EFFICIENT DECODING OF BLOCK CODES IN DVB-S2 STANDARD

IBRAHIM SAMI SULAIMAN ATTAR

MAY 2015

EFFICIENT DECODING OF BLOCK CODES IN DVB-S2 STANDARD

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES OF
ÇANKAYA UNIVERSITY

BY
IBRAHIM SAMI SULAIMAN ATTAR

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE
IN
THE DEPARTMENT OF
ELECTRONIC AND COMMUNICATION ENGINEERING

MAY 2015

Title of the Thesis: **Efficient Decoding of Block Codes in DVB-S2 Standard**

Submitted by **Ibrahim Sami Sulaiman ATTAR**

Approval of the Graduate School of Natural and Applied Sciences, Anaya University.

Prof. Dr. Taner ALTUNOK
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.

Prof. Dr. Halil Tanyer EYYUBOĞLU
Head of Department

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

Assoc. Prof. Dr. Orhan GAZİ
Supervisor

Examination Date: 04.05.2015

Examining Committee Members:

Assoc. Prof. Dr. Orhan GAZİ	(Çankaya Univ.)
Dr. Zeki KOCABIYIKOĞLU	(TOBB Univ.)
Assist. Prof. Dr. Göker ŞENER	(Çankaya Univ.)

STATEMENT OF NON-PLAGIARISM PAGE

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name : Ibrahim, ATTAR

Signature :

Date : 04.05.2015

iii

ABSTRACT

EFFICIENT DECODING OG BLOCK CODES IN DVB-S2 STANDARD

ATTAR, Ibrahim
M.Sc., Department of Electronic and Communication Engineering
Supervisor: Assoc. Prof. Dr. Orhan GAZİ

May 2015, 39 pages

In this thesis work, decoding algorithms for BCH and LDPC codes used in DVB-S2 standard are implemented in MATLAB environment. For the decoding of LDPC codes, soft decision based belief propagation algorithm is used and for the decoding of BCH codes, a new recently developed technique, which is called soft decoding based on error magnitudes, is employed. The new approach has lower complexity compared to its counterparts and easy to implement in hardware and for this reason it can be used in future DVB-S2 standards. Bit error rate performance graphs for the decoding algorithms are obtained via computer simulation in MATLAB environment.

Keywords: DVB-S2 Standard, Digital Video Broadcasting, Second Generation, Low Density Parity Check Codes, BCH Codes.

ÖZ

DVB-S2 STANDARDINDAKİ BLOK KODLARININ ETKİN ÇÖZÜMLERİ

ATTAR, IBRAHIM
Yüksek Lisans, Elektronik ve Haberleşme Mühendisliği Anabilim Dalı
Tez Yöneticisi: Doç. Dr. Orhan GAZİ

Mayıs 2015, 39 sayfa

Bu tez çalışmasında DVB-S2 standardında bulunan LDPC ve blok kodlarının benzetimleri yapılmıştır. BCH kodlarının çözümü için hata genliklerini baz alan son zamanlarda geliştirilen bir yöntem denenmiştir. Kullanılan yöntem düşük karmaşıklık miktarına sahip olmakta ve donanımsal olarak gerçekleştirilmesi oldukça kolaydır. Benzetimi yapılan bu yeni teknik gelecekteki DVB-S standartlarında kullanılabilir. LDPC kodlarının çözümü için yumuşak karar yayılması algoritması kullanılmıştır. Bütün benzetimler MATLB ortamında yapılmış ve bit hata oranı performans grafikleri çizilmiştir.

Anahtar Kelimeleri: DVB-S2 Standardı, Dijital Video Yayımı, İkinci Nesil, Düşük Yoğunluklu Parity Kontrol Kodları, BCH Kodları

ACKNOWLEDGEMENTS

I would like to take this opportunity to pass my honest appreciation to Assoc. Prof. Dr. Orhan Gazi for the continuous backup and the valuable information and guidance, which helped me in completing this thesis through, varies stages.

I want also to make use of this this chance to thank my father, the man who spends his life just to effort a better life for us no matter what I am going to say about him, it will not be enough, never.

Mother, you are the brightness of my life, to be or not to be an important person in this world I am certain that you will always love me, support me and you will be proud of me. You are the kindest person in this world.

I also wish to thank my friends and all those who supported me with my appreciation and respect for their encouragement and assistance.

TABLE OF CONTENTS

LIST OF FIGURES

FIGURES

LIST OF TABLES

TABLES

LIST OF ABBREVIATIONS

AWGN	Additive White Gaussian Noise
BP	Belief Propagation
DVB-S2	Digital Video Broadcasting - Satellite - Second Generation
ETSI	European Telecommunications Standards Institute
MPEG-4 AVC	Advanced Video Coding
FEC	Forward Error Correction
LDPC	Low-Density Parity Check
BCH	Bose-Chaudhuri-Hocquenghem
DVB-S	Digital Video Broadcasting-Satellite
GF	Galois Field
LCM	Least Common Multiple
SNR	Signal to Noise Ratio
LFRS	Linear Feedback Shift Register
SDTV	Standard Definition Television
HDTV	High Definition Television
BPSK	Binary Phase-Shift Keying

CHAPTER 1

INTRODUCTION

1.1 Coding Theory

The block diagram of a classical communication system is illustrated in the Fig. 1 where channel encoder and decoder units are also displayed. The resulting errors in the transmitted message of communication over an unreliable channel represent the main problem of coding theory. The communications over the noisy channel make the coding theory worried about the reliability. Error correcting codes are utilized in extent range of communication system. The main concerns of coding theory can be summarized as:

1. Construct codes that can correct a maximal number of errors while using a minimal amount of redundancy.

2. Construct codes (as above) with efficient encoding and decoding procedures. Studying of error control codes is called coding theory [1]. This zone of discrete applied mathematics contains study and discovery of several coding schemes that are used to correct the errors occurred during data transmission. Error control coding is an important part of the overall communication system. Suppose that using an infrared connection to ray mp3 files from the computer to the smart phone. The transmitted data in this connection are 0's and 1's string. Generally, the smart phone receives a 0 when the computer sent a 0. Sometimes, malfunction hardware or channel noise causes the sent data 0 will be received as a 1. The challenge is to develop methods to overcome these errors which occur over data transmissions. Errors control codes are utilized for detecting and correcting errors that happen in the data transmissions over noisy channel.

1.2 Channel Coding

In digital communication system, Channel coding is used for protecting the message bits from interference and noise to reduce errors bit numbers. Channel coding operation is done by adding some selectively parity check bits to the transmitted information stream. The extra bits allow the operations of detection and correction of error bits in the received bits sequence and reliable data transmission is achieved. Protection of the information is the main aim of using channel coding; this is done by an expansion in bandwidth or a reduction in data rate. There are two main types of channel codes, convolutional codes and block codes [2].

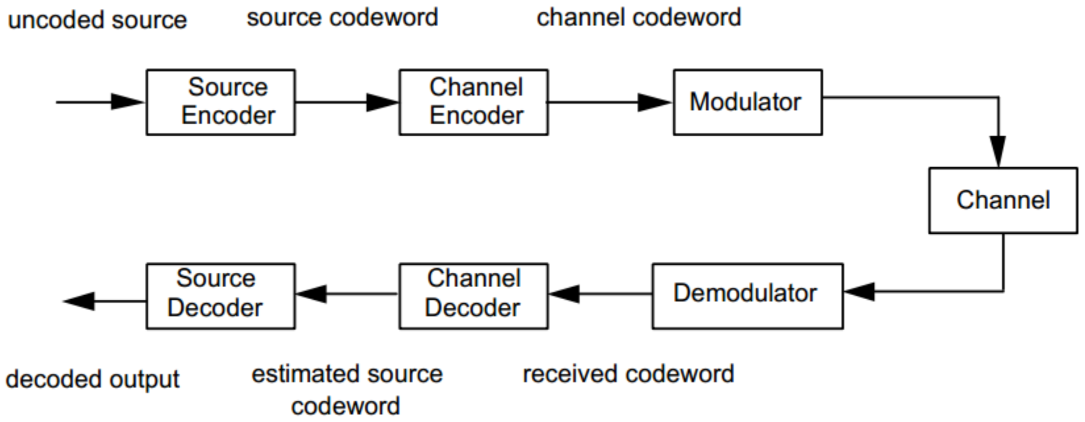

Figure 1: Block Diagram of Digital Communication Systems

1.3 Linear Block Codes

Block codes act on a block of bits. Block codes are indicated to (n, k) codes. A block of k data bits are mapped to a block of n sequence of bits. The amount of redundancy added to the information blocks equals to n-k. Using code generator, the code picks k data bits and calculates (n-k) parity bits. Generally, block codes are systematic and because of that the information bits stay without change attached either to the front or to the back with parity bits stream. After that, the receiver makes the decision about the received sequence validity. Shannon proved that the codes are existed with decoding errors of probability as small as coveted, and high rate of information (depends on the channel) [3]. Assuming that information source output is a stream of binary bits 0's and 1's. In block codes coding, this data binary stream is divided into message or data blocks of specific and fixed length, every message or data block consists of data bits k [4]. There are 2^k distinct messages totally. The block message (input message) is transformed into binary n-tuple c with n>k by the encoder. This binary n-tuple c is called the codeword of the message block. Thus, there are 2^k codewords corresponding to the possible messages 2^k. 2^k codewords is referred to block codes. The 2^k codewords must be distinct to make the block code useful. Therefore, there is a correspondence one-to-one between the block messages and their codewords. With the structure of block codes, the encoding complexity will be greatly reduced.

1.4 Additive White Gaussian Noise (AWGN)

White noise has a flat power spectral density and it is process signal (or random). Therefore, at any center frequency within a fixed bandwidth, AWGN is the signal which contains equal power. There are many applications and communication circuits in a wide band can be tested and measured by this white noise [5].

The most accepted model of the noise in communications channels is the following suppositions

- Additive noise, i.e. the received signal is the transmitted signal added with some noise and the noise is independent statistically from the signal.
- White noise, i.e. the noise power spectral is flat, therefor, for any non-zero time offset autocorrelation of this noise with the time is zero.
- All the noise samples have Gaussian distributions.

In communication channels, (AWGN) is random radio statistically noise differentiated by a wide frequency range with respect to the signal in a communication channels [6]. The channel capacity if AWGN channel is defined as

$$C = B \log_2(1 + \frac{p}{N_0 B})$$

Where C, B are the channel capacity and the analog signal bandwidth respectively, N_0 is the noise power spectral density (watts/hertz) and p is the power of the received signal.

1.5 Belief Propagation

Belief propagation (BP) is a message passing algorithm proposed by Judea Pearl in 1982 [7]. Belief propagation (BP) has been applied in a wide variety of problems in computer vision and pattern recognition that uses graphical models. Belief propagation (BP) method is a comparatively new and powerful approach for inference and overcome problems. There are many fields of statistical physics, machine learning and error-correcting codes that infer problems. Interestingly, several of the previous developed approaches such as the transfer matrix approach and turbo codes are actually just dissimilarity of the same belief propagation technique. Inference the problem deals usually with many questions as: There is a set of variables is given with statistical dependencies, when only the states of a possible little group of these variables from data is known, what are their most probable states? Also Belief Propagation is a dynamic programming approach to answer conditional probability queries in a graphical model. It is a Bayesian procedure inherently, conditioned on the observed nodes which computes the marginal distribution for each unobserved node.

Belief propagation works by sending messages along the edges of the graph factor [8]. Recall that a variable node i in a graph factor is associated with a random variable X_i, which can take a value from its state space X_i. A belief of a variable node i, denoted by $b_i(x_i)$ represents the likeness of random variable X_i to take value $x_i \in X_i$. In BP, the beliefs are calculated as a function of messages received by the nodes. It turns out that the beliefs are equal to the marginal probabilities for tree-like graphs.

1.6 Thesis Outline

Chapter one is the introduction to the channel coding and communication system generally with some explanations. In Chapter-2, the main thesis topic DVB-S2) is introduced and its encoder and decoder blocks (LDPC, BCH codes) with their encoding and decoding algorithms are explained.

The proposed soft decoding algorithms are presented in Chapter-3 with numerical examples. Finally, Chapter-4 includes Conclusion and Future work.

CHAPTER 2

DVB-S2 ENCODING AND DECODING

2.1 DVB-S2 Introduction

Transmitting information from one side to another side with high performance over a communication channel using the limited sources efficiently is the main goal of the modern communication systems. Twenty first century communication platforms are more effective in delivering messages from the sender to the recipient when compared to the ancient communication channels. Arguably, they are more effective and cost friendly when compared to the ancient communication methods, which were not only slow but also ineffective in delivering messages, especially the urgent ones. Modern technology has bred digital communication channels that make it possible for one to transmit messages via a wireless media and satellite. They have become more popular over the recent decades. Notably, these platforms' popularity is attributed to the efficiency and privacy provided by mobile networks. Hence, users may use these without fear that a third party may access their messages. Moreover, modern communication gadgets such as mobile phones and tablets are portable. Thus, one is sure to remain connected even in remote areas where communication was previously a big issue. Although wireless communication channels have eased communication in many ways, it has posed many challenges. Scholars are on the run to come up with new ways of handling many of these challenges. Some of these technologies include DVB-S2 [9]. The method has become increasingly popular considering its ability to transmit messages over many media platforms to its subscribers.

DVB-S2 (2nd Generation Digital Satellite Television Broadcasting) refers to a satellite application that came into being back in 2005 [10]. The application meets the set ETSI standards. The DVB-S2 standard revolves around three main principles. They include a high definition performance capacity, reliability, and wide range of media reception. The principles are in line with the modern source coding methods such as MPEG-4 AVC (Advanced Video Coding), and new FEC coding approaches, such as LDPC (Low-Density Parity Check) [11], BCH (Bose-Chaudhuri-Hocquenghem) codes [12]. For FEC, the DVB-S2 standard goes through two main stages. They include the inner (LDPC) and the outer coding (BCH) respectively. The DVB-S2 standard is more preferred over other channels due to its ability to transmit data more effectively when compared to other systems in various generations. Some of its advantages over other systems include flawless data, alongside flexibility to the receiver. One may acquire a good ration between good performance and complexity of DVBS2 through an appropriate modulation and coding of the channels. DVB-S2 profits from the latest modern evolutions in modulation and channel coding to achieve a well-balanced proportion between performance and complexity. The two methods are likely to generate around 30% efficiency of DVBS2 when compared to that of DVB-S [13] even without changing the environment. The transmitter side of the DVB-S2 is depicted in Fig. 2.

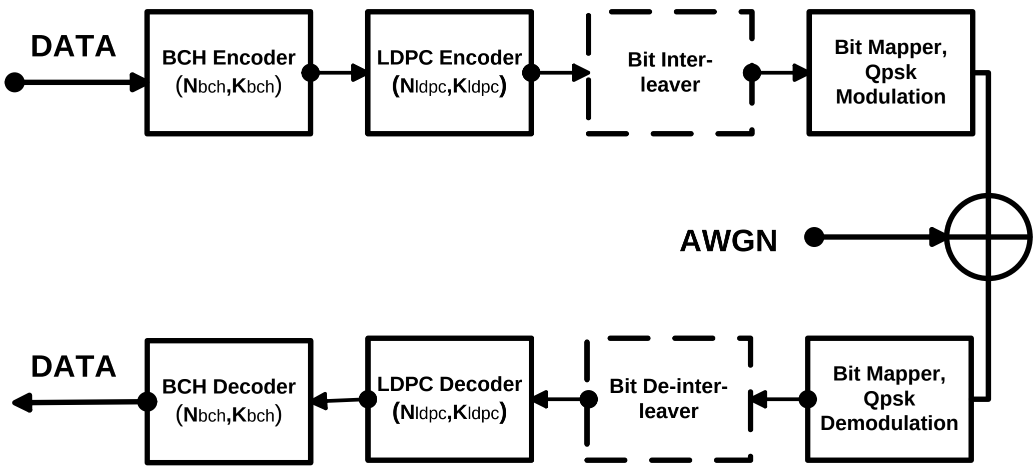

Figure 2: Basic DVB-S2 Block Diagram

2.2 BCH Codes

The Bose, Chaudhuri, and Hocquenghem (BCH) codes are the largest category of powerful error-correction cyclic codes [14]. It is one of the block codes that are the generalization of the Hamming codes for multiple-error corrections. The independent founders of binary BCH code was Bose and Chaudhuri back in the wake of 1960

[15]. Initially, the codes were invented back in 1959 by Hocquenghem. They refer to big groups of cyclic codes, which contain the binary and non-binary codes. BCH (n,k) binary codes can be constructed with any integer positive number $m \geq 3$ with these parameters.

$$n = 2^m - 1 \qquad n - k \leq mt \qquad d_{\min} \geq 2t + 1 = \delta$$

Where t, δ refer to the error−correction capability and the design distance code respectively.

The formula above represents a BCH code with special parameters. It guarantees the correction of *t* errors that occur in the *n* bit codewords. The generator polynomial of the BCH codes $g(x)$ is a polynomial in GF(2) for t-error correction capability. It has roots $\alpha, \alpha^2, \alpha^3, ..., \alpha^{2t}$ ($i.e\ g(\alpha^i) = 0$,i=1,2,...,2t). Assume $\phi_i(x)$ is the minimal polynomial corresponding to α^i. The generator polynomial is the least common multiple (LCM) of $\phi_i(x)$, i=1, 2, 2t. Hence,

$$g(x) = LCM\{\phi_1(x), \phi_2(x), \phi_3(x), ..., \phi_{2t}(x)\} \tag{2.1}$$

Since the minimal polynomials of $\alpha, \alpha^2, \alpha^3, ..., \alpha^{2t}$ are the same, the expression in (2.1) is reduced to

$$g(x) = LCM\{\phi_1(x), \phi_3(x), \phi_5(x), ..., \phi_{2t-1}(x)\} \tag{2.2}$$

The BCH codes in the above formula are referred to as the primitive, narrow-sense BCH codes [16]. Therefore, codeword polynomials becomes as $c(x) = c_0 + c_0 x + ... + c_{n-1} x^{n-1}$ which is a multiples of the generator polynomial $g(x)$ $\left(i.e.\ c(x) = m(x)\ g(x)\right)$, $c(x)$ has $\alpha, \alpha^1, \alpha^2, ..., \alpha^{2t}$ which serve as roots. Consequently,

$c(\alpha^i) = c_0 + c_1 \alpha^i + ... + c_{n-1} \alpha^{i(n-1)} = 0$, i=1, 2, 3, 2t or using matrix notation we can write

$$[c_0, c_1, c_2, ..., c_{2t}][H] = 0 \tag{2.3}$$

2.3 LDPC Codes

Low-Density Parity-Check (LDPC) code was proposed firstly by Gallagher in his Ph.D. thesis in 1960 [11]. After that, LDPC code rediscovered in 1996 by MacKay and Neal [20]. Shortly after the re-invention of the Low-Density Parity-Check (LDPC), the code has become quite popular in the modern society. The popularity is attributed to the fact that Low-Density Parity-Check (LDPC) may perform relatively well close to the Shannon limit [3]. Similarly, one may apply a simple algorithmic decoding procedure referred to as belief propagation to these codes. Moreover, they show high and good quality output bearing in mind that they are specialized error correction codes.

Low-Density Parity-Check (LDPC) code has a special matrix structure that contains an abundance of 0's and some few 1's. An (n, j, k) Low-Density Parity-Check (LDPC) has a block length n. All the columns have a minimal number j of 1's. On the other hand, all the rows have a minimal but fixed number k of 1's. The latter is an example of a matrix whose digits does not appear diagonally.

The design for Low-Density Parity-Check (LDPC) emanates from the parity check matrix. Here, the two types of nodes namely; variable and check nodes are joined to some points in each other. The distinction between the two nodes allows one to compute for parallel codes (i.e. the separation of sets allows parallel decoding computations). On the contrary, the decoding procedures for the turbo codes (which are the most competitors to LDPC codes) are different [21]. In fact, it is directly dependent on the codes in other blocks especially in serial computing. Arguably, Low-Density Parity-Check (LDPC) can be represented by a simple graph as illustrated by the Tanner's graph [22]. It helps one to analyze the performance of the codes accurately when compared to other methods. Moreover, it is effective when one wishes to optimize both the regular and irregular construction designs. Low–Density Parity–Check (LDPC) codes are used in Digital Video Broadcast–Satellite –Second Generation (DVB-S2) standard. In fact, for the very first time, Low–Density Parity-Check (LDPC) codes are used for standard broadcasting which came into being back in 2003 [23].

Tanner graph representing Low–Density Parity–Check (LDPC) is generally bipartite because it has many variable and constraint nodes on either of the sides [23]

[24]. Each of the variable nodes corresponds directly to a bit. Moreover, each of the constraint nodes is directly correspondent to the parity check when defining an appropriate codeword. A parity-check constraint is only applicable to a particular codeword sub-set that takes part in the constraint. The parity check may only be completed effectively if the XOR of the bits that take part is 0, 1 and they equal the sum of modulo 2.

The graph's edges have variable nodes attached to the constraint nodes to indicate that all the bits related to the variable will take part in the parity check constraints. In this case, a codeword refers to all the bit sequences related to the variable nodes on condition such that all the parity checks have been conducted successfully. The representation of a Low-Density Parity-Check (LDPC) in a Tanner graph reflects the representation of the code's parity check matrix. The code is represented in this latter description as a sequence of bits $x = (x_1, x_2, ..., x_n)$ to a linear algebraic simple equation. There are two elements of a parity-check matrix. They are 0s and 1s, complete with modulo 2 arithmetic. It entails multiplying x by a row of H. In addition, it involves taking the XOR of the bits in x to correspond the 1's in the row of H. The relationship between the parity-check matrix representations with the Tanner graph is clear. Its illustration in the Tanner graph depends on the structure of the bits involved. The Low-Density Parity-Check (LDPC) as illustrated by Gallager can be decoded using flipping and iterative decoding algorithms [13].

2.4 DVB-S2 Encoder

The DVB-S2 FEC part consists of two block codes concatenated in serial manner; these serially concatenated codes are the BCH and LDPC block codes. We can denote the BCH codes as outer codes and the Low-Density Parity-Check (LDPC) as the inner codes. The encoded frame structure for the concatenated system is given in Fig. 3. Low-Density Parity-Check (LDPC) starts the decoding process earlier than BCH. The signal to noise may affect the decoding procedure if it is relatively high. Hence, BCH offers a state of the art protection against many residual errors that may alter the performance and cases of degradation. Therefore, it enhances the strength of FEC if the SNR is higher. The Low-Density Parity-Check (LDPC) is deemed as being a super-channel if the Low-Density Parity-Check (LDPC) encoder and decoder are fully operational. Such an intuitive representation gives the two codes a competitive advantage over others. Choosing the concatenation carefully may enhance the performance of FEC decoding.

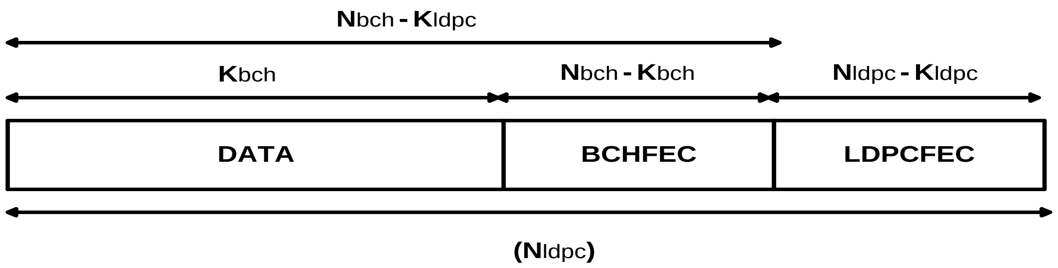

Figure 3: FEC Frame Structure of DVB-S2

Table 1 shows short FEC-FRAME (n_{ldpc} = 16200 bits) parameters, and Table 2 lists some of the coding parameters of normal FEC-FRAME (n_{ldpc} = 64800 bits), similarly.

Table 1: Coding Parameters of Short FECFRAME (n_{ldpc} = 16200 bits) [13]

LDPC Code Identifier	**BCH Uncoded Block k_{bch}**	**BCH Coded Block N_{bch} LDPC Uncoded Block K_{ldpc}**	**BCH T-Error Correction**	**Effective LDPC Rate K_{ldpc}/16 200**	**LDPC Coded Block n_{ldpc}**
1/4	3 072	3 240	12	1/5	16 200
1/3	5 232	5 400	12	1/3	16 200
2/5	6 312	6 480	12	2/5	16 200
1/2	7 032	7 200	12	4/9	16 200
3/5	9 552	9 720	12	3/5	16 200
2/3	10 632	10 800	12	2/3	16 200
3/4	11 712	11 880	12	11/15	16 200
4/5	12 432	12 600	12	7/9	16 200
5/6	13 152	13 320	12	37/45	16 200
8/9	14 232	14 400	12	8/9	16 200
9/10	NA	NA	NA	NA	NA

Table 2: Coding Parameters of Normal FECFRAME (n_{ldpc} = 64800 bits)

LDPC Code Identifier	**BCH Uncoded Block k_{bch}**	**BCH Coded Block N_{bch} LDPC Uncoded Block K_{ldpc}**	**BCH T-Error Correction**	**Effective LDPC Rate K_{ldpc}/16 200**	**LDPC Coded Block n_{ldpc}**
1/4	3 072	3 240	12	1/5	16 200
1/3	5 232	5 400	12	1/3	16 200
2/5	6 312	6 480	12	2/5	16 200
1/2	7 032	7 200	12	4/9	16 200
3/5	9 552	9 720	12	3/5	16 200
2/3	10 632	10 800	12	2/3	16 200
3/4	11 712	11 880	12	11/15	16 200
4/5	12 432	12 600	12	7/9	16 200
5/6	13 152	13 320	12	37/45	16 200
8/9	14 232	14 400	12	8/9	16 200
9/10	NA	NA	NA	NA	NA

2.4.1 BCH encoder

BCH's t-error correcting codes (N_{bch}, K_{bch}) are generally applicable to the random message bits (K_{bch}) for generating error-protected packets. Many BCH codes have the parameters for n_{ldpc} = 64800 are listed in Table 2 and the code parameters for n_{ldpc} = 16200 are available in Table 1. The generator polynomials of the BCH encoder for $\mathrm{t}-$error correcting are achieved by the multiplication of the first $\mathrm{t}-$ polynomials given in Table 3 for n_{ldpc} = 64800 and Table 4 for n_{ldpc} = 16200.

Table 3: BCH Polynomials (for Normal FECFRAME n_{ldpc} = 64800)

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

Table 4: BCH Polynomials (for Short FECFRAME n_{ldpc} = 16200)

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

BCH encoding [18] for the information bits $m = (m_{kbch-1}, m_{kbch-2}, ..., m_1, m_0)$ to get a codeword of the form $c = (m_{kbch-1}, m_{kbch-2}, ..., m_1, m_0, d_{nbch-kbc-1}, d_{nbch-kbc-2}, d_1, d_0)$ can be achieved using the following [18]:

- Multiplying the generated message polynomial $m(x) = m_{kbch-1}x^{kbch-1} + m_{kbch-2}x^{kbch-2} + ... + m_1 x + m_0 x$ by $x^{n_{bch}-k_{bch}}$.
- Divide $x^{n_{bch}-k_{bch}} m(x)$ by $g(x)$, the polynomial generator.

  Let $d(x) = d_{n_{bch}-k_{bch}-1} x^{n_{bch}-k_{bch}-1} + ... + d_1 x + d_0$ be the remainder.
- Set the codeword polynomial as $c(x) = x^{n_{bch}-k_{bch}} m(x) + d(x)$.

2.4.2 LDPC encoder

The BCH encoder output is fed to the inner LDPC [19]. The number of the parity bits added by LDPC codes is indicated in Tables 1 and 2 as shown in the following formula:

$$P_{ldpc} = N_{ldpc} - N_{bch}$$

LDPC encoder offers a support to 11 coding rates. The coding rates show the ratio between the number of message bits (K_{bch} bits) and the number of LDPC coded block bits (FEC-FRAME). For instance, the rate of 1/4 in a normal frame indicates:

$$\frac{N_{bch}}{N_{ldpc}} = \frac{16200}{64800} = \frac{1}{4}$$

Therefore, each information bit received from the FEC outer encoder (BCH), three parity check bits are added by LDPC encoder. A low ratio is preferred because it gives the data a stronger protection against errors that emanate from the LDPC encoder. Similarly, it strengthens the data transmission capacity and minimizes the systems' errors.

The LDPC decoder on the receiving end checks the sequence received in an iterative manner using fifty iterations. Such an error correction procedure employs a sparse parity-check matrix although the decoding algorithms are quite challenging to interpret.

Richardson and Urbanke [6] came up with the process of encoding that has successfully been useful for a long time. The procedure is applicable to all the codes that have a sparse equality-check matrix. The stages involved in this method include pre-processing and encoding. During the pre-processing stage, one follows all the steps in section H as illustrated in the following Figure's columns and rows.

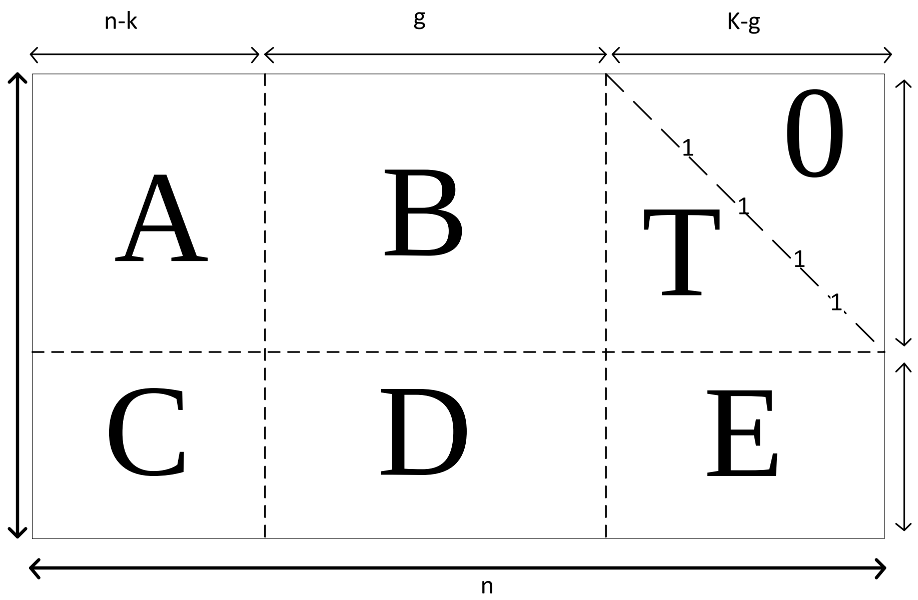

Figure 4: Richardson and Urbanke's Row and Column Variations

In matrix notation,

$$H = \begin{pmatrix} A & B & T \\ & & \\ C & D & E \end{pmatrix} \tag{2.4}$$

In this structure, T is a triangular matrix and it is quite weak although all the entrances have a value that equals to 1. The process involves permutation of the columns and rows in parity check matrix H. Parity check matrix H is a sparse matrix and also A, B, C, D, E, and T are sparse matrices. For instance, the distance assesses how close H can be, by row and column permutation, to the inferior triangular matrix. Multiply H from the left by

$$\begin{pmatrix} I & 0 \\ -ET^{-1} & I \end{pmatrix} \tag{2.5}$$

The results are

$$\begin{pmatrix} A & B & T \\ -ET^{-1}A + C & -ET^{-1}B + D & 0 \end{pmatrix} \tag{2.6}$$

Let the code parameters be as follows:

C is the codeword. Hence, c = (s, p_1, p_2)

s refers to the data bits

p_1 and p_2 refer to the parity bits,

p_1 has length g, p_2 has length $k - g$.

By $H^T C = 0$, we get

$$\begin{pmatrix} A & B & T \\ -ET^{-1}A + C & -ET^{-1}B + D & 0 \end{pmatrix} \begin{bmatrix} s \\ p1 \\ p2 \end{bmatrix} = 0 \tag{2.7}$$

from which we get,

$$As^T + Bp_1^T + Tp_2^T = 0 \tag{2.8}$$

And

$$(-ET^{-1}A + C)\mathrm{s}^T + (-ET^{-1}B + D)p_1^T = 0 \tag{2.9}$$

Computation of p_1 using Richardson and Urbanke’s encoding operation is explained in Table 5.

Table 5: Calculating p_1 Using Richardson and Urbanke's Encryption Approach

Operation	**Comment**	**Complexity**
As^T	Multiply the data by A sub-matrix	O(n)
$T^{-1}[As^T]$	Back exchange, the lower triangular T	O(n)
$-E[T^{-1}\ As^T]$	Multiplication by sparse matrix	O(n)
Cs^T	Multiplication by sparse matrix	O(n)
$[-ET^{-1}As^T]+[Cs^T]$	Addition	O(n)
$-\phi^{-1}(-ET^{-1}As^T + Cs^T)$	Multiply by g×g matrix	O(n+g2)

Computation of p_2 using Richardson and Urbanke's encoding operation is explained in Table 6.

Table 6: Calculating p_2 Using Richardson and Urbanke's Encryption Approach

Operation	**Comment**	**Complexity**
As^T	Multiplication by sparse matrix	O(n)
Bp_1^T	Multiplication by sparse matrix	O(n)
$[As^T\ \ Bp_1^T]$	Addition	O(n)
$-T^{-1}(As^T + Bp_1^T)$	Back exchange, the lower triangular T	O(n)

And the parameter ϕ is defined as:

$$\phi = -ET^{-1}B + D \tag{2.10}$$

Assuming that ϕ is non-singular we get

$$\begin{aligned} P_1^T &= \phi^{-1}(-ET^{-1}A + C)s^T \\ &= \phi^{-1}(-ET^{-1}A + C) \end{aligned} \tag{2.11}$$

The first step is to multiply the data by A submatrix As^T. The calculation complexity of A is on the order of n, i.e., O(n) due to the sparseness of A. Initially, it would compute $T-1[As^T]=yT$. Since $[As^T]=TyT$ and T is an inferior triangular. Through back-alteration yT can be computed in linear time. The calculation for $(-EyT)$ and (Cs^T) can also be done through O(n) complexity as (E,C) because they also sparse. Let:

$$(-ET-1As^T+Cs^T) = z^T \tag{2.12}$$

Then ϕ is g×g, and p_1 is calculated from (2.11) with complexity order O $(n+g^2)$. In addition we can calculate p_2 using

$$p_2^T = -T^{-1}(As^T + Bp_1^T) \tag{2.13}$$

The formula for computing p_2 is almost identical to that of computing p_1, and p_2 can be calculated with complexity order O(n), C can be calculated with complexity order $O(n+g^2)$. The arithmetic and encrypting procedure are run in $O(n^2)$. During the encoding, the performance of the encoder is still economical on time and less complexity. Richardson and Urbanke also proved that for custom codes, the predictable g is restricted by O(n), thus, the encoder works in O(n).

2.5 DVB-S2 Decoder

DVB-S2 decoder consists of two block codes (LDPC and BCH decoders). LDPC decoder receives the coded sequence and decodes it using belief propagation algorithm, and sends it to BCH decoder. After receiving soft information from LDPC decoder, BCH decoder starts to achieve its decoding algorithm. Recently a new approach for the decoding of BCH codes using error magnitudes is proposed in [26]. In this thesis work we will use this new approach to decode the BCH codes available in DVB-S2 structure.

2.5.1 LDPC decoder

The selected decoder should be as suitable as possible when examining provisional data with a bit of 0 or 1 after the receiving the vector y. Hence, $p_i = P_r \; [c_i \; = 1 | \; y]$ given that c_i is the code symbol transmitted given the received signal vector y. Therefore,

$$pr[c_i = 0 | y] = 1 - p_i \tag{2.14}$$

Let $q^l{}_{ij}$ represents the information (probability) flowing through check node c_j from the variable node v_i. Then we have the probabilities $qij^l(0)$ and $qij^l(1)$ which can also be names as the belief scores for the y_i values, i.e. $q_{ij}{}^l(0) = prob(y_i = 0),\ q_{ij}{}^l(1) = prob(y_i = 1)$ and they satisfy

$$qij^l(0) + qij^l(1) = 1 \tag{2.15}$$

Let, $q^l_{ij}(1) = P_i$ and $q^l_{ij}(0) = 1 - P_i$, and let r^l_{ij} be the probability transferred from check node c_i to data node v_j in the first round. The probability r_{ij} is calculated for bits 0 and 1, i.e. $r^l{}_{ij}(0)$, and $r^l{}_{ij}(1)$ are the probabilities for y_i being equal to 0 or 1 respectively such that:

$$rij^l(0) + rij^l(1) = 1$$

In addition, $rij^l(0)$ is a likelihood that an even number of 1's are received from all other data nodes. Initially, let's suppose that there are an even number of 1's on the second data node. Let q_1 be the likelihood that there is 1 at data node v_1 and q_2 is the likelihood that there is a 1 at data node v_2 . It is computed as:

$$\Pr(d_1 \oplus d_2) = q_1 q_2 + (1-q_1)(1-q_2) \tag{2.16}$$
$$= 1 - q_1 - q_2 + 2q_1 q_2$$
$$= \frac{1}{2}(2 - 2q_1 - 2q_2 + 4q_1 q_2)$$
$$= \frac{1}{2}[1 + (1-2q_1)(1-2q_2)] = q$$

Then consider the likelihood of having an even number of 1's on three data nodes, v_1, v_2, v_3, (1 – q) is the likelihood that there is an odd number of 1's on v_1 and v_2: which is calculated as

$$= \frac{1}{2}[1 + (1-2q_1)(1-2q_2)(1-2q_3)]$$

Then,

$$\Pr(d_1 \oplus d_2 \cdots \oplus \mathrm{d_n} = 0) = \frac{1}{2} + \frac{1}{2}\prod_{i=1}^{n}(1-2q_i)$$

It is important to note that the information which c_j transmits to v_i during the first round is

$$r_{ji}^{(l)}(0) = \frac{1}{2} + \frac{1}{2}\prod_{i' \in V_j \neq i}^{n} (1-2q_{i'j}(1))] \tag{2.17}$$

With

$$r_{ji}^{(l)}(1) = 1 - r_{ji}^{(l)}(0)$$

If v_j represents the sets of data nodes attached to the check codes, the probability which v_j passes to c_i on the first round can be computed for the bits 0 and 1 as:

$$q_{ji}^{(l)}(0) = k_{ij}(1-p_i)\prod_{j' \in C_i \neq j}^{n} r_{j'i}^{(l-1)}(0) \tag{2.18}$$

$$q_{ji}^{(l)}(1) = k_{ij}\, p_i \prod_{j' \in C_i \neq j}^{n} r_{j'i}^{(l-1)}(1) \tag{2.19}$$

Where C_i represents all the check nodes. The constant k_{ij} can be calculated using the equality:

$$q_{ji}^{(l)}(0) + q_{ji}^{(l)}(1) = 1 \tag{2.20}$$

For each data node, the following formulas are used to compute the bit probabilities:

$$\begin{aligned} Q_i^{(l)}(0) &= k_i\,(1-p_i)\prod_{j \in C_i}^{n} r_{ji}^{(l-1)}(0) \\ Q_i^{(l)}(1) &= k_i\, p_i \prod_{j \in C_i}^{n} r_{ji}^{(l-1)}(1) \end{aligned} \tag{2.21}$$

Where $Q_i^{(l)}$ refers to the probability of 0 and 1 at data node v_i during round l.

Belief propagation algorithm can also be implemented in logarithmic scale where the likelihood formulas are given as:

$$L_i = \left(\frac{\Pr(v_i = 0|y)}{\Pr(v_i = 1|y)}\right) = \frac{1-p_i}{p_i} \tag{2.23}$$

$$l_i = \ln(L_i) = \left(\frac{\Pr(v_i = 0|y)}{\Pr(v_i = 1|y)}\right) \tag{2.24}$$

In this case, L_i is the likelihood ratio and l_i is the log-likelihood ratio for v_i through the log ratio, multiplications are turned into sums because they are quite cheap to use in hardware. Using the probability ratio, we end up with:

$$P_i = \frac{1}{1+L_i} \tag{2.25}$$

From (2.14) and (2.15), the information that v_j transmits to c_i at the first round is:

$$m_{ji}^{(l)} = \ln\frac{r_{ji}^{(l)}(0)}{r_{ji}^{(l)}(1)} = \ln\frac{\frac{1}{2}+\frac{1}{2}\prod_{i\in Mj\neq i}(1-2q_{i^{'}j}^{(l-1)}(1))}{\frac{1}{2}-\frac{1}{2}\prod_{i\in Mj\neq i}(1-2q_{i^{'}j}^{(l-1)}(1))} \tag{2.26}$$

$$= \ln\frac{1+\prod_{i^{'}\in Vj\neq i}\tanh(\frac{m_{i^{'}j}^{(l-1)}}{2})}{1-\prod_{i^{'}\in Vj\neq i}\tanh(\frac{m_{i^{'}j}^{(l-1)}}{2})}$$

Using (2.24) and (2.26) we get

$$e^{m_{i^{'}j}} = \frac{1-q_{i^{'}j}(1)}{q_{i^{'}j}(1)} \tag{2.27}$$

Which leads to

$$q_{i^{'}j}(1) = \frac{1}{1+e^{m_{i^{'}j}}} \tag{2.28}$$

$$1-2q_{i^{'}j}(1) = \frac{e^{m_{i^{'}j}}-1}{e^{m_{i^{'}j}}+1} = \tanh(\frac{m_{i^{'}j}}{2}) \tag{2.29}$$

Both equations (26), (28) fit into

$$l_i^{(l)} = \ln\frac{Q_i^{(l)}(0)}{Q_i^{(l)}(1)} = l_i^{(0)} + \sum_{j\in C_i} m_{ji}^{(l)} \tag{2.30}$$

And decoding decision is made according to If $l_i^{(l)} > 0$ then $c_i = 0$, else $c_i = 1$.

During a training session, broadcasting is used when one has to repeat the process many times or if they have to do it until the probability becomes close to the certainty value. A particular probability is $l_i = \pm\infty$, where $P_i = 0$ for $l_i = \infty$ and $P_i = 1$ for $l_i = -\infty$.

One of the major features of belief propagation is the timing of its performance although it is directly linear to the size of the code. The process traverses through the various check and data codes.

2.5.2 BCH decoder

Decoding the binary BCH code using an algebraic procedure entails the following steps.

a. Computing the value of the syndrome
b. Identifying the location of the polynomial which locates errors. There are two methods to locate the errors. They are the Berlekamp-Massey algorithm and Peterson's algorithm.
c. And determining the error locator polynomial's roots. This can be done through Chien search.

2.5.2.1 Error detection

It is important to detect preliminary errors before decoding any of the BCH codewords. The syndrome of the code is determined through:

$$S = S_1 = c(\alpha),\ \mathrm{S}_2 = c(\alpha^2),\ \mathrm{S}_{2t} = c(\alpha^{2t})$$

If the received codeword is not corrupted by the noise, then the syndrome has zero value since $g(\alpha) = g(\alpha^2) = ... = g(\alpha^{2t})$, otherwise the syndrome is not zero. And the decoding should be accomplished to trade on redundancy introduced and correct the errors occurred. If an error is detected, then the codeword errors can be corrected by using the Berlekamp-Massey algorithm.

2.5.2.2 Berlekamp-Massey algorithm

Before describing the algorithm, let's get acquainted with mathematical concepts. Let r be the received vector that contains v errors in locations $i_1, i_2, ..., i_v$ so that each element of S can be calculated as follows:

$$S_j = \sum_{l=1}^{v} (\alpha^j)^{i_l} = \sum_{l=1}^{v} (\alpha^{i_l})^j$$

Letting $X_l = \alpha^{i_l}$, we get the alternative expression for the syndromes as shown:

$$S_j = \sum_{l=1}^{v} X_l^{\,j} \quad \text{j=1,2,...,2t} \tag{2.31}$$

The error locator polynomial is written as:

$$\wedge(x) = \prod_{l=1}^{v} (1 - X_l x) = \wedge_v x^v + ... + \wedge_1 x + \wedge_0 \tag{2.32}$$

Where $\wedge_0 = 1$. The roots of (2.32) connote the error locators' reciprocal. A linear relationship between the coefficients in (2.32) and the syndromes can be written as:

$$S_j = -\sum_{i=1}^{v} \wedge_i S_{j-i} \quad \text{j=v+1,v+2,...,2t} \tag{2.33}$$

The above formula is a description of the output of the linear feedback shift register (LFSR) with the co-efficient $\wedge_1, \wedge_2, ..., \wedge_v$. Through the Berlekamp-Massey algorithm, it is possible to come up with a LFRS that generates a complete sequence $\{S_1, S_2, ..., S_{2t}\}$. Any LFRS that is potentially capable of producing S_1 alongside a sequence of $\{S_1, S_2\}$.

Once such a LFRS is found, there is no need for any modification and vice versa. It would need appropriate connotation to represent $\wedge(x)$ in various algorithmic stages before computing. Assume that L_k denotes generated at stage k. Hence:

$$\wedge^{[k]}(x) = 1 + \wedge_1^{[k]} x + ... + \wedge_{L_k}^{[k]} x^{L_k} \tag{2.34}$$

Indicates that the LFSR can produce a $\{S_1, S_2, ..., S_k\}$ output. That is:

$$S_j = -\sum_{i=1}^{L_k} \wedge_i^{[k]} S_{j-i} \qquad \text{j=L}_k + 1, \text{L}_k + 2, ...k \tag{2.35}$$

There are instances where we may have a polynomial connection of $\wedge^{[k-1]}(x)$ of length L_{k-1} that produces $\{S_1, S_2, ..., S_{k-1}\}$ for some $k-1 < 2t$. We can verify the connection of the polynomial through calculating the value of the output using the following formula:

$$\hat{S}_k = -\sum_{i=1}^{L_k - 1} \wedge_i^{[k-1]} S_{k-i} \tag{2.36}$$

If $\hat{S}_k$ has a similar value as S_k, it is not necessary to update LFSR. Hence, $\wedge^k(x) = \wedge^{[k-1]}(x)$ and $L_k = L_{k-1}$. If there is a non- zero discrepancy that emanates from $\wedge^{[k-1]}(x)$, thus:

$$d_k = S_k - \hat{S}_k = S_k + \sum_{i=1}^{L_k - 1} \wedge_i^{[k-1]} S_{k-i} = \sum_{i=0}^{L_k - 1} \wedge_i^{[k-1]} S_{k-i} \tag{2.37}$$

In such an instance, we may update of the polynomial using the following formula:

$$\wedge^{[k]}(x) = \wedge^{[k-1]}(x) + A x^l \wedge^{[m-1]}(x) \tag{2.38}$$

The new polynomial helps the calculating of the value of the new discrepancy as illustrated below:

$d_k{}'$, as

$$d_k{}' = \sum_{i=0}^{L_k} \wedge_i{}^{[k]} S_{k-i} \tag{2.39}$$
$$= \sum_{i=0}^{L_k-1} \wedge_i{}^{[k-1]} Ss_{k-i} + A \sum_{i=0}^{L_m-1} \wedge_i{}^{[m-1]} S_{k-i-l}$$

The second summation generates:

$$A \sum_{i=0}^{L_m-1} \wedge_i{}^{[m-1]} S_{m-i} = A d_m \tag{2.40}$$

If $A = -d_m{}^{-1} d_k$, the computation in (2.39) breeds:

$$d_k{}' = d_k - d_m{}^{-1} d_k d_m = 0$$

2.6 DVB-S2 Applications

Multimedia communications supplied to low population with wide geographical areas, the deployment of the satellites in this case is less costly than the corresponding terrestrial networks to achieve the same services. Similarly, regions with low population have experienced the advantages of the communication media. Notably, communication using the satellite media is quite cheaper than other media. Some of the modern day communication media include maritime communications related to the radio navigation systems. Television broadcasting uses one signal to reach a big community in a certain region. All the modern media links require state of the art technology and terrestrial applications to remain effective for long. Some of the advantages of Broadcasting Services of High Definition TeleVision (HDTV) and Standard Definition TeleVision (SDTV) are satellite News Gathering, use of digital transmitters and interactive services.

CHAPTER 3

SOFT DECISION DECODING OF DVB-S2 BLOCK CODES

3.1. The First Approach of Soft Decision Decoding of BCH Codes Using Error Magnitudes.

The BCH error correcting codes have 2t syndromes that can be used to correct errors in the t location. Similarly, they may be used to resolve any erasures of 2t syndromes during the decoding process. Any erasures during the decoding procedure should be solved to enhance the magnitude of the identified errors [25]. Similarly, the erasures should be identified in the binary codes to ensure that the magnitudes stand at either 0 or 1. The relationship between the receiver input and the output is affected with β_i, i=1,...,n . It indicates that the location of the codeword appears in the metric m_i . Solving the 2t magnitude equations is unlikely to breed a solution using GF (2). It is one of the methods that are used to acquire a solution that has more than 2t channel outputs. Solving problems involving error magnitudes comes up with a soft decision decoding process that has been proved to be fast and efficient. The procedure entails solving any emerging the extended error magnitude equation given as

$$\begin{bmatrix} \beta_1\beta_2\ldots\ldots\ldots\ldots\beta_{2t} \\ \beta_1^2\beta_2^2\ldots\ldots\ldots\ldots\beta_{2t}^2 \\ \beta_1^3\beta_2^3\ldots\ldots\ldots\ldots\beta_{2t}^3 \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ \beta_1^{2t}\beta_2^{2t}\ldots\ldots\ldots\beta_{2t}^{2t} \end{bmatrix} \begin{bmatrix} \gamma_1 \\ \gamma_2 \\ \gamma_3 \\ .. \\ .. \\ \gamma_{2t} \end{bmatrix} = \begin{bmatrix} s_1 \ \Delta s_{1,2t+1}\ldots\ldots\ldots\ldots\Delta s_{1,n} \\ s_2 \ \Delta s_{2,2t+1}\ldots\ldots\ldots\ldots\Delta s_{2,n} \\ s_3 \ \Delta s_{3,2t+1}\ldots\ldots\ldots\ldots\Delta s_{3,n} \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ s_{2t} \ \Delta s_{2t,2t+1}\ldots\ldots\ldots\Delta s_{2t,n} \end{bmatrix} \tag{3.1}$$

Where γ_i represents the magnitude that contains the error which corresponds to location β_i, and S_i are the syndromes. $\Delta S_{i,j}$ is the incremental syndrome which is calculated by changing the bit in location β_j . A solution γ's exists since the matrix of β's is a Vandermondematrix. However, the solution is not in GF(2). The matrix solution can be presented as:

$$\begin{bmatrix} \gamma_1 \\ \gamma_2 \\ \gamma_3 \\ .. \\ .. \\ \gamma_{2t} \end{bmatrix} = \begin{bmatrix} \mathrm{e}_{1,0} \ \Delta e_{1,2t+1} .... \Delta e_{1,n} \\ \mathrm{e}_{2,0} \ \Delta e_{2,2t+1} .... \Delta e_{2,n} \\ \mathrm{e}_{3,0} \ \Delta e_{3,2t+1} .... \Delta e_{3,n} \\ .... \\ .... \\ \mathrm{e}_{2\mathrm{t},0} \ \Delta e_{2\mathrm{t},2t+1} .... \Delta e_{2\mathrm{t},n} \end{bmatrix} \tag{3.2}$$

In case where the first column of the above matrix appears in Galois Field GF(2), there are many errors exist in the first 2t locations. However, if the first column’s elements do not appear in GF(2), the resolution is derived by adding successive column (or sets of columns) to the first one to get a column which the elements of it in GF(2). After that, the outcome column together with the corresponding best metric totally produces the maximum likelihood decoding. Then, the error positions are included in the consonant locations of the added column(s) plus the first 2t most possible locations. Generally, few of the added columns may be used to get a solution to these problems. The decoding time depends on the probability of solving any equation of the extended error magnitude through the fast algorithm. Many sub-optimal decision of decoding algorithms entails doing away with the complete search through adding more columns in the extended error magnitude.

3.1.1 Numerical example

As a simple example consider the cyclic (7,4,1) BCH codes with the generator polynomial $g(x) = x^3 + x + 1$ and GF(8), the data is $d = [1\ 0\ 0\ 0]$.

The encoding operation for the given data starts with calculating

$$\left[X^{N-K} * d(\mathrm{x})\right]$$

Then the remainder

$$r(x) = [X^{N-K} * d(x)] / g(x)$$

Is calculated and code polynomial is formed using

$$c(x) = [X^{N-K} * d(x)] + r(x)$$

As

$$c(\mathrm{x}) = [1\ 1\ 0\ 1\ 0\ 0\ 0]$$

The second step is modulation of the codeword. We used BPSK modulation, the modulated codeword is:

$$c(\mathrm{x}) = [+1\ +1\ -1\ +1\ -1\ -1\ -1]$$

Then, adding the noise (AWGN) channel to the modulated signal with SNR approximately 7 dB, so that two errors are occurred, we get the received signal as

$$r = (-0.218,\ 2.228,\ 0{,}991,\ 3.071,\ -0.537,\ -1.269,\ -2.108)$$

Then we take the absolute values for the received signal and rearrange these absolute values in ascending order and indicating the locations of each value by powers of α in the sorted vector we get the β vector as:

$$\beta = (1, \alpha^4, \alpha^2, \alpha^5, \alpha^6, \alpha\ , \alpha^3)$$

Whose elements can be converted to binary using the table given below

Table 7: Representation for the Elements of GF(2) Generated by p(x) = $1+x+x^3$

Power representation	**Polynomial representation**	**3-tuple representation**
0	0	(0 0 0)
1	1	(1 0 0)
α	α	(0 1 0)
α^2	α^2	(0 0 1)
α^3	$\alpha+1$	(1 1 0)
α^4	$\alpha+\alpha^2$	(0 1 1)
α^5	$\alpha+1+\alpha^2$	(1 1 1)
α^6	α^2+1	(1 0 1)
α^7	1	(1 0 0)

In the next step we calculate syndromes $(s_1 .....s_{2t})$ using:

$$S_i = r(\alpha^i),\ \text{i=[1,......,2t]},\ (S_i)^2 = S_{2i}$$

And for the received demodulated signal

$$r = (0\ 1\ 1\ 1\ 0\ 0\ 0)$$

The syndromes are calculated as

$$S_1 = \alpha^6,\ S_2 = S_1{}^2 \rightarrow S_2 = \alpha^5$$

Next we will calculate $\Delta S_{i,j}$, let remind the equations as given below

$$\beta = (1\ ,\ \alpha^4, \alpha^2, \alpha^5,\ \alpha^6, \alpha\ ,\ \alpha^3)$$
$$\uparrow\ \uparrow\ \uparrow\ \uparrow\ \uparrow\ \uparrow\ \uparrow$$
$$r = (0\ ,\ 1\ ,1\ ,1\ ,0\ ,0\ ,0\ )$$
$$1,\ \mathrm{x}\ ,\mathrm{x}^2\ ,\mathrm{x}^3\ ,\mathrm{x}^4\ ,\mathrm{x}^5\ ,\mathrm{x}^6$$
$$r(\mathrm{x}) = \mathrm{x} + \mathrm{x}^2 + \mathrm{x}^3 \qquad , r(\mathrm{x}^2) = \mathrm{x}^2 + \mathrm{x}^4 + \mathrm{x}^6$$
$$r(\alpha) = \alpha + \alpha^2 + \alpha^3 \qquad , r(\alpha^2) = \alpha^2 + \alpha^4 + \alpha^6$$

To full all the elements in the matrix (3.1), $\Delta s_{n,2t}$ should be calculated as shown below:

$$\Delta s_{1,3} = s_1 - \tilde{s}_3 \qquad \Delta s_{1,4} = s_1 - \tilde{s}_4$$
$$r(\mathrm{x}) = \mathrm{x} + \mathrm{x}^3 \qquad r(\mathrm{x}) = \mathrm{x} + \mathrm{x}^2 + \mathrm{x}^3 + x^5$$
$$r(\alpha) = \alpha + \alpha^3 \qquad r(\alpha) = \alpha + \alpha^2 + \alpha^3 + \alpha^5$$
$$\tilde{s}_3 = \cancel{\alpha} + \cancel{\alpha} + 1 \qquad \tilde{s}_4 = \alpha + \cancel{\alpha}^2 + \cancel{\alpha} + 1 + \cancel{\alpha}^2 + \cancel{\alpha} + 1$$
$$\tilde{s}_3 = 1 \qquad \tilde{s}_4 = \alpha$$
$$\Delta s_{1,3} = \alpha^6 + 1 \qquad \Delta s_{1,4} = \alpha^6 + \alpha$$
$$\Delta s_{1,3} = \alpha^2 + 1 + 1 \qquad \Delta s_{1,4} = \alpha^2 + \alpha + 1$$
$$\Delta s_{1,3} = \alpha^2 \qquad \Delta s_{1,4} = \alpha^5$$

$$\Delta s_{1,5} = s_1 - \tilde{s}_5 \qquad \Delta s_{1,6} = s_1 - \tilde{s}_6$$
$$r(\mathrm{x}) = \mathrm{x} + \mathrm{x}^2 + \mathrm{x}^3 + x^6 \qquad r(\mathrm{x}) = \mathrm{x}^2 + \mathrm{x}^3$$
$$r(\alpha) = \alpha + \alpha^2 + \alpha^3 + \alpha^5 \qquad r(\alpha) = \alpha^2 + \alpha^3$$
$$\tilde{s}_5 = \cancel{\alpha} + \cancel{\alpha}^2 + \cancel{\alpha} + 1 + \cancel{\alpha}^2 + 1 \qquad \tilde{s}_6 = \alpha^2 + \alpha + 1$$
$$\tilde{s}_5 = 0 \qquad \tilde{s}_6 = \alpha^5$$
$$\Delta s_{1,5} = s_1 \qquad \Delta s_{1,6} = \alpha^6 + \alpha^5$$
$$\Delta s_{1,5} = \alpha^6 \qquad \Delta s_{1,6} = \cancel{\alpha}^2 + 1 + \cancel{\alpha}^2 + \alpha + 1$$
$$\Delta s_{1,6} = \alpha$$

$$\Delta s_{1,7} = s_1 - \tilde{s}_7$$
$$r(\mathrm{x}) = \mathrm{x} + \mathrm{x}^2$$
$$r(\alpha) = \alpha + \alpha^2$$
$$\tilde{s}_7 = \alpha$$
$$\Delta s_{1,7} = \alpha^6 + \alpha^4$$
$$\Delta s_{1,7} = \cancel{\alpha}^2 + 1 + \cancel{\alpha}^2 + \alpha$$
$$\Delta s_{1,7} = \alpha + 1$$
$$\Delta s_{1,7} = \alpha^3$$

And,

$$r_a = r_3(\mathrm{x}) = \mathrm{x} + \mathrm{x}^3$$
$$r_b = r_4(\mathrm{x}) = \mathrm{x} + \mathrm{x}^2 + \mathrm{x}^3 + \mathrm{x}^5$$
$$r_c = r_5(\mathrm{x}) = \mathrm{x} + \mathrm{x}^2 + \mathrm{x}^3 + \mathrm{x}^6$$
$$r_d = r_6(\mathrm{x}) = \mathrm{x}^2 + \mathrm{x}^3$$
$$r_e = r_7(\mathrm{x}) = \mathrm{x} + \mathrm{x}^2$$

$$r_a^2 = r_3(\mathrm{x}^2)$$
$$r_a^2 = \mathrm{x}^2 + \mathrm{x}^6$$
$$r_a^2 = \cancel{\mathrm{x}}^2 + \cancel{\mathrm{x}}^2 + 1$$
$$r_a^2 = 1$$

$$r_b^2 = r_4(\mathrm{x}^2)$$
$$r_b^2 = \mathrm{x}^2 + \mathrm{x}^4 + \mathrm{x}^6 + \mathrm{x}^{10}$$
$$r_b^2 = \cancel{\mathrm{x}}^2 + \cancel{\mathrm{x}}^2 + \cancel{\mathrm{x}} + \mathrm{x}^2 + \cancel{1} + \cancel{\mathrm{x}} + \cancel{1}$$
$$r_b^2 = \mathrm{x}^2$$

$$r_c^2 = r_5(\mathrm{x}^2)$$
$$r_c^2 = \mathrm{x}^2 + \mathrm{x}^4 + \mathrm{x}^6 + \mathrm{x}^{12}$$
$$r_c^2 = \cancel{\mathrm{x}}^2 + \cancel{\mathrm{x}}^2 + \cancel{\mathrm{x}} + \cancel{\mathrm{x}}^2 + \cancel{1} + \cancel{\mathrm{x}}^2 + \cancel{\mathrm{x}} + \cancel{1}$$
$$r_c^2 = 0$$

$$r_d^2 = r_6(\mathrm{x}^2)$$
$$r_d^2 = \mathrm{x}^4 + \mathrm{x}^6$$
$$r_d^2 = \mathrm{x}^2 + \mathrm{x} + \mathrm{x}^2 + 1$$
$$r_d^2 = \mathrm{x} + 1$$
$$r_d^2 = \mathrm{x}^3$$

$$r_e^2 = r_7(\mathrm{x}^2)$$
$$r_e^2 = \mathrm{x}^2 + \mathrm{x}^4$$
$$r_e^2 = \cancel{\mathrm{x}}^2 + \cancel{\mathrm{x}}^2 + \mathrm{x}$$
$$r_e^2 = \mathrm{x}$$

In the other word:

$r_a(\mathrm{x}) = 1$ [For the 1st Δs column] $r_b(\mathrm{x}) = \mathrm{x}$ [For the 2nd Δs column]

$r_c(\mathrm{x}) = 0$ [For the 3rd Δs column] $r_d(\mathrm{x}) = \mathrm{x}^5$ [For the 4th Δs column]

$r_e(\mathrm{x}) = \mathrm{x}^4$ [For the 5th Δs column]

$$r_a(\mathrm{x}) = 1 \qquad r_b(\mathrm{x}) = \mathrm{x} \qquad r_c(\mathrm{x}) = 0$$
$$r_a(\mathrm{x}^2) = (1)^2 \qquad r_b(\mathrm{x}^2) = (\mathrm{x})^2 \qquad r_c(\mathrm{x}^2) = (0)^2$$
$$r_a(\mathrm{x}^2) = 1 \qquad r_b(\mathrm{x}^2) = \mathrm{x}^2 \qquad r_c(\mathrm{x}^2) = 0$$

$$r_d(\mathrm{x}) = \mathrm{x}^5$$
$$r_d(\mathrm{x}^2) = (\mathrm{x}^5)^2$$
$$r_d(\mathrm{x}^2) = \mathrm{x}^{10}$$
$$r_d(\mathrm{x}^2) = \mathrm{x}^3(\mathrm{x}^7)$$
$$r_d(\mathrm{x}^2) = \mathrm{x}^3$$

$$r_e(\mathrm{x}) = \mathrm{x}^4$$
$$r_e(\mathrm{x}^2) = (\mathrm{x}^4)^2$$
$$r_e(\mathrm{x}^2) = \mathrm{x}^8$$
$$r_e(\mathrm{x}^2) = \mathrm{x}(\mathrm{x}^7)$$
$$r_e(\mathrm{x}^2) = \mathrm{x}$$

$$\Delta s_{2,3} = s_2 + \tilde{s}_3$$
$$\Delta s_{2,3} = s_2 + r_a(\alpha^2)$$
$$\Delta s_{2,3} = \alpha^5 + 1$$
$$\Delta s_{2,3} = \alpha^2 + \alpha + \cancel{1} + \cancel{1}$$
$$\Delta s_{2,3} = \alpha^2 + \alpha$$
$$\Delta s_{2,3} = \alpha^4$$

$$\Delta s_{2,4} = s_2 + \tilde{s}_4$$
$$\Delta s_{2,4} = s_2 + r_b(\alpha^2)$$
$$\Delta s_{2,4} = \alpha^5 + \alpha^2$$
$$\Delta s_{2,4} = \cancel{\alpha}^2 + \alpha + 1 + \cancel{\alpha}^2$$
$$\Delta s_{2,4} = \alpha + 1$$
$$\Delta s_{2,4} = \alpha^3$$

$$\Delta s_{2,5} = s_2 + \tilde{s}_5$$
$$\Delta s_{2,5} = s_2 + r_c(\alpha^2)$$
$$\Delta s_{2,5} = \alpha^5 + 0$$
$$\Delta s_{2,5} = \alpha^5$$

$$\Delta s_{2,6} = s_2 + \tilde{s}_6$$
$$\Delta s_{2,6} = s_2 + r_d(\alpha^2)$$
$$\Delta s_{2,6} = \alpha^5 + \alpha^3$$
$$\Delta s_{2,6} = \alpha^2 + \cancel{\alpha} + \cancel{1} + \cancel{\alpha} + \cancel{1}$$
$$\Delta s_{2,6} = \alpha^2$$

$$\Delta s_{2,7} = s_2 + \tilde{s}_7$$
$$\Delta s_{2,7} = s_2 + r_e(\alpha^2)$$
$$\Delta s_{2,7} = \alpha^5 + \alpha$$
$$\Delta s_{2,7} = \alpha^2 + \cancel{\alpha} + 1 + \cancel{\alpha}$$
$$\Delta s_{2,7} = \alpha^2 + 1$$
$$\Delta s_{2,7} = \alpha^6$$

$$\begin{bmatrix} 1 & \alpha^4 \\ 1 & \alpha \end{bmatrix} \begin{bmatrix} \gamma_1 \\ \gamma_2 \end{bmatrix} = \begin{bmatrix} \alpha^6 & \alpha^2 & \alpha^5 & \alpha^6 & \alpha & \alpha^3 \\ \alpha^5 & \alpha^4 & \alpha^3 & \alpha^5 & \alpha^2 & \alpha^6 \end{bmatrix}$$

The inverse of the matrix $\begin{bmatrix} 1 & \alpha^4 \\ 1 & \alpha \end{bmatrix}$ is:

$$\begin{bmatrix} \alpha^6 & \alpha^2 \\ \alpha^5 & \alpha^5 \end{bmatrix}$$

So, the solution matrix (3.2) will be:

$$\begin{bmatrix} \gamma_1 \\ \gamma_2 \end{bmatrix} = \begin{bmatrix} \alpha^4 & \alpha^5 & 1 & \alpha^4 & \alpha^5 & \alpha^4 \\ \alpha^6 & \alpha^6 & 1 & \alpha^6 & \alpha^2 & \alpha^2 \end{bmatrix}$$

Notice that the elements which contained in the first column are not in $GF(2)$, thus, adding a column to the first one to get elements $\in GF(2)$.

So, adding the second column to the first column:

$$\alpha^4 + \alpha^5 = \cancel{\alpha}^2 + \cancel{\alpha} + \cancel{\alpha}^2 + \cancel{\alpha} + 1$$

$$\alpha^4 + \alpha^5 = 1$$

$$\alpha^6 + \alpha^6 = \cancel{\alpha}^2 + \cancel{1} + \cancel{\alpha}^2 + \cancel{1}$$

$$\alpha^6 + \alpha^6 = 0$$

Now, we got $[1,0]^T$ which they $\in GF(2)$.

Then, we sum the first β_{2t} elements plus the added column(s) to get the errors:

$$\underbrace{1 + \alpha^4}_{first\ \ 2t} + \underbrace{\alpha^5 + \alpha^6}_{added\ \ column} = 1 + \alpha^2 + \cancel{\alpha} + \cancel{\alpha}^2 + \cancel{\alpha} + \cancel{1} + \cancel{\alpha}^2 + \cancel{1}$$

$$\underbrace{1 + \alpha^4}_{first\ \ 2t} + \underbrace{\alpha^5 + \alpha^6}_{added\ \ column} = \underbrace{1 + \alpha^2}_{the\ position\ of\ \ the\ two\ errors}$$

$$c = (1,1,0,1,0,0,0)$$

$$r = (0,1,1,1,0,0,0)$$

$$\beta = (1, \alpha^4, \alpha^2, \alpha^5, \alpha^6, \alpha, \alpha^3)$$

Since the elements are not in GF(2) in the first column but the summation of the first two columns is $[1,0]^T$, so the errors are in locations 1 and α^2. Therefore, the two errors in positions 1 and α^2 are corrected to get the original codeword.

3.2 The Second Approach of Soft Decision Decoding of BCH Codes Using Error Magnitudes

There is another way to find the errors positions and this method is easier than the previous one. The previous solution is complex and quite hard to achieve on matlab. for less hardware complexity, In place of the whole codeword, the soft information may assist the decoder to select the minimum bits for reliability that decoded by the soft decoder. Depending on the concepts of [26], there are three main procedures: syndrome calculator, error locator evaluator, and error magnitude solver that proposed in soft BCH decoder.

When there are soft inputs, the decoder selects those whose inputs are not reliable and examines its error locators to come up with the following set of error locators $\beta = [\beta_{l_1}, \beta_{l_2}, ..., \beta_{l_{2t}}]^T$. Additionally, the error location set appears as follows $L = [l_1, l_2, ..., l_{2t}]^T$. It may be computed using β because β_{l_i} serves as the error locator of the l_i location and $\beta_{l_i} = \alpha^{l_i}$.

The relationship between β and the syndrome vector $S = [S_1, S_2, ..., S_{2t}]^T$ can represented as follows:

$$\begin{bmatrix} \beta_1 \beta_2 ....\beta_{2t} \\ \beta_1^2 \beta_2^2 ....\beta_{2t}^2 \\ \beta_1^3 \beta_2^3 ....\beta_{2t}^3 \\ .... \\ .... \\ \beta_1^{2t} \beta_2^{2t} ....\beta_{2t}^{2t} \end{bmatrix} \begin{bmatrix} \gamma_1 \\ \gamma_2 \\ \gamma_3 \\ .. \\ .. \\ \gamma_{2t} \end{bmatrix} = \begin{bmatrix} s_1 \\ s_2 \\ s_3 \\ .. \\ .. \\ s_{2t} \end{bmatrix} \tag{3.3}$$

where γ_i is the error magnitude in the l_i th location.

Noteworthy is the fact that BCH codes have an error magnitude set that is valid as illustrated $\Gamma = [\gamma_1, \gamma_2, ..., \gamma_{2t}]^T$ which should be zeros and ones (i.e. binary vectors). If the position of l_i is the correct error position, γ_i is 1; otherwise, γ_i is 0. The $2t \times 2t$ matrix in (4) is serves as the error locator of the β matrix. The approximate codeword polynomial $\hat{C}(x)$ may be acquired through XORing γ_i with R_{j_i} . In the algorithm 2t locators move towards L in a way that corrects 2t errors. The difference of S and the multiple of β and Γ is a discrepancy vector $\Delta = [\gamma_1, \gamma_2, ..., \gamma_{2t}]^T$ illustrated as follows:

$$\Delta = \beta \text{ X } \Gamma + S \tag{3.4}$$

Observe that the two operations in (3.3) and (3.4) are under $GF(2^m)$. Obviously, the valid Γ can be computed for making Δ be all zeros (vector of zeros) in case of whole the errors are occurred in the position vector L, otherwise, the decoding operation fails to correct the errors and computes Γ as a non-binary vector.

Solving the previous example by this way is easier and less complex comparing with the previous algorithm.

3.2.1 Numerical example

Same steps in the previous example are done to generate the code $c(x) = (1\ 1\ 0\ 1\ 0\ 0\ 0)$ with the received values

$r = (0.218,\ 2.228,\ -2.8,\ -0.3,\ -0.5,\ -1.8,\ -2.2)$,

But we introduce an artificial error in the 4th bit.

β and syndromes $(s_1 .....s_{2t})$ are computed as the previous example as shown

$$\beta = (1\ \alpha^3\ \ \alpha^4\ \ \alpha^5\ \ \alpha^6\ \ \alpha\ \ \alpha^2)$$
$$s_1 = 1 + \alpha$$
$$s_1 = \alpha^3$$
$$s_2 = 1 + \alpha^2$$
$$s_2 = \alpha^6$$

Then, put down all the elements in matrices in (3.4) as follows:

$$\Delta = \beta\ \mathrm{X}\ \Gamma + S$$
$$\Delta = \begin{bmatrix} 1 & \alpha^3 \\ 1 & \alpha^6 \end{bmatrix} \begin{bmatrix} \gamma_1 \\ \gamma_2 \end{bmatrix} + \begin{bmatrix} \alpha^3 \\ \alpha^6 \end{bmatrix}$$

For the equation above, we try all the possible γ_1, γ_2 values such that Δ equals to zero. If it equals to zero that means $[\gamma_1\ \gamma_2]^T$ represent the error locations (i.e. every 1 in this matrix means there is an error in this location and otherwise). For instance,

$$\Delta = \begin{bmatrix} 1 & \alpha^3 \\ 1 & \alpha^6 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} + \begin{bmatrix} \alpha^3 \\ \alpha^6 \end{bmatrix} \qquad\qquad \Delta = 0$$

That means the error occurs in the α^3 position. The error corrected by XORing Γ vector with the received vector to get the original codeword.

3.3 Simulation Results

In this section, we present the simulation results done for LDPC and BCH codes.

3.3.1 LDPC code simulation

In this Figure, it is shown the result of the LDPC performance which it is used in DVB-S2 blocks.

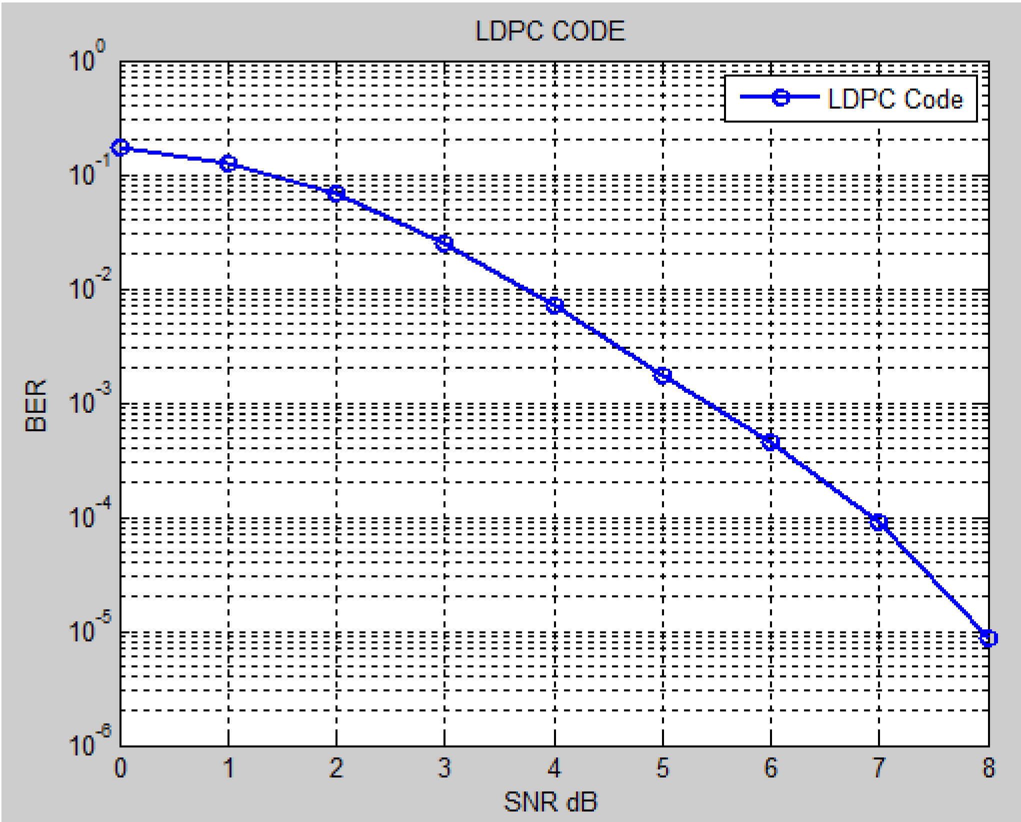

Figure 5: LDPC Code Performance

Fig. 5 shows LDPC code presentation with rate ½. The size of code word is N=1032 and data bits is K=516 and by using BPSK modulation over AWGN channel, with 20 repetition and signal to noise ratio change from 0 up to 8 dB.

3.3.2 BCH code simulation

Fig. 6 shows the performance of the BCH (255,239,2) codes over AWGN channel using error magnitude bases soft decision decoding algorithm. It is seen from the graph that error free transmission is possible at an SNR value of 7.5dB

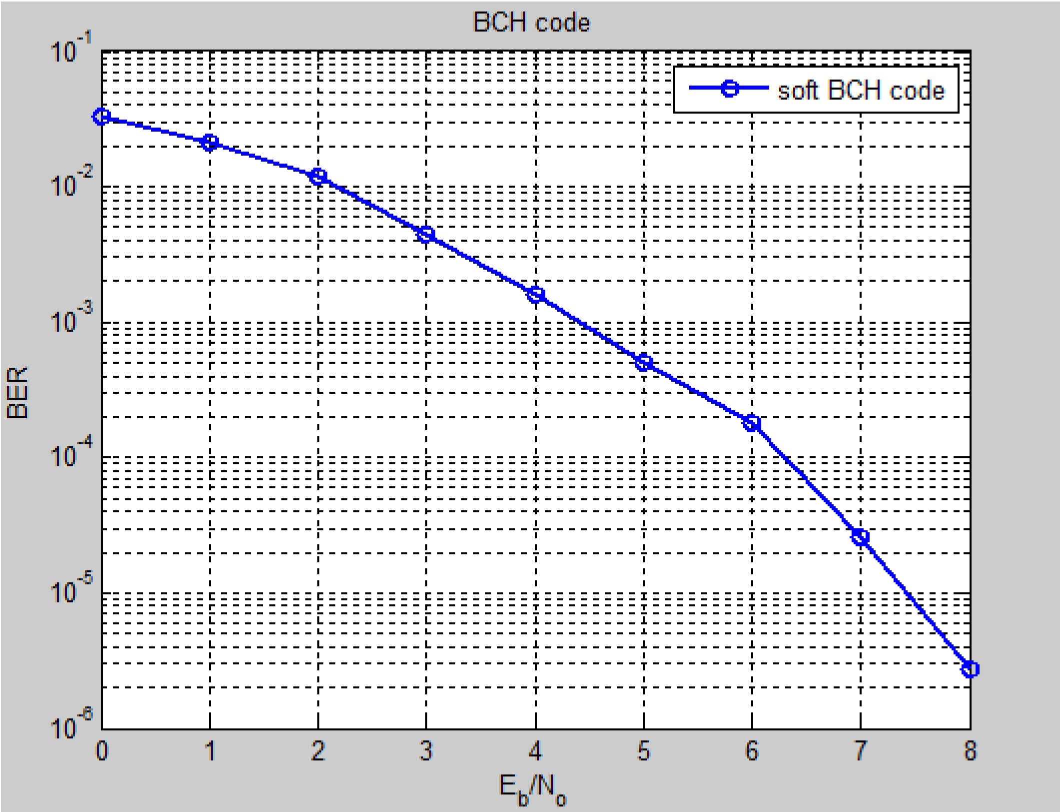

Figure 6: BCH Code Performance

CHAPTER 4

CONCLUSION AND FUTURE WORK

4.1 Conclusion

In this thesis work, efficient decoding algorithms are used to decode block codes used in DVB-S2 system. LDPC codes are obtained using the Richardson and Urbanke [6] encoding method which is used in the literature for a long time. In DVB-S2 standard there are two types of forward error correction frame structures which are short and long frames. In this thesis study short frame structures are employed. The size of the codewords is chosen as 1024. The dataword size is taken as 512. LDPC code rate is ½. For the decoding of LDPC codes soft decision based LDPC decoding algorithm called as Belief propagation algorithm is employed. An iterative decoding approach has been followed during the application of the Belief propagation algorithm for the soft decision decoding of LDPC codes. During the decoding operation 20 iterations are performed. For the simulations, signal to noise ratio range is taken from 0dB to 8dB and bit error graphs are obtained. From the simulation results it is seen that LDPC code with the mentioned parameters above achieves a BER of 10^{-5} at 8dB. For BCH codes we used a new approach recently introduced in the literature, which is the error magnitude based soft decision algorithm. Currently the present DVB-S2 system does not use soft decoding algorithms, however, soft decoding algorithms are better in BER performance than the hard decoding algorithms. The BCH code used in our simulation has the parameters BCH(255, 239, 2). It is seen from the simulation results that a BER of 10^{-5} is achieved at 7.5dB for the BCH(255, 239, 2) codes.

4.2 Future Work

Our work here presents an improved architecture for DVB-S2 systems. Employing soft decoding algorithms for LDPC and BCH codes enables us for iterative decoding of this concatenated system. To achieve superior performance in communication systems it is obvious that most of the future communication systems will employ concatenated systems using iterative decoding. For a continuation of the work presented in this thesis, we can state that a hardware implementation of the decoding algorithms presented in this thesis can be performed. In addition, the iterative decoding of the concatenated system employing BCH and LDPC codes can be performed using the presented algorithms in the thesis. The iterative system employing BCH and LDPC codes can employ Belief propagation algorithm. And complexity reduction of the joint system employing LDPC and BCH codes employing Belief propagation algorithm is another future research area.

REFERENCES

1. **Clark J. M., Paivio, A., (1991),** *“Dual Coding Theory and Education”*, Educational Psychology Review, vol. 3, no. 3, pp. 149-210.

2. **Chauhan N., Yadav P., Kumari P., (2014),** *“Error Detecting and Error Correcting Codes”*. vol. 29, no. 2, pp. 147-160.

3. **Shannon C. E., (2001),** *“A Mathematical Theory of Communication”,* ACM SIGMOBILE Mobile Computing and Communications Review, vol. 5, no. 1, pp. 3-55.

4. **Darnell M., (1985),** *“Error Control Coding: Fundamentals and Applications”,* In IEE Proceedings F-Communications, Radar and Signal Processing, vol. 1, no. 132, pp. 68-74.

5. **Hathal H. M., Abdulhussein R. A., Ibrahim S. K., (2014),** *“Lyapunov Exponent Testing for AWGN Generator System”,* Communications and Network, vol. 6, no. 4, pp. 201.

6. **Rappaport T. S., Reed J. H., Woerner B. D, (1996),** *“Position Location Using Wireless Communications on Highways of the Future”,* Communications Magazine, IEEE, vol. 34, no. 10, pp. 33-41.

7. **Felzenszwalb P. F., Huttenlocher D. P., (2006),** *“Efficient Belief Propagation for Early Vision”,* International Journal of Computer Vision, vol. 70, no. 1, pp. 41-54.

8. **Richardson T. J., Urbanke R. L., (2001),** *“ The Capacity of Low-Density Parity-Check Codes Under Message-Passing Decoding”*, Information Theory, IEEE Transactions on, vol. 47, no. 2, pp. 599-618.

9. **Morello A., Mignone V., (2006),** *“DVB-S2: The Second Generation Standard for Satellite Broad-Band Services”,* Proceedings of the IEEE, vol. 94, no. 1, pp. 210-227.

10. **Yedidia J. S., Freeman W. T., Weiss Y., (2003),** *“Understanding Belief propagation and Its Generalizations”,* Exploring Artificial Intelligence in the New Millennium, vol. 8, pp. 236-239.

11. **Gallager R. G., (1962),** *“Low-Density Parity-Check Codes”,* Information Theory, IRE Transactions on, vol. 8, no. 1, pp. 21-28.

12. **MacKay D. J., (1999),** *“Good Error-Correcting Codes Based on Very Sparse Matrices”,* Information Theory, IEEE Transactions on, vol. 45, no. 2, pp. 399-431.

13. **Azarbad B., Sali A., Ali B. M., Karim H. A., (2011),** *“Study of BER in DVB-S2 Satellite Implemented in Matlab*”, In Space Science and Communication (IconSpace), IEEE International Conference on, pp. 221-224.

14. **Hocquenghem A., (1959),** *“Codes Correcteurs D'erreurs”,*Chiffres, vol. 2, no. 116, pp. 147-156.

15. **Bose R. C., Ray-Chaudhuri D. K., (1960),** *“On a Class of Error Correcting Binary Group Codes”,* Information and Control, vol. 3, no. 1, pp. 68-79.

16. **Imai H., Hirakawa S., (1977),** *“A New Multilevel Coding Method Using Error-Correcting Codes”,* Information Theory, IEEE Transactions on, vol. 23, no. 3, pp. 371-377.

17. **Urard P., Paumier L., Georgelin P., Michel T., Lebars V., Yeo E., Gupta B., (2005),** *“A 135Mbps DVB-S2 Compliant Codec Based on 64800-Bit LDPC and BCH Codes”,* (ISSCC paper 24.3) In Proceedings of the 42nd Annual Design Automation Conference, pp. 547-548.

18. **Morello A., Mignone V., (2006),** *“DVB-S2: The Second Generation Standard for Satellite Broad-Band Services”*, Proceedings of the IEEE, vol. 94, no. 1 pp. 210-227.

19. **Urard P., Yeo E., Paumier L., Georgelin P., Michel T., Lebars V., Gupta B., (2005),** *“A 135Mb/s DVB-S2 Compliant Codec Based on 64800b LDPC and BCH Codes”,* In Solid-State Circuits Conference, Digest of Technical Papers. ISSCC. IEEE International, pp. 446-609.

20. **Bates S., Gunthorpe L., Pusane A. E., Chen Z., Zigangirov K., Costello D. J., (2006),** *"Decoders for Low-Density Parity-Check Convolutional Codes with Large Memory",* In Circuits and Systems, ISCAS, Proceedings. IEEE International Symposium on pp. 4-10.

21. **Benedetto S., Montorsi G., (1996),** *"Unveiling Turbo Codes: Some Results on Parallel Concatenated Coding Schemes",* Information Theory, IEEE Transactions on, vol. 42, no. 2, pp. 409-428.

22. **Morello A., Mignone V., (2006),** *"DVB-S2: The Second Generation Standard for Satellite Broad-Band Services",* Proceedings of the IEEE vol. 94, no. 1, pp. 210-227.

23. **Hu X. Y., Eleftheriou E., (2004),** *"Binary Representation of Cycle Tanner-Graph GF (2 b) Codes",* In Communications, 2004 IEEE International Conference on, vol. 1, pp. 528-532.

24. **Vardy A., Be'ery Y., (1994),** *"Maximum-Likelihood Soft Decision Decoding of BCH Codes",* Information Theory, IEEE, vol. 40, no. 2, pp. 546-554.

25. **Reid III W. J., Joiner L. L., Komo J. J., (1997),** *"Soft Decision Decoding of BCH Codes Using Error Magnitudes",* In Information Theory, Proceedings, IEEE International Symposium on, pp. 303.

26. **Lin Y. M., Chen C. L., Chang H. C., Lee C. Y., (2010),** *"A 26.9 K 314.5 Mb/s Soft (32400, 32208) BCH Decoder Chip for DVB-S2 System",* Solid-State Circuits, IEEE Journal of, vol. 45, no. 11, pp. 2330-2340.

CURRICULUM VITAE

PERSONAL INFORMATION

Date and Place of Birth: 11.02.1990

Marital Status: Single

Phone: 05514084186

Email: ibrahimsami_100@yahoo.com

EDUCATION

Degree	Institution	Year of Graduation
M.Sc.	Cankaya University	2015
B.Sc.	Al-Mustansiriya University	2012
High School	Al-Kindy High School	2008

WORK EXPERIENCE

Year	Place	Enrollment
2011- Present	Umniah Mobile	Registration Engineer

FOREIN LANGUAGES

Arabic, English, Beginner Turkish.

HOBBIES

Football, Swimming, Travelling.