained by observing the received signal amplitudes. Single-carrier transmission never uses QAM constellations larger than 64QAM. However, impulse samples in subscriber lines usually have much higher amplitudes. The receiver can therefore decide that there is an impulse noise when it receives several high-amplitude symbols in a short period. This short period should include some time duration before and after these high-amplitude samples. Furthermore, it has been shown that when high-amplitude impulses occur, the LLR values during the impulse duration for a turbo decoding error are significantly higher than other LLR values. This can also be used as an indication for impulse location detection.

Impulse detection based on the received time-domain signal amplitudes is especially difficult in DMT transmissions, where the transmitted channel symbols are obtained by performing DFT over a block of QAM symbols, as shown in section 2.1. When the number of DFT-points is large, the resulted transmission symbols have a Gaussian distribution with a large peak-to-average power ratio. Therefore, it is hard for the receiver to tell if a large value received signal is a valid signal or an impulse sample.

However, detection of impulse noise can be performed in the frequency domain. For DMT systems, an impulse gives almost equal effect to all the subchannels in one DMT symbol. Some subchannels, especially the subchannels that carry small constellation QAM symbols, could be used to determine whether there is an impulse. Subchannels

supporting smaller constellations are good candidates for this because at the receiver, with equal background noise variance, the subchannels have almost equal minimum inter-symbol distance. Impulse noise also tends to have an equal effect on each subchannel. In this case, it is much easier to tell if there is an abnormal high amplitude impulse by using small constellations than by using large constellations. Usually, there are always some subchannels carrying 4QAM signals. Using a majority rule on these subchannels will give us a reliable detection of impulse noise.

5.7. SUMMARY

In this chapter, we investigated the performance of ADSL DMT systems with the proposed TTCM in the presence of impulse noise.

- In section 5.1, a review on impulse noise in subscriber lines was presented. Impulse noise in subscriber lines is characterized by its high-amplitude, short-duration, and small occurrence probability. It is one of the major impairments in high-speed subscriber line transmissions.
- In section 5.2, we presented a survey on various impulse statistical characteristics, which were obtained by several groups through extensive on-site measurements. The distributions of the impulse peak amplitude, the impulse length and impulse occurrence are the most important statistical characteristics of impulse noise.
- Two simple, yet flexible, mathematical models are proposed to investigate the performance of turbo coded systems in the presence of impulse noise: the Gaussian burst model and the two-state Markov burst channel model. A Gaussian burst model gives us the maximum flexibility in characterizing the impulse noise. However, certain assumptions have to be made in the performance analysis of the turbo coded systems with this impulse model. Furthermore, all of its parameters are defined independently, where in reality, these parameters are correlated. A two-state Markov channel model reflects some of the relationship between the different characteristics of impulse noise. However, the flexibility of this model is limited because all the characteristics have to be determined by the two state transition probabilities.

- A performance analysis was performed in section 5.3, based on these impulse noise models, for turbo coded BPSK transmission, turbo coded QAM transmission and turbo-coded ADSL DMT systems.
- In section 5.4, simulation results of the effect of impulse noise to turbo coded BPSK transmissions were presented. It was observed that impulse noise, even the very short ones, could cause devastating effects to a turbo decoding. Two methods were proposed to improve the performance: using an external interleaver or erasure turbo decoding. It was shown that the external interleaver only works when the impulse is very short. The erasure, however, is a very effective method for performance improvement. Furthermore, if an external RS code is employed to improve the system performance, using erasure turbo decoding can significantly reduce the interleave depth requirement from the RS code. It was also observed that by sacrificing the turbo code coding gain slightly against the background AWGN noise, the performance of erasure turbo decoder could be improved significantly. Simulations were performed on Gaussian burst model only.
- In section 5.5, simulation results of effect of impulse noise to turbo coded QAM transmissions were presented. It was observed that impulse noise, even the very short ones, could cause devastating effect to a turbo decoding. The two methods proposed for turbo coded BPSK transmissions were also considered and simulated for performance improvement. Erasure turbo decoding was again shown to provide better performance and short delay in a concatenated RS-TTCM structure. Simulations were performed with both the Gaussian burst model and the two-state Markov channel model.
- In section 5.6, the performance of the proposed TTCM in practical ADSL systems were investigated. Simulation results were presented for both single-carrier ADSL transmissions and ADSL DMT transmissions. It was observed that erasure turbo decoding provides better performance in single-carrier transmissions. For ADSL DMT systems, there was a BER floor at about 10^{-6} . This BER floor can be lowered either by increasing the turbo code interleaver size or by using an external RS code. Erasure turbo decoding could be employed to reduce the interleaving requirement for the external RS code.

Chapter 6.

Conclusions and Recommendations for Future Work

Conclusions

In this thesis, we proposed a TTCM structure to be applied in ADSL DMT systems. To reduce the decoding complexity, a simplified TTCM decoding utilizing a conventional binary turbo decoder was proposed. The basic idea is to obtain the reliability information of the turbo-coded bits from the received multi-level symbols and perform the binary turbo decoding. For best performance with this structure, Gray mapping was employed in the constellation mapping. It is shown that with Gray mapping, the performance of the simplified TTCM decoding of a QAM TTCM is at least as good as that of a turbo-coded BPSK transmission over the same AWGN channel, as long as the QAM and the BPSK constellations have the same minimum constellation distance.

It was observed by simulations that, in a stationary noise environment, applying our proposed TTCM to ADSL DMT systems provides a coding gain of 5.4 dB at a BER of 10^{-5} . By adding an outer RS code, a 7 dB coding gain could be obtained without any bandwidth expansion at a BER of 10^{-7} . Performance of using the simplified TTCM decoding is only 0.2 dB worse than that obtained when the optimum SMAP decoding algorithm is used.

Large interleave depth for the outer RS code with a concatenated RS-TTCM is required in order to lower the BER floor of the turbo decoder. A Serial-Parallel-Concatenated-RS-TTCM is shown to reduce the interleave depth requirement significantly. Furthermore, erasure RS decoding is shown to be a promising technique to decrease the interleave depth.

The simulations were performed with a turbo code interleaver length of only 1024 bits because of the delay involved in the interleaving and deinterleaving process. The performance can be improved by increasing the size of the turbo code interleaver. However, this also increases the delay. The performance can also be improved by using a SISO decoding that tries to minimize the parity bit error probability during the decoding process (in addition to minimizing the systematic bit error probability).

In the second part of this thesis, we performed a thorough investigation on the performance of ADSL DMT systems with the proposed TTCM in the presence of impulse noise. To obtain reasonable performance evaluation results, two simple, yet practical impulse noise models were proposed.

The effects of impulse noise on turbo decoding were presented. It was shown that even very short impulses could cause devastating effect to the turbo decoding performance. Erasure turbo decoding was found to be an efficient way to combat impulse noise. It was shown that erasure turbo decoding provides much better performance for short and moderate impulse lengths. Furthermore, erasure turbo decoding was shown to result in much shorter error bursts compared to a conventional turbo decoding. Therefore, it can reduce the interleaving requirement from the outer RS code significantly.

The performances of practical ADSL transmission were presented, with and without erasure turbo decoding. It was shown that the turbo coded ADSL DMT systems always give a BER of 10^{-5} to 10^{-6} . This is because the short duration impulse noise is converted

into a long duration impulse by the DFT demodulation at the DMT receiver. An effective way to combat this is to use a longer turbo code interleaver. Another method is to slightly sacrifice the signal-to-background-noise ratio to obtain better erasure turbo decoding performance.

The erasure detection could be performed by monitoring the received symbol sequence in a single-carrier transmission or by monitoring some low-rate subchannels in DMT systems.

Contributions

The list of contributions of this thesis (in order of presentation, not in order of importance) can be summarized as follows.

- We proposed a flexible bandwidth efficient turbo trellis coded modulation structure for ADSL DMT transmissions. A low complexity implementation was proposed which utilizes a conventional binary turbo decoder. Performance of the proposed TTCM was investigated and presented.
 - A survey over the existing bandwidth efficient turbo coded modulation schemes was performed. Comparisons between these turbo-coded modulations were made based on the performance, the implementation complexity, and the flexibility of being implemented in a multicarrier system.
 - A practical bandwidth efficient turbo trellis coded modulation structure is designed based on Benedetto's TTCM scheme that fits directly in current ADSL DMT systems.
 - A simplified decoding structure was proposed to significantly reduce the decoding computational load with only minor performance degradation. Most importantly, the proposed decoding structure is based on the utilization of a standard binary turbo decoder, for which commercial ASIC chipsets are already available.
 - Gray mapping was proposed in the constellation encoding process. It was proved that with the simplified TTCM decoder, using Gray mapping provides a

performance at least as good as a turbo coded BPSK transmission with the same turbo decoder over the same AWGN channel, as long as the BPSK transmission and the QAM transmission have the same minimum constellation distance.

- A theoretical performance evaluation of the proposed TTCM in QAM transmissions and ADSL DMT transmissions was performed.
- A BER upper bound is derived for QAM transmissions using the proposed TTCM, based on the use of a *uniform interleaver*.
- The performance of QAM transmissions with the proposed TTCM was estimated based on the best binary turbo code performance reported in the literature.
- Design criteria of the turbo codes used in the proposed TTCM were discussed. It was shown that our proposed TTCM complies with the optimum multi-level coding design criteria, i.e., design with equivalent channel capacities.
- Gray mapping provides an unequal error protection structure. It was shown that careful allocation of turbo-coded bits during the constellation mapping could also provide further performance improvement.
- Simulation results of using the proposed TTCM in QAM transmissions were presented. It was shown that about 5.3 dB coding gain could be obtained with an interleaver length of 1024-bit. Using the simplified TTCM decoding degrades the coding gain performance only by 0.2 dB.
- Simulation results of concatenating the TTCM with an outer RS code were obtained and presented. This concatenated coding structure achieves 7.0 dB coding gain at a BER of 10^{-7} . This coding gain is at least 1.5 dB better than the current channel coding scheme defined in ADSL DMT standard.
- In order to provide satisfactory performance, the outer RS code has to correct long error burst output from the turbo decoder. A discussion on the requirements on the outer RS code was presented. It was shown that the outer RS code has to be used with an interleaver with a large interleave depth, which, unfortunately, results in a large system delay. Several methods to reduce the delay were presented, including employing erasure RS decoding and using a Serial-Parallel-Concatenated RS-TTCM structure.

- It was proved that all the simulation and analysis results could be applied directly to ADSL DMT systems.
- Furthermore, various existing techniques to further improve the TTCM decoding performance were introduced, including using a SISO that also tries to minimize the parity bit error probability and using a combined design of the constituent codes and the interleaver.
- We performed an investigation on turbo coding performance in the presence of impulse noise. Two impulse noise models were proposed and employed in the performance evaluation. Performance evaluation of turbo coded ADSL DMT systems was performance from the simulation and analysis.
 - A survey of impulse noise in subscriber lines was performed. Important characteristics of impulse noise were reviewed. Two simple, yet practical impulse noise models were proposed to be used in our later simulations.
 - The performance analysis of turbo coded BPSK and QAM transmission systems in the presence of impulse noise was performed and theoretical expressions were presented.
 - Simulation results of the effect of impulse noise on turbo decoding were presented. It was shown that impulses, even very short ones, could have a devastating effect on turbo decoding. It was observed that impulse noise sample amplitude, impulse length and impulse occurrence frequency all play important roles in the turbo decoding performance.
 - An analysis of the effect of impulse noise on turbo-coded ADSL DMT system was presented. It was proved that the DMT demodulation converts a short-duration, high-amplitude impulse into a relatively long-duration, low-amplitude impulse. Effectively, the demodulation process averages the effect of impulse noise into all subchannels. This long-duration, low-amplitude impulse is much harder to correct.
 - The performance evaluation of employing turbo coding in practical ADSL systems were performed and simulation results were presented. It was shown that by using erasure turbo decoding, single-carrier ADSL transmissions are more robust against impulse noise. ADSL DMT systems always give a BER at around

10^{-6} . An additional outer RS code will be necessary to lower the BER to the required 10^{-7} . Employing a longer turbo code interleaver could be another solution, with the price of large system delay.

- We proposed several techniques to improve the performance of turbo coded system, including adding an additional interleaver, employing erasure turbo decoding, using erasure RS decoding and employing a concatenated RS-TTCM.
 - It was proposed to improve the turbo decoding performance by adding an external interleaver. This technique works well for very short impulses.
 - Erasure turbo decoding was proposed to improve the turbo decoding performance in the presence of impulse noise. It was further shown that erasure turbo decoding usually results in a short error burst that is limited to the impulse duration. This can significantly reduce the interleave depth requirement for the outer RS code if a concatenated RS-TTCM (or RS-Turbo code) structure is employed. By increasing the signal-to-background noise ratio slightly, the performance of erasure turbo decoding is significantly improved and the output error pattern has much better chance to be a very short burst. This advantage disappears for long impulses (long erasures).
 - Several erasure detection methods were proposed. First of all, in BPSK transmissions and single-carrier QAM transmissions, erasure detection can be implemented by monitoring the received signal amplitude. Secondly, in DMT systems, it was shown that monitoring multiple low-rate subchannels can give accurate impulse noise detection. Then the whole DMT symbol containing the impulse noise could be erased. Furthermore, LLR values output from a turbo decoder can also be used as an accurate indication whether there is an impulse noise in current turbo block or not.
 - The requirements on the external RS code in an RS-TTCM (RS-turbo code) system against impulse noise were present. It was observed that by employing erasure turbo decoding, the interleave depth requirement is significantly reduced.

Recommendations for Future Work

1. Performance of employing the TTCM in ADSL DMT systems is evaluated through the simulation result obtained from turbo coded single-carrier QAM transmissions and theoretical analysis. However, the best way to test the performance of the proposed TTCM in a practical ADSL DMT system is to perform simulations of a practical turbo coded ADSL DMT transmission with practical loop model, bit allocation algorithm and crosstalk noise models.
2. Gray mapping provides unequal protection in the TTCM. Is there a way of utilizing this unequal error protection to improve the decoding performance? It would be an interesting topic to investigate the optimum bit assignment of the bits position in one turbo block with the Gray mapping. The effective minimum free Euclidean distance can be increased by proper bit assignment.
3. A BER upper bound of using TTCM has been derived and presented in section 4.2.2. It would be interesting to derive a BER upper bound of TTCM in an impulse noise environment using a reasonable impulse model in subscriber line transmissions.
4. Can we find a practical TTCM structure where the decoding provides no BER floor before 10^{-7} on the turbo-coded bits? In this case, only an RS code would be needed for un-turbo-coded bits. What would be the minimum length of the turbo code interleaver to meet this requirement?
5. It has been shown by Benedetto that his TTCM in a 16QAM transmission with rate-1/2 turbo code achieves a BER floor lower than 10^{-7} . However, the decoding was based on SMAP algorithm. It is possible to derive the 2-bit symbol reliability from the received QAM symbols and then use SMAP. Will this decoding structure provide optimum performance? What is the minimum requirement on the interleaver length?
6. It would be an interesting project to find the optimal compromise of the turbo code parameters in a turbo coded ADSL DMT systems, such as the implementation complexity, the system delay, the performance against stationary noise and the performance against impulse noise.

Appendix A. Turbo Code Components

A.1 Recursive Systematic Convolutional Codes

The performance of a turbo code is determined by its weight distribution, which again depends mainly on both the weight distributions of the CC's and the interleavers. Therefore, the choice of the constituent convolutional codes plays an important role in turbo codes. The better is the performance of each constituent code, the better the performance of the turbo codes is. Simulation results in [14] showed that, for high SNR, significant performance improvement could be achieved by increasing the constraint length of the CC's. An improvement of about 6 dB is reported by increasing the constraint length from 2 states to 16 states for both block length of 100 bits and 1000 bits. This can help in situations where the system delay must be kept low, in the sense that interleaver length (and thus delay) can be traded with CC's complexity.

Recursive systematic convolutional codes were proved necessary in achieving the outstanding performance of turbo codes. A recursive systematic convolutional code is a convolutional code in which the next state and current output depends not only on the current state and the current input, but also on the previous output, as shown in Figure A-1. Any non-recursive non-catastrophic systematic convolutional encoder (NRSC) is equivalent to an RSCC in that they possess that same set of code sequences [51]. The

difference between them is the mapping from the data sequence to the code sequence. It was shown in [1] that there is a slight difference between the performance of the NRSCs and RSCCs. However, it will be shown in the following that using RSCCs is necessary for turbo codes to achieve their exceptional performance.

With an NRSC, whose code generator is $G_{NR}(D)=[g_1(D), g_2(D)]$, a corresponding RSCC can be obtained by the generator

$$G_R(D) = \begin{bmatrix} 1 & \frac{g_2(D)}{g_1(D)} \end{bmatrix}$$

Eq. A-1

From (Eq. A-1), it is clear that only the error sequence, which can be divided by $g_1(D)$ will produce finite-weight coded bit sequence. Turbo codes are still linear codes, so it is valid to make the assumption that the all zero sequence is transmitted and that any error sequence will be a valid non-zero data sequence. From [51], the following important factors are given as,

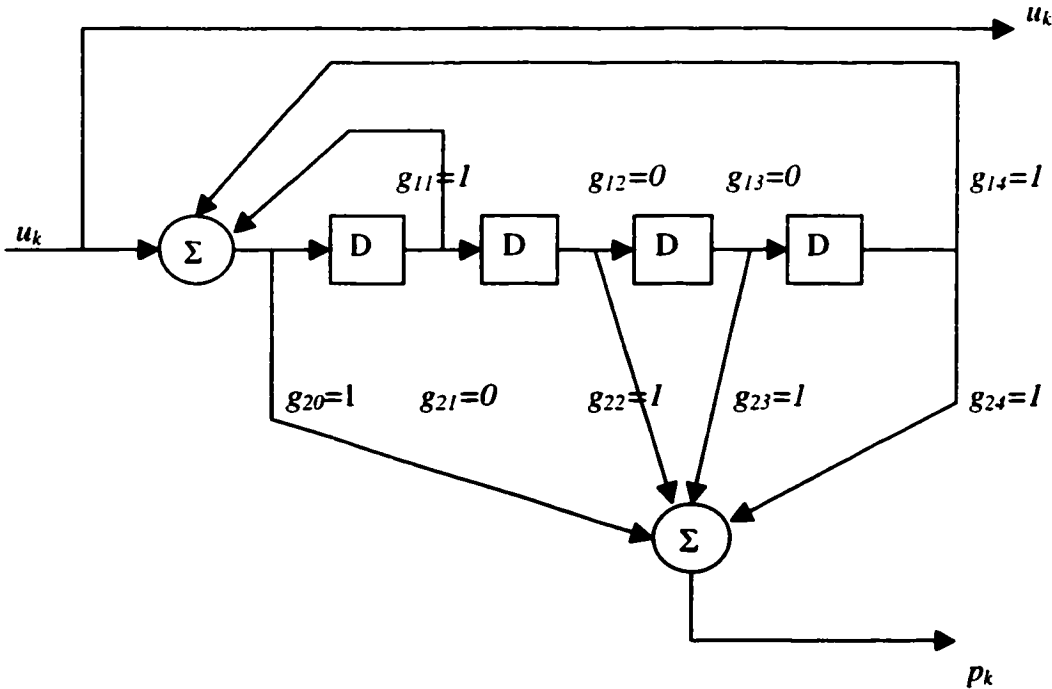


Figure A-1 – Rate 1/2 recursive systematic convolutional encoder structure

- A weight-1 input sequence produces an infinite weight output, because such an input is never divisible by a polynomial $g_1(D)$. Whereas, weight-1 input produces low weight output in both CC's if they are NRSC codes. This explains why using NRSC in turbo codes will not give much coding gain improvement.
- For any non-trivial $g_1(D)$, there exists a family of weight-2 inputs of the form $D^j(1+D^{q-1})$, $j \geq 0$, which produce finite weight outputs, i.e., which are divisible by $g_1(D)$. When $g_1(D)$ is a primitive polynomial of degree m , $q=2^m$, otherwise, q is the size of the *splitting field* of $g_1(D)$. By using primitive polynomial $g_1(D)$, the number of weight-2 input sequences that will give finite weight outputs is greatly decreased.

With careful design of interleavers, the number of weight-2 information sequences that give finite weight code word can be set to zero, which could improve the performance of the turbo codes. However, some side effects may occur, for example, the number of weight-3 data words that give finite weight code words may increase, which may cancel off the improvement.

As a conclusion, using RSCC is necessary for the excellent performance of turbo codes

A.2 Interleaver Design

Using a proper interleaver between the two parallel-concatenated RSCCs is one of the most important reasons of the excellent performance of turbo codes. The weight distribution of the code words out of this parallel concatenation depends on how the code words from one of the component encoders are combined with code words from the other encoder(s). Intuitively, we would like to avoid pairing low-weight code words from one encoder with low-weight codewords from the other encoder. Many such pairings can be avoided by proper design of the interleaver. It was shown in [1], [52], [53] and [54] that an interleaver that permutes data in a random fashion provides better performance than the conventional interleavers (block and convolutional interleavers) [55]. This is because conventional interleavers are very inefficient in breaking up the error patterns that give the shortest distance in turbo codes. In [54], it is shown that using conventional interleavers also causes severe correlation between the extrinsic information used in the iterative decoding of turbo code, which will, in turn, degrade the decoding performance.

Therefore, *non-conventional interleavers* are always used in turbo codes. The *non-conventional interleavers* used in turbo codes should meet two conditions. Firstly, they should give a good *spreading factor*, which will set the distance between two adjacent bits before interleaving as far as possible after interleaving. Secondly, the interleavers should have a large *dispersion factor*, which means that the distribution of pair-wise distance spectrum of the interleaved sequence should be as dispersive as possible. Large dispersion will lower the probability of having certain error patterns that cause low weight output by both CCs.

Pseudo-random interleavers are known to give very high *dispersion factor*, which actually approaches 1. However, they also give zero *spreading factor*. *Conventional interleavers* will give good *spreading factors*, but with very low *dispersion factor*, which is always less than 1%. Therefore, the performance of using either *pseudo-random interleavers* or *conventional interleavers* is not expected to be very good. One solutions of designing interleavers in turbo codes is the so-called '*S-random*' interleavers. An '*S-random*' interleaver is a randomly generated interleaver with a designed *spreading factor*, *S*, which is called the *S-parameter*. This interleaver is always found by computer search. Another solution is given in [54], where the interleaver design is based on the correlation property of *extrinsic* information at the output of soft-output decoders. Simulations have shown that both of them can achieve excellent results. In this thesis, '*s-random*' interleavers are used. Table I gives the parameters of the *s-random* interleavers that are employed in the simulations.

Table A- 1 – Parameters of some *S-random* interleavers

Interleaver length	S-parameter	Dispersion ratio
100	6	0.82
200	9	0.8130
400	11	0.8019
800	13	0.6810
1024	17	0.8109
2048	19	0.8130
4096	22	0.8135

The performance of turbo codes can be divided into two phases along the SNR range. At low SNR, the BER of turbo codes decreases very fast, which is called 'waterfall' region of turbo codes in [56]. However, it is shown in [1] that the slope of the BER vs.

SNR curve becomes much smaller after a BER of 10^{-6} . The SNR range after this point is referred to as BER floor region. Therefore, after 10^{-5} , the coding gain performance will not be so significant. It can be concluded that turbo codes provide excellent performance for moderate BER requirements. The BER floor in the turbo codes can be lowered by carefully designed constituent codes and interleavers. In [56], different CCs are used to lower the 'error floor' at the cost of a worse performance in the 'waterfall' region. In [57], the combined turbo codes and interleaver design achieves BER floor after BER values lower than $2 \cdot 10^{-7}$.

Appendix B. SISO Algorithms

B.1. Symbol-by-symbol MAP algorithm

The MAP algorithm for trellis codes was proposed simultaneously by Gahl, Cocke, Jelinek, and Raviv in [19] and by McAdam, Welch, and Weber in [58]. In [1], the algorithm was adapted to systematic convolutional codes. In this algorithm, the information bits are processed in blocks. The algorithm is based on maximizing the *a posteriori probability* (APP) of the bits in each block, with the information provided by the trellis structure of the convolutional code. The soft output, i.e., the reliability information of bit d_k is the log-likelihood ratio $L(d_k)$, which is defined as:

$$L(d_k) = \log \frac{\Pr\{d_k = 1/R\}}{\Pr\{d_k = 0/R\}}$$

Eq. B- 1

where R is the observed sequence at the receiver. The decision is forced to a '1' if the log-likelihood ratio (LLR) is positive, and '0' if the LLR is negative. $\Pr\{.\}$ is the APP's of d_k equals to one and zero.

The LLR is calculated in the summation form as the following equation,

$$\begin{aligned}
 L(d_k) &= \log \frac{\Pr\{d_k = 1, R\} / \Pr\{R\}}{\Pr\{d_k = 0, R\} / \Pr\{R\}} \\
 &= \log \frac{\sum_m \sum_{m'} \Pr\{d_k = 1, S_k = m, S_{k-1} = m', R_1^{k-1}, R_k, R_{k+1}^N\}}{\sum_m \sum_{m'} \Pr\{d_k = 0, S_k = m, S_{k-1} = m', R_1^{k-1}, R_k, R_{k+1}^N\}}
 \end{aligned}$$

Eq. B- 2

where, S_k is the state at the k^{th} instant, R_i^j is the vector from the i^{th} observation to the j^{th} observation.

Equation (Eq. B- 2) can be further expressed as:

$$\Lambda(d_k) = \log \frac{\sum_m \sum_{m'} \Pr\{R_{k+1}^N / S_k = m\} \Pr\{S_{k-1} = m' / R_1^{k-1}\} \Pr\{d_k = 1, S_k = m, R_k / S_{k-1} = m'\}}{\sum_m \sum_{m'} \Pr\{R_{k+1}^N / S_k = m\} \Pr\{S_{k-1} = m' / R_1^{k-1}\} \Pr\{d_k = 0, S_k = m, R_k / S_{k-1} = m'\}}$$

Eq. B- 3

With the following definitions:

$$\alpha_k(m) = \Pr\{S_k = m / R_1^k\}$$

Eq. B- 4

$$\beta_k(m) = \frac{\Pr\{R_{k+1}^N / S_k = m\}}{\Pr\{R_{k+1}^N / R_1^k\}}$$

Eq. B- 5

and

$$\gamma_i(R_k, m', m) = \Pr\{d_k = i, S_k = m, R_k / S_{k-1} = m'\}$$

Eq. B- 6

Then equation (Eq. B- 3) can be expressed as

$$\Lambda(d_k) = \log \frac{\sum_m \sum_{m'} \gamma_1(R_k, m', m) \alpha_{k-1}(m') \beta_k(m)}{\sum_m \sum_{m'} \gamma_0(R_k, m', m) \alpha_{k-1}(m') \beta_k(m)}$$

Eq. B- 7

The $\alpha_k(m)$ and $\beta_k(m)$ can be calculated iteratively by,

$$\alpha_k(m) = \frac{\sum_{m'} \sum_{i=0}^1 \gamma_i(R_k, m', m) \alpha_{k-1}(m')}{\sum_m \sum_{m'} \sum_{i=0}^1 \gamma_i(R_k, m', m) \alpha_{k-1}(m')}$$

Eq. B- 8

$$\beta_k(m') = \frac{\sum_m \sum_{u_i} \gamma_i(R_k, m, m') \cdot \beta_{k+1}(m)}{\sum_m \sum_{m'} \sum_{u_i} \gamma(R_{k+1}, m, m') \cdot \alpha_k(m)}$$

Eq. B- 9

The convolutional encoder always starts from the zero state. Therefore, the following initial values for $\alpha_k(m)$, are used in the calculation,

$$\alpha_0(0) = 1; \quad \alpha_0(m) = 0 \quad \forall m \neq 0$$

Eq. B- 10

As for $\beta_k(m)$, if the trellises of both CCs end at zero state, similar initial state will be used as for $\alpha_k(m)$, beginning from the last time instant though.

$$\beta_N(0) = 1; \quad \beta_N(m) = 0 \quad \forall m \neq 0$$

Eq. B- 11

where, K is the length of a block.

A simple way of initializing $\beta_k(m)$, in case of non-terminated encoder, is to assign a uniform value to all the possible states at the last time instance, where we have

$$\beta_N(m) = \frac{1}{N} \quad \forall m$$

Eq. B- 12

Apparently, with condition (Eq. B- 12), the last bits of each block are not taken into account because no reliable initial information is available for them.

The probability $\gamma_i(R_k, m', m)$ can be determined from the transition probabilities of the discrete Gaussian memoryless channel and the transition probabilities of the encoder trellis:

$$\begin{aligned} \gamma_i(R_k, m', m) &= p(R_k/d_k = i, S_k = m, S_{k-1} = m') \cdot \\ &\quad q(d_k = i/S_k = m, S_{k-1} = m') \cdot \\ &\quad \pi(S_k = m/S_{k-1} = m') \end{aligned}$$

Eq. B- 13

where $p(/)$ is the transition probability of the discrete Gaussian memoryless channel, and $\pi(S_k=m/S_{k-1}=m')$ is the *a priori* probability of the current bit.

The final output of this SISO decoding algorithm is the LLR. However, in the iterative decoding, the whole LLR cannot be transferred to the next decoder as

independent *a priori* information, because the LLRs are dependent on the current systematic symbol. It has been shown in [1], [2], [51] and [59] that the using MAP, the LLR can be expressed as:

$$L(d_k) = L_c(d_k) + L_{apri}(d_k) + L_e(d_k)$$

Eq. B- 14

where, the L_c express the channel information of the current systematic channel symbol, the L_{apri} is the *a priori* value for the current bit and L_e is the *extrinsic* information obtained based on the trellis structure together with the other systematic symbol and the received parity symbols. If a proper interleaver is chosen, L_e will be only weakly correlated with the received systematic symbol. Therefore, it can be fed into the second decoder as additional information, because in the second decoder, the systematic symbols and only the parity symbols from the second encoder are used. Still, because the *extrinsic* information at time instance k from the first decoder is correlated to all the systematic symbols except the k^{th} systematic symbol, the output *extrinsic* information will be weakly correlated to every systematic symbol in the symbol block under consideration after the second iteration. The more iteration the decoder employs, the more correlation between the extrinsic information and the systematic symbols. This can explain the fact that until some point, performing more iteration will not give any additional coding gain performance improvement.

B.2. Varieties of MAP algorithm [20], [18]

We have shown that the LLR of the bits in one frame could be expressed as shown in equation Eq. B- 7. However, implementation wise, it is too complex to be feasible. The calculation of the actual probabilities can be avoided by using the logarithm of probabilities and the approximation $\log(e^{L_1} + e^{L_2}) \cong \max(L_1, L_2)$. In this case, the algorithm works with $\log\alpha_k(m)$, $\log\beta_{k-1}(m')$, and $\log\gamma_k(m', m, R_k)$ and the summation in Eq. B- 7, Eq. B- 8, and Eq. B- 9 are replaced by the corresponding maximization. This sub-optimal realization of the “symbol-by-symbol” MAP rule is named the Max-log-MAP rule realization. The performance shown in [17] illustrates a coding gain loss less than 1 dB for this complexity reduction.

In [17], a Log-MAP algorithm is proposed to improve the approximation by adding a correction term to the approximation made in the Max-log-MAP algorithm, with which

the near optimum performance can be achieved. This modification is obtained only by adding a look-up table, therefore, the complexity of the Log-MAP will be a little more than Max-Log-MAP, while still much less than MAP algorithm.

B.3. SOVA – Soft-in/Soft-out Viterbi Algorithm

Viterbi algorithm is known as the maximum-likelihood sequence estimation algorithm. However, the output of the traditional VA outputs hard-decisions only. In [21], a modified VA was proposed with soft output. The soft-output is calculated by use of the difference between the survivor and the discarded paths. The SOVA is described in details in the following, because it is the most attractive SISO in practical system design in implementation issue.

Assume that the soft Viterbi decoder makes a decision with delay δ , δ being large enough so that all 2^v surviving paths have been merged with sufficiently high probability, where 2^v is the number of states in the trellis. The SOVA selects a survivor for state s_k as conventional VA, $0 \leq s_k \leq S = 2^v - 1$ at time k . It does so by selecting the path with the smallest ML metric, which, for the AWGN channel, is

$$M_m = \frac{E_s}{N_0} \sum_{j=k-\delta}^k \sum_{n=1}^{N_b} (y_{jn} - x_{jn}^{(m)})^2, \quad m = 1, 2 \quad \text{Eq. B- 15}$$

where $x_{jn}^{(m)}$ is the n^{th} bit of the N_b bits on the branch of the m^{th} path at time j , y_{jn} is the received value at the same position, and E_s/N_0 is the SNR. It follows that

$$\text{Prob}(\text{path } m) \sim e^{-M_m}, \quad m = 1, 2 \quad \text{Eq. B- 16}$$

The path with the smaller metric is labeled by $m = 1$. This means that $M_1 \leq M_2$, which implies that the VA selects path 1 (neglecting ties). Therefore, the probability of selecting the wrong surviving path is

$$p_{sk} = \frac{e^{-M_2}}{e^{-M_2} + e^{-M_1}} = \frac{1}{1 + e^{M_2 - M_1}} = \frac{1}{1 + e^{\Delta}} \quad \text{Eq. B- 17}$$

with $\Delta = M_2 - M_1 \geq 0$. p_{sk} approaches 0.5 if $M_1 \equiv M_2$ and 0 if $M_2 - M_1 \gg 1$. The p_{sk} is the probability that the VA has made errors in all the e positions where the information bits of path 2 differ from path 1; in other words, if

$$u_j^{(1)} \neq u_j^{(2)}, \quad j = j_1, j_2, \dots, j_e$$

Eq. B- 18

Positions where $u_j^{(1)} = u_j^{(2)}$ are not affected by the survivor decision. Let δ_m be the length of those two paths until they merge. There are e different information values, and $\delta_m - e$ nondifferent values. Assume that the probabilities $\hat{p}_j$ of pervious erroneous decisions with path 1 have been stored. Under the assumption that path 1 has been selected the probabilities for the e differing decisions on this path are modified according to

$$\hat{p}_j \leftarrow \hat{p}_j \cdot (1 - p_{sk}) + (1 - \hat{p}_j) \cdot p_{sk}, \quad j = j_1, \dots, j_e$$

Eq. B- 19

where $0 \leq \hat{p}_j \leq 0.5$.

This formula requires statistical independence between the random variables $\hat{p}_j$ and p_{sk} , which is approximately true for most of the practical codes. The recursion could be directly performed on the log-likelihood ratio

$$\hat{L}_j = \frac{1}{\alpha} \log \frac{1 - \hat{p}_j}{\hat{p}_j}, \quad 0 \leq \hat{L}_j \leq \infty$$

Eq. B- 20

Using Eq. B- 17, Eq. B- 19 and Eq. B- 20, the following is obtained after some calculation

$$\hat{L}_j \leftarrow f(\hat{L}_j, \Delta) = \frac{1}{\alpha} \log \frac{1 + e^{(\alpha \hat{L}_j + \Delta)}}{e^\Delta + e^{\alpha \hat{L}_j}}$$

Eq. B- 21

with $\Delta = M_2 - M_1 \geq 0$, $j = j_1, \dots, j_e$. The function $f(\hat{L}_j, \Delta)$ should be tabulated with $\hat{L}_j$ and Δ as input variables and need not to be calculated at each step. The factor α prevents overflow with increasing SNR. A proper choice to achieve asymptotically $E[\hat{L}_j] = 1$ is

$$\alpha = 4d_{free} \frac{E_s}{N_0}$$

Eq. B- 22

where d_{free} is the free distance of the code. A good approximation of Eq. B- 21 is

$$f(\hat{L}_j, \Delta) = \min(\hat{L}_j, \Delta / \alpha)$$

Eq. B- 23

which requires neither a table nor the SNR.

In [60], a modification is made to the traditional SOVA talked above to improve its performance. The soft decisions of the nondifferent bits along the two paths entering one state node are also changed after the metric comparison, according to the following:

$$\hat{L}_{j-new} = \min\{\Delta + \hat{L}_j^2, \hat{L}_j^1\}$$

Eq. B- 24

This equation is base on the fact that, for the nondifferent bits, $\hat{L}_j^2$ is the metric difference between path 2 and a previously discarded path in the trellis, which has a different bit decision corresponding to the j^{th} position in path 2. Therefore, it also has different bit decision at this position with path 1. This path, now, should be taken into account as the most likely path with the different decision on this bit. Therefore, the soft decision of this bit should be updated again in a similar way presented in Eq. B- 21, of which the simplified form is shown as Eq. B- 24.

To implement the SOVA algorithm, some modifications must be made to the traditional VA, which will increase the complexity. The soft-output will also take additional bits to be represented, which also requires more memory.

It is shown in [17] that the complexity of SOVA is less than one half of the Max-log-MAP algorithm. It is also shown in [17] that the performance of SOVA is approximately 1 dB less than MAP algorithm, a little less than Max-Log-MAP. However, it is proved in [61] that after the modification in Eq. B- 24, the SOVA algorithm provides exactly the same soft information as the Max-Log-MAP algorithm.

B.4. Soft decoding algorithm comparisons

VA is known to give the ML performance on sequence estimation. However, it doesn't necessary provide minimum symbol (bit) error probability for each symbol (bit). On the other hand, symbol-by-symbol MAP algorithm described in [19] is an algorithm for minimum symbol error probability. The modified version of this algorithm was used in [1] iteratively to perform the turbo decoding, which is still referred to as MAP algorithm. Max-Log-MAP is an approximation of the MAP algorithm by using some mathematical approximations to reduce the huge algorithm of MAP. It is thus a

suboptimal soft-decoding algorithm. SOVA is a modified version of VA, which could provide soft output for each decoded bit. It is also only suboptimal in the sense of minimizing the symbol error probability.

Using a binary code as an example, the performance difference can be explained by the decoding operations of these decoding algorithms. First of all, MAP searches both forward and backward along the trellis structure. At each stage of the trellis, it finds out the sum of all state transition for both situations of input '0' or '1'. Apparently, it takes into account of all of the paths across the trellis. Max-Log-MAP makes the simplifications that at each stage, it only computes the probability of the best state transition for the inputs of '0' and '1', and neglects the other paths. It is proven in [61] that the modified SOVA provides exactly the same performance of Max-Log-MAP algorithm. It is also worth noting that the performance difference we are looking at here is only several tenths of a dB.

The complexities of these three algorithms are very different. The MAP algorithm, which is usually performed in logarithm form to reduce the complexity, performs the forward recursion and backward recursion; then calculates the LLR values of the block of bits under processing. The Max-Log-MAP replaces the sum over K with taking the MAX operation in the MAP algorithm, which also performs two-way recursion. SOVA only gives some additional computation to VA and uses some more memory units to save the soft output. It only needs a one-way recursion. Apparently, for the same turbo code block length, SOVA requires the minimum complexity. Furthermore, there has been well-defined implementation of VA in the literature for a long time.

References

- [1] C. Berrou, A. Glavieux and P. Thitimajshima, "Near Shannon Limit Error-Correcting Coding and Decoding: Turbo-Codes," ICC'93, pp. 1064-1070.
- [2] C. Berrou and A. Glavieux, "Near Optimum Error Correcting Coding And Decoding: Turbo-Codes," IEEE Trans. On Commun., VOL. 44, No. 10, Oct. 1996, pp. 1261-1271.
- [3] S. Benedetto, D. Divsalar, G. Montorsi, F. Pollara, "Parallel Concatenated Trellis Coded Modulation", ICC'96, pp. 974-978.
- [4] W. J. Blackert and S. G. Wilson, "Turbo Trellis Coded Modulation", Proc. of CISS'96, March 1996, Princeton, NJ, USA.
- [5] S. G. Wilson, "Digital Modulation and Coding," Prentice-Hall, Inc, 1996.
- [6] J. A. C. Bingham, "Multicarrier Modulation for Data Transmission: An Idea Whose Time Has Come", IEEE Commun. Magazine, pp. 5-14, May 1990.
- [7] T. Nicholas Zogakis, James T. Aslanis Jr., John M. Cioffi, "A Coded and Shaped Discrete Multitone System", IEEE Trans. on Commun., Vol. 43, No. 12, pp. 2942-2949, Dec. 1995.

- [8] S. B. Weinstein and P. M. Ebert, "Data Transmission by Frequency-Division Multiplexing Using the Discrete Fourier Transform", IEEE Trans. on Commun. Tech., Vol. Com-19, No. 5, pp. 628-634, 1971.
- [9] A. Peled, A. Ruiz, "Frequency Domain Data Transmission Using Reduced Computational Complexity Algorithms", Proc. IEEE ICASSP, pp. 964-967, 1980.
- [10] J.-J. Werner, "The HDSL Environment", IEEE Journal on Selected Areas in Commun., Vol. 9, No. 6, pp. 785-800, Aug. 1991.
- [11] "ADSL DMT STANDARD, T1.413, Issue 2", Sept. 26, 1997.
- [12] P. S. Chow, J. C. Tu, J. M. Cioffi, "Performance Evaluation of a Multichannel Transceiver System for ADSL and VHDSL Services", IEEE Journal on Selected Areas in Commun., Vol. 9, No. 6, pp. 909-919, Aug. 1991.
- [13] S. Le Goff, A. Glavieux and C. Berrou, "Turbo-Codes and High Spectral Efficiency Modulation", ICC'94, pp. 645-649.
- [14] S. Benedetto, G. Montorsi, "Unveiling Turbo Codes: Some Results on Parallel Concatenated Coding Schemes", IEEE Trans. Inform. Theory, Vol. 42, No. 2, March 1996.
- [15] D. Divsalar and R. J. McEliece, "Effective Free Distance of Turbo Codes," Elect. Lett., vol. 32, no. 5, Feb. 29, 1996, pp. 445-446.
- [16] D. Divsalar and F. Pollara, "On the Design of Turbo Codes," TDA progress rep. 42-123, Jet Propulsion Lab., Pasadena, CA, Nov. 15, 1995, pp. 99-121.
- [17] P. Robertson, E. Villebrun and P. Hoeher, "A Comparison of Optimal and Sub-Optimal MAP Decoding Algorithm Operating in the Log Domain", ICC '95, pp. 1009 – 1013.
- [18] J. Hagenauer, E. Offer and L. Papke, "Iterative Decoding of Binary Block and Convolutional Codes," IEEE Trans. On Inform. Theory, VOL. 42, No. 2, March 1996, pp. 429-445.

- [19] R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal Decoding of Linear Codes for Minimizing Symbol Error Rate," *IEEE Trans. On Inform. Theory*, pp. 284-287.
- [20] S. S. Pietrobon, "Implementation and Performance of A Turbo/MAP Decoder," *International Journal of Satellite Commun.*, 16, 23-46, 1998.
- [21] J. Hagenauer, P. Hoeher, "A Viterbi Algorithm with Soft-Decision Outputs and its Applications", in *Proc., IEEE Globecom Conf.* (Dallas, TX, Nov. 1989), pp. 1680-1686.
- [22] G. Ungerboeck, "Channel coding with multilevel/phase signals," *IEEE Trans. Inform. Theory*, vol. IT-28, pp.55-67, Jan. 1982.
- [23] P. Robertson, T. Worz, "A Novel Bandwidth Efficient Coding Scheme Employing Turbo Codes," *ICC'96*, pp. 962-967.
- [24] J. P. Lauer and J. M. Cioffi, "Turbo Coding for Discrete Multitone Transmission," *Globecom'98*, pp. 3256-3260.
- [25] H. R. Sadjadpour, "Application of Turbo Codes for Discrete Multi-Tone Modulation Schemes," *ICC'99*, pp. 1022-1027.
- [26] P. Robertson and T. Worz, "Extensions of Turbo Trellis Coded Modulation to High Bandwidth Efficiencies," *ICC'97*, v 3.
- [27] A. J. Viterbi, J. K. Wolf, E. Zehavi, R. Padovani, "A Pragmatic Approach to Trellis-Coded Modulation", *IEEE Commun. Magazine*, July 1989, pp. 11-19.
- [28] V. Demjanenko, A. Torres, "Considerations about the power penalty of Reed-Solomon Forward Error Correction in ADSL Systems in regarding to the use of inner encoders," *ITU contribution, Temporary Document NG-097*, August 1998.
- [29] S. Benedetto, G. Montorsi, "Design of Parallel Concatenated Convolutional Codes," *IEEE Trans. on Commun.*, Vol. 44, No. 5, pp. 600, May 1996.
- [30] T. M. Duman, "Turbo Codes and turbo-coded modulation: Analysis and performance bounds," *Ph.D. dissertation, Elect. Comput. Eng. Dep., Northeastern Univ., Boston, MA*, May, 1998.

- [31] S. Benedetto, R. Garelo and G. Montorsi, "A Search for Good Convolutional Codes to be Used in the Construction of Turbo Codes," IEEE Trans. on Commun., Vol. 46, No. 9, pp. 1101 – 1105, Sept. 1998..
- [32] J. Yuan, B. Vucetic, and W. Feng, "Combined Turbo Codes and Interleaver Design," IEEE Trans. on Commun., Vol. 47, No. 4, April 1999.
- [33] U. Wachsmann, R. F. H. Fischer, and J. B. Huber, "Multilevel Codes: Theoretical Concepts and Practical Design Rules," IEEE Trans. on Inform. Theory, Vol. 45, No. 5, July 1999.
- [34] U. Wachsmann, J. Huber, "Power and Bandwidth Efficient Digital Communication Using Turbo Codes in Multilevel Codes," European Trans. on Telecommunications & Related Technologies. v 6 n x5, pp. 557-567, Sep-Oct 1995.
- [35] W. J. Blackert, E. K. Hall and S. G. Wilson, "An Upper Bound on Turbo Code Free Distance," ICC (or GLOBLECOM)'96, pp. 957-961.
- [36] J. Lodge, R. Young, P. Hoeher, J. Hagerauer, "Separable MAP 'filters' for the decoding of product and concatenated codes," ICC'93, pp. 1740-1745, May 1993.
- [37] O. F. Acikl, W. E. Ryan, "Punctured Turbo-Codes for BPSK/QPSK Channels," IEEE Trans. on Commun., Vol. 47, No. 9, pp. 1315-1323, Sept. 1999.
- [38] M. Robin, "Impulse Noise Modeling and Simulation in the Loop Plant", Internship Report, University of Ottawa, 1996.
- [39] S. A. Losmopoulos, P. TakisMathiopoulos, and M. D. Gouta, "Fourier-Bessel Error Performance Analysis and Evaluation of M-ary QAM schemes in an Impulsive Noise Environment," IEEE Trans. on Commun., Vol. 39, No. 3, pp. 398-404, march 1991.
- [40] M. Ghosh, "Analysis of the Effect of Impulse Noise on Multicarrier and Single Carrier QAM Systems," IEEE Trans. on Commun., Vol. 44, No. 2, pp. 145-147, Feb. 1996.

-
- [41] M. Barton, "Impulse Noise Performance of an Asymmetric Digital Subscriber Lines Passband Transmission System", *IEEE Trans. On Commun.*, Vol. 43, No. 2/3/4, Feb./Mar./Apr., 1995.
 - [42] W. Henkel, T. Kebler, "A Wide-band Impulsive Noise Survey in the German Telephone Network: Statistical Description and Modeling", *AEU*, Vol. 48, No. 6, pp. 277-288, May 1994.
 - [43] J. W. Cook, "Wideband Impulsive Noise Survey of the Access Network", *BT Tech. J.*, vol. 11, No. 3, July 1993.
 - [44] J. J. Werner, "Impulse Noise in the Loop Plant", *ICC'90*, Session 348.1.
 - [45] C. F. Valenti and K. Kerpez, "Analysis of Wideband Noise Measurements and Implications for Signal Processing in ADSL Systems", *ICC'94*, Vol. 2, pp. 826-832.
 - [46] NYNEX Corp., "Characteristics of Impulse Noise on Selected NYNEX Metropolitan Loops," *ECSA Contribution T1D1.3/86-144*.
 - [47] K. Szechenyi, "On the NEXT and Impulse Noise Properties of Subscriber Loops," *Conference Record GLOBECOM'89*, pp. 1569-1573, Dallas, Nov. 27-30, 1989.
 - [48] R. M. Fano, "A Theory of Impulse Noise in Telephone Networks," *IEEE Trans. Commun.*, vol. COM-25, pp. 577-588, 1977.
 - [49] L. Zhang, "Error Control Coding in ADSL DMT Systems," Master's thesis, University of Ottawa, Sept. 1998.
 - [50] Y. Kofman, E. Zehavi, S. Shamai (Shitz): "A Multilevel coded modulation scheme for fading channels," *IEEE Trans. on Inform. Theory*, Vol. 42, 1994, pp. 299-312.
 - [51] W. E. Ryan, "A Turbo Code Tutorial", New Mexico State University, Box 30001 Dept. 3-O, Las Cruces, NM 88003.

- [52] P. Robertson, "Illuminating the Structure of Code and Decoder of parallel Concatenated Recursive Systematic (Turbo) Codes", Globecom'94, pp. 1298-1303.
- [53] C. Heegard, S. B. Wicker, "Turbo Coding," Kluwer Academic Publishers, Boston/Dordrecht/London, 1999.
- [54] J. Hokfelt, O. Edfors and T. Maseng, "Interleaver design for turbo codes based on the performance of iterative decoding," ICC'99, pp. 93-97.
- [55] S. Dolinar and D. Divsalar, "Weight Distributions for Turbo Codes Using Random and Nonrandom Permutation," TDA progress rep. 42-122, Jet Propulsion Lab., Pasadena, CA, Aug. 15, 1995, pp. 56-65.
- [56] O. Y. Takeshita, O. M. Collins, P. C. Massey, and D. J. Costello, Jr., "A Note on Asymmetric Turbo-Codes," IEEE Commun. Letters, Vol. 3, No. 3, pp. 69-71, March 1999.
- [57] J. Yuan, B. Vucetic, and W. Feng, "Combined Turbo Codes and Interleaver Design," IEEE Trans. on Commun., vol. 47, No. 4, pp. 484-487, April 1999.
- [58] P. L. McAdam, L. Welch, and C. Weber, "M.A.P. bit decoding of convolutional codes," in Abstracts of Papers, Int. Sump. On Information Theory, (Asilomar, CA, Jan. 1972), p. 91.
- [59] B. Sklar, "A Primer on Turbo Code Concepts", IEEE Comm. Magazine, pp. 94-101, Dec. 1997.
- [60] L. Lin and R. S. Cheng, "Improvements in SOVA-based decoding for turbo codes," ICC'97, pp. 1473-1478, June 1997.
- [61] M. P. C. Fossorier, F. Burkert, S. Lin, and J. Hagenauer, "On the Equivalence Between SOVA and Max-Log-MAP Decodings," IEEE Commun. Letters, Vol. 2, No. 5, May 1998.
- [62] J. Hagenauer, E. Offer and L. Papke, "Iterative Decoding of Binary Block and Convolutional Codes," IEEE Trans. On Inform. Theory, VOL. 42, No. 2, March 1996, pp. 429-445.