Investigation of Non-Binary Trellis Codes Designed for Impulsive Noise Environments

Wael Shalan Abd Alaziz

Newcastle University

Newcastle Upon Tyne, UK

A thesis submitted for the degree of

Doctor of Philosophy

March 2018

To my loving parents and wife.

Declaration

NEWCASTLE UNIVERSITY

SCHOOL OF ELECTRICAL AND ELECTRONIC ENGINEERING

I, Wael Shalan Abd Alaziz, declare that this thesis is my own work and it has not been previously submitted, either by me or by anyone else, for a degree or diploma at any educational institute, school or university. To the best of my knowledge, this thesis does not contain any previously published work, except where another person's work used has been cited and included in the list of references.

Signature:

Student: Wael Shalan Abd Alaziz

Date:

SUPERVISOR'S CERTIFICATE

This is to certify that the entitled thesis "Investigation of Non-Binary Trellis Codes Designed for Impulsive Noise Environments" has been prepared under my supervision at the school of Electrical and Electronic Engineering / Newcastle University for the degree of PhD in Electrical and Electronic Engineering.

Signature:

Supervisor: Dr. Martin Johnston

Date:

Acknowledgements

Gratitude to the created is gratitude to the creator. Man does not forget where he comes from, nor does he forget the ones who were by his side throughout the turbulences of time. This thesis would not have been possible without the direction, leadership, and kind nature of Dr. Martin Johnston who was truly the light guiding the way for me during my years at Newcastle university. Not only did he offer me a path on how to conduct my thesis, he also drew a roadmap on how to understand myself. I am indebted to him and always will be. I cannot forget Dr. Stéphane Le Goff who with the slightest touches enhanced the thesis ensuring it reaches the necessary standards.

On my PhD journey I was also blessed with the support of one who was more than a friend, Khaled M. Rabie, research assistant at the school of engineering in Manchester Metropolitan university. He was there for me like a brother during the ups an downs and encouraged me never to give up. Zhen Mei, my colleague at the university of Newcastle, was not only a confidant, but also a consultant on the challenges I faced during my project. My dear friend Omar Hussein Harran who accepted me with open arms when I first arrived in Newcastle and did everything for me to settle in, is the type of man I can truly say I was privileged to know as his selfless character was legendary. I thank Iman Ahmadian who through his kind humour made the darkest hours of my journey into drops of sunlight where we all smiled wishing the best for one another.

Under no circumstance would I ever come this far in my education without my loving wife who believed in me and my parents who demanded me to accomplish as much as I possibly could. They were the fuel which kept my engine going throughout my PhD journey. Hassan and Ali, my wonderful sons who were born during my studies here at Newcastle gave

me hope for life and made my efforts at success much stronger. I owe all of them and hope to make them proud in the years to come.

I also feel obligated to express my sincere gratitude to the University of Thi-Qar for choosing me for their PhD scholarship and the Iraqi Ministry of Higher Education and Scientific Research (MOHESR) for granting me the funds to study in UK. In addition, there was ample support towards my stay in the UK from the Iraqi Cultural Attaché in London whom I give thanks for.

Abstract

It is well known that binary codes with iterative decoders can achieve near Shannon limit performance on the additive white Gaussian noise (AWGN) channel, but their performance on more realistic wired or wireless channels can become degraded due to the presence of burst errors or impulsive noise. In such extreme environments, error correction alone cannot combat the serious effect of the channel and must be combined with the signal processing techniques such as channel estimation, channel equalisation and orthogonal frequency division multiplexing (OFDM). However, even after the received signal has been processed, it can still contain burst errors, or the noise present in the signal maybe non Gaussian. In these cases, popular binary coding schemes such as Low-Density Parity-Check (LDPC) or turbo codes may not perform optimally, resulting in the degradation of performance. Nevertheless, there is still scope for the design of new non-binary codes that are more suitable for these environments, allowing us to achieve further gains in performance. In this thesis, an investigation into good non-binary trellis error-correcting codes and advanced noise reduction techniques has been carried out with the aim of enhancing the performance of wired and wireless communication networks in different extreme environments. These environments include, urban, indoor, pedestrian, underwater, and powerline communication (PLC). This work includes an examination of the performance of non-binary trellis codes in harsh scenarios such as underwater communications when the noise channel is additive $S\alpha S$ noise. Similar work was also conducted for single input single output (SISO) power line communication systems for single carrier (SC) and multi carrier (MC) over realistic multi-path frequency selective channels. A further examination of multi-input multi-output (MIMO) wired and wireless systems on Middleton class A noise channel was carried out. The main focus of the

project was non-binary coding schemes as it is well-known that they outperform their binary counterparts when the channel is bursty. However, few studies have investigated non-binary codes for other environments. The major novelty of this work is the comparison of the performance of non-binary trellis codes with binary trellis codes in various scenarios, leading to the conclusion that non-binary codes are, in most cases, superior in performance to binary codes. Furthermore, the theoretical bounds of SISO and MIMO binary and non-binary convolutional coded OFDM-PLC systems have been investigated for the first time. In order to validate our results, the implementation of simulated and theoretical results have been obtained for different values of noise parameters and on different PLC channels. The results show a strong agreement between the simulated and theoretical analysis for all cases.

Contents

List of Figures

List of Tables

Nomenclature

Symbols

α	Characteristic exponent of α-stable process
α_i	The forward recursion parameter of BCJR algorithm
β	Skewness of α-stable pdf
β_i	The backward recursion parameter of BCJR algorithm
δ	Location parameter of α-stable process
Γ	The ratio of Gaussian to impulsive noise power
γ^α	Dispersion of α-stable process
γ_i	The state transition parameter of BCJR algorithm
μ	The mean of the normal distribution
σ^2, σ_G	The variance of the Gaussian noise
σ_i^2	The variance of the impulsive noise
σ_m^2	The variance of Middleton class A noise
A	The index of Middleton class A noise
B	The channel bandwidth
C_g	The exponential of the Euler constant
d_{free}	The minimum free distance
$E\{\cdot\}$	Expectation Operator
E_b	The energy per bit

E_s	The energy per symbol
N_0	Noise power
$Q(x)$	Q-function of x
R_c	Code rate
$\mathrm{SNR_G}$	Geometric signal-to-noise ratio

Acronyms/Abbreviations

APP	A posterior probability
AWGN	Additive white Gaussian noise
BCJR	Bahl-Cocke-Jelinek-Raviv algorithm
BER	Bit error rate
BI-AWGN	Binary input-AWGN
BPSK	Binary phase-shift keying
BSC	Binary symmetric channel
EMI	Electromagnetic interference
FEC	Forward error correction
FFT	Fast Fourier transform
GF	Galois field
IFT	Inverse Fourier transform
LDPC	Low-density parity-check
LLR	Log-likelihood ratio
MAP	Maximum a posterior probability
MC	Multi-carrier
MIMO	Multi-input multi-output
MISO	Multi-input single-output

ML	Maximum likelihood
MLD	Maximum likelihood detector
MMSE	Maximum mean square error
M-PSK	Multi-level-PSK
MSE	Mean square error
OFDM	Orthogonal frequency-division multiplexing
PDF	Probability density function
PLC	Power line communication
PSK	Phase shift-keying
QAM	Quadrature amplitude modulation
QPSK	Quadrature phase shift-keying
RSC	Recursive systematic convolutional code
SC	Single carrier
SER	Symbol error rate
SIMO	Single-input multi-output
SNR	Signal to noise ratio
SOVA	Soft output Viterbi algorithm
SαS	Symmetric α-stable
ZF	Zero forcing

Chapter 1

Introduction

1.1 Introduction

Error control coding is essential to achieve reliable digital communication and many different coding schemes have been proposed over the years: Ring-TCM (trellis coded modulation), low-density parity-check (LDPC) codes, Turbo codes, Algebraic-Geometric codes, etc. [1–4]. LDPC and turbo codes are now commonly used in modern communication systems due to their capacity-approaching performance on the additive white Gaussian noise (AWGN) channel, but their performance is degraded when the channel is more severe. When combined with advanced signal processing techniques, LDPC and turbo code performance can be improved, but there can still be burst errors present or the channel can become non-Gaussian that binary codes have difficulty correcting. Therefore, there is a case for employing non-binary codes as a replacement for their binary counterparts. In this thesis the consideration of non-Gaussian channels will be the main focus, to evaluate the efficiency of non-binary trellis codes in harsh environments.

The design of non-binary error-correcting codes is a niche research area, but there are several advantages to employing them, such as improved performance, a larger selection of codes and robustness to burst errors. There is great scope for producing novel work in this area by also combining non-binary code design with signal processing techniques used in wireless communications, such as equalization, OFDM, channel estimation and MIMO. In this project, good non-binary codes will be designed for harsh environments in combination with these signal processing techniques to improve performance. At the same time, there will be an inevitable

increase in complexity which will also be addressed in this project.

1.2 Motivation and Challenges

Due to the serious impact caused by impulsive noise on the performance of communication systems, there have been various well known channel models proposed to model impulsive noise distributions. These models include Middleton class A, Gaussian-mixture and symmetric alpha stable ($S\alpha S$) distributions [5–7]. They can all model noise channels on power line communication (PLC), under water and in-home telecommunications services. Unlike AWGN which has a closed-form expression, some of impulsive noise models are not closed form. In this case, evaluating a tight bound to the bit error rate (BER) performance on these channels is very challenging. In addition, error correcting codes alone cannot combat the negative effects of impulsive noise on the performance of any communication systems.

In order to reduce the impact of impulsive noise, we employ signal processing techniques along with non-binary turbo codes, such as; orthogonal frequency-division multiplexing (OFDM), zero forcing detector (ZF), minimum mean square error equalizer (MMSE), blanking and clipping. These proposed systems have been examined on very harsh, impulsive noise environments and for various wired and wireless telecommunication applications.

1.3 Aims and Objectives

The main objective of this thesis is to design good non-binary error-correcting codes to reduce the impact of impulsive noise channels on wired and wireless medium. These non-binary codes plus signal processing techniques have been implemented to enhance the BER performance of various telecommunication applications. To reach these targets it is necessary to accomplish other intermediate objectives:

- To conduct a deep review into the state of the art of non-binary coding schemes.
- To study the different existing decoding algorithms and investigate methodologies to make them appropriate for non-Gaussian channels.

- To investigate non-binary trellis code performance with a Cauchy receiver for the case of $S\alpha S$ noise channel.

- To introduce a theoretical analysis of the BER performance of non-binary coded systems on impulsive noise channel.

- To examine non-binary trellis codes for recent applications such as multiple-input multiple-output (MIMO) on Middleton class A noise and an in-home PLC services.

1.4 Statement of Originality

The thesis contributions are mainly focused on the BER performance of uncoded systems and coded systems based on binary or non-binary trellis codes in the presence of impulsive noise. The novelty of the thesis is illustrates as follows:

- In chapter 3, the error probability of non-binary convolutional codes on the AWGN channel has been introduced. The BER performance of the $(1, 7/5)_8$ binary and $\beta\beta^2/1$ and $\beta 1\beta/\beta^2 1$ non-binary convolutional codes on $S\alpha S$ channel are presented for various values of α. Furthermore, non-binary turbo coded communication systems on $S\alpha S$ channels have been implemented and a fair comparison with binary turbo coded system on the same environments is shown. Ultimately, the appropriate modifications to the decoding algorithms of binary and non-binary trellis codes have been carried out for the Cauchy receiver.

- In chapter 4, the performance of single carrier (SC) non-binary trellis codes on Middleton class A noise channel is presented. The BER performance analysis of binary and non-binary convolutional codes on multi-carrier (MC) communication system on impulsive noise is then introduced. A performance analysis of binary and non-binary convolutional codes on realistic PLC multi path frequency-selective channels with ZF equalizer is derived for the first time. Again, a modification is made to the conventional decoding algorithm of binary and non-binary trellis codes to make them tailored to PLC receivers. In addition, a fair comparison between the performance of binary and non-binary

trellis coded OFDM-PLC systems has been presented. Finally, noise cancellation or reduction such as blanking and clipping have been utilized with binary and non-binary turbo coded OFDM-PLC systems.

- In chapter 5, the BER performance analysis of binary and non-binary convolutional coded OFDM-MIMO in the presence of impulsive noise for wireless applications has been presented. Also, the synthetic statistical MIMO-PLC channel has been employed for the first time to examine our proposed system on semi-realistic PLC-MIMO environments. Finally, to reduce the effect of the multi path frequency selective and impulsive noise channels, we have utilized signal processing techniques such as OFDM, MMSE, blanking and clipping.

1.5 Organization of the Thesis

The thesis is organized as follows:

Chapter 2 presents the theoretical background and the relevant studies on impulsive noise channels and their applications.

Chapter 3 introduces two classes of non-binary convolutional code and their construction. Also, the flow-graph of $\beta\beta^2/1$ non binary codes is explained in detail. In addition, the decoding algorithm (BCJR) which has been employed in this chapter is presented with the appropriate modifications that make it suitable for the Cauchy receiver. Simulation results of the BER performance of binary and non-binary convolutional codes, binary and non binary turbo codes and uncoded systems on $S\alpha S$ noise channels are presented.

Chapter 4, gives an overview of the Middleton class A noise channel and the BER performance of SC coded binary and non-binary convolutional systems on additive Middleton class A noise is presented. Also, realistic multi-path frequency-selective channels used in power line communication have been explained in detail and the performance analysis of binary and non-binary convolutional coded OFDM-PLC systems has been introduced. The encoding and decoding procedures for non-binary turbo codes on the Middleton class A noise channel have been illustrated. Furthermore, in this chapter the non-binary turbo coded OFDM-PLC system with non-linear pre-processing has been investigated. Simulation results of the BER performance of binary and non-binary turbo coded OFDM-PLC systems for various

values of Γs and on different realistic PLC channels have been presented.

Chapter 5, presents the implementation of coded and uncoded OFDM-MIMO systems for wireless applications. In addition, the analysis of uncoded OFDM-MIMO systems on the Middleton class A noise channel is carried out. Also, the simulated BER performance of binary and non-binary turbo codes have been investigated. The performance analysis of wireless OFDM-MIMO on impulsive noise channels with a comparison to the simulated performance has been introduced in this chapter. Furthermore, an investigation into wired applications of binary and non-binary trellis codes on PLC MIMO channels have been carried out. Moreover, the synthetic MIMO PLC channel has been described in detail and the non-binary turbo coded MIMO PLC-OFDM transmitter and receiver have been introduced. Finally, the numerical and simulation results of binary and non-binary trellis coded OFDM-PLC-MIMO have been presented.

Chapter 6 presents the thesis conclusions and suggestions for future research in this field.

1.6 Publications Related to the Thesis

1.6.1 Published

1. Abd-Alaziz, Wael, Martin Johnston, and Stephane Le Goff. "Non-binary turbo codes on additive impulsive noise channels." Communication Systems, Networks and Digital Signal Processing (CSNDSP), 2016 10th International Symposium on. IEEE, 2016.

2. Abd-Alaziz, Wael, et al. "Non-binary turbo-coded OFDM-PLC system in the presence of impulsive noise." Signal Processing Conference (EUSIPCO), 2017 25th European. IEEE, 2017.

3. Abd-Alaziz, Wael, Martin Johnston, and Stephane Le Goff, "Non-Binary Trellis Codes on the Synthetic Statistical MIMO Power Line Channel," in 2018 IEEE International Symposium on power Line Communications and its Applications (ISPLC), April 2018, pp.1-5.

Chapter 2

Theoretical Background and Literature Survey

2.1 Introduction

In this chapter, the necessary prerequisite information for channel modelling, binary codes, non-binary fields, PLC and MIMO systems, will be presented. However, various kinds of noise models will be illustrated as a preface to these. Obviously, not all communication channels will be covered, but it will include the related additive noise channels for the proposed communication systems in this thesis. These related channels are AWGN, alpha stable $S\alpha S$ and Middleton class A. Furthermore, a brief overview on the structure of the decoder and the encoder of convolutional and turbo coding schemes on AWGN is included. Finite fields will be explained briefly in order to understand the construction, encoding and decoding of non-binary codes. To understand chapter 6, MIMO systems will be explained and various channel estimation techniques will be presented. This is followed by a review of the available literature on important and current research in these areas.

2.2 Channel Model

2.2.1 Additive White Gaussian Noise Channel

AWGN is a typical noise model that models the impact of random processes in nature or by humans [8, 9]. Some causal processes that generate AWGN are: electromagnetic fields from various sources, the motion of atoms in any conductors, the

earth, and any warm objects. AWGN follows a normal distribution (Gaussian) and it can be defined as:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{2.1}$$

where x is the random variable (symbols), μ is the mean value, and σ is the standard deviation. Mostly in conventional communication systems the mean of the AWGN distribution is zero and with variance σ^2 and it's written as $\mathcal{N}(0, \sigma^2)$. The variance σ^2 of AWGN in communication systems that is utilizing BPSK modulation, calculated as

$$\sigma^2 = \frac{N_0}{2} \tag{2.2}$$

where N_0 the noise power spectral density.

Figure 2.1 shows the probability density function (PDF) of Gaussian distribution vs random variables, for different σ values.

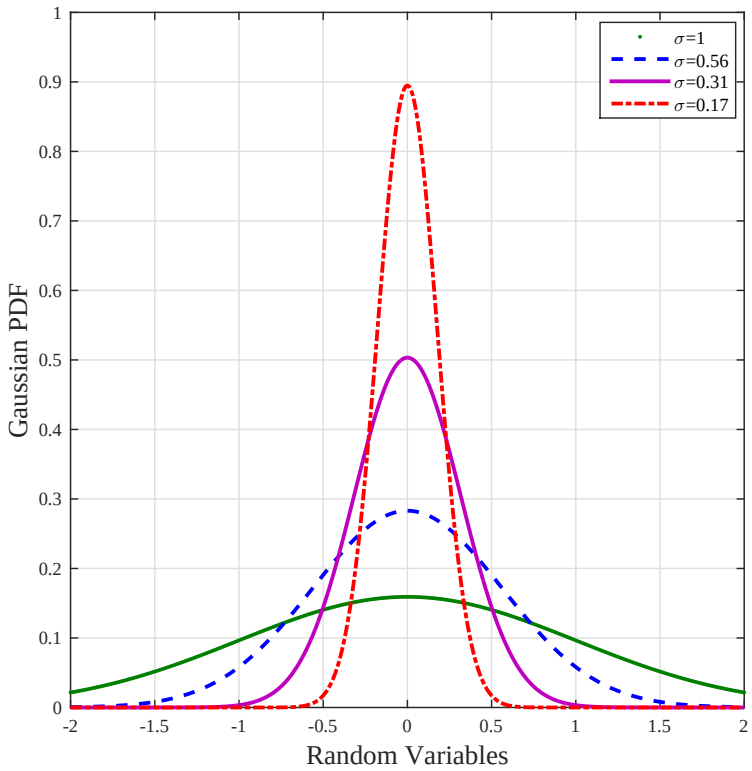

Figure 2.1: PDF of AWGN $(0, \sigma)$.

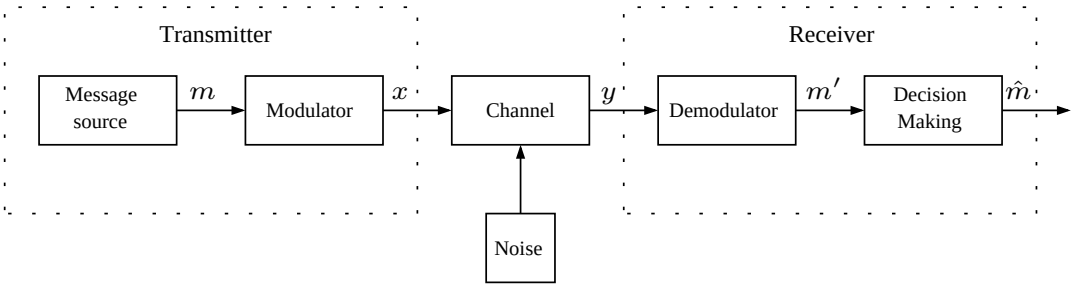

Figure 2.2: Conventional information systems.

In order to demonstrate AWGN channel on an ordinary communication system, figure 2.2, illustrates a general block diagram of a typical communication system. A message source generates a stream of bits and these bits will pass into a modulator. There are many analogue and digital modulation schemes, but in this thesis, binary phase shift keying (BPSK) and quadrature phase shift keying (QPSK) are mainly utilized for our proposed systems [10, 11]. Next, the mapped symbols will be transmitted via a communication medium which can be a wired or wireless channel, and the noise will be added at the receiver.

Generally, the received signal is defined as

$$y = x + n \tag{2.3}$$

where y is the received signal, x is the transmitted and n is the noise. While in the receiver side, the demodulation technique will be applied first and then the decision will be made to estimate the transmitted message $\hat{m}$.

The theoretical symbol-error rate (SER) performance analysis of this typical system for M-level PAM modulation can be defined as [12]

$$P_M = \frac{2(M-1)}{M} Q\left(\sqrt{\frac{6(\log_2 M)E_b}{(M^2-1)N_0}}\right) \tag{2.4}$$

where M is the modulation order and can take values 2, 4, 8 ,16, ...etc, and Q can be evaluated by

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty \exp -\frac{\mu^2}{2} d\mu \tag{2.5}$$

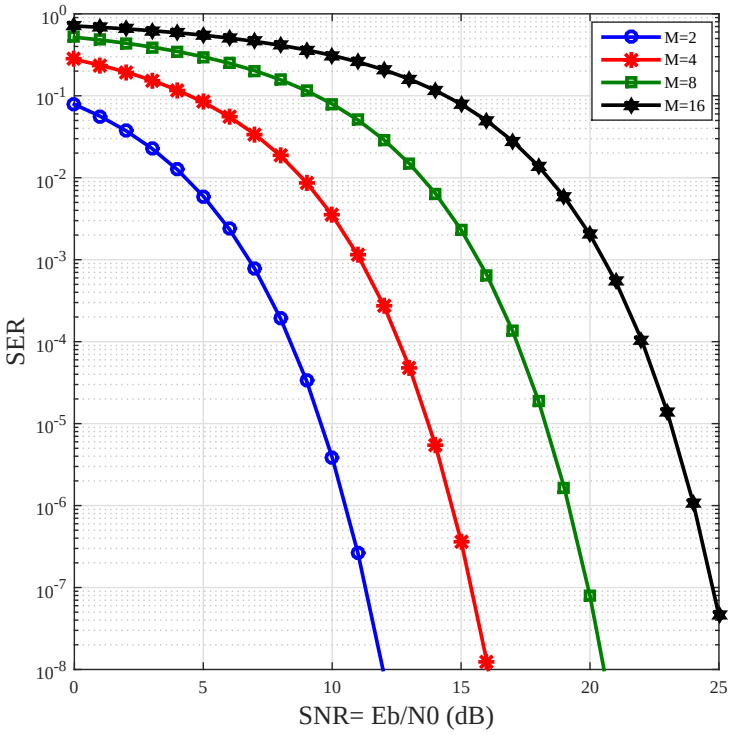

Figure 2.3: SER for M-level PAM on AWGN channel.

Figure 2.3 illustrates the SER performance for various modulation orders as a function of SNR.

2.2.2 Symmetric Alpha-Stable Distributions ($S\alpha S$)

Symmetric alpha-stable distributions are commonly used to model heavy-tailed distributions (i.e. the PDF tails are not exponentially bounded) [13]. Their probability density functions are defined as

$$f_\alpha(x) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \exp(j\delta t - \gamma^\alpha |t|^\alpha) e^{-itx} dt, \tag{2.6}$$

where α is the characteristic exponent which increases the impulsive behaviour as it becomes smaller, with a range (0,2]. γ is the dispersion parameter which measures the spread of the $S\alpha S$ pdf. δ is the mean or the median of an $S\alpha S$ pdf and depends on the value of α, because the stable distribution has undefined mean for $\alpha \leq 1$. Figure 2.4 illustrates $S\alpha S$ pdfs for different values of α and shows how the impulsivity nature of the distribution increases as α decreases. One way to obtain a reliable receiver for impulsive noise channels is to study the original $S\alpha S$ pdf in

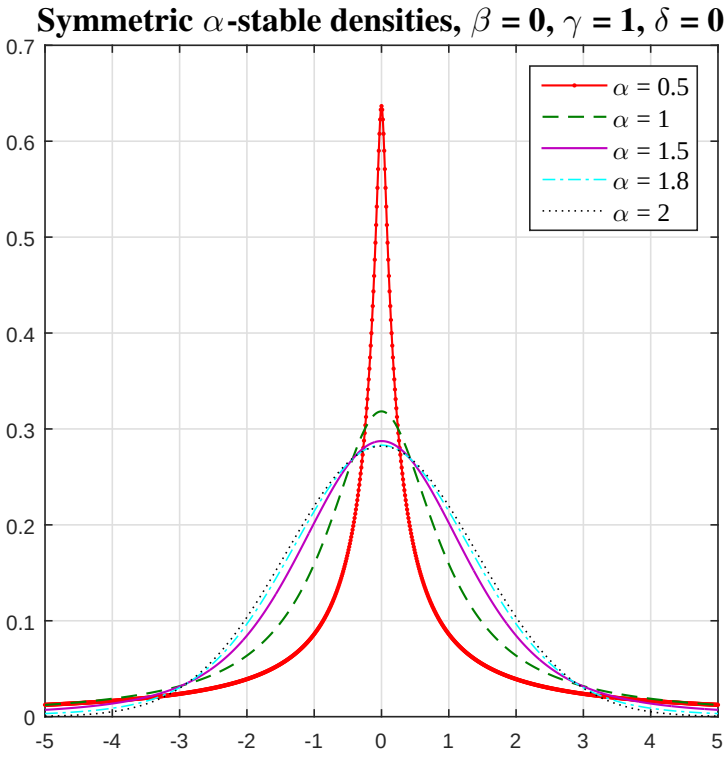

Figure 2.4: Standard $S\alpha S$ distributions($\gamma = 1$, $\delta = 0$).

(2.6) for certain values of α. Depending on α there are some special cases of $S\alpha S$ pdfs that have closed form expressions: These are the Gaussian distribution when $\alpha = 2$ and the Cauchy distribution when $\alpha = 1$. In this thesis the Cauchy receiver will be implemented for both binary and non-binary turbo codes since this has been shown to be a good receiver for impulsive noise channels for a range of α values [14]. The pdf of a Cauchy distribution is defined as

$$f_1(x) = \frac{\gamma}{\pi[\gamma^2 + (x-u)^2]}. \tag{2.7}$$

Where u is the mapping symbols and for BI-AWGN $u \in \{-1, +1\}$. As we can see in (2.7) we must know the dispersion parameter γ which is given by

$$\gamma = \sqrt{\frac{A^2}{4C_g^{(\frac{2}{\alpha}-1)}(\frac{E_b}{N_0})}} \tag{2.8}$$

In equation (2.8) $C_g \approx 1.78$, which is the exponential of the Euler constant and A is the peak amplitude of the transmitted signal. $\frac{E_b}{N_0}$ can be defined in terms of

the geometric SNR and code rate R_c, which for BPSK modulation is [13],

$$\frac{E_b}{N_0} = \frac{SNR_G}{2R_c} = \frac{1}{4R_cC_g}(\frac{A}{S_0})^2, \tag{2.9}$$

where S_0 is the geometric noise power of an $S\alpha S$ distribution given by

$$S_0 = \frac{(C_g)^{1/\alpha}\gamma}{C_g}. \tag{2.10}$$

Henc, the difference between the Gaussian noise power N_0 and the geometric $S\alpha S$ noise power S_0 is that the non-Gaussian distributions have infinite variance. In other words, the conventional definition of the SNR based on second order statistics is no longer applicable.

2.2.2.1 Generation of Alpha Stable $S\alpha S$ Random Variables

The generation of $S\alpha S$ variables is described in [14, 15] and the procedure of this generation will be explained in this section. Let U be a uniform random variable in $\left(\frac{-\pi}{2}, \frac{\pi}{2}\right)$, and W is the standard exponential. In order to generate U and W, we need to introduce two uniformly distributed samples u_1 and u_2 in (0,1). Then U and W can be evaluated, respectively, as

$$U = \pi(u_1 - 0.5) \tag{2.11}$$

$$W = \ln(u_2) \tag{2.12}$$

Thus, the $S\alpha S$ random variable Z can be generated as:

$$Z = \gamma\frac{\sin(\alpha U)}{(\cos(U))^{\frac{1}{\alpha}}}\left[\frac{\cos[(1-\alpha)U]}{W}\right]^{\frac{1-\alpha}{\alpha}} \tag{2.13}$$

while, when α=1

$$Z = \gamma\tan(U) \tag{2.14}$$

For clarification and comparison purposes, figure 2.5 shows that as α becomes smaller, the density and the strength of the deviation increase to an extremely impulsive behaviour. Figure 2.5 displays the behaviour of the noise from Gaussian when $\alpha = 2$ and to quite impulsive when $\alpha = 0.5$.

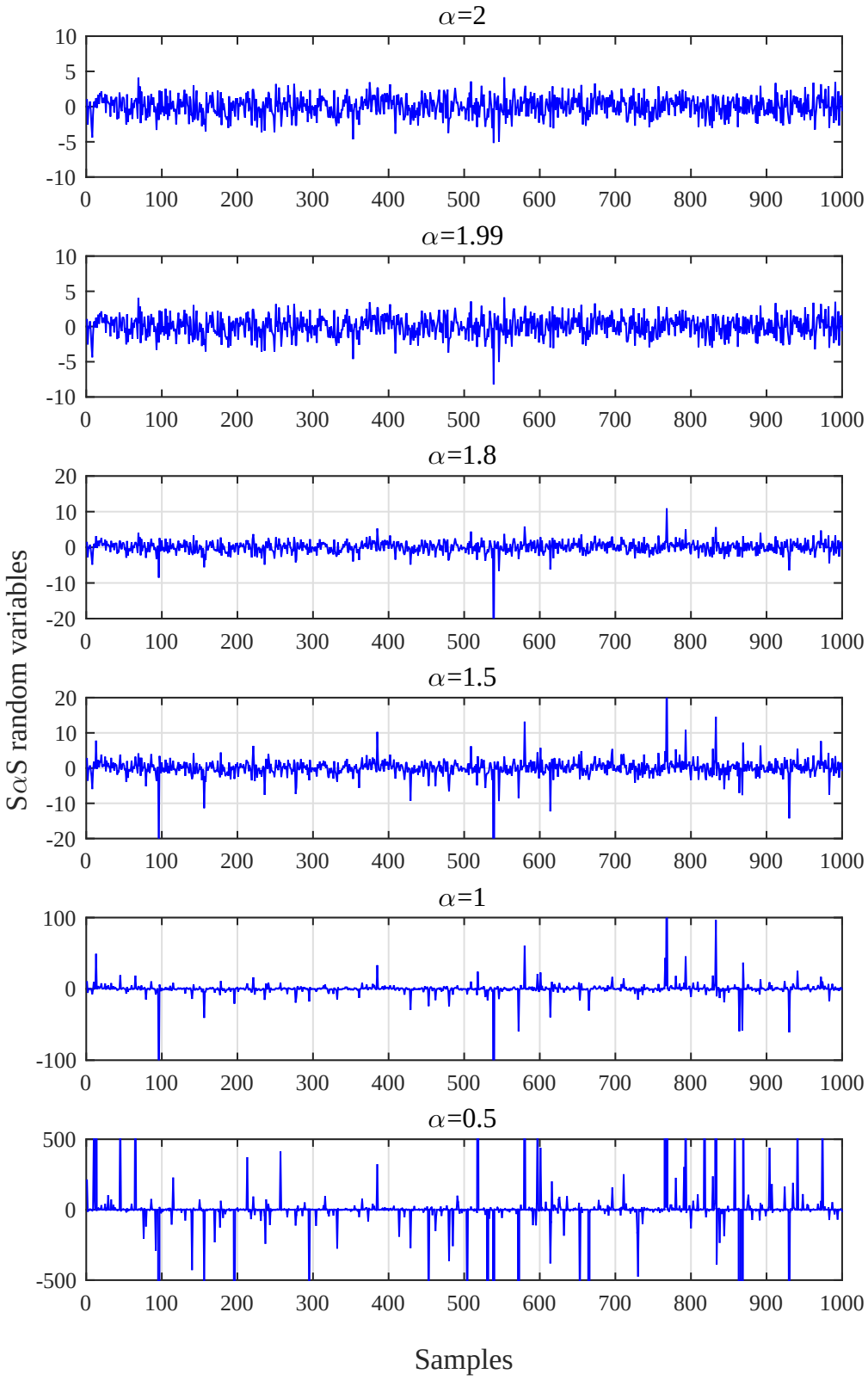

Figure 2.5: Simulation of 1000 samples of independent $S\alpha S$ stable noise,($\gamma = 1$).

2.2.2.2 BER Performance Analysis of Uncoded Systems on $S\alpha S$ Noise Channel

To illustrate the $S\alpha S$ noise impact on the communication systems which is displayed in figure 2.2, Z in equations (2.14) and (2.13) will be substituted into equation (2.3) instead of n, when γ is calculated from equation (2.8). Then a hard decision will be taken on the demodulated received signal to extract the simulated results.

The analytic BER performance of BPSK uncoded system was defined in [16] as

$$P_e = Q_\alpha \left(\sqrt{4R_c C_g^{\frac{2}{\alpha}-1} \frac{E_b}{N_0}} \right) \tag{2.15}$$

where $R_c = 1$ for coded systems.

Ultimately, figure 2.6 displays the comparison between the simulated and theoretical BER performance of an uncoded communication system on $S\alpha A$ stable channel for different values of α.

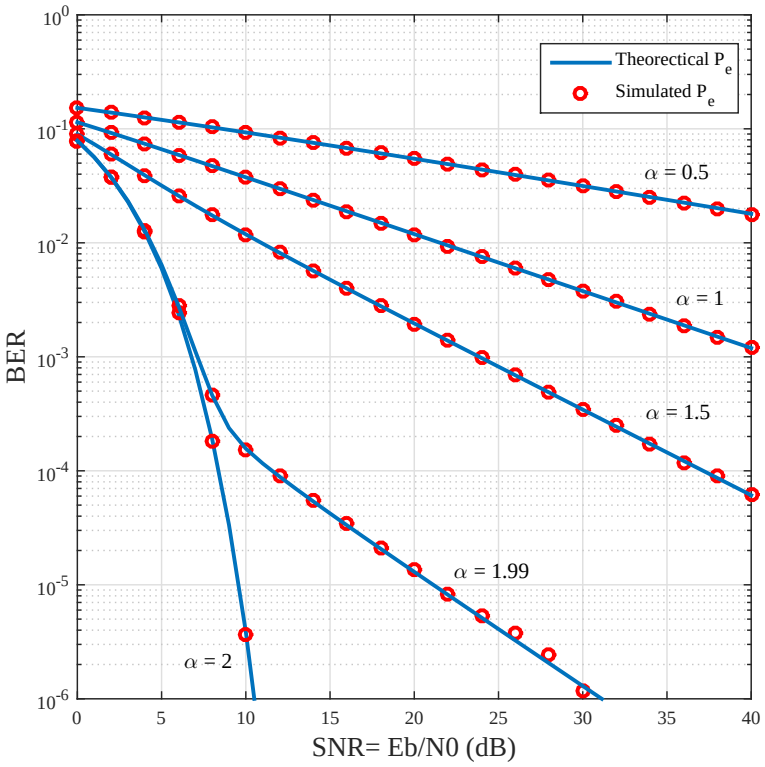

Figure 2.6: Simulated and theoretical BER performance of uncoded system on $S\alpha S$ noise channel for various values of α.

2.2.3 Middleton Class A Distributions

Middleton developed three statistical noise models for non-Gaussian channels. These models are; class A, which represents narrowband noise, class B, that describes broadband noise and class C, which is the sum of class A and B models. Since Middleton class A is well known to mimic the channel behaviour in a power line environment, it will be the only considered model in this section. Middleton class A was first introduced in [5, 17].

Middleton class A distributions are commonly used to model the impulsive noise of power-line channels [5] and their PDF is defined as

$$p(X) = \sum_{m=0}^{\infty} \frac{e^{-A} A^m}{m!} . \frac{1}{\sqrt{2\pi\sigma_m^2}} \exp\left(-\frac{|X|^2}{2\sigma_m^2}\right), \tag{2.16}$$

where the variance σ_m^2 is given as

$$\sigma_m^2 = \sigma_u^2 \left(\frac{\frac{m}{A} + \Gamma}{1 + \Gamma}\right) \tag{2.17}$$

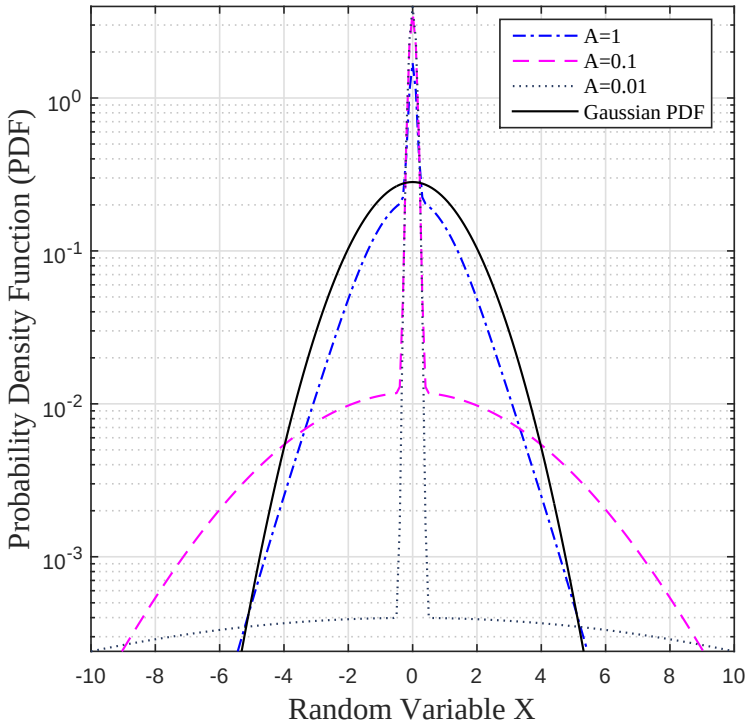

Figure 2.7: PDF Middleton class A distribution.

and

$$\sigma_u^2 = \sigma_G^2 + \sigma_I^2, \qquad \Gamma = \frac{\sigma_G^2}{\sigma_I^2}. \tag{2.18}$$

The parameters σ_G^2 and σ_I^2 are the variances of Gaussian noise and impulsive noise, respectively.

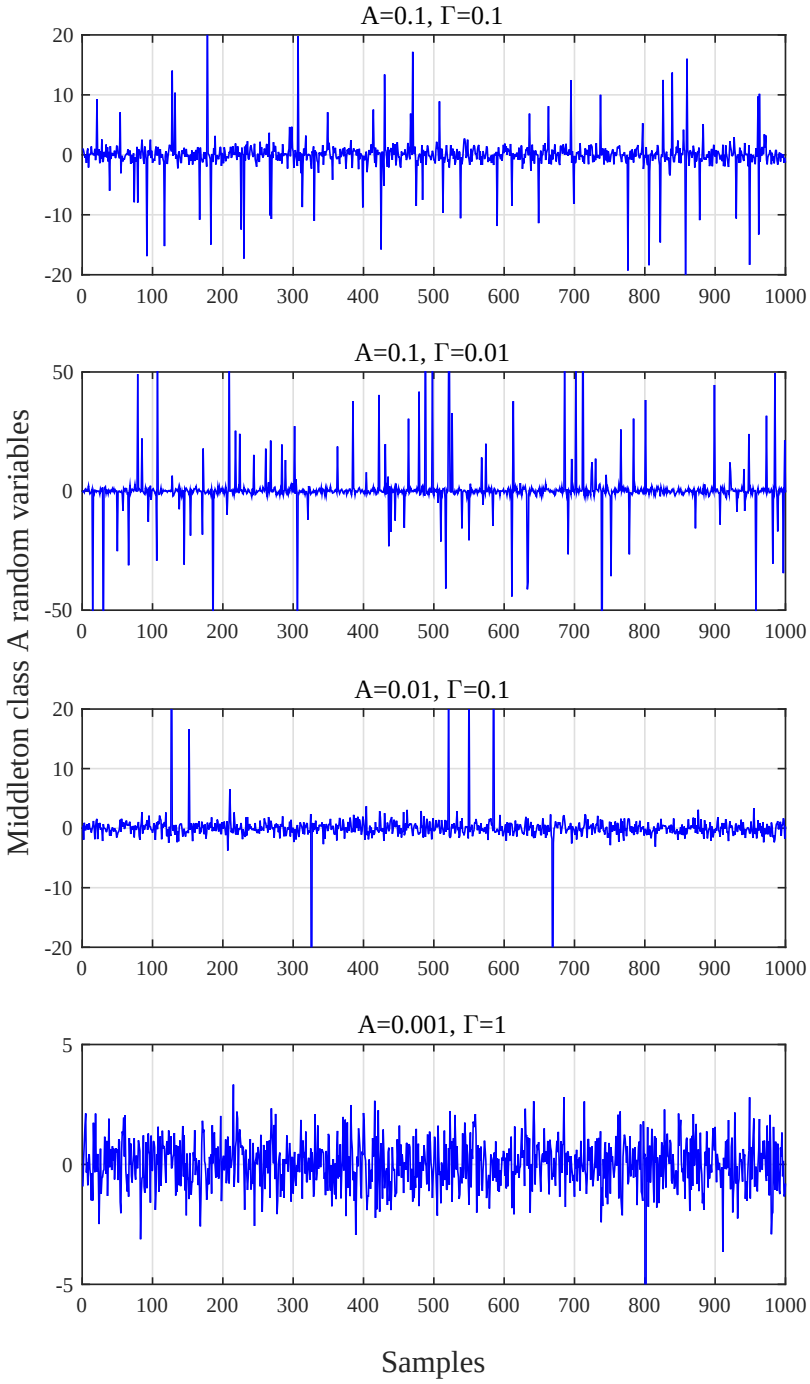

Figure 2.8: Simulation of 1000 samples of independent Middleton class A noise channel, for various values of Asand Γ.

Γ is the background to impulsive noise ratio parameter which indicates the strength of impulsive noise compared to Gaussian noise. A is the impulsive index which increases the impulsive behavior as it becomes larger that takes values $\leqslant 1$ and becomes Gaussian when A is small. Figure 2.7 illustrates the Middleton class A noise PDF vs random variables for various values of A and also the comparison between the AWGN and the impulsive PDFs.

For a further demonstration of Middleton class A parameters influence on the noise pattern, figure 2.8 shows Middleton class A random variables vs samples pattern for a few scenarios and for 10000 samples. For more detail about Middleton class A impulsive noise model, which is extensively studied and employed in the literature, please refer to chapter 4, chapter 5 and [6, 18–22].

2.2.4 Multi-Path Frequency Selective Channels

Figure 2.9 shows uncoded BPSK on a frequency-selective channel with AWGN and the output y is given by

$$y = h * x + n \tag{2.19}$$

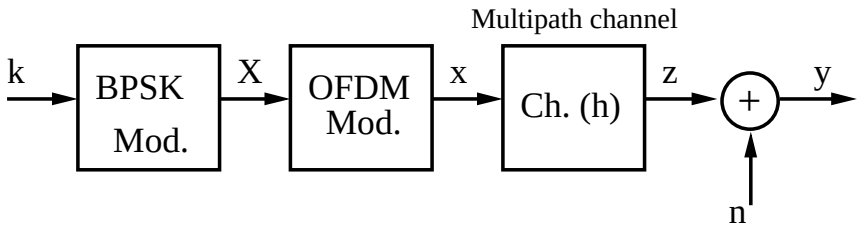

Figure 2.9: General diagram of uncoded BPSK modulation in frequency-selective channel in the presence of AWGN noise channel

where x is [$\pm$1] BPSK symbols, h is the frequency-selective channel coefficients and n is the AWGN samples.

Hence, there are several advantages of utilizing OFDM, mainly to eliminate the effect of multi-path channels and also it's suitability for high data rate communication services. OFDM is implementing by taking discrete Fourier transform (DFT) to block of K complex baseband symbols, and digital to analog (DA) conversion to the complex OFDM signals. In addition the cyclic prefix will be added to the OFDM signals as a guard interval to protect the signals from inter-symbol interference. see section 4.2.1.2 for more details.

However, according to [23] in chapter 11 (equation 11.1-10), the BER performance of this system can be estimated by

$$P_e = Q\left(\sqrt{2\gamma_b}\right) \tag{2.20}$$

where γ_b is the SNR per bit and is evaluated as

$$\gamma_b = \frac{Eb}{2\sigma_G^2} \sum_{n=1}^{N} |H(n)|^2 \tag{2.21}$$

Here N is the number of OFDM sub-carriers and H is the frequency response of the channel and can be evaluated by

$$H(n) = \sum_{l=1}^{L} h(l) e^{-2j\pi \frac{lk}{n}} \tag{2.22}$$

2.3 Channel Coding

Shannon in 1948 introduced that information could be encoded and decoded reliably in the presence of noise at any information rate less than the channel capacity [24]. He also described the existence of good codes, but he could not explain how to construct these codes. However, he opened the door for a large body of research on designing a good code [25, 26]. In 1950 R. W. Hamming, first presented the Hamming code from his research 'Error Detecting and Error Correcting Codes' [27]. The Hamming code is a linear error correcting code, it can detect up to two bits and correct one bit only. The main focus of the research in the period from early 1950's and late 1960's was developing efficient encoding and decoding approaches [28–37]. For instance, Reed and Muller presented their codes and decoder (Reed and Muller code) in 1954. Moreover, in 1955, Elias introduced the convolutional code in [28].

While in 1962 Gallager presented the Low-Density Parity-Check (LDPC) for first time in [35]. Today LDPC codes are iterative codes that can perform within 0.0045 dB away from Shannon limit. This optimum performance is a result of the sparseness of H (the parity check matrix) which can achieve large d_{free} and reduce the complexity of the decoding. Unfortunately, this invention was ignored until 1995 when Mackay rediscovered LDPC codes to create the powerful error-correcting code we use today [38].

In the 1970s, the ML and BCJR algorithms were introduced by Viterbi and Bahl [39] in [40] and respectively. Moreover, in 1981 Goppa introduced algebraic-geometric codes and after one year, Ungerboeck presented trellis coded modulation [4, 41]. Another iterative code was published under the title of 'Near Shannon limit error-correcting coding and decoding: Turbo-codes' in 1993 by C. Berrou [3]. This coding scheme with non-binary GF(q) will be the interest of this thesis. Non-binary turbo codes are formed from the parallel concatenation of two non-binary recursive systematic convolutional codes separated by an interleaver. The difference between non-binary and binary convolutional codes is the coefficients of their generator polynomials are now defined in a finite field GF(q), where $q = 2^p$ and p is an integer greater than 1. Non-binary convolutional codes can be decoded with a trellis-based decoder such as Viterbi's algorithm [42] and the BCJR algorithm [40], but the complexity is higher compared with binary decoders. However, despite the increase in complexity the advantages of non-binary turbo codes compared to binary turbo codes are: better convergence for iterative decoding, larger minimum Hamming distances, less sensitivity to puncturing patterns and robustness towards the flaws of the component decoding algorithm, in particular when the Maximum A Posterior algorithm is simplified to the Max-Log-MAP [43]. However, the latest coding scheme was the Polar code which was proposed by Arikan in 2009 [44]. Arikan describes Polar code as a class of capacity-achieving with a low encoding and decoding processes [45].

2.3.1 Channel Capacity

Error correcting codes allow errors in the received signal to be detected and/or corrected by a decoder in the receiver. Hence, in [24] Shannon proved that utilizing a proper decoding approach can effectively eliminate errors resulting from the noisy channels, given that the data rate is less that the channel capacity. Thus, the Shannon capacity limit in terms of the transmission rate R_c, the bit energy E_b and the noise power N_0, can be estimated by

$$C_{max} = \log_2\left(1 + \frac{R_c E_b}{N_0 B}\right) \tag{2.23}$$

where B is the bandwidth of the channel in Hz. This is can be approximated to get the minimum required SNR for reliable communications, and it is known as the

Shannon bound

$$\frac{E_b}{N_0} \geqslant \frac{2^{C_{max}} - 1}{R_c C_{max}} \tag{2.24}$$

For very large values of C_{max} or unlimited B *i.e.*, $C_{max} \longrightarrow 0$, then the minimum SNR threshold can be obtained as

$$\frac{E_b}{N_0} \geqslant \lim_{C_{max} \longrightarrow 0} \frac{2^{C_{max}} - 1}{C_{max}} = \ln(2) = -1.59dB \tag{2.25}$$

Figure 2.10 shows this limit, and also demonstrates the relation between Shannon limit with the code rate R_c [26, 41]. In essence, this curve shows the minimum required SNR to obtain error-free communications with a certain information rate R_c. For example, to achieve BER performance of 1×10^{-5} with a rate $R_c = \frac{1}{2}$ coded system on AWGN channel with BPSK modulation, the minimum SNR is 0.188 dB.

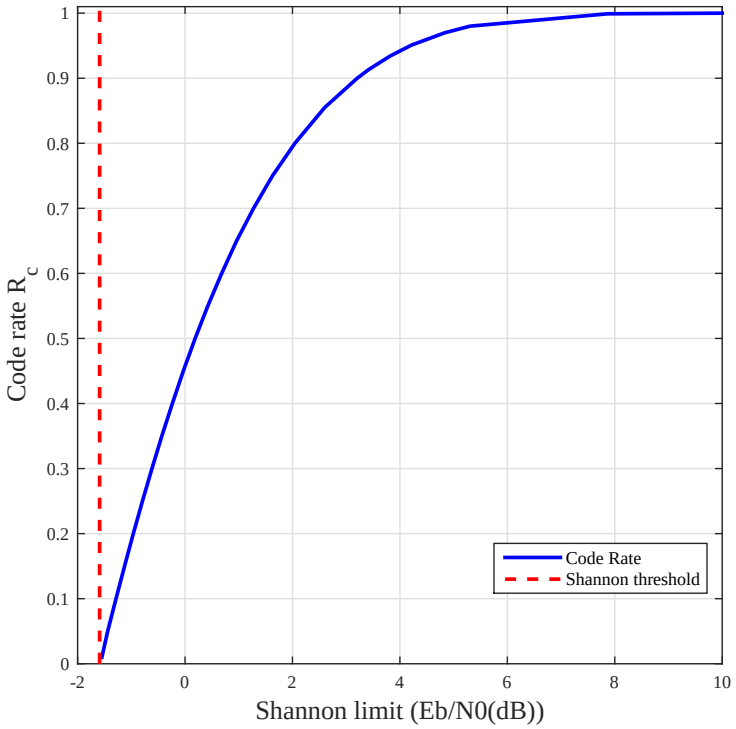

Figure 2.10: Shannon limit as a function of information rate code R_c.

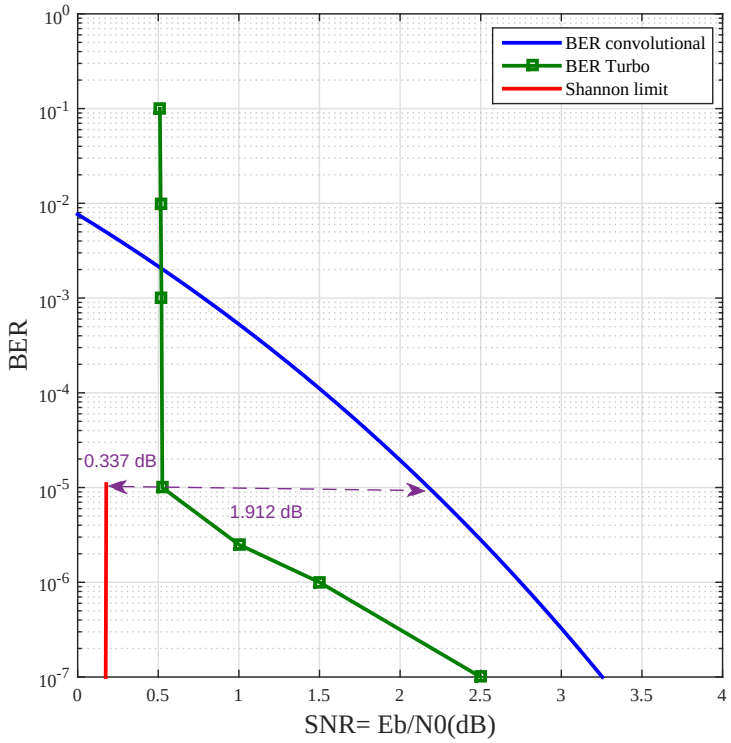

Figure 2.11: BER performance of information rate $\frac{1}{2}$ (63057, 44735) convolutional code and (37, 12, 65536) rate $\frac{1}{2}$ turbo code with 18 iterations.

Figure 2.11 displays the comparison between the Shannon limit for rate $\frac{1}{2}$ coded systems and a (63057, 44735) convolutional codes with rate $\frac{1}{2}$. It shows that at a BER of 10^{-5} the error performance of this code is 1.912 dB away from Shannon limit. In addition, this figure also shows the comparison between the optimal limit and the performance of a rate $\frac{1}{2}$ turbo code with 18 iterations, and the finding is that the E_b/N_0 required by this code to achieve a BER=10^{-5} is greater than the Shannon limit by 0.337 dB only. It can be concluded from this figure that the Shannon limit can be approached with a perfectly designed code with appropriate code length and optimal decoding algorithm [46].

However, there are many coding approaches, but the main interest of this thesis is on convolutional and turbo codes. In the next sections the theoretical background behind these coding schemes will be described.

2.3.2 Convolutional Codes

The convolutional code was introduced first in 1955 by Elias [28]. A convolutional encoder is a set of linear time-invariant digital filters, consisting of n 1-bit stor-

age elements (D-type flip-flops). Figure 2.12 shows a $(1, 5/7)_8$ recursive systematic convolutional (RSC) code as an example of a convolutional encoder.

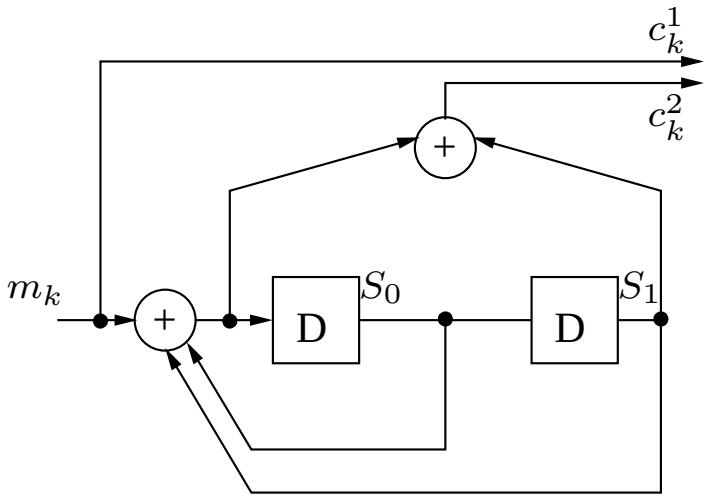

Figure 2.12: $(1, 5/7)_8$ Recursive systematic convolutional code encoder diagram.

The encoding process starts when the input stream passes through two filters producing two output streams. These two streams are mixed together to produce the coded stream (codewords), $[c_1^1 c_1^2\ c_2^1 c_2^2\ ... c_k^1 c_k^2]$. The corresponding codewords of this example can be evaluated by

$$c_k^1 = m_k \tag{2.26}$$

$$c_k^2 = m_k \oplus S_0 \tag{2.27}$$

Thus, for every bit of input, there are two coded output bits, resulting in a rate $R_c = \frac{1}{2}$ code. It is worth mentioning that the constraint length K is an integer that specifies the memory of the code (number of memory elements plus one) and for our example $K = 3$. Thus, codes with bigger K can achieve better performance, but at the cost of increasing the decoding complexity.

2.3.2.1 Representations of Convolutional Codes

There are many ways to describe convolutional codes, such as state tables, state diagrams, trellis diagrams and flow-graphs.

2.3.2.2 State Tables

State tables are an essential way to present convolutional code construction. Also, all other representation approaches will build on the state tables elements. Basically, the memory elements can contain 2^K possible values, where K is the number of memory elements. So, for the example in figure 2.12 there are 4 possible values (00, 01, 10, and 11). These values are called states and only certain state transitions are allowed. For instance, we cannot go from state 00 to state 11 in one step or stay in state 10. All possible state transitions, along with their corresponding input and outputs, can be determined from the encoder and are recorded in a state table. The state table for the non-symmetric convolutional codes is shown in table 2.1.

Table 2.1: $(1, 5/7)_8$ RSC state table.

Input	Initial States		Next States		Output	
m_k	S_0	S_1	$\hat{S}_0$	$\hat{S}_1$	c_k^1	c_k^2
0	0	0	0	0	0	0
1	0	0	1	0	1	1
0	0	1	1	0	0	0
1	0	1	0	0	1	1
0	1	0	0	1	0	1
1	1	0	1	1	1	0
0	1	1	1	1	0	1
1	1	1	0	1	1	0

2.3.2.3 State Diagrams

A state diagram is just a compact form of the state table. However, it is much easier to determine a codeword corresponding to a message using the state diagram.

Starting from state 00, trace the message through the state diagram and record each output as showing in figure 2.13.

2.3.2.4 Trellis Diagrams

The trellis diagram is made up of nodes connected by edges. Each of the nodes represents the states in the state diagram. Also, every column of the nodes represents all the possible states at a certain time. In addition, each node can have two possible paths (in binary systems and more in non-binary system) to move to the next state. The figure 2.14 shows the binary trellis diagram of the $(1, 5/7)_8$ RSC. It clearly can

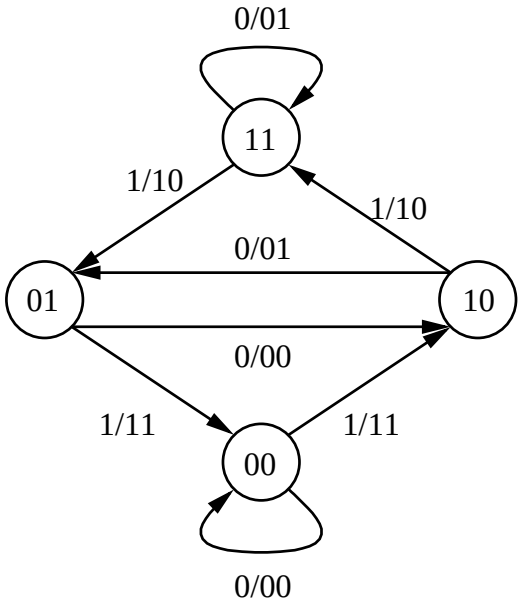

Figure 2.13: $(1, 5/7)_8$ RSC state diagram.

be noticed that each edge is labelled with an input and its corresponding coded output.

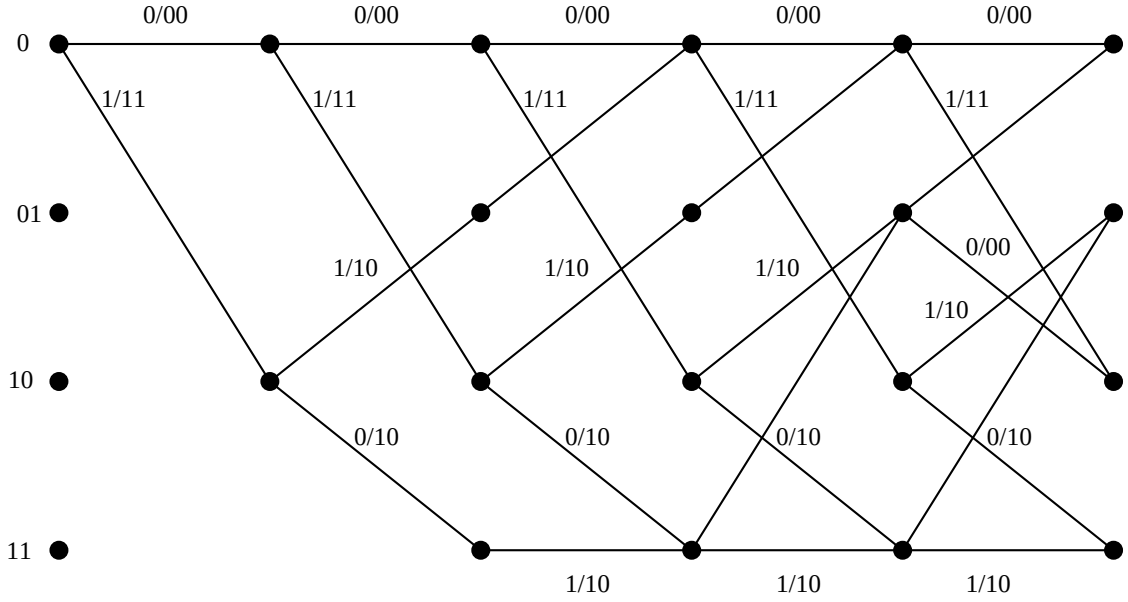

Figure 2.14: Trellis diagram of $(1, 5/7)_8$ RSC.

2.3.2.5 Transfer Function and Flow-Graph

In order to compute the performance of a convolutional code, it is essential to have a method that helps to enumerate all paths in the trellis. This method is mainly the transfer function, which takes into account all paths through the trellis that split

from the all zero path and then re-merge [46]. The transfer function can express the Hamming distance of each path diverging from the all-zero state, and it can evaluate the number of paths which are with a given Hamming distance. Furthermore, it can tell the length of each path and the number of input bits which are associated with each path from the all-zero path. In order to derive a transfer function, it is very convenient first to draw the corresponding flow-graph.

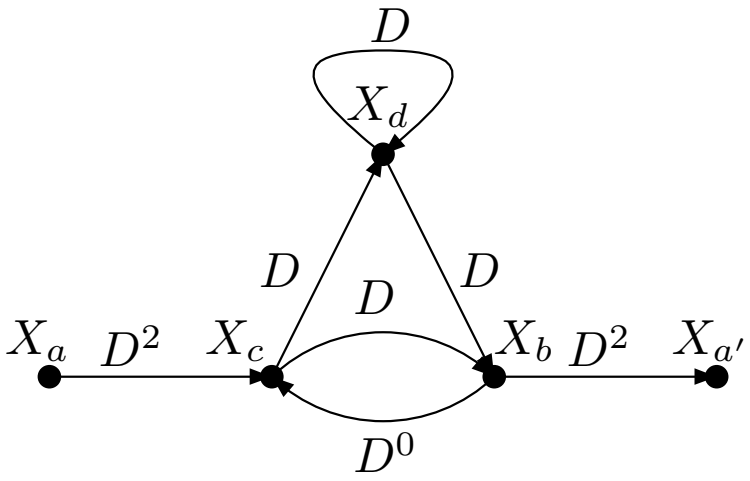

Figure 2.15: Signal flow-graph diagram of $(1, 5/7)_8 8$ RSC.

Figure 2.15 displays the signal flow-graph of the $(7, 5)_8$ RSC code. X_a, X_b, X_c and X_d are representing the states [00, 01, 10, and 11] respectively. While, X'_a is the duplication of state 00. Also, The power of D is reflecting the Hamming distance between the output bits corresponding to each state transition and the output 00. Hence, the transfer function $T(D)$ defined by

$$T(D) = \frac{X_{a'}}{X_a} \tag{2.28}$$

and it can be evaluated from the state equation with the help of the flow-graph, which are

$$\begin{aligned} X_b &= DX_d + DX_c \\ X_c &= D^2X_a + X_b \\ X_d &= DX_d + DX_c \\ X_{a'} &= D^2X_b \end{aligned} \tag{2.29}$$

After solving this set of equations the transfer function will be

$$T(D) = D^5 + 2D^6 + 4D^7 + 8D^8..... \tag{2.30}$$

Equation (2.30) explicitly tells that the $(1, 5/7)_8$ RSC code has one path with distance of five, two paths with distance of six and so on. Clearly, the minimum free distance d_{free} for this code is five, therefore, this code can correct up to $t = \frac{d_{free}-1}{2} = 2$ errors for any input length.

2.3.2.6 Decoding Convolutional Codes

Several algorithms have been developed for decoding convolutional codes. The one most commonly used is the Viterbi algorithm, which is a maximum likelihood sequence estimator [40,42,47]. A variation on the Viterbi algorithm is the soft-output Viterbi algorithm (SOVA), which provides not only decoded symbols but also an indication of the reliability of the decoded values [48]. Another algorithm is the maximum a posteriori (MAP) decoder which is referred to as the BCJR algorithm that computes probabilities of decoded bits [40]. The BCJR algorithm is somewhat more complex than the Viterbi algorithm. However, there are a few approximations to this approach with less complexity and close performance such as Log-MAP and Max-Log-MAP algorithms which are ideally suited for decoding turbo codes [43, 46, 49]. Since the Max-Log-MAP is implemented in chapters 3, 4 and 5 and it is just a modified BCJR algorithm, the BCJR algorithm will be explained in detail.

The BCJR decoding algorithm is suitable for estimating bit and/or state probabilities for a finite-state Markov system. The BCJR algorithm computes the posterior probability of symbols from Markov sources transmitted through discrete memoryless channels. Since the output of a convolutional coder that passes through a memoryless channel (such as an AWGN channel) forms a Markov source, the BCJR algorithm can be used for maximum posterior probability decoding of convolutional codes. Symbol by symbol MAP algorithm is an optimal decoding minimizing symbol error probability. The goal of the decoder is to determine log likelihood ratios (LLR) for all possible inputs. For a rate $R_c = \frac{1}{2}$ RSC code on a BI-AWGN channel, the LLR can be defined as

$$LLR(u_i|\boldsymbol{y}) = \ln \frac{P(u_i = +1|y)}{P(u_i = -1|y)} \tag{2.31}$$

where $P(u_i|\boldsymbol{y})$ is a posterior probability (APP) of u_i given the received y and u_i are the information bits $\in \{0, 1\}$ that are mapped into $\{0 \leftrightarrow +1, 1 \leftrightarrow -1\}$.

In order to get these logarithm likelihood ratios, all symbol posteriori probabilities $P(u_i = +1|\boldsymbol{y})$ must be obtained as,

$$P(u_i = +1|\boldsymbol{y}) = \sum_{u^+} p(s_{i-1} = s', s_i = s, y) \tag{2.32}$$

where s_i is the state at time i, u^+ is a set of state pairs (s, s') which state transition $(s_{i-1} = s') \rightarrow (s_i = s)$ is corresponding to the input (0) or $u_i = +1$. In the same way the denominator of equation (2.31) can be evaluated, when the state transition $(s_{i-1} = s') \rightarrow (s_i = s)$ is corresponding to the input (1) or $u_i = -1$. With the help of Bayes' rule, we need only calculate $p(s', s, y) = p(s_{i-1} = s', s_i = s, y)$ [50]. Then we need to sum all these state transitions in the numerator and denominator. The pdf $p(s', s, y) = p(s_{i-1} = s', s_i = s', y)$ is defined as

$$p(s', s, \boldsymbol{y}) = \alpha_{i-1}(s')\gamma_i(s', s)\beta_i(s) \tag{2.33}$$

where $\alpha_i(s)$ is the forward recursion, $\beta_i(s)$ is the backward recursion and $\gamma_i(s', s)$ is the state transitions factor, and these parameters can be evaluated, respectively, by

$$\alpha_i(s) = \sum_{s'} \gamma_i(s', s)\alpha_{i-1}(s') \tag{2.34}$$

$$\beta_{i-1}(s') = \sum_{s} \beta_i(s)\gamma_i(s', s) \tag{2.35}$$

$$\begin{aligned} \gamma_i(s', s) &= p(s|s')p(\boldsymbol{y}_i|s', s) \\ &= p(\boldsymbol{u}_i)p(\boldsymbol{y}_i|\boldsymbol{u}_i) \end{aligned} \tag{2.36}$$

Hence, the initial values for the forward recursion $\alpha_i(s)$ is

$$\alpha_0(s) = \begin{cases} 1, & s = 0, \\ 0, & s \neq 0, \end{cases} \tag{2.37}$$

and the backward recursion $\beta_i(s)$ is initialized according to

$$\beta_L(s) = \begin{cases} 1, & s = 0, \\ 0, & s \neq 0, \end{cases} \tag{2.38}$$

where L is the time when the termination bits will be padded at the end of codeword [51].

Finally, for more simplification $\gamma_i(s', s)$ can be rewritten for the AWGN channel as

$$\gamma_i(s', s) = \frac{1}{2\pi N_0} \exp\left[-\frac{||\boldsymbol{y}_i - \boldsymbol{c}_i||^2}{N_0}\right] \tag{2.39}$$

2.3.3 Non-Binary Codes

2.3.3.1 Non-Binary Fields Construction

Finite fields are very important in many areas, especially for channel coding theory. For example, binary is the smallest finite field which consists of {0,1} elements. For general definition, let q be positive prime integer, then the finite field will be GF(q)={0, 1, ..., $q-1$}. These fields must satisfy the following conditions under the two binary real operations (+), and (.):

1. Associativity.
2. Distributivity.
3. Commutativity under addition.

Table 2.2 showing one of these operations (+) on field of 7 as an example.

Table 2.2: Addition table for GF(7)

+	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	2	3	4	5	6	0
2	2	3	4	5	6	0	1
3	3	4	5	6	0	1	2
4	4	5	6	0	1	2	3
5	5	6	0	1	2	3	4
6	6	0	1	2	3	4	5

In addition, we can extend the finite field so that it is represented by a primitive polynomial which is used to define GF(p^q), where p is the primitive element with a

root of β. This class of finite field is called an extension field and will be considered in the next chapters. In order to clarify the idea of constructing the extension field GF(2^3) will be illustrated in table 2.3 as an example. Hence, the extension field GF(2^3) can be generated by using the primitive polynomial $f(x) = x^3 + x + 1$. If we consider β as the root of $f(x)$ then $\beta^3 = \beta + 1$ [52].

Table 2.3: Construction of GF(2^3).

Element in GF(2^3)	Element represented as the sum of lower powers of β	Element represented as 3-tuple vector over GF(2)
0	0	000
1	1	001
β	β	010
β^2	β^2	100
β^3	$\beta + 1$	011
β^4	$\beta^2 + \beta$	110
β^5	$\beta^3 + \beta^2 = \beta + 1 + \beta^2$	111
β^6	$\beta^4 + \beta^3 = \beta^2 + \beta + \beta + 1 = \beta^2 + 1$	101

2.3.3.2 Non-binary Convolutional Codes

In this section, we will shortly, describe the encoding scheme of a non-binary convolutional code. R. Baldini, F and P.G. Farrell in 1994 introduced the non-binary convolutional code. They constructed this code based on a ring of positive integers modulo-q, and this class of codes is suitable for M-PSK modulation technique [53]. Later on this code has been widely used with PSK modulation for many applications [54]. Figure 2.16 shows the general diagram of a recursive systematic non-binary convolutional encoder with code rate $R = \frac{m}{m+1}$ and a constraint length K defined as the number of memory elements plus one [1]. The m inputs take non-binary values in Galois field GF(q) .

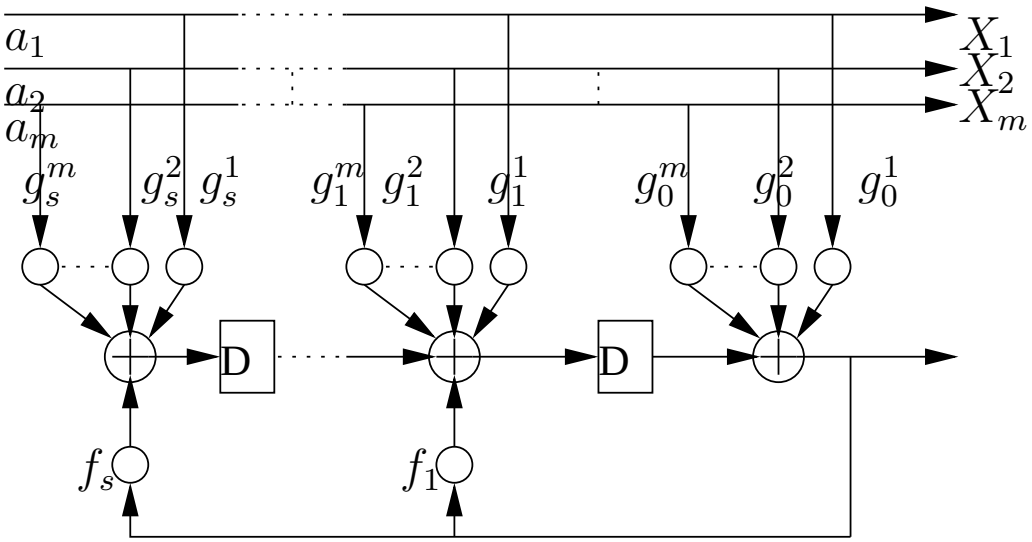

Figure 2.16: General Diagram for Non-Binary Convolutional Codes.

2.4 Turbo Codes

2.4.1 Turbo Encoder

Turbo codes are a class of high-performance FEC codes presented by Berrou, Glavieux and Thitimajshima in 1993 [55]. Turbo codes are built from a parallel concatenation of two Recursive Systematic Convolutional (RSC) codes separated by an interleaver, denoted by Π, as shown in figure 2.17.

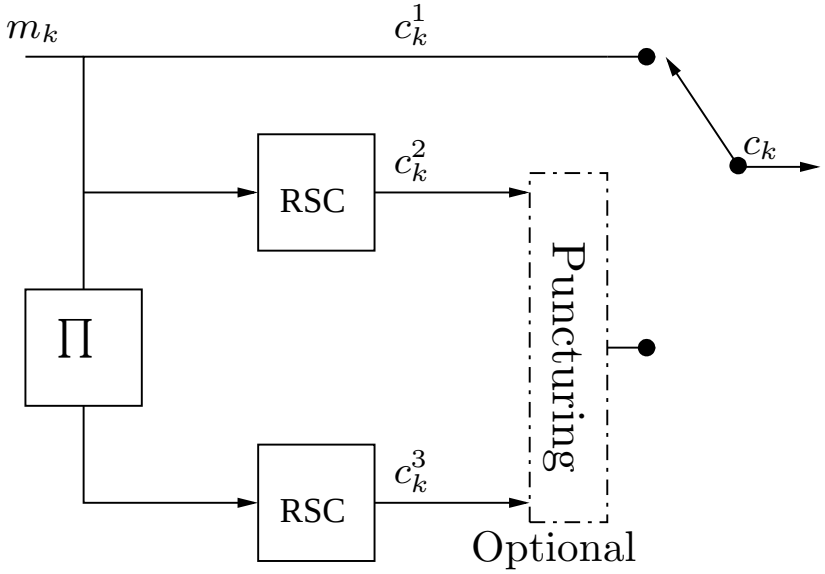

Figure 2.17: Turbo encoder general diagram.

The length-K message m is encoded directly by the first component encoder,

which produces the parity bits c^2; however, it is interleaved before being encoded by the second convolutional encoder, which produces the parity bits c^3. Turbo codes are systematic codes; thus the bits c^1, c^2 and c^3 are all transmitted and the turbo codeword is $c_k = [c_1^1 c_1^2 c_1^3, c_2^1 c_2^2 c_2^3, \ldots, c_k^1 c_k^2 c_k^3]$ [56].

2.4.2 Turbo Decoder

There are various optimal and sub-optimal algorithms used to decode binary turbo codes on the additive white Gaussian noise (AWGN) channel [43]. In the next chapters, the Max-Log-MAP algorithm is employed to decode non-binary turbo codes on impulsive noise channels. Basically, this algorithm finds the maximum probability input symbol by estimating the probability of each trellis edge that corresponds to one of the q inputs. The probability of the state transitions from the state s_0 at time $t-1$ to state s_1 at time t is given as

$$P(s_0, s_1, \boldsymbol{y}) = P(s_0, \boldsymbol{y}_t^-) P(s_1, \boldsymbol{y}_t | s_0) P(\boldsymbol{y}_t^+ | s_1), \tag{2.40}$$

where $\boldsymbol{y}_t$, $\boldsymbol{y}_t^-$, and $\boldsymbol{y}_t^+$ are the received symbols at time (t), the received symbols before time (t), and the received symbols after time (t), respectively [56]. If we take the logarithm of equation (2.40), we obtain

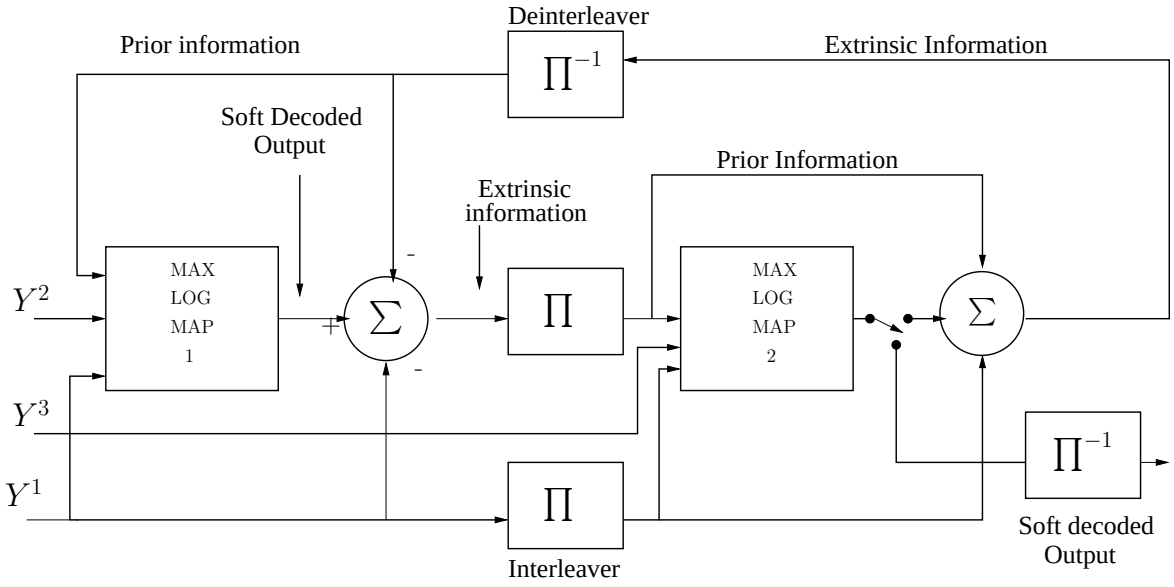

Figure 2.18: Turbo decoder general diagram.

$$\begin{aligned} m_t(s_0, s_1) &= \ln P(s_0, \boldsymbol{y}_t^-) + \ln P(s_1, \boldsymbol{y}_t | s_0) + \ln P(\boldsymbol{y}_t^+ | s_1) \\ &= \alpha_{t-1}(s_0) + \gamma_t(s_0, s_1) + \delta_t(s_1) \end{aligned} \tag{2.41}$$

where γ_t is the probability transition metric between s_0 and s_1, α_t is the forward recursion, β_t is the backward recursion. In the Max-Log-MAP algorithm, α_t and β_t can be calculated as

$$\alpha_t(s_0) = \max_i \left\{ \alpha_{t-1}(s_i) + \gamma_t(s_i, s_0) \right\}, \tag{2.42}$$

$$\delta_t(s_1) = \max_i \left\{ \delta_{t+1}(s_i) + \gamma_{t+1}(s_1, s_i) \right\}. \tag{2.43}$$

The turbo decoder consists of two component decoders in series as shown in figure 2.18. Each constituent decoder has two inputs:

- Received soft channel values.
- A priori information.

In general, the output from the 1st decoder is used to assist the 2nd one and vice versa, thus completing a loop. This cycle is known as a decoding iteration. Errors that were not corrected in one decoding iteration may be decoded in the next iteration. Hence, improvements in performance can be achieved with each iteration, but we find that the improvements become less with each iteration until we reach an optimal solution [2].

When the performance gains get smaller, that means an optimal solution is reached. This property is known as convergence.

To further explain the decoding procedure and with the help of the trellis and the BCJR algorithm for convolutional codes, algorithm 1.1 illustrates the decoding procedure step by step as can be found in [56].

Algorithm 2.1 Binary turbo decoding

procedure DECODER(trellis encoder1, trellis2, interleaver index $\prod$, Number of iterations(I), Y^1, Y^2, Y^3)
    $E^1 = 0$
    for L=1:I do
    $A^1 = \prod^{-1}(E^2)$
    $L^1 = logBCJR(trellis1, Y^1, Y^2, A^1)$
    $E^1 = L^1 - R - A^1$
    $A^2 = \prod(E^1)$
    $L^2 = logBCJR(trellis2, Y^1, Y^3, A^2)$
    $E^2 = L^2 - \prod(R) - A^2$
    End for
    $LLR = \prod^{-1}(L^2)$
end procedure

2.5 Multi-Input Multi-Output

Employing more than one antenna at the receiver of the communication systems has the potential of boosting coverage and capacity. Figure 2.19 illustrates a general block diagram of the communication systems with multiple transmitting antennas N_t and multiple receiving antennas N_r on slow fading channels. Where η is the AWGN component, $H(N_t \times N_r)$ is the MIMO channel matrix. In this case the received signal y can be calculated at each rth antenna as

$$y(i) = \sum_{i=0}^{N_t} s_n h(i, r) + \eta_r \tag{2.44}$$

In order to eliminate the effect of the fading MIMO channels there are several types of channel estimations. In this section, three commonly used channel detectors are listed bellow [12, 23, 57–59]

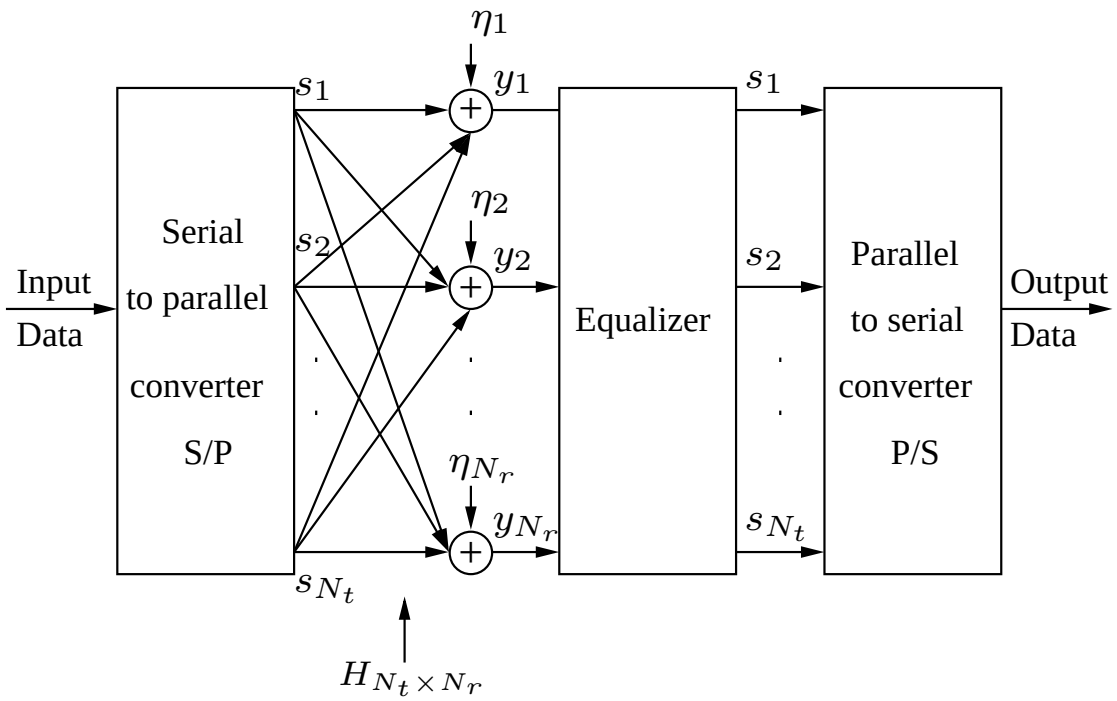

Figure 2.19: General MIMO system block diagram.

- Maximum Likelihood Detector (MLD)

  MLD is the optimum equalizer by minimizing the probability of error and the output of this detector can be defined as

$$\hat{s_i} = \sum_{m=1}^{N_r} \left| \boldsymbol{y}_m - \sum_{n=1}^{N_t} h_{mn} s_n \right|^2 \tag{2.45}$$

  It is very obvious that the complexity of this detector increases exponentially with M^{N_t}, where M is the number of modulation symbols.

- Minimum Mean-Square-Error Detector (MMSE). The MMSE detector weight matrix W that is selected to minimize the mean error, can be calculated as

$$W = (HH^H + \sigma^2 \boldsymbol{I}) h_i, \tag{2.46}$$

  where $\boldsymbol{I}$ is the $N_t \times N_r$ identity matrix.

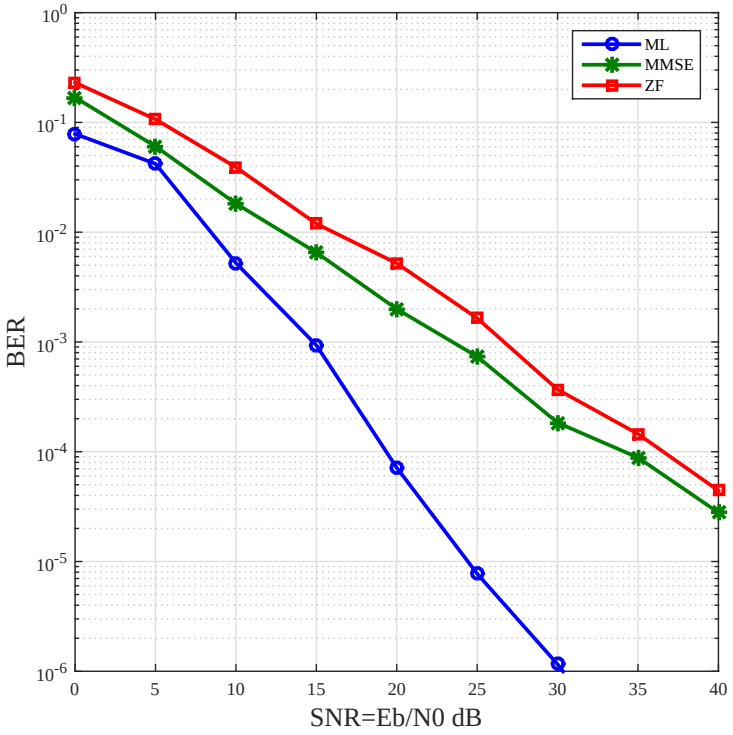

Figure 2.20: Performance of ZF, MMSE, and MLD with $N_r = N_t = 2$ MIMO systems.

- Zero Forcing Detector (ZF)

  ZF applies the inverse of the channel frequency response (CRF), and the weight matrix W of ZF detector can be estimated as

$$w(k) = \frac{H^*(k)}{|H(k)^2|} \tag{2.47}$$

Figure 2.20 compares the BER performance of these detectors, and it shows that the MLD sufficiently achieves a larger gain that the ZF and MMSE.

2.6 Literature Survey of Transmission Methods over Impulsive Channels

2.6.1 Error Correcting Codes on Non-Gaussian Channels

Turbo codes and LDPC codes are very well known coding schemes that achieve near-Shannon limit performance on the AWGN channel. However, their performance

degrades when the channel exhibits fading effects or suffers from impulsive noise. Impulsive noise or non-Gaussian noise can be generated by natural phenomena or be man-made. Commonly presented examples of this natural phenomena are underwater acoustic noise and the lightening in the atmosphere [6,17], while electromagnetic interference EMI, the background noise on PLC channels and unwanted radiation from the electrical and electronic devices such as; computers, monitors and printers are examples of the man-made impulsive noise [60–63]. In addition, telecommunication equipment such as the antennas and imperfect designed receivers, can be one of the most damaging sources of impulsive noise. A conventional communication system that considers Gaussian noise as a default scenario would be severely degraded in performance.

In this thesis, we investigate the performance of non-binary turbo codes on additive alpha stable $S\alpha S$ noise channels. There appears to be no work in the literature on non-binary coding schemes and their performance on alpha stable $S\alpha S$ noise, but there is some work in the literature on binary turbo codes and impulsive noise. In [64], an alternative way to design a receiver was proposed where the conventional Euclidean distance metric was replaced by p-norm metric to mitigate the impulsive noise. Binary turbo decoding in a Middleton class A noise environment with a class A filter was presented in [65] as a power line communication receiver. Burnic et al. [66] also evaluated the performance of binary turbo codes on Gaussian mixture distributions noise channels.

However, the main advantage of PLC is in utilizing the established electrical grid which provides every single property on the planet with electricity. Since the power networks are not designed for communication services, there are many obstacles to overcome for reliable communication over the transmission lines. These obstacles are: attenuation, impulsive noise and multipath frequency selectivity. Communication over the transmission line is most likely to be exposed and affected by Middleton class A noise model due to the electromagnetic interference [5]. Using coded PLC system is an efficient way to enhance the bit error rate performance as can be seen in [65,67,68]. The authors used different coding schemes and compared coded BER with uncoded BER and showed the significant gain over uncoded PLC system. Even though they illustrated a great gain, their systems still lack practicality by not considering a realistic multipath frequency selective channel. Typical PLC multipath frequency selective channels specifications are well presented in [69,70].

In this thesis to prevent the burst errors due to the impulsive noise, non-binary turbo code is used and to mitigate the effect of large noise signal values, blanking is applied [52, 71, 72]. To the best of the authors' knowledge, non-binary turbo codes have not been considered on PLC systems.

For low complexity and near optimal performance, a sub-optimal decoding algorithm, Max-Log-Map, is used [43] which is a simplified Maximum A Posteriori algorithm. Furthermore, orthogonal frequency-division multiplexing is employed to combat the frequency selectivity. In addition, OFDM is a powerful tool which is highly resistant to the effect of impulsive noise by spreading the noise signal energy simultaneously over sub-carriers [73, 74].

2.6.2 Coded Wired and Wireless MIMO on Impulsive Noise

Exploiting the already established electrical grid does position PLC favorably, although the unsuitability of power line networks for communication services can make transmission quite unreliable because of vulnerability to impulsive noise and the frequency-selective channels. In such extreme environments, error correction alone cannot combat the serious effect of the channel and must be combined with signal processing techniques such as channel estimation, channel equalization and OFDM, as can be seen in [75–77]. These papers show the benefits of using coding channel on SISO PLC systems.

However, recently, the demand for high data rate communication networks has been increasing. Even though MIMO systems have been showing a great performance in the matter of coverage, reliability and capacity only a few research papers have deployed coded MIMO with PLC. Since the power line networks are not designed for communication purposes, transferring data over these cables is vulnerable to impulsive noise due to electromagnetic interference [5]. However, it is well known that utilizing channel coding would enhance the overall communication bit error rate performance on such a harsh environment, as we can conclude from the point to point coded PLC systems in [65, 67, 68, 78]. But in large houses or buildings these SISO coded systems might lack in coverage and capacity [79].

However, since the 1990s, there are many studies aiming to enhance the communication link reliability and increase the data rate on wireless telecommunication systems [80, 81]. The first released MIMO standard for power line networks was

in 2011 by the International Telecommunications Union-Telecommunication Standardization Sector (ITU-T) in their standard (ITU-T G.hn) and also named by the Institute of Electrical and Electronics Engineers (IEEE) as IEEE1901 [82, 83].

In order to achieve that goal, the need for accurate statistical physical models to imitate the behaviour of MIMO PLC channels is increasing. Authors in [84–89] are interested in modeling a MIMO-PLC channel based on mathematical algorithms and/or by modifying the SISO-PLC channels. In [85] and [87], the authors proposed a model of MIMO channel that was totally reliant on the multi-path frequency selective channel model for SISO, which is presented in [69] by M. Zimmermann and K. Doster. While in [84] and [88], they depend on a mathematical analysis that represents some physical phenomena in the realistic channel, such as the coupling effects and the multi-path propagation. Moreover, several models have been designed based on multi-conductor transmission theory (MTL), which can be achieved by computing the corresponding parameters mathematically or by measuring them [90–92].

In this thesis, the synthetic statistical MIMO PLC channel model in [89] has been used. The rationale behind utilizing this model is the fact that it is grounded in a pure phenomenological approach and also has the added benefit of being compatible with the measured channel.

2.7 Summary

In this chapter, the theoretical background of the main concepts that this work relies on have been presented, such as basic knowledge about non-Gaussian noise models and their applications. Also, the comparison between uncoded communication systems on AWGN channels with the uncoded systems on impulsive noise have been carried out. In addition , an introduction to channel coding and the Shannon limit have been explained briefly. Since trellis codes are the main interest of this thesis, the decoding and encoding procedure of trellis codes have been illustrated. A brief knowledge about non-binary field construction and non-binary convolutional codes have been presented as well. Furthermore, MIMO communication systems with most common channel estimators have been illustrated.

Chapter 3

Non-Binary Trellis Codes on Additive $S\alpha S$ Noise Channels

3.1 Introduction

Binary codes with iterative decoders are thought to be able to accomplish optimum levels of performance on conventional communication channels. However, their performance on more practical channels can deteriorate because of the existence of burst errors or impulsive noise. A better performing coding scheme is the class of non-binary codes, which are known to be more effective in correcting burst errors, but interestingly there is very little research reported in the literature investigating non-binary codes on impulsive noise channels. Therefore, the performance analysis of non-binary trellis codes on AWGN channels has been introduced in this chapter with numerical examples. Furthermore, an investigation into the performance of non-binary convolutional codes and turbo codes defined in a finite field GF(4) on symmetric alpha-stable impulsive noise channels and compared with binary codes that achieve similar BER performance on the AWGN channel has been carried out. A Cauchy receiver is also employed to mitigate the effects of the channel to assist the turbo decoding. The simulation results show that although the non-binary trellis codes performs similarly to the binary turbo code on the AWGN channel, it achieves a significant coding gain over the binary trellis code as impulsiveness increases. This chapter is organized as follows: Section 3.2, introduces two classes of non-binary convolutional codes and their construction. In section 3.3, the decoding algorithm (BCJR) employed in this chapter is presented with the appropriate mod-

ifications that make it suitable for use with the Cauchy receiver. In section 3.4, the flow-graph of the $\beta\beta^2/1$ non binary convolutional code is explained in detail. Simulation results of the BER performance of non-binary, binary convolutional codes, binary, non-binary turbo codes and uncoded systems on $S\alpha S$ noise channels are presented in section 3.7. Finally, section 3.8 presents our conclusions. This chapter is based on our published work in [71].

3.2 Non-Binary Convolutional Codes

3.2.1 4-States Non-Binary Convolutional Code

Figure 3.1 shows a rate $\frac{1}{2}$ RSC encoder defined in GF(4) = $\{0, 1, \beta, \beta^2\}$, where β is a primitive element in GF(4). This code consists of one memory element D, two forward multipliers G=[β , β^2], and one feedback multiplier F=[1]. The outputs of this code $c1$ and $c2$ can be described by:

$$c_1(i) = m(i), \tag{3.1}$$

where m is the input symbols.

$$c_2(i) = (G(2) \times m(i)) \oplus S_0, \tag{3.2}$$

where S_0 is current state.

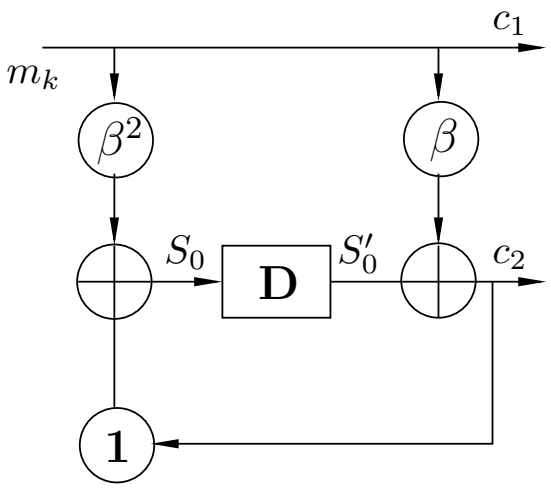

Figure 3.1: β $\beta^2/1$ Non-binary convolutional encoder.

Hence, the next state S'_0 can be calculated by:

$$S'_0 = G(1) \times m(i) \oplus (F \times (G(2) \oplus m(i))) \oplus S_0 \tag{3.3}$$

This code is known to have the maximal free distance, $d_{free} = 10$, for a non-binary RSC code in GF(4) and constraint length $K = 2$.

Table 3.1 shows all possible inputs and states transitions $S \rightarrow S'$ and the corresponding coded symbols.

Table 3.1: State Table of β $\beta^2/1$ non-binary convolutional code

Input	Initial State S	Next State S'	Output c_1c_2	Input	Initial State S	Next State S'	Output c_1c_2
0	0	0	00	0	β	β	0 β
1	0	1	1 β	1	β	β^2	10
β	0	β	β 0	β	β	0	β β
β^2	0	β^2	β^2 β	β^2	β	1	β^2 0
0	1	1	01	0	β^2	β^2	0 β^2
1	1	β	1 β^2	1	β^2	0	11
β	1	β^2	β 1	β	β^2	1	β β^2
β^2	1	0	β^2 β^2	β^2	β^2	β	β^2 1

In addition, the trellis diagram is shown in Figure 3.2. Comparing to the binary trellis, we can clearly see that although the number of states for the field of GF(4) non-binary convolutional code is the same as the $(7, 5)_8$ binary convolutional code (see Figure 2.14), the number of branches in the trellis are doubled since there are four branches entering and leaving each node. Furthermore, from the state table, we can draw the signal-flow graph as shown in Figure 3.5, which is more complicated as compared to the signal flow graph of a 4-state binary convolutional code in Figure 2.15.

However, as shown in Figure 3.2, one of the shortest path is $state_0 \longrightarrow state_{\beta^2} \longrightarrow state_0$ in terms of squared euclidean distance, which can be calculated as:

$$d^2_{free} = d^2_{(0-\beta^2)} + d^2_{(\beta^2-0)} = 10$$

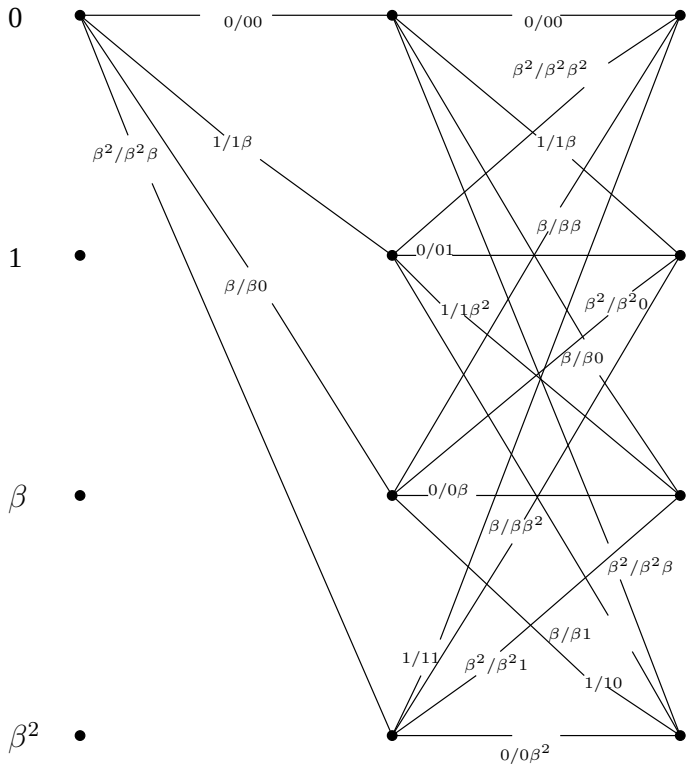

Figure 3.2: The trellis diagram of the β $\beta^2/1$ convolutional code.

Then the asymptotic coding gain over uncoded BPSK is calculated as

$$\xi = 10\log_{10}\left(\frac{d^2_{free/coded}}{d^2_{free/uncoded}}\right) = 10\log_{10}\left(\frac{10}{4}\right) = 3.98dB. \tag{3.4}$$

To implement this code in Matlab, it is worth mentioning that normally we add an extra symbol to the end of the message to make sure the codeword is terminated. In this way, we can always trace back path $state_0$ during the decoding procedure.

3.2.2 16-States Non-Binary Convolutional Code

Fig 3.3 illustrates the $\beta 1\beta/\beta^2 1$ RSC non-binary convolutional encoder with rate $\frac{1}{2}$. As we can see, this code consists of; two memory elements D, three forward multipliers G=[β , 1, β], and two feedback multipliers F=[β^2 , 1]. Its state table is given in Table 3.2 and the corresponding codeword (c_1c_2) and the states transitions can be obtained by:

$$c_1(i) = m(i), \tag{3.5}$$

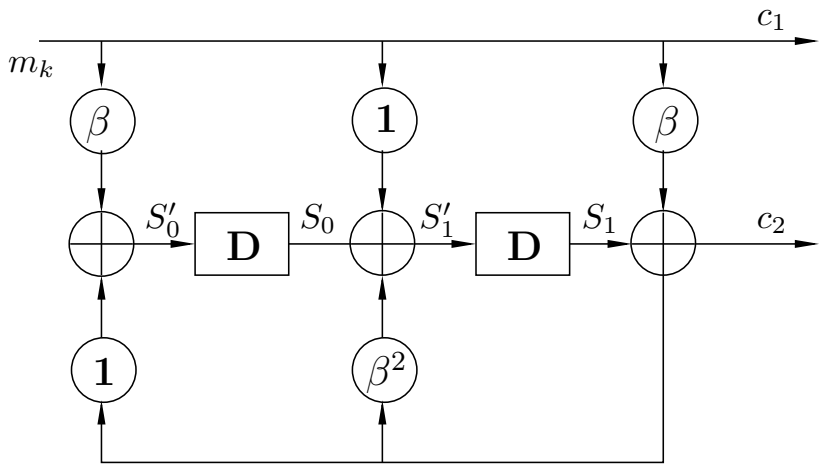

Figure 3.3: $\beta 1\beta/\beta^2 1$ Non-binary convolutional encoder.

$$c_2(i) = S_1 \oplus (G(1) \times m(i)), \tag{3.6}$$

$$S'_0 = (m(i) \times G(3)) \oplus (c_2(i) \times F(2)) \tag{3.7}$$

$$S'_1 = S_0 \oplus (((m(i) \times G(2)) \oplus (c_2(i) \times F(1))) \tag{3.8}$$

where S_0, S_1 , S'_0, S'_1 is current state1, current state2, next state1 and next state2, respectively.

Furthermore, the corresponding state table which shows all the input and output of the 16-state $\beta 1\beta/\beta^2 1$ non-binary convolutional code is presented in table 3.2. From the state table, we can easily draw the trellis diagram as shown in Figure 3.4.

According to the state table and the trellis diagram, one of the shortest path is $state_{00} \longrightarrow state_{0\beta} \longrightarrow state_{\beta 0} \longrightarrow state_{00}$. The free distance is given as:

$$d^2_{free} = d^2_{(00-0\beta)} + d^2_{(0\beta-\beta 0)} + d^2_{(\beta 0-00)} = 16$$

then the asymptotic coding gain over uncoded BPSK is calculated as

$$\xi = 10 \log_{10} \left(\frac{16}{4} \right) = 6.02 dB.$$

Table 3.2: State table of $\beta 1\beta/\beta^2 1$ non-binary convolutional code.

m	S_0	S_1	S_0'	S_1'	c_1	c_2	m	S_0	S_1	S_0'	S_1'	c_1	c_2
0	0	0	0	0	0	0	0	β	0	0	β	0	0
1	0	0	β^2	β^2	1	β	1	β	0	β^2	1	1	β
β	0	0	1	1	β	β^2	β	β	0	1	β^2	β	β^2
β^2	0	0	β	β	β^2	1	β^2	β	0	β	0	β^2	1
0	0	1	1	β^2	0	1	0	β	1	1	1	0	1
1	0	1	β	0	1	β^2	1	β	1	β	β	1	β^2
β	0	1	0	β	β	β	β	β	1	0	0	β	β
β^2	0	1	β^2	1	β^2	0	β^2	β	1	β^2	β^2	β^2	0
0	0	β	β	1	0	β	0	β	β	β	β^2	0	β
1	0	β	1	β	1	0	1	β	β	1	0	1	0
β	0	β	β^2	0	β	1	β	β	β	β^2	β	β	1
β^2	0	β	0	β^2	β^2	β^2	β^2	β	β	0	1	β^2	β^2
0	0	β^2	β^2	β	0	β^2	0	β	β^2	β^2	0	0	β^2
1	0	β^2	0	1	1	1	1	β	β^2	0	β^2	1	1
β	0	β^2	β	β^2	β	0	β	β	β^2	β	1	β	0
β^2	0	β^2	1	0	β^2	β	β^2	β	β^2	1	β	β^2	β
0	1	0	0	1	0	0	0	β^2	0	0	β^2	0	0
1	1	0	β^2	β	1	β	1	β^2	0	β^2	0	1	β
β	1	0	1	0	β	β^2	β	β^2	0	1	β	β	β^2
β^2	1	0	β	β^2	β^2	1	β^2	β^2	0	β	1	β^2	1
0	1	1	1	β	0	1	0	β^2	1	1	0	0	1
1	1	1	β	1	1	β^2	1	β^2	1	β	β^2	1	β^2
β	1	1	0	β^2	β	β	β	β^2	1	0	1	β	β
β^2	1	1	β^2	0	β^2	0	β^2	β^2	1	β^2	β	β^2	0
0	1	β	β	0	0	β	0	β^2	β	β	β	0	β
1	1	β	1	β^2	1	0	1	β^2	β	1	1	1	0
β	1	β	β^2	1	β	1	β	β^2	β	β^2	β^2	β	1
β^2	1	β	0	β	β^2	β^2	β^2	β^2	β	0	0	β^2	β^2
0	1	β^2	β^2	β^2	0	β^2	0	β^2	β^2	β^2	1	0	β^2
1	1	β^2	0	0	1	1	1	β^2	β^2	0	β	1	1
β	1	β^2	β	β	β	0	β	β^2	β^2	β	0	β	0
β^2	1	β^2	1	1	β^2	β	β^2	β^2	β^2	1	β^2	β^2	β

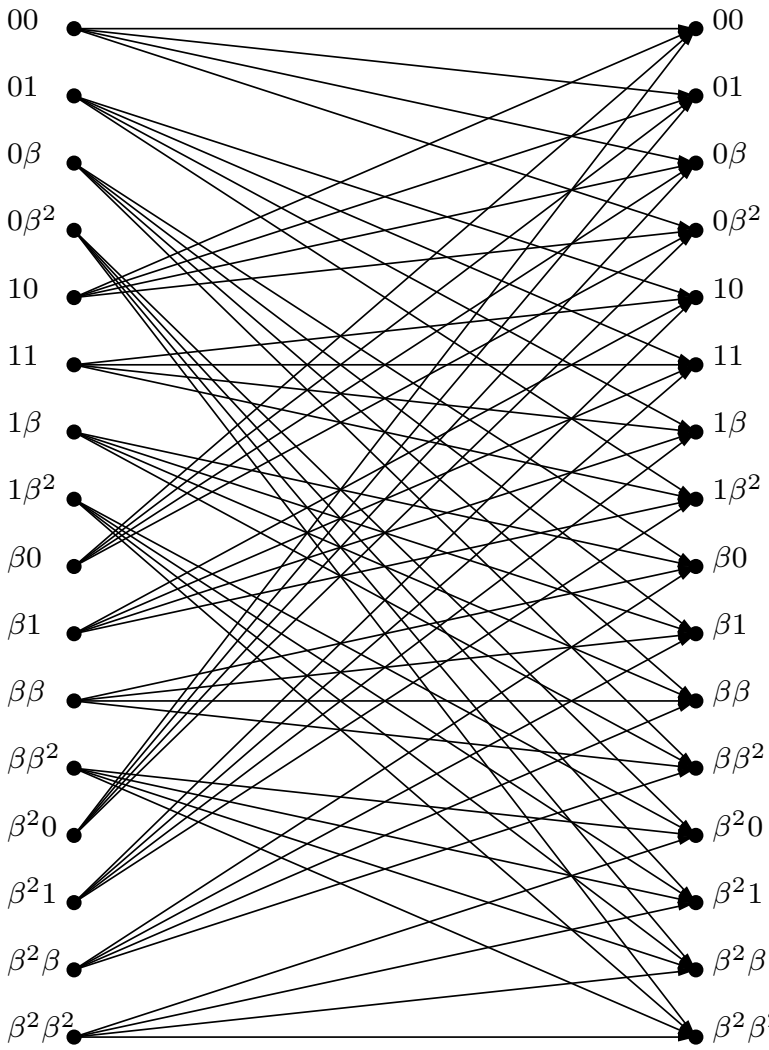

Figure 3.4: The trellis diagram of the $\beta 1\beta/\beta^2 1$ convolutional code.

3.3 Decoding Algorithm of Non-Binary Convolutional Codes on $S\alpha S$ Noise Channels

In this thesis, the modified Max-Log-MAP decoding algorithm will be considered for decoding both non-binary convolutional codes and act as component decoders for the non-binary turbo decoder, since it has lower complexity with only minimal degradation in performance compared to the MAP algorithm. Furthermore, the main modified equations of the MAP algorithm will be presented in this section. This method depends on finding the maximum probability input symbol by estimating the probability of each trellis edge which corresponds to one of the inputs. There is a relation between states transitions (S,S') and their corresponding input(k) as

illustrated in table 3.1 and if we can estimate any two of them, we can find the third one. In other words, by calculating the probability of the state transition at time t, we can estimate the received symbol at time $t+1$, Y_t^1 [56].

$$P(S, S', \boldsymbol{Y}) = P(S, Y_t^0)P(S', Y_t|S)P(Y_t^1|S'), \tag{3.9}$$

where, Y_t, Y_t^0, and Y_t^1 are the received symbols at time (t), the received symbols before time (t), and the received symbols after time (t), respectively. These three terms in equation (3.9) are commonly written as:

$$\Gamma_t(S, S') = P(S', \boldsymbol{Y}_t|S) \tag{3.10}$$

$$A_t(S') = P(S', \boldsymbol{Y}_t^0) \tag{3.11}$$

$$B_{t+1}(S) = P(\boldsymbol{Y}_t^1|S) \tag{3.12}$$

First, the probability (Γ_t) of a transition from S and S' is calculated. Since we are working on both Gaussian and $S\alpha S$ noise distributions, Γ for both channels must be determined as shown below:

$$\Gamma_t(S, S'))_{AWGN(MaxLog)} = -\frac{1}{2\sigma^2}||\boldsymbol{Y}_1 - \boldsymbol{X}_1||^2 - \frac{1}{2\sigma^2}||\boldsymbol{Y}_2 - \boldsymbol{X}_2||^2 \tag{3.13}$$

$$\Gamma_t(S, S'))_{AWGN(Map)} = \exp\left(\frac{((\boldsymbol{Y}_1 - \boldsymbol{X}_1)^2 + (\boldsymbol{Y}_2 - \boldsymbol{X}_2))^2}{2\sigma^2}\right) \tag{3.14}$$

$$\begin{aligned}\Gamma_t(S, S'))_{Cauchy(MaxLog)} = &\log\left(\frac{\gamma}{\pi[\gamma^2 + ||\boldsymbol{Y}_1 - \boldsymbol{X}_1||^2]}\right) \\ &+ \log\left(\frac{\gamma}{\pi[\gamma^2 + ||\boldsymbol{Y}_2 - \boldsymbol{X}_2||^2]}\right),\end{aligned} \tag{3.15}$$

$$\begin{aligned}\Gamma_t(S, S'))_{Cauchy(Map)} = &\frac{\gamma}{\pi[\gamma^2 + ||\boldsymbol{Y}_1 - \boldsymbol{X}_1||^2]} \\ &\times \frac{\gamma}{\pi[\gamma^2 + ||\boldsymbol{Y}_2 - \boldsymbol{X}_2||^2]},\end{aligned} \tag{3.16}$$

where X_1 and X_2 are the $M-PSK$ symbols of the message and parity check symbols respectively. Y_1 and Y_2 are the received message and parity check symbols respectively, σ^2 is the variance of the additive white Gaussian noise and γ is the dispersion of the $S\alpha S$ distribution. The remaining parameters (A and B_{t+1}) depend only on Γ_t and can be calculated by:

$$A_t(S') = \max\{\sum_{i=0}^{2^v-1} A_t(S_i) + \Gamma_t(S'_i, S)\} \tag{3.17}$$

$$B_{t+1}(S) = \max\{\sum_{i=0}^{2^v-1} B_{t+1}(S_i) + \Gamma_{t+1}(S', S_i)\} \tag{3.18}$$

where v is the number of memories of the encoder, and $2^v - 1$ is the number of all possible states. Finally, the log-likelihood ratios of the decoded message symbols $\hat{X}_t$ are defined as

$$\boldsymbol{L}(\hat{X}_t = x|Y_t) = \max_{m=x}\{A + B + \Gamma\} - \max_{m=0}\{A + B + \Gamma\}, \tag{3.19}$$

where $x \in GF(4)$.

3.4 Performance Estimates for Non-Binary Convolutional Codes

It is well known that the free distance of a binary and non-binary convolutional code give an approximated first order of the error performance. The following equation is an upper bound on the error performance of a non-binary convolutional codes:

$$P_b \lesssim \sum_{w=d_{free}}^{\infty} c_w P_w, \tag{3.20}$$

where c_w is the number of codewords of weight w, and P_w is the pairwise error probability given by:

$$P_w = Q\left(\frac{d_E}{2\sigma}\right), \tag{3.21}$$

where, d_E is the Euclidean distance between the transmitted all zero codeword $m(0)$ and the received codewords $c(\hat{m})$. For the BI-AWGN channel, let $m(0) =$

$[(+\sqrt{E_s}, +\sqrt{E_s})]$, $m(1) = [(-\sqrt{E_s}, +\sqrt{E_s})]$, $m(\beta) = [(+\sqrt{E_s}, -\sqrt{E_s})]$ and $m(\beta^2) = [(-\sqrt{E_s}, -\sqrt{E_s})]$, where E_s is the average code-bit energy and it is related to the code rate R_c and the average data-bit energy E_b by $E_s = R_c E_b$.

However, the Euclidean distance in this case is calculated as:

$$d_E = \sqrt{(2\sqrt{wE_s})^2 + (2\sqrt{wE_s})^2 + (4\sqrt{wE_s})^2} = \sqrt{24wE_s} \tag{3.22}$$

P_w is the probability of the noise at the decoder output and it is true only if the received signal have been effected by a noise magnitude greater than $d_E/q = \frac{\sqrt{24wE_s}}{4} = \sqrt{\frac{3wE_s}{2}}$.

Now, the pairwise error probability can be evaluated by:

$$P_w(binary) = Q\left(\sqrt{\frac{2wE_s}{N_0}}\right) = Q\left(\sqrt{\frac{2wR_cE_b}{N_0}}\right). \tag{3.23}$$

$$P_w(non-binary) = \frac{2^{k-1}}{2^k - 1} Q\left(\sqrt{\frac{3wE_s}{N_0}}\right) = \frac{2^{k-1}}{2^k - 1} Q\left(\sqrt{\frac{3wR_cE_b}{N_0}}\right). \tag{3.24}$$

where $k = log_2(q)$ and the term on the left converts the symbol errors to bit errors.

Thus, the probability of a codeword error P_{cw} and the bit error probability P_e can be obtained from Eq. (3.50) as (see [93] Eq. 4.46 and Eq. 4.47):

$$P_{cw} = \frac{1}{n} \sum_{w=d_{free}}^{\infty} T_w P_w \tag{3.25}$$

$$P_e = \frac{1}{n} \sum_{w=d_{free}}^{\infty} T'_w P_w \tag{3.26}$$

where T'_w is the number of non-zero bits identical to all the codewords with a weight of w which is estimated from the transfer function as explained in section 3.4.1.

3.4.1 Transfer Function and Weight Enumerators for Non-Binary Convolutional Codes

In this section, the procedure to enumerate the code weight will be explained with the help of an example. Figure 3.5 presents the signal flow graph of the β $\beta^2/1$ non-binary code. Hence, the power of D is the Hamming distance between the output

bits corresponding to each state transition and the output (00). With help from the signal flow graph, the transfer function can be evaluated. The figure below is obtained from the trellis diagram and shows paths diverging and re-merging to the all-zero state. X_a represents state 0, X_b stands for state 1, X_c is state β and X_d is state β^2.

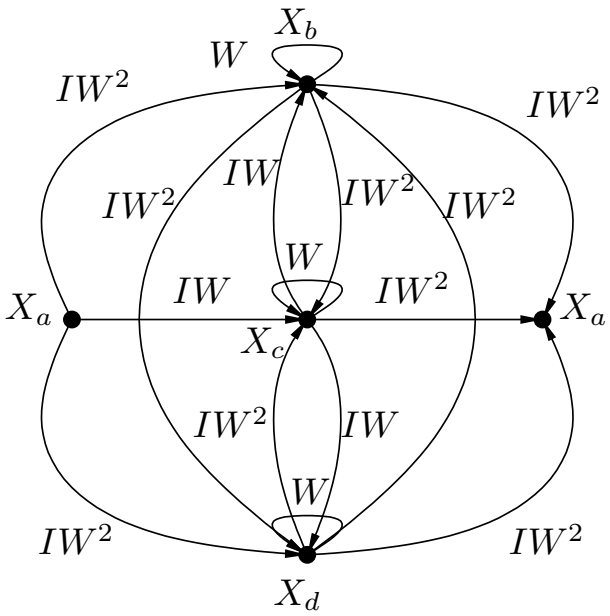

Figure 3.5: The signal flow-graph of the β $\beta^2/1$ non-binary convolutional code.

$$X_b = IW^2X_a + IWX_b + WX_c + IW^2X_d \tag{3.27}$$

$$X_c = IWX_a + IW^2X_b + WX_c + IW^2X_d \tag{3.28}$$

$$X_d = IW^2X_a + IW^2X_b + IWX_c + W^2X_d \tag{3.29}$$

$$X'_a = IW^2X_b + IW^2X_c + IW^2X_d \tag{3.30}$$

By subtracting 3.27 from 3.49, it is easy to find that $X_b = X_d$, then the transfer function is

$$X_b - X_d = (W - IW^2)X_b + (IW^2 - W)X_d$$
$$(1 - W + IW^2)X_b = (1 - W + IW^2)X_d$$
$$\therefore X_b = X_d.$$

The transfer function $T(I, W)$ of the signal-flow-graph in figure 3.5 is:

$$T(I,W) = \frac{X'_a}{X_a} = \frac{2W^2X_b + WX_c}{X_a} \tag{3.31}$$

$$T(I,W) = \frac{2IW^2X_b + I^2W^3X_a + I^2W^4X_b + IW^3X_c + I^2W^3X_b}{X_a}$$

$$= \frac{X_b}{X_a}(2IW^2 + I^2W^4 + I^2W^3) + I^2 + W^3 + \frac{X_c}{X_b}(IW^3) \tag{3.32}$$

We can show that:

$$X_b = \frac{IW^2X_a + IWX_c}{1 - W - IW^2} \tag{3.33}$$

and:

$$X_c = \frac{IWX_a + 2IW^2X_b}{1 - W} \tag{3.34}$$

Substituting X_c into X_b gives:

$$X_b = \frac{IW^2X_a}{1 - W - IW^2} + \frac{IW(IWX_a + 2IW^2X_b)}{(1 - W - IW^2)(1 - W)}$$

$$\therefore \frac{X_b}{X_a} = \frac{IW^2 - IW^3 + I^2W^2}{1 - 2W - IW^2 + W^2 + IW^3 - 2I^2W^3} \tag{3.35}$$

Likewise:

$$\frac{X_c}{X_a} = \frac{IW - IW^2 - I^2W^3 + 2I^2W^4}{1 - 2W - IW^2 + W^2 + IW^3 - 2I^2W^3} \tag{3.36}$$

By submitting (3.35) and (3.36) into (3.32), the transfer function can be written as:

$$T(I,W) = \frac{I^2W^3 + 2I^2W^4 + 2I^3W^4 - 2I^2W^5 - I^3W5 + 2I^3W^6}{1 - 2W - IW^2 + W^2 + IW^3 - 2IW^3} \tag{3.37}$$

Evaluating (3.37) gives:

$$T(I,W) = I^2W^3 + 4I^2W^4 + 7I^2W^5 + ... \tag{3.38}$$

Formula (3.38) indicates that there is one path of Hamming distance 3 symbols, four paths of distance 4 symbols and seven paths of distance 5 symbols etc.

In order to demonstrate the accuracy of equations 3.25 and 3.26 for both binary $(1, 7/5)_8$ and non-binary (β $\beta^2/1$ and $\beta 1\beta/\beta^2 1$) codes, T'_w for non-binary codes can be calculated as:

$$T'_w = \left[\frac{d}{dI}T'(I,W)\right]_{I=1} \tag{3.39}$$

$$T'_w = 2W^3 + 8W^4 + 14W^5..... \tag{3.40}$$

also the $T(I,W)$ and T'_w of the $\beta 1\beta/\beta^2 1$ code are evaluated manually from the trellis respectively as:

$$T(I,W) = (I^2 + I^3)W^4 + 0W^5 + (7I^3 + 4I^4)W^6..... \tag{3.41}$$

$$T'_w = 5W^4 + 0W^5 + 33W^6..... \tag{3.42}$$

While T'_w for the binary code can be found in [94]:

$$T'_w = W^5 + 4W^6 + 12W^7..... \tag{3.43}$$

Obviously the binary code has larger free distance as can be concluded from 3.40 and 3.43. This binary code has one path of Hamming distance 5 bits, four paths of distance 6 bits and twelve paths of distance 7 bits etc. Even though, the binary code has a large minimum free distance of 5 bits, the non-binary can actually correct more bit errors with a free distance of 3 symbols (i.e. 6 bits) on AWGN channels at higher SNRs.

Figures 3.7 and 3.6 illustrating the frame or codeword error rate P_{cw} (FER) and P_e (BER) of the $(1, 7/5)_8$ binary code and the β $\beta^2/1$ non-binary respectively.

The results show that there is a close agreement between the bounds and the simulation results for both binary and non-binary codes. The BER curves acquired by using equation (3.26) and the FER curves obtained by using equation (3.25) when K =100 , 1000, 5000 and 6000.

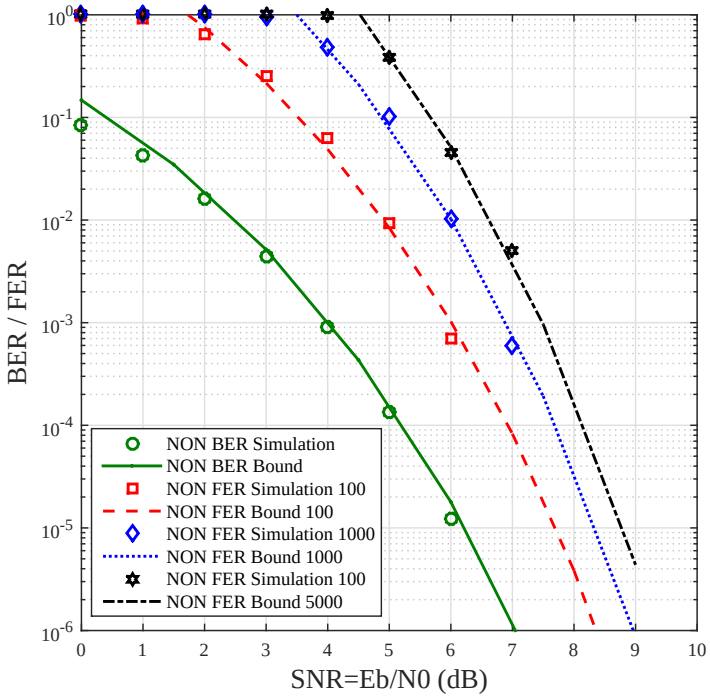

Figure 3.6: Theoretical BER and FER bounds of β $\beta^2/1$ Non-Binary convolutional code on AWGN channels.

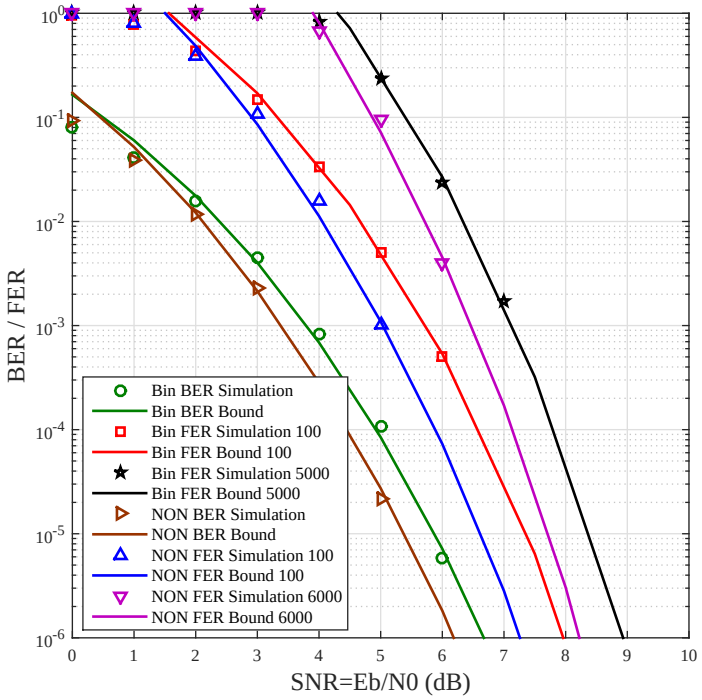

Figure 3.7: Theoretical BER and FER bounds of $(1, 7/5)_8$ binary and $\beta 1 \beta/\beta^2 1$ non-binary convolutional codes on AWGN channels.

3.5 $\beta\beta^2/1$ Non-Binary Turbo Encoder

A non-binary turbo encoder is the parallel concatenation of two non-binary RSC codes separated by an interleaver, denoted by Π, as shown in figure 3.8.

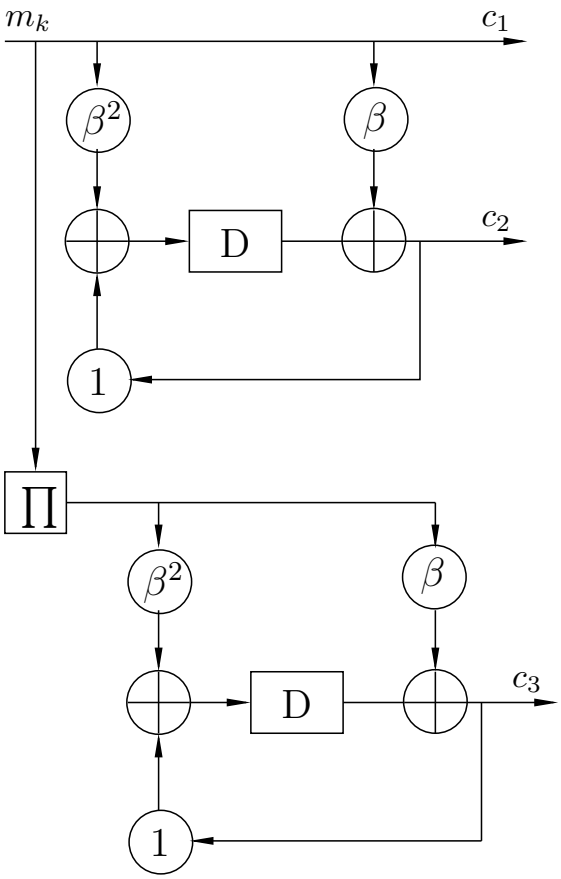

Figure 3.8: β $\beta^2/1$ Non-binary turbo encoder.

The length k input message m is encoded directly by the first component encoder, which produces the parity symbols c_2. At the same time, m is interleaved before being encoded by the second convolutional encoder, which produces the parity symbols c_3. The codeword is mapped to an M-PSK constellation X and then transmitted. At the receiver side the received signal Y is denoted by

$$Y = X + n, \tag{3.44}$$

where n is a sequence of $S\alpha S$ noise samples.

3.5.1 Non-Binary Turbo Decoder on Additive $S\alpha S$ Noise Channels

The turbo decoder consists of two Max-Log-Map component decoders in series. To incorporate the a priori LLR values, (3.13) and (3.48) are rewritten for both the Gaussian receiver and Cauchy receiver respectively.

$$\Gamma_t(S,S'))_{AWGN} = L_a(x) - \frac{1}{2\sigma^2}||\boldsymbol{Y}_1 - \boldsymbol{X}_1||^2 - \frac{1}{2\sigma^2}||\boldsymbol{Y}_2 - \boldsymbol{X}_2||^2 \tag{3.45}$$

$$\Gamma_t(S,S'))_{AWGN(Map)} = L_a(x) - \exp\left(\frac{((\boldsymbol{Y}_1 - \boldsymbol{X}_1)^2 + (\boldsymbol{Y}_2 - \boldsymbol{X}_2))^2}{2\sigma^2}\right) \tag{3.46}$$

$$\begin{aligned}\Gamma_t(S,S'))_{Cauchy(MaxLog)} = L_a(x) + \Bigg(&\log\frac{\gamma}{\pi[\gamma^2 + ||\boldsymbol{Y}_1 - \boldsymbol{X}_1||^2]} \\ &+ \log\frac{\gamma}{\pi[\gamma^2 + ||\boldsymbol{Y}_2 - \boldsymbol{X}_2||^2]}\Bigg),\end{aligned} \tag{3.47}$$

$$\begin{aligned}\Gamma_t(S,S'))_{Cauchy(Map)} = L_a(x) - \Bigg(&\frac{\gamma}{\pi[\gamma^2 + ||\boldsymbol{Y}_1 - \boldsymbol{X}_1||^2]} \\ &\times \frac{\gamma}{\pi[\gamma^2 + ||\boldsymbol{Y}_2 - \boldsymbol{X}_2||^2]}\Bigg),\end{aligned} \tag{3.48}$$

The extrinsic LLR values, L_e, passed on from one decoder to the other are obtained by subtracting the original LLR values of the message symbols and the a priori LLR values from the decoder LLR values $L(\hat{X}_t|Y_t)$.

$$L_e(X_t) = L(\hat{X}_t|Y_t) - L_a(X_t) - L_Y, \tag{3.49}$$

where L_Y is the LLR of the received message. Improvements in performance can be achieved with each iteration, but the improvements become less with each iteration until the decoder converges to an optimal performance.

3.6 Error Floor Bound for Non-Binary Turbo Codes

It is well known that the turbo code performance levels off rapidly at high SNRs. In addition, utilizing of the interleaver would make the Maximum Likelihood ML

decoding process very complex for turbo codes.

Since turbo codes are a set of two or more convolutional codes, the free distance of convolutional codes can help to bound the error floor of turbo codes. In this thesis we consider turbo codes with two parallel concatenated convolutional components encoders only.

To bound the slop of this drop off at high SNRs the following equation has been used by [95]:

$$Pb(binary) \leq \sum_{w=1}^{N} \sum_{v=1}^{\binom{N}{w}} \frac{w}{N} Q\left(\sqrt{\frac{2d_{wv}E_b}{N_0}}\right) \tag{3.50}$$

By using the same methodology for deriving equation (3.26), ultimately, the above equation can be modified to bound the error floor of non-binary turbo codes as:

$$Pb(nonbinary) \leq \frac{2^{k-1}}{2^k - 1} \sum_{w=1}^{N} \sum_{v=1}^{\binom{N}{w}} \frac{w}{N} Q\left(\sqrt{\frac{3d_{wv}E_b}{N_0}}\right) \tag{3.51}$$

Where, d_{wv} is the minimum weight of a sequence that is produced by different weight (w) input, the first summation is over w and the second summation is over the $\binom{N}{w}$ different weight input. It is obvious that $w = 1$ results in large values for d_{wv} and very low bit error probability, so that $w = 1$ will be neglected. Furthermore, When $w = 2$ the terms in 3.51 will generate a minimum weight due to the the interleaver, and the minimum d_{wv} can be bounded as:

$$d_{2v}(turbo) \geq min(2d_{2v}(conv.) - Æ) \tag{3.52}$$

where Æ is the weight of the padding sequence to terminate the first encoder and return to state (00). In this case, when $w = 2$ is the dominated value, equation (3.51) can be approximated to:

$$Pb \simeq \frac{2^{k-1}}{2^k - 1} \frac{2n_2}{N} Q\left(\sqrt{\frac{3d_{2v}E_b}{N_0}}\right) \tag{3.53}$$

where, n_2 is the number of weight-2 codewords that generate the minimum weight $d2v$. Hence, a similar argument can be applied to different weights of information sequences.

Figure 3.9 demonstrates the error floor bounds for binary and non binary turbo codes for different codes where the length (N) of the interleaver is 1500 symbols (for

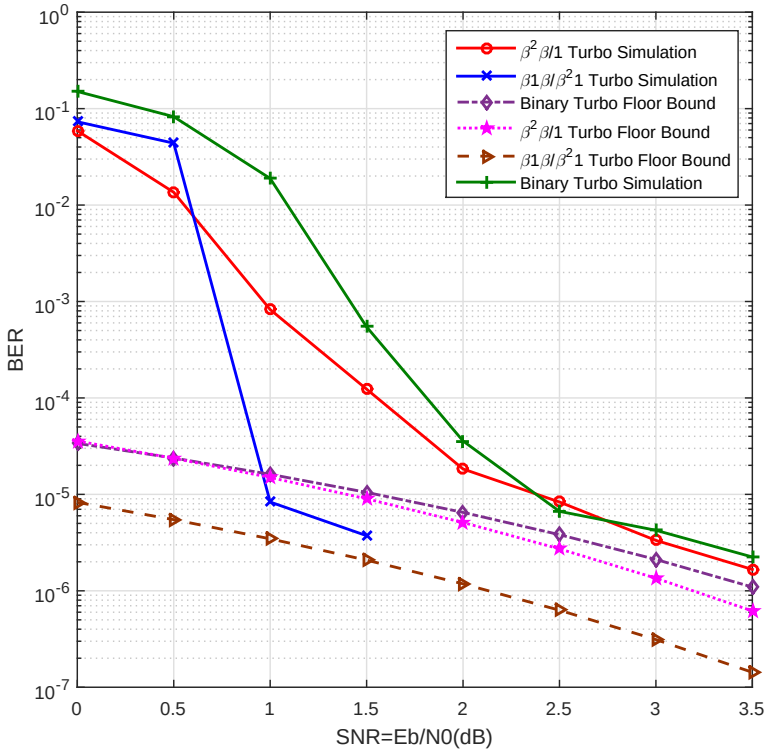

Figure 3.9: Error floor bound for β $\beta^2/1$, $\beta 1\beta/\beta^2 1$ non-binary turbo codes and $(1, 7/5)_8$ binary turbo codes.

non-binary) and 3000 bits (for binary code), using a pseudo-random interleaver, 5 iterations and code rate $R_c = \frac{1}{3}$ for all codes. There is a good agreement between the simulation results with the theoretical error floor bounds result as we can see in figure 3.9. It is clear that the BER drops rapidly to 10^{-5} in the region between 0-1.5 dB which known as a turbo cliff as mentioned before. After that sharp drop the performance curves start flattening and this region is known as a free distance asymptote or error floor region.

3.7 Simulation Results

3.7.1 Non-Binary and binary Convolutional Codes Simulation Results

We now compare the BER performance of a 4-state non-binary convolutional code defined in GF(4) and a 4-state binary convolutional code on impulsive noise channels, for various values of characteristic exponent (α). Simulation results of a rate $R = \frac{1}{2}$

binary convolutional with $(1, 7/5)_8$ RSC component encoders and message length m=2048 bits are compared with a rate $R = \frac{1}{2}$ non-binary turbo code with $\beta\beta^2/1$ RSC component encoders and message length of 1024 symbols (2048 bits). In addition, uncoded BER performance on impulsive noise, and a 16-states $\beta 1\beta/\beta^2 1$ non-binary convolutional code BER performance simulation is implemented.

Since we are working on $S\alpha S$ impulsive noise channels, a Cauchy receiver is used. Figure 3.10 illustrates how this receiver closely matches the optimal LLR (i.e. when LLR $\neq 0$), and also shows that the Cauchy is a good receiver for different values of α.

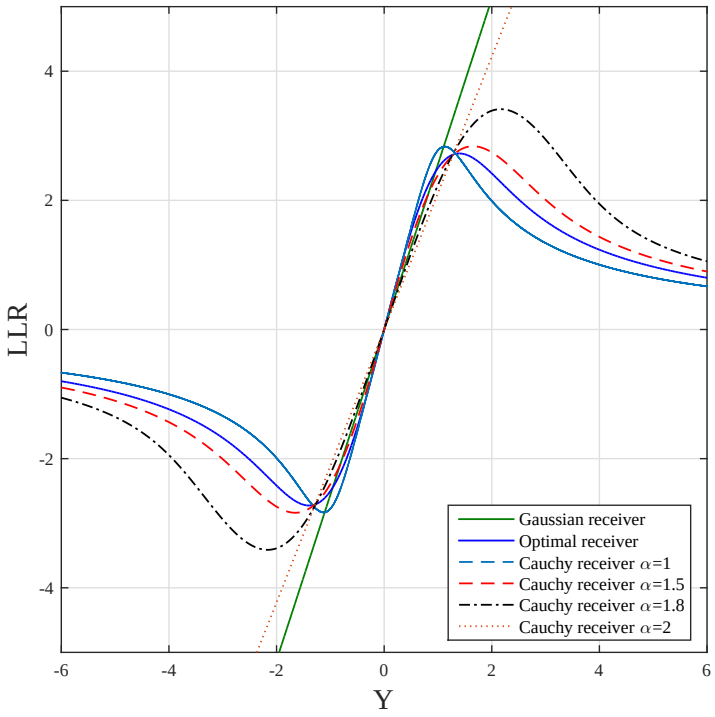

Figure 3.10: LLR demappers for the optimal and sub-optimal receivers (α=1, 1.5, 1.8 and 2 E_b/N_0=2 dB).

Figure 3.11 shows the bit error rate (BER) of all decoders when the channels are very impulsive (α=1). It can be clearly seen that the non-binary code achieves a great coding gain of 4 dB over the binary at low SNRs and 2 dB at high SNRs at the BER of 10^{-5}. In addition, both codes gained more than 40 dB over the uncoded system. This figure also shows that the performance of the 16-state $\beta 1\beta/\beta^2 1$ non-binary convolutional code achieved a gain of 2.5 dB over the $\beta\beta^2/1$ non-binary convolutional code.

Again, the BER for all coded and uncoded systems on impulsive noise channel are presented in Figure 3.12, when α=1.5. It is clear that the performance of the non-binary code exceeds the performance of the binary code by at least 4 dB at low SNRs region and 1.5 dB at high SNRs region at a BER of 10^{-5}. Meanwhile, both codes (non-binary and binary) have achieved 38.6 dB and 37.1 dB over the uncoded systems respectively. However, the 16-states non-binary code is achieving a significant gain of 2.4 dB, 4 dB, and 41 dB over the 4-stats non-binary, binary and uncoded system at a BER of 10^{-5} respectively.

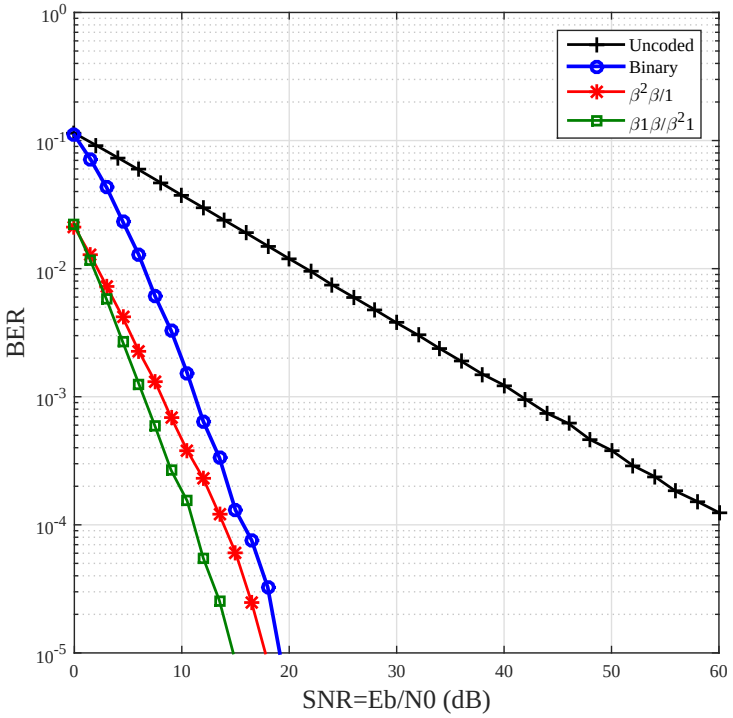

Figure 3.11: BER performance VS SNR for $(1, 7/5)_8$ binary convolutional code, $\beta\beta^2/1$, $\beta 1\beta/\beta^2 1$ non-binary codes, and uncoded system, when the characteristic exponent α=1.

In Figure 3.13, we also compare the performance of binary and non-binary convolutional codes when the channel is slightly impulsive (α=1.8). As before, the non-binary codes achieved a gain over the binary codes of 3 to 1.1 dB from low to high SNRs regions with a BER of 10^{-5}. Likewise, the 4-state and 16-state non-binary convolutional codes have about 29.75 dB and 32 dB gain over the uncoded system, respectively at BER of 10^{-5}.

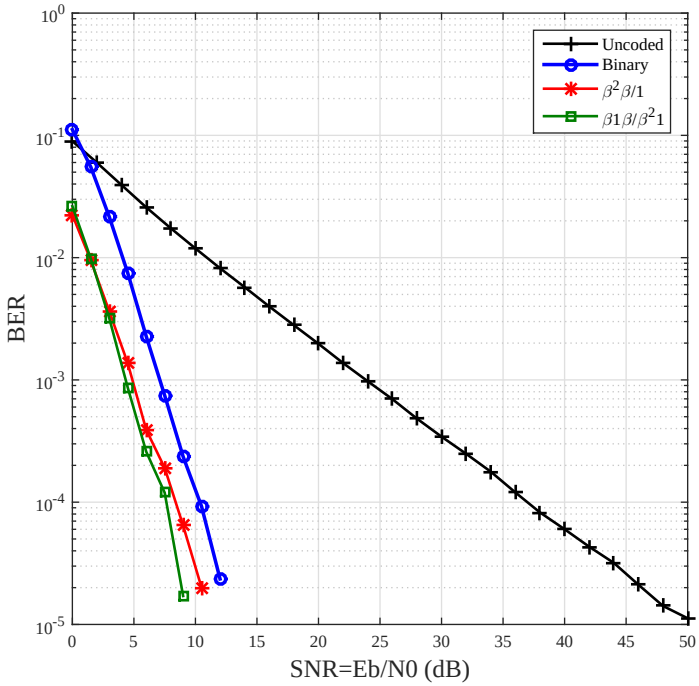

Figure 3.12: BER performance VS SNR for $(1, 7/5)_8$ binary convolutional code, $\beta\beta^2/1$, $\beta 1\beta/\beta^2 1$ non-binary codes, and uncoded system, when the characteristic exponent α=1.5.

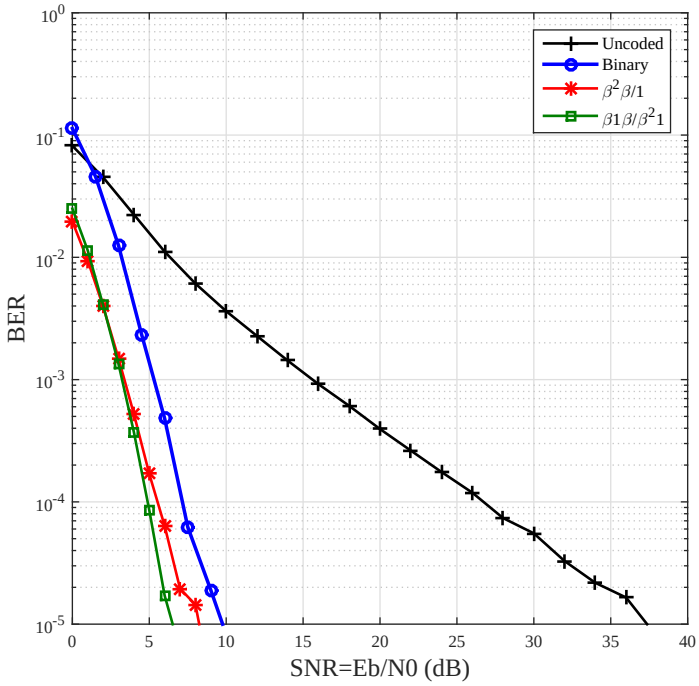

Figure 3.13: BER performance VS SNR for $(1, 7/5)_8$ binary convolutional code, $\beta\beta^2/1$, $\beta 1\beta/\beta^2 1$ non-binary codes, and uncoded system, when the characteristic exponent α=1.8.

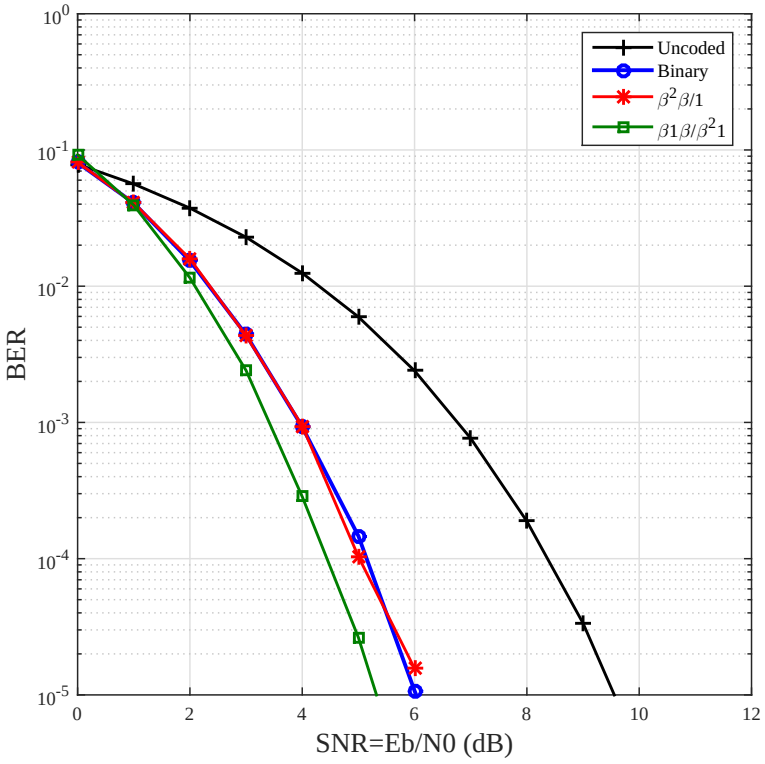

Figure 3.14: BER performance VS SNR for $(1, 7/5)_8$ binary convolutional code, $\beta\beta^2/1$, $\beta 1\beta/\beta^2 1$ non-binary codes, and uncoded system, when the characteristic exponent α=2.

Finally, when the channel is Gaussian,(i.e. $\alpha = 2$), as shown in figure 3.14 the performance for both decoders match. Again both systems are showing significant coding gains over the uncoded system by 3.6 dB at BER of 10^{-5}. In addition, the 16-state non-binary convolutional code performs slightly better than the 4-state non binary code by 0.6 dB and by 4.1 dB over the uncoded system with a BER of 1×10^{-5}.

3.7.2 Non-Binary and Binary Turbo Codes Simulation Results

In this section, the comparison of the BER performance of a non-binary turbo code defined in GF(4) and a binary turbo on $S\alpha S$ noise channels, for various values of the characteristic exponent (α) is carried out. Simulation results of a rate $R = \frac{1}{3}$ binary turbo code with $(1, 7/5)_8$ RSC component encoders and message length of m=2048 bits are compared with a rate $R = \frac{1}{3}$ non-binary turbo code with $\beta\beta^2/1$ RSC component encoders and message length of 1024 symbols (2048 bits). The

maximum iterations for both decoders is set to five.

Additionally, the Cauchy receiver is employed for all systems, as explained in the previous section 3.7.1 and Figure 3.10.

Figure 3.15 shows the bit error rate of the 1st and 5th iterations for both decoders when the channels are very impulsive (α=1). It can be clearly seen that the non-binary turbo code achieves a greater coding gain of 1.8 dB over the binary turbo code at low SNRs and 1.1 dB at high SNRs.

The BERs for both decoders are presented in figure 3.16 when α=1.5. It is clear that the performance of the non-binary code exceeds the performance of the binary code by 2 dB at low SNRs region and 1.7 dB at high SNRs region with a BER of 10^{-5}. In figure 3.17, we also compare the performance of binary and non-binary turbo codes when the channel is slightly impulsive (α=1.8). As before, the non-binary codes achieves a gain over the binary codes of 2 to 0.8 dB from low to high SNRs regions with a BER of 6.6×10^{-5}.

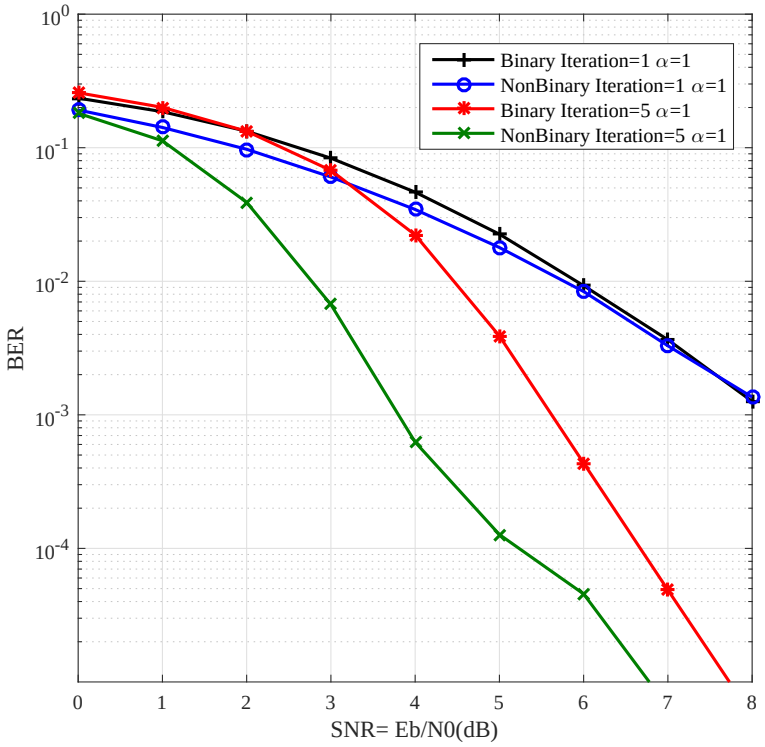

Figure 3.15: BER performance VS SNR for $(1, 7/5)_8$ binary turbo code and $\beta\beta^2/1$ non-binary turbo codes, when the characteristic exponent α=1.

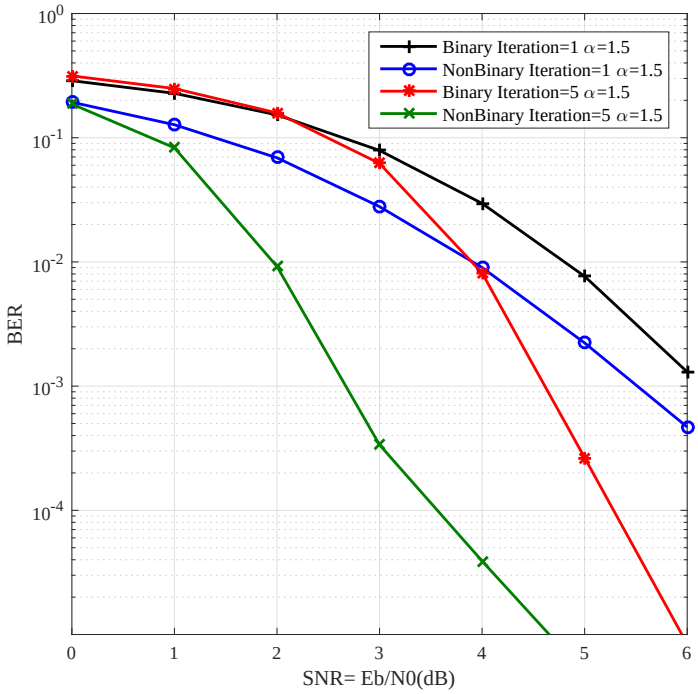

Figure 3.16: BER performance VS SNR for $(1, 7/5)_8$ binary turbo code and $\beta\beta^2/1$ non-binary turbo codes, when the characteristic exponent α=1.5.

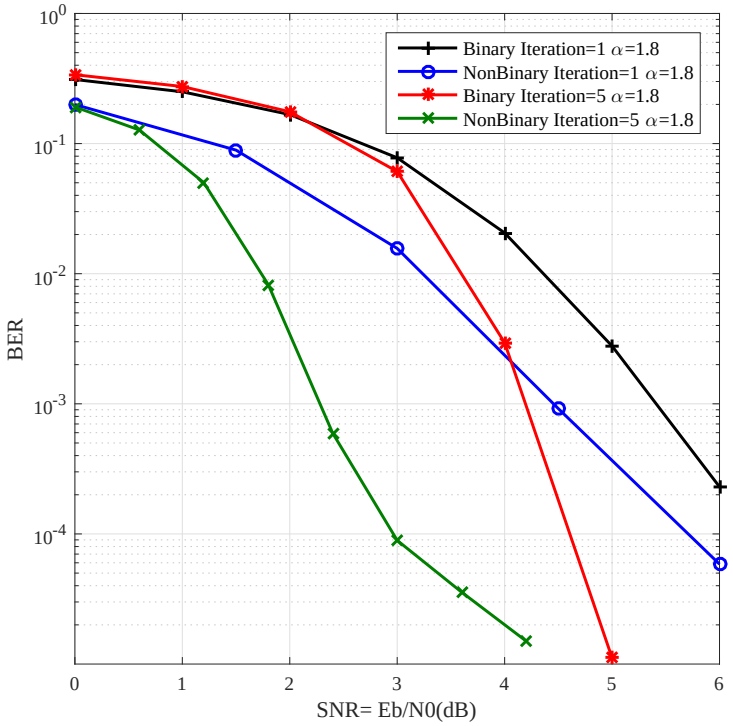

Figure 3.17: BER performance VS SNR for $(1, 7/5)_8$ binary turbo code and $\beta\beta^2/1$ non-binary turbo codes, when the characteristic exponent α=1.8.

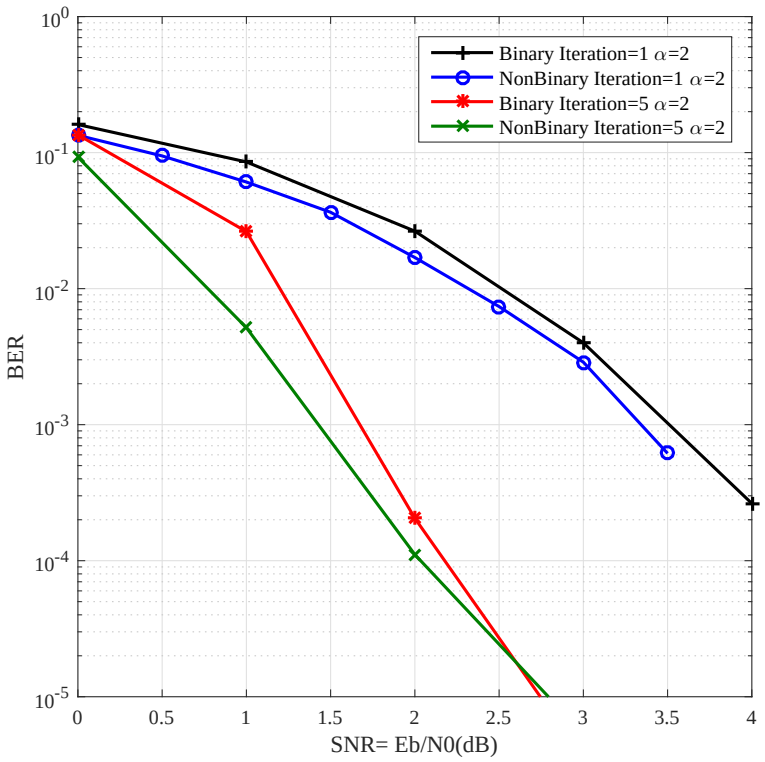

Figure 3.18: BER performance VS SNR for $(1, 7/5)_8$ binary turbo code and $\beta\beta^2/1$ non-binary turbo codes, when the characteristic exponent α=2.

Moreover, when the channel is a Gaussian channel as shown in figure 3.18 the performance for both decoders are close, but the non-binary turbo code performs slightly better by 0.5-0.1 dB for different SNRs with a BER of 10^{-5}. Finally, figures 3.15,3.16 and 3.17 show that the non-binary codes need less iterations to converge to the optimal performance than binary codes on both Gaussian and non-Gaussian channels.

3.8 Summary

In this chapter, the construction of convolutional non-binary codes has been explained in detail and the main differences with binary convolutional codes have been highlighted. Also, the representation of non-binary convolutional codes ($\beta\beta^2/1$ and $\beta 1\beta/\beta^2 1$) such as the state tables, trellis diagrams and the signal flow-graph is presented in sections [3.2-3.4].

It is very important to mention that the difference in the construction and the representation between the binary and non-binary convolutional codes means the

original Max-Log MAP decoding algorithm is not valid in the case of decoding non-binary codes. However, sections 3.3 and 3.5.1 show the modified decoding equations and these changes had been justified and proved as can be seen in the performance analysis figures 3.6 and 3.7. On the other hand, the analysis of non-binary codes on $GF(4)$ has been investigated and the required equations have been derived.

Even though the interleaver contributes a significant complexity to the ML decoding algorithm, the bit error rate curves on AWGN channels have been bounded successfully as section 3.6 illustrates. The proposed performance analysis of non-binary trellis codes have been examined for two different numerical examples and the results show a good agreement between the theoretical analysis and the simulation results.

Additionally, in this chapter, the performance of non-binary trellis codes on additive impulsive noise channels has been investigated. The turbo encoder and decoder structures have been introduced and the max log-MAP algorithm for non-binary codes employed as the component non-binary decoders has been explained. Our simulation results show that the proposed non-binary trellis code achieves notable coding gains compared to binary trellis codes when the channel is impulsive. Furthermore, it can be observed that the non-binary turbo decoder converges to the optimal performance in less iterations than the binary turbo decoder. This initial investigation has shown that there is much promise for non-binary trellis codes on impulsive noise channels and there is great scope for further research in this area.

Chapter 4

Non-Binary Trellis-Coded OFDM-PLC System in the Presence of Middleton Class A Impulsive Noise

4.1 Introduction

Power-line communication utilizes the established electrical grid, but since power networks are not designed for communication services, there are many factors that make reliable communication over transmission lines challenging. These include: attenuation, impulsive noise and multipath frequency selectivity. Communication over the transmission line is most likely to suffer from impulsive noise due to electromagnetic interference and this is commonly modelled by the Middleton Class A probability density function [5].

The power-line communication channel causes information-bearing signals to be affected by impulsive noise and the effects of the multipath fading. To help mitigate these effects, we propose the employment of non-binary trellis codes, since non-binary error-correcting codes generally promise an enhanced performance in such harsh environments. The negative effect of impulsive noise is illustrated in this chapter and the BER performance of single carrier and multi-carrier uncoded, binary and non-binary convolutional codes on impulsive noise channels over realistic PLC multi-path frequency selective channels will be presented.

Furthermore, we investigate the performance of non-binary turbo codes on PLC channels that exhibit frequency selectivity with additive Middleton Class A noise and compare with a comparable binary turbo coded PLC system. In order to reduce the effect of multipath and impulsive noise, orthogonal frequency-division multiplexing with non-linear receivers (blanking and clipping) has been employed. The system is examined on extremely impulsive channels where the value of the the noise ratio Γ is 0.01 (i.e $\frac{\sigma_i}{\sigma_g} = 100$), see section 2.2.3 for more details.

Finally, this chapter is organized as follows: Section 4.2.1, gives brief knowledge on Middleton class A noise channels and the BER performance of single carrier (SC) coded binary and non-binary convolutional systems on additive Middleton class A noise is presented. In section 4.2.2, the multi-path frequency-selective realistic channels that is used in power line communication is explained in detail and the performance analysis binary and non-binary convolutional coded OFDM-PLC systems are introduced. In sections 4.3.1 and 4.3.2, the encoding and decoding procedure for non-binary turbo codes on Middleton class A noise channels is illustrated. While, in section 4.4 the non-binary turbo coded OFDM-PLC system with Non-linear preprocessing is investigated. Simulation results of the BER performance of non-binary and binary turbo coded OFDM-PLC systems for various values of Γ and on different realistic PLC channels are presented in section 4.5. Finally, section 4.6 presents the chapter conclusions. The contribution of this chapter is published in the 25th European signal processing conference [78].

4.2 Channel Model

4.2.1 Middleton Class A Distributions

The probability of an error for Middleton Class A noise when employing M-ary phase-shift keying (MPSK) is given in [6] as

$$P_e = (1-A)\frac{M-1}{M}Q\left(\sqrt{\frac{E_b}{\sigma_G^2}}\right) + A\frac{M-1}{M}Q\left(\sqrt{\frac{E_b}{\sigma_G^2(1+\frac{1}{A\Gamma})}}\right), \tag{4.1}$$

where M is the order of the PSK modulation, and E_b is the bit energy.

4.2.1.1 Performance of SC Binary and Non-Binary Convolutional Codes on Middleton Class A Noise Channels

In order to understand the impulsive behavior of Middleton class A channels, the BER of single carrier BPSK uncoded systems on impulsive noise channel is implemented in figure 4.1. This figure demonstrates equation (4.1) and shows the effect of different values of A and Γ on the system performance. It is very clear that the performance degrades as the value of Γ decreases and the SNR needs to be about 50 dB to reach a BER of 10^{-5} when $\Gamma = 0.001$, while the SNR needs to be 45 dB to achieve the same BER when $\Gamma = 0.01$.

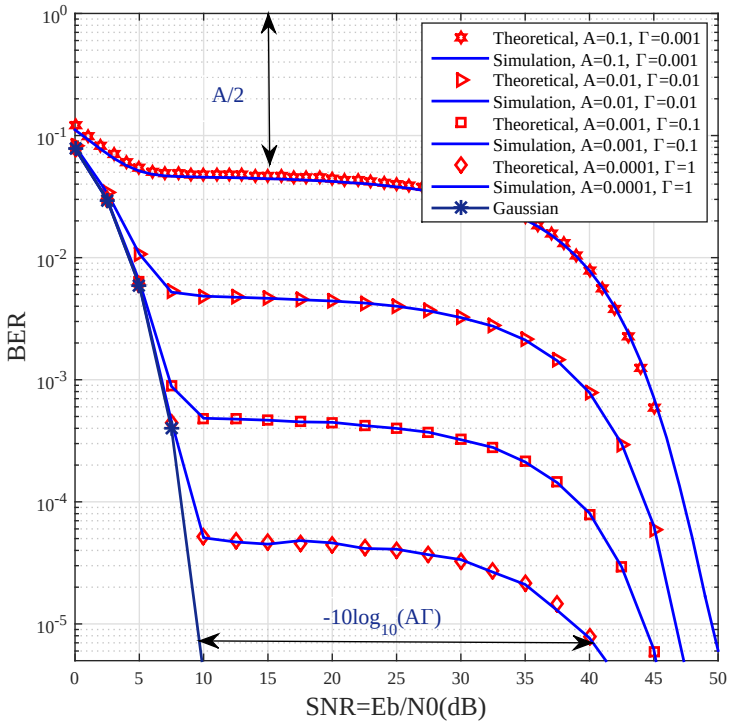

Figure 4.1: Theoretical and simulation BER Performance of single carrier uncoded BPSK system on additive Middleton class A noise channel for various values of A and Γ vs. SNR in dB.

It is worth mentioning that the value of A effects the performance negatively as it becomes larger as we can see in the curve when $A = 0.1$ and the BER almost matches the Gaussian BER when $A = 0.001$ and $\Gamma = 1$, see figure 2.8.

However, for single carrier convolutional coded systems, some modifications need to be made to the conventional Max-Log algorithm since the channel PDF is now

Middleton class A. The modified LLR for a BPSK binary convolutional system is estimated as

$$\begin{aligned} LLR(c=1|\boldsymbol{y}) &= \ln \frac{P(c=1|\boldsymbol{y})}{P(c=0|\boldsymbol{y})} \\ &= \ln \frac{\sum_{m=0}^{\infty} \frac{e^{-A}A^m}{m!} \cdot \frac{1}{\sqrt{2\pi\sigma_m^2}} \exp\left(-\frac{|y-1|^2}{2\sigma_m^2}\right)}{\sum_{m=0}^{\infty} \frac{e^{-A}A^m}{m!} \cdot \frac{1}{\sqrt{2\pi\sigma_m^2}} \exp\left(-\frac{|y+1|^2}{2\sigma_m^2}\right)} \end{aligned} \tag{4.2}$$

and for BPSK Non-binary convolutional system as

$$\begin{aligned} LLR^z(c=z|\boldsymbol{y}) &= \ln \frac{P^z(c=z|\boldsymbol{y})}{P^z(c=0|\boldsymbol{y})} \\ &= \ln \frac{\sum_{m=0}^{\infty} \frac{e^{-A}A^m}{m!} \cdot \frac{1}{\sqrt{2\pi\sigma_m^2}} \exp\left(-\frac{|y-z|^2}{2\sigma_m^2}\right)}{\sum_{m=0}^{\infty} \frac{e^{-A}A^m}{m!} \cdot \frac{1}{\sqrt{2\pi\sigma_m^2}} \exp\left(-\frac{|y+1|^2}{2\sigma_m^2}\right)} \end{aligned} \tag{4.3}$$

The state transitions parameter Γ is calculated for binary convolutional by

$$\begin{aligned} \Gamma(S,S')_{Binary} &= \ln \sum_{m=0}^{\infty} \frac{e^{-A}A^m}{m!} \cdot \frac{1}{\sqrt{2\pi\sigma_m^2}} \exp\left(-\frac{|\boldsymbol{y}-\boldsymbol{c}|^2}{2\sigma_m^2}\right) \\ &+ \ln \sum_{m=0}^{\infty} \frac{e^{-A}A^m}{m!} \cdot \frac{1}{\sqrt{2\pi\sigma_m^2}} \exp\left(-\frac{|\boldsymbol{y}_1-\boldsymbol{c}_1|^2}{2\sigma_m^2}\right) \end{aligned} \tag{4.4}$$

where c is the modulated binary input and it takes values (1,-1), and c_1 are the modulated binary parity check bits. While for non-binary convolutional system Γ will be evaluated by

$$\begin{aligned} \Gamma(S,S')_{Non-binary} &= \ln \sum_{m=0}^{\infty} \frac{e^{-A}A^m}{m!} \cdot \frac{1}{\sqrt{2\pi\sigma_m^2}} \exp\left(-\frac{|\boldsymbol{y}-\boldsymbol{z}|^2}{2\sigma_m^2}\right) \\ &+ \ln \sum_{m=0}^{\infty} \frac{e^{-A}A^m}{m!} \cdot \frac{1}{\sqrt{2\pi\sigma_m^2}} \exp\left(-\frac{|\boldsymbol{y}_1-\boldsymbol{z}_1|^2}{2\sigma_m^2}\right) \end{aligned} \tag{4.5}$$

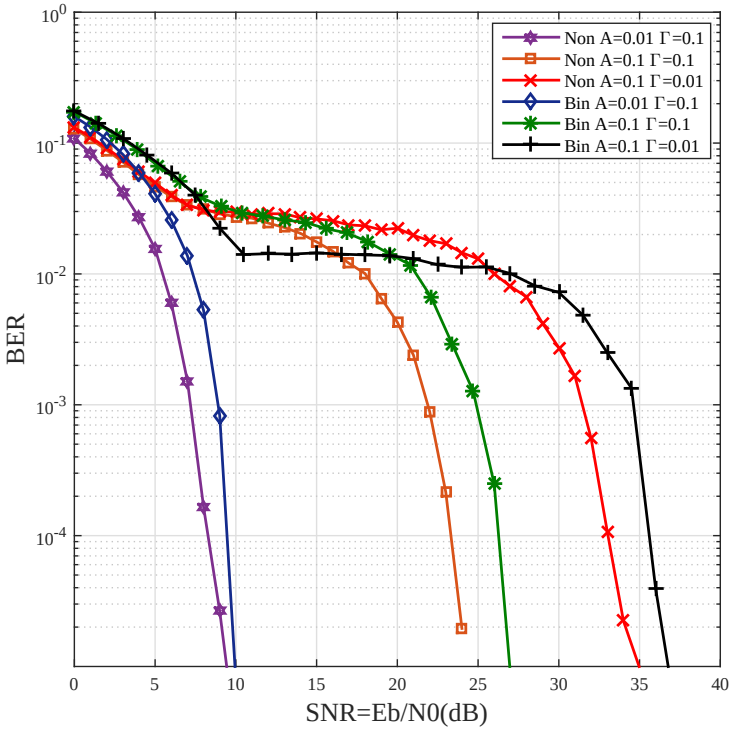

Figure 4.2: Simulation BER Performance of single carrier binary and non-binary BPSK systems on additive Middleton class A noise channel for different values of A and Γ vs. SNR in dB.

where z and z_1 are the mapped input non-binary symbol and the parity check symbol respectively and take values of $[(1,1),(-1,1),(1,-1),(-1,-1)]$.

The simulation results of implementing the modified Max-Log map are displayed in figure 4.2.

The figure above shows that the performance of the non-binary convolutional code is better than the binary code in all cases. In addition, the gain of the non-binary code over the binary code is increasing as the channel become more impulsive. For instance, when the channel is not very impulsive (i.e when $A = 0.01$ and $\Gamma = 0.1$) the non-binary code has only about 1 dB gain at BER=10^{-5}, while it is about 2.5 dB when the channel is very impulsive (i.e $A = 0.1$ and 0.01) at the same BER.

4.2.1.2 Performance of MC Binary and Non-binary Convolutional Codes on Middleton Class A Noise Channels

OFDM is an effective technique that is used to reduce the impact of multi-path channels which will be explained in the next section. The usage of OFDM is to mitigate the effect of strong impulsive noise by spreading the noise energy equally

over all symbols. The OFDM modulator applies an inverse fast Fourier transform (IFFT) to generate a complex baseband OFDM signal as

$$x(t) = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \boldsymbol{X}_k e^{\frac{j2\pi kt}{T_s}}, \qquad 0 < t < T_s, \tag{4.6}$$

where X_k is the data after the mapping process, N is the number of sub-carriers, which is equal to the number of the transmitted signals and T_s is the active symbol interval.

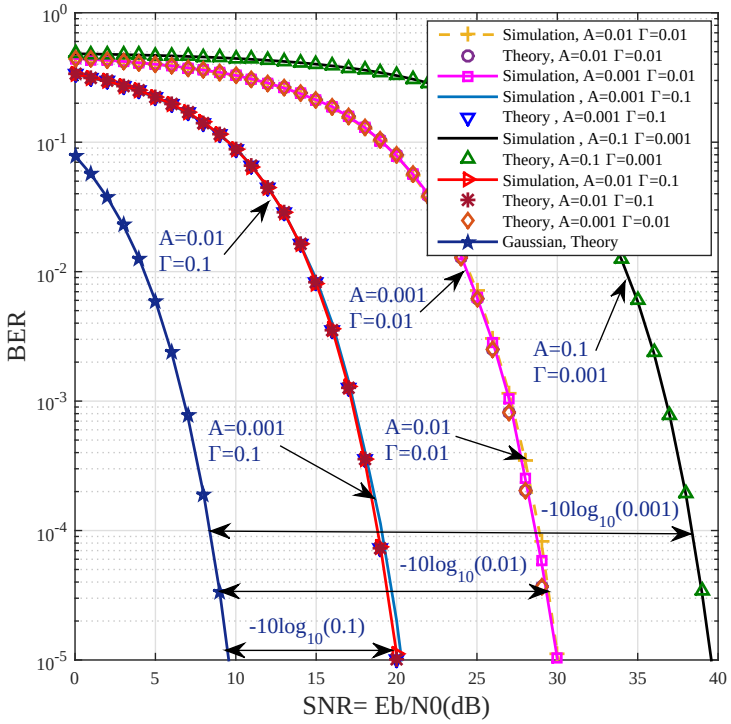

Figure 4.3: Theoretical and simulated BER Performance of multi-carrier uncoded BPSK system on additive Middleton class A noise channel for variant values of A and Γ vs. SNR in dB.

Figure 4.1 and figure 4.3 show the idea of spreading the noise energy equally on every symbol and the BER curves look like AWGN performance curves, but with different variance depending on the values of A and Γ [75].

In other words, the variance σ^2 of Middleton Class A noise in the frequency domain (after OFDM) can be approximated as a Gaussian distribution $\mathcal{N}(0, \sigma_Z)$ by:

$$\sigma_Z^2 = \frac{1}{N} \sum_{n=0}^{N-1} \sigma_z^2 = \sigma_G^2 \left(1 + \frac{1}{\Gamma}\right). \tag{4.7}$$

Recalling equation (1.1) and substituting σ_Z^2 as the noise variance, the theoretical

analysis of OFDM Middleton class A noise is calculated as

$$P_e = Q\left(\sqrt{\frac{2E_b}{N_0}}\right) = Q\left(\sqrt{\frac{E_b}{\sigma_Z^2}}\right) \tag{4.8}$$

Equation (4.8) is used to show an exact BER performance bound for different values of impulsive parameters in figure 4.3. As we can see in this figure, the value A does not affect the performance of PLC systems that utilize OFDM.

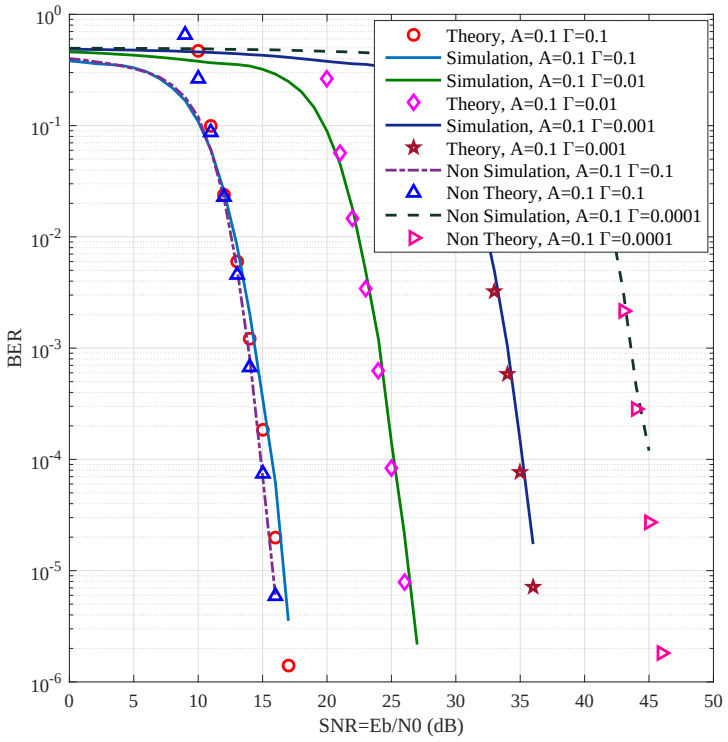

Figure 4.4: Theoretical and simulation BER Performance of multi-carrier $(1, 7/5)_8$ binary and $\beta 1 \beta/\beta^2 1$ non-binary convolutional codes on additive Middleton class A noise channel for different values of A and Γ vs. SNR in dB.

In the same way, the performance analysis of coded OFDM binary and non-binary convolutional codes can be estimated by substituting the new variance σ_Z^2 into equations (3.22) and (3.23) as

$$P_w(binary) = Q\left(\sqrt{\frac{wR_cE_b}{\sigma_Z^2}}\right). \tag{4.9}$$

$$P_w(non-binary) = \frac{2^{k-1}}{2^k-1} Q\left(\sqrt{\frac{3wR_cE_b}{2\sigma_Z^2}}\right). \tag{4.10}$$

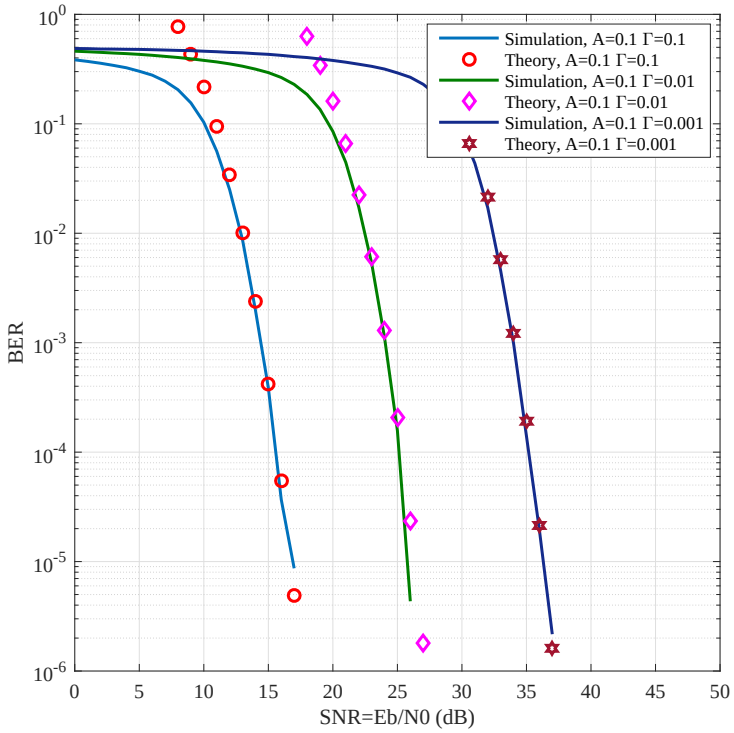

Figure 4.5: Theoretical and simulation BER Performance of multi-carrier $\beta\beta^2/1$ non-binary code on additive Middleton class A noise channel for different values Γ vs. SNR in dB.

In order to validate this assumption and to examine the accuracy of these equations in the matter of coded systems, binary and non-binary convlutional coded OFDM on impulsive noise have been implemented in figures 4.4 and 4.5. The message length is $k = 2048$ bits in the binary case and $k = 1024$ symbols for the non-binary codes, with a rate of $R_c = \frac{1}{2}$ and the cyclic prefix length is $CP = 256$. Figure 4.4 illustrates the simulation and theoretical BER of the $(1, 7/5)_8$ binary and $\beta 1\beta/\beta^2 1$ non-binary convolutional codes, while figure 4.5 shows the performance analysis of $\beta\beta^2/1$ non-binary convolutional code. Both figures show a good agreement between the simulation and the theoretical results for different values of A and Γ.

4.2.2 The Multipath Model for the Power-Line Channel

The behavior of a PLC multipath channel can be described by its frequency response as

$$H(f) = \sum_{i=0}^{L-1} g_l e^{-(a_0+a_1 f^k)d_i} e^{2\pi f \frac{d_i}{v_p}}, \tag{4.11}$$

where L is the number of paths, g_i is the weighting factor, a_0 and a_1 are attenuation parameters, d_i is the path length in meters (m), $k \in [0.5,1]$ is the respective attenuation of an echo and v_p is the phase velocity. This is calculated as

$$v_p = \frac{c_0}{\sqrt{\varepsilon_r \mu_r}}, \tag{4.12}$$

where c_0 is the speed of light, ε_r is the dielectric constant and μ_r is the permeability of the metal.

Table 4.1: 4 multi-path channel parameters

Path Parameters		
i	d_i/m	g_i
1	200	0.64
2	222.4	0.38
3	244.8	-0.15
4	267.5	-0.05

In this chapter, we use realistic 4 and 15 path channels from [70] and the paths parameters for both channels are shown in table 4.1 and 4.2. In addition the attenuation parameters a_0 and a_1 for both channels when $k = 1$ are 0 and $7.8 \times 10^{-10} s/m$ respectively. Furthermore, figures 4.6 and 4.7 illustrate the magnitude of the transfer function and the frequency response for both channels when the bandwidth ranges between 5KHz - 20MHz. By considering all these data the values of the multi-path taps of the multi-path channels are [27, 29, 32, 35] and [12, 14, 15, 19, 20, 27, 35, 43, 55, 65, 76, 99, 128, 151, 167] respectively. Obviously, the attenuation caused by the 15 paths channel is greater than the attenuation caused by the 4 paths channel, by a value of 30dB at 20MHz.

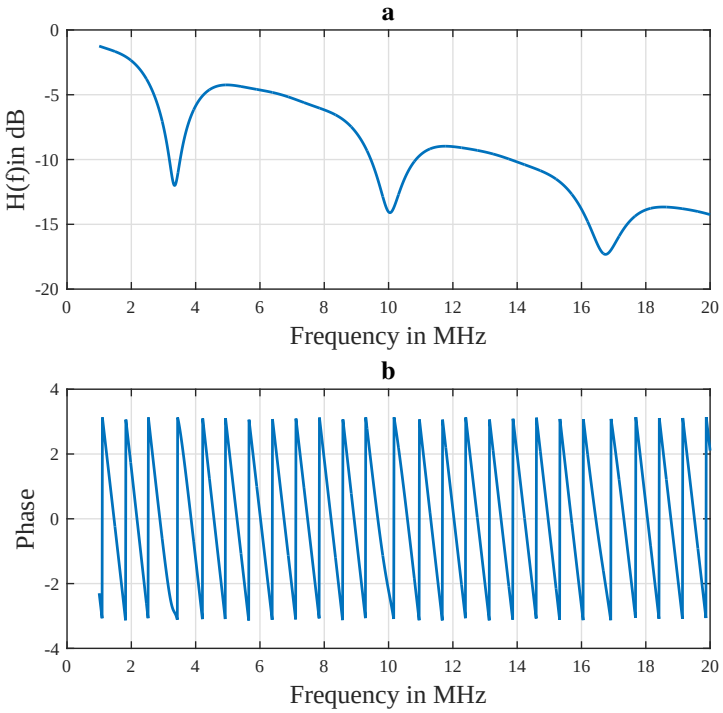

Figure 4.6: Frequency and phase response of the realistic PLC multipath channels. a) Frequency response for 4 path PLC channel. b) Phase response for 4 path PLC channel.

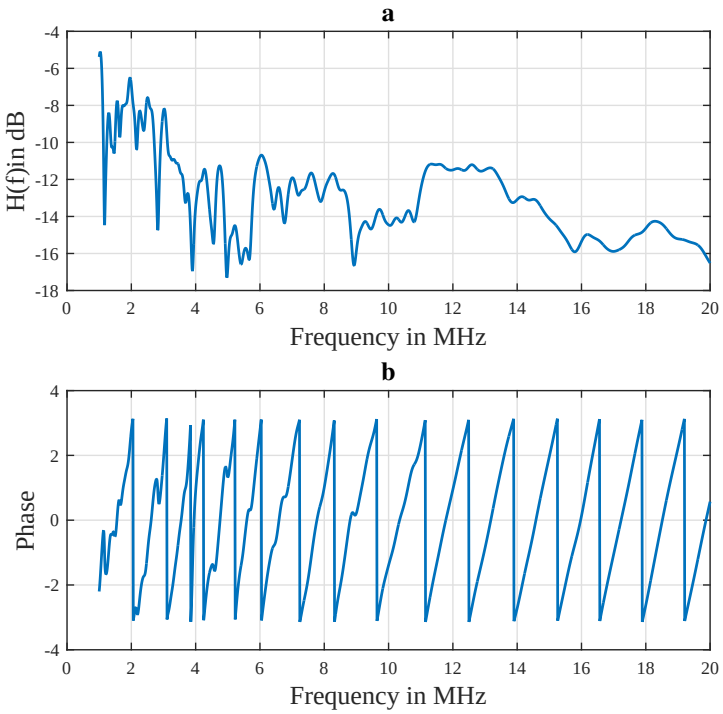

Figure 4.7: Frequency and phase response of the realistic PLC multipath channels. a) Frequency response for 15 path PLC. channel b) Phase response for 15 path PLC channel.

Table 4.2: 15 multi-path channel parameters

Path Parameters		
i	d_i/m	g_i
1	90	0.029
2	102	0.043
3	113	0.103
4	143	-0.058
5	148	-0.045
6	200	-0.040
7	260	0.038
8	322	-0.038
9	411	0.071
10	490	-0.035
11	567	0.065
12	740	-0.055
13	960	0.042
14	1130	-0.059
15	1250	0.049

4.2.2.1 Performance Analysis of Coded Binary and Non-Binary Convolutional OFDM-PLC Systems

In order to illustrate the accuracy of the proposed methodology of implementing the BER bounds, first we will introduce the performance analysis of uncoded OFDM-PLC system that is showing in figure 2.9.

Basically, equation (2.20) can be considered for Middleton class A noise channel on frequency selective Rayleigh fading channel by substituting σ_Z in equation (4.7) into 2.22.

Figure 4.8 displays a perfect match between the simulation results and the theoretical for both channels and for all different values of A and Γ. Furthermore, the impact of different PLC channels can be observed clearly and the degradation in the BER performance caused by the 15 multi-path frequency selective channels is significant compared to the effect of the 4 multi-path channel by at least 10 dB for all scenarios.

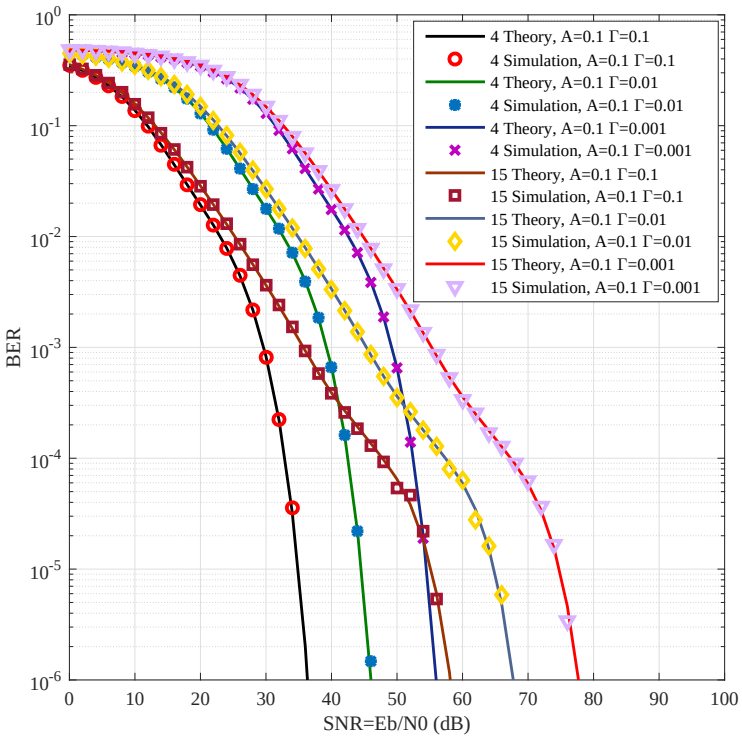

Figure 4.8: BER performance analysis of uncoded OFDM-PLC systems for various values of Middleton class A noise parameters and 4 and 15 path channels vs. SNR in dB

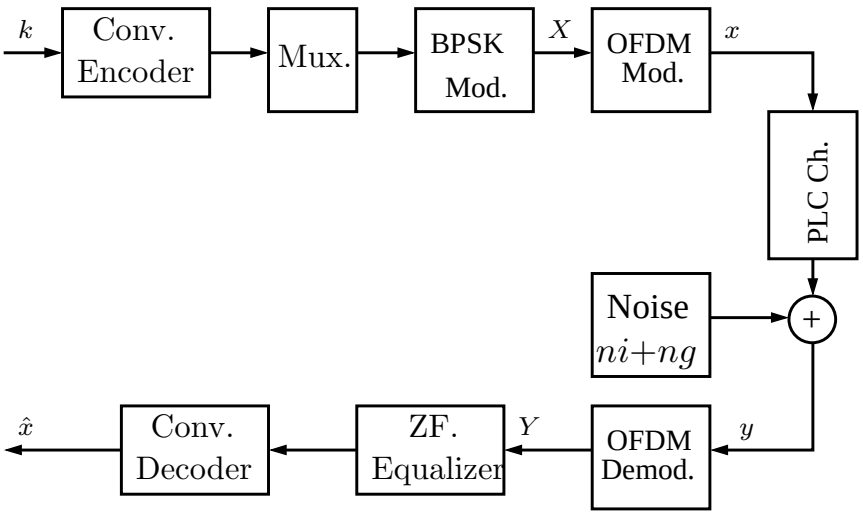

Figure 4.9: Convolutional coded OFDM-PLC system diagram.

Figure 4.9 illustrates the block-diagram of a general convolutional coded OFDM-PLC system with ZF equalizer. Basically, a k length data will pass through a rate half convolutional encoder and then the output message and parity bits or symbols will be mixed into one stream of data. This stream will be mapped by a BPSK

mapper and IFFT will be applied to the mapper output and the resulting signal x will be transmitted via PLC channels.

At the receiver side, a FFT will be applied to the received signal y and then a ZF equalizer will be used to eliminate the multi-path frequency selective impact. Ultimately, the equalizer output will pass through a Max-Log MAP convolutional decoder to obtain $\hat{x}$.

Again, by using the same procedure of the uncoded OFDM-PLC systems, the performance analysis of coded binary and non-binary convolutional OFDM-PLC systems can be accomplished. In another word, the pairwise error probability P_w of binary and non-binary convolutional codes in equations 4.9 and 4.10 are written respectively as

$$P_w(binary) = \frac{1}{K}\sum_{n=1}^{K} Q\left(\sqrt{\frac{wR_cE_b}{\sigma^2}}\right). \tag{4.13}$$

$$P_w(non-binary) = \frac{2^{k-1}}{2^k-1}\frac{1}{K}\sum_{n=1}^{K} Q\left(\sqrt{\frac{3wR_cE_b}{2\sigma^2}}\right). \tag{4.14}$$

By submitting the new P_w for both codes into equation (3.26), the probability of error P_e can be estimated as shown in the figures 4.10, 4.11 and 4.12.

Figure 4.10 compares the analytical results with the simulated results of coded $(1, 7/5)_8$ binary code and $\beta 1\beta/\beta^2 1$ non-binary code for several values of Γ over 4 multi-path frequency selective channel. This figure shows an excellent agreement between the analytical and the simulated results. While, figure 4.11 illustrates the performance of coded $(1, 7/5)_8$ binary and $\beta 1\beta/\beta^2 1$ non-binary code for various values of Γ over 15-tap multi-path channel. Clearly, this figure shows the accuracy of the performance analysis of coded OFDM-PLC system when the channel is extremely impulsive and the number of paths is 15.

Figure 4.12 demonstrates the BER performance analysis of the coded $\beta\beta^2/1$ non-binary convolutional OFDM-PLC system on different realistic PLC channels. This figure shows a very tight performance bound for all cases and this reflects the precision of the above equations.

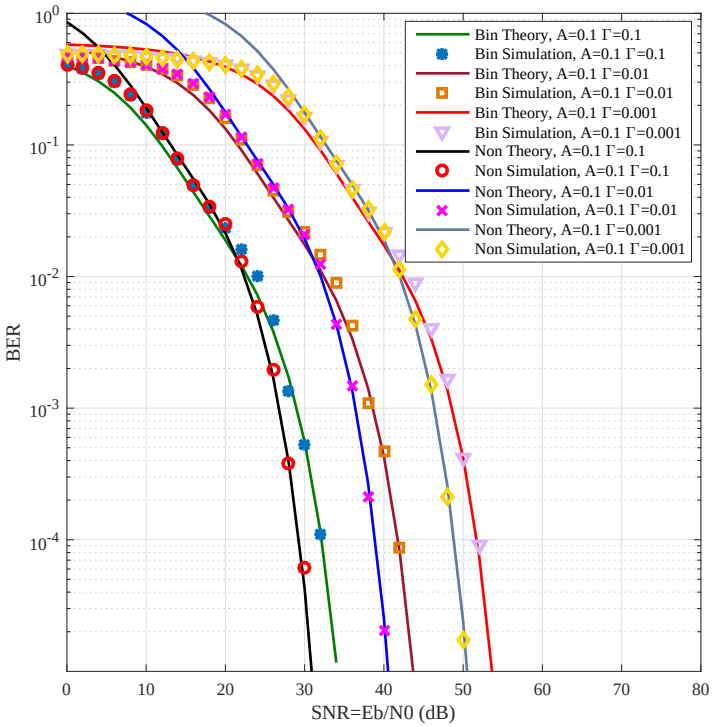

Figure 4.10: BER performance analysis of coded $(1, 7/5)_8$ binary and $\beta 1\beta/\beta^2 1$ non-binary convolutional OFDM-PLC systems for 4 path channels vs. SNR in dB

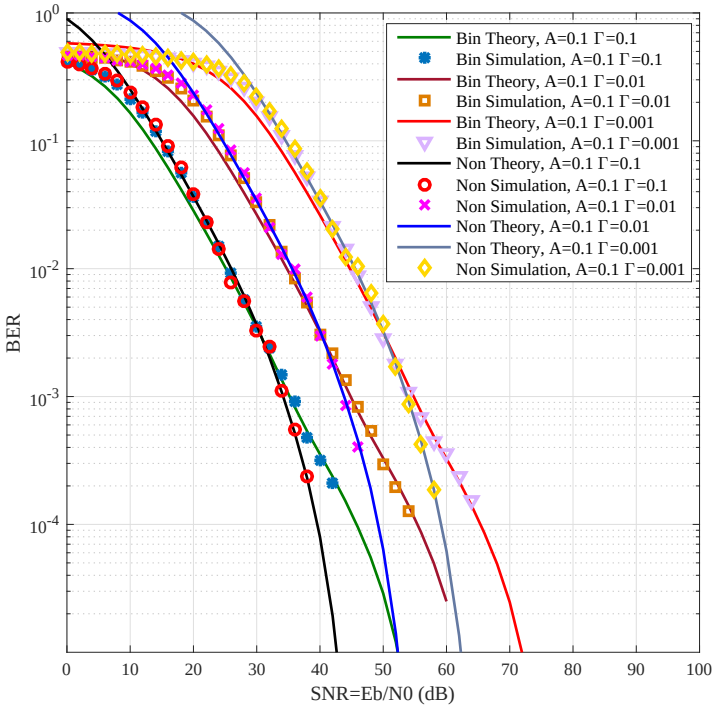

Figure 4.11: BER performance analysis of coded $(1, 7/5)_8$ binary and $\beta 1\beta/\beta^2 1$ non-binary convolutional OFDM-PLC systems for 15 path channels vs. SNR in dB

It can be noticed that the all coded OFDM-PLC systems achieve a large gain over uncoded OFDM-PLC system. Furthermore, since the channels after employing OFDM are no longer behaving as an impulsive channel, the performance of the coded $\beta\beta^2/1$ non-binary convolutional system is slightly better than the $(1, 7/5)_8$ binary, especially when the channel has many taps.

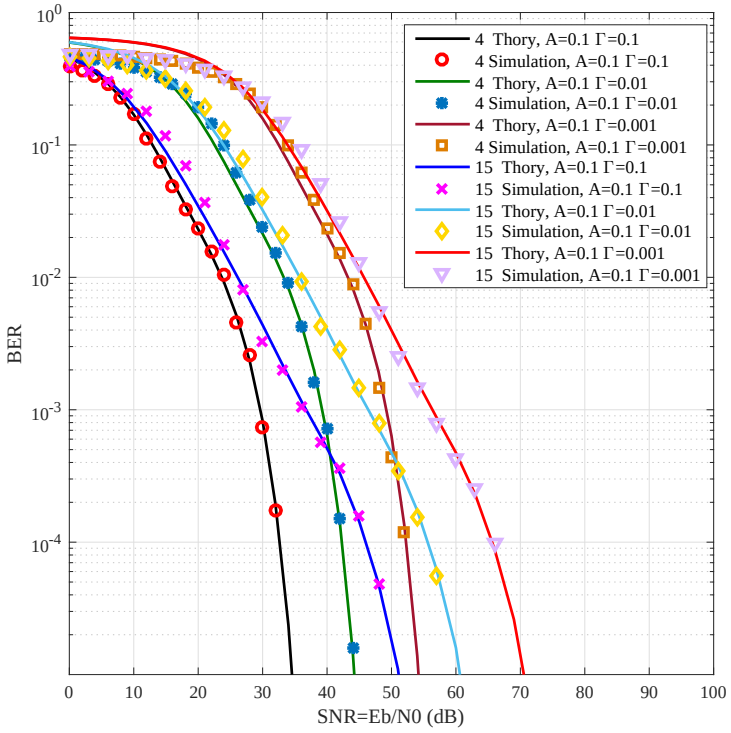

Figure 4.12: BER performance analysis of coded $\beta\beta^2/1$ non-binary convolutional OFDM-PLC systems for 4 and 15 path channels vs. SNR in dB

4.3 Non-Binary Turbo Code

4.3.1 Non-Binary Turbo Encoder

As been mentioned in chapter 3, the non-binary turbo code is a set of parallel concatenated non-binary convolutional encoders separated by a pseudo-random interleaver Π. A non-binary convolutional encoder comprises a set of memory elements and multipliers defined in a finite field $GF(q)$ with q elements $\{0, 1, \beta, \beta^2, ..., \beta^{q-2}\}$, where β is a primitive element. Figure 3.1 shows the $\beta^2\beta/1$ non-binary convolutional encoder defined in GF(4) with code rate $R = \frac{1}{2}$ [52] and figure 3.6 illustrates the $\beta^2\beta/1$ non-binary turbo encoder with rate $R_c = \frac{1}{3}$.

4.3.2 Non-Binary Turbo Decoding over Impulsive Noise

The PLC channel has a severe impulsive nature, but OFDM can spread the noise over all sub-carriers during the FFT process. Hence the noise after OFDM demodulation can be regarded as Gaussian but with a different variance and the channel log-likelihood ratio (LLR) can be calculated as

$$\begin{aligned} L^z(c=z|\boldsymbol{y}) &= \ln \frac{P^z(c=z|\boldsymbol{y})}{P^z(c=0|\boldsymbol{y})} \\ &= \ln \frac{P^z(\boldsymbol{y}|c=z)}{P^z(\boldsymbol{y}|c=0)} + \ln \frac{P^z(c=z)}{P^z(c=0)} \\ &= L^z(\boldsymbol{y}|c=z) + L^z(c) \end{aligned} \tag{4.15}$$

where $L^z(c)$ is the a priori LLR and $\boldsymbol{y}$ is the received sequence. For $z \in GF(2^p)$, each element z contains p bits and $\boldsymbol{y}$ also consists of p bits. $L^z(\boldsymbol{y}|c=z)$ is calculated as

$$L^z(\boldsymbol{y}|c=z) = \ln \frac{P^z(y_1,...,y_p|c=z)}{P^z(y_1,...,y_p|c=0)} = \sum_{l:c_l=1} \frac{2y_l}{\sigma_Z^2},$$

In this case, σ_Z^2 is the variance of the Middleton Class A noise in the frequency domain and calculated from equation (4.7)

Hence, γ_t can be calculated as

$$\gamma_t(s_0, s_1) = L^z(c) + \ln P^z(\boldsymbol{y}|\boldsymbol{x}), \tag{4.16}$$

where $\boldsymbol{x}$ is the vector of modulated symbols. Finally, the output LLR of the decoded message symbols are given as

$$\begin{aligned} L^z(c=z|\boldsymbol{y}) = &\max_{s^i - s^j \in s^z} \{\alpha_{t-1}(s_0) + \gamma_t(s_0, s_1) + \delta_t(s_1)\} \\ &- \max_{s^i - s^j \in s^0} \{\alpha_{t-1}(s_0) + \gamma_t(s_0, s_1) + \delta_t(s_1)\}, \end{aligned} \tag{4.17}$$

where s^z represents the set of all state transitions corresponding to $c \neq 0$ and s^0 is the set of all state transitions corresponding to $c = 0$. This output LLR will be used as the extrinsic information for the other component decoder.

4.4 Non-Binary Turbo Coded OFDM-PLC System with Non-Linear Processing

Fig 5.8 shows the system model that is used in this section. The input is a set of non-binary symbols k, where $k \in \mathrm{GF}(4)$. First, k message symbols are encoded by a non-binary turbo encoder and then modulated using binary phase shift keying modulation. This is then passed to the orthogonal frequency division multiplexing block. The OFDM modulator applies an inverse fast Fourier transform to generate a complex baseband OFDM signal as equation (4.6).

At the receiver, after adding the Middleton Class A noise, the received signal will be processed by the blanking or clipping operation to reduce the effect of impulsive noise. Blanking is a non-linear process that is used to reduce the impulsive noise effect on the received signal y and this block is also shown in figure 4.13. After blanking, the received signal is given as

$$\boldsymbol{r}_i = \begin{cases} \boldsymbol{y}_i, & |\boldsymbol{y}_i| < T_B \\ & , 0 \leq i \leqslant K-1, \\ 0, & Otherwise \end{cases} \tag{4.18}$$

where T_B is the blanking threshold and y_i is the received signal, given by $y_i = x_i + n_i$ and n_i is the Middleton Class A noise. Clipping is another non-linear process which limits the received signal and the output is given as:

$$\boldsymbol{r}_i = \begin{cases} \boldsymbol{y}_i, & |\boldsymbol{y}_i| < T_C \\ & , 0 \leq i \leqslant K-1\,, \\ T_C e^{jarg(\boldsymbol{y}_i)}, & Otherwise \end{cases} \tag{4.19}$$

where T_C is the clipping threshold value. These non-linear operations are applied before the OFDM demodulator on the receiver side. Then the OFDM demodulator is performed by a fast Fourier transform. After OFDM demodulation the signal will pass through a zero forcing (ZF) detector to compensate for the channel distortion, defined by [23]

$$w(k) = \frac{H^*(k)}{|H(k)^2|}, \tag{4.20}$$

where $H(k)$ is the channel frequency response. Finally, a Max-Log-Map non-binary

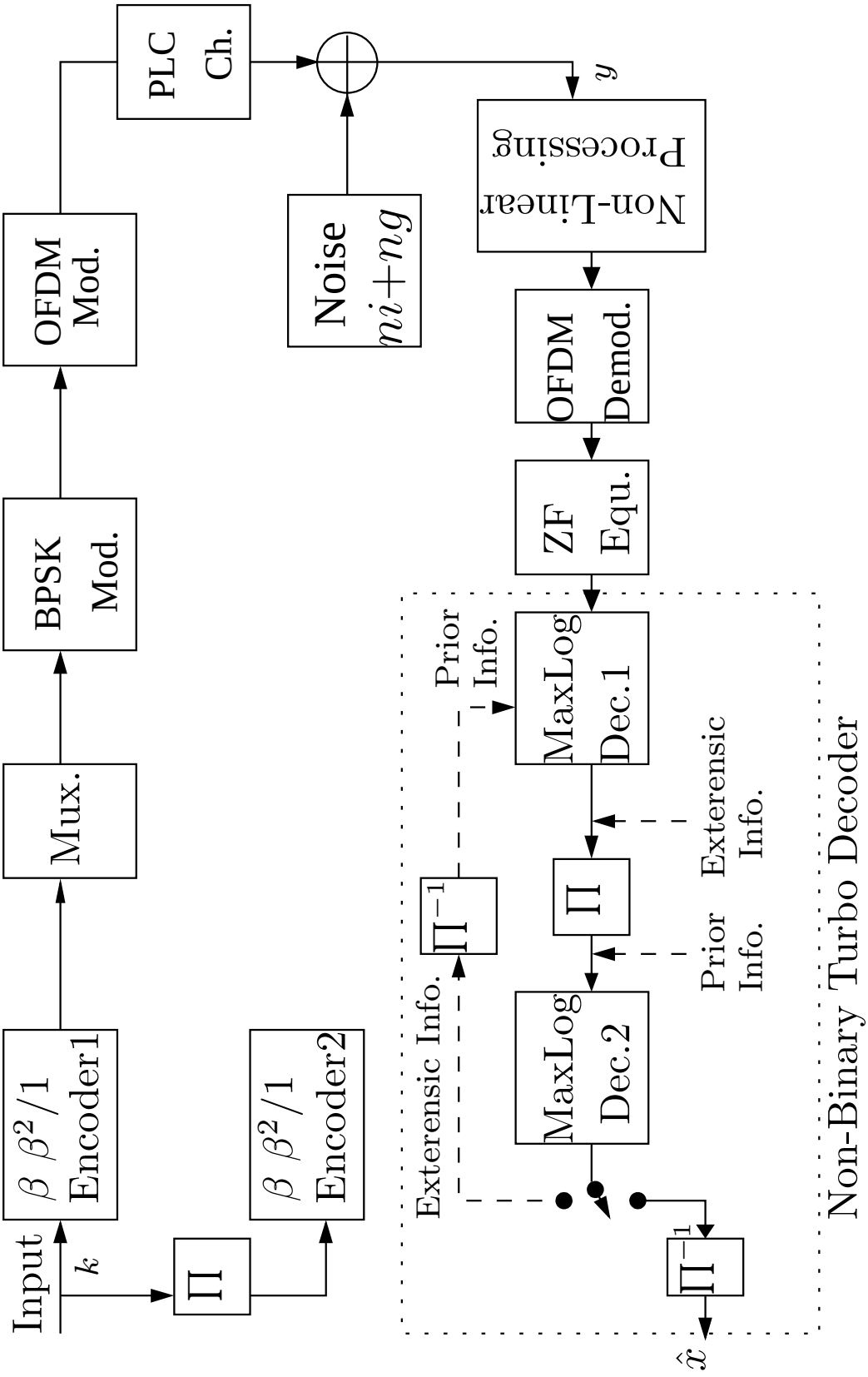

Figure 4.13: Non-binary turbo coded OFDM-PLC system.

turbo decoder.

4.5 BER Simulation Results of Non-Binary Turbo Coded OFDM-PLC systems

In this section we present simulation results for a non-binary turbo (NBT) coded PLC-OFDM system, binary turbo (BT) coded PLC-OFDM system and uncoded PLC-OFDM system. BPSK modulation has been used for all simulated and theoretical implementations. The system is examined on very impulsive channels, where the impulsive index A=0.01 and the impulsiveness is greater than the Gaussian noise by 100 times (i.e. Γ=0.01). The 4 and 15 paths PLC channels are used to model the realistic measurement given in [69]. To make a fair comparison, simulation results for a comparable BT code comprising $(1, 7/5)_8$ recursive systematic convolutional codes and message length $k = 2048$ bits are compared with the $(\beta^2\beta/1)$ NBT code and message length of 1024 symbols. Both codes have a code rate of $\frac{1}{3}$ and can be realized by a 4-state trellis diagram. The maximum iterations for both decoders is set to 5. A channel bandwidth 5KHz - 20MHz is considered.

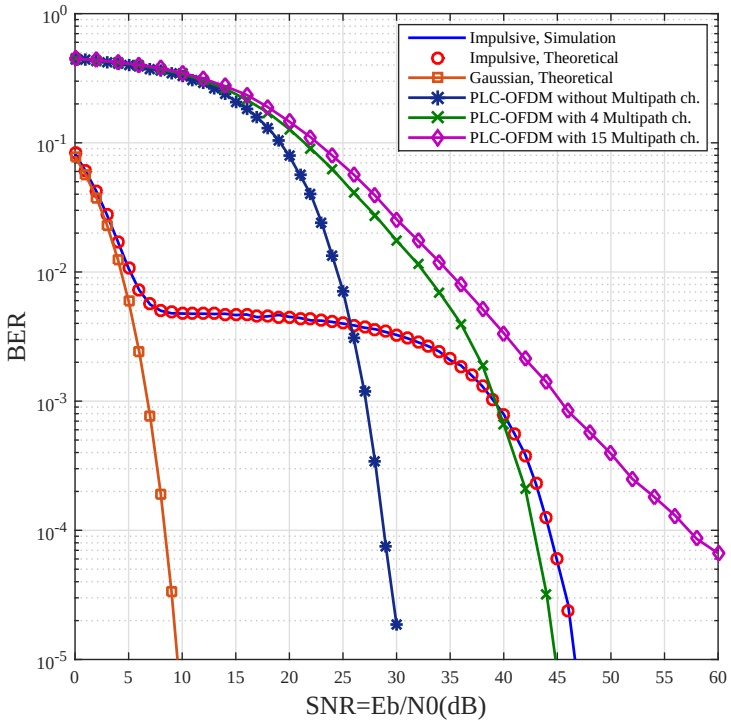

Figure 4.14: BER performance of uncoded AWGN, Middleton Class A, and uncoded OFDM-PLC on different realistic PLC channels, versus SNR (dB).

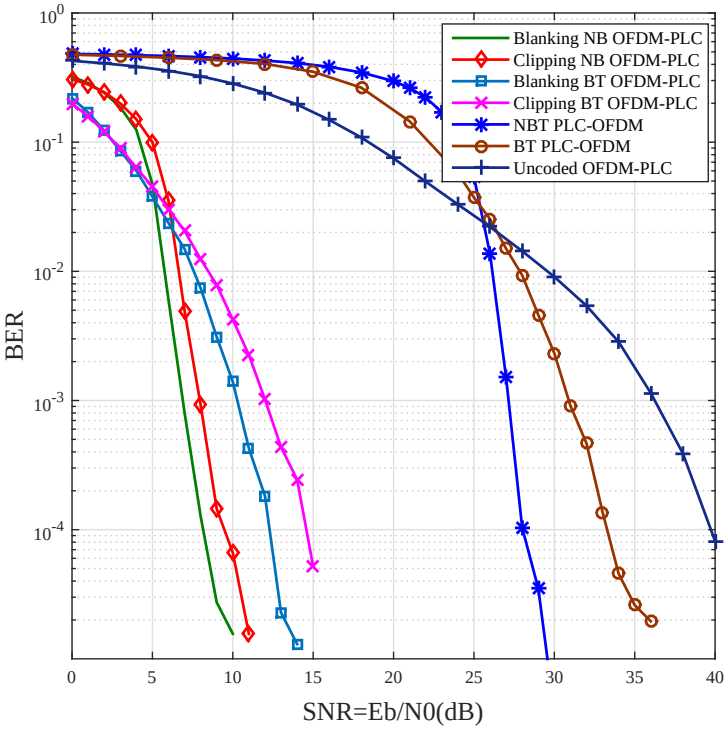

Figure 4.15: BER performance of uncoded OFDM-PLC and coded (BT and NBT) OFDM-PLC versus SNR (dB) on 4 path frequency selective channel.

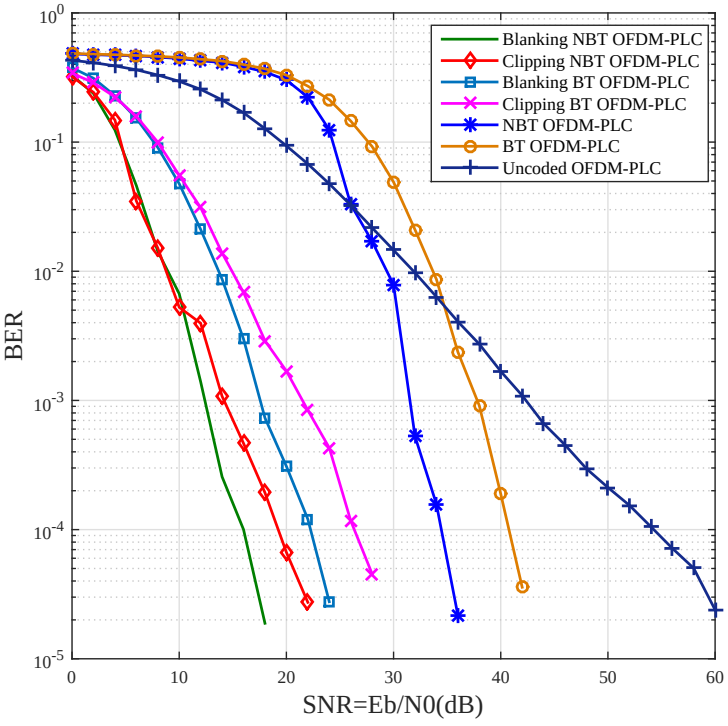

Figure 4.16: BER performance of uncoded OFDM-PLC and coded (BT and NBT) OFDM-PLC versus SNR (dB) on 15 path frequency selective channel.

Fig. 4.14 shows theoretical and simulated BER vs. SNR(dB) for the AWGN channel, the Middleton class A channel with and without OFDM and BER for uncoded PLC system over 4 and 15 multipath frequency selective channels. We can observe the hurdles that degrade the performance of the communications over PLC when compared with the conventional AWGN channel. It also illustrates the benefits of using OFDM on impulsive channel since the BER performance is enhanced by 15 dB at the low BER region. In addition, Fig. 4.14 displays the impact of different multipath models of PLC channels on the system performance.

Fig. 4.15 compares the BERs for NBT coded, BT coded and uncoded PLC-OFDM system on the 4-path frequency selective channel with and without the blanking and clipping techniques. Although the performance of BT PLC-OFDM system with blanking has already shown a 25 dB gain over the uncoded system, the NBT coded PLC-OFDM system offers a 30dB and 5 dB further gain, over uncoded and coded BT PLC-OFDM system respectively. We also notice that the blanking is a more effective process to eliminate the effect of impulsive noise compared with the clipping operation.

Fig. 4.16 demonstrates the BERs of NBT and BT codes, similar to previous case but with a 15-path channel model. It is shown that the NBT code again shows a superior performance compared with the BT system with a 6dB gain for all situations at a BER of 10^{-4}. It should be noticed that the blanking and clipping techniques are still showing further improvement to the BER performance for both NBT and BT OFDM-PLC systems. Finally, employing NBT codes on PLC systems can achieve significant coding gain over using uncoded PLC systems, by 24dB.

4.6 Summary

Coded power line communication (PLC) promises significant enhancement to the overall system performance over uncoded PLC systems. Yet, there is limited contribution on the matter of employing coding schemes on PLC systems and its corresponding theoretical analysis. In this chapter, the $\beta\beta^2/1$ and $\beta 1\beta/\beta^2 1$ non-binary convolutional codes have been examined in an extreme PLC environment and an investigation of the BER performance analysis of coded OFDM-PLC systems has been carried out. This analysis is accomplished by considering realistic PLC frequency-selective multi-path channels and for various values of the background to impulsive

noise ratio Γ.

Furthermore, the theoretical bounds of binary and non-binary convolutional coded OFDM-PLC systems have been investigated for the first time. In order to validate our results, the implementation of simulated and theoretical results have been obtained for different values of noise parameters and on different PLC channels. Our results showed a good agreement between the simulated and theoretical analysis for all cases.

However, it is worth mentioning that the advantage of utilizing OFDM in such a harsh medium is that it is a very robust technique to dispose of the impact of the multi-path frequency-selective channel, and it significantly lessens the impulsiveness of the Middleton class A noise channel by spreading the noise energy equally over each received symbols and converts the PDF approximately into AWGN with different variant.

Also in this chapter, an investigation into the performance of non-binary turbo codes on power-line channels, in term of BER, has been presented. A non-Binary turbo-coded OFDM-PLC system employing two non-linear receivers, blanking and clipping, has been proposed. The system has been examined on realistic multipath frequency selective PLC channels with extremely impulsive Middleton class A noise, with different values of A and Γ. Finally, a fair comparison with a binary turbo-coded PLC system in the same environment has been evaluated and simulation results have shown that the non-binary turbo code offers a superior performance on power-line channels.

Chapter 5

Non-Binary Trellis Coded MIMO for Wireless and Wired Applications on Middleton Class A Noise Channel

5.1 Introduction

MIMO systems have shown great performance in terms of coverage, reliability and capacity, but only a few studies have focused on coded MIMO on impulsive noise channels. Since the communication channels not always follow a Gaussian distribution, the consideration of a non-Gaussian noise channels can be essential to examine the reliability of the existing communication systems in extreme environments. In this chapter various binary and non-binary coding schemes have been utilized on wireless OFDM-MIMO systems when the channel is impulsive. Furthermore, the performance analysis of binary and non-binary convolutional coded OFDM-MIMO systems on AWGN and Middleton class A noise channel has been investigated.

Moreover, in this chapter the implementation of OFDM-MIMO systems on PLC channels will be carried out. However, since power line networks were not originally designed for communication purposes, transferring data over these cables is vulnerable to impulsive noise due to electromagnetic interference [5].

Issues of reliability, speed and large data transfer have become very challenging in PLC due to the negative impact of the electrical cables, which are not designed

for communication services. In order to enhance PLC BER performance, non-binary and binary iterative codes have been employed with MIMO in this chapter.

Although, non-binary codes are more robust to impulses caused by non-Gaussian noise, OFDM with non-linear preprocessing techniques (blanking, and clipping) have been utilized to eliminate the impulsive noise effects. In addition, we compare the proposed non-binary trellis codes with comparable binary codes trellis codes. Finally, the system is examined for various values of the impulsive index (A) and noise ratio (Γ) over the synthetic statistical MIMO PLC channel and a SISO scheme.

The chapter is organized as follows: Section 5.2 presents the implementation of coded and uncoded OFDM-MIMO systems for wireless applications. In section 5.2.1 the analysis of uncoded OFDM-MIMO systems on Middleton class A noise channel is carried out. While in section 5.2.2, the simulated BER performance of coded binary and non-binary turbo codes have been investigated. The performance analysis of wireless OFDM-MIMO on an impulsive noise channel is derived and compared to simulation results in section 5.2.3. Section 5.3 introduces the wired application of binary and non-binary trellis codes on PLC MIMO channels. The synthetic MIMO PLC channel is described in detail in 5.3.3. In section 5.3.2.1 and 5.3.2.2 the non-binary turbo coded MIMO PLC-OFDM transmitter and receiver are introduced, respectively. In section 5.3.3, numerical and simulation results of binary and non-binary trellis coded OFDM-PLC-MIMO are presented. Finally, section 5.4 presents the conclusions.

5.2 Non-Binary Trellis Coded OFDM-MIMO for Wireless Applications on Additive Impulsive Noise

5.2.1 Uncoded OFDM-MIMO for Wireless Applications on Additive Impulsive Noise

This section considers uncoded OFDM-MIMO systems on impulsive noise channels with a MMSE equalizer on an ideal fast fading channel. However, due to the central limit theorem, the SER analysis will be obtained exactly the same as [96], by considering the total impulsive noise variance of equation (2.40).

The PDF of the SINR at the output of the detector on the AWGN channel of $N_t \times N_r$ is calculated as

$$S_{N_t \times N_r}(\gamma) = \int_0^{\infty} \cdots \int_0^{\infty} f_{N_t \times N_r}(\gamma | \lambda_1, \cdots, \lambda_{N_r-1}) \times f(\lambda_1, \cdots, \lambda_{N_r-1}) d\lambda_1 \cdots d\lambda_{N_r-1} \tag{5.1}$$

where, λ_i is an eigenvalue of HH^H, and γ is the SINR. Since equation (5.1) is very complicated for a high number of antennas, the following three PDFs will be considered for the upcoming analysis to validate the simulated SER and BER systems.

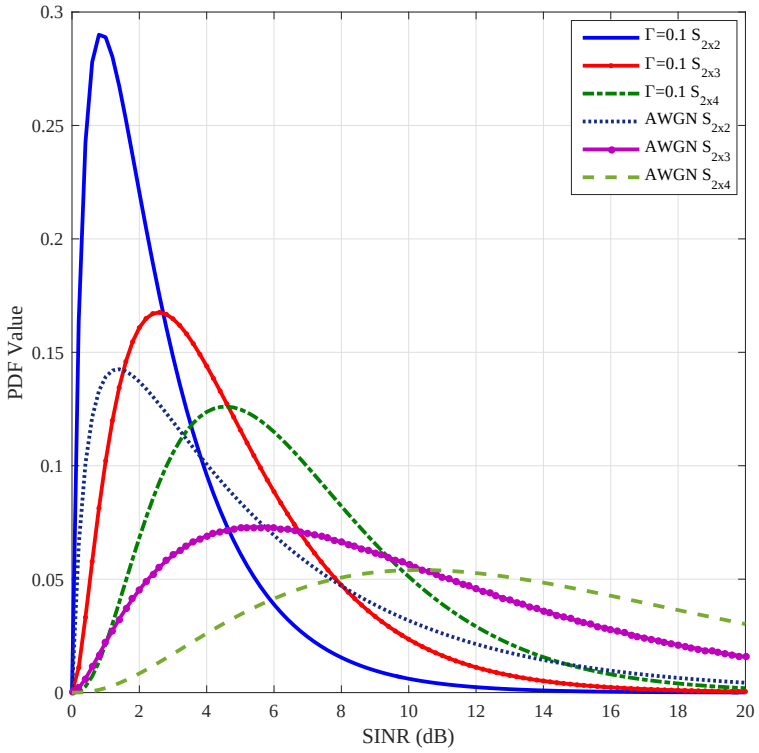

Figure 5.1: The SINR PDF of $N_t \times N_r$ for AWGN and Middleton Class A Channels.

$$S_{2\times2}(\gamma) = s \exp^{-s\gamma} \left[1 + s - \frac{1}{(\gamma+1)^2} - \frac{s}{\gamma+1}\right] \tag{5.2}$$

$$S_{2\times3}(\gamma) = s^2 \gamma \exp^{-s\gamma} \left[1 + \frac{s}{2} - \frac{1}{2(\gamma+1)^2} - \frac{s+1}{2(\gamma+1)}\right] \tag{5.3}$$

$$S_{2\times4}(\gamma) = s^3\gamma^2 \exp^{-s\gamma}\left[\frac{1}{2} + \frac{s}{6} - \frac{1}{6(\gamma+1)^2} - \frac{s+2}{6(\gamma+1)}\right] \tag{5.4}$$

where, s is calculated by

$$s = \frac{N_t N_0}{Es} \tag{5.5}$$

However, in order to evaluate the SINR PDF of MIMO-OFDM PLC systems, N_0 in equation (5.5) can be calculated by using equation (4.7).

Figure 5.1 illustrates the SINR PDF for AWGN and Middleton class A noise channels when the SNR is set to 10 dB.

In [97], the error probability of MQAM with a MMSE equalizer is defined as

$$P_{\sqrt{M}} = 2\left(1 - \frac{1}{\sqrt{M}}\right) Q\left(\sqrt{\frac{3E_s\Lambda^2}{(M-1)N_t\sigma_G^2}}\right) \tag{5.6}$$

where Λ is the interference plus noise coefficient and is estimated as

$$\Lambda = W_i^H h_i \tag{5.7}$$

where W_i is the ith column of the MMSE detector filter vector and h_i is the ith column of H.

In order to estimate the performance of MMSE detection with a $N_t \times N_r$ MIMO system, the approximation of the $Q(.)$ function by applying numerical simulation will be [96]

$$Q(x) = \frac{1}{12}e^{-\frac{x^2}{2}} + \frac{1}{6}e^{-\frac{2x^2}{3}} \tag{5.8}$$

By using equation (5.6), the conditional probability of the SER given SINR is defined as [97]

$$\begin{aligned} Pe_{(N_t\times N_r)}(e|\gamma) &= 2P_{\sqrt{M}} - P_{\sqrt{M}}^2 \\ &= 4\left(1 - \frac{1}{\sqrt{M}}\right)\left[\frac{1}{12}\exp\left(-\frac{3\gamma}{2(M-1)}\right)\right. \\ &\left. + \frac{1}{6}\exp\left(-\frac{2\gamma}{M-1}\right)\right] \end{aligned} \tag{5.9}$$

By employing equations 5.4 and 5.9 the SER can be obtained as

$$P_{(N_t \times N_r)} \simeq \left(1 - \frac{1}{\sqrt{M}}\right)\left[\frac{1}{3}G_{N_t \times N_r}\left(\frac{3}{2(M-1)}\right) + \frac{2}{3}G_{N_t \times N_r}\left(\frac{2}{M-1}\right)\right] \quad (5.10)$$

where $G_{N_t \times N_r}$ is the moment generation function (MGF) of γ, and it has been presented in [96] for low $N_t \times N_r$. In this chapter, three examples will be implemented to validate the simulated coded and uncoded results and these MGFs are

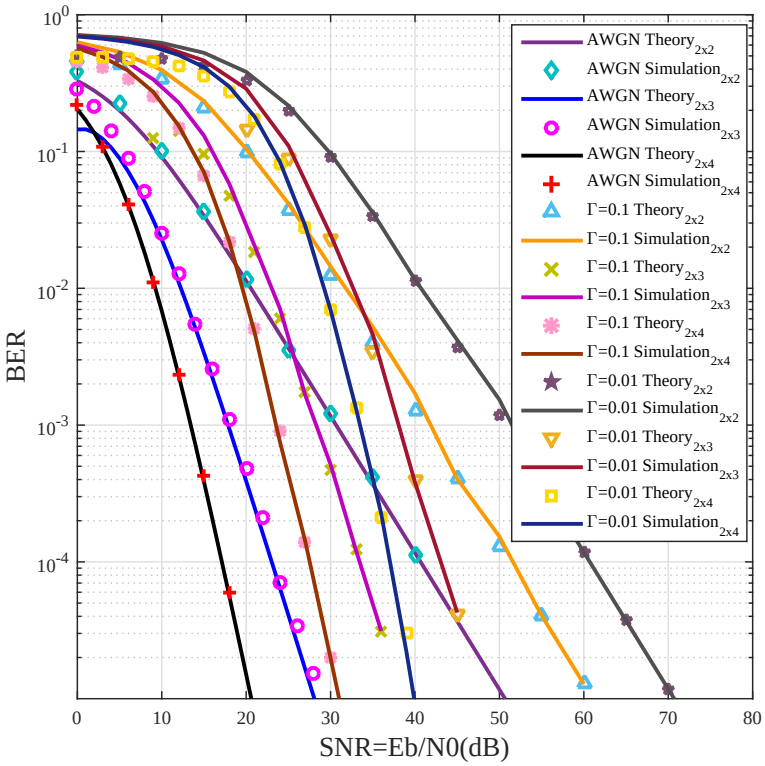

Figure 5.2: BER performance analysis for uncoded MIMO on AWGN channel and for uncoded ODFM-MIMO on impulsive noise channel.

$$G_{2\times 2}(m) = s\left[\frac{1-m}{s+m} + m\exp^{s+m} Ei(s+m)\right] \quad (5.11)$$

$$G_{2\times 3}(m) = \frac{s^2}{2}\left[\frac{2-m}{(s+m)^2} + \frac{m}{s+m} - m\exp^{s+m} Ei(s+m)\right] \quad (5.12)$$

$$G_{2\times 4}(m) = \frac{s^3}{6}\left[\frac{6-2m}{(s+m)^3} + \frac{m}{(s+m)^2} - \frac{m}{s+m} + m\exp^{s+m} Ei(s+m)\right] \quad (5.13)$$

where $E_i(.)$ is the exponential integral function defined as

$$E_i(x) = \int_x^{\infty} \frac{\exp -t}{t} dt \qquad for \qquad x > 0 \tag{5.14}$$

Figure 5.2 shows the theoretical analysis in equation (5.10) for various MIMO scenarios on AWGN and impulsive noise channels. The simulated and theoretical BER are obtained for QAM modulation and for messages of length 2000 bits. It is important to mention that the OFDM is utilized in the impulsive noise scenario and in the case of AWGN the results are obtained for single carrier only.

Furthermore, the BER curves obtained by multiplying the SER with $(\frac{2^{k-1}}{2^k-1})$ factor as been mentioned in chapter 3. In addition, the analysis of OFDM-MIMO on the Middleton Class A Noise Channel has been examined for different values of Γ (i.e. Γ=0.1, and 0.01) and for three MIMO systems scenarios (i.e. $N_t \times N_r = 2 \times 2, 2 \times 3$, and 2×4). It can be seen that there is a good agreement between the simulated and analysis BER performance curves for all cases.

5.2.2 Trellis Coded OFDM-MIMO for Wireless Applications on Additive Impulsive Noise

In this chapter, various codes will be considered to enhance the overall MIMO system performance. These codes are the $(1, 7/5)_8$ binary convolutional and turbo codes, and the $\beta\beta^2/1$ non-binary convolutional and turbo codes.

The block diagram 5.3 shows the general structure of the trellis coded OFDM-MIMO systems. In the transmitter, a stream of k bits or symbols will pass through one of the above listed encoders. The output codewords from these encoders will be interleaved randomly and then will pass to the QAM modulator. In order to mitigate the effects of the impulsive noise, OFDM modulation will be utilized after the QAM modulation process. Then, the output of the OFDM will be divided into N_t vectors to transmitted via a fast fading MIMO channel that has dimensions of $N_t \times N_r$ and is defined as

$$\boldsymbol{H} = \begin{bmatrix} h_{1,1} & h_{1,2} & \cdots & h_{1,N_r} \\ h_{2,1} & h_{2,2} & \cdots & h_{2,N_r} \\ \vdots & \vdots & \ddots & \vdots \\ h_{N_t,1} & h_{N_t,2} & \cdots & h_{N_t,N_r} \end{bmatrix} \tag{5.15}$$

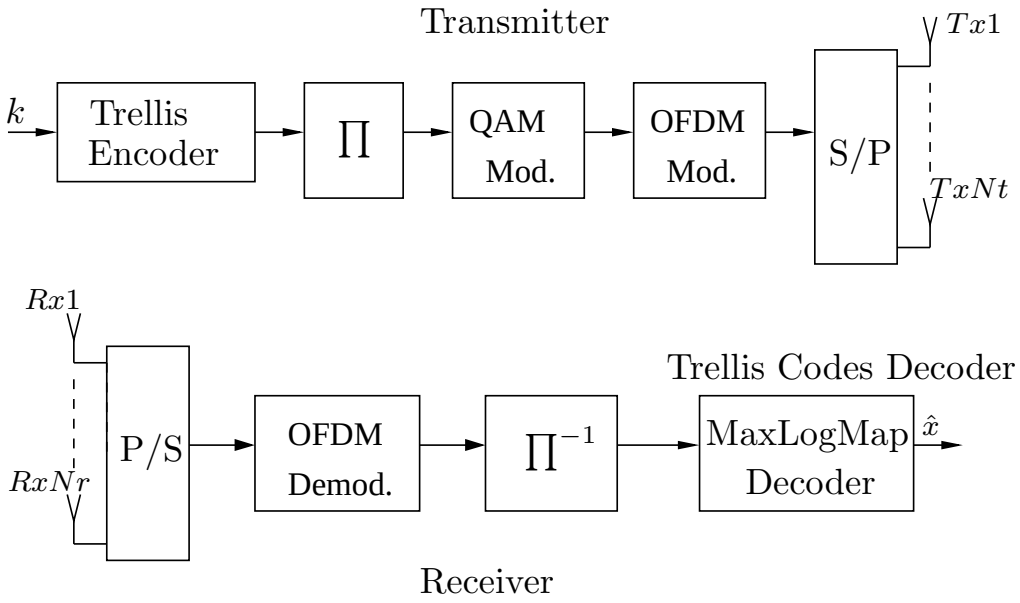

Figure 5.3: General diagram of trellis coded OFDM-MIMO system.

where $h_{i,j}$ are the Rayleigh fading coefficients, and i, j are the number of received and transmitted antennas, respectively.

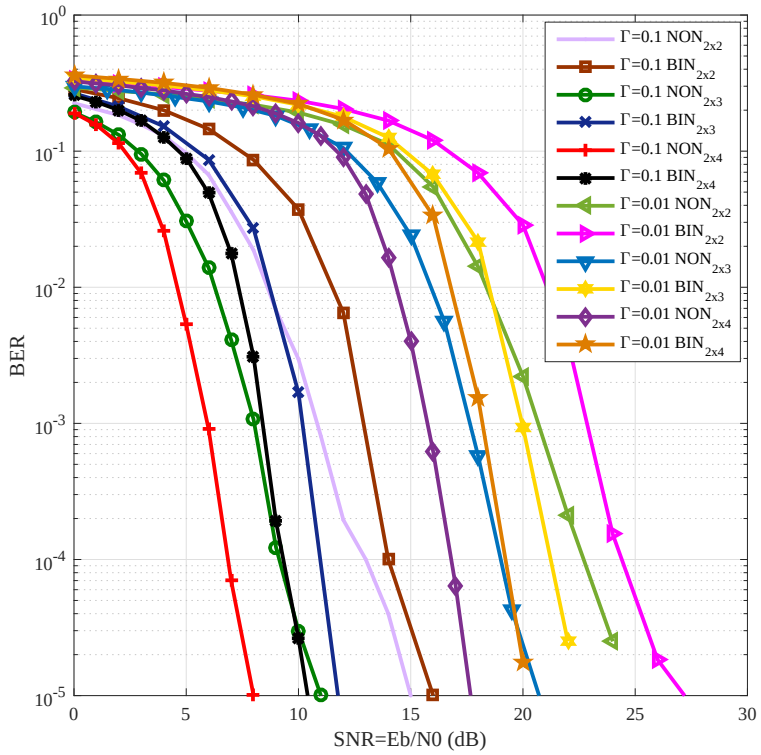

Figure 5.4: Simulation BER performance of the $(1, 7/5)_8$ binary and $\beta\beta^2/1$ non-binary turbo coded OFDM-MIMO on impulsive noise channels.

At the receiver side, N_r vectors of symbols will be received and the first operation

will be to reform these vectors into one stream of symbols. Removing the cyclic prefix and taking the FFT of the S/P block output will be the second process. After that, the output will pass through the deinterleaver before it passes to the decoder. Finally, the Max-Log-MAP algorithm will be applied for all trellis codes that been used in this chapter.

In this section, the above procedure has been considered for a rate $\frac{1}{3}$ binary and non-binary turbo codes and figure 5.4 shows the simulated BER performance vs. SNR.

The results have been obtained for a binary code with message length of 2000 bits and a 1000-symbol message length for the non-binary turbo code system. In addition, the CP length was 256 and the impulsive noise parameters are $A = 0.1$ and $\Gamma = 0.1$, and 0.01. It can be noticed from figures 5.2 and 5.4 that the coded BER performance exceeds the performance of the uncoded systems by 15 dB for the non-binary and about 13 dB for the binary codes, at a BER of 10^{-5}. It is also clear that the non-binary coded OFDM-MIMO performance achieves a gain over the binary of at least 2dB for all MIMO scenarios.

In the next section, the simulated and theoretical analysis of binary and non-binary convolutional coded OFDM-MIMO on impulsive noise channels will be investigated.

5.2.3 BER Analysis of Convolutional Coded Wireless OFDM-MIMO Additive Impulsive Noise

Recalling equation (3.26), the pairwise error probability P_w must be calculated when considering MIMO channels and the interference. In other words, the LLR should be calculated for non-Binary convolutional codes with a MMSE detector, which is represented as

$$LLR_i = \ln \frac{\sum_{\tilde{x} \in \mathcal{S}_z^l} \Pr(\boldsymbol{y}_i | \tilde{x}, \beta_i, n_i)}{\sum_{\tilde{x} \in \mathcal{S}_0^l} \Pr(\boldsymbol{y}_i | \tilde{x}, \beta_i, n_i)} \tag{5.16}$$

Since the all zero codeword is a valid codeword, the following inequality will be valid as well

$$P_w(d) = Pr\left(\sum_{i=1}^{d_f} LLR_i > 0 \right) \tag{5.17}$$

Finding a closed form for this probability is very difficult, therefore it can be approximated by evaluating the MGF for the LLR in 5.17, and the result will be as presented in [98]

$$P(w) = \frac{1}{v}\sum_{k=1}^{v/2}\left[\sum_{i} p_{M,i} M_{Nt\times Nr}\left(\frac{(1+t_k^2)N_t}{4E_s}\varepsilon_{M,i}\right)\right]^d \tag{5.18}$$

where v is the number of nodes that is sufficient for relevant accuracy which is used for the Gauss-Chebyshev quadrature role (23) [99–101]. In addition, $P_{M,i}$ and $\varepsilon_{M,i}$ are the frequency of occurrence of each distance and the squared Euclidean distance, respectively defined in Table 1 in [101]. While t_k is given by

$$t_k = tan\left(\frac{\pi(2k-1)}{2v}\right). \tag{5.19}$$

With regard to examining the accuracy of the above equations, equation (5.18) will be substituted into equation (3.26). Figure 5.5 illustrates the comparison between the simulated and theoretical BER performance of binary convolutional coded OFDM-MIMO on AWGN and also on Middleton class A noise channels.

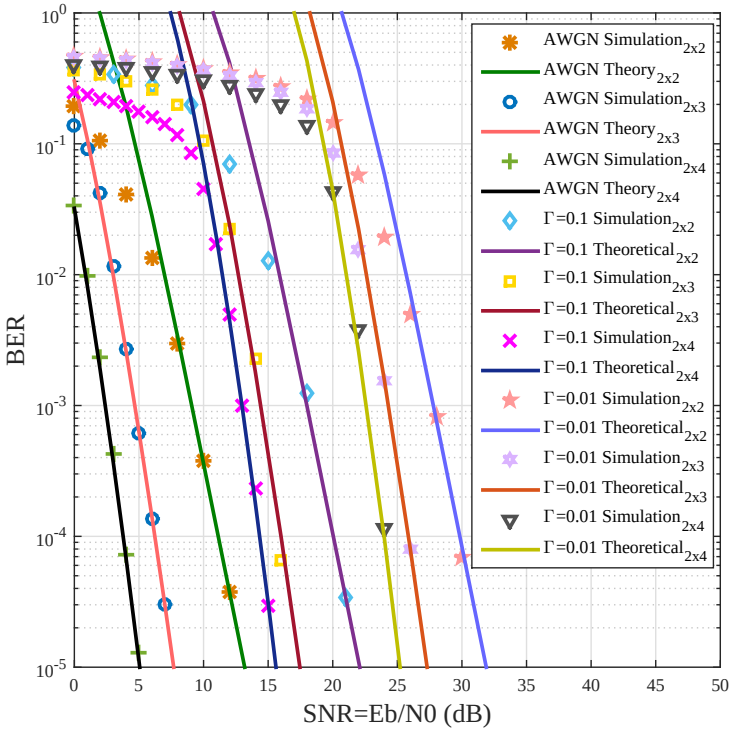

Figure 5.5: Simulated and theoretical BER performance of binary convolutional coded OFDM-MIMO on AWGN and impulsive noise channels.

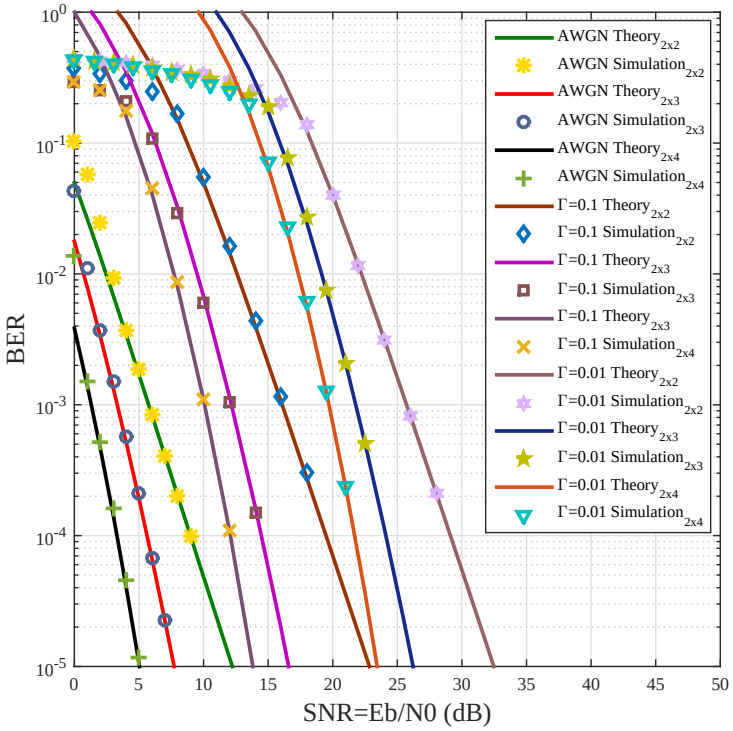

Figure 5.6: Simulated and theoretical BER performance of non-binary convolutional coded OFDM-MIMO on AWGN and impulsive noise channels.

The implemented results have been obtained for a 2000-bit message length, with QAM modulation and with a CP of length 256 symbols. Furthermore, the values of the impulsive to Gaussian ratio Γ are 0.1 and 0.01 and A takes the value of 0.1.

It is very clear that the analysis shows a very close bound to the simulated results for all cases. Likewise, figure 5.6 demonstrates the analysis of non-binary convolutional coded OFDM-MIMO on AWGN and on impulsive noise channels. The results have been achieved by considering the same parameters of figure 5.5, although the message length is 1000 symbols.

Again the results display a great agreement between the simulated and the theoretical in the high SNR region at a BER of 10^{-5}.

5.3 Non-Binary Trellis Codes on the Synthetic Statistical MIMO Power Line Channel

5.3.1 A Synthetic Statistical MIMO PLC Channel

The most common impulsive noise distribution that is used to model the power line channel is Middleton class A noise and its pdf is defined in equation 4.1.

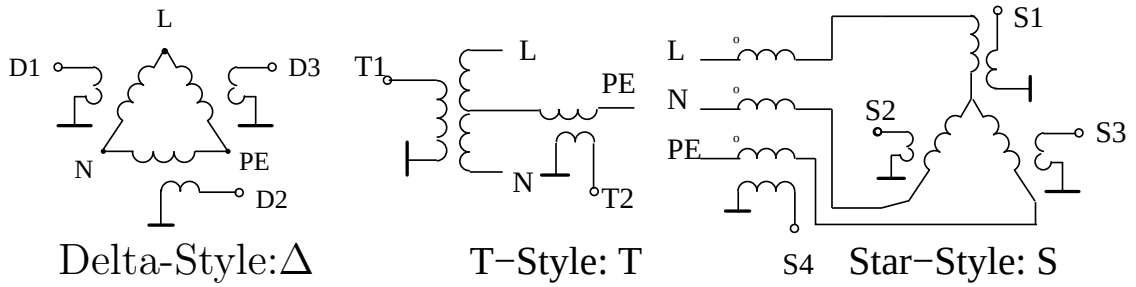

Figure 5.7: Inductive MIMO PLC couplers.

Generally, the low voltage (LV) in-home electrical networks employ three wires, two for power supply which are known as Phase and Neutral (P and N), and one for protection which is labeled Protection Earth (PE). These three wires can be connected in different styles such as: Delta-style (Δ), T-style (T) and Star-style (S), as shown in Fig 5.7. These topologies can be used as MIMO transmitters and/or receivers. For example, we can use $P-N$ and $P-E$ conductors of Δ-Style as two transmitting ports, and P, N, PE conductors of S-style as three receiving ports [102] [79].

In this chapter, we employ the synthetic statistical MIMO PLC channel that was introduced in [89] [103] and its channel matrix is shown in Fig 5.8. Basically, the authors obtained the MIMO PLC channel parameters based on an absolute phonological outlook, with the help of the in-home 2x3 MIMO PLC channel data that is provided by the European Telecommunication Standards Institutes (ETSI) [102] [104]. This database consists of 353 MIMO channel frequency responses (CFRs) and 1588 samples in the 1.8-100 MHz frequency band.

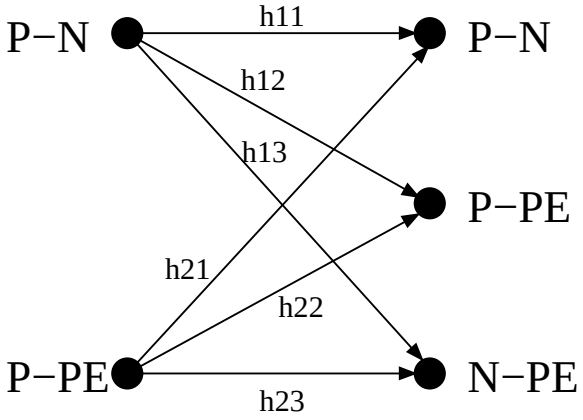

Figure 5.8: 2x3 MIMO Channel Matrix.

5.3.2 System Model

5.3.2.1 Transmitter

The block diagram in Fig 5.9 shows the MIMO OFDM-PLC transmitter. At the beginning, the input symbols pass through a $\beta^2\beta/1$ non-binary convolutional encoder ($encoder_1$) and the output will be c_1c_2, where c_1 is the input symbol and c_2 is the parity check symbol. Simultaneously, the interleaved input passes through another $\beta^2\beta/1$ non-binary convolutional encoder ($encoder_2$) to produce the 2nd party check symbol c_3 [71] [52].

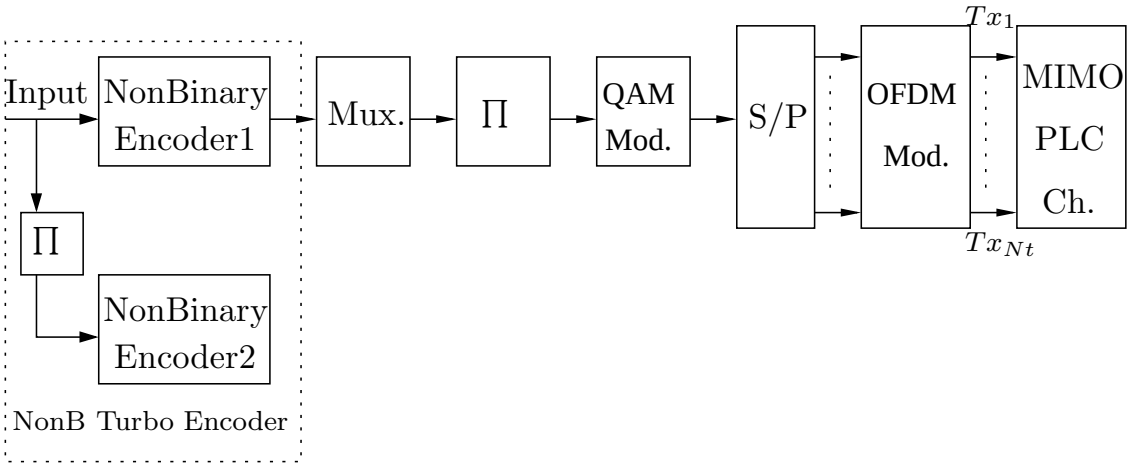

Figure 5.9: Non-Binary Coded MIMO PLC-OFDM System Transmitter.

The three outputs are mixed together to obtain the codeword as the sequence $c_1c_2c_3$. This codeword will be interleaved then mapped into baseband symbols using quadrature amplitude modulation (QAM). Next, complex baseband symbols will be

divided into N_t vectors and an IDFT will be applied to each vector as in equation 4.2. Finally, the N_t vectors will be transmitted via PN and/or PE conductors.

5.3.2.2 Receiver

Fig 5.10 illustrates a non-binary turbo coded MIMO PLC receiver in detail. At the receiver socket, the received signal y is given by:

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{n} \tag{5.20}$$

where H is the MIMO PLC channel matrix, x are the mapped transmitted symbols and n is the Middleton class A noise. First, the received signal will be processed by a non-linear pre-processing (Blanking or Clipping) to mitigate the effect of the impulsive noise by applying equation (4.18) or (4.19). Second, the cyclic prefix CP will be removed from each received signal and the OFDM demodulator is performed by a FFT.

In this chapter, a Minimum Mean Square (MMSE) equalizer is used to estimate the PLC MIMO channel and the output will be given by:

$$\hat{x_i} = W_i^H Y_i, \tag{5.21}$$

where W is the MMSE detector filter vector given by [105]:

$$W_i = (HH^H + \sigma_z^2 I)h_i, \tag{5.22}$$

Here, h_i is the $i-th$ column of H, I is an $N_t \times N_t$ identity matrix and σ_z is the variance of the Middleton Class A noise, which can be estimated by equation (4.7) [75]. The received signal, before proceeding to the non-binary turbo decoder, will be reshaped as a series of symbols and interleaved. Finally, the Max-Log-Map has been utilized to obtain the transmitted signal as explained in chapter 4, section 4.3.2.

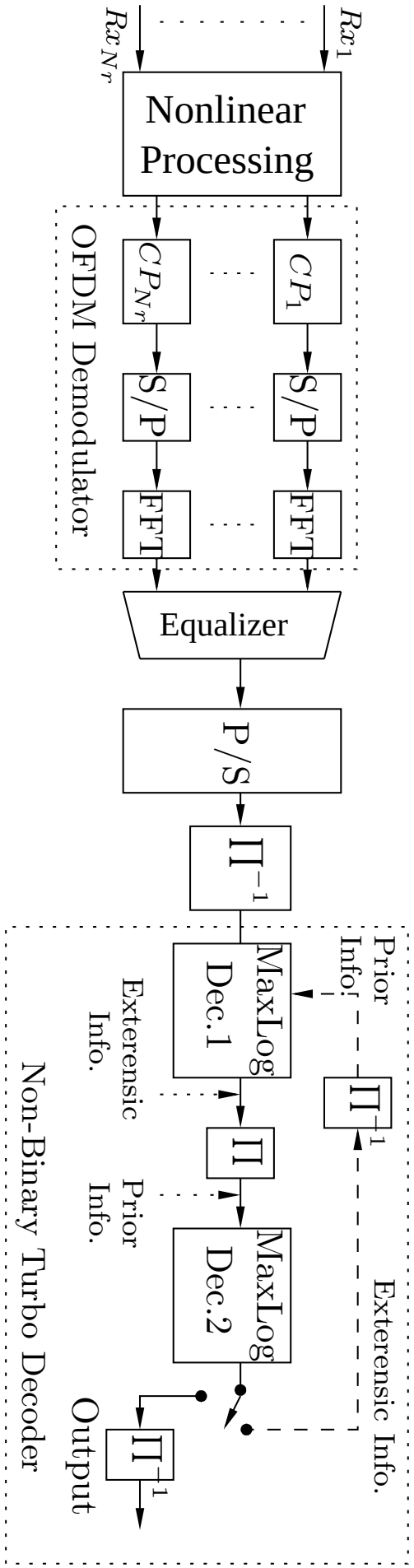

Figure 5.10: Coded OFDM-PLC system.

5.3.3 Numerical and Simulation Results

In this section, we present simulation results showing the bit error rate (BER) for;

- Non-binary convolutional (NonB) coded MIMO OFDM-PLC system.
- Binary convolutional coded MIMO OFDM-PLC system.
- Non-binary turbo coded MIMO OFDM-PLC system.
- Binary turbo coded MIMO OFDM-PLC system.
- Uncoded MIMO OFDM-PLC system.

Figs 5.11-5.14 show the performance of the systems above for various values of impulsive index A and background to impulsive noise ratio parameter Γ. In addition, these figures show the BER for several MIMO PLC patterns. QAM modulation has been used for all the above systems. For a fair comparison, the $(1, 7/5)_8$ recursive systematic convolutional code with data length of 2048 bits is compared to the $(\beta^2\beta/1)$ non-binary convolutional code with 1024 symbols data length. Both convolutional codes have a $\frac{1}{2}$ code rate and can be realized by a four-state trellis diagram. Moreover, both turbo codes have a $\frac{1}{3}$ rate code and the maximum iterations used for both turbo codes is 5. The (standard/mean) MIMO PLC channel matrix had been extracted from the p-function in [103] with 1000 MIMO realizations and 100 frequency samples in the 1.8-100 MHz band, for all systems.

Fig 5.11 shows the BER performance curves for both binary and non-binary convolutional codes on the AWGN channel for a point to point communication scenario. It is clear that the performance of both convolutional codes are very similar and that is why we chose these two codes to make a fair comparison on a non-Gaussian channel. Even though, the performance of both convolutional codes are the same on the AWGN channel, we can see that the non-binary convolutional code has a better performance for all MIMO-PLC scenarios. For instance, we can see 1.4 and 2 dB gain for the non-binary code over a binary convolutional code on 2x2 and 2x3 MIMO-PLC, respectively, when A=0.1 and Γ=0.1, at a BER of 10^{-5}.

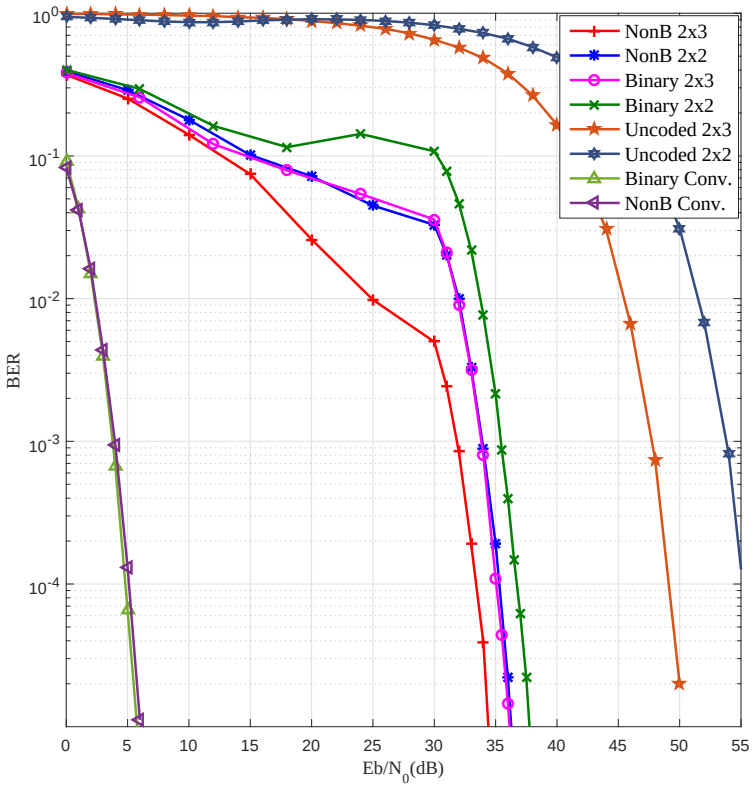

Figure 5.11: BER performance of binary and non-binary convolutional coded 2x3 MIMO PLC-OFDM system, with the using of blanking and clipping, uncoded MIMO PLC-OFDM when Γ= 0.1 and A= 0.1, and SISO binary and non-binary convolutional on AWGN channel versus SNR (dB).

Fig 5.12 compares the BERs for non-binary and binary turbo coded SISO PLC-OFDM systems with clipping and blanking techniques to reduce the effect of the impulsive noise when A=0.1 and Γ=0.01. We can conclude from fig 5.12 that the non-binary turbo BER performance is always better than the performance of the binary turbo for all SIMO systems (1x2 and 1x3) by at least 2dB at a BER of 10^{-5}. In addition, we notice that the blanking is more effective (non-linear pre-processing) for eliminating the effect of Middleton class A noise compared with the clipping operation, which is worse by 1.2dB for all cases of the proposed system. The BERs for both non-binary and binary turbo coded MIMO OFDM-PLC systems are presented in Fig 5.13, when A=0.1 and Γ=0.01 (i.e. the impulsiveness is greater than Gaussian noise by 100 times). However, it is clear that the proposed system is very robust and achieves a large gain over the binary system.

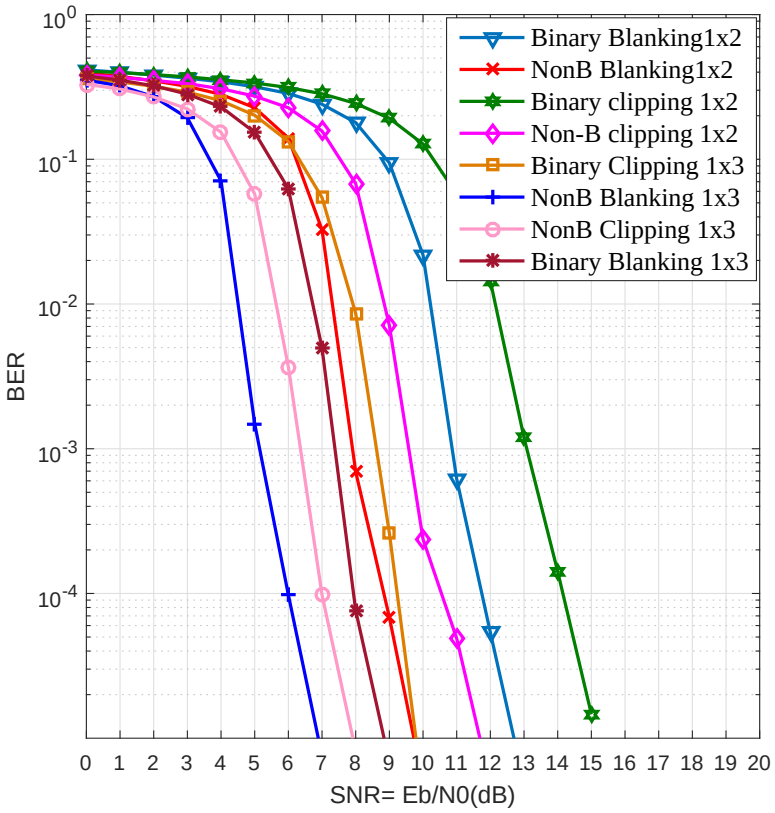

Figure 5.12: BER performance of binary and non-binary turbo coded 1x2, and 1x3 SIMO PLC-OFDM system, with the using of blanking and clipping, when Γ= 0.1 and A= 0.1, versus SNR (dB).

Fig 5.13 illustrates that the performance of the non-binary code exceeds the binary code by at least 2 dB for both clipping and blanking at a BER of 10^{-5}. Furthermore, Fig 5.13 also shows that blanking is always better than clipping.

Finally, Fig 5.14 shows the comparison between the proposed non-binary 2x3 MIMO PLC-OFDM system with the binary turbo, when the impulsive index A is taking various values (A=0.01,0.1,0.2 and 0.3), Γ= 0.1 and the blanking technique is applied.

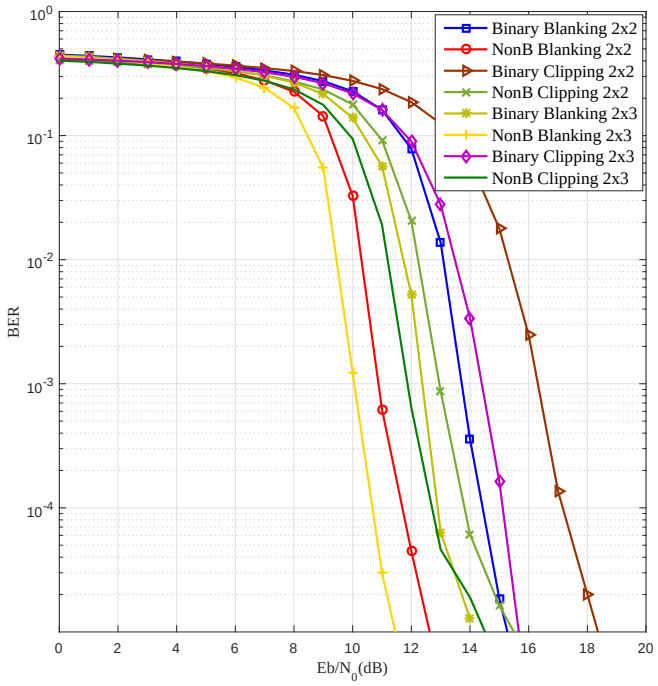

Figure 5.13: BBER performance of binary and non-binary turbo coded 2x2, and 2x3 MIMO PLC-OFDM system, with the using of blanking and clipping, when Γ= 0.1 and A= 0.1, versus SNR (dB).

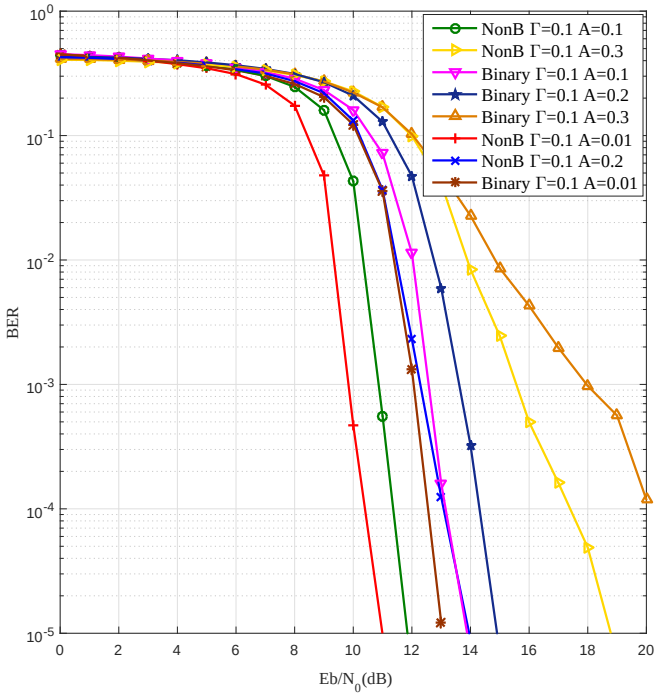

Figure 5.14: BER performance of binary and non-binary turbo coded 2x3 MIMO PLC-OFDM system, with the using of blanking, when Γ= 0.1 and A= [0.01, 0.1, 0.2, and 0.3], versus SNR (dB).

It should be noticed that employing non-binary turbo codes on MIMO PLC-OFDM systems can achieve significant coding gain for all different values of impulsive noise over the binary MIMO PLC-OFDM systems, notably by 2dB at a BER of 10^{-5}.

5.4 Summary

Impulsive noise is most commonly associated with in-home communications. Normally, impulsive noise has a duration of more than one pulse long and that causes a burst error, which significantly degrade the overall communication systems performance. For that and more, in this chapter the main focus was on implementing a robust communication system that can combat this harsh environment.

Middleton class A is one of the most impulsive noise types that affect communication system performance in many applications. That is because this kind of noise is generated by many natural or man-made phenomena. For instance, electromagnetic fields surrounding the telecommunication devices, thunderstorms and atmospheric phenomena.

However, adapting the conventional communication systems to resist the impact of impulsive noise was the main purpose of this chapter. In order to achieve this goal, various coded schemes have been examined on a very extreme environments. In addition, due to the complexity which has been added to the communication systems by considering these kind of channels, the analysis of only uncoded and binary and non-binary convolutional coded OFDM-MIMO on additive impulsive noise channels have been presented. The simulated and theoretical BER performance of wireless OFDM-MIMO systems has showed a close match in the high SNR region. It is worth mentioning that by using the same analysis of wireless OFDM-MIMO, the analysis of OFDM-MIMO on PLC channels can be achieved.

Furthermore, an investigation into non-binary turbo coded MIMO PLC-OFDM on different channels has been carried out. The transmitter and receiver structures have been presented and the Max-Log-MAP algorithm for non-binary codes employed as the component decoders, which were explained in chapter 4. The simulation results show that the proposed non-binary turbo coded MIMO PLC-OFDM

system achieves significant coding gains compared to the binary turbo coded system when the channel is impulsive.

This initial investigation has shown that there is much promise for non-binary turbo codes on impulsive noise channels and there is great scope for further research in this area.

Chapter 6

Thesis Summary and Future Work

6.1 Thesis Summary

Most studies consider conventional communication systems where the noise channel is AWGN. However, it is not always the case, due to man-made or natural phenomena that add unwanted noise to the transmitted signal. Also, AWGN is not suitable to model numerous applications, such as communication over power line transmission cables, underwater communications, and in-home telecommunications.

Basically, non-Gaussian additive noise generate samples with a higher energy and longer than one pulse duration as compared to AWGN symbols. Moreover, this type of additive noise gives rise to burst errors that degrade the communication system performance. In this thesis, two of the commonly used noise distribution to model the noise channel on PLC, underwater and in-home communications, have been implemented in various wired and wireless trellis coded and uncoded systems. The noise distributions that were considered in this thesis were; $S\alpha S$ stable and Middleton class A. The main contributions offered in this thesis are summarised below:

- In chapter 3, the BER performance of non-binary convolutional codes on the AWGN channel has been investigated, along with a comparison of the BER performance of binary convolutional codes. In order to validate this analysis approach, the simulated and theoretical BER performance of two non-binary codes ($\beta\beta^2/1$ and $\beta 1\beta/\beta^2 1$) have been presented. Both codes have a rate of $\frac{1}{2}$ and message length 1000 symbols, utilizing BPSK modulation.

  In addition, GF(4) has been considered in this chapter and the state tables of

both codes with an example of the signal-flow-graph for non-binary codes and a comparison between the performance of binary and non-binary convolutional codes has been carried out. Also, the transfer function and weight enumerators of non-binary convolutional codes have been derived. Furthermore, in this chapter, the theoretical results showed a good agreement with the simulated results.

Moreover, an investigation of binary and non-binary trellis codes on $S\alpha S$ stable channel have been carried out. A fair comparison between the $(1, 7/5)_8$ binary and $\beta\ \beta^2/1$ non-binary convolutional codes on $S\alpha S$ stable noise channel has been presented. Even though, these codes have a very close BER performance on AWGN, non-binary convolutional code showed great gain over binary codes on impulsive noise channels.

Finally, a comparison between rate $\frac{1}{3}$ binary and non-binary turbo codes on $S\alpha S$ stable noise channels has been presented and non-binary codes again achieved a significant coding gain.

- There are a large number of PLC technologies that are utilized by many applications. For example, Internet access, which is known as broadband over power line (BPL) and smart home applications. Thus, chapter 4 focused on the implementation of different coding schemes on PLC noise channels. Since the commonly used distribution to model the noise channel on PLC is Middleton class A, a brief exploration of these channel characteristics have been presented. The effect of the impulsive noise index A and the noise background ratio Γ, have been illustrated for single carrier coded and uncoded communication systems on additive impulsive noise channels.

  In addition, an investigation into MC binary and non-binary trellis coded systems on impulsive noise have been presented and a fair BER performance comparison between the binary and non-binary systems has been carried out. In order to validate the simulated results of MC binary and non-binary convolutional codes on Middleton class A noise channels, a comparison between simulated and theoretical results have been introduced in this chapter, and it showed a good agreement between them.

  With regards to examining the proposed systems on a realistic PLC environ-

ment, two multipath models for the powerline channel have been considered in the implementation. The simulated BERs of uncoded and coded binary and non-binary convolutional OFDM-PLC systems on multipath frequency selective channels have been compared with the analytical bounds for the first time. With a view to mitigating the impact of the multipath frequency selective channel and impulsive noise effects, OFDM has been utilized at the transmitter. Additionally, non-linear preprocessing techniques such as clipping and blanking have been implemented to reduce the high energy noise pulses.

Furthermore, non-binary turbo decoding and encoding on the PLC channel has been described in detail for SC and MC systems. Finally, the simulated BER performance of non-binary turbo coded OFDM-PLC systems has been presented and a comparison with binary coded systems has been carried out.

- Interest in MIMO systems has increased rapidly, therefore binary and non-binary trellis codes with OFDM-MIMO on impulsive noise for wireless applications have been implemented in chapter 5. validation of the simulated BER performance of binary and non-binary convolutional coded OFDM-MIMO on Middleton class A noise channel has been introduced for various noise parameters and for different MIMO scenarios.

  Utilizing the existing power line network cables to achieve reliable communications over these grids is of great interest .In addition, high data rates with large capacity are always the main target of any communication system. Therefore, in chapter 5, OFDM-MIMO trellis coded systems have been applied to electric grid wires to enhance the overall telecommunication system efficiency. Also, signal processing techniques have been implemented to reduce the negative effect of impulsive noise MIMO channels. These techniques are ODFM which is an effective tool to spread the impulsive noise energy equally over all symbols and reduce the impact of multipath channels, and blanking or clipping techniques. By the same token, the system model of non-binary turbo coded OFDM-MIMO-PLC has been explained in detail. An examination of the proposed system on synthetic MIMO-PLC channels, which is generated based on data set provided by ETSI, has been presented. This database consists of 353 MIMO CFRs and 1588 samples in the frequency range of 1.8-100

MHz. Finally, numerical and simulated results of the proposed system have been illustrated along with a fair comparison between the BER performance of binary and non-binary codes.

6.2 Future Work

Investigations into the BER performance of non-binary trellis codes on some wired and wireless applications over extreme environments were the main focus in this thesis. However, there is still scope to achieve further gains in performance by employing more recently proposed techniques or ideas that focus on channel estimation, energy and time consumption and noise cancellation. Some of these ideas and techniques that can be carried out on the future research are listed as follows :

- Employing non-binary coding exposed additional complexity to the decoding and encoding processes, but it is worth considering higher Galois fields such as; GF(8), GF(16),..., etc, to investigate their behaviour in impulsive noise environments. In addition, we could utilize other coding schemes on impulsive noise channels and making fair comparisons between them to find the robust codes for such harsh medium. For example, LDPC codes is one of the standard coding schemes used on PLC and we might employ it with non-binary fields and derive the corresponding analysis bounds.

- It is well known that the performance of non-binary codes is better than the binary, but it adds complexity to the communication system as we mentioned before. Therefore, reducing the complexity of their decoding algorithms would be one of the main considerations for future study.

- Energy consumption must also be taken into consideration since it is likely that the wireless network will comprise portable nodes powered by battery. It is therefore important to minimize the transmission power and the complexity of the encoding and decoding operations while maintaining an acceptable quality of service. Using optimization tools such as CVX with Matlab or other programming language, might be a great approach to these goals.

References

[1] F. Baldini and P. Farrell, "Coded modulation based on rings of integers modulo-q. 2. convolutional codes," *IEE Proc.-Commun.*, vol. 141, no. 3, pp. 137–142, 1994.

[2] R. Gallager, "Low-density parity-check codes," *IRE Transactions on information theory*, vol. 8, no. 1, pp. 21–28, 1962.

[3] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes. 1," in *Communications, 1993. ICC'93 Geneva. Technical Program, Conference Record, IEEE International Conference on*, vol. 2. IEEE, 1993, pp. 1064–1070.

[4] V. D. Goppa, "Codes on algebraic curves," in *Soviet Math. Dokl.*, vol. 24, 1981, pp. 170–172.

[5] D. Middleton, "Statistical-physical models of electromagnetic interference," *IEEE Transactions on Electromagnetic Compatibility*, no. 3, pp. 106–127, 1977.

[6] T. Shongwey, A. H. Vinck, and H. C. Ferreira, "On impulse noise and its models," in *Power Line Communications and its Applications (ISPLC), 2014 18th IEEE International Symposium on*. IEEE, 2014, pp. 12–17.

[7] M. Shao and C. L. Nikias, "Signal processing with fractional lower order moments: stable processes and their applications," *Proceedings of the IEEE*, vol. 81, no. 7, pp. 986–1010, 1993.

[8] S. C. Schwartz, J. B. Thomas, and E. J. Wegman, *Topics in non-Gaussian signal processing*. Springer, 1989.

[9] K. McClaning and T. Vito, *Radio receiver design*. Noble Publishing, 2000.

[10] S. Haykin, *Communication systems.* John Wiley & Sons, 2008.

[11] B. P. Lathi, *Modern Digital and Analog Communication Systems 3e Osece.* Oxford university press, 1998.

[12] J. Proakis, G. Bauch, and M. Salehi, *Modern Communication Systems Using MATLAB.* Cengage Learning, 2013.

[13] J. G. Gonzalez, J. L. Paredes, and G. R. Arce, "Zero-order statistics: a mathematical framework for the processing and characterization of very impulsive signals," *Signal Processing, IEEE Transactions on*, vol. 54, no. 10, pp. 3839–3851, 2006.

[14] G. A. Tsihrintzis and C. L. Nikias, "Performance of Optimum and Suboptimum Receivers in the Presence of Impulsive Noise Modeled as an Alpha-Stable Process," *Communications, IEEE Transactions on*, vol. 43, no. 2/3/4, pp. 904–914, 1995.

[15] V. M. Zolotarev, *One-dimensional stable distributions.* American Mathematical Soc., 1986, vol. 65.

[16] Z. Mei, M. Johnston, S. Le Goff, and L. Chen, "Performance analysis of LDPC-coded diversity combining on rayleigh fading channels with impulsive noise," *IEEE Transactions on Communications*, vol. 65, no. 6, pp. 2345–2356, 2017.

[17] D. Middleton, I. of Electrical, and E. Engineers, *An introduction to statistical communication theory.* IEEE press Piscataway, NJ, 1996.

[18] L. A. Berry, "Understanding middleton's canonical formula for class a noise," *IEEE Transactions on Electromagnetic compatibility*, no. 4, pp. 337–344, 1981.

[19] K. Vastola, "Threshold detection in narrow-band non-gaussian noise," *IEEE Transactions on Communications*, vol. 32, no. 2, pp. 134–139, 1984.

[20] S. Miyamoto, M. Katayama, and N. Morinaga, "Performance analysis of qam systems under class a impulsive noise environment," *IEEE transactions on electromagnetic compatibility*, vol. 37, no. 2, pp. 260–267, 1995.

[21] R. Haring and A. H. Vinck, "Performance bounds for optimum and suboptimum reception under class-a impulsive noise," *IEEE Transactions on Communications*, vol. 50, no. 7, pp. 1130–1136, 2002.

[22] K. Wiklundh, P. Stenumgaard, and H. Tullberg, "Channel capacity of middleton's class a interference channel," *Electronics letters*, vol. 45, no. 24, pp. 1227–1229, 2009.

[23] J. Proakis, *Digital Communications*, ser. McGraw-Hill series in electrical and computer engineering : communications and signal processing. McGraw-Hill, 2001.

[24] C. E. Shannon, "A mathematical theory of communication," *The Bell System Technical Journal*, vol. 27, no. 4, pp. 623–656, Oct 1948.

[25] H. Mercier, V. K. Bhargava, and V. Tarokh, "A survey of error-correcting codes for channels with symbol synchronization errors," *IEEE Communications Surveys Tutorials*, vol. 12, no. 1, pp. 87–96, First 2010.

[26] S. Lin and D. Costello, *Error-Correcting Codes*. 2nd ed. Paerson Prentice Hall, 2004.

[27] R. W. Hamming, "Error detecting and error correcting codes," *The Bell System Technical Journal*, vol. 29, no. 2, pp. 147–160, April 1950.

[28] P. Elias, "Coding for noisy channels," in *IRE International Convention Record*, 1955, pp. 37–46.

[29] E. Prange, *Cyclic Error-Correcting codes in two symbols*. Air force Cambridge research center, 1957.

[30] A. Hocquenghem, "Codes correcteurs d'erreurs," *Chiffres*, vol. 2, no. 2, pp. 147–56, 1959.

[31] R. C. Bose and D. K. Ray-Chaudhuri, "On a class of error correcting binary group codes," *Information and control*, vol. 3, no. 1, pp. 68–79, 1960.

[32] I. S. Reed and G. Solomon, "Polynomial codes over certain finite fields," *Journal of the society for industrial and applied mathematics*, vol. 8, no. 2, pp. 300–304, 1960.

[33] W. Peterson, "Encoding and error-correction procedures for the Bose-Chaudhuri codes," *IRE Transactions on Information Theory*, vol. 6, no. 4, pp. 459–470, 1960.

[34] W. W. Peterson and E. J. Weldon, *Error-correcting codes*. MIT press, 1972.

[35] R. Gallager, "Low-density parity-check codes," *IRE Transactions on Information Theory*, vol. 8, no. 1, pp. 21–28, January 1962.

[36] J. L. Massey, "Threshold decoding," MASSACHUSETTS INST OF TECH CAMBRIDGE RESEARCH LAB OF ELECTRONICS, Tech. Rep., 1963.

[37] G. Forney, "On decoding BCH codes," *IEEE Transactions on Information Theory*, vol. 11, no. 4, pp. 549–557, Oct 1965.

[38] D. J. MacKay and R. M. Neal, "Good codes based on very sparse matrices," in *IMA International Conference on Cryptography and Coding*. Springer, 1995, pp. 100–111.

[39] A. Viterbi, "Convolutional codes and their performance in communication systems," *IEEE Transactions on Communication Technology*, vol. 19, no. 5, pp. 751–772, October 1971.

[40] F. J. L. R. Bahl, J. Cocke and J. Raviv, "Optimal decoding of linear codes for minimal symbol error rate," *Information Theory, IEEE Transactions on*, vol. IT-20, pp. 284–287, 1974.

[41] G. Ungerboeck, "Channel coding with multilevel/phase signals," *IEEE transactions on Information Theory*, vol. 28, no. 1, pp. 55–67, 1982.

[42] A. J. Viterbi, "Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm," *Information Theory, IEEE Transactions on*, vol. 13, no. 2, pp. 260–269, 1967.

[43] P. Robertson, P. Hoeher, and E. Villebrun, "Optimal and sub-optimal maximum a posteriori algorithms suitable for turbo decoding," *European Transactions on Telecommunications*, vol. 8, no. 2, pp. 119–125, 1997.

[44] E. Arikan, "Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels," *IEEE Transactions on Information Theory*, vol. 55, no. 7, pp. 3051–3073, 2009.

[45] E. Arıkan, H. Kim, G. Markarian, U. Ozgur, and E. Poyraz, "Performance of short polar codes under ml decoding," *Proc. ICT MobileSummit,(Santander, Spain)*, pp. 10–12, 2009.

[46] T. K. Moon, "Error correction coding," *Mathematical Methods and Algorithms. Jhon Wiley and Son*, 2005.

[47] G. D. Forney, "The viterbi algorithm," *Proceedings of the IEEE*, vol. 61, no. 3, pp. 268–278, 1973.

[48] J. Hagenauer and P. Hoeher, "A Viterbi algorithm with soft-decision outputs and its applications," in *Global Telecommunications Conference and Exhibition'Communications Technology for the 1990s and Beyond'(GLOBECOM), 1989. IEEE.* IEEE, 1989, pp. 1680–1686.

[49] M. P. Fossorier, F. Burkert, S. Lin, and J. Hagenauer, "On the equivalence between sova and max-log-map decodings," *IEEE Communications Letters*, vol. 2, no. 5, pp. 137–139, 1998.

[50] T. Bayes, "An essay towards solving a problem in the doctrine of chances," *Studies in the History of Statistics and Probability*, vol. 1, pp. 134–153, 1970.

[51] J. Liu and G. Tu, "Iterative decoding of non-binary turbo codes using symbol based sova algorithm," in *Communications, Circuits and Systems Proceedings, 2006 International Conference on*, vol. 2. IEEE, 2006, pp. 689–693.

[52] R. A. Carrasco and M. Johnston, *Non-binary error control coding for wireless communication and data storage.* John Wiley & Sons, 2008.

[53] F. Baldini and P. G. Farrell, "Coded modulation based on rings of integers modulo-q. 2. convolutional codes," *IEE Proceedings - Communications*, vol. 141, no. 3, pp. 137–142, Jun 1994.

[54] R. A. Carrasco and P. G. Farrell, "Ring-TCM for fixed and fading channels: land-mobile satellite fading channels with QAM," *IEE Proceedings - Communications*, vol. 143, no. 5, pp. 281–288, Oct 1996.

[55] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes. 1," in *Communications, 1993. ICC '93 Geneva. Technical Program, Conference Record, IEEE International Conference on*, vol. 2, May 1993, pp. 1064–1070 vol.2.

[56] S. J. Johnson, *Iterative error correction: Turbo, low-density parity-check and repeat-accumulate codes.* Cambridge University Press, 2009.

[57] Z. Xie, R. T. Short, and C. K. Rushforth, "A family of suboptimum detectors for coherent multiuser communications," *IEEE Journal on Selected Areas in Communications*, vol. 8, no. 4, pp. 683–690, May 1990.

[58] A. Klein, G. K. Kaleh, and P. W. Baier, "Zero forcing and minimum mean-square-error equalization for multiuser detection in code-division multiple-access channels," *IEEE Transactions on Vehicular Technology*, vol. 45, no. 2, pp. 276–287, May 1996.

[59] A. Duel-Hallen, "A family of multiuser decision-feedback detectors for asynchronous code-division multiple-access channels," *IEEE Transactions on Communications*, vol. 43, no. 2/3/4, pp. 421–434, Feb 1995.

[60] F. Leferink, F. Silva, J. Catrysse, S. Batterman, V. Beauvois, and A. Roc'h, "Man-made noise in our living environments," *URSI Radio Science Bulletin*, vol. 2010, no. 334, pp. 49–57, Sept 2010.

[61] L. F. Lind and N. A. Mufti, "Efficient Method for Modelling Impulse Noise in a Communication system," *Electronics Letters*, vol. 32, no. 16, pp. 1440–1441, Aug 1996.

[62] M. Zimmermann and K. Dostert, "Analysis and Modeling of Impulsive Noise in Broad-Band Powerline Communications," *IEEE transactions on Electromagnetic compatibility*, vol. 44, no. 1, pp. 249–258, 2002.

[63] B. Hu and N. C. Beaulieu, "On Characterizing Multiple Access Interference in TH-UWB Systems with Impulsive Noise Models," in *Radio and Wireless Symposium, 2008 IEEE.* IEEE, 2008, pp. 879–882.

[64] W. Gu and L. Clavier, "Decoding Metric Study for Turbo Codes in Very Impulsive Environment," *Communications Letters, IEEE*, vol. 16, no. 2, pp. 256–258, 2012.

[65] D. Umehara, H. Yamaguchi, and Y. Morihiro, "Turbo decoding in impulsive noise environment," in *Global Telecommunications Conference, 2004. GLOBECOM'04. IEEE*, vol. 1. IEEE, 2004, pp. 194–198.

[66] A. Burnic, A. Hessamian-Alinejad, T. Scholand, T. E. Faber, G. H. Bruck, and P. Jung, "Error correction in impulsive noise environments by applying

turbo codes," in *Proc.of the 2006 IEEE Int. Symp. on Personal, Indoor and Mobile Radio Communications,*, 2006, pp. 1–5.

[67] H. Nakagawa, D. Umehara, S. Denno, and Y. Morihiro, "A decoding for low density parity check codes over impulsive noise channels," in *International Symposium on Power Line Communications and Its Applications, 2005.* IEEE, 2005, pp. 85–89.

[68] A. Hadi, K. M. Rabie, and E. Alsusa, "Polar codes based ofdm-plc systems in the presence of middleton class-a noise," in *Communication Systems, Networks and Digital Signal Processing (CSNDSP), 2016 10th International Symposium on.* IEEE, 2016, pp. 1–6.

[69] M. Zimmermann and K. Dostert, "A multipath model for the powerline channel," *IEEE Transactions on communications*, vol. 50, no. 4, pp. 553–559, 2002.

[70] H. C. Ferreira, L. Lampe, J. Newbury, and T. G. Swart, *Power Line Communications: Theory and Applications for Narrowband and Broadband Communications over Power Lines.* John Wiley & Sons, 2011.

[71] W. Abd-Alaziz, M. Johnston, and S. Le Goff, "Non-binary turbo codes on additive impulsive noise channels," in *Communication Systems, Networks and Digital Signal Processing (CSNDSP), 2016 10th International Symposium on.* IEEE, 2016, pp. 1–5.

[72] S. V. Zhidkov, "Performance analysis and optimization of ofdm receiver with blanking nonlinearity in impulsive noise environment," *IEEE transactions on vehicular technology*, vol. 55, no. 1, pp. 234–242, 2006.

[73] E. Alsusa and K. M. Rabie, "Dynamic peak-based threshold estimation method for mitigating impulsive noise in power-line communication systems," *IEEE Transactions on Power Delivery*, vol. 28, no. 4, pp. 2201–2208, 2013.

[74] K. Rabie, E. Alsusa, A. Familua, and L. Cheng, "Constant envelope ofdm transmission over impulsive noise power-line communication channels," in *Power Line Communications and its Applications (ISPLC), 2015 International Symposium on.* IEEE, 2015, pp. 13–18.

[75] C. Hsu, N. Wang, W.-Y. Chan, and P. Jain, "Improving a power line communications standard with ldpc codes," *EURASIP Journal on Advances in Signal Processing*, vol. 2007, no. 1, pp. 1–9, 2007.

[76] G. A. Al-Rubaye, C. C. Tsimenidis, and M. Johnston, "Non-binary LDPC Coded OFDM in impulsive power line channels," in *Signal Processing Conference (EUSIPCO), 2015 23rd European.* IEEE, 2015, pp. 1431–1435.

[77] N. Andreadou, C. Assimakopoulos, and F.-N. Pavlidou, "Performance Evaluation of LDPC Codes on PLC Channel Compared to other Coding Schemes," in *Power Line Communications and Its Applications, 2007. ISPLC'07. IEEE International Symposium on.* IEEE, 2007, pp. 296–301.

[78] W. Abd-Alaziz, Z. Mei, M. Johnston, and S. Le Goff, "Non-binary turbo-coded ofdm-plc system in the presence of impulsive noise," in *Signal Processing Conference (EUSIPCO), 2017 25th European.* IEEE, 2017, pp. 2576–2580.

[79] L. T. Berger, A. Schwager, P. Pagani, and D. Schneider, *MIMO power line communications: narrow and broadband standards, EMC, and advanced processing.* CRC Press, 2014.

[80] G. J. Foschini and M. J. Gans, "On limits of wireless communications in a fading environment when using multiple antennas," *Wireless personal communications*, vol. 6, no. 3, pp. 311–335, 1998.

[81] J. Mietzner, R. Schober, L. Lampe, W. H. Gerstacker, and P. A. Hoeher, "Multiple-Antenna Techniques for Wireless Communications-a Comprehensive Literature Survey," *IEEE communications surveys & tutorials*, vol. 11, no. 2, 2009.

[82] S. Galli and O. Logvinov, "Recent Developments in the Standardization of Power Line Communications within the IEEE," *IEEE Communications Magazine*, vol. 46, no. 7, 2008.

[83] L. T. Berger, A. Schwager, and J. J. Escudero-Garzás, "Power Line Communications for Smart Grid Applications," *Journal of Electrical and Computer Engineering*, vol. 2013, p. 3, 2013.

[84] L. Stadelmeier, D. Schill, A. Schwager, D. Schneider, and J. Speidel, "Mimo for inhome power line communications," in *Source and Channel Coding (SCC), 2008 7th International ITG Conference on*. VDE, 2008, pp. 1–6.

[85] J. Yoo, S. Choe, and N. Pine, "SISO/MIMO-OFDM Based Power Line Communication Using MRC," *Electrical Engineering and Control*, pp. 853–860, 2011.

[86] F. Versolatto and A. M. Tonello, "An MTL Theory Approach for the Simulation of MIMO Power-Line Communication Channels," *IEEE Transactions on power delivery*, vol. 26, no. 3, pp. 1710–1717, 2011.

[87] N. Pine and S. Choe, "Modified Multipath Model for Broadband MIMO Power Line Communications," in *Power Line Communications and Its Applications (ISPLC), 2012 16th IEEE International Symposium on*. IEEE, 2012, pp. 292–297.

[88] A. M. Tonello, F. Versolatto, B. Bejar, and S. Zazo, "A fitting algorithm for random modeling the plc channel," *IEEE Transactions on Power Delivery*, vol. 27, no. 3, pp. 1477–1484, 2012.

[89] A. Pittolo and A. Tonello, "A synthetic statistical mimo plc channel model applied to an in-home scenario," *IEEE Transactions on Communications*, 2017.

[90] C. R. Paul, *Analysis of Multiconductor Transmission Lines*. John Wiley & Sons, 2008.

[91] D. M. Pozar, *Microwave Engineering*. John Wiley & Sons, 2009.

[92] F. Versolatto and A. M. Tonello, "A MIMO PLC Random Channel Generator and Capacity Analysis," in *Power Line Communications and Its Applications (ISPLC), 2011 IEEE International Symposium on*. IEEE, 2011, pp. 66–71.

[93] W. Ryan and S. Lin, *Channel codes: classical and modern*. Cambridge University Press, 2009.

[94] P. Frenger, P. Orten, and T. Ottosson, "Convolutional codes with optimum distance spectrum," *IEEE Communications Letters*, vol. 3, no. 11, pp. 317–319, 1999.

[95] C. Schlegel, *Trellis coding*. Wiley Online Library, 1997.

[96] N. Kim, Y. Lee, and H. Park, "Performance analysis of mimo system with linear mmse receiver," *IEEE Transactions on Wireless Communications*, vol. 7, no. 11, 2008.

[97] G. L. Stüber, *Principles of mobile communication*. Springer, vol. 3.

[98] N. Kim and H. Park, "Bit error performance of convolutional coded mimo system with linear mmse receiver," *IEEE Transactions on Wireless Communications*, vol. 8, no. 7, 2009.

[99] M. Abramowitz and I. A. Stegun, *Handbook of mathematical functions: with formulas, graphs, and mathematical tables*. Courier Corporation, 1964, vol. 55.

[100] E. Biglieri, G. Caire, G. Taricco, and J. Ventura-Traveset, "Simple method for evaluating error probabilities," *Electronics Letters*, vol. 32, no. 3, pp. 191–192, 1996.

[101] M. R. McKay and I. B. Collings, "Error performance of MIMO-BICM with Zero-Forcing Receivers in Spatially-Correlated Rayleigh Channels," *IEEE Transactions on Wireless Communications*, vol. 6, no. 3, 2007.

[102] *PowerLine Telecommunications (PLT); MIMO PLT; Part 1: Measurement Methods of MIMO PLT*. document ETSI TR 101 562-1 V 1.3. 1, European Telecommunication Standardization Institute, 2012.

[103] A. Pittolo and A. Tonello, *Synthetic MIMO PLC Channel Emulator Software*, (Feb. 2017). [Online]. Available: http://www.andreatonello.com/openresearch

[104] *PowerLine Telecommunications (PLT); MIMO PLT; Part 3: Setup and Statistical Results of MIMO PLT Channel and Noise Measurments*. document ETSI TR 101 562-3 V 1.1. 1, European Telecommunication Standardization Institute, 2012.

[105] D. Tse and P. Viswanath, *Fundamentals of wireless communication*. Cambridge university press, 2005.