LDPC Coded OFDM And It's Application To DVB-T2, DVB-S2 And IEEE 802.16e

Edmond Nurellari

Submitted to the
Institute of Graduate Studies and Research
in partial fulfillment of the requirements for the degree of

Master of Science
in
Electrical and Electronic Engineering

Eastern Mediterranean University
January, 2012
Gazimagusa, North Cyprus

Approval of the Institute of Graduate Studies and Research

Prof. Dr. Elvan Yılmaz
Director

I certify that this thesis satisfies the requirements as a thesis for the degree of Master of Science in Electrical and Electronic Engineering

Assoc. Prof. Dr. Aykut Hocanın
Chair, Department of Electrical and Electronic Engineering

We certify that we have read this thesis and that in our opinion, it is fully adequate, in scope and quality, as a thesis of the degree of Master of Science in Electrical and Electronic Engineering

Assoc. Prof. Dr. Erhan A. İnce
Supervisor

Examining Committee

1. Assoc. Prof. Dr. Aykut Hocanın
2. Assoc. Prof. Dr. Erhan A. İnce
3. Assoc. Prof. Dr. Hüseyin Bilgekul

ABSTRACT

Since the invention of Information Theory by Shannon in 1948, coding theorists have been trying to come up with coding schemes that will achieve capacity dictated by Shannon's Theorem. The most successful two coding schemes among many are the LDPCs and Turbo codes. In this thesis, we focus on LDPC codes and in particular their usage by the second generation terrestrial digital video broadcasting (DVB-T2), second generation satellite digital video broadcasting (DVB-S2) and IEEE 802.16e mobile WiMAX standards. Low Density Parity Check (LDPC) block codes were invented by Gallager in 1962 and they can achieve near Shannon limit performance on a wide variety of fading channels. LDPC codes are included in the DVB-T2 and DVB-S2 standards because of their excellent error-correcting capabilities. LDPC coding has also been adopted as an optional error correcting scheme in IEEE 802.16e mobile WiMAX.

This thesis focuses on the bit error rate (BER) and PSNR performance analysis of DVB-T2, DVB-S2 and IEEE 802.16e transmission using LDPC coding under additive white Gaussian noise (AWGN) and Rayleigh Fading channel scenarios. The power delay profile for all transmissions was adopted from the ITU channel model. For modelling the fading environment, Jakes fading channel model[7] together with ITU Vehicular-A and ITU Vehicular-B[13] power delay profile parameters were used considering also the Doppler effect. The three scenarios presented in this thesis are the following: (i) simulation of LDPC coding for DVB-S2 standard, (ii) optional LDPC coding as suggested by the WiMAX standard and (iii) simulation of DVB-T2 using LDPC without outer BCH encoder and with outer BCH encoder. During the simulations the encoding algorithm used was the Forward Substitution algorithm.

Even though the second generation DVB standards and WiMAX standard has been out since 2009, not many comparative results have been published for BCH and LDPC concatenated coding schemes making use of either a normal FEC frame or a shortened FEC frame. By carrying out the work presented here we tried to contribute towards this end.

Throughout the simulations, we have considered two different size images as the source of information to transmit. Performance analysis have been presented by making comparisons between BER and PSNR values and psychovisually.

Keywords: Low Density Parity Check Coding; BCH coding; OFDM; WiMAX; Digital Video Broadcasting; Rayleigh Fading Channel; Shortening; Zero-Padding; Digital Image Processing; Iterative decoding.

ÖZ

1948 de Shannon tarafından bilişim kuram geliştirildikten sonra, bir çok kodlama kuramcısı Shanon teoreminde dikte edilen kapasiteye ulaşabilmek için farklı kodlama yöntemleri tasarlamışlardır. Bunlar arasında en başarılı alan ikisi, düşük yoğunluklu eşlik kontrol (DYEK) kodları ve Turbo kodlarıdır. Bu tezde ilgi odağı DYEK kodları ve bu kodların ikinci nesil yerüstü sayısal video yayıncılığı (DVB-T2), ikinci nesil uydu sayısal video yayıncılığı (DVB-S2) ve IEEE 802.16e mobil iletişim alanına uyarlanması olacaktır. Düşük yoğunluklu eşlik kontrol kodları 1962 de Gallager tarafındar keşfedilmiş ve sönümlemeli kanallar üzerinde Shanon sınırına yakın performans elde ettikleri gözlemlenmiştir. Bu özelliklerinden dolayı DYEK kodları DVB-T2 ve DVB-S2 standartlarında yerlerini almış ve IEEE 802.16e mobil WiMAX standardında ise CC ve RS-CC kodlama yöntemleri yanında bir seçenek olarak kabul görmüştür.

Bu tezde, bit hata oranı (BHO) ve tepe işaret gürültü oranı metrikleri kullanılarak DVB-T2, DVB-S2 ve IEEE 802.16e fiziki iletişim sistemlerinin toplanır beyaz Gaus gürültülü kanal ve sönümlemeli kanalla üzerindeki performans analizleri sunulmaktadır. Tüm senaryolarda kullanılan gecikme profili, ITU kanal modelinden alınmıştır. Sönümlemeli ortamı modelleme ise Jake kanal modeli ve ITU Taşıtsal- A ve Taşıtsal- B[13] güç gecikme profillerini kullanarak yapılmıştır. Modelleme Dopler değişimlerini de göz önüne almıştır.

Sunulan üç senaryo aşağıdaki gibidir: (i) DYEK destekli DVB-S2 benzetimleri, (ii) seçmeli DYEK destekli WiMAX benzetimleri ve (iii) DYEK veya DYEK-BCH seri bağlı kodlama destekli benzetimler. Benzetim çalışmaları esnasında kullanılan kodlama algoritması ileri ornatımlı bir algoritma idi.

Hem ikinci nesil sayısal video kodlama standardı, hem de WiMAX standardı, 2009 dan beri bilinmesine rağmen literatürde BCH ve DYEK kodlarını ardışık birleştiren ve hem normal FEC çerçevesi hem de kısaltılmıs FEC çerçevesi kullanan benzetim çalışmaları bulunmadığından bu çalışmayla bu alanda katkı koymaya çalışılmıştır.

Benzetim çalışmaları esnasında, boyutları farklı iki imge iletilmesi arzu edilen veri olarak kabul edilmiştir. Tezde, BHO, tepe sinyal gürültü oranı ve görüntüsel kaliteye bağlı kıyaslamalar sunulmaktadır.

Anahtar kelimeler: Düşük yoğunluklu eşlik kontrol kodları, BCH kodlama; OFDM; WiMAX; Sayısal Video Yayıncılığı, Rayleigh sönümlemeli kanal; Kısaltma; sıfır dolgulama; sayısal imge işleme; Özyineli kod çözümleme.

DEDICATION

Dedicated to my parents for their immense love and support.

ACKNOWLEDGMENTS

Looking back at the years I spent at Eastern Mediterranean University, I would like to express my deepest gratitude to my supervisor Assoc. Prof. Dr. Erhan A. İnce. Without his support and guidance this thesis would have been just a dream. It was his enthusiastic encouragement that firstly attracted me to work with him, then his appreciation led me to further successes in my academic career. I also would like to thank him for his suggestion and support to write this thesis in LaTeX.

My heartfelt thanks go to Assoc. Prof. Dr. Aykut Hocanın. It was his wonderful lectures on Communication Theory II and on Information Theory that first attracted me into this field. Not only Assoc. Prof. Dr. Aykut Hocanın insights and enthusiasm on research problems, but also his devotion to academic education have significantly shaped my future. Discussions and collaborations with him were always useful and informative.

My special thanks also go to Assoc. Prof. Dr. Hüseyin Bilgekul. It was his lecture on probability theory and stochastic processes and selected topics in digital communications which helped me to further improve my knowledge and understanding of communications systems.

I also would like to thank Assoc. Prof. Dr. Hüseyin Bilgekul, Assoc. Prof. Dr. Aykut Hocanın and Assoc. Prof. Dr. Hasan Demirel for serving in my MS examination comittee, suggesting corrections for my thesis manuscripts, and giving their insightful advice.

Many thanks to all the professors and personnel at the Electrical and Electronic Engineering department for providing their help and support during my course of study and stay at EMU.

I am indebted to all of my friends for their presence and help in my stay at EMU. Their presence has always motivated and encouraged me.

Lastly, I would like to thank my family. My parents, Fatos Nurellari and Libonike Nurellari, taught me those most valuable ”theorems of life” which I could not find them in any book or in any paper. Their love and support have been with me in every moment of my life. I am deeply indebted to them, as well as to my brothers Levent Nurellari and Evi Nurellari. Not to forget my grandmother for caring so much for me. This thesis is dedicated to my family and to her as an inadequate but sincere expression of appreciation and love.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

B	Transmission bandwidth (hertz)
C	Channel capacity (bits/s)
$c_n(t)$	The tap coefficients
c_i	Check node
$c_r(t)$ and $c_i(t)$	Gaussian with zero mean values
$\hat{c}w_i$	Hard decision decoding output
$c_0, c_1, c_2, c_3, ..., c_n$	Codeword
d_l	Maximum variable nodes degree
d_r	Maximum check nodes degree
E_b/N_0	Energy per bit to noise power spectral density ratio
$f(\alpha)$	PDF of Rayleigh fading signal amplitude
f_c	Carrier frequency
fd	Doppler frequency associated with Rayleigh fading channels
f_m	Maximum doppler frequency
GF	Galois Field
$g(t)$	Complex envelope
$g(x)$	Generator polynomial
H	Parity check matrix
$h(\tau;t)$	Temporal dispersion of the time-variant wireless propagation channels
I_{n-k}	Identity matrix
k	Length of input message
K_{bch}	Number of bits of BCH uncoded Block

K_{ldpc}	Number of bits of LDPC uncoded Block
K_{sig}	Input binary data that have to be transmitted
$L(c_i)$	Initial Log likelihood ratio value
$L(Q_i)$	Soft decoding output
M	Number of OFDM symbols
m	Number of parity check bits in the code
$m_0, m_1, m_2, ..., m_k$	Message bits
N	Number of sinusoids in Jakes' fading simulator
N_{bch}	Number of bits of BCH coded Block
N_{ldpc}	Number of bits of LDPC coded Block
N_{group}	Number of bit-groups for BCH shortening
N_{pad}	Number of BCH bit-groups in which all bits will be padded
N_0	Single-sided noise power spectral density (watts/hertz)
n	Code length
$n_{k,t}$	zero mean Gaussian noise with variance $N_0/2$
P	Received signal power (watts)
P	Coefficient matrix
$P(c_i\|y_i)$	Probability value for given input y_i
QC	Quasi- Cyclic coding techniques
R	Code rate
v_i	Variable node
w_c	Number of 1's in each column
w_r	Number of 1's in each row
$1/W$	Time resolution
σ^2	Channel noise variance

α	Normalized Rayleigh fading factor
$\alpha(t)$	Rayleigh fading signal amplitude
$\lambda(x)$	Degree polynomials for parameterizing irregular LDPC codes
$\lambda_i(x)$	Fractions of edges belonging to degree-i variable and check nodes
π_s	Permutation operator
$\phi(t)$	Independent random variable being uniform on $[0, 2\pi]$
$\rho(x)$	Degree polynomials for parameterizing irregular LDPC codes
$\rho_i(x)$	Fractions of edges belonging to degree-i variable and check nodes
AWGN	Additive White Gaussian Noise
BBFRAME	The set of K_{BCH} bits which form the input to one FEC encoding process
BCH	Bose- Chaudhuri- Hochquenghem multiple error code
BER	Bit Error Rate
BPA	Believe Propagation Algorithm
bps	Bit per second
CP	Cyclic Prefix (copy of the last part of OFDM symbol)
DMT	Discrete Multitone
DSNG	Digital Satellite News Gathering
DVB	Digital Video Broadcasting project
DVB-S	Digital Video Broadcasting- Satellite
DVB-S2	Second generation Digital Video Broadcasting-Satellite
DVB-T	Digital Video Broadcasting- Terrestrial specified in EN 300 421
DVB-T2	Second generation Digital Video Broadcasting-Terrestrial
ETSI	European Telecommunications Standards Institute
FDX	Full Duplex (communication channel)
FEC	Forward error correction

FECFRAME	The set of N_{ldpc} (16200 or 64800) bits from one LDPC encoding operation.
FFT	Fast Fourier Transform
girth	Length of the shortest cycles in the code's Tanner graph
HDX	Half Duplex (communication channel)
ICI	Inter Carrier Interfierence
IFFT	Inverse Fourier Transform
IMT-2000	International Mobile Telecommunications-2000
IRA	Irregular Repeat- Accumulate
ISDN	Integrated Services Digital Network
ISI	Inter Symbol Interfierence
ITU	International Telecommunications Union
LDPC	Low Density Parity Check (codes)
LLR	Log-likelihood Ratio
MCM	Multi Carrier Modulation
MPA	Message Passing Algorithm
NFFT	Size of FFT
OFDM	Orthogonal Frequency- Division Multiplexing
PSTN	Public Switched Telephone Network
QAM	Quadrature Amplitude Modulation
QC	Quasi Cyclic codes are generalization of cyclic codes
QPSK	Quadrature Phase Shift Keying
RMS	Root Mean Square
RS	Reed Solomon
RS-CC	Reed Solomon- Convolution Code
SNR	Signal-to-noise Ratio

SPA	Sum- Product Algorithm
Tanner Graph	Bipartite graph used to specify error correcting codes
TC˙s	Turbo Codes
TV	Television
UMTS	Universal Mobile Telecommunications System
WiMAX	Worldwide Interoperability for Microwave Access
8PSK	8-ary Phase Shift Keying
16APSK	16-ary Amplitude Phase Shift Keying
16QAM	16-ary Quadrature Amplitude Modulation
32APSK	32-ary Amplitude Phase Shift Keying

Chapter 1

INTRODUCTION

Modern communication systems aim to transmit information from one point to another over a communication channel, with high performance using efficiently the limited sources available. The need to transmit digital multimedia over wireless channels and through the satellite has become an important issue over the years motivated by the freedom provided by wireless mobile networks to its users in terms of mobility and continuous network connectivity. The challenge of the wireless channel however is overwhelming. Thus researchers have come up with various solutions to minimize or possibly overcome the adverse effects of the channel. Advanced technologies such as WiMAX [1], DVB-T and DVB-T2[2] have been developed to meet the needs of the teeming consumers. Such technologies have gained acceptance because of their capabilities to reliably deliver multimedia content to end users.

Some of the FEC schemes adopted by the above mentioned standards include convolutional coding, Reed Solomon (RS) coding, LDPC coding and/or concatenated BCH and LDPC coding. In concatenated coding typically, there is an outer code and an inner code. The code rate and the data rate of the transmission is mainly controlled by the inner code[3]. After FEC, the data is modulated either by vector modulation, amplitude modulation, frequency modulation or in this case, orthogonal frequency multiplexing (OFDM). OFDM is suitable for outdoor mobile communications because of its advantageous features[4]. The disadvantages associated with the technology come at a relatively cheap cost; thus making it the choice modulation for WiMAX, DVB-S2and DVB-T2 schemes.

Low-density parity-check codes and Turbo Codes (TCs)[5] are among the known FEC codes that give performances nearing the Shannon limit. In this work we have chosen to concentrate on LDPC usage instead of the TCs since LDPC decoding algorithms have more parallelism, less implementation complexity, less decoding latency linear and time complex algorithms for decoding[6].

1.1. Background

In 1948 Claude Shannon published a landmark paper in information theory for AWGN channel which is referred to as the noisy channel coding theorem[4]. Shannon's Theorem gives an upper bound to the capacity of a link, in bits per second (bps), as a function of the available bandwidth and the signal-to-noise ratio of the link.[1].

Stated by Claude Shannon in 1948, the theorem describes the maximum possible efficiency of error-correcting methods versus levels of noise interference and data corruption. He proposed forward error correcting (FEC) codes but he didn't describe how to construct the error-correcting method, however the theorem tells us how good the best possible method can be. In fact, it was shown that LDPC codes can reach within 0.0045 dB of the Shannon limit (for very long block lengths).[2]. Hence, finding a practical solution to this problem was left open to the scientific community.

Forward error correcting codes selectively introduce redundant bits into the transmitted data packet which aid to correct bit errors introduced by noise in the received data stream at the receiver. Low-density parity-check (LDPC) codes are a class of linear block LDPC codes. The name comes from the characteristic of their parity-check matrix which contains only a few 1's in comparison to the amount of 0's. By introducing redundant bits to reduce bit error rate is gained at the cost of reducing data transmission rate. In the following years, iterative decoding algorithm were the main focus of coding theorists. It was already stated by Gallager in 1962 that LDPC codes are suitable for iterative decoding algorithm but due to lack of required hardware at that time they were almost forgotten. It took almost forty five years for communication researchers to find computationally feasible FEC codes over AWGN channels, capable of delivering low bit error rate close to the channel capacity limit as suggested by Shannon. These outstanding codes named "turbo codes" were first presented by Berrou, Glavieux and Thitimajshima[10] in 1993.

The requirement of of high data transmission reliability and efficiency in the mobile multimedia and digital video broadcasting services puts forward a great challenge for channel coding techniques. Rediscovered by Mackey and Neal in 1990's [5], LDPC codes has recently become a hot research topic because of their excellent properties. They are considered as strong competitor of Turbo Codes especially when used in fading channel. Their inherent interleaving property as discussed in [6] due to random generation of the parity-check matrix makes LDPC an excellent choice for data transmission over fading channels.

Before the rediscovery of LDPC codes by Mackay *et al.*, only work by Tanner [8] and Wiberg [9] used Gallager's codes. Later, the idea of LDPC codes was extended to irregular LDPC codes by Luby *et al.* [11, 12] which even provide superior performance in comparison to their regular counterparts. After this fundamental theoretical work, turbo and LDPC codes moved into standards like DVB-S2, DSL, WLAN, WiMax, etc. and are under consideration for others.

1.2. Thesis Description

Our simulations were carried out for additive White Gaussian Noise channel and a fading channel with AWGN. For the fading channel, the Jakes fading channel model [7] together with ITU Vehicular-A and ITU Vehicular-B [13] power delay profile parameters were used considering also the Doppler effect. LDPC codes that supports DVB-S2, DVB-T2 and

WiMAX (IEEE802.16e) standard will be presented in this thesis. Flat fading channel is assumed throughout for all standards.

In this thesis, the Forward Substitution decoding algorithm is used for DVB-S2, DVB-T2 and WiMAX. Three scenarios are presented in the paper: simulation of DVB-S2 using the specified LDPC coding, simulation of optional LDPC coding as suggested by the WiMAX standard and simulation of DVB-T2 using LDPC with or without outer BCH encoding.

The reminder of this thesis is organized as follows. Chapter 2 introduces a description of the AWGN and Jakes fading channel models. The normalized probability density functions along with their mean and variance for Rayleigh, distribution are also provided to understand the characteristics of fading models. Chapter 3 introduces and defines the concept of LDPC codes and the concept of representing a code (or more specifically, it's parity check matrix) in terms of a bipartite graph. We present the hard decision iterative decoding algorithm as well. Lastly, we also introduce how to design the Quasi- Cyclic LDPC codes, which are used in IEEE 802.16e standard and Irregular Repeat- Accumulate (IRA) LDPC codes used in second generation Digital Video Broadcasting.

The practical issues related to implementation of LDPC codes in two of the standards are discussed in Chapter 4. We discuss the importance of the code length choice and the code rate on the performance of the FEC scheme. In Chapter 5 we provide an overview of our transmission block diagram that is simulated using MATLAB to evaluate the error correction ability of the LDPC FEC scheme and compare it with RS-CC. We also discuss the various assumption under which the FEC schemes are compared. Chapter 6 is completely devoted to presenting and analyzing our experimental results. We present the BER vs. Eb/N0 curves for different code rates and different standards. We also provide the recovered image under different code rate and different standard and discuss the performance of our systems. Finally, in the concluding chapter of this thesis, Chapter 7, we provide a summary of this thesis, state important conclusions that we have reached, and discuss recommendations that can be taken into consideration for future work on closely related topics.

Chapter 2

SYSTEM MODEL

Shannon in his landmark paper stated that, if the information or entropy rate is below the capacity of the channel, then it is possible to encode information messages and receive them without errors even if the channel distorts the message during transmission [25]. Recent developments in coding theory, have come out with channel codes which have performance very close to the channel capacity. Use of error control coding has become a crucial part of the modern communication system. A typical Digital communication model is represented by block diagram as shown in Figure 2.1. This model is suitable from coding theory and signal processing point of view. Information is generated by source which may be human speech, data source, video or a computer. This information is then transformed to electric signals by source encoder which are suitable for digital communication system. To ensure reliable transmission over communication channel encoder is introduced which accumulate redundant bits to the user information. The modulator is a system component which transforms the message to signal suitable for the transmission over channel.

In communications, a communication channel, or channel, refers to a physical transmission medium such as a wire, or to a logical connection over an environmental medium such as a wireless channel. A channel is used to convey an information signal, fin our study a digital bit stream, from transmitters to receivers. Error may be introduced from the channel noise during transmission, so FEC encoder and decoder blocks must be design in such a way to possibly minimize the errors introduced by channel.

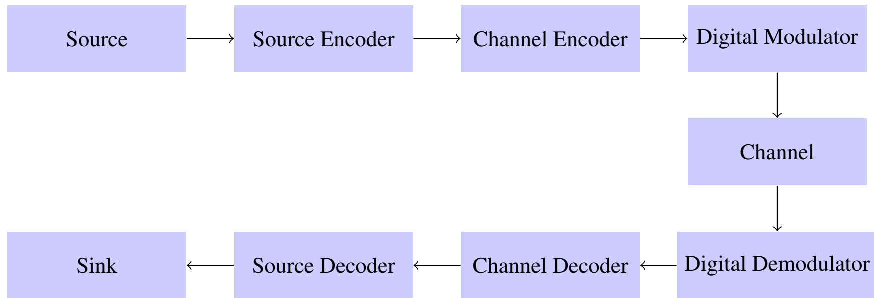


Figure 2.1: Basic Elements of Digital Communication System.

2.1. Channel Modeling

The channel is defined as a single path for transmitting signals in either one direction only HDX or in both directions FDX. The aim of wireless channel modeling is to find useful analytical models for the variations in the channel. The most prominent drawback of the wireless communications is channel fading. Various properties such as multipath propagation, terminal mobility and user interference, result in channel with time-varying parameters. Fading of the wireless channel can be classified into large-scale and small-scale fading. Large-scale fading involves the variation of the mean of the received signal power over large distances relative to the signal wavelength. On the other hand, small-scale fading involves the modulation and demodulation schemes that are robust to these variations. Hence we focus on the small scale variations in this class. Reflection, diffraction and scattering in the communication channel causes rapid variations in the received signal. The reflected signals arrive at different delays which cause random amplitude and phase of the received signals. This phenomenon is called multipath fading. If the product of the root mean square (RMS) delay spread (standard deviation of the delay spread) and the signal bandwidth is much less than unity, the channel is said to suffer from fading.The relative motion between the transmitter and the receiver (or vice versa) causes the frequency of the received signal to be shifted relative to that of the transmitted signal. The frequency shift, or Doppler frequency, is proportional to the velocity of the receiver and the frequency of the transmitted signal . A signal undergoes slow fading

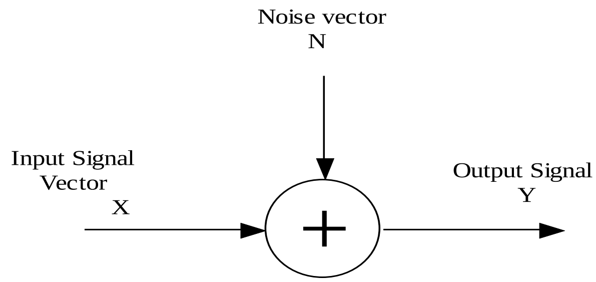


Figure 2.2: Additive white Gaussian noise channel model.

when the bandwidth of the signal is much larger than the Doppler spread (defined as a measure of the spectral broadening caused by the Doppler frequency). The combination of the multipath fading with its time variations causes the received signal to degrade severely. This degradation of the quality of the received signal caused by fading needs to be compensated by various techniques such as diversity and channel coding. In the forthcoming subsections we will briefly discuss a few of standard channel models which we will frequently use in our simulations.

2.1.1. AWGN Channel

Additive white Gaussian noise (AWGN) is a channel model which can be expressed as linear addition of wideband or white noise with a constant spectral density and an amplitude of Gaussian distribution [14]. Any wireless system in AWGN channel can be expressed as $y = x + n$, where n is the additive white Gaussian noise, x and y are the input and output signals respectively. The AWGN channel model does not account for fading, frequency selectivity or dispersion. The source of Gaussian noise comes from many natural sources such as thermal vibrations of atoms in antennas, shot noise, black body radiation from the warm objects and etc. However this channel is very useful model for many satellite and deep space communication links. The AWGN channel can be illustrated as in Figure 2.2 Channel capacity formula is a function of channel characteristics such as received signal and noise powers. As a matter of fact a number of different formulas are commonly used for calculating channel capacity. For additive Gaussian noise channel the channel capacity can be expressed as in (2.1).

$$C = B\log_2\left(1+\frac{P}{N_0B}\right) \tag{2.1}$$

where,

C=channel capacity (bits/s)

B=transmission bandwidth (hertz)

P=received signal power (watts)

N_0= single-sided noise power spectral density (watts/hertz)

2.1.2. Rayleigh Fading Channel

The Rayleigh fading channel, usually referred as a worst-case fading channel is a statistical model for the effect of a propagation environment on a radio signal, such as that used by wireless devices [15]. It assumes that the magnitude of a signal that has passed through such a transmission medium (also called a communications channel) will vary randomly, or fade, according to a Rayleigh distribution. Received signal can be modeled as $y = \alpha * t_e + n$. The "α" is the normalized Rayleigh fading factor and related to the fading coefficient of the channel $c(t)$ through $\alpha = |c(t)|$, where the real and imaginary components of $c(t)$ are Gaussian random variables. If sufficient channel interleaving is introduced, then fading coefficients of $c(t)$ are independent. Rayleigh fading is viewed as a reasonable model for heavily built-up urban environments on radio signals [24]. Rayleigh fading is most applicable when there is no dominant propagation along a line of sight between the transmitter and the receiver. If there is a dominant line of sight, Rician fading may be more applicable. A general model for time-variant multipath channel is shown in figure 2.3. The channel model consists of a tapped delay line with uniformly spaced taps. The tap spacing is $1/W$, where W amount of the signal transmitted through the channel. As a result $1/W$ is the time resolution that can

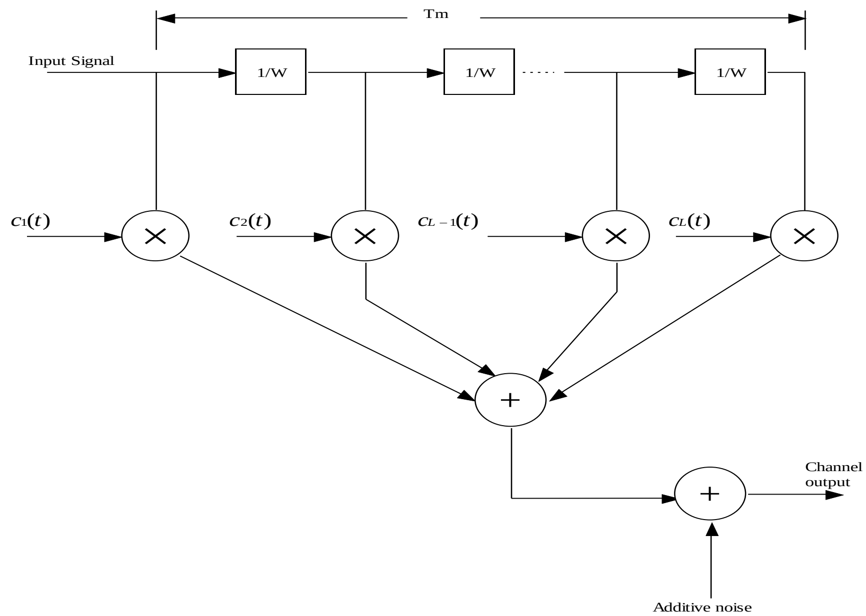


Figure 2.3: Model for time-invariant multipath channel[50].

possibly be achieved by transmitting a signal with bandwidth W. The tap coefficients are denoted as $c_n(t) \equiv \alpha_n(t) \exp^{j\phi_n(t)}$ are usually modeled as complex valued, Gaussian random processes. Each of the tap coefficients can be expressed as

$$c(t) = c_r(t) + jc_i(t) \tag{2.2}$$

$$c(t) = \alpha_t e^{j\phi(t)} \tag{2.3}$$

where

$$\alpha(t) = \sqrt{c_r^2(t) + c_i^2(t)} \tag{2.4}$$

$$\phi(t) = \tan^{-1} \frac{c_i(t)}{c_r(t)} \tag{2.5}$$

The Rayleigh fading signal amplitude is described by the PDF

$$f(\alpha) = \frac{\alpha}{\sigma^2} e^{-\alpha^2/2\sigma^2}, \alpha \geq 0. \tag{2.6}$$

In this representation "$c_r(t)$" and "$c_i(t)$" are Gaussian with zero-mean values, the amplitude $\alpha(t)$ is characterized statistically by the Rayleigh probability distribution and $\phi(t)$ is independent random variable which is uniform on [0, 2π].

2.1.3. ITU Vehicular- A & ITU Vehicular- B channel Model

The ITU Vehicular-A and the ITU Vehicular-B adopted channel models are empirical, based on measured data in the field. They are well-established channel models for research purposes in mobile communication systems. Moreover specification of channel conditions for various operating environments encountered in third-generation wireless systems, e.g the UMTS Terrestrial Radio Access System (UTRA) standardized by 3GPP are well defined. The ITU channel models are in fact approximating the temporal dispersion of the time-variant wireless propagation channels, $h(\tau;t)$, in a model with discrete tapped-delay-line with K taps.

$$h(\tau;t) = \sum_{k=1}^{K} a_k \delta(\tau - \tau_k). \tag{2.7}$$

The tapped-delay-line parameters for ITU Vehicular-A channel and ITU Vehicular-B channel are shown in Table 2.1 and Table 2.2 respectively.

The tapped-delay-line parameters for ITU Vehicular-B channel are shown in Table 2.2.

2.1.4. Jakes' Fading Simulator

Jakes' model which is based on summation of sinusoids can be easily modeled as described in [7]. The aim is to produce a signal that possesses the same Doppler spectrum as that of the classic Doppler spectrum. Details of the channel model depicted in Figure 2 can be found in [7]. It is possible for one to simulate this model by generating the $x(t)$ and $y(t)$ which

Table 2.1: Tapped-Delay-Line Parameters for ITU Vehicular A Channel

Tap Index	Relative delay(ns)	Average power (dB)
1	0	0
2	310	-1
3	710	-9
4	1090	-10
5	1730	-15
6	2510	-20

Table 2.2: Tapped-Delay-Line Parameters for ITU Vehicular B Channel

Tap Index	Relative delay(ns)	Average power (dB)
1	0	-2.5
2	300	0
3	8.900	-12.8
4	12900	-10
5	17100	-25.2
6	20000	-16

constitute the in-phase and quadrature parts of the complex envelope g(t). Jakes' model is based on summing sinusoids as defined by the following equations:

$$g(t) = x(t) + jy(t) \tag{2.8}$$

$$\begin{aligned} g(t) = \sqrt{2}\Bigg\{ & \left[\sum_{n=1}^{M} \cos\beta_n \cos 2\pi f_n t + \sqrt{2}\cos 2\pi f_m t\right] \\ & + j\left[2\sum_{n=1}^{M} \cos\beta_n \cos 2\pi f_n t + \sqrt{2}\sin\alpha \cos 2\pi f_m t\right]\Bigg\} \end{aligned} \tag{2.9}$$

$$\alpha = \hat{\phi_N} = -\hat{\phi_{-N}} \tag{2.10}$$

where,

$$\beta_N = \hat{\phi_n} = -\hat{\phi_{-n}} \tag{2.11}$$

$\hat{\phi}$ is the random phase given by:

$$\hat{\phi} = -2\pi(f_c + f_m)\tau_n$$

where:

$$f_m = \frac{v}{\lambda_c}$$

is the maximum Doppler frequency, and f_c is the carrier frequency.In the fading simulator there are M low frequency oscillators with frequency f_n=$f_m cos 2\pi n$, $n = 1,2,3,...,M$, where M=$\frac{1}{2}(\frac{N}{2} - 1)$, and N is the number of sinusoids. The amplitudes of the oscillators are all

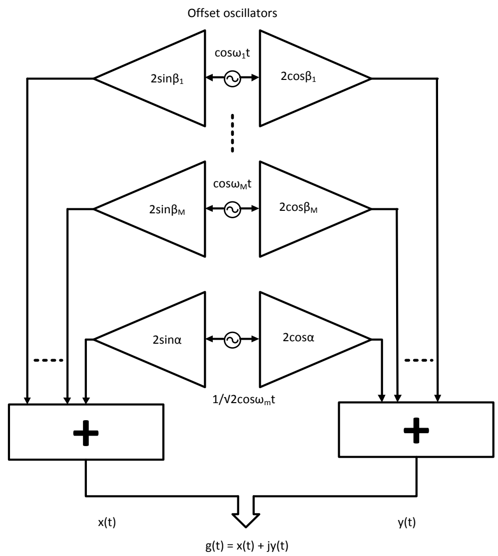


Figure 2.4: Jakes' fading channel model [7].

unity except for the oscillator at frequency f_m which has amplitude $\frac{1}{\sqrt{2}}$. Note that Figure 2.4 implements 5 low frequency oscillators except for the scaling factor of $\sqrt{2}$. It is desirable that the phase of (5) be uniformly distributed. Jakes' model which is based on summation of sinusoids can be easily modeled as described in [7]. The aim is to produce a signal that possesses the same Doppler spectrum as that of the classic Doppler spectrum.

2.2. OFDM-based Wireless Communication systems

Orthogonal frequency-division multiplexing (OFDM), in some cases known as multicarrier modulation (MCM) or discrete multitone (DMT) is a well known modulation technique that is tolerant to channel disturbances and impulse noise. Multi carrier modulation have been developed 1950's by introducing two modems, the Collins Kineplex system [18] and the one so called Kathryn modem[19]. OFDM has remarkable properties such as bandwidth efficiently,highly flexible in terms of its adaptability to channels and robustness to multipath. OFDM is used in many applications including high data rate transmission over twisted pair lines and fiber, digital video broadcasting terrestrial (DVBT), personal communications services and etc.

2.2.1. OFDM

To achieve higher spectral efficiency in multicarrier system, the sub-carriers must have overlapping transmit spectra but at the same time they need to be orthogonal to avoid complex separation and processing at the receiving end [48]. As it is stated in [48], the orthogonal set can be represented as such:

$$\psi_k(t) = \left\{ \frac{1}{\sqrt{T_s}} \exp^{jw_k t} \, for \, t \in [0, T_s] \right\} \tag{2.12}$$

$$with \; w_k = w_0 + kw_s; \quad k = 0, 1, ..., N_c - 1 \tag{2.13}$$

w_0 is the lowest frequency used and w_k is the subcarrier frequency. Multicarrier modulation schemes that fulfil above mentioned conditions are called orthogonal frequency division multiplex (OFDM) systems. Instead of baseband modulator and bank of matched filters, Inverse Fast Fourier Transform (IFFT) and Fast Fourier Transform (FFT) is efficient method of OFDM system implementation as shown in Figure 3.1 since it is cheap and does not suffer from inaccuracies in analogue oscillators. Inter symbol interference occurs when the signal passes through the time dispersive channel. In an OFDM system, it is also possible that orthogonality of the subscribers may be lost, resulting in inter carrier interference. OFDM system uses cyclic prefix (CP) to overcome these problems. A cyclic prefix is the copy of the last part of the OFDM symbol to the beginning of transmitted symbol and removed at the receiver before demodulation. The cyclic prefix should be at least as long as the length of impulse response. The use of prefix has two advantages: it serves as guard space between successive symbols to avoid ISI and it converts linear convolution with channel impulse re-

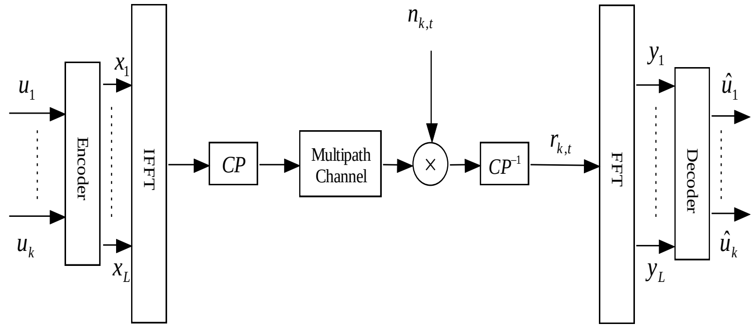


Figure 2.5: Model of OFDM system [41].

sponse to circular convolution. As circular convolution in time domain translates into scalar multiplication in frequency domain, the subcarrier remains orthogonal.Moreover there is no ICI . In Figure 3.1, L coded vector x_i are generated by proper coding, interleaving and mapping. After adding cyclic prefix, OFDM signal is passed through multipath channel. At the receiver the cyclic prefix is removed and received signal is passed through FFT block to get L received vectors y_i; where $n_{k,t}$ are zero mean Gaussian noise with variance $N_0/2$ of k_th sample of the t_th OFDM symbol. N_0 is the noise power, $k = (1,2,...,NFFT-1)$ and $t = (1,2,...,M)$, where M is the number of OFDM symbols and NFFT is the size of FFT.

Chapter 3

LDPC CODES

Low-density parity-check (LDPC) codes are a class of linear block LDPC codes. An H matrix with size m by n is low density because the number of 1s in each row w_r is$<< m$ and the number of 1s in each column w_c is $<< n$. A LDPC is regular if w_c is constant for every column and $w_r = w_c(n/m)$ is also constant for every row. Otherwise it is irregular. In LDPC encoding, the codeword $(c_0, c_1, c_2, c_3, ..., c_n)$ consists of the message bits $(m_0, m_1, m_2, ..., m_k)$ and some parity check bits and the equations are derived from H matrix in order to generate parity check bits. Their main advantage is that they provide a performance which is very close to the capacity for a lot of different channels and linear time complex algorithms for decoding. Furthermore they are suited for implementations that make heavy use of parallelism. They were first introduced by Gallager in his PhD thesis in 1960. But due to the computational effort in implementing decoder and encoder for such codes and the introduction of Reed-Solomon codes, they were mostly ignored until about ten years ago.

3.1. Regular LDPC Codes

Regular LDPC codes have been and are still playing a crucial role in the history of LDPC coding. Different types of regular coding can be stressed in coding theory field. Mainly, the well known ones can be listed as follows: Gallager Codes, Quasi-Cyclic Codes, Array Codes and Random Codes. Moreover different code rates are possible for different techniques.

A LDPC code is regular if the number of 1s in column w_c and the number of 1s in row w_r are constant for a given parity-check matrix. A sample of regular matrix is shown in (3.1)

$$H = \begin{bmatrix} 0 & 1 & 0 & |1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & |0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & |0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & |1 & 1 & 0 & 1 & 0 \end{bmatrix} \tag{3.1}$$

The example matrix from (3.1) is regular with w_c=2 and w_r=4. It is also possible to see the regularity of this code while looking at the graphical representation in Figure 3.1. There is the same number of incoming edges for every v-node and also for all the c-nodes.

As we mentioned above, Low-density parity-check (LDPC) codes are used as optional coding schemes in IEEE 802.16e (WiMAX) [28]. The base model matrices given in the standard for different code rate are fully based on quasi-cyclic (QC) coding techniques. Given the base model matrix, the parity-check matrix H can be generated from blocks of permutation sub-matrix [29]. In section *Constructing Quasi-cyclic LDPC codes* will be given a guide and criterions how to construct those QC LDPC codes.

3.2. Irregular LDPC Codes

A LDPC code is irregular if the number of 1s in columns and rows are not constant for a given parity-check matrix. Irregular LDPC Codes have an important impact in the coding theory since as it is stated in [32] they perform better than regular ones. Different types of irregular codes have been developed. They can be listed as follow: Modified Array Codes, Poisson, Sub-Poisson, Moderately Super-Poisson, Very Super-Poisson, Fast encoding versions. Irregular LDPC codes can be parameterized by the degree polynomials $\lambda(x)$ and $\rho(x)$, which can be defined as

$$\lambda(x) = \sum_{i=2}^{d_l} \lambda_i x^{i-1} \tag{3.2}$$

$$\rho(x) = \sum_{i=2}^{d_r} \rho_i x^{i-1} \tag{3.3}$$

where $\lambda_i(x)$ and $\rho_i(x)$ are the fractions of edges belonging to degree-i variable and check nodes, and d_l and d_r are the maximum variable and check node degrees respectively. The optimization of the $\lambda_i(x)$ and $\rho_i(x)$) is found by optimization algorithm.

3.3. Representations of LDPC codes

Basically there are two different possibilities to represent LDPC codes. Like all linear block codes they can be described via matrices. The second possibility is a graphical representation.

3.3.1. Matrix Representation

Each LDPC code is defined by a matrix H of size $(m-n)$, where n defines the code length and m defines the number of parity check bits in the code. The number of systematic bits would then be $k = n - m$. The parity check matrix can be represented in the form $H = [I_{n-k}$ $| \ P^T]$ where I_{n-k} is Identity matrix and P is the coefficient matrix. A sample (4×10) parity check matrix given in (3.4):

$$H = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix} \tag{3.4}$$

3.3.2. Graphical Representation of LDPC Codes

In coding theory, codes connected with graphs have been defined in a variety of ways. Tanner graph is the best way to represent the LDPC codes as it is simple, gives good information about parity check matrix, and it simplifies the explanation of decoding algorithm. Tanner graphs of LDPC codes are called bipartite graphs because they are represented mainly with two opposite nodes. One of them is called variable node which represents message node

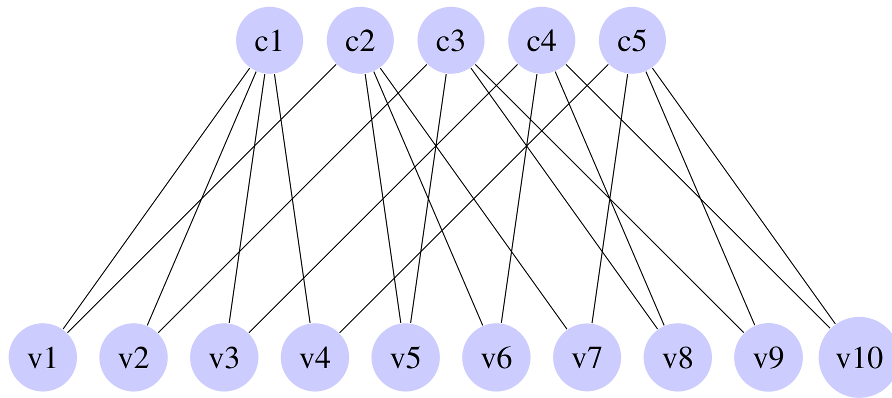


Figure 3.1: Tanner Graph of LDPC Code[8].

and the other one is called check nodes. Each variable node corresponds to a bit, and each parity-check node corresponds to parity check equations on the bits of the code word. The tanner graph representation of the LDPC codes is closely analogous to the more standard parity-check matrix representation of a code. The graph contains m check nodes (number of parity bits) and n variable nodes (number of bits in codeword). Check node c_i is connected to a variable node v_i if the element h_{ij} of H is ”1”. Parity-check matrices for the LDPC codes of DVB-T2 standard with code rates $R(1/4, 1/3, 2/5, 1/2, 3/5, 2/3, 3/4, 4/5, 5/6, 8/9, 9/10)$ are possible but in this work we have simulated the performances of H matrix supporting $R = 1/4$ and $R = 1/2$ code rates; detailed description of how the LDPC coding is done is given in [3]. The block length of the code is fixed to $16,200$ for the short FEC frame mode.

3.4. Quasi-cyclic LDPC codes

Different types of codes have the specifics how to design the respective parity-check matrix in order to perform near Shannon limit performance. Since the Quasi-Cyclic LDPC codes are used as an optional FEC scheme in IEEE 802.16e (WiMAX) in this section showing how to construct them is really important.

3.4.1. Constructing Quasi-cyclic codes

In constructing the H matrix for Low-density Parity-Check codes a couple of things have to be kept in mind. As it is stated in [33], an LDPC code has to be defined as the null space of a sparse parity-check matrix H over Galois Field GF(q) with the following properties:

1. each row must have constant weight λ

2. each column must have constant weight γ

3. two rows or two columns must not have more than one element in common.

The parity-check matrix obsessing the above properties is called a $(\gamma,\lambda)-regular$ Low-density Parity-check code. The third property restricts and makes sure that the Tanner graph of the H matrix is free of cycles of length four. As it is stated in [34], the minimum distance of the code will be greater or qual to $\gamma+1$. Regarding to a Quasi-cyclic LDPC code the matrix H is given by the null space of an matrix of sparse circulants [35]. Obviously the performance of an LDPC coding depends on the minimum distance of H matrix. Other important factors shaping the performance are related to the structural properties of the parity-check matrix. The common and important one is the so called girth of the code. As it is defined in [34], *"the girth is the length of the shortest cycles in the code's Tanner graph"*. Short cycles are not desired in coding theory and they should be avoided since they are going to affect decoding performance. The shortest cycle length that mostly affects performance is the magic number "4". Almost in all the methods available for constructing LDPC codes the girth "4" has a crucial impact in degrading the performance and should be eliminated. As it is stated in [30] and [36] a girth of length six can approach the performance near the Shannon limit. Settling the length of the girth limit to six, we have to keep in mind the minimum distance. A code with a girth greater than six does not necessarily perform well if the minimum distance is relatively small. Relatively small minimum distance causes the output of decoding to suffer from high error floor. Now that we settled down the required properties for a H matrix to perform near Shannon limit we are almost ready to start designing it. The so called base matrix can be constructed by different methods. Herein we are going to consider a general method for constructing a q-ray QC-LDPC.

Consider α to be a primitive element of GF(q) field. Lets represent the base matrix $H_b(m \times n)$ over GF(q) such as:

$$H_b = \begin{bmatrix} P_0 \\ P_1 \\ P_2 \\ \vdots \\ P_{m-1} \end{bmatrix} = \begin{bmatrix} P_{0,0} & P_{0,1} & P_{0,2} & \cdots & P_{0,n-2} & P_{0,n-1} \\ P_{1,0} & P_{1,1} & P_{1,2} & \cdots & P_{1,n-2} & P_{1,n-1} \\ P_{2,0} & P_{2,1} & P_{2,2} & \cdots & P_{2,n-2} & P_{2,n-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ P_{m-1,0} & P_{m-1,1} & P_{m-1,2} & \cdots & P_{m-1,n-2} & P_{m-1,n-1} \end{bmatrix} \tag{3.5}$$

As it is stated in [37], the matrix defined above should have the following structural properties:

1. for $0 \le i < m$ and $0 \le k,\ l < q-1$ and $k \neq l$, $\alpha^k w_i$ and $\alpha^l w_i$ should have at most one place where they have equal element in GF(q).

2. for $0 \le i,\ j < m$, $i \neq j$ and $0 \le k,\ l < q-1$, $\alpha^k w_i$ and $\alpha^l w_i$ are different in at least $n-1$ locations.

Property number one can be translated such that each row of matrix H_b has at most one 0 element. Property number two can be translated such that any two rows in matrix H_b has at most one place where they both have the same element. As it is stated in [37] these two properties are called $\alpha - multiplied\ row - constraints$. The matrix H_{bi} with size $((q-1) \times n)$

over GF(q) field for a particular interval $0 \leq i < m$ can be represented as follows:

$$H_{bi} = \begin{bmatrix} P_i \\ \alpha P_i \\ \vdots \\ \alpha^{q-2} P_i \end{bmatrix} = \begin{bmatrix} P_{i,0} & P_{i,1} & \cdots & P_{i,n-2} & P_{i,n-1} \\ \alpha P_{i,0} & \alpha P_{i,1} & \cdots & \alpha P_{i,n-2} & \alpha P_{i,n-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \alpha^{q-2} P_{i,0} & \alpha^{q-2} P_{i,1} & \cdots & \alpha^{q-2} P_{i,n-2} & \alpha^{q-2} P_{i,n-1} \end{bmatrix} \quad (3.6)$$

From the matrix above, similar properties can be noticed. Any two different rows of H_{bi} matrix are different in at least $n-1$ places. The matrix H_{bi} is simply obtained by expanding the ith row P_i of H_b $(q-1)$ times. Each of the respective entries of H_{bi} matrix can be replaced by its q-array and we can produce a sub matrix Q_i with a given size $(q-1) \times n(q-1)$ over GF(q) field. Any component $P_{i,j} \neq 0$ is replaced by $Q_{i,j}$ submatrix which is a circulant permutation matrix of size $(q-1) \times (q-1)$, otherwise it will be a $(q-1) \times (q-1)$ zero matrix.

$$H = \begin{bmatrix} Q_0 \\ Q_1 \\ \vdots \\ Q_{m-1} \end{bmatrix} = \begin{bmatrix} Q_{0,0} & Q_{0,1} & \cdots & Q_{0,n-2} & Q_{0,n-1} \\ Q_{1,0} & Q_{1,1} & \cdots & Q_{1,n-2} & Q_{1,n-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ Q_{m-1,0} & Q_{m-2,1} & \cdots & Q_{m-1,n-2} & Q_{m-1,n-1} \end{bmatrix} \quad (3.7)$$

Defining k to be the length of input message, n to be the length of total encoded message, the so called code rate R is given by (3.8):

$$R = \frac{k}{n} \quad (3.8)$$

Given a matrix H with the dimension $(n \times k)$, each column is a representative of a single bit in the codeword. On the other hand each respective row of the matrix represents the so called parity check codes.

3.4.2. Features of Quasi-Cyclic Codes

QC LDPC codes have many advantages over other types of linear LDPC codes. In term of encoding they are easier to be implemented using shift-registers in linear time [38]. Looking at the structure feature of QC LDPC, we can easily see that the parity-check matrix consists of circular right shifts submatrices which in WiMAX, those submatrices are identity matrices [39], [40]. Usually permutation vectors are used to create circulant matrices.

3.5. Encoding

Similar to all other linear block codes, we have the relation given by the following equation:

$$C_{(1\times n)}H^T_{(n\times m)} = 0 \tag{3.9}$$

where C is a codeword matrix, and H is a parity check matrix. In a systematic form, C can be written as:

$$C_{(1\times n)} = \begin{bmatrix} m_{(1\times n)} & P_{(1\times n-m)} \end{bmatrix} \tag{3.10}$$

where $P_{(1\times n-m)}$ denotes the parity portion and $m_{(1\times n)}$ denotes the message portion respectively.

$$CH^T = \begin{bmatrix} m & p \end{bmatrix} \begin{bmatrix} H_1^T \\ H_2^T \end{bmatrix} = mH_1^T + pH_2^T = 0 \tag{3.11}$$

or

$$p = mH_1^T + (H_2^T)^{-1} \tag{3.12}$$

The task of the encoder is then to compute the parity matrix P that can be directly appended to the message to produce the codeword. For the matrix H to be more manageable, the LU decomposition method can be preferably applied; i.e. [H]=[L][U]

$$\begin{bmatrix} l_{1,1} & l_{1,2} & \cdots & l_{1,n} \\ l_{2,1} & l_{2,2} & \cdots & l_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ l_{m,1} & l_{m,2} & \cdots & l_{m,n} \end{bmatrix} \begin{bmatrix} u_{1,1} & u_{1,2} & \cdots & u_{1,n} \\ u_{2,1} & u_{2,2} & \cdots & u_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ u_{m,1} & u_{m,2} & \cdots & u_{m,n} \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ \vdots \\ p_n \end{bmatrix} = \begin{bmatrix} m_1 \\ m_2 \\ \vdots \\ m_n \end{bmatrix} \tag{3.13}$$

Representing the matrix $[Y]$ such as $[Y]$=$[U][P]$, we can use forward substitution to solve $[L][Y]$=$[M]$

$$\begin{bmatrix} l_{1,1} & l_{1,2} & \cdots & l_{1,n} \\ l_{2,1} & l_{2,2} & \cdots & l_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ l_{m,1} & l_{m,2} & \cdots & l_{m,n} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} m_1 \\ m_2 \\ \vdots \\ m_n \end{bmatrix} \tag{3.14}$$

Finally the backward substitution is used to solve for the matrix P given the relation $[U][P]$=$[Y]$. From there we can easy figure out and calculate the $\{p_i\}$ as required.

$$\begin{bmatrix} u_{1,1} & u_{1,2} & \cdots & u_{1,n} \\ u_{2,1} & u_{2,2} & \cdots & u_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ u_{m,1} & u_{m,2} & \cdots & u_{m,n} \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ \vdots \\ p_n \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \tag{3.15}$$

3.6. LDPC-IRA Codes

The second generation Digital Video Broadcasting satellite has adopted recently a special class of LDPC codes. They are so called Irregular Repeat- Accumulate (IRA), having linear decoding complexity [45]. The parity check matrix H for this class of special codes can be represented in the form: $H_{(n-k)\times n} = \left[A_{(n-k)\times k} | B_{(n-k)\times(n-k)}\right]$

$$H_{(n-k)\times n} = \left[\begin{array}{ccccc|cccccc} a_{0,0} & a_{0,1} & \cdots & a_{0,k-2} & a_{0,k-1} & 1 & 0 & \cdots & \cdots & \cdots & 0 \\ a_{1,0} & a_{1,1} & \cdots & a_{1,k-2} & a_{1,k-1} & 1 & 1 & 0 & & & \vdots \\ \vdots & & & & \vdots & 0 & 1 & 1 & \ddots & & \vdots \\ \vdots & & & & \vdots & \vdots & \ddots & \ddots & \ddots & 0 & \vdots \\ a_{n-k-2,0} & a_{n-k-2,1} & \cdots & a_{n-k-2,k-2} & a_{n-k-2,k-1} & \vdots & & \ddots & 1 & 1 & 0 \\ a_{n-k-1,0} & a_{n-k-1,1} & \cdots & a_{n-k-1,k-2} & a_{n-k-1,k-1} & 0 & \cdots & \cdots & 0 & 1 & 1 \end{array}\right] \quad (3.16)$$

where A is a sparse matrix and B is a staircase lower triangular matrix [45]. The codewords generated in DVB-S2 standard are a result of concatenation of parity bits $p = (p_0, p_1, ..., p_{n-k-1})$ and information bits $i = (i_0, i_1, ..., i_{k-1})$. The information bits have been associated to matrix A and the parity check bits to the matrix B.

As it is stated in [47], parity check bits can be obtained form the matrix A in the following manner:

$$\begin{aligned} p_0 &= a_{0,0}i_0 \oplus a_0 i_1 \oplus \cdots \oplus a_{0,k-1}i_{k-1} \\ p_1 &= a_{1,0}i_0 \oplus a_{1,1}i_1 \oplus \cdots \oplus a_{1,k-1}i_{k-1} \oplus p_0 \\ &\vdots \\ p_{n-k-1} &= a_{n-k-1,0}i_0 \oplus a_{n-k-1,1}i_1 \oplus a_{n-k-1,1}i_1 \oplus \cdots \oplus a_{n-k-1,0}i_{k-1} \oplus p_{n-k-2} \end{aligned} \quad (3.17)$$

3.7. Decoding LDPC codes

The algorithm used to decode LDPC codes was discovered independently several times so as a matter of fact there are several methods used in decoding LDPC codes. The most commons one are Believe Propagation algorithm (BPA), the message passing algorithm (MPA) and the Sum-Product algorithm (SPA).

The Tanner graph shown in Figure 3.1 can be easily drawn from the matrix H given in (3.4) as shown in this section. The tanner graph contains m check nodes (number of parity bits) labeled with 'c' and n variable nodes (number of bits in a codeword) labeled with 'v'. Check node c_i is connected to a variable node v_i if the element h_{ij} of H is "1". In the Log domain, the binary message passes between check nodes and variable nodes. In each pass the log likelihood ratio (LLR) is recorded to figure out the probability of its likely symbol. As it is stated in [27], generally the decoder goes through this typically steps:

Step1:

Compute the initial value of LLR transmitted from the variable node v_i to check node c_i; for all i; $1 \leq i \leq n$.

$$L(q_{ij}) = L(c_i) = \frac{2y_i}{\sigma^2} = LLR_i = \log \frac{P(c_{i=0}|y_i)}{P(c_{i=1}|y_i)} \tag{3.17}$$

where $L(c_i)$ denotes log likelihood ratio (LLR), σ^2 denotes the channel noise variance, $P(c_{i=0}|y_i)$ denotes probability value for given input y_i.

Step2:

Compute $L(r_{ij})$ transmitted from the check node c_i to variable node v_i for all i; $1 \leq i \leq n$. Denote $\phi(x) = \log\left(\frac{e^x+1}{e^x-1}\right)$.

$$L(r_{ij}) = \prod_{i' \in V_j/i} \alpha_{i'j}\, \phi\left(\sum_{i' \in V_j/i} \phi\left(\beta_{i'j}\right)\right) \tag{3.18}$$

where $\alpha_{i'j} = sgn\left\{L\left(q_{ij}\right)\right\}$, and $\beta_{ij} = \left|L\left(q_{ij}\right)\right|$.

Step3:

After obtaining $L\left(q_{ij}\right)$ it is necessary to modify it so that we can use it as data transmitted from the variable node v_i to check node c_i for all i; $1 \leq i \leq n$.

$$L\left(q_{ij}\right) = L(c_i) + \sum_{j' \in C_i/j} L\left(r_{j'i}\right) \tag{3.19}$$

Step4:

The soft output can be represented such as:

$$L(Q_i) = L(c_i) + \sum_{j \in C_i} L\left(r_{ji}\right) \tag{3.20}$$

<u>Step5:</u>

Now that we have already obtained the soft output it can be used to figure out the hard decision output which is given by the following equation:

$$\hat{c}w_i = 1 \text{ if } L(Q_i) < 0, \text{ otherwise } \hat{c}w_i = 0$$

Chapter 4

DIGITAL VIDEO BROADCASTING and IEEE 802.16e

The Digital Video Broadcasting (DVB) specifications cover digital services delivered via cable, satellite and terrestrial transmitters, as well as by the internet and mobile communication systems. Digital Video Broadcasting (DVB) is playing a crucial role in digital television and data broadcasting world-wide. DVB services have recently been introduced in Europe, in North- and South America, in Asia, Africa and Australia. Among the more recent achievements are the standard for terrestrial transmission, for microwave distribution and for interactive services via PSTN/ISDN and via (coaxial) cable [26]. As it is stated by the standard in [22]techniques used by DVB are able to deliver data at approximately 38 Mbit/s within one satellite or cable channel or at 24 Mbit/s within one terrestrial channel. The satellite member of the DVB family, DVB-S, is defined in European Standard EN 300 421 [18]. September 1993, and at the end of the same year produced its first specification, DVB-S [20], the satellite delivery specification now used by most satellite broadcasters around the world for DTH (direct-to-home) television services. The DVB-S system is based on QPSK modulation and convolutional forward error correction (FEC), concatenated with Reed-Solomon coding. In 1998, DVB produced its second standard for satellite applications, DVB-DSNG [21], extending the functionalities of DVB-S to include higher order modulations (8PSK and 16QAM) for DSNG and other TV contribution applications by satellite.

In the last decade, studies in the field of digital communications and, in particular, of error correcting techniques suitable for recursive decoding, have brought new impulse to the technology innovations. The results of this evolutionary trend, together with the increase in the operators' and consumers' demand for larger capacity and innovative services by satellite, led DVB to define in 2003 the second-generation system for satellite broad-band services, DVB-S2 [22], now recognized as ITU-R and European Telecommunications Standards Institute (ETSI) standards.

4.1. Second Generation Digital Video Broadcasting Over Satellite (DVB-S2)

Digital satellite transmission technology has evolved considerably since the publication of the original DVB-S specification. New coding and modulation schemes permit greater flexibility and more efficient use of capacity, and additional data formats can now be handled without significant increase of system complexity. DVB-S2 has a range of constellations on offer. DVB-S2 supports a wide range of modulation schemes, including QPSK ($2bits/symbol$), 8PSK ($3bits/symbol$), 16APSK ($4bits/symbol$) and 32APSK ($5bits/symbol$). These APSK modulation schemes provide superior compensation for transponder non-linearities than QAM. DVB-S2 is so flexible that it can cope with any existing satellite transponder characteristics, with a large variety of spectrum efficiencies and associated SNR requirements. Furthermore it is designed to handle a variety of advanced audiovideo formats which the DVB Project is currently defining [23].

4.1.1. The FEC Scheme

The FEC, together with the modulation, is the key subsystem to achieve excellent performance by satellite, in the presence of high levels of noise and interference. The DVB-S2 FEC selection process, based on computer simulations, compared seven proposals over the AWGN channel's parallel or serially concatenated convolutional codes, product codes, low density parity check codes (LDPC)"all using " turbo (i.e., recursive) decoding techniques. The winning system was based on LDPC codes, and offered the minimum distance from the Shannon limit in the linear AWGN channel, under the constraint of maximum decoder complexity of $14mm$ of silicon ($0.13-m$ technology).

At the heart of the DVB-S2 system is the LDPC, BCH FEC engine. DVB-S2 allows for two different LDPC block sizes - a short 16*k* block or the normal 64*k* block. Systems using the 16k short block codes are expected to perform 0.2 to 0.3 dB worse than those employing the normal 64*k* block codes. The output of the FEC engine is an FECFRAME. The FECFRAME is always of constant length, either a 16*k* or 64*k* block depending on the choice of a normal or short FEC system. The amount of real data carried by each FECFRAME is dependent upon how much overhead the chosen FEC code uses. The FEC rates defined for use within DVB-S2 are shown in Table 4.1 along with the modulation formats for which they are valid.

Table 4.1: FEC Rates Applicable to the Various Modulation Formats [22].

FEC	QPSK	8PSK	16APSK	32APSK
1/4	✓	x	x	x
1/3	✓	x	x	x
2/5	✓	x	x	x
1/2	✓	x	x	x
3/5	✓	✓	x	x
2/3	✓	✓	✓	x
3/4	✓	✓	✓	✓
4/5	✓	x	✓	✓
5/6	✓	✓	✓	✓
8/9	✓	✓	✓	✓
9/10	✓	✓	✓	✓

The selected LDPC codes [17] use very large block lengths (64800 bits for applications not too critical for delays, and 16200 bits). Code rates of $R = (1/4, 1/3, 2/5, 1/2, 3/5, 2/3, 3/4, 4/5, 5/6, 8/9, 9$ are available, depending on the selected modulation and the system requirements. Coding rates $R = 1/4$, $R = 1/3$ and $R = 2/5$ have been introduced to operate, in combination with QPSK, under exceptionally poor link conditions, where the signal level is below the noise level. Concatenated BCH outer codes are introduced to avoid error floors at low bit error rates (BER).

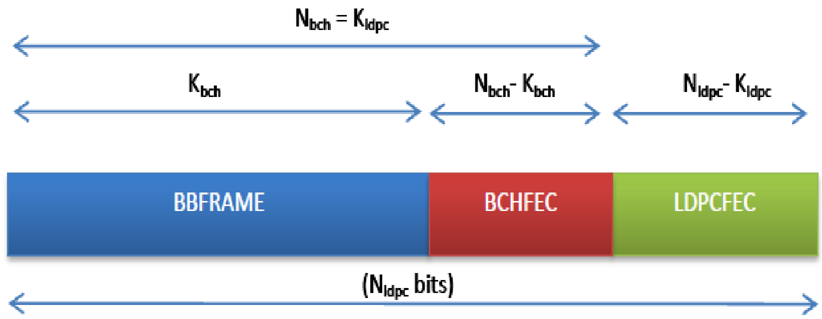


Figure 4.1: Format of data before bit interleaving[21].

4.1.2. Normal FEC Frame

The output of the FEC engine is an FECFRAME. The FECFRAME is always of constant length, either a 16k or 64k block depending on the choice of a normal or short FEC system.

Table 4.2: Coding Parameters for normal FECFRAME $N_{ldpc} = 64800$ bits

LDPC Code	BCH Uncoded Block K_{Bch}	BCH Coded Block N_{Bch}	BCH t-error Correction	$N_{bch} - K_{bch}$	LDPC Coded Block N_{ldpc}
1/2	32 208	32 400	12	192	64 800
3/5	38 688	38 800	12	192	64 800
2/3	43 040	43 200	10	160	64 800
3/4	48 408	48 600	12	192	64 800
4/5	51 648	51 840	12	192	64 800
5/6	53 840	54 000	10	160	64 800

Addresses of parity bit accumulators for code rate $R = 1/4$ and $n_{ldpc} = 64800$ bits are shown in (4.1) and (4.2).

$$c_1(t) = \begin{bmatrix} 23606 & 36098 & 1140 & 28859 & 18148 & 18510 & 6226 & 540 & 42014 & 20879 & 23802 & 47088 \\ 16419 & 24928 & 16609 & 17248 & 7693 & 24997 & 42587 & 16858 & 34921 & 21042 & 37024 & 20692 \\ 1874 & 40094 & 18704 & 14474 & 14004 & 11519 & 13106 & 28826 & 38669 & 22363 & 30255 & 31105 \\ 22254 & 40564 & 22645 & 22532 & 6134 & 9176 & 39998 & 23892 & 8937 & 15608 & 16854 & 31009 \\ 8037 & 40401 & 13550 & 19526 & 41902 & 28782 & 13304 & 32796 & 24679 & 27140 & 45980 & 10021 \\ 40540 & 44498 & 13911 & 22435 & 32701 & 18405 & 39929 & 25521 & 12497 & 9851 & 39223 & 34823 \\ 15233 & 45333 & 5041 & 44979 & 45710 & 42150 & 19416 & 1892 & 23121 & 15860 & 8832 & 10308 \\ 10468 & 44296 & 3611 & 1480 & 37581 & 32254 & 13817 & 6883 & 32892 & 40258 & 46538 & 11940 \\ 6705 & 21634 & 28150 & 43757 & 895 & 6547 & 20970 & 28914 & 30117 & 25736 & 41734 & 11392 \\ 22002 & 5739 & 27210 & 27828 & 34192 & 379924 & 10915 & 6998 & 3824 & 42130 & 4494 & 35739 \\ 8515 & 1191 & 13642 & 30950 & 25943 & 12673 & 16726 & 34261 & 31828 & 3340 & 8747 & 39225 \\ 18979 & 17058 & 43130 & 4246 & 4793 & 44030 & 19454 & 29511 & 47929 & 15174 & 24333 & 19354 \\ 16694 & 8381 & 29642 & 46516 & 32224 & 26344 & 9405 & 18292 & 12437 & 27316 & 35466 & 41992 \\ 15642 & 5871 & 46489 & 26723 & 23396 & 7257 & 8974 & 3156 & 37420 & 44823 & 35423 & 13541 \\ 42858 & 320008 & 41282 & 38773 & 26570 & 2702 & 27260 & 46974 & 1469 & 20887 & 27426 & 38553 \end{bmatrix} \tag{4.1}$$

$$c_2(t) = \begin{bmatrix} 22152 & 24261 & 8297 \\ 19347 & 9978 & 27802 \\ 34991 & 6354 & 33561 \\ 29782 & 30875 & 29523 \\ 9278 & 48512 & 14349 \\ 38061 & 4165 & 43878 \\ 8548 & 33172 & 34410 \\ 22535 & 28811 & 23950 \\ 20439 & 4027 & 24186 \\ 38618 & 8187 & 30947 \\ 35538 & 43880 & 21459 \\ 7091 & 45616 & 15063 \\ 5505 & 9315 & 21908 \\ 36046 & 32914 & 11836 \\ 16905 & 29962 & 12980 \\ \vdots & \vdots & \vdots \end{bmatrix} \begin{bmatrix} \vdots & \vdots & \vdots \\ 11171 & 23709 & 22460 \\ 34541 & 9937 & 44500 \\ 14035 & 47316 & 8815 \\ 15057 & 45482 & 24461 \\ 30518 & 36877 & 879 \\ 7583 & 13364 & 24332 \\ 448 & 27056 & 4682 \\ 12083 & 31378 & 21670 \\ 1159 & 18031 & 2221 \\ 17028 & 38715 & 9350 \\ 17343 & 24530 & 29574 \\ 46128 & 31039 & 32818 \\ 20373 & 36967 & 18345 \\ 46685 & 20622 & 32806 \end{bmatrix} \tag{4.2}$$

Addresses of parity bit accumulators for code rate $R = 1/3$ and $n_{ldpc} = 64800$ bits are shown in (4.3) and (4.4).

$$
c_1(t) = \begin{bmatrix}
34903 & 20927 & 32093 & 1052 & 25611 & 16093 & 16454 & 5520 & 506 & 37399 & 18518 & 21120 \\
16636 & 14594 & 22158 & 14763 & 15333 & 6838 & 22222 & 37856 & 14985 & 31041 & 18704 & 32910 \\
29235 & 19780 & 36056 & 20129 & 20029 & 5457 & 8157 & 35554 & 21237 & 7943 & 13873 & 14980 \\
9912 & 7143 & 35911 & 12043 & 17360 & 37253 & 25588 & 11827 & 29152 & 21936 & 24125 & 40870 \\
40701 & 36035 & 39556 & 12366 & 19946 & 29072 & 16365 & 35495 & 22686 & 11106 & 8756 & 34863 \\
19165 & 15702 & 13536 & 40238 & 4465 & 40034 & 40590 & 37540 & 17162 & 1712 & 20577 & 14138 \\
31338 & 19342 & 9301 & 39375 & 3211 & 1316 & 33409 & 28670 & 12282 & 6118 & 29236 & 35787 \\
11504 & 30506 & 19558 & 5100 & 24188 & 24738 & 30397 & 33775 & 9699 & 6215 & 3397 & 37451 \\
34689 & 23126 & 7571 & 1058 & 12127 & 27518 & 23064 & 11265 & 14867 & 30451 & 28289 & 2966 \\
11660 & 15334 & 16867 & 15160 & 38843 & 3778 & 4265 & 39139 & 17293 & 26229 & 42604 & 13486 \\
31497 & 1365 & 14828 & 7453 & 26350 & 41346 & 28643 & 23421 & 8354 & 16255 & 11055 & 24279 \\
15687 & 12467 & 13906 & 5215 & 41328 & 23755 & 20800 & 6447 & 7970 & 2803 & 33262 & 39843 \\
5363 & 22469 & 38091 & 28457 & 36696 & 34471 & 23619 & 2404 & 24229 & 41754 & 1297 & 18563 \\
3673 & 39070 & 14480 & 30279 & 37483 & 7580 & 29519 & 30519 & 39831 & 20252 & 18132 & 20010 \\
34386 & 7252 & 27526 & 12950 & 6875 & 43020 & 31566 & 39069 & 18985 & 15541 & 40020 & 16715 \\
1721 & 37332 & 39953 & 17430 & 32134 & 29162 & 10490 & 12971 & 28581 & 29331 & 6489 & 35383 \\
736 & 7022 & 42349 & 8783 & 6767 & 11871 & 21675 & 10325 & 11548 & 25978 & 431 & 24085 \\
1925 & 10602 & 28585 & 12170 & 15156 & 34404 & 8351 & 13273 & 20208 & 5800 & 15367 & 21764 \\
16279 & 37832 & 34792 & 21250 & 34192 & 7406 & 41488 & 18346 & 29227 & 26127 & 25493 & 7048
\end{bmatrix} \tag{4.3}
$$

$$c_2(t) = \begin{bmatrix} 39948 & 28229 & 24899 \\ 17408 & 14274 & 38993 \\ 38774 & 15968 & 28459 \\ 41404 & 27249 & 27425 \\ 41229 & 6082 & 43114 \\ 13957 & 4979 & 40654 \\ 3093 & 3438 & 34992 \\ 34082 & 6172 & 28760 \\ 42210 & 34141 & 41021 \\ 14705 & 17783 & 10134 \\ 41755 & 39884 & 22773 \\ 14615 & 15593 & 1642 \\ 29111 & 37061 & 39860 \\ 9579 & 33552 & 633 \\ 12951 & 21137 & 39608 \\ 38244 & 27361 & 29417 \\ 2939 & 10172 & 36479 \\ 29094 & 5357 & 19224 \\ 9562 & 24436 & 28637 \\ 40177 & 2326 & 13504 \\ 6834 & 21583 & 42516 \\ 40651 & 42810 & 25709 \\ 31557 & 32138 & 38142 \\ 18624 & 41867 & 39296 \\ 37560 & 14295 & 16245 \\ 6821 & 21679 & 31570 \\ 25339 & 25083 & 22081 \\ 8047 & 697 & 35268 \\ 9884 & 17073 & 19995 \\ 26848 & 35245 & 8390 \\ 18658 & 16134 & 14807 \\ 12201 & 32944 & 5035 \\ 25236 & 1216 & 38986 \\ 42994 & 24782 & 8681 \\ 28321 & 4932 & 34249 \\ 4107 & 29382 & 32124 \\ 22157 & 2624 & 14468 \\ 38788 & 27081 & 7936 \\ 4368 & 26148 & 10578 \\ 25353 & 4122 & 39751 \end{bmatrix} \tag{4.4}$$

Addresses of parity bit accumulators for rate $R = 1/2$ and $n_{ldpc} = 64800$ bits are shown in (4.5) and (4.6).

$$
c_1(t) = \begin{bmatrix}
54 & 9318 & 14392 & 27561 & 26909 & 10219 & 2534 & 8597 \\
55 & 7263 & 4635 & 2530 & 28130 & 3033 & 23830 & 3651 \\
56 & 24731 & 23583 & 26036 & 17299 & 5750 & 792 & 9169 \\
57 & 5811 & 26154 & 18653 & 11551 & 15447 & 13685 & 16264 \\
58 & 12610 & 11347 & 28768 & 2792 & 3174 & 29371 & 12997 \\
59 & 16789 & 16018 & 21449 & 6165 & 21202 & 15850 & 3186 \\
60 & 31016 & 21449 & 17618 & 6213 & 12166 & 8334 & 18212 \\
61 & 22836 & 14213 & 11327 & 5896 & 718 & 11727 & 9308 \\
62 & 2091 & 24941 & 29966 & 23634 & 9013 & 15587 & 5444 \\
63 & 22207 & 3983 & 16904 & 28534 & 21415 & 27524 & 25912 \\
64 & 25687 & 4501 & 22193 & 14665 & 14798 & 16158 & 5491 \\
65 & 4520 & 17094 & 23397 & 4264 & 22370 & 16941 & 21526 \\
66 & 10490 & 6182 & 32370 & 9597 & 30841 & 25954 & 2762 \\
67 & 22120 & 22865 & 29870 & 15147 & 13668 & 14955 & 19235 \\
68 & 6689 & 18408 & 18346 & 9918 & 25746 & 5443 & 20645 \\
69 & 29982 & 12529 & 13858 & 4746 & 30370 & 10023 & 24828 \\
70 & 1262 & 28032 & 29888 & 13063 & 24033 & 21951 & 7863 \\
71 & 6594 & 29642 & 31451 & 14831 & 9509 & 9335 & 31552 \\
72 & 1358 & 6454 & 16633 & 20354 & 24598 & 624 & 5265 \\
73 & 19529 & 295 & 18011 & 3080 & 13364 & 8032 & 15323 \\
74 & 11981 & 1510 & 7960 & 21462 & 9129 & 11370 & 25741 \\
75 & 9276 & 29656 & 4543 & 30699 & 20646 & 21921 & 28050 \\
76 & 15975 & 25634 & 5520 & 31119 & 13715 & 21949 & 19605 \\
77 & 18688 & 4608 & 31755 & 30165 & 13103 & 10706 & 29224 \\
78 & 21514 & 23117 & 12245 & 26035 & 31656 & 25631 & 30699 \\
79 & 9674 & 24966 & 31285 & 29908 & 17042 & 24588 & 31857 \\
80 & 21856 & 27777 & 29919 & 27000 & 14897 & 11409 & 7122 \\
81 & 29773 & 23310 & 263 & 4877 & 28622 & 20545 & 22092 \\
82 & 15605 & 5651 & 21864 & 3967 & 14419 & 22757 & 15896 \\
83 & 30145 & 1759 & 10139 & 29223 & 26086 & 10556 & 5098 \\
84 & 18815 & 16575 & 2936 & 24457 & 26738 & 6030 & 505 \\
85 & 30326 & 22298 & 27562 & 20131 & 26390 & 6247 & 24791 \\
86 & 928 & 29246 & 21246 & 12400 & 15311 & 32309 & 18608 \\
87 & 20314 & 6025 & 26689 & 16302 & 2296 & 3244 & 19613 \\
88 & 6237 & 11943 & 22851 & 15642 & 23857 & 15112 & 20947 \\
89 & 26403 & 25168 & 19038 & 18384 & 8882 & 12719 & 7093
\end{bmatrix} \tag{4.5}
$$

$$
c_2(t) = \begin{bmatrix} 0 & 14567 & 24965 \\ 1 & 3908 & 100 \\ 2 & 10279 & 240 \\ 3 & 24102 & 764 \\ 4 & 12383 & 4173 \\ 5 & 13861 & 15918 \\ 6 & 21327 & 1046 \\ 7 & 5288 & 14579 \\ 8 & 28158 & 8069 \\ 9 & 16583 & 11098 \\ 10 & 16681 & 28363 \\ 11 & 13980 & 24725 \\ 12 & 32169 & 17989 \\ 13 & 10907 & 2767 \\ 14 & 21557 & 3818 \\ 15 & 26676 & 12422 \\ 16 & 7676 & 8754 \\ 17 & 14905 & 20232 \\ 18 & 15719 & 28646 \\ 19 & 31942 & 8589 \\ 20 & 19978 & 27197 \\ 21 & 27060 & 15071 \\ 22 & 6071 & 26649 \\ 23 & 10393 & 11176 \\ 24 & 9597 & 13370 \\ 25 & 7081 & 17677 \\ \vdots & \vdots & \vdots \end{bmatrix} \begin{bmatrix} \vdots & \vdots & \vdots \\ 26 & 1433 & 19513 \\ 27 & 26925 & 9014 \\ 28 & 19202 & 8900 \\ 29 & 18152 & 30647 \\ 30 & 20803 & 1737 \\ 31 & 11804 & 25221 \\ 32 & 31683 & 17783 \\ 33 & 29694 & 9345 \\ 34 & 12280 & 26611 \\ 35 & 6526 & 26122 \\ 36 & 26165 & 11241 \\ 37 & 7666 & 26962 \\ 38 & 16290 & 8480 \\ 39 & 11774 & 10120 \\ 40 & 30051 & 30426 \\ 41 & 1335 & 15424 \\ 42 & 6865 & 17742 \\ 43 & 31779 & 12489 \\ 44 & 32120 & 21001 \\ 45 & 14508 & 6996 \\ 46 & 979 & 25024 \\ 47 & 4554 & 21896 \\ 48 & 7989 & 21777 \\ 49 & 4972 & 20661 \\ 50 & 6612 & 2730 \\ 51 & 12742 & 4418 \\ 52 & 29194 & 595 \\ 53 & 19267 & 20113 \end{bmatrix} \tag{4.6}
$$

4.1.3. Shortened FEC Frame

Table 4.3: Coding Parameters for shortened FECFRAME $N_{ldpc} = 16200$ bits

LDPC Code	BCH Uncoded Block K_{Bch}	BCH Coded Block N_{Bch}	BCH t-error Correction	$N_{bch} - K_{bch}$	Effective LDPC Rate	LDPC Coded Block N_{ldpc}
1/4	3 072	32 40	12	168	1/5	16 200
1/2	7 032	7 200	12	168	4/9	16 200
3/5	9 552	9 720	12	168	3/5	16 200
2/3	10 632	10 800	12	168	2/3	16 200
3/4	11 712	11 880	12	168	11/15	16 200
4/5	12 432	12 600	12	168	7/9	16 200
5/6	13 152	13 320	12	168	37/45	16 200

Addresses of parity bit accumulators for code rate $R = 1/4$ and $n_{ldpc} = 16200$ bits are given in (4.7) and (4.8).

$$c_1(t) = \begin{bmatrix} 6295 & 9626 & 304 & 7695 & 4839 & 4936 & 1660 & 144 & 11203 & 5567 & 6347 & 12557 \\ 10691 & 4988 & 3859 & 3734 & 3071 & 3494 & 7687 & 10313 & 5964 & 8069 & 8296 & 11090 \\ 10774 & 3613 & 5208 & 11177 & 7676 & 3549 & 8746 & 6583 & 7239 & 12265 & 2674 & 4292 \\ 11869 & 3708 & 5981 & 8718 & 4908 & 10650 & 6805 & 3334 & 2627 & 10461 & 9285 & 11120 \end{bmatrix} \tag{4.7}$$

$$c_2(t) = \begin{bmatrix} 7844 & 3079 & 10733 \\ 3385 & 10854 & 5747 \\ 1360 & 12010 & 12202 \\ 6189 & 4241 & 2343 \\ 9840 & 12726 & 4977 \end{bmatrix} \tag{4.8}$$

Addresses of parity bit accumulators for code rate $R = 1/3$ and $n_{ldpc} = 16200$ bits are shown in (4.9) and (4.10).

$$c_1(t) = \begin{bmatrix} 416 & 8909 & 4156 & 3216 & 3112 & 2560 & 2912 & 6405 & 8593 & 4969 & 6723 & 6912 \\ 8978 & 3011 & 4339 & 9312 & 6396 & 2957 & 7288 & 5485 & 6031 & 10218 & 2226 & 3575 \\ 3383 & 10059 & 1114 & 10008 & 10147 & 9384 & 4290 & 434 & 5139 & 3536 & 1965 & 2291 \\ 2797 & 3693 & 7615 & 7077 & 743 & 1941 & 8716 & 6215 & 3840 & 5140 & 4582 & 5420 \\ 6110 & 8551 & 1515 & 7404 & 4879 & 4946 & 5383 & 1831 & 3441 & 9569 & 10472 & 4306 \end{bmatrix} \tag{4.9}$$

$$c_2(t) = \begin{bmatrix} 1505 & 5682 & 7778 \\ 7172 & 6830 & 6623 \\ 7281 & 3941 & 3505 \\ 10270 & 8669 & 914 \\ 3622 & 7563 & 9388 \\ 9930 & 5058 & 4554 \\ 4844 & 9609 & 2707 \\ 6883 & 3237 & 1714 \\ 4768 & 3878 & 10017 \\ 10127 & 3334 & 8267 \end{bmatrix} \tag{4.10}$$

Addresses of parity bit accumulators for code rate $R = 1/2$ and $n_{ldpc} = 16200$ bits are shown in (4.11) and (4.12).

$$c_1(t) = \begin{bmatrix} 20 & 712 & 2386 & 6354 & 4061 & 1062 & 5045 & 5158 \\ 21 & 2543 & 5748 & 4822 & 2348 & 3089 & 6328 & 5876 \\ 22 & 926 & 5701 & 269 & 3693 & 2438 & 3190 & \\ 23 & 2802 & 4520 & 3577 & 5324 & 1091 & 4667 & 4449 \\ 24 & 5140 & 2003 & 1263 & 4742 & 6497 & 1185 & 6202 \end{bmatrix} \tag{4.11}$$

$$c_2(t) = \begin{bmatrix} 0 & 4046 & 6934 \\ 1 & 2855 & 66 \\ 2 & 6694 & 212 \\ 3 & 3439 & 1158 \\ 4 & 3850 & 4422 \\ 5 & 5924 & 290 \\ 6 & 1467 & 4049 \\ 7 & 7820 & 2242 \\ 8 & 4606 & 3080 \\ 9 & 4633 & 7877 \\ 10 & 3884 & 6868 \\ 11 & 8935 & 4996 \\ 12 & 3028 & 764 \\ 13 & 5988 & 1057 \\ 14 & 7411 & \end{bmatrix} \tag{4.12}$$

Addresses of parity bit accumulators for code rate $R = 2/3$ and $n_{ldpc} = 16200$ bits are shown in (4.13) and (4.14).

$$c_1(t) = \begin{bmatrix} 0 & 2084 & 1613 & 1548 & 1286 & 1460 & 3196 & 4297 & 2481 & 3369 & 3451 & 4620 & 2622 \\ 1 & 122 & 1516 & 3448 & 2880 & 1407 & 1847 & 3799 & 3529 & 373 & 971 & 4358 & 3108 \\ 2 & 259 & 3399 & 929 & 2650 & 864 & 3996 & 3833 & 107 & 5287 & 164 & 3125 & 2350 \end{bmatrix} \tag{4.13}$$

$$c_2(t) = \begin{bmatrix} 3 & 342 & 3529 \\ 4 & 4198 & 2147 \\ 5 & 1880 & 4836 \\ 6 & 3864 & 4910 \\ 7 & 243 & 1542 \\ 8 & 3011 & 1436 \\ 9 & 2167 & 2512 \\ 10 & 4606 & 1003 \\ 11 & 2835 & 705 \\ 12 & 3426 & 2365 \\ 13 & 3848 & 2474 \\ 14 & 1360 & 1743 \\ 0 & 163 & 2536 \\ 1 & 2583 & 1180 \\ \vdots & \vdots & \vdots \end{bmatrix} \begin{bmatrix} \vdots & \vdots & \vdots \\ 2 & 1542 & 509 \\ 3 & 4418 & 1005 \\ 4 & 5212 & 5117 \\ 5 & 2155 & 2922 \\ 6 & 347 & 2696 \\ 7 & 226 & 4296 \\ 8 & 1560 & 487 \\ 9 & 3926 & 1640 \\ 10 & 149 & 2928 \\ 11 & 2364 & 563 \\ 12 & 635 & 688 \\ 13 & 231 & 1684 \\ 14 & 1129 & 3894 \end{bmatrix} \tag{4.14}$$

4.2. Second Generation Terrestrial Digital Video Broadcasting (DVB-T2)

The DVB-T standard is the most successful digital terrestrial television standards in the world. First published in 1995, it has been adopted by more than half of all countries in the world.

Since the publication of the DVB-T standard, however, research in transmission technology has continued, and new options for modulating and error-protecting broadcast steams have been developed. Simultaneously, the demand for broadcasting frequency spectrum has increased as has the pressure to release broadcast spectrum for non-broadcast applications, making it is ever more necessary to maximize spectrum efficiency. In response, the DVB Project has developed the second-generation digital terrestrial television (DVB-T2) standard. The specification, first published by the DVB Project in June 2008, has been standardized by European Telecommunication Standardizations Institute (ETSI) since September 2009. Implementation and product development using this new standard has already begun. In comparison with the current digital terrestrial television standard, DVB-T, the second-generation standard, DVB-T2, provides a minimum increase in capacity of at least 30 % in equivalent reception conditions using existing receiving antennas. Two excellent documents, the DVB-T2 specification (ETSI EN302755) and the Implementation Guidelines (DVB Bluebook A133), are available with the details of the technology. Like the DVB-S2 standard, the

Table 4.4: Example of MFN mode in the United Kingdom [21]

	Current Uk DVB-T mode	Selected DVB-T2 mode
Modulation	64 QAM	256 QAM
FFT size	2K	32K
Guard Interval	1/32	1/128
FEC	2/3 CC+RS	2/3 LDPC+BCH

DVB-T2 specification makes use of LDPC (Lowdensity parity-check) codes in combination with BCH (Bose-Chaudhuri- Hocquengham) to protect against high noise levels and interference. In comparison, the DVB-T standard, which makes use of convolutional coding and Reed-Solomon, two further code rates have been added. Compared with the DVB-T standard, the DVB-T2 specification allows for a reduction in the peak to average power used in the transmitter station. The peak amplifier power rating can be reduced by 25% which can significantly reduce the total amount of power that must be made available for the functionality of high power transmission stations.

4.2.1. Outer encoding (BCH)

BCH (Bose-Chaudhuri-Hocquenghem) codes form a large class of multiple random error-correcting codes. They were first discovered by A. Hocquenghem in 1959 and independently by R. C. Bose and D. K. Ray-Chaudhuri in 1960 [16]. BCH codes are classified as cyclic codes. However at that time just the codes were invented, the decoding algorithm were not discovered yet. The first decoding algorithm for binary BCH codes was discovered by Peterson in 1960. Since then, many coding theorist have tried to refine it.

4.2.2. Binary Primitive BCH codes

A binary primitive BCH code is a BCH code defined using a primitive element α. Taking α to be a primitive element of $GF(2^m)$, then the block length is $n = 2^m - 1$. The parity check matrix for a t-error-correcting primitive narrow-sense BCH code is

$$\begin{bmatrix} 1 & \alpha & \alpha^2 & \cdots & \alpha^{(n-1)} \\ 1 & \alpha^2 & \alpha^4 & \cdots & \alpha^{2(n-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \alpha^{2t} & \alpha^{4t} & \cdots & \alpha^{2t(n-1)} \end{bmatrix} \tag{4.15}$$

For any integer $m \geq 3$ and $t < 2^{m-1}$ there exists a primitive BCH code with the following parameters: $n = 2^{m-1}$, $n - k \leq mt$, $d_{min} \geq 2t + 1$. The generator polynomial g(x) of this codes is specified in terms of its roots from the Galois Field $GF(2^m)$ is the lowest degree polynomial over GF(2) which has α. α^2. α^3... α^{2t}. as its roots.

Practically BCH code can be represented in most of the cases such as $BCH(n,k)$. A t-error correcting $BCH(N_{bch}, K_{bch})$ shall be applied to each BBFRAME (K_{bch}) The BCH code parameters are given in Table 4.2 for normal frame and in Table 4.4 for short frame. The generator of the t-errors correcting BCH encoder is obtained by simply multiplying the first t polynomials in table 4.5 for $n_{ldpc} = 64800$ bits and in Table 4.6 for $n_{ldpc} = 16200$ bits. Refereing

Table 4.5: BCH polynomials for normal FECFRAME $n_{ldpc} = 64800$ bits

$g_1(x)$	$1+x^2+x^3+x^5+x^{16}$
$g_2(x)$	$1+x+x^4+x^5+x^6+x^8+x^{16}$
$g_3(x)$	$1+x^2+x^3+x^4+x^5+x^7+x^8+x^9+x^{10}+x^{11}+x^{16}$
$g_4(x)$	$1+x^2+x^4+x^6+x^9+x^{11}+x^{12}+x^{14}+x^{16}$
$g_5(x)$	$1+x+x^2+x^3+x^5+x^8+x^9+x^{10}+x^{11}+x^{12}+x^{16}$
$g_6(x)$	$1+x^2+x^4+x^5+x^7+x^8+x^9+x^{10}+x^{12}+x^{13}+x^{14}+x^{15}+x^{16}$
$g_7(x)$	$1+x^2+x^5+x^6+x^8+x^9+x^{10}+x^{11}+x^{13}+x^{15}+x^{16}$
$g_8(x)$	$1+x+x^2+x^5+x^6+x^8+x^9+x^{12}+x^{13}+x^{14}+x^{16}$
$g_9(x)$	$1+x^5+x^7+x^9+x^{10}+x^{11}+x^{16}$
$g_{10}(x)$	$1+x+x^2+x^5+x^7+x^8+x^10+x^{12}+x^{13}+x^{14}+x^{16}$
$g_{11}(x)$	$1+x^2+x^3+x^5+x^9+x^{11}+x^{12}+x^{13}+x^{16}$
$g_{12}(x)$	$1+x+x^5+x^6+x^7+x^9+x^{11}+x^{12}+x^{16}$

to the standard of DVB-T2 the coding parameters for short FECFRAME $n_{ldpc} = 16200$ bits are given in Table 4.4. Looking at the given LDPC code rate, we can easily find out the required K_{bch} and N_{bch}. For instance for code rate $R = 1/4$, $K_{bch} = 3072$ and $N_{bch} = 3240$. The difference N_{bch}-K_{bch}=168. By multiplying the 12 polynomials given in Table 4.6 we will be able to obtain the so called 168^{th} grade generator polynomial. The reason why we need the generator polynomial is that BCH encoder have to obey the code length "n" for given "m". As we know $n = 2^m - 1$ for any $m \geq 3$. Given an integer $m \geq 3$ is impossible to get an "n" value which obeys the given relation above. By finding out the 168^{th} grade polynomial and using it in BCH encoder we will be able to perform the encoding part as required by the standard.

Table 4.6: BCH polynomials for short FECFRAME $n_{ldpc} = 16200$ bits

$g_1(x)$	$1+x^3+x^5+x^{14}$
$g_2(x)$	$1+x^6+x^8+x^{11}+x^{14}$
$g_3(x)$	$1+x+x^2+x^6+x^9+x^{10}+x^{14}$
$g_4(x)$	$1+x^4+x^7+x^8+x^{10}+x^{12}+x^{14}$
$g_5(x)$	$1+x^2+x^4+x^6+x^8+x^9+x^{11}+x^{13}+x^{14}$
$g_6(x)$	$1+x^3+x^7+x^8+x^9+x^{13}+x^{14}$
$g_7(x)$	$1+x^2+x^5+x^6+x^7+x^{10}+x^{11}+x^{13}+x^{14}$
$g_8(x)$	$1+x^5+x^8+x^9+x^{10}+x^{11}+x^{14}$
$g_9(x)$	$1+x+x^2+x^3+x^9+x^{10}+x^{14}$
$g_{10}(x)$	$1+x^3+x^6+x^9+x^{11}+x^{12}+x^{14}$
$g_{11}(x)$	$1+x^4+x^{11}+x^{12}+x^{14}$
$g_{12}(x)$	$1+x+x^2+x^3+x^5+x^6+x^7+x^8+x^{10}+x^{13}+x^{14}$

4.2.3. Zero Padding of BCH information bits

As mentioned above the BCH encoder will be an outer encoder. Refereing to the Table 4.4 on page 40 taken from the DVB-T2 standard we can easily figure out the respective BCH information bits (K_{bch}). Defining K_{sig} as the input binary data that have to be transmitted, if $K_{sig} \neq K_{bch}$ zero padding must be done. Part of information bits of the 16K LDPC code shall be zero padded in order to fill K_{bch}. For the given K_{sig} the number of zero padding bits is calculated as $(K_{bch} - K_{sig})$. As it is clearly stated in [22] the shorten procedure is as follows:

Step1) Compute the number of groups in which all the bits shall be padded, N_{pad} such that:

If $0 < K_{sig} \leq 360$, $N_{pad} = N_{group} - 1$

Otherwise, $N_{pad} = \left\lfloor \frac{K_{bch}-K_{sig}}{360} \right\rfloor$

Step2) For N_{pad} groups $X_{\pi_s(0)}, X_{\pi_s(1)}, \ldots, X_{\pi_s(m-1)}, X_{\pi_s(N_{pad}-1)}$, all information bits of the groups shall be padded with zeros. π_s is defined to be the permutation operator depending on the code rate and the modulation order as described in Table 4.7

Step3) If $N_{pad} = N_{group} - 1$, $(360 - K_{sig})$ information bits in the last part of the bit group $X_{\pi_s(N_{group}-1)}$ shall be additionally padded with zeros. Otherwise, for the group $X_{\pi_s(N_{pad})}$,

Modulation and Code rate		N_{group}	$\pi_s(j)$ $(0 \le j < N_{group})$								
			$\pi_s(0)$	$\pi_s(1)$	$\pi_s(2)$	$\pi_s(3)$	$\pi_s(4)$	$\pi_s(5)$	$\pi_s(6)$	$\pi_s(7)$	$\pi_s(8)$
QPSK	1/4	9	7	3	6	5	2	4	1	8	0

Table 4.7: Permutation sequence of information bit group to be padded.

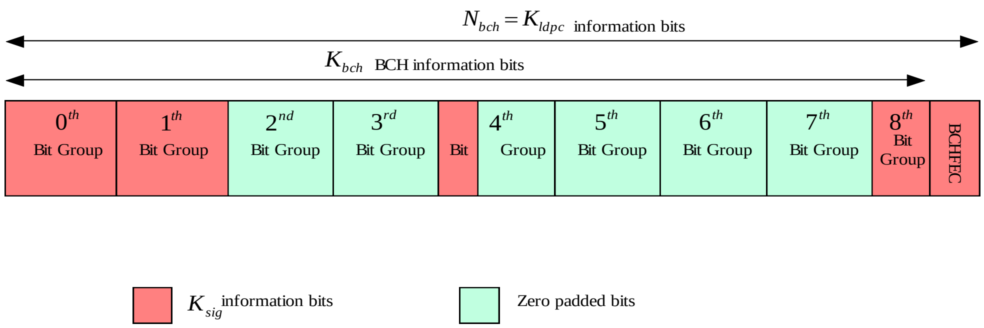


Figure 4.2: Example of shortening of BCH information part.

$\left(K_{bch} - K_{sig} - 360 \times N_{pad}\right)$ information bits in the last part of $X_{\pi_s(N_{pad})}$ shall be additionally padded.

Step4) Finally, K_{sig} information bits are sequentially mapped to bit positions which are not padded in K_{bch} BCH information bits, $\left(m_0, m_1, ..., m_{K_{bch}-1}\right)$ by the above procedure.

4.2.4. Low Density Parity Check code (optional)in WiMAX

As already mentioned in one of the sections above there are mainly two types of LDPC codes: Regular and irregular. The H matrix for optional LDPC coding has been defined in the WiMAX standard IEEE Std 802.16e^{TM}-2005 and is as follows:

$$H = \begin{bmatrix} P_{0,0} & P_{0,1} & P_{0,2} & \cdots & P_{0,n_b-2} & P_{0,n_b-1} \\ P_{1,0} & P_{1,1} & P_{1,2} & \cdots & P_{1,n_b-2} & P_{1,n_b-1} \\ P_{2,0} & P_{2,1} & P_{2,2} & \cdots & P_{2,n_b-2} & P_{2,n_b-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ P_{m_b-1,0} & P_{m_b-1,1} & P_{m_b-1,2} & \cdots & P_{m_b-1,n_b-2} & P_{m_b-1,n_b-1} \end{bmatrix} \quad (4.16)$$

Here $P_{i,j}$ corresponds to either a $(z \times z)$ permutation matrix or $(z \times z)$ zeros matrix. The matrix H given in the above form can be expanded to a binary base matrix H_b of size $(m_b \times n_b)$ where $n = z \times n_b$ and $m = z \times m_b$ as stated in [28].

The permutations used are circular right shifts, moreover the set of permutations matrices contains the $(z \times z)$ identity matrix and circular right shifted versions of the identity matrix. In [16] a binary base matrix H has been defined for the largest codeword length (n=2304) for various code rates. Since the base model matrix has 24 columns, the so called expansion factor $z_f = n/24$ for codeword length of n. For codeword length of 2304 the expansion factor would be 2304/24=96. Given a base model matrix H_{bm}, when $p(i,j) = -1$ it will be replaced by a $(z \times z)$ all-zero matrix and the other elements which correspond to $p(i,j) \geq 0$ will be replaced by circularly shifting the identity matrix by p(i,j). For code rate $\frac{1}{2}$, the base model matrix H_{bm} is defined as:

$$\begin{array}{cccccccccccccccccccccccc}
-1 & 94 & 73 & -1 & -1 & -1 & -1 & -1 & 55 & 83 & -1 & -1 & 7 & 0 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
-1 & 27 & -1 & -1 & -1 & 22 & 79 & 9 & -1 & -1 & -1 & 12 & -1 & 0 & 0 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
-1 & -1 & -1 & 24 & 22 & 81 & -1 & 33 & -1 & -1 & -1 & 0 & -1 & -1 & 0 & 0 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
61 & -1 & 47 & -1 & -1 & -1 & -1 & -1 & 65 & 25 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
-1 & -1 & 39 & -1 & -1 & -1 & 84 & -1 & -1 & 41 & 72 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & -1 & -1 & -1 & -1 & -1 & -1 \\
-1 & -1 & -1 & -1 & 46 & 40 & -1 & 82 & -1 & -1 & -1 & 79 & 0 & -1 & -1 & -1 & -1 & 0 & 0 & -1 & -1 & -1 & -1 & -1 \\
-1 & -1 & 95 & 53 & -1 & -1 & -1 & -1 & -1 & 14 & 18 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & -1 & -1 & -1 & -1 \\
-1 & 11 & 73 & -1 & -1 & -1 & 2 & -1 & -1 & 47 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & -1 & -1 & -1 \\
12 & -1 & -1 & -1 & 83 & 24 & -1 & 43 & -1 & -1 & -1 & 51 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & -1 & -1 \\
-1 & -1 & -1 & -1 & -1 & 94 & -1 & 59 & -1 & -1 & 70 & 72 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & -1 \\
-1 & -1 & 7 & 65 & -1 & -1 & -1 & -1 & 39 & 49 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 0 & 0 \\
43 & -1 & -1 & -1 & 66 & -1 & 41 & -1 & -1 & -1 & 26 & 7 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 0
\end{array} \tag{4.17}$$

For code rate $\frac{2}{3}A$, the base model matrix H_{bm} is defined as:

$$\begin{array}{cccccccccccccccccccccccc}
3 & 0 & -1 & -1 & 2 & 0 & -1 & 3 & 7 & -1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 & 0 & -1 & -1 & -1 & -1 & -1 & -1 \\
-1 & -1 & 1 & -1 & 36 & -1 & -1 & 34 & 10 & -1 & -1 & 18 & 2 & -1 & 3 & 0 & -1 & 0 & 0 & -1 & -1 & -1 & -1 & -1 \\
-1 & -1 & 12 & 2 & -1 & 15 & -1 & 40 & -1 & 3 & -1 & 15 & -1 & 2 & 13 & -1 & -1 & -1 & 0 & 0 & -1 & -1 & -1 & -1 \\
-1 & -1 & 19 & 24 & -1 & 3 & 0 & -1 & 6 & -1 & 17 & -1 & -1 & -1 & 8 & 39 & -1 & -1 & -1 & 0 & 0 & -1 & -1 & -1 \\
20 & -1 & 6 & -1 & -1 & 10 & 29 & -1 & -1 & 28 & -1 & 14 & -1 & 38 & -1 & -1 & 0 & -1 & -1 & -1 & 0 & 0 & -1 & -1 \\
-1 & -1 & 10 & -1 & 28 & 20 & -1 & -1 & 8 & -1 & 36 & -1 & 9 & -1 & 21 & 45 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & -1 \\
35 & 25 & -1 & 37 & -1 & 21 & -1 & -1 & 5 & -1 & -1 & 0 & -1 & 4 & 20 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 0 & 0 \\
-1 & 6 & 6 & -1 & -1 & -1 & 4 & -1 & 14 & 30 & -1 & 3 & 36 & -1 & 14 & -1 & 1 & -1 & -1 & -1 & -1 & -1 & -1 & 0
\end{array} \tag{4.18}$$

For code rate $\frac{2}{3}$ B, the base model matrix H_{bm} is defined as:

$$\begin{matrix}
2 & -1 & 19 & -1 & 47 & -1 & 48 & -1 & 36 & -1 & 82 & -1 & 47 & -1 & 15 & -1 & 95 & 0 & -1 & -1 & -1 & -1 & -1 & -1 \\
-1 & 69 & -1 & 88 & -1 & 33 & -1 & 3 & -1 & 16 & -1 & 37 & -1 & 40 & -1 & 48 & -1 & 0 & 0 & -1 & -1 & -1 & -1 & -1 \\
10 & -1 & 86 & -1 & 62 & -1 & 28 & -1 & 85 & -1 & 16 & -1 & 34 & -1 & 73 & -1 & -1 & -1 & 0 & 0 & -1 & -1 & -1 & -1 \\
-1 & 28 & -1 & 32 & -1 & 81 & -1 & 27 & -1 & 88 & -1 & 5 & -1 & 56 & -1 & 37 & -1 & -1 & -1 & 0 & 0 & -1 & -1 & -1 \\
23 & -1 & 29 & -1 & 15 & -1 & 30 & -1 & 66 & -1 & 24 & -1 & 50 & -1 & 62 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & -1 & -1 \\
-1 & 30 & -1 & 65 & -1 & 54 & -1 & 14 & -1 & 0 & -1 & 30 & -1 & 74 & -1 & 0 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & -1 \\
32 & -1 & 0 & -1 & 15 & -1 & 56 & -1 & 85 & -1 & 5 & -1 & 6 & -1 & 52 & -1 & 0 & -1 & -1 & -1 & -1 & -1 & 0 & 0 \\
-1 & 0 & -1 & 47 & -1 & 13 & -1 & 61 & -1 & 84 & -1 & 55 & -1 & 78 & -1 & 41 & 95 & -1 & -1 & -1 & -1 & -1 & -1 & 0
\end{matrix} \tag{4.19}$$

For code rate $\frac{3}{4}$ A, the base model matrix H_{bm} is defined as:

$$\begin{matrix}
6 & 38 & 3 & 93 & -1 & -1 & -1 & 30 & 70 & -1 & 86 & -1 & 37 & 38 & 4 & 11 & -1 & 46 & 48 & 0 & -1 & -1 & -1 & -1 \\
62 & 94 & 19 & 84 & -1 & 92 & 78 & -1 & 15 & -1 & 92 & -1 & 45 & 24 & 32 & -1 & 30 & -1 & -1 & 0 & 0 & -1 & -1 & -1 \\
71 & -1 & 55 & -1 & 12 & 66 & 45 & 79 & -1 & 78 & -1 & -1 & 10 & -1 & 22 & 55 & 70 & 82 & -1 & -1 & 0 & 0 & -1 & -1 \\
38 & 61 & -1 & 66 & 9 & 73 & 47 & 64 & -1 & 39 & 61 & 43 & -1 & -1 & -1 & -1 & 95 & 32 & 0 & -1 & -1 & 0 & 0 & -1 \\
-1 & -1 & -1 & -1 & 32 & 52 & 55 & 80 & 95 & 22 & 6 & 51 & 24 & 90 & 44 & 20 & -1 & -1 & -1 & -1 & -1 & -1 & 0 & 0 \\
-1 & 63 & 31 & 88 & 20 & -1 & -1 & -1 & 6 & 40 & 56 & 16 & 71 & 53 & -1 & -1 & 27 & 26 & 48 & -1 & -1 & -1 & -1 & 0
\end{matrix} \tag{4.20}$$

For code rate $\frac{3}{4}$ B, the base model matrix H_{bm} is defined as:

$$\begin{matrix}
-1 & 81 & -1 & 28 & -1 & -1 & 14 & 25 & 17 & -1 & -1 & 85 & 29 & 52 & 78 & 95 & 22 & 92 & 0 & 0 & -1 & -1 & -1 & -1 \\
42 & -1 & 14 & 68 & 32 & -1 & -1 & -1 & -1 & 70 & 43 & 11 & 36 & 40 & 33 & 57 & 38 & 24 & -1 & 0 & 0 & -1 & -1 & -1 \\
-1 & -1 & 20 & -1 & -1 & 63 & 39 & -1 & 70 & 67 & -1 & 38 & 4 & 72 & 47 & 29 & 60 & 5 & 80 & -1 & 0 & 0 & -1 & -1 \\
64 & 2 & -1 & -1 & 63 & -1 & -1 & 3 & 51 & -1 & 81 & 15 & 94 & 9 & 85 & 36 & 14 & 19 & -1 & -1 & -1 & 0 & 0 & -1 \\
-1 & 53 & 60 & 80 & -1 & 26 & 75 & -1 & -1 & -1 & -1 & 86 & 77 & 1 & 3 & 72 & 60 & 25 & -1 & -1 & -1 & -1 & 0 & 0 \\
77 & -1 & -1 & -1 & 15 & 28 & -1 & 35 & -1 & 72 & 30 & 68 & 85 & 84 & 26 & 64 & 11 & 89 & 0 & -1 & -1 & -1 & -1 & 0
\end{matrix} \tag{4.21}$$

For code rate $\frac{5}{6}$, the base model matrix H_{bm} is defined as:

$$\begin{matrix}
1 & 25 & 55 & -1 & 47 & 4 & -1 & 91 & 84 & 8 & 86 & 52 & 82 & 33 & 5 & 0 & 36 & 20 & 4 & 77 & 80 & 0 & -1 & -1 \\
-1 & 6 & -1 & 36 & 40 & 47 & 12 & 79 & 47 & -1 & 41 & 21 & 12 & 71 & 14 & 72 & 0 & 44 & 49 & 0 & 0 & 0 & 0 & -1 \\
51 & 81 & 83 & 4 & 67 & -1 & 21 & -1 & 31 & 24 & 91 & 61 & 81 & 9 & 86 & 78 & 60 & 88 & 67 & 15 & -1 & -1 & 0 & 0 \\
50 & -1 & 50 & 15 & -1 & 36 & 13 & 10 & 11 & 20 & 53 & 90 & 29 & 92 & 57 & 30 & 84 & 92 & 11 & 66 & 80 & -1 & -1 & 0
\end{matrix} \tag{4.22}$$

Chapter 5

OVERVIEW OF TRANSMISSION BLOCK DIAGRAM

In order to test the performance of Low-density Parity-check codes, a transmission system is adopted. The block diagram of our simulation system used in MATLAB to evaluate the error correction ability of the LDPC FEC scheme is described in Figure 5.1. The RGB im-

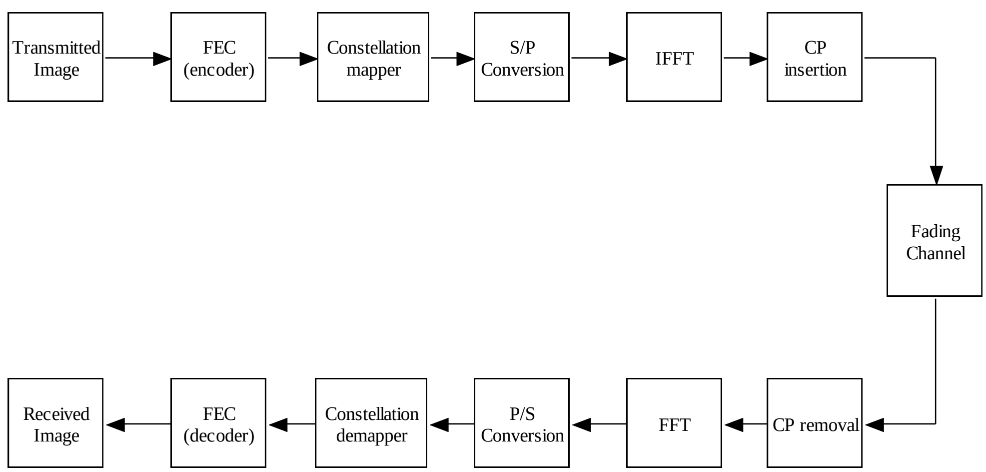


Figure 5.1: Image transmission and Reception model.

age is acquired and then it is converted to gray scale. In order for our system to be robust after converting the image to gray scale, the image will be resized to 180×225 using bicubic method. The original images used are shown in Figure 5.2. After getting the binary data of our test images, they are protected by using FEC channel coding. For comparison purposes as the FEC scheme we are using LDPC coding and RS-CC coding. For DVB-S2 and DVB-T2 standard, LDPC coding is used with the appropriate parameters obtained from the standard. RS-CC coding is used in case of DVB-S and DVB-T standard. As mentioned above, we know that LDPC FEC scheme is used as optional encoding scheme in WiMAX standard and the parameters used have been obtained from that standard as well [17]. The encoded stream is then fed into the constellation mapper, QPSK in our studies. This constellation mapper produces one symbol for every two bits, after which the signal is modulated by IFFT and lengthened by addition of a cyclic prefix of a certain length. The cyclic prefix is a unique feature of OFDM that protects the data from inter-symbol interference (ISI). The sequence of blocks is modulated according to the OFDM technique, using 2048, 4096, or 8192 carriers (2k, 4k, 8k mode, respectively). Once this has been done, the image is then transmitted over the channel where it is affected by additive noise and multipath fading channel.

The FEC code rates adopted by our simulations, the maximum Doppler frequency and the type of fading channels used are summarized in Table 5.1 and Table 5.2. A 180×225 grey

Table 5.1: Systems parameters with BCH-LDPC encoder.

Parameter	Value
FEC	BCH(3240,3072,12) LDPC(3240,16200) BCH(7200,3240,12) LPDC(7200,16200)
Channel	ITU-Vehicular A ITU-Vehicular B
Doppler spectrum	Jakes'
Max f_d	300 Hz

scale image was protected by the FEC schemes and transmitted over the AWGN and fading channels. The quality of reception was measured by observing the bit error rate (BER) and peak signal to noise ratio (PSNR) values over a set of SNR values.

Table 5.2: System Parameters with just LDPC encoder

Parameters	WiMAX	DVB-T	DVB-T2
FEC	RS(255,239,8) CC(1,2,7) LDPC(1152,2304) LDPC(1536,2304)	RS(204,188,8) CC(1,2,7)	LDPC(16200,64800) LDPC(21600,64800) LDPC(21600,64800)
Channel	ITU-Vehicular A & ITU-Vehicular B channel		
Doppler spectrum	Jakes'		
Max f_d	300 Hz		

(a) (b)

Figure 5.2: Transmitted images

5.1. FEC Frame Formation

The FEC frame is the output of the FEC sub-system when a BBFrame is the input; that is after BCH and LDPC encoding. This frame as specified in [17], and shown in Figure 4.2, is made up of the BB Frame, BCHFEC, and the LDPCFEC. The BB Frame is of length K_{bch}

and is the input to the BCH encoder. The BCH code will require shortening and zero padding if the size of the data to be encoded is not perfectly divisible by K_{bch}. This padding process is described in [19]. For example if the size of the transmitted grey scale image is 160×200; corresponding to a total of 256000 bits; for a code rate of $\frac{1}{4}$, the value of K_{bch} is 3072; this value does not perfectly divide the length of our data thus, if we shorten the BCH code by choosing a [19] of 2000, this would mean that the input data will be encoded in 128 separate data blocks each of length . After BCH encoding, parity bits are appended to the BB Frame and then the resulting output is LDPC encoded to form the FEC frame.

5.2. Cyclic Prefix

Inter-symbol interference occurs when the signal passes through the time dispersive channel. In an OFDM system, it is also possible that orthogonality of the subscribers may be lost, resulting in inter carrier interference. OFDM system uses cyclic prefix (CP) to overcome these problems. A cyclic prefix is the copy of the last part of the OFDM symbol to the beginning of transmitted symbol and removed at the receiver before demodulation. The cyclic prefix should be at least as long as the length of impulse response. However, there is a limit on energy while increasing the length of cyclic prefix. As it is expected the energy increases as the cyclic prefix length increases. As it is stated in [41] the SNR loss due to the usage of cyclic prefix can be evaluated using equation 5.1.

$$SNR_{loss} = -10\ log_{10}\left(1 - \frac{T_{cp}}{T}\right) \tag{5.1}$$

In the equation 5.1 T_{cp} refers to the cyclic prefix length. We can express the length of the transmitted symbol $T = T_{cp} + T_s$. Choosing the length of the cyclic prefix must be done carefully. The following matters should be considered,

1. Number of symbols per second decreases to $R(1 - T_{cp}/T)$

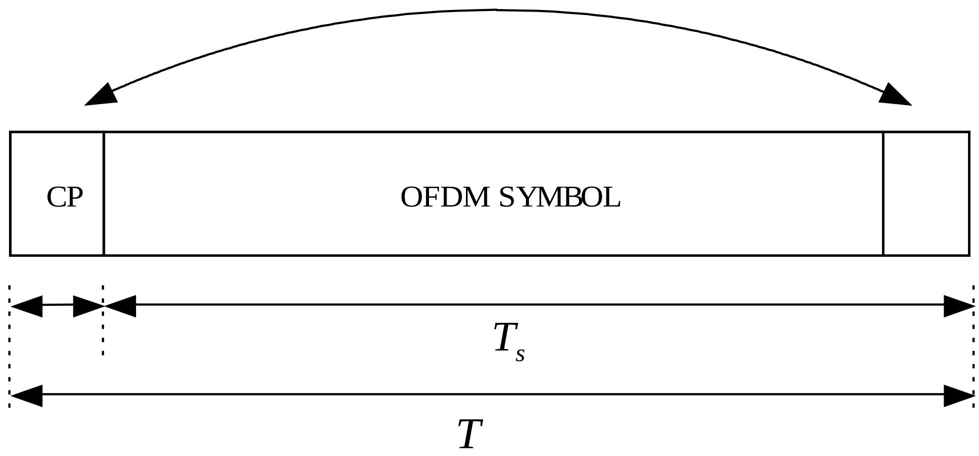


Figure 5.3: Cyclic Prefix.

2. The ratio T_{cp}/T must be kept as small as possible

As it is stated in [17] the width of the guard interval can be $R = 1/32$, $R = 1/16$, $R = 1/8$, or $R = 1/4$ that of the original block length. In our simulation we are using a guard interval width $R = 1/4$ of the original block length.

Chapter 6

SIMULATIONS AND PERFORMANCE ANALYSIS

This section sets out to show the BER, PSNR and psychovisual performances of LDPC-only and concatenated BCH-LDPC coded QPSK-OFDM over AWGN and multipath Rayleigh fading channels. Firstly, simulations are carried out over the AWGN channel for concatenated BCH-LDPC coding and compared with the LDPC only scheme; the observation of the performance of the concatenated BCH-LDPC scheme is based mostly on the analysis of the BER curve since the BER performance is clearly reflected to the PSNR and psychovisually performances. The same simulations are carried out over the Rayleigh fading channel with the fading parameters presented in Table 4.2, 4.4, 5.1 and 5.2. Furthermore, the simulations are repeated for the DVB-S2 and WiMAX standards.

6.1. DVB-S2 Channel Coding

This section sets out to show the link-level BER and PSNR performances of RS-CC and LDPC coded QPSK-OFDM over AWGN and multipath Rayleigh fading channels. Four different scenarios are considered. Firstly the RS-CC concatenated coding with RS(255,239,8) and $CC(1,2,7)$ as suggested in the mobile WiMAX standard is simulated. Then, $RS(204,188,8)$ and $CC(1,2,7)$ stated by the European DVB-T standard is simulated and compared against previous set of results. In order to compare and contrast the performance of concatenated coding with those of LDPC coded system performances the code rates and corresponding parity check matrices provided in Table 4.3 (as suggested in DVB-T2 and mobile WiMAX) were also simulated. For LDPC coded system no interleavers were employed since LDPC encoders themselves have inherently good interleaving properties.

6.1.1. Image transmission over AWGN channel

Figure 6.1, depicts the BER performance of the RS-CC coded system over the AWGN channel using the image shown in Figure 5.2b and the RS and CC parameters stated in the mobile WiMAX and DVB-T standards. The slight difference in coding gains achieved by the two

Table 6.1: PSNR Performance using LDPC codes over the AWGN channel

SNR(dB)	WiMAX		DVB-T2	
	R=1/2	R=2/3B	R=1/4	R=1/3
	PSNR (dB)			
0	13.87	11.05	–	–
1	19.49	11.48	10.07	9.93
2	inf	12.12	10.83	10.31
3	inf	12.87	14.85	10.94

RS-CC curves is as a result of shortening the code word length. As noted in [42] a shorter code word length will improve the performance of the RS encoder. In order to assess the quality of the recovered images the peak signal to noise ratio (PSNR) was also examined for the LDPC code rates depicted in Figure 6.2. For the various SNR values shown in Table 6.1 the PSNRs were computed using (6.1) and (6.2) where, $\max(g(x,y))$ is the maximum possible pixel value in the $(u \times v)$ image.

$$PSNR(dB) = 10 \times \log \frac{\max(g(x,y))}{MSE} \tag{6.1}$$

$$MSE = \sum_{i=1}^{v} \sum_{j=1}^{v} \frac{(g(x,y) - \hat{g}(x,y))^2}{uv} \tag{6.2}$$

The system's BER performance over the AWGN channel using the optional LDPC coding

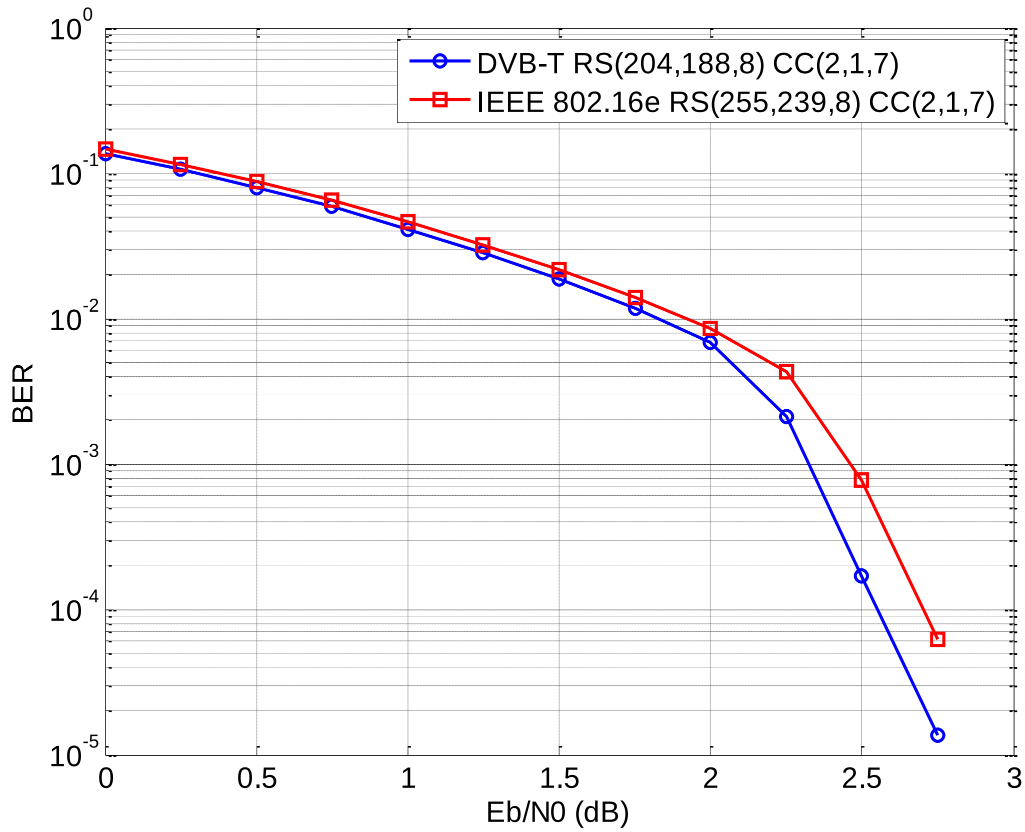


Figure 6.1: BER performance over the AWGN channel using RS-CC coding.

of mobile WiMAX and LDPC coding of DVB-T2 has been summarized in Figure 6.2. Even though more than two code rates are possible for each standard, in this work only two code rates leading to better performances were chosen for each standard. As can be observed from the Figure 6.2 the best BER is obtained using the rate $R = \frac{1}{2}$ LDPC code for IEEE 802.16e. Zero error decoding becomes possible after an SNR of approximately 1.5 dB. For the code rate $R = \frac{2}{3}A$ the BER performance is clearly worst than the code rate $R = \frac{1}{2}$ as it is expected. Here Zero error decoding becomes possible after 6 dB. Furthermore, attaining a BER level of 10^{-3} is possible at an SNR of 1.5 dB for code rate $R = \frac{1}{2}$ and at a SNR of 5.5 dB for code rate $R = \frac{2}{3}B$.

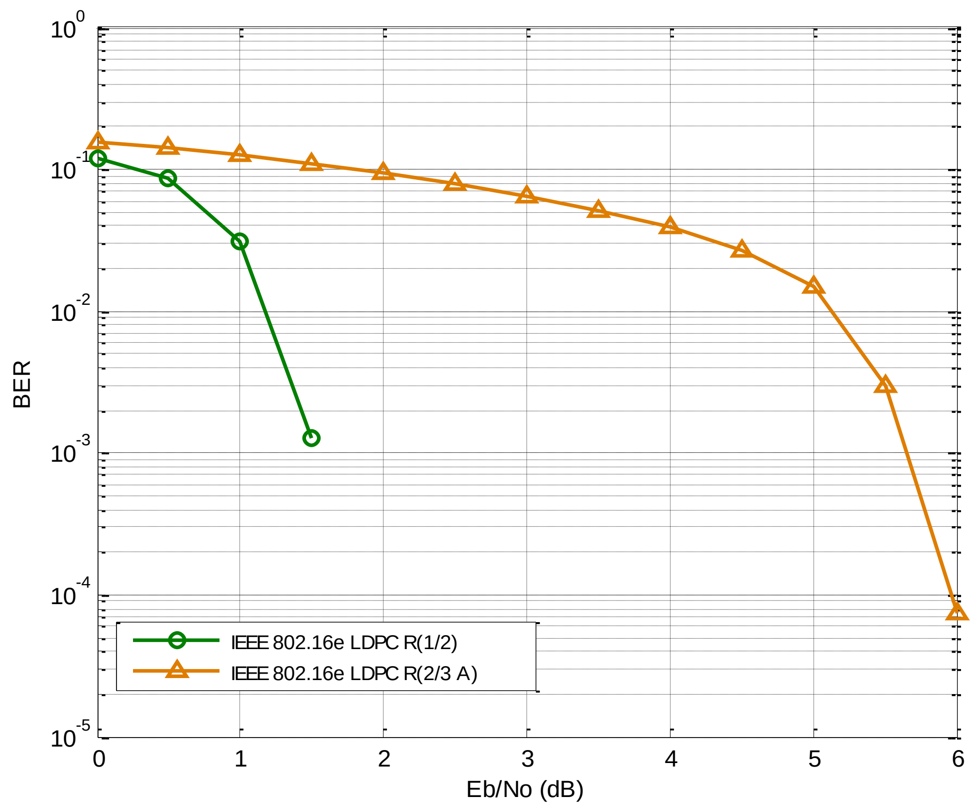


Figure 6.2: BER performance over AWGN channel using LDPC coding

6.1.2. Image transmission over Fading channels

This section provides a comparative analysis for RS-CC and LDPC coded system performances over the ITU Vehicular-A channel as well as gives the performance of LDPC coded system over ITU Vehicular-B channel. Fading channels are known to degrade the system's BER performance more than an AWGN channel and they are refereed as worst degrading channels. The parameter which affects data transmission the most in the context of small scale fading is the Doppler frequency. In this work, the Doppler frequency assumed to be 300 Hz. This amount of shift roughly corresponds to a speed of 90 km/hr for the ITU Vehicular A channel and to a speed of 120 km/hr for the ITU Vehicular B channel. Comparison for LDPC code for shorten length and normal length frame is shown in Figure 6.3. As we can

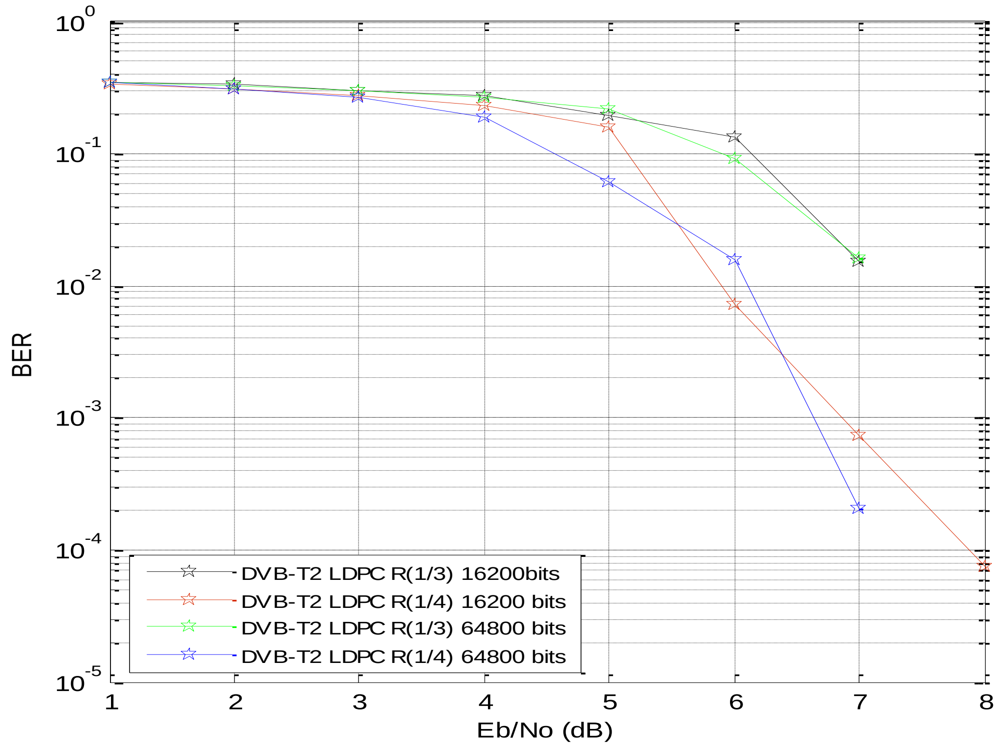


Figure 6.3: BER performance over the ITU Vehicular-A channel using LDPC coding for DVB-T2

see the LDPC with codeword length 64800 bits, with code rate $R = \frac{1}{4}$ has an improvement performance comparing with the same code rate but for codeword length of 16200 bits. As it was expected the LPDC codes usually performs better for long codeword length. However we are aware that the given BER performance is not so accurate and more bits need to be transmitted through the system in order to obtain a more accurate BER performance for both codeword length. Figure 6.4 shows the LDPC coded system performance for the DVB-T2 standard over the ITU-A channel for two different code rate [43]. Figure 6.5 depicts the recovered images transmitted using DVB-T2 over the ITU Vehicular-A channel by means of LDPC coding scheme for SNR values of $4, 10, 16$ and 20 dB. As can be observed, the quality of the received image progressively improves as the SNR increases. For instance given the value of $SNR = 4$ dB the received image condition is subject to discussion to decide if it is acceptable psychovisually or not.

Having obtained the respective PSNR values of received images, looking at the visually performance of them, some filtering methods can be chosen to be applied to possible minimize the effect of the noise and smoothen the image. For SNR values equal to and greater than 20 dB, error free reception is achieved [43]. Looking at the given BER performance we can say that achieving BER level of 10^{-5} is possible at a SNR of 8.2 dB for code rate $R = \frac{1}{4}$ and at a SNR of 9 dB for a code rate $R = \frac{1}{3}$. Clearly, as it was expected the LDPC $R = \frac{1}{4}$ performs better and in this case has a gain of 1 dB.

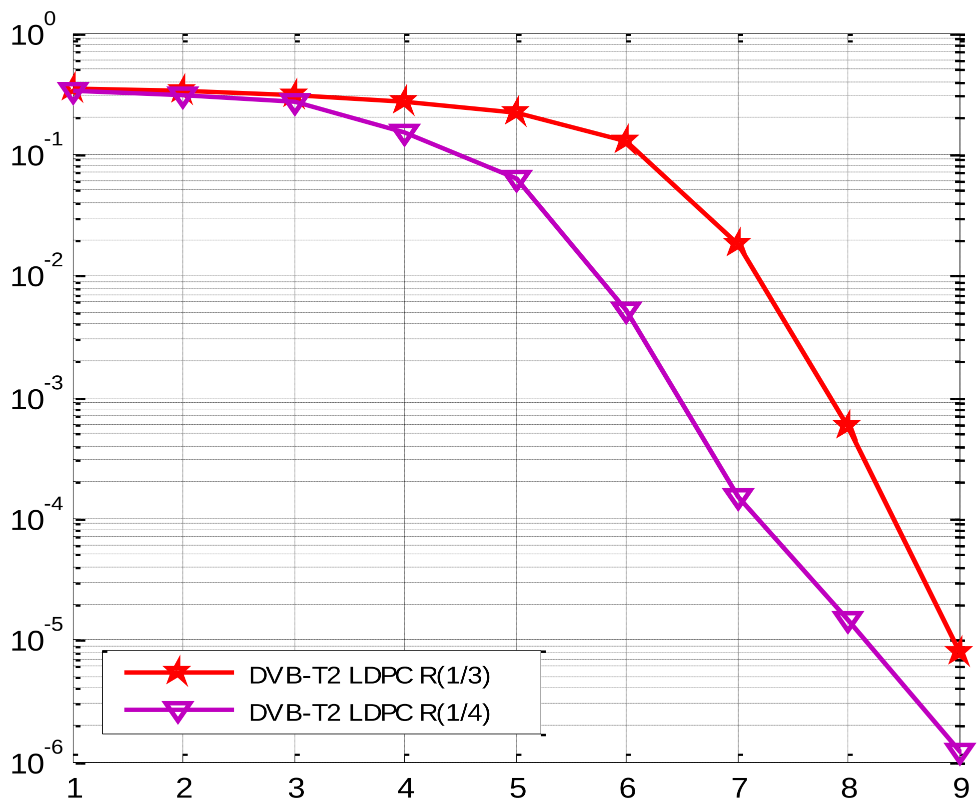


Figure 6.4: BER performance over the ITU-A channel using LDPC coding.

The computed PSNR values for the RS-CC coding of DVB-T standard has been summarized for both the AWGN and ITU Vehicular-A channels in Table 6.2. Note that over the AWGN channel a PSNR value of 30.37 dB is attained for an SNR value of 2.25 dB. However on the ITU Vehicular-A channel a similar performance is only possible around 15 dB. For AWGN channel the free error reception is possible for a SNR $\geq$2.5 dB. However, for the fading channel the free error reception is possible only for SNR $\geq$ 18 dB. This clearly points out the degrading effect of the fading mobile communication channel. The next set of simulation results are from using LDPC parameters for WiMAX and DVB-T2. In Figure 6.6, the IEEE 802.16e LDPC code with rate $R = 1/2$ performs best with zero error decoding starting at an SNR of about 5 dB.

The second best performance is attained by using the rate $R = \frac{1}{4}$ LDPC code dictated by the DVB-T2 standard as the FEC scheme. Comparing the code rate $R = \frac{1}{2}$ and $R = \frac{2}{3}B$ for IEEE 802.16e, leads to a conclusion that the trade off between the two code rates can be done by giving up a 5 dB performance for obtaining a free error reception. In Figure 6.7 we compare the best LDPC codes with the concatenated RS-CC codes in order to highlight the drastic improvement in the performance of the system when LDPC codes are used in a Rayleigh fading channel with the consideration of Doppler effect. For example there is a coding gain of about 9 dB for a target BER of 10^{-2} when the IEEE 802.16e LDPC $R = (\frac{1}{2})$ is used instead of the IEEE 802.16e $RS(255,239,8)$ $CC(2,1,7)$. Clearly the usage of LDPC encoders brings a big improvement to the system's BER performance. All the PSNR values for received images while using rate $R = \frac{1}{4}$ and $R = \frac{2}{3}B$ WiMAX LDPCs and rate $R = \frac{1}{4}$ and $R = \frac{1}{3}$ DVB-T2 LDPC encoders have been provided in Table 6.3. Figure 6.8 and Figure 6.9 depict the recovered images after LDPC decoding of the received data sequences. Looking at the image received under 1 dB SNR, we can say that the image is unrecognizable and surrounded by noise. Even by filtering the image it is quite hard to smoothen it and remove the noise.

(a) 4 dB

(b) 10 dB

(c) 16 dB

(d) 20 dB

Figure 6.5: Recovered images transmitted using DVB-T over the ITU Vehicular-A channel.

Table 6.2: PSNR performance using RS-CC scheme of DVB-T standard over additive and fading channels

SNR v.s. PSNR results for DVB-T RS-CC coding over additive and fading channels (RS(204,188,8) and CC(1,2,7))			
AWGN		Fading Channel ITU Vehicular-A	
SNR	PSNR	SNR	PSNR
0	13.04	0	9.46
0.25	14.14	2	11.26
0.50	15.50	4	13.57
0.75	16.69	6	15.88
1	18.22	8	19.02
1.25	19.81	10	22.83
1.5	21.74	12	22.34
1.75	23.47	14	26.82
2	26.16	16	32.41

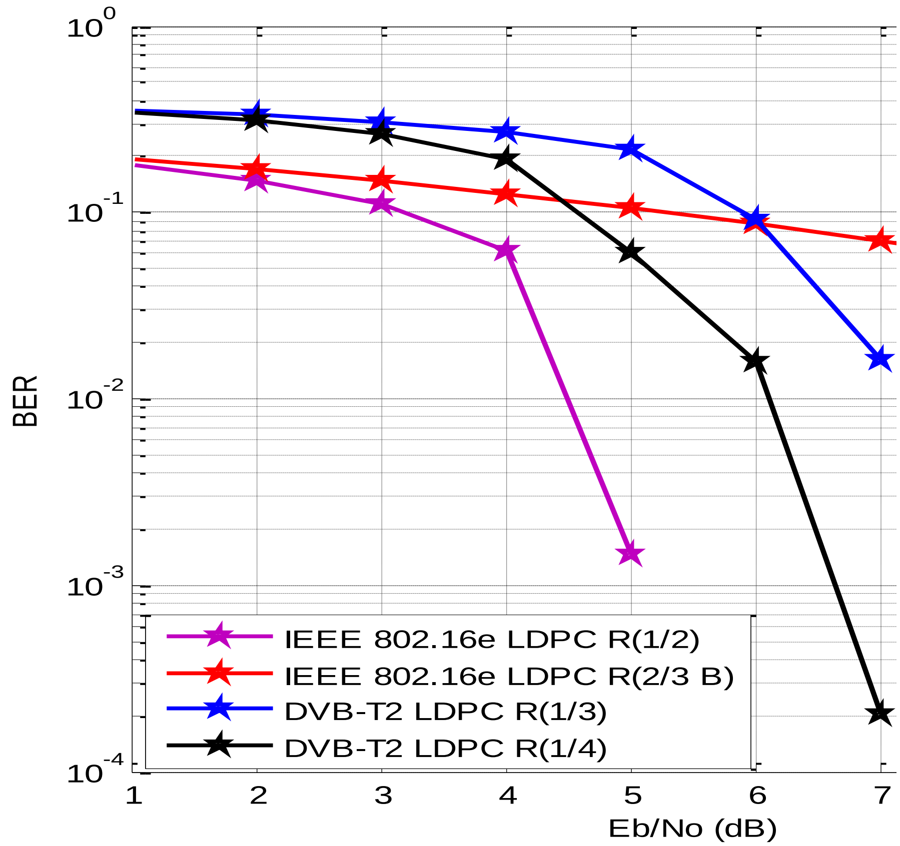


Figure 6.6: BER performance over Rayleigh fading channel using LDPC coding.

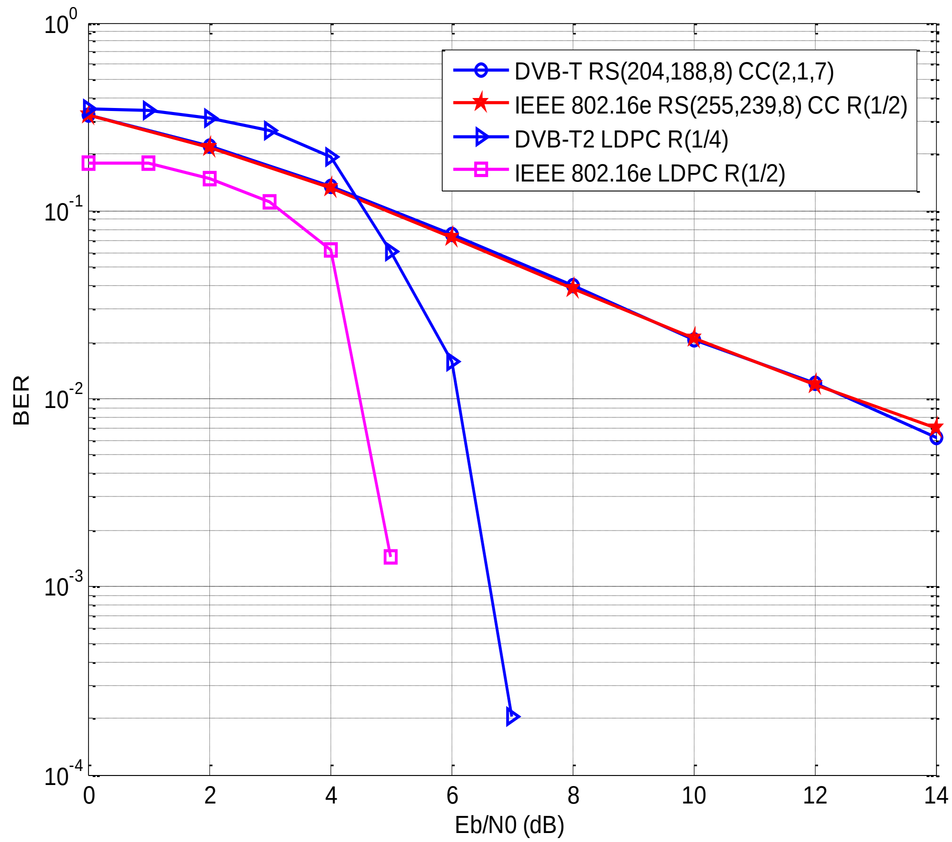


Figure 6.7: Comparison of BER performance over Rayleigh fading channel using LDPC coding and concatenated RS-CC coding.

Table 6.3: PSNR Performance using LDPC codes over the ITU-Vehicular A channel

SNR(dB)	WiMAX		DVB-T2	
	R=1/2	R=2/3B	R=1/4	R=1/3
	PSNR (dB)			
1	12.29	11.92	9.54	9.47
2	13.22	12.44	9.96	9.72
3	14.35	13.01	10.61	10.15
4	32.85	13.68	12.05	10.54

However looking at the PSNR value, it gives us a taste that the image can be reconstructed by means of filtering or some other algorithms. For WiMAX with $R = (\frac{1}{2})$ error free reception is possible after 5 dB. Looking at the received images using DVB-T2 channel for SNR values 1 dB and 3 dB the image is quite disturbed and a lot of effort must be made probably to minimize the error level. However looking at the image received under 5 dB SNR level we can conclude that the image is probably filterable and easily can be smoothen out. Similarly for the DVB-T2 LDPC with rate $R = (\frac{1}{4})$ error free reception starts around 8 dB.

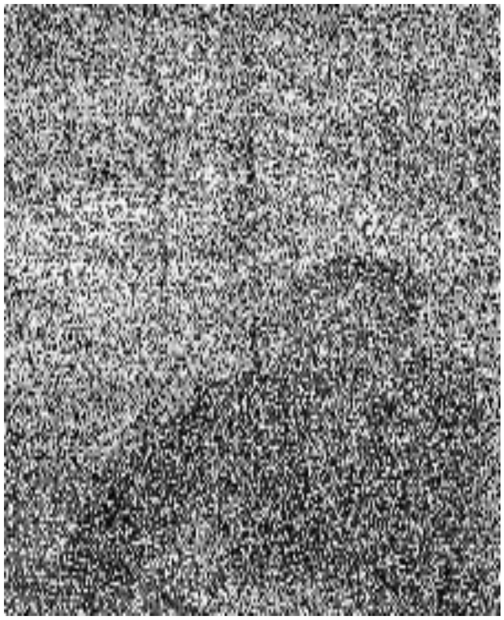
(a) 1 dB

(b) 3 dB

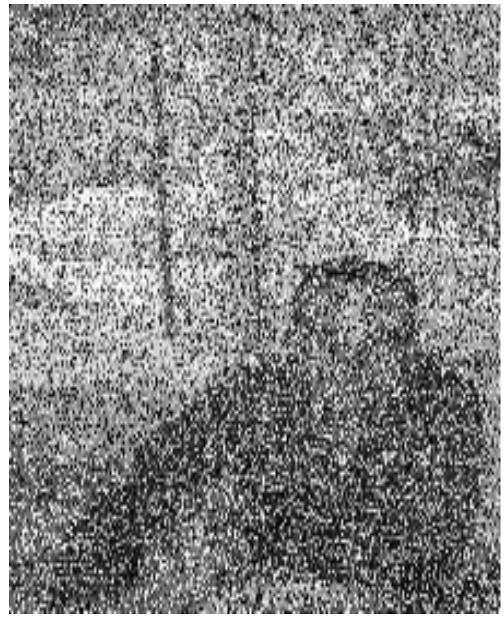
(c) 5 dB

(d) 6 dB

Figure 6.8: Recovered image transmitted over ITU-Vehicular A channel using (R = 1/2) LDPC as FEC scheme.



(a) 1 dB



(b) 3 dB

(c) 5 dB

(d) 7 dB

Figure 6.9: Received image transmitted over ITU Vehicular-A channel using (R = 1/4)LDPC as the FEC scheme.

6.2. DVB-T2 Channel Coding

This section sets out to show the BER, PSNR and psycho-visual performances of LDPC-only and concatenated BCH-LDPC coded QPSK-OFDM over AWGN and multipath Rayleigh fading channels. Firstly simulations are carried out over the AWGN channel for concatenated BCH-LDPC coding and for the LDPC only scheme; the comparison between them is clearly stated. Furthermore, the comparison between two different code rates for BCH-LDPC coding schemes is provided and a brief discussion is settled. The same simulations are carried out over the Rayleigh fading channel with the fading parameters presented in Table 4.4 and in Table 5.2

6.2.1. Image transmission over AWGN channel

Figure 6.10 presents the BER curves obtained for rate $R = \frac{1}{3}$ and rate $R = \frac{1}{4}$ BCH-LDPC coded systems. As can be observed from the figure the best BER is obtained for the rate $R = \frac{1}{4}$ system as it was expected. We note that after an SNR of 3 dB all decoding will be error free for code rate $R = \frac{1}{4}$. Similarly, for code rate $R = \frac{1}{3}$ the free error decoding will be possible for an SNR level of 4.75 dB. The generator polynomial specified for the given BCH encoder is of the grade 168^{th} and can correct up to 12 bit errors. Related to the generator and primitive polynomials please refer to Table 4.5 on page 42 and Table 4.6 on page 43.

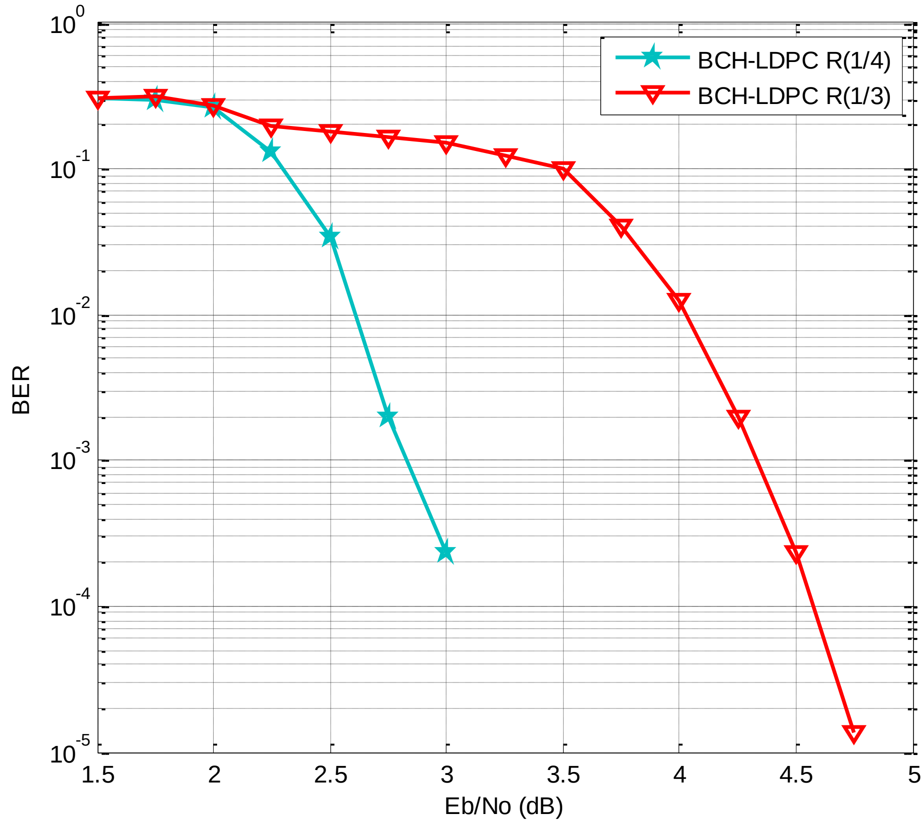


Figure 6.10: BER performance over AWGN channel using concatenated BCH-LDPC coding

In Figure 6.11 the performance of LDPC-only coded with code rate $R = \frac{1}{3}$ is shown. In this case we are expecting a degradation of our system performance because we are using just LDPC-only coding schemes. However this discussion will be settled down when Figure 6.12 is considered.

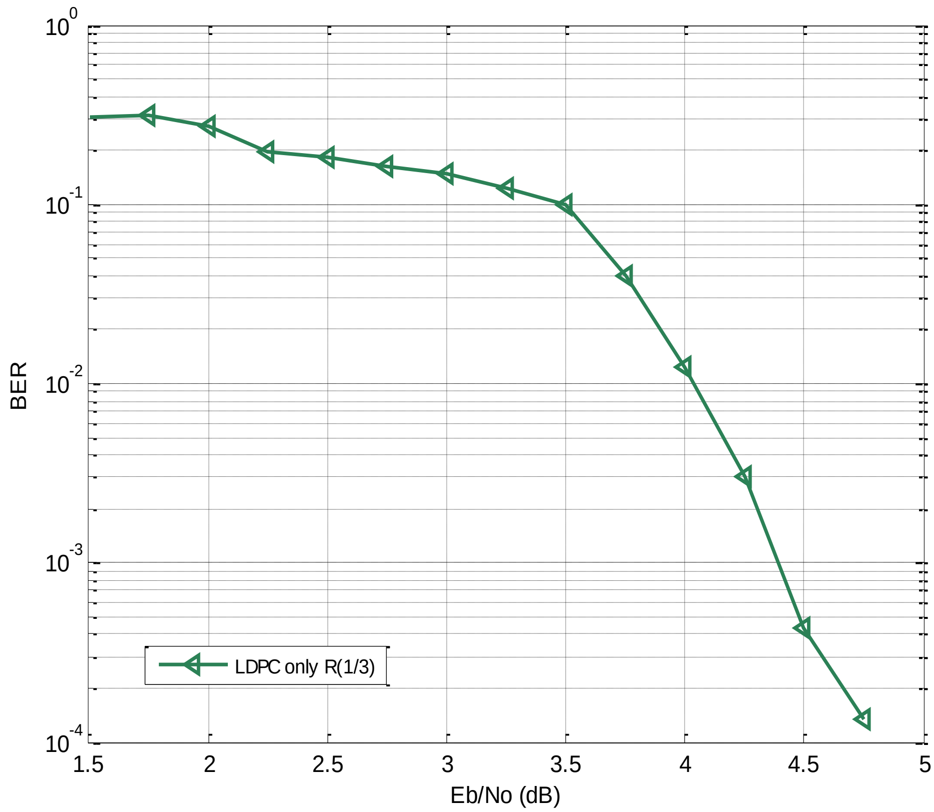


Figure 6.11: BER performance over AWGN channel using LDPC-only coding

Figure 6.12 shows the performance of concatenated BCH-LDPC coding and LDPC only coding over an AWGN channel. As we can clearly see from the Figure 6.12 after an SNR of 4 dB we can observe a slightly amount of BER gain. For instance, for a target BER of 10^{-4} with LDPC only coding scheme this is possible at a SNR of 4.75 dB. However, for BCH-LDPC coding scheme that is possible for a SNR of 4.55 dB. That is as it was expected. We know that for high SNR values BCH-LDPC scheme performs better than LDPC only coding scheme with an BER gain of 0-0.3 dB. In our case this BER gain is about 0.2 dB. For low SNR values the performance is exactly the same for both coding schemes.

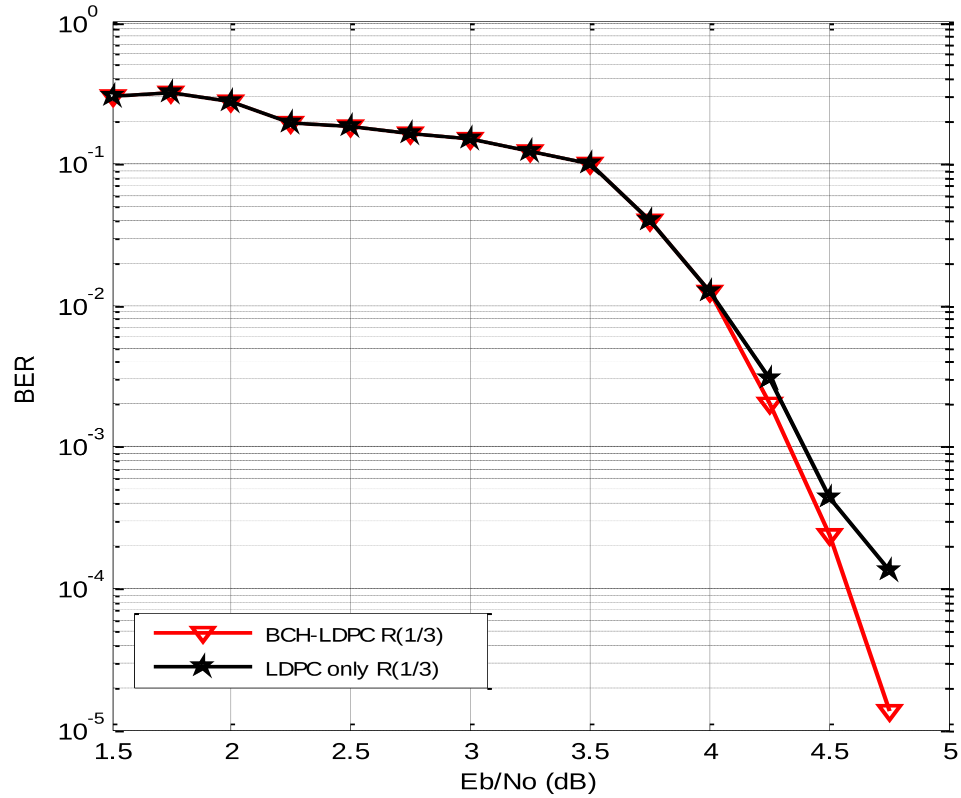


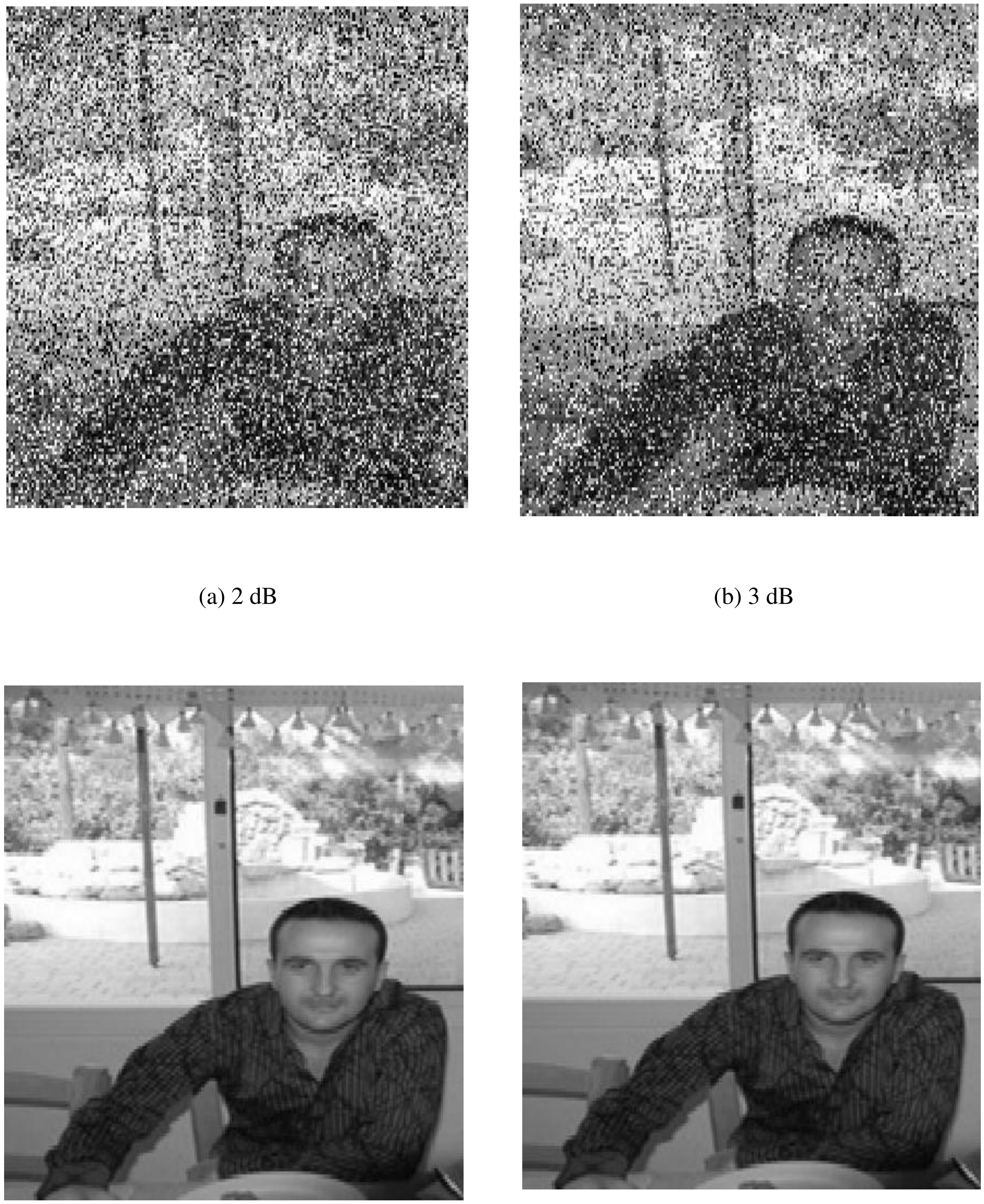
Figure 6.12: BER performance over AWGN channel using concatenated BCH-LDPC coding and LDPC coding

Table 6.4 summarizes the received image PSNR values for both coding schemes (BCH-LDPC and LDPC). According to the results, when BCH-LDPC coding is used in the presence of bit errors, it is possible to receive the transmitted image without any errors after an SNR value of 3.5 dB; but when LDPC-only is used under the same conditions, a slightly degradation in the system is observed. Moreover looking at the table results and comparing the two schemes we can observe that we have a slight gain of PSNR dB values. For instance for a SNR level of 1 dB the LDPC coding scheme gives us 10.01 dB PSNR level. However for the same SNR level the LDPC-BCH coding scheme gives us 10.33 dB PSNR level. Figure 6.13 depicts the quality of decoded images after the test image has been transmitted over the AWGN channel.

Table 6.4: PSNR performance using rate $R = \frac{1}{4}$ LDPC and BCH-LDPC codes over the AWGN channel

SNR (dB)	BCH-LDPC	LDPC
0	9.75	9.6
1	10.33	10.01
2	10.99	10.75
3	14.85	12.52

Free error decoding (high PSNR level) will be possible for a SNR level of approximately 4.5 dB for a code rate $R = \frac{1}{4}$. Psychovisualy, we can state that the received images under an SNR level of 2 dB and 3 dB are so hard to be filtered out in order to probably smoothen the images or minimize the psychovisually error level.



(a) 2 dB

(b) 3 dB

(c) 4 dB

(d) 7 dB

Figure 6.13: Decoded image at various SNR values for concatenated BCH-LDPC coding over the AWGN channel.

6.2.2. Image transmission over Fading channels

Figure 6.14 shows the rate $R = \frac{1}{4}$ BCH-LDPC coded system BER performance over the ITU Vehicular-A channel. Herein the free error decoding may be possible for a SNR level of greater than 10 dB. For instance, for a target BER of 10^{-5} the corresponding SNR level will be approximately 7.3 dB. For a range of SNR values between 7 dB to 8.5 dB the correspond-

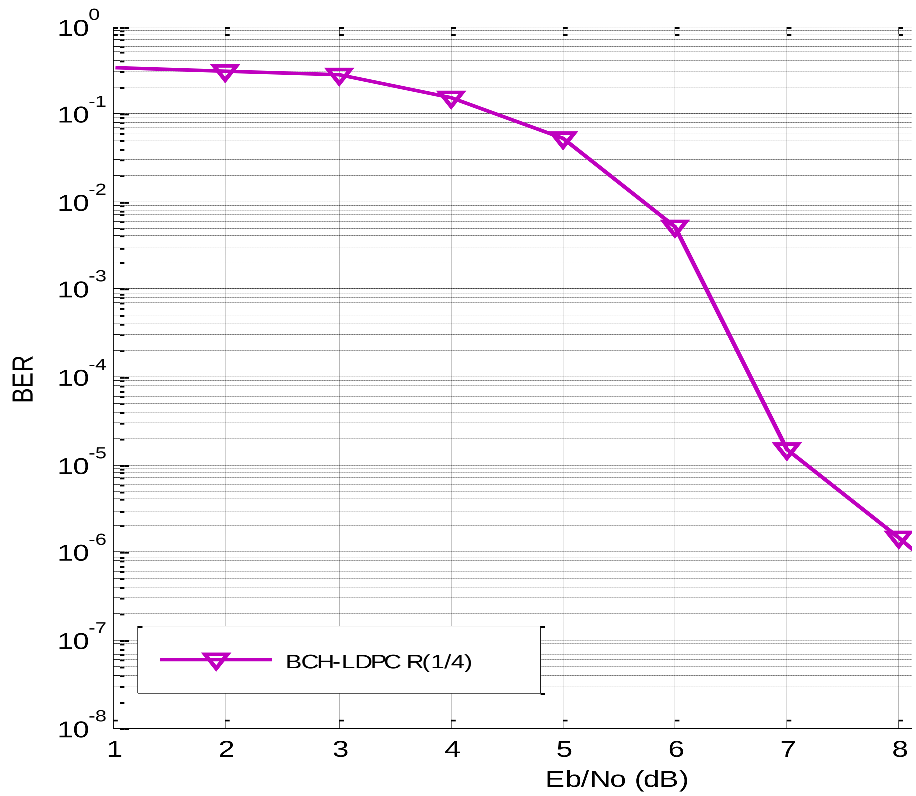


Figure 6.14: BER performance over Rayleigh fading channel using concatenated BCH-LDPC coding

ing BER level is approximately around 10^{-6} which means that in 1000000 bits transmitted one of them is decoded not correctly. This level of bit errors rate is translated most probably to a high PSNR values of recovered images and good looking psychovisually.

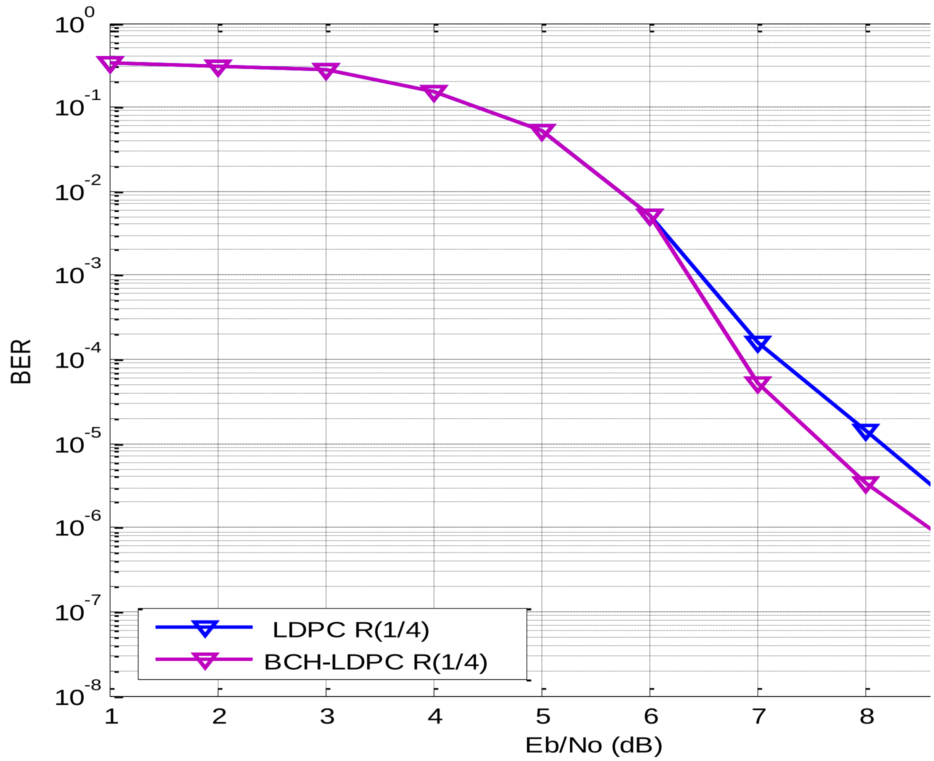
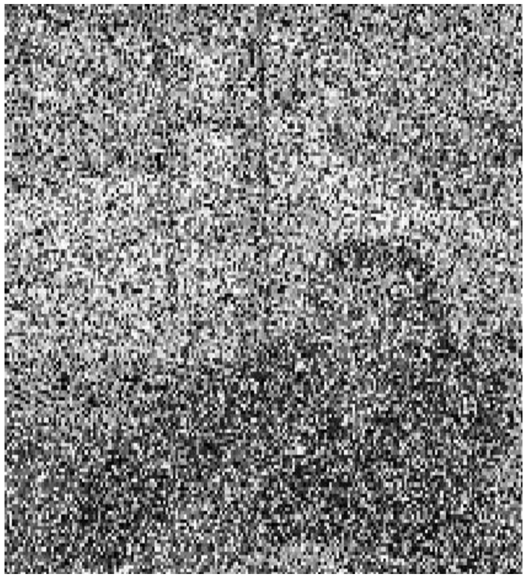


Figure 6.15: BER performance over Rayleigh fading channel for LDPC-only coding and BCH-LDPC coding over ITU-A

In Figure 6.15 the BER performance for BCH-LDPC and LDPC coding scheme is provided. Again, as for the AWGN case, for high SNR values we have a BER gain in case of BCH-LDPC coding scheme. For instance, for a target BER of 10^{-6} the corresponding BER level is 8.7 dB for BCH-LDPC and 9.5 dB for LDPC only coding scheme. We have a BER gain of about 0.8 dB. However, for low SNR values the BER level is quite similar for both coding schemes.



(a) 0 dB

(b) 3 dB

(c) 5 dB

(d) 7 dB

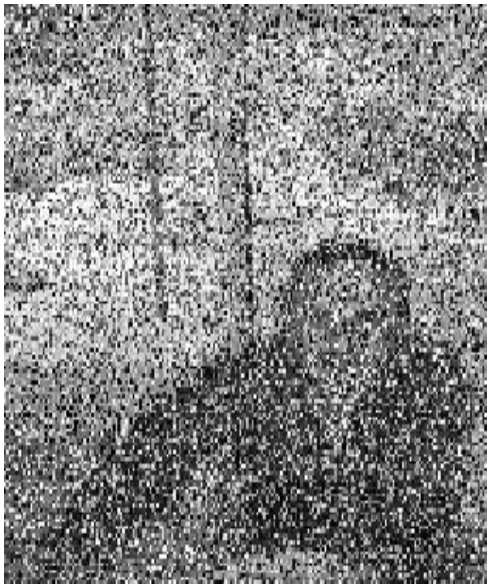
Figure 6.16: Decoded image at various SNR values for concatenated BCH-LDPC coding over the ITU Vehicular-A channel.

The psycho-visual performance of the received image at various SNR values is depicted in Figure 6.16. The results shown are from simulations carried out for rate $R = \frac{1}{4}$ BCH-LDPC over the ITU Vehicular-A channel. As can be observed, the quality of the received image progressively improves as the SNR increases. For SNR values greater than 5 dB, the received image becomes visually appealing, the background and foreground features of the image are visible and distinguishable.

6.2.3. ITU-Vehicular B

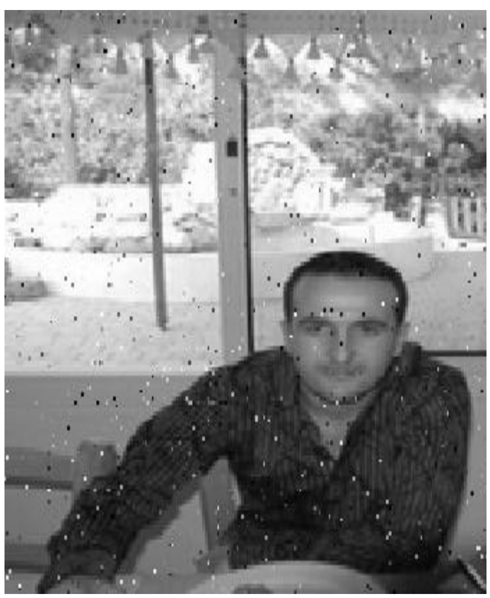
Depicted in Figure 6.17 is the performance of the received images at different SNR values. The results shown in the figure mentioned above are carried out for a code rate $R = \frac{1}{4}$ BCH-LDPC over the ITU Vehicular-B channel. For SNR level of 4 dB the received image is almost unrecognizable. Similarly given the SNR level of 6 dB the image recovered is more distinguishable, however the quality level of the image is subject to discussion. After an SNR level of 7 dB the received image is appearing much better.

As we can obviously see the noise introduced in our received images can be modeled as a salt and pepper noise. Different types of filter are capable to filter out such kind of noises with very high output performances. In Figure 6.18 the BER performance is given for BCH-LDPC with code rate $R = \frac{1}{4}$. For a SNR level of 7 dB, the corresponding BER level is approximately on the range of 10^{-4}. Probably this high level of BER value is translated to a high PSNR value.Considering the SNR value of 8 dB the corresponding BER level is roughly around 10^{-5}. However we are aware that increasing the number of bits to be transmitted through the systems will give us a more clear picture of what really will happen after an SNR level of 8 dB.



(a) 4 dB

(b) 6 dB



(c) 7dB

(d) 9 dB

Figure 6.17: Decoded image at various SNR values for concatenated BCH-LDPC coding over the ITU Vehicular-B channel.

Comparing the performance depicted in Figure 6.18 on page 77 with the performance depicted in Figure 6.14 on page 72 we can see that ITU-Vehicular B channel is a more difficult channel than ITU-Vehicular A. For instance for a SNR level of 7 dB the corresponding BER level are roughly 10^{-2} and 10^{-4} respectively.

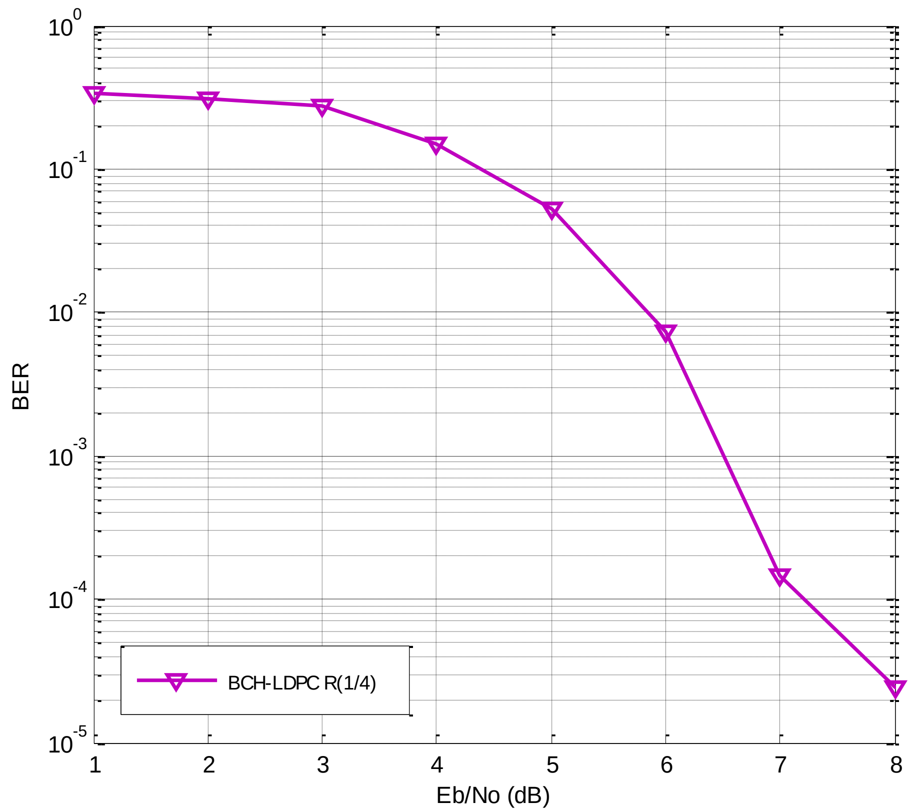


Figure 6.18: BER performance over Rayleigh fading channel using BCH-LDPC over ITU-B

In Figure 6.19 a comparative BER performance of LDPC is shown, which is carried out over the Rayleigh fading channel, ITU-Vehicular B and ITU-Vehicular A. The code rate is $R = \frac{1}{4}$ and the standard obeyed is Digital Video Broadcasting Terrestrial Second Generation (DVB-T2). As it can be seen from the Figure the ITU-Vehicular B channel is harder than ITU-Vehicular A channel. For instance refereing to the same BER level (10^{-3}) for both channels we can say that for ITU- Vehicular A this is possible for a E_b/N_0 of 6.5 dB, however for ITU-Vehicular B this BER level is only possible for an E_b/N_0 of 7.5 dB. As it was expected and as mentioned above the ITU Vehicular-B channel is harder than ITU Vehicular A-channel. A difference of 2 dB gain can be stressed out when comaring both channels.

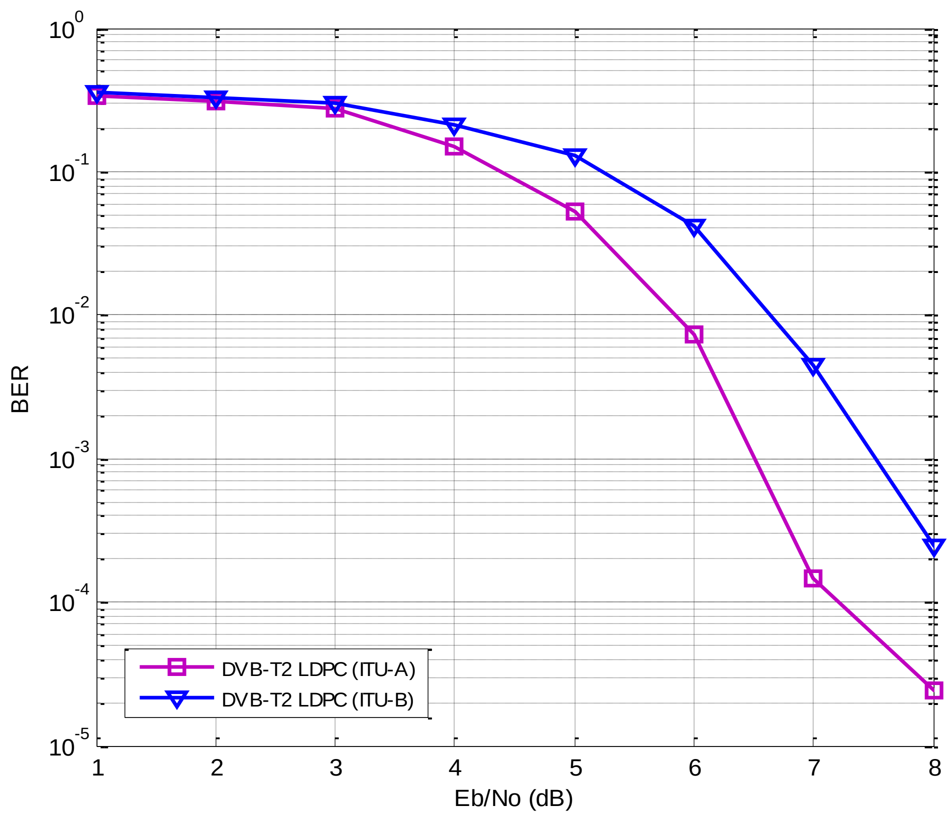
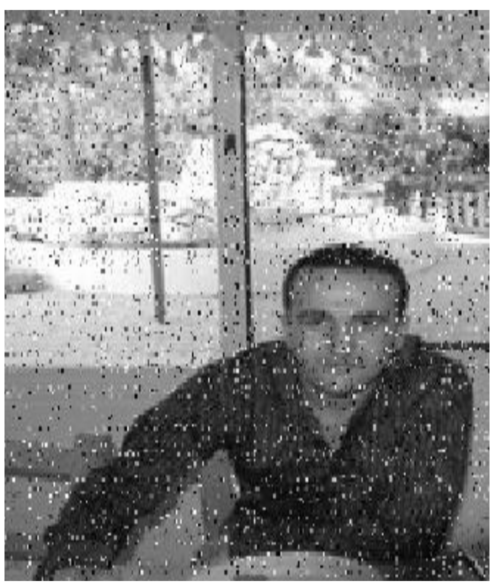


Figure 6.19: BER performance over Rayleigh fading channel using LDPC-only coding over ITU-A and ITU-B

(a) 3 dB

(b) 5 dB



(c) 6 dB

(d) 7 dB

Figure 6.20: Decoded image at various SNR values for concatenated LDPC coding over the ITU Vehicular-B channel.

Chapter 7

CONCLUSIONS AND FUTURE WORK

7.1. Conclusion

Even though the study of the complete system for DVB-S2, DVB-T2 and IEEE 802.16e standards is beyond the scope of this thesis, a detailed study and analysis of the important parts of the systems such as LDPC coding part, BCH coding, OFDM as well as polynomials for generating the short/normal FEC frame. The performance analysis provided in Chapter 6 agrees with the present publications and literature.

BER and PSNR performances of the three systems were obtained over AWGN and fading channel models (ITU- Vehicular A and ITU- Vehicular B). For AWGN channel, the best BER performance was obtained using the rate $R = 1/2$ LDPC code specified in IEEE 802.16e, where zero- error decoding becomes possible after an SNR of 1.5 dB. The second best BER is attained while using the rate $R = 1/4$ LDPC for the DVB-T2. Here zero- error decoding was shown to be possible after 3.5 dB. It has been shown that there is a coding gain of about 9 dB for a target BER of 10^{-2} when the IEEE 802.16e LDPC is used instead of the IEEE 802.16e RS(255;239;8) CC(2;1;7) concatenated coding. Clearly the usage of LDPC encoders brings a big improvement to the system's BER performance. Also it has been shown that in the case of many bit errors introduced by the channel the error floor has been removed by the concatenation of an outer BCH encoder. Similar many error correcting codes LDPC codes also have a limit for the number of errors they can fix. If the errors introduced by the channel are more than this limit an error floor would be observed. It was shown by simulation that concatenating a BCH encoder with the LDPC coding block would help to reduce or eliminate this error floor. However the maximum number of errors the BCH-LDPC concatenated coder can fix is also limited. This is because the generator polynomials are designed to fix only a maximum number of errors. In the case of DVB-T2 this number is 12. Hence if more than 12 errors per block occurs the error floor will not be removed even using BCH-LDPC encoding.

According to the results presented in Chapter 6, when BCH-LDPC coding is used in the presence of bit errors, it is possible to receive the transmitted image without any errors after an SNR value of 3.5 dB in case of AWGN channel; but when LDPC-only is used under the same conditions, a degradation in the performance is observed. This error floor might keep the PSNR of the received image at a fairly constant value, thus limiting the received image quality. Comparing the performance results for ITU- Vehicular A and ITU-Vehicular B channels, we can see that ITU-Vehicular B channel is a more difficult channel than ITU-Vehicular A. For instance a target BER level of (10^{-3}) can be attained at 5.5 dB and 7.5 dB respectively.

7.2. Future work

Facing the need for transmitting reliable data over the modern communications channel, many researchers focused in channel coding and in the features of LDPC codes. It is important to mention that great progress has been made in this area. As it is stated in this work LDPC codes performs best for long codeword length. However, need of the communication industry to shorten the length of codeword gives to the researchers another assignment. Shortening the LDPC codeword raise up the problem of so called “girth4”. Girth 4 cycles leads performance degradation and should be avoided.

As a future work designing the Low-Density Parity-Check matrix for shorter codeword lengths in order to extend the applications of LDPC channel coding is recommended. Some results on this issue has been published but a lot more remains to be done because even though the

LDPC codes designed give good performance they still do not attain the Shannon limit as explained in [5].

REFERENCES

[1] L L Peterson and B S Davie, *Computer Networks:a systems approach* (Morgan Kaufmann), 1996. ISBN: 1-55860-368-9 (Paperback ISBN: 1-55860-404-9) pp 94-95.

[2] Sae-Young Chung, G. David Forney, Jr., Thomas J. Richardson, and Rdiger Urbanke, "On the Design of Low-Density Parity-Check Codes within 0.0045 dB of the Shannon Limit", *IEEE Communication Letters*, vol. 5, pp.58-60, Feb. 2001. ISSN 1089-7798.

[3] "Standard for Local and Metropolitan Area Networks - Part 16: Air Interface for Fixed and Mob'le Broadband Wireless Access Systems - Amendment 3: Management Plane Procedures and Services (Amendment to IEEE 802.16-2004)," IEEE Approved Drafts Std P802.16g/D9, Apr 2007.

[4] C. E. Shannon, "A Mathematical Theory of Communication," *Bell System Technical Journal*, Vol. 27, pp. 379-423 (Part One), pp. 623-656 (Part Two), October 1948.

[5] J. C. MacKay and R. M. Neal, "Near Shannon Limit Performance of Low-Density Parity-Check Codes", *Electronics Letters*, vol. 32, pp 1645-1646, Aug. 1996.

[6] Jilei Hou, Paul H. Siegel and Laurence B. Milstein, "Performance Analysis and Code Optimization of Low Density Parity-Check Codes on Rayleigh Fading Channels", *IEEE Journal on Selected Areas in Communications*, vol. 19, no.5, May. 2001, pp 924-934

[7] W.C. Jakes, *Microwave Mobile Communications*, Piscataway, NJ:IEEE Press, 1994.

[8] R. M. Tanner, "A Recursive Approach to Low Complexity Codes," *IEEE Transactions on Information Theory*, Vol. IT-27, No. 5, pp. 533-547, September 1981.

[9] N. Wiberg, Codes and Decoding on General Graphs, Linköping studies in science and technology, Dissertation No. 440, Linköping University, Linköping, Sweden, 1996.

[10] C. Berrou, A. Glavieux and P. Thitimajshima, ”Near Shannon Limit Error-Correcting Coding and Decoding: Turbo Codes,” in *IEEE International Conference on Communications (ICC)*, pp. 1064-1070, 1993.

[11] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi and D. A. Spielman, “Improved Low-Density Parity-Check Codes Using Irregular Graphs and Belief Propagation,” in *Proc. of ISIT*, pp. 117, 1998.

[12] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi and D. A. Spielman, “Improved Low-Density Parity-Check Codes Using Irregular Graphs,” *IEEE Transactions on Information Theory*, VOL. 47, No. 2, pp. 585-598, February 2001.

[13] ITU-R Recommendation M.1225, “Guidelines for evaluation of radio transmission technologies for IMT-2000, 1997.

[14] Wireless Communication. Retrieved June 15, 2011, from http://www.wirelesscommunication.nl/reference/chaptr05/digimod/awgn.htm.

[15] Rayleigh Fading. Retrieved February 10, 2011, from http://en.wikipedia.org/wiki/Rayleigh˙fading.

[16] BCH code. Retrieved May 5, 2011, from http://en.wikipedia.org/wiki/BCH˙code.

[17] ETSI EN 302 755 V1.1.1(2009-09): Digital Video Broadcasting (DVB): Frame structure channel coding and modulation for a second generation digital terrestrial television broadcasting system (DVB-T2), Sep 2009.

[18] ETSI EN 300 421 V1.1.2(1997-08): Digital Video Broadcasting (DVB): Frame structure channel coding and modulation for a 11/12 GHz satellite services, Sep 1997.

[19] Digital Video Broadcasting, “Frame structure channel coding and modulation for a second generation digital terrestrial television broadcasting system (DVB-T2),” DVB Document A122, June 2008.

[20] Digital video broadcasting (DVB); Framing structure, channel coding and modulation for 11/12 GHz satellite services, EN 300 421 (V1.1.2), European Telecommunications Standards Institute (ETSI).

[21] DVB: Framing structure, channel coding and modulation for DSNG and other contribution applications by satellite, EN 301 210, European Telecommunications Standards Institute (ETSI).

[22] Digital video broadcasting (DVB); Second generation framing structure, channel coding and modulation systems for broadcasting, interactive services, news gathering and other broad-band satellite applications, EN 302 307, European Telecommunications Standards Institute (ETSI).

[23] “Digital video broadcasting (DVB); Implementation guidelines for the use of MPEG-2 systems, video and audio in satellite, cable and terrestrial broadcasting applications,” European Telecommunications Standards Institute (ETSI), TR 101 154.

[24] Bernard Sklar (July 1997). “Rayleigh Fading Channels in Mobile Digital Communication Systems Part I: Characterization”, *IEEE Communications Magazine 35*(7): 90-100.

[25] C.E.Shannon, *“A Mathematical Theory of Communication ” , Bell System Technical Journal*, vol. 27, pp. 397-423 and 623-656, July/Oct, 1948.

[26] Reimers, Ulrich: A Guideline for the Use of DVB Specifications and Standards, DVB-Document TM 1694, rev.1, September 1996.

[27] T.J. Richardson, M.A. Shokrollahi, and R.L. Urbanke, "Design of Capacity-approaching Low-density Parity-check Codes," *IEEE Trans. on Info. Theory, Vol*. 47, pp.619-637, Feb. 2001.

[28] "Part 16: Air Interface for Fixed Broadband Wireless Access Systems Amendment for Physical and Medium Access Control Layers for Combined Fixed Operation in Licensed Bands",*IEEE P802.16-2004*, October 2004.

[29] R. M. Tanner, D. Sridhara, A. Sridharan, T. E. Fuja, and D. J. Costello, "LDPC block and convolutional codes based on circulant Matrices," *IEEE Trans. Info. Theory,* vol. 50, pp. 2966-2984, Dec. 2004.

[30] H. Tang, J. Xu, Y. Kou, S. Lin, and K. Abdel-Ghaffar, "On algebraic construction of Gallager and circulant low density parity-check codes," *IEEE Trans. Inf. Theory,* vol. 50, no. 6, pp. 1269-1279, Jun. 2004.

[31] M. Fossorier, "Quasi-cyclic low density parity check codes from circulant permutation matrices," *IEEE Trans. Inf. Theory,* no. 8, pp. 1788-1793, Aug. 2004.

[32] T. J. Richardson, M. A. Shokrollahi, and R. L. Urbanke, "Design of capacity-approaching irregular low density parity-check codes," *IEEE Trans. Inf. Theory,* vol. 47, no. 2, pp. 619-637, Feb. 2001.

[33] R. G. Gallager, "Low density parity-check codes," *IRE Trans. Inf. Theory,* vol. IT-8, no. 1, pp. 21-28, Jan. 1962.

[34] Y. Kou, S. Lin, and M. Fossorier, "Low density parity-check codes based on finite geometries: A discovery and new results," *IEEE Trans. Inf. Theory,* vol. 47, no. 11, pp. 2711-2736, Nov. 2001.

[35] R. M. Tanner, "Spectral graphs for quasi-cyclic LDPC codes," *in Proc. IEEE Int. Symp. Inf. Theory, Washington,* DC, Jun. 2001, p. 226.

[36] L. Chen, J. Xu, I. Djurdjevic, and S. Lin, "Near-Shannon-limit quasicyclic low-density parity-check codes," *IEEE Trans. Commun.*, vol. 52, no. 7, pp. 1038-1042, Jul. 2004

[37] S. Lin, S. Song, L. Lan, L. Zeng and Y. Y. Tai, Construction of Nonbiary Quasi-Cyclic LDPC Codes: A finite Field Approach. Retrieved October 12, 2011, from http://www.itu.int/home/imt.html.

[38] Z. Li, L. Chen, L. Zeng, S.Lin, and W. H. Fong, "Efficient Encoding of Quasi-Cyclic Low-Density Parity-Check Codes", *IEEE Transactions on Communications,* vol. 54, no. 1, Jan. 2006, pp. 71-81.

[39] S. Myung, K. Yang, and J. Kim, "Quasi-Cyclic LDPC Codes for Fast Encoding", *IEEE Transactions on Information Theory,* vol. 51, no. 8, pp. 2894 - 2901, Aug. 2005.

[40] E. Eleftheriou and S. Olcer, "Low-density parity-check codes for multilevel modulation," in Proc. *IEEE Int. Symp. Information Theory (ISIT2002)*, Lausanne, Switzerland, Jun./Jul. 2002, p. 442.

[41] R. W. Chang, "Orthogonality Frequency Division Multiplexing", U.S. Patent 3,488,455, field 1966, issued Jan. 1970.

[42] L. J. Deutsch, "The Effects of Reed-Solomon Code Shortening on the Performance of Coded Telemetry Systems", TDA Progress Report 42-75, Jul-Sept 1983.

[43] E. A. Ince, E. Nurellari and L. O. Iheme, "Image Transmission Over Fading Channels Using RS-CC Versus LDPC Coding," *The thirteenth International Conference on Signal and Image processing (SIP 2011),* Dallas, USA, December 14 - 16, 2011.

[44] ITU-R Recommendation M.1225, "Guidelines for evaluation of radio transmission technologies for IMT-2000," 1997.

[45] H. Jin, A. Khandekar, and R. McEliece, "Irregular repeat-accumulate codes," In. Proc. *2nd International Symposium on Turbo Codes* & *Related Topics, Brest*, France, Sept 2000.

[46] M. Eroz, F. W. Sun, and L. N. Lee, "DVB-S2 low density parity check codes with near Shannon limit performance," *International Journal of Satellite Communications and Networking*, vol. 22, pp. 269-279, 2004.

[47] S. W. Kim, C. S. Park, S. Y. Hwang, "Design of High-throughput LDPC Decoder for DVB-S2 Using Local Memory Banks," *IEEE Transcations on Consumer Electronics*, vol. 55, No.3, Aug 2009.

[48] M. Engels., *Wireless OFDM Systems: How to make them work?*, Kluwer Academic Publishers, 2002. 35, 37

[49] T. S. Rappaport, *Wireless Communications Principles & Practice*, Prentice Hall, 1996.

[50] A. Papoulis and S. U. Pillai, *Probability, Random Variables and Stochastic Processes*, McGraw Hill, Fourth Edition, 2002.

[51] J. G. Proakis, *Digital Communications*, Fourth Edition, McGraw Hill, 2000.

[52] C. E. Shannon, "Probability of error for optinmal codes in Gaussian channel," *Bell System Technical Journal*, Vol. 38, pp. 611-656, 1959.

APPENDIX

Appendix A: Addresses of parity bit accumulators

Addresses of parity bit accumulators for rate $R = 2/5$, $n_{ldpc} = 64800\ bits$

$$
c_1(t) = \begin{bmatrix}
31413 & 18834 & 28884 & 947 & 23050 & 14484 & 14809 & 4968 & 455 & 33659 & 16666 & 19008 \\
13172 & 19939 & 13354 & 13719 & 6132 & 20086 & 34040 & 13442 & 27958 & 16813 & 29619 & 16553 \\
1499 & 32075 & 14962 & 11578 & 112049 & 9217 & 10485 & 23062 & 30936 & 17892 & 24204 & 24885 \\
32490 & 18086 & 18007 & 4957 & 7285 & 32073 & 19038 & 7152 & 12486 & 13483 & 24808 & 21759 \\
32321 & 10839 & 15620 & 33521 & 23030 & 10646 & 26236 & 19744 & 21713 & 36784 & 8016 & 12869 \\
35597 & 11129 & 17948 & 26160 & 14729 & 31943 & 20416 & 10000 & 7882 & 31380 & 27858 & 33356 \\
14125 & 12131 & 36199 & 4058 & 35992 & 36594 & 33698 & 15475 & 1566 & 18498 & 12725 & 7067 \\
17406 & 8372 & 35437 & 2888 & 1184 & 30068 & 25802 & 11056 & 5507 & 26313 & 32205 & 37232 \\
15254 & 5365 & 17308 & 22519 & 35009 & 718 & 5240 & 16778 & 23131 & 24092 & 20587 & 33385 \\
27455 & 17602 & 4590 & 21767 & 22266 & 27357 & 30400 & 8732 & 5596 & 3060 & 33703 & 3596 \\
6882 & 873 & 10997 & 24738 & 20770 & 10067 & 13379 & 27409 & 25463 & 2673 & 6998 & 31378 \\
15181 & 13645 & 34501 & 3393 & 3840 & 35227 & 15562 & 23615 & 38342 & 12139 & 19471 & 15483 \\
13350 & 6707 & 23709 & 37204 & 25778 & 21082 & 7511 & 14588 & 10010 & 21854 & 28375 & 33591 \\
12514 & 4695 & 37190 & 21379 & 18723 & 5802 & 7182 & 2529 & 29936 & 35860 & 28338 & 10835 \\
34283 & 25610 & 33026 & 31017 & 21259 & 2165 & 21807 & 37578 & 1175 & 16710 & 21939 & 30841 \\
27292 & 33730 & 6836 & 26476 & 27539 & 35784 & 18245 & 16394 & 17939 & 23094 & 19216 & 17432 \\
11655 & 6183 & 38708 & 28408 & 35157 & 17089 & 13998 & 36029 & 15052 & 16617 & 5638 & 36464 \\
15693 & 28923 & 26245 & 9432 & 11675 & 25720 & 26405 & 5838 & 31851 & 26898 & 8090 & 37037 \\
24418 & 27583 & 7959 & 35562 & 37771 & 17784 & 11382 & 11156 & 37855 & 7073 & 21685 & 34515 \\
10977 & 13633 & 30969 & 7516 & 11943 & 18199 & 5231 & 13825 & 19589 & 23661 & 11150 & 35602 \\
19124 & 30774 & 6670 & 37344 & 16510 & 26317 & 23518 & 22957 & 6348 & 34069 & 8845 & 20175 \\
34985 & 14441 & 25668 & 4116 & 3019 & 21049 & 37308 & 24551 & 24727 & 20104 & 24850 & 12114 \\
38187 & 28527 & 13108 & 13985 & 1425 & 21477 & 30807 & 8613 & 26241 & 33368 & 35913 & 32477 \\
5903 & 34390 & 24641 & 26556 & 23007 & 27305 & 38247 & 2621 & 9122 & 32806 & 21554 & 18685
\end{bmatrix} \tag{7.1}
$$

$$
c_2(t) = \begin{bmatrix}
17287 & 27292 & 19033 \\
25796 & 31795 & 12152 \\
12184 & 35088 & 31226 \\
38263 & 33386 & 24892 \\
23114 & 37995 & 29796 \\
34336 & 10551 & 36245 \\
35407 & 175 & 7203 \\
14654 & 38201 & 22605 \\
28404 & 6595 & 1018 \\
19932 & 3524 & 29305 \\
31749 & 20247 & 8128 \\
18026 & 36357 & 26735 \\
7543 & 29767 & 13588 \\
13333 & 25965 & 8463 \\
14504 & 36796 & 19710 \\
4528 & 25299 & 7318 \\
35091 & 25550 & 14798 \\
7824 & 215 & 1248 \\
30848 & 5362 & 17291 \\
28932 & 30249 & 27073 \\
13062 & 2103 & 16206 \\
7129 & 32062 & 19612 \\
9512 & 21936 & 38833 \\
35849 & 33754 & 23450 \\
18705 & 28656 & 18111 \\
22749 & 27456 & 32187 \\
\vdots & \vdots & \vdots
\end{bmatrix}
\begin{bmatrix}
\vdots & \vdots & \vdots \\
28229 & 31684 & 30160 \\
15293 & 8483 & 28002 \\
14880 & 13334 & 12584 \\
28646 & 2558 & 19687 \\
6259 & 4499 & 26336 \\
11952 & 28386 & 8405 \\
10609 & 961 & 7582 \\
10423 & 13191 & 26818 \\
15922 & 36654 & 21450 \\
10492 & 1532 & 1205 \\
30551 & 36482 & 22153 \\
5156 & 11330 & 34243 \\
28616 & 35369 & 13322 \\
8962 & 1485 & 21186 \\
23541 & 17445 & 35561 \\
33133 & 11593 & 19895 \\
33917 & 7863 & 33651 \\
20063 & 28331 & 10702 \\
13195 & 21107 & 21859 \\
4364 & 31137 & 4804 \\
5585 & 2037 & 4830 \\
30672 & 16927 & 14800
\end{bmatrix} \tag{7.2}
$$

Addresses of parity bit accumulators for rate $R = 3/5$, $n_{ldpc} = 64800\ bits$

$$c_1(t) = \begin{bmatrix}
22422 & 10282 & 11626 & 19997 & 11161 & 2922 & 3122 & 99 & 5625 & 17064 & 8270 & 179 \\
25087 & 16218 & 17015 & 828 & 20041 & 25656 & 4186 & 11629 & 22599 & 17305 & 22515 & 6463 \\
11049 & 22853 & 25706 & 14388 & 5500 & 19245 & 8732 & 2177 & 13555 & 11346 & 17265 & 3069 \\
16581 & 22225 & 12563 & 19717 & 23577 & 11555 & 25496 & 6853 & 25403 & 5218 & 15925 & 21766 \\
16529 & 14487 & 7643 & 10715 & 17442 & 11119 & 5679 & 14155 & 24213 & 21000 & 1116 & 15620 \\
5340 & 8636 & 16693 & 1434 & 5635 & 6516 & 9482 & 20189 & 1066 & 15013 & 25361 & 14243 \\
18506 & 22236 & 20912 & 8952 & 5421 & 15691 & 6126 & 21595 & 500 & 6904 & 13059 & 6802 \\
8433 & 4694 & 5524 & 14216 & 3685 & 19721 & 25420 & 9937 & 23813 & 9047 & 25651 & 16826 \\
21500 & 24814 & 6344 & 17382 & 7064 & 13929 & 4004 & 16552 & 12818 & 8720 & 5286 & 2206 \\
22517 & 2429 & 19065 & 2921 & 21611 & 1873 & 7507 & 5661 & 23006 & 23128 & 20543 & 19777 \\
1770 & 4636 & 20900 & 14931 & 9247 & 12340 & 11008 & 12966 & 4471 & 2731 & 16445 & 791 \\
6635 & 14556 & 18865 & 22421 & 22124 & 12697 & 9803 & 25485 & 7744 & 18254 & 11313 & 9004 \\
19982 & 23963 & 18912 & 7206 & 12500 & 4382 & 20067 & 6177 & 21007 & 1195 & 23547 & 24837 \\
756 & 11158 & 14646 & 20534 & 3647 & 17728 & 11676 & 11843 & 12937 & 4402 & 8261 & 22944 \\
9306 & 24009 & 10012 & 11081 & 3746 & 24325 & 8060 & 19826 & 842 & 8836 & 2898 & 5019 \\
7575 & 7455 & 25244 & 4736 & 14400 & 22981 & 5543 & 8006 & 24203 & 13053 & 1120 & 5128 \\
3482 & 9270 & 13059 & 15825 & 7453 & 23747 & 3656 & 24585 & 16542 & 17507 & 22462 & 14670 \\
15627 & 15290 & 4198 & 22748 & 5842 & 13395 & 23918 & 16985 & 14929 & 3726 & 25350 & 24157 \\
24896 & 16365 & 16423 & 13461 & 16615 & 8107 & 24741 & 3604 & 25904 & 8716 & 9604 & 20365 \\
3729 & 17245 & 18448 & 9862 & 20831 & 25326 & 20517 & 24618 & 13282 & 5099 & 14183 & 8804 \\
16455 & 17646 & 15376 & 18194 & 25528 & 1777 & 6066 & 21855 & 14372 & 12517 & 4488 & 17490 \\
1400 & 8135 & 23375 & 20879 & 8476 & 4084 & 12936 & 25536 & 22309 & 16582 & 6402 & 24360 \\
25119 & 23586 & 128 & 4761 & 10443 & 22536 & 8607 & 9752 & 25446 & 15053 & 1856 & 4040 \\
377 & 21160 & 13474 & 5451 & 17170 & 5938 & 10256 & 11972 & 24210 & 17833 & 22047 & 16108 \\
13075 & 9648 & 24546 & 13150 & 23867 & 7309 & 19798 & 2988 & 16858 & 4825 & 23950 & 5125 \\
20526 & 3553 & 11525 & 23366 & 2452 & 17626 & 19265 & 20172 & 18060 & 24593 & 13255 & 1552 \\
18839 & 21132 & 20119 & 15214 & 14705 & 7096 & 10174 & 5663 & 18651 & 19700 & 12524 & 14033 \\
4127 & 2971 & 17499 & 16287 & 22368 & 21463 & 7943 & 18880 & 5567 & 8047 & 23363 & 6797 \\
10651 & 24471 & 14325 & 4081 & 7258 & 4949 & 7044 & 1078 & 797 & 22910 & 20474 & 4318 \\
21374 & 13231 & 22985 & 5056 & 3821 & 23718 & 14178 & 9978 & 19030 & 23594 & 8895 & 25358 \\
6199 & 22056 & 7749 & 13310 & 3999 & 23697 & 16445 & 22636 & 5225 & 22437 & 24153 & 9442 \\
7978 & 12177 & 2893 & 20778 & 3175 & 8645 & 11863 & 24623 & 10311 & 25767 & 17057 & 3691 \\
20473 & 11294 & 9914 & 22815 & 2574 & 8439 & 3699 & 5431 & 24840 & 21908 & 16088 & 18244 \\
8208 & 5755 & 19059 & 8541 & 24924 & 6454 & 11234 & 10492 & 16406 & 10831 & 11436 & 9649 \\
16264 & 11275 & 24953 & 2347 & 12667 & 19190 & 7257 & 7174 & 24819 & 2938 & 2522 & 11749 \\
3627 & 5969 & 13862 & 1538 & 23176 & 6353 & 2855 & 17720 & 2472 & 7428 & 573 & 15036
\end{bmatrix} \tag{7.3}$$

$$c_2(t) = \begin{bmatrix} 0 & 18539 & 18661 \\ 1 & 10502 & 3002 \\ 2 & 9368 & 10761 \\ 3 & 12299 & 7828 \\ 4 & 15048 & 13362 \\ 5 & 18444 & 24640 \\ 6 & 20775 & 19175 \\ 7 & 18970 & 10971 \\ 8 & 5329 & 19982 \\ 9 & 11296 & 18655 \\ 10 & 15046 & 20659 \\ 11 & 7300 & 22140 \\ 12 & 22029 & 14477 \\ 13 & 11129 & 742 \\ 14 & 13254 & 13813 \\ 15 & 19234 & 13273 \\ 16 & 6079 & 21122 \\ 17 & 22782 & 5828 \\ 18 & 19775 & 4247 \\ 19 & 1660 & 19413 \\ 20 & 4403 & 3649 \\ 21 & 13371 & 25851 \\ 22 & 22770 & 21784 \\ 23 & 10757 & 14131 \\ 24 & 16071 & 21617 \\ 25 & 6393 & 3725 \\ 26 & 597 & 19968 \\ 27 & 5743 & 8084 \\ 28 & 6770 & 9548 \\ 29 & 4285 & 17542 \\ 30 & 13568 & 22599 \\ 31 & 1786 & 4617 \\ 32 & 23238 & 11648 \\ 33 & 19627 & 2030 \\ 34 & 13601 & 13458 \\ 35 & 13740 & 17328 \\ 36 & 25012 & 13944 \\ 37 & 22513 & 6687 \\ \vdots & \vdots & \vdots \end{bmatrix} \begin{bmatrix} \vdots & \vdots & \vdots \\ 38 & 4934 & 125872 \\ 39 & 21197 & 5133 \\ 40 & 22705 & 6938 \\ 41 & 7534 & 24633 \\ 42 & 24400 & 12797 \\ 43 & 21911 & 25712 \\ 44 & 12039 & 1140 \\ 45 & 24306 & 1021 \\ 46 & 14012 & 20747 \\ 47 & 11265 & 15219 \\ 48 & 4670 & 15531 \\ 49 & 9417 & 14359 \\ 50 & 2415 & 6504 \\ 51 & 24964 & 24690 \\ 52 & 14443 & 8816 \\ 53 & 6926 & 1291 \\ 54 & 6209 & 20806 \\ 55 & 13915 & 4079 \\ 56 & 24410 & 13196 \\ 57 & 13505 & 6117 \\ 58 & 9869 & 8220 \\ 59 & 1570 & 6044 \\ 60 & 25780 & 17387 \\ 61 & 20671 & 24913 \\ 62 & 24558 & 20591 \\ 63 & 12402 & 3702 \\ 64 & 8314 & 1357 \\ 65 & 20071 & 14616 \\ 66 & 17014 & 3688 \\ 67 & 19837 & 946 \\ 68 & 15195 & 12136 \\ 69 & 7758 & 22808 \\ 70 & 3564 & 2925 \\ 71 & 3434 & 7769 \end{bmatrix} \tag{7.4}$$

Addresses of parity bit accumulators for rate $R = 2/3$, $n_{ldpc} = 64800$ *bits*

$$
c_1(t) = \begin{bmatrix}
0 & 10491 & 16043 & 506 & 12826 & 8065 & 8226 & 2767 & 240 & 18673 & 9279 & 10579 & 20928 \\
1 & 17819 & 8313 & 6433 & 6224 & 5120 & 5824 & 12812 & 17187 & 9940 & 13447 & 13825 & 18483 \\
2 & 17957 & 6024 & 8681 & 18628 & 12794 & 5915 & 14576 & 10970 & 12064 & 20437 & 4455 & 7151 \\
3 & 19777 & 6183 & 9972 & 14536 & 8182 & 17749 & 11341 & 5556 & 4379 & 17434 & 15477 & 18532 \\
4 & 4651 & 19689 & 1608 & 659 & 16707 & 14335 & 6143 & 3058 & 14618 & 17894 & 20684 & 5306 \\
5 & 9778 & 2552 & 12096 & 12369 & 15198 & 16890 & 4851 & 3109 & 1700 & 18725 & 1997 & 15882 \\
6 & 486 & 6111 & 13743 & 11537 & 5591 & 7433 & 15227 & 14145 & 1483 & 3887 & 17431 & 12430 \\
7 & 20647 & 14311 & 11734 & 4180 & 8110 & 5525 & 12141 & 15761 & 18661 & 18441 & 10569 & 8192 \\
8 & 3791 & 14759 & 15264 & 19918 & 10132 & 9062 & 10010 & 12786 & 10675 & 9682 & 19246 & 5454 \\
9 & 19525 & 9485 & 7777 & 19999 & 8378 & 9209 & 3163 & 20232 & 6690 & 16518 & 716 & 7353 \\
10 & 4588 & 6709 & 20202 & 10905 & 915 & 4317 & 11073 & 13576 & 16433 & 368 & 3508 & 21171 \\
11 & 14072 & 4033 & 19959 & 12608 & 631 & 19494 & 14160 & 8249 & 10223 & 21504 & 12395 & 4322
\end{bmatrix} \tag{7.5}
$$

$$
c_2(t) = \begin{bmatrix}
12 & 13800 & 14161 \\
13 & 2948 & 9647 \\
14 & 14693 & 16027 \\
15 & 20506 & 11082 \\
16 & 1143 & 9020 \\
17 & 13501 & 4014 \\
18 & 1548 & 2190 \\
19 & 12216 & 21556 \\
20 & 2095 & 19897 \\
21 & 4189 & 7958 \\
22 & 15940 & 10048 \\
23 & 515 & 12614 \\
24 & 8501 & 8450 \\
25 & 17595 & 16784 \\
26 & 5913 & 8495 \\
27 & 16394 & 10423 \\
28 & 7409 & 6981 \\
29 & 6678 & 15939 \\
30 & 20344 & 12987 \\
31 & 2510 & 14588 \\
32 & 17918 & 6655 \\
33 & 6703 & 19451 \\
34 & 496 & 4217 \\
35 & 7290 & 5766 \\
36 & 10521 & 8925 \\
37 & 20379 & 11905 \\
38 & 4090 & 5838 \\
39 & 19082 & 17040 \\
40 & 20233 & 12352 \\
41 & 19365 & 19546 \\
42 & 6249 & 19030 \\
43 & 11037 & 19193 \\
44 & 19760 & 11772 \\
45 & 19644 & 7428 \\
46 & 16076 & 3521 \\
47 & 11779 & 21062 \\
48 & 13062 & 9682 \\
49 & 8934 & 5217 \\
\vdots & \vdots & \vdots
\end{bmatrix}
\begin{bmatrix}
\vdots & \vdots & \vdots \\
50 & 11087 & 3319 \\
51 & 18892 & 4356 \\
52 & 7894 & 3898 \\
53 & 5963 & 4360 \\
54 & 7346 & 11726 \\
55 & 5182 & 5609 \\
56 & 2412 & 17295 \\
57 & 9845 & 20494 \\
58 & 6687 & 1864 \\
59 & 20564 & 5216 \\
0 & 18226 & 17206 \\
1 & 9380 & 8266 \\
2 & 7073 & 3065 \\
3 & 18252 & 13437 \\
4 & 9161 & 15642 \\
5 & 10714 & 10153 \\
6 & 11585 & 9078 \\
7 & 5359 & 9418 \\
8 & 9024 & 9515 \\
9 & 1206 & 16354 \\
10 & 14994 & 1102 \\
11 & 9375 & 20796 \\
12 & 15964 & 6027 \\
13 & 14789 & 6452 \\
14 & 8002 & 18591 \\
15 & 14742 & 14089 \\
16 & 253 & 3045 \\
\vdots & \vdots & \vdots
\end{bmatrix}
\begin{bmatrix}
\vdots & \vdots & \vdots \\
17 & 1274 & 19286 \\
18 & 14777 & 2044 \\
19 & 13920 & 9900 \\
20 & 452 & 7374 \\
21 & 18206 & 9921 \\
22 & 6131 & 5414 \\
23 & 10077 & 9726 \\
24 & 12045 & 5479 \\
25 & 4322 & 7990 \\
26 & 15616 & 5550 \\
27 & 15561 & 10661 \\
28 & 20718 & 7387 \\
29 & 2518 & 18804 \\
30 & 8984 & 2600 \\
31 & 6516 & 17909 \\
32 & 11148 & 98 \\
33 & 20559 & 3704 \\
34 & 7510 & 1569 \\
35 & 16000 & 11692 \\
36 & 9147 & 10303 \\
37 & 16650 & 191 \\
38 & 15577 & 18685 \\
39 & 17167 & 20917 \\
40 & 4256 & 3391 \\
41 & 20092 & 17219 \\
42 & 9218 & 5056 \\
43 & 18429 & 8472 \\
44 & 12093 & 20753 \\
45 & 16345 & 12748 \\
46 & 16023 & 11095 \\
47 & 5048 & 17595 \\
48 & 18995 & 4817 \\
49 & 16483 & 3536 \\
50 & 1439 & 16148 \\
51 & 3661 & 3039 \\
52 & 19010 & 18121 \\
53 & 8968 & 11793 \\
54 & 13427 & 18003 \\
55 & 5303 & 3083 \\
56 & 531 & 16668 \\
57 & 4771 & 6722 \\
58 & 5695 & 7960 \\
59 & 3589 & 14630
\end{bmatrix}
\tag{7.6}
$$

Addresses of parity bit accumulators for rate $R = 3/4$, $n_{ldpc} = 64800\ bits$

$$c_1(t) = \begin{bmatrix} 0 & 6385 & 7901 & 14611 & 13389 & 11200 & 3252 & 5243 & 2504 & 2722 & 821 & 7374 \\ 1 & 11359 & 2698 & 357 & 13824 & 12772 & 7244 & 6752 & 15310 & 852 & 2001 & 11417 \\ 2 & 7862 & 7977 & 6321 & 13612 & 12197 & 14449 & 15137 & 13860 & 1708 & 6399 & 13444 \\ 3 & 1560 & 11804 & 6975 & 13292 & 3646 & 3812 & 8772 & 7306 & 5795 & 14327 & 7866 \\ 4 & 7626 & 11407 & 14599 & 9689 & 1628 & 2113 & 10809 & 9283 & 1230 & 15241 & 4870 \\ 5 & 1610 & 5699 & 15876 & 9446 & 12515 & 1400 & 6303 & 5411 & 14181 & 13925 & 7358 \\ 6 & 4059 & 8836 & 3405 & 7853 & 7992 & 15336 & 5970 & 10368 & 10278 & 9675 & 4651 \\ 7 & 441 & 3963 & 9153 & 2109 & 12683 & 7459 & 12030 & 12221 & 629 & 15212 & 406 \\ 8 & 6007 & 8411 & 5771 & 3497 & 543 & 14202 & 875 & 9186 & 6235 & 13908 & 3563 \\ 9 & 3232 & 6625 & 4795 & 546 & 9781 & 2071 & 7312 & 3399 & 7250 & 4932 & 12652 \\ 10 & 8820 & 10088 & 11090 & 7069 & 6585 & 13134 & 10158 & 7183 & 488 & 7455 & 9238 \\ 11 & 1903 & 10818 & 119 & 215 & 7558 & 11046 & 10615 & 11545 & 14784 & 7961 & 15619 \\ 12 & 3655 & 8736 & 4917 & 15874 & 5129 & 2134 & 15944 & 14768 & 7150 & 2692 & 1469 \\ 13 & 9316 & 3820 & 505 & 8923 & 6757 & 806 & 7957 & 4216 & 15589 & 13244 & 2622 \\ 14 & 14463 & 4852 & 15733 & 3041 & 11193 & 12860 & 13673 & 8152 & 6551 & 15108 & 8758 \end{bmatrix} \tag{7.7}$$

$$c_2(t) = \begin{bmatrix} 15 & 3149 & 11981 \\ 16 & 13416 & 6906 \\ 17 & 13098 & 13352 \\ 18 & 2009 & 14460 \\ 19 & 7207 & 4314 \\ 20 & 3312 & 3945 \\ 21 & 4418 & 6248 \\ 22 & 2669 & 139754 \\ 23 & 7571 & 9023 \\ 24 & 14172 & 2967 \\ 25 & 7271 & 7138 \\ 26 & 6135 & 13670 \\ 27 & 7490 & 6981 \\ 28 & 8657 & 2466 \\ 29 & 8599 & 12834 \\ 30 & 3470 & 3152 \\ 31 & 13917 & 4365 \\ 32 & 6024 & 13730 \\ 33 & 10973 & 14182 \\ 34 & 2464 & 13167 \\ 35 & 5281 & 15049 \\ 36 & 1103 & 1849 \\ 37 & 2058 & 1069 \\ 38 & 9654 & 6095 \\ 39 & 14311 & 7667 \\ 40 & 15617 & 8146 \\ 41 & 4588 & 11218 \\ 42 & 13660 & 6243 \\ 43 & 8578 & 7874 \\ 44 & 11741 & 2686 \\ 0 & 1022 & 1264 \\ 1 & 12604 & 9965 \\ 2 & 8217 & 2707 \\ 3 & 3156 & 11793 \\ 4 & 354 & 1514 \\ 5 & 6978 & 14058 \\ 6 & 7922 & 16079 \\ 7 & 15087 & 12138 \\ 8 & 5053 & 6470 \\ 9 & 12687 & 14932 \\ 10 & 15458 & 1763 \\ 11 & 8121 & 1721 \\ 12 & 12431 & 549 \\ \vdots & \vdots & \vdots \end{bmatrix} \begin{bmatrix} \vdots & \vdots & \vdots \\ 13 & 4129 & 7091 \\ 14 & 1426 & 8415 \\ 15 & 9783 & 7604 \\ 16 & 6295 & 11329 \\ 17 & 1409 & 12061 \\ 18 & 8065 & 9087 \\ 19 & 2918 & 8438 \\ 20 & 1293 & 14115 \\ 21 & 3922 & 13851 \\ 22 & 3851 & 4000 \\ 23 & 5865 & 1768 \\ 24 & 2655 & 14957 \\ 25 & 5565 & 6332 \\ 26 & 4303 & 12631 \\ 27 & 11653 & 12236 \\ 28 & 16025 & 7632 \\ 29 & 4655 & 14128 \\ 30 & 9584 & 13123 \\ 31 & 13987 & 9597 \\ 32 & 15409 & 12110 \\ 33 & 8754 & 15490 \\ 34 & 7416 & 15325 \\ 35 & 2909 & 15549 \\ 36 & 2995 & 8257 \\ 37 & 9406 & 4791 \\ 38 & 11111 & 4854 \\ 39 & 2812 & 8521 \\ 40 & 8476 & 14717 \\ 41 & 7820 & 15360 \\ 42 & 1179 & 7939 \\ 43 & 2357 & 8678 \\ 0 & 3477 & 7067 \\ 1 & 3931 & 13845 \\ 2 & 7675 & 12899 \\ 3 & 1754 & 8187 \\ 4 & 7785 & 1400 \\ 5 & 9213 & 5891 \\ 6 & 2494 & 7703 \\ 7 & 2576 & 7902 \\ 8 & 4821 & 15682 \\ 9 & 10426 & 11935 \\ \vdots & \vdots & \vdots \end{bmatrix} \begin{bmatrix} \vdots & \vdots & \vdots \\ 10 & 1810 & 904 \\ 11 & 11332 & 9264 \\ 12 & 11312 & 3570 \\ 13 & 14916 & 2650 \\ 14 & 7679 & 7842 \\ 15 & 6089 & 13084 \\ 16 & 3938 & 2751 \\ 17 & 8509 & 4648 \\ 18 & 12204 & 8917 \\ 19 & 5749 & 12433 \\ 20 & 12613 & 4431 \\ 21 & 1344 & 4014 \\ 22 & 8488 & 13850 \\ 23 & 1730 & 14896 \\ 24 & 14942 & 7126 \\ 25 & 14983 & 8863 \\ 26 & 6578 & 8564 \\ 27 & 4947 & 396 \\ 28 & 297 & 12805 \\ 29 & 13878 & 6692 \\ 30 & 11857 & 11186 \\ 31 & 14395 & 11493 \\ 32 & 16145 & 12251 \\ 33 & 13462 & 7428 \\ 34 & 14526 & 13119 \\ 35 & 2535 & 11243 \\ 36 & 6465 & 12690 \\ 37 & 6872 & 9334 \\ 38 & 15371 & 14023 \\ 39 & 8101 & 10187 \\ 40 & 11963 & 4848 \\ 41 & 15125 & 6119 \\ 42 & 8051 & 14465 \\ 43 & 11139 & 5167 \\ 42 & 2883 & 14521 \end{bmatrix} \tag{7.8}$$

Addresses of parity bit accumulators for rate $R = 4/5$, $n_{ldpc} = 64800\ bits$

$$c_1(t) = \begin{bmatrix} 0 & 149 & 11212 & 5575 & 6360 & 12559 & 8108 & 8505 & 408 & 10026 & 12828 \\ 1 & 5237 & 490 & 10677 & 4998 & 3869 & 3734 & 3092 & 3509 & 7703 & 10305 \\ 2 & 8742 & 5553 & 2820 & 7085 & 12116 & 10485 & 564 & 7795 & 2972 & 2157 \\ 3 & 2699 & 4304 & 8350 & 712 & 2841 & 3250 & 4731 & 10105 & 517 & 7516 \\ 4 & 12067 & 1351 & 11992 & 12191 & 11267 & 5161 & 537 & 6166 & 4246 & 2363 \\ 5 & 6828 & 7107 & 2127 & 3724 & 5743 & 11040 & 10756 & 4073 & 1011 & 3422 \\ 6 & 11259 & 1216 & 9526 & 1466 & 10816 & 940 & 3744 & 2815 & 11506 & 11573 \\ 7 & 4549 & 11507 & 1118 & 1274 & 11751 & 5207 & 7854 & 12803 & 4047 & 6484 \\ 8 & 8430 & 4115 & 9440 & 413 & 4455 & 2262 & 7915 & 12402 & 8579 & 7052 \\ 9 & 3885 & 9126 & 5665 & 4505 & 2343 & 253 & 4707 & 3742 & 4166 & 1556 \\ 10 & 1704 & 8936 & 6775 & 8639 & 8179 & 7954 & 8234 & 7850 & 8883 & 8713 \\ 11 & 11716 & 4344 & 9087 & 11264 & 2274 & 8832 & 9147 & 11930 & 6054 & 5455 \\ 12 & 7323 & 3970 & 10329 & 2170 & 8262 & 3854 & 2087 & 12899 & 9497 & 11700 \\ 13 & 4418 & 1467 & 2490 & 5841 & 817 & 11453 & 533 & 11217 & 11962 & 5251 \\ 14 & 1541 & 4525 & 7976 & 3457 & 9536 & 7725 & 3788 & 2982 & 6307 & 5997 \\ 15 & 11484 & 2739 & 4023 & 12107 & 6516 & 551 & 2572 & 6628 & 8150 & 9852 \\ 16 & 6070 & 1761 & 4627 & 6534 & 7913 & 3730 & 11866 & 1813 & 12306 & 8249 \\ 17 & 12441 & 5489 & 8748 & 7837 & 7660 & 2102 & 11341 & 2936 & 6712 & 11977 \end{bmatrix} \tag{7.9}$$

$$
c_2(t) = \begin{bmatrix}
18 & 10155 & 4210 \\
19 & 1010 & 10483 \\
20 & 8900 & 10250 \\
21 & 10243 & 12278 \\
22 & 7070 & 4397 \\
23 & 12271 & 3887 \\
24 & 11980 & 6836 \\
25 & 9514 & 4356 \\
26 & 7137 & 10281 \\
27 & 11881 & 2526 \\
28 & 1969 & 11477 \\
29 & 3044 & 10921 \\
30 & 2236 & 8724 \\
31 & 9104 & 6340 \\
32 & 7342 & 8582 \\
33 & 11675 & 10405 \\
34 & 6467 & 12775 \\
35 & 3186 & 12198 \\
0 & 9621 & 11445 \\
1 & 7486 & 5611 \\
2 & 4319 & 4879 \\
3 & 2196 & 344 \\
4 & 7527 & 6650 \\
5 & 10693 & 2440 \\
6 & 6755 & 2706 \\
7 & 5144 & 5998 \\
8 & 11043 & 8033 \\
9 & 4846 & 4435 \\
10 & 4157 & 9228 \\
11 & 12270 & 6562 \\
12 & 11954 & 7592 \\
13 & 7420 & 2592 \\
14 & 8810 & 9636 \\
15 & 689 & 5430 \\
16 & 920 & 1304 \\
17 & 253 & 11934 \\
18 & 9559 & 6016 \\
19 & 312 & 7589 \\
20 & 4439 & 4197 \\
21 & 4002 & 9555 \\
22 & 12232 & 7779 \\
23 & 1494 & 8782 \\
24 & 10749 & 3969 \\
\vdots & \vdots & \vdots
\end{bmatrix}
\begin{bmatrix}
\vdots & \vdots & \vdots \\
25 & 4368 & 3479 \\
26 & 6316 & 5342 \\
27 & 2455 & 3493 \\
28 & 12157 & 7405 \\
29 & 6598 & 11495 \\
30 & 11805 & 4455 \\
31 & 9625 & 2090 \\
32 & 4731 & 2321 \\
33 & 3578 & 2608 \\
34 & 8504 & 1849 \\
35 & 4027 & 1151 \\
0 & 5647 & 4935 \\
1 & 4219 & 1870 \\
2 & 10968 & 8054 \\
3 & 6970 & 5447 \\
4 & 3217 & 5638 \\
5 & 8972 & 669 \\
6 & 5618 & 12472 \\
7 & 1457 & 1280 \\
8 & 8868 & 3883 \\
9 & 8866 & 1224 \\
10 & 8371 & 5972 \\
11 & 266 & 4405 \\
12 & 3706 & 3244 \\
13 & 6039 & 5844 \\
14 & 7200 & 3283 \\
15 & 1502 & 11282 \\
16 & 12318 & 2202 \\
17 & 4523 & 965 \\
18 & 9587 & 7011 \\
19 & 2552 & 2051 \\
20 & 12045 & 10306 \\
21 & 11070 & 5104 \\
22 & 6627 & 6906 \\
23 & 9889 & 2121 \\
24 & 829 & 9701 \\
25 & 2201 & 1819 \\
26 & 6689 & 12925 \\
27 & 2139 & 8757 \\
28 & 12004 & 5948 \\
29 & 8704 & 3191 \\
\vdots & \vdots & \vdots
\end{bmatrix}
\begin{bmatrix}
\vdots & \vdots & \vdots \\
30 & 8171 & 10933 \\
31 & 6297 & 7116 \\
32 & 616 & 7146 \\
33 & 5142 & 9761 \\
34 & 10377 & 8138 \\
35 & 7616 & 5811 \\
0 & 7285 & 9863 \\
1 & 7764 & 10867 \\
2 & 12343 & 9019 \\
3 & 4414 & 8331 \\
4 & 3464 & 642 \\
5 & 6960 & 2039 \\
6 & 786 & 3021 \\
7 & 710 & 2086 \\
8 & 7423 & 5601 \\
9 & 8120 & 4885 \\
10 & 12385 & 11990 \\
11 & 9739 & 10034 \\
12 & 424 & 10162 \\
13 & 1347 & 7597 \\
14 & 1450 & 112 \\
15 & 7965 & 8478 \\
16 & 8945 & 7397 \\
17 & 6590 & 8316 \\
18 & 6838 & 9011 \\
19 & 6174 & 9410 \\
20 & 255 & 113 \\
21 & 6197 & 5835 \\
22 & 12902 & 3844 \\
23 & 4377 & 3505 \\
24 & 5478 & 8672 \\
25 & 44531 & 2132 \\
26 & 9724 & 1380 \\
27 & 12131 & 11526 \\
28 & 12323 & 9511 \\
29 & 8231 & 1752 \\
30 & 497 & 9022 \\
31 & 9288 & 3080 \\
32 & 2481 & 7515 \\
33 & 2696 & 268 \\
34 & 4023 & 12341 \\
35 & 7108 & 5553
\end{bmatrix}
\qquad (7.10)
$$

Addresses of parity bit accumulators for rate $R = 5/6$, $n_{ldpc} = 64800$ *bits*

$$c_1(t) = \begin{bmatrix} 0 & 4362 & 416 & 8909 & 4156 & 3216 & 3112 & 2560 & 2912 & 6405 & 8593 & 4969 & 6723 \\ 1 & 2479 & 1786 & 8978 & 3011 & 4339 & 9313 & 6397 & 2957 & 7288 & 5484 & 6031 & 10217 \\ 2 & 10175 & 9009 & 9889 & 3091 & 4985 & 7267 & 4092 & 8874 & 5671 & 2777 & 2189 & 8716 \\ 3 & 9052 & 4795 & 3924 & 3370 & 10058 & 1128 & 9996 & 10165 & 9360 & 4297 & 434 & 5138 \\ 4 & 2379 & 7834 & 4835 & 2327 & 9843 & 804 & 329 & 8353 & 7167 & 3070 & 1528 & 7311 \\ 5 & 3435 & 7871 & 348 & 3693 & 1876 & 6585 & 10340 & 7144 & 5870 & 2084 & 4052 & 2782 \\ 6 & 3917 & 3111 & 3476 & 1304 & 10331 & 5939 & 5199 & 1611 & 1991 & 699 & 8316 & 9960 \\ 7 & 6883 & 3237 & 1717 & 10752 & 7891 & 9764 & 4745 & 3888 & 10009 & 4176 & 4614 & 1567 \\ 8 & 10587 & 2195 & 1689 & 2968 & 5420 & 2580 & 2883 & 6496 & 111 & 6023 & 1024 & 4449 \\ 9 & 3786 & 8593 & 2074 & 3321 & 5057 & 1450 & 3840 & 5444 & 6572 & 3094 & 9892 & 1512 \\ 10 & 8548 & 1848 & 10372 & 4585 & 7313 & 6536 & 6379 & 1766 & 9462 & 2456 & 5606 & 9975 \\ 11 & 8204 & 10593 & 7935 & 3636 & 3882 & 394 & 5968 & 8561 & 2395 & 7289 & 9267 & 9978 \\ 12 & 7795 & 74 & 1633 & 9542 & 6867 & 7352 & 6417 & 7568 & 10623 & 725 & 2531 & 9115 \\ 13 & 7151 & 2482 & 4260 & 5003 & 10105 & 7419 & 9203 & 6691 & 8798 & 2092 & 8263 & 3755 \\ 14 & 3600 & 570 & 4527 & 200 & 9718 & 6771 & 1995 & 8902 & 5446 & 768 & 1103 & 6520 \end{bmatrix} \tag{7.11}$$

$$
c_2(t) = \begin{bmatrix}
15 & 6304 & 7621 \\
16 & 6498 & 9209 \\
17 & 7293 & 6786 \\
18 & 5950 & 1708 \\
19 & 8521 & 1793 \\
20 & 6174 & 7854 \\
21 & 9773 & 1190 \\
22 & 9517 & 10268 \\
23 & 2181 & 9349 \\
24 & 1949 & 5560 \\
25 & 1556 & 555 \\
26 & 8600 & 3827 \\
27 & 5072 & 1057 \\
28 & 7928 & 3542 \\
29 & 3226 & 3762 \\
0 & 7045 & 2420 \\
1 & 9645 & 2641 \\
2 & 2774 & 2452 \\
3 & 5331 & 2031 \\
4 & 9400 & 7503 \\
5 & 1850 & 2338 \\
6 & 10456 & 9774 \\
7 & 1692 & 9276 \\
8 & 10037 & 4038 \\
9 & 3964 & 338 \\
10 & 2640 & 5087 \\
11 & 858 & 3473 \\
12 & 5582 & 5683 \\
13 & 9523 & 916 \\
\vdots & \vdots & \vdots
\end{bmatrix}
\begin{bmatrix}
\vdots & \vdots & \vdots \\
14 & 4107 & 1559 \\
15 & 4506 & 3491 \\
16 & 8191 & 4182 \\
17 & 10192 & 6157 \\
18 & 5668 & 3305 \\
19 & 3449 & 1540 \\
20 & 4766 & 2697 \\
21 & 4069 & 6675 \\
22 & 1117 & 1016 \\
23 & 5619 & 3085 \\
24 & 8483 & 8400 \\
25 & 8255 & 394 \\
26 & 6338 & 5042 \\
27 & 6174 & 5119 \\
28 & 7203 & 1989 \\
29 & 1781 & 5174 \\
0 & 1464 & 3559 \\
1 & 3376 & 4214 \\
2 & 7238 & 67 \\
3 & 10595 & 8831 \\
4 & 1221 & 6513 \\
5 & 5300 & 4652 \\
6 & 1429 & 9749 \\
7 & 7878 & 5131 \\
8 & 4435 & 10284 \\
9 & 6331 & 5507 \\
10 & 6662 & 4941 \\
11 & 9614 & 10238 \\
12 & 8400 & 8025 \\
13 & 9156 & 5630 \\
14 & 7067 & 8878 \\
\vdots & \vdots & \vdots
\end{bmatrix}
\begin{bmatrix}
\vdots & \vdots & \vdots \\
15 & 9027 & 3415 \\
16 & 1690 & 3866 \\
17 & 2854 & 8469 \\
18 & 6206 & 630 \\
19 & 363 & 5453 \\
20 & 4125 & 7008 \\
21 & 1612 & 6702 \\
22 & 9069 & 9226 \\
23 & 5767 & 4060 \\
24 & 3743 & 9237 \\
25 & 7018 & 5572 \\
26 & 8892 & 4536 \\
27 & 853 & 6064 \\
28 & 8069 & 5893 \\
29 & 2051 & 2885 \\
0 & 10691 & 3153 \\
1 & 3602 & 4055 \\
2 & 328 & 1717 \\
3 & 2219 & 9299 \\
4 & 31939 & 7898 \\
5 & 617 & 206 \\
6 & 8544 & 1374 \\
7 & 10676 & 3240 \\
8 & 6672 & 9489 \\
9 & 3170 & 7457 \\
10 & 7868 & 5731 \\
11 & 6121 & 10732 \\
12 & 4843 & 9132 \\
13 & 580 & 91 \\
14 & 6267 & 9290 \\
15 & 3009 & 2268 \\
16 & 195 & 2419 \\
17 & 8016 & 1557 \\
18 & 1516 & 9195 \\
19 & 8062 & 9064 \\
20 & 2095 & 8968 \\
21 & 753 & 7326 \\
22 & 6291 & 3833 \\
23 & 2614 & 7844 \\
24 & 2303 & 646 \\
\vdots & \vdots & \vdots
\end{bmatrix}
\begin{bmatrix}
\vdots & \vdots & \vdots \\
25 & 2075 & 611 \\
26 & 4687 & 362 \\
27 & 8684 & 9940 \\
28 & 4830 & 2065 \\
29 & 7038 & 1363 \\
0 & 1769 & 7837 \\
1 & 3801 & 1689 \\
2 & 10070 & 2359 \\
3 & 3667 & 9918 \\
4 & 1914 & 6920 \\
5 & 4244 & 5669 \\
6 & 10245 & 7821 \\
7 & 7648 & 3944 \\
8 & 3310 & 5488 \\
9 & 6346 & 9666 \\
10 & 7088 & 6122 \\
11 & 1291 & 7827 \\
12 & 10592 & 8945 \\
13 & 3609 & 7120 \\
14 & 9168 & 9112 \\
15 & 6203 & 8052 \\
16 & 3330 & 2895 \\
17 & 4264 & 10563 \\
18 & 10556 & 6496 \\
19 & 8807 & 7645 \\
20 & 1999 & 4530 \\
21 & 9202 & 6818 \\
22 & 3403 & 1734 \\
23 & 2106 & 9023 \\
24 & 6881 & 3883 \\
25 & 3895 & 2171 \\
26 & 4062 & 6424 \\
27 & 3755 & 9536
\end{bmatrix}
\tag{7.12}
$$

Addresses of parity bit accumulators for rate $R = 8/9$, $n_{ldpc} = 64800$ *bits*

$$c_1(t) = \begin{bmatrix} 0 & 6235 & 2848 & 3222 \\ 1 & 5800 & 3492 & 5348 \\ 2 & 2757 & 927 & 90 \\ 3 & 6961 & 4516 & 4739 \\ 4 & 1172 & 3237 & 6264 \\ 5 & 1927 & 2425 & 3683 \\ 6 & 3714 & 6309 & 2495 \\ 7 & 3070 & 6342 & 7154 \\ 8 & 2428 & 613 & 3761 \\ 9 & 2906 & 264 & 5927 \\ 10 & 1716 & 1950 & 4273 \\ 11 & 4613 & 6179 & 3491 \\ 12 & 4865 & 3286 & 6005 \\ 13 & 1343 & 5923 & 3529 \\ 14 & 4589 & 4035 & 2132 \\ 15 & 1579 & 3920 & 6737 \\ 16 & 1644 & 1191 & 5998 \\ 17 & 1482 & 2381 & 4620 \\ 18 & 6791 & 6014 & 6596 \\ 19 & 2738 & 5918 & 3786 \end{bmatrix} \tag{7.13}$$

$$
c_2(t) = \begin{bmatrix}
0 & 5156 & 6166 \\
1 & 1504 & 4356 \\
2 & 130 & 1904 \\
3 & 6027 & 3187 \\
4 & 6718 & 759 \\
5 & 6240 & 2870 \\
6 & 2343 & 1311 \\
7 & 1039 & 5465 \\
8 & 6617 & 2513 \\
9 & 1588 & 5222 \\
10 & 6561 & 535 \\
11 & 4765 & 2054 \\
12 & 5966 & 6892 \\
13 & 1969 & 3869 \\
14 & 3571 & 2420 \\
15 & 4632 & 981 \\
16 & 3215 & 4163 \\
17 & 973 & 3117 \\
18 & 3802 & 6198 \\
19 & 3794 & 3948 \\
0 & 3196 & 6126 \\
1 & 573 & 1909 \\
2 & 850 & 4034 \\
3 & 5622 & 1601 \\
4 & 6005 & 524 \\
5 & 5251 & 5783 \\
6 & 172 & 2032 \\
7 & 1875 & 2475 \\
8 & 497 & 1291 \\
9 & 2566 & 3430 \\
10 & 1249 & 740 \\
11 & 2944 & 1948 \\
12 & 6528 & 2899 \\
\vdots & \vdots & \vdots
\end{bmatrix}
\begin{bmatrix}
\vdots & \vdots & \vdots \\
13 & 2243 & 3616 \\
14 & 867 & 3733 \\
15 & 1374 & 4702 \\
16 & 4698 & 2285 \\
17 & 4760 & 3917 \\
18 & 1859 & 4058 \\
19 & 6141 & 3527 \\
0 & 2148 & 5066 \\
1 & 1306 & 145 \\
2 & 2319 & 871 \\
3 & 3463 & 1061 \\
4 & 5554 & 6647 \\
5 & 5837 & 339 \\
6 & 5821 & 4932 \\
7 & 6356 & 4756 \\
8 & 3930 & 418 \\
9 & 211 & 3094 \\
10 & 1007 & 4928 \\
\vdots & \vdots & \vdots
\end{bmatrix}
\begin{bmatrix}
\vdots & \vdots & \vdots \\
11 & 3584 & 1235 \\
12 & 6982 & 2869 \\
13 & 1612 & 1013 \\
14 & 953 & 4964 \\
15 & 4555 & 4410 \\
16 & 4925 & 4842 \\
17 & 5778 & 600 \\
18 & 6509 & 2417 \\
19 & 1260 & 4903 \\
0 & 3369 & 3031 \\
1 & 3557 & 3224 \\
2 & 3028 & 583 \\
3 & 3258 & 440 \\
4 & 6226 & 6655 \\
5 & 4895 & 1094 \\
6 & 1481 & 6847 \\
7 & 4433 & 1932 \\
8 & 2107 & 1649 \\
9 & 2119 & 2065 \\
10 & 4003 & 6388 \\
11 & 6720 & 3622 \\
12 & 3694 & 4521 \\
13 & 1164 & 7050 \\
14 & 1965 & 3613 \\
15 & 4331 & 66 \\
16 & 2970 & 1796 \\
17 & 4652 & 3218 \\
18 & 1762 & 4777 \\
19 & 5736 & 1399 \\
0 & 970 & 2572 \\
1 & 2062 & 6599 \\
2 & 4597 & 4870 \\
3 & 1228 & 6913 \\
4 & 4159 & 1037 \\
\vdots & \vdots & \vdots
\end{bmatrix}
\begin{bmatrix}
\vdots & \vdots & \vdots \\
5 & 2916 & 2362 \\
6 & 395 & 1226 \\
7 & 6911 & 4548 \\
8 & 4618 & 2241 \\
9 & 4120 & 4280 \\
10 & 5825 & 474 \\
11 & 2154 & 5558 \\
12 & 3793 & 5471 \\
13 & 5707 & 1595 \\
14 & 1403 & 325 \\
15 & 6601 & 5183 \\
16 & 6369 & 4569 \\
17 & 4846 & 896 \\
18 & 7092 & 6184 \\
19 & 6764 & 7127 \\
0 & 6358 & 1951 \\
1 & 3117 & 6960 \\
2 & 2710 & 7062 \\
3 & 1133 & 3604 \\
4 & 3694 & 657 \\
5 & 1355 & 110 \\
6 & 3329 & 6736 \\
7 & 2505 & 3407 \\
8 & 2462 & 4806 \\
9 & 4216 & 214 \\
10 & 5348 & 5619 \\
11 & 6627 & 6243 \\
12 & 2644 & 5073 \\
13 & 4212 & 5088 \\
14 & 3463 & 3889 \\
15 & 5306 & 478 \\
16 & 4320 & 6121 \\
\vdots & \vdots & \vdots
\end{bmatrix}
\begin{bmatrix}
\vdots & \vdots & \vdots \\
17 & 3961 & 1125 \\
18 & 5699 & 1195 \\
19 & 6511 & 792 \\
0 & 3934 & 2778 \\
1 & 3238 & 6587 \\
2 & 1111 & 6596 \\
3 & 1547 & 6226 \\
4 & 1446 & 3885 \\
5 & 3907 & 4043 \\
6 & 6839 & 2873 \\
7 & 1733 & 5615 \\
8 & 5202 & 4269 \\
9 & 3024 & 4722 \\
10 & 5445 & 6372 \\
11 & 370 & 1828 \\
12 & 4695 & 1600 \\
13 & 680 & 2074 \\
14 & 1801 & 6690 \\
15 & 2669 & 1377 \\
16 & 2463 & 1681 \\
17 & 5972 & 5171 \\
18 & 5728 & 4284 \\
19 & 1696 & 1459
\end{bmatrix}
\tag{7.14}
$$

Addresses of parity bit accumulators for rate $R = 9/10$, $n_{ldpc} = 64800\ bits$

$$c_1(t) = \begin{bmatrix} 0 & 5611 & 2563 & 2900 \\ 1 & 5220 & 3143 & 4813 \\ 2 & 2481 & 834 & 81 \\ 3 & 6265 & 4064 & 4265 \\ 4 & 1055 & 2914 & 5638 \\ 5 & 1734 & 2182 & 3315 \\ 6 & 3342 & 5678 & 2246 \\ 7 & 2185 & 552 & 3385 \\ 8 & 2615 & 236 & 5334 \\ 9 & 1546 & 1755 & 3846 \\ 10 & 4154 & 5561 & 3142 \\ 11 & 4382 & 2957 & 5400 \\ 12 & 1209 & 5329 & 3179 \\ 13 & 1421 & 3528 & 6063 \\ 14 & 1480 & 1072 & 5398 \\ 15 & 3843 & 1777 & 4369 \\ 16 & 1334 & 2145 & 4163 \\ 17 & 2368 & 5055 & 260 \end{bmatrix} \tag{7.15}$$

$$c_2(t) = \begin{bmatrix} 0 & 6118 & 5405 \\ 1 & 2994 & 4370 \\ 2 & 3405 & 1669 \\ 3 & 4640 & 5550 \\ 4 & 1354 & 3921 \\ 5 & 117 & 1713 \\ 6 & 5425 & 2866 \\ 7 & 6047 & 683 \\ 8 & 5616 & 2582 \\ 9 & 2108 & 1179 \\ 10 & 933 & 4921 \\ 11 & 5953 & 2261 \\ 12 & 1430 & 4699 \\ 13 & 5905 & 480 \\ 14 & 4289 & 1846 \\ 15 & 5374 & 6208 \\ 16 & 1775 & 3476 \\ 17 & 3216 & 2178 \\ 0 & 4165 & 884 \\ 1 & 2896 & 3744 \\ 2 & 874 & 2801 \\ 3 & 3423 & 5579 \\ 4 & 3404 & 3552 \\ 5 & 2876 & 5515 \\ 6 & 516 & 1719 \\ 7 & 765 & 3631 \\ 8 & 5059 & 1441 \\ 9 & 5629 & 598 \\ 10 & 5405 & 473 \\ 11 & 4724 & 5210 \\ 12 & 155 & 1832 \\ \vdots & \vdots & \vdots \end{bmatrix} \begin{bmatrix} \vdots & \vdots & \vdots \\ 13 & 1689 & 2229 \\ 14 & 449 & 1164 \\ 15 & 2308 & 3088 \\ 16 & 1122 & 6669 \\ 17 & 2268 & 5758 \\ 0 & 5878 & 2609 \\ 1 & 782 & 3359 \\ 2 & 1231 & 4231 \\ 3 & 4225 & 2052 \\ 4 & 4286 & 3517 \\ 5 & 5531 & 3184 \\ 6 & 1935 & 4560 \\ 7 & 1174 & 131 \\ 8 & 3115 & 956 \\ 9 & 3129 & 1088 \\ 10 & 5238 & 4440 \\ \vdots & \vdots & \vdots \end{bmatrix} \begin{bmatrix} \vdots & \vdots & \vdots \\ 11 & 5722 & 4280 \\ 12 & 3540 & 375 \\ 13 & 191 & 2782 \\ 14 & 906 & 4432 \\ 15 & 3225 & 1111 \\ 16 & 6296 & 2583 \\ 17 & 1457 & 903 \\ 0 & 855 & 4475 \\ 1 & 4097 & 3970 \\ 2 & 4433 & 4361 \\ 3 & 5198 & 541 \\ 4 & 1146 & 4426 \\ 5 & 3202 & 2902 \\ 6 & 2724 & 525 \\ 7 & 1083 & 4124 \\ 8 & 2326 & 6003 \\ 9 & 5605 & 5990 \\ 10 & 4376 & 1579 \\ 11 & 4407 & 984 \\ 12 & 1332 & 6163 \\ 13 & 5359 & 3975 \\ 14 & 1907 & 1854 \\ 15 & 3601 & 5748 \\ 16 & 6056 & 3266 \\ 17 & 3322 & 4085 \\ 0 & 1768 & 3244 \\ 1 & 2149 & 144 \\ 2 & 1589 & 4291 \\ 3 & 5154 & 1252 \\ 4 & 1855 & 5939 \\ \vdots & \vdots & \vdots \end{bmatrix} \begin{bmatrix} \vdots & \vdots & \vdots \\ 5 & 4820 & 2706 \\ 6 & 1475 & 3360 \\ 7 & 4266 & 693 \\ 8 & 4156 & 2018 \\ 9 & 2103 & 752 \\ 10 & 3710 & 3853 \\ 11 & 5123 & 931 \\ 12 & 6146 & 3323 \\ 13 & 1939 & 5002 \\ 14 & 5140 & 1437 \\ 15 & 1263 & 293 \\ 16 & 5949 & 4665 \\ 17 & 4548 & 6380 \\ 0 & 3171 & 4690 \\ 1 & 5204 & 2114 \\ 2 & 6384 & 5565 \\ 3 & 5722 & 1757 \\ 4 & 2805 & 6264 \\ 5 & 1202 & 2616 \\ 6 & 1018 & 3244 \\ 7 & 4018 & 5289 \\ 8 & 2257 & 3067 \\ 9 & 2483 & 3073 \\ 10 & 1196 & 5329 \\ 11 & 649 & 3918 \\ 12 & 3791 & 4581 \\ 13 & 5028 & 3803 \\ 14 & 3119 & 3506 \\ 15 & 4779 & 431 \\ 16 & 3888 & 5510 \\ \vdots & \vdots & \vdots \end{bmatrix} \begin{bmatrix} \vdots & \vdots & \vdots \\ 17 & 4387 & 4084 \\ 0 & 5836 & 1692 \\ 1 & 5126 & 1078 \\ 2 & 5721 & 6165 \\ 3 & 3540 & 2499 \\ 4 & 2225 & 6348 \\ 5 & 1044 & 1484 \\ 6 & 6323 & 4042 \\ 7 & 1313 & 5603 \\ 8 & 1303 & 3496 \\ 9 & 3516 & 3639 \\ 10 & 5161 & 2293 \\ 11 & 4682 & 3845 \\ 12 & 3045 & 643 \\ 13 & 2818 & 2616 \\ 14 & 3267 & 649 \\ 15 & 6236 & 593 \\ 16 & 646 & 2948 \\ 17 & 4213 & 1442 \\ 0 & 5779 & 1596 \\ 1 & 2403 & 1237 \\ 2 & 2217 & 1514 \\ 3 & 5609 & 716 \\ 4 & 5155 & 3858 \\ 5 & 1517 & 1312 \\ 6 & 2554 & 3158 \\ 7 & 5280 & 2643 \\ 8 & 4990 & 1353 \\ 9 & 5648 & 1170 \\ 10 & 1152 & 4366 \\ 11 & 3561 & 5368 \\ 12 & 3581 & 1411 \\ 13 & 5647 & 4661 \\ 14 & 1542 & 5401 \\ 15 & 5078 & 2687 \\ 16 & 316 & 1755 \\ 17 & 3392 & 1991 \end{bmatrix} \tag{7.16}$$