Quasi-Cyclic Low-Density Parity-Check Codes: Applications and Iterative Decoding Hardware Implementations

Thesis · March 2012

DOI: 10.13140/RG.2.2.26202.90564

Waheed Ullah
University of the Witwatersrand
22 PUBLICATIONS **139** CITATIONS

中图分类号 TN911.2　　　　　　　　　　　　　　论文编号 1028704 12-S144
学科分类号 081001

硕士学位论文

准循环低密度奇偶校验码的应用和迭代译码器硬件实现

研究生姓名	**Waheed Ullah**
学科、专业	通信与信息系统
研究方向	数字通信
指导教师	仰枫帆 教授

南京航空航天大学

研究生院 电子信息工程学院

二〇一一年十二月

Nanjing University of Aeronautics and Astronautics
The Graduate School

College of Electronics and Information Engineering

Quasi-Cyclic Low-Density Parity-Check Codes: Applications and Iterative Decoding Hardware Implementations

A Thesis in

Communication and Signal Processing

By

Waheed Ullah

Advised by

Prof. Dr. Yang Fengfan

Submitted in Partial Fulfillment

Of the Requirements

For the Degree of

Master of Engineering

December 2011

承诺书

本人声明所呈交的博/硕士学位论文是本人在导师指导下进行的研究工作及取得的研究成果。除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得南京航空航天大学或其他教育机构的学位或证书而使用过的材料。

本人授权南京航空航天大学可以将学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存、汇编学位论文。

（保密的学位论文在解密后适用本承诺书）

作者签名：

日　　期：

摘 要

纠错编码的使用，使得在噪声信道中即使采用低传输功率的数字传输系统的性能亦可得到了明显的改善。低密度奇偶检验码的使用已被证实能给予数字系统的传输提供最优解，但它的复杂性也是实际系统应用中的最主要的障碍。原始的 LDPC 码的和积(Sum-Product)译码算法是很有效的，但译码处理时间亦较长。因此我们在寻求复杂性和译码性能之间的折衷付出了很多努力。如果在规模动态且系统又复杂时，简化的 LDPC 码最小和(min-sum)译码算法给出了最接近最优性能的近似方法。本文工作为了以下目的

1）了解复杂性和译码性能的最佳平衡点。

2）为了改进性能找寻找数学模型。

3）学习和研究 LDPC 码的高效硬件实现机理。

4）在 MIMO 和协作通信系统中的使用 QC-LDPC 码，并研究和分析其系统性能。

仿真结果表明改进和优化方法能有效提高 LDPC 码的性能。这些被优化的 LDPC 码同样可应用于 MIMO 系统中并显著改善的其系统性能。

关键字：和积译码算法，最小和译码算法,归一化译码算法，偏移译码算法，Tanner 图，准循环低密度奇偶校验码，MIMO，协作通信

(Translated by Miss Wu Xiaohua and approved by Prof. Yang Fengfan)

Abstract

With the use of error correction coding, a significant improvement in the data transmission with low transmitter power over a noisy channel is achieved .The use of low density parity check codes has proven to give the optimum solutions for the data transmission but its complexity is the main hurdle in its application to practical systems .The original LDPC sum product (SP) algorithms is the most powerful but need much time to process. So all efforts are being made to give optimum solution between the complexity and performance. The simplified form of LDPC called min-sum gives close approximation if it is scaled dynamically but again makes it complex. The aim is to

1) Investigate the optimum tradeoff for complexity and performance
2) Finding mathematical model for improved performance
3) Studying hardware efficient mechanism for implementation
4) Use in the communication systems like MIMO and cooperative communication and investigating its behavior and analyzing the performance

The simulation results showed the improved and optimized normalization technique for LDPC codes. The same has been applied to MIMO and showed significantly improved performance.

Keywords: Sum product LDPC. Min sum LDPC, Normalized min sum , offset min sum decoding, Tanner graph , QC LDPC, MIMO, Cooperative communication ,SPA LDPC

Table of Contents

List of Figures

List of Tables

Abbreviations

σ^2	Noise variance
μ	Mean value of noise
AWGN	additive white Gaussian Noise
E_b	Energy per bit
SNR	Signal to noise ratio
C	Channel Capacity
N_0	Noise Power
FEC	Forward Error Correction
DVB	Digital video braodcasting
QC	Quasi cyclic
LDPC	Low density parity ceheck
d_v	number of 1s in a column of matrix
d_c	number of 1s in a row of a matrix
LLR	Log likelihood ratio
SPA	sum product algorithm
MIMO	Multiple input multiple output

Chapter 1 Introduction

1.1 Historical Perspective and State of the Art

After the first successful data transmission, the greed for information sending with high rate and high reliability increased day by day. After the advent of first wireless system, the question always attracts scientists and engineers; “**How to transmit the data reliably over a noisy channel?** .Noise is the main limitation and hurdle in communication and data storage systems. The main aim of a communication system is to transmit information reliably over a noisy channel. The available amount of transmitter power and bandwidth are the major constraints in the design of a communication system. The channel is subject to various types of noise, interference and distortion due to hardware imperfections, or physical limitations. Also some communication systems have limitation on transmitter power. All these may lead to errors. Consequently we may need some form of error control encoding to recover the information. This greed of human to transmit and receive data without any error or with minimum error gives birth to “**Erro**r **Control Coding”**.

Claude E.Shannon (1916-2001), a mathematician then working at Bell Laboratories, is generally regarded as the father of Information Theory. Prior to Shannon, it was believed that in order to achieve higher reliability in communications over noisy channels, it was necessary to keep increasing the transmit power [1]. In 1948, Claude E.Shannon presented his historical ground breaking theory in two parts[2], this gave birth to a new subject called “Information Theory”. He showed that even with a noisy channel, still there exists a way to encode a data in such a way that have an arbitrary good chance of being transmitted safely. This was a theoretical limit and method but no practical coding technique was provided through which this accuracy could be achieved. This practical code was found by his colleague at Bell Labs, Richard Hamming (1915-1998). Hamming constructed a code which

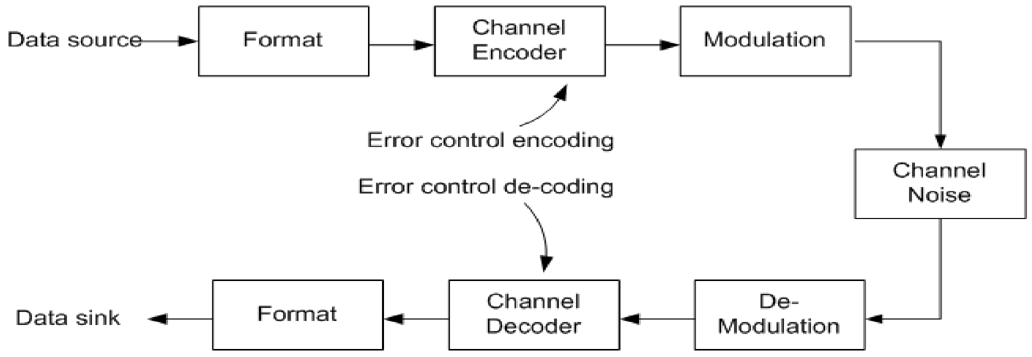


Figure 1.1 Typical Flow Diagram for Error Control Coding in a Communication System

added three additional "parity" bits to four information bits. The Hamming code was the first result in the field that is now known as Coding Theory, the object of which is to find good codes that are also easy to implement[1]. Figure 1.1 shows the typical communication system with error control coding and figure 1.2 shows the gradual development of the different codes with passage of time.

Coding research is focused mainly on two areas, *algebraic codes* and *trellis codes*. Codes like Reed-Solomon and BCH build algebraic structure into the code such that the code can be decoded using computationally efficient algorithms for solving systems of equations. Trellis codes are easily decoded using trellis-based algorithms such as the Viterbi algorithm[3]. The research on coding was greatly revolutionarised by the invention of Turbo iterative decoding[4] which shows performance close to the theoretical limit set by Shannon theorem.

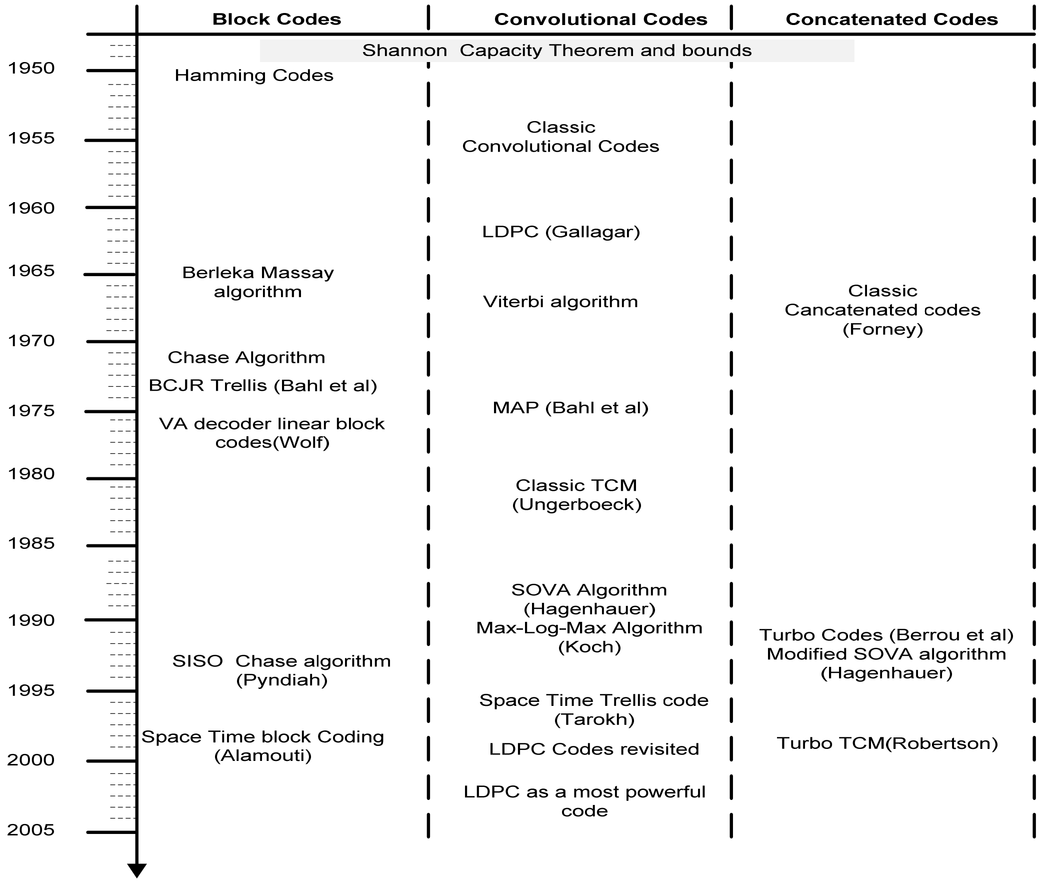

Block Codes
Convolutional Codes
Concatenated Codes
Shannon Capacity Theorem and bounds
1950
Hamming Codes
1955
Classic Convolutional Codes
1960
LDPC (Gallagar)
1965
Berleka Massay algorithm
Viterbi algorithm
Classic Cancatenated codes (Forney)
1970
Chase Algorithm
BCJR Trellis (Bahl et al)
1975
MAP (Bahl et al)
VA decoder linear block codes(Wolf)
1980
Classic TCM (Ungerboeck)
1985
SOVA Algorithm (Hagenhauer)
1990
Max-Log-Max Algorithm (Koch)
Turbo Codes (Berrou et al)
Modified SOVA algorithm (Hagenhauer)
SISO Chase algorithm (Pyndiah)
1995
Space Time Trellis code (Tarokh)
Space Time block Coding (Alamouti)
LDPC Codes revisited
Turbo TCM(Robertson)
2000
LDPC as a most powerful code
2005

Figure 1.2 Historical View of Channel Coding Algorithms

Low density parity check (LDPC) codes , a new coding scheme developed by Gallager[5] and then rediscovered by Macky and Neal[6, 7] showed great performance and have taken the attentions of all the scientists and engineers working on channel coding. The IEEE Information Theory Society recognized major contributions in this area through the award of the 2002 paper prize jointly to T.J. Richardson et al[8] and M.G. Luby et al[9].Both of these papers appeared in the February 2001 issue of the IEEE Transactions on Information Theory [1].

1.2 Trends in Communication Technology

History of communication and signaling can be traced back to the ancient civilization of Rome, Egypt and China. Different methods were adapted for messages between distant points. In some era and area; fire, smoke and flags were the main communication sources. The communication systems and methods were revolutionized by invention of the electromagnet. In 1825, British inventor William Sturgeon (1783-1850) revealed the invention that laid the foundations for a large scale evolution in electronic communications. The work was further carried by Maxwell in 1867 and then the first radio

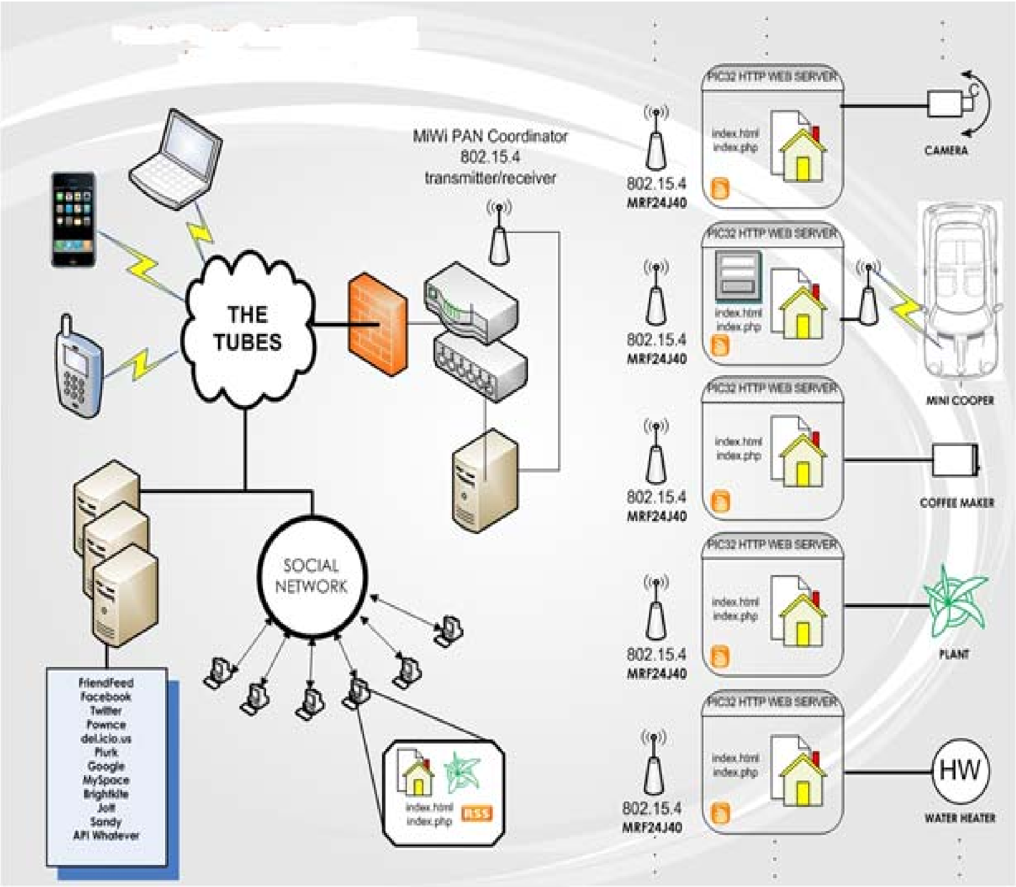


Figure 1.3 Every where, all the time (Communication in Daily life)

invented by Marconi in 1897.

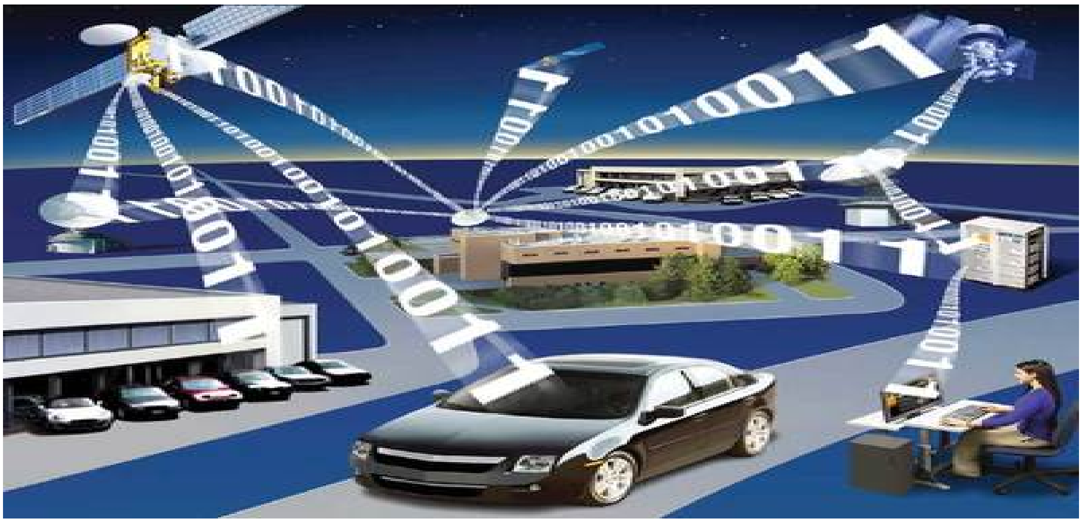


Figure 1.4 Data Transmission over wireless channel

A question arises "What is the driving force behind all these advancement?" The obvious answer is the never ending wish list of human beings for more and more. They want to perform the daily life tasks in a quick possible way and undoubtedly communication technology plays an important role in all these situations. Looking at emerging trends and standards in communication systems, one can easily perceive that all these areas exactly affect human being daily life. Business, education, social life and defense are all the broad areas which are directly influenced by the communication technology. On individual level, making calls, and messaging on phone, connected to internet, watching movies and listing music and video conferencing and navigating locations are the important today's applications and demands.

With all the applications, today's demand for communication systems is **"Every where, all the time"** along with small size, low power and low cost as shown in figure 1.3. Different communication standards like GSM and WCDMA (3.5G and now 4G), WiMax etc, evolve to full fill the demands. With all these standards, there is always a question "What is the best way to send data over noisy channel reliably" as shown in figure 1.4**.** The answer is the error correction methodologies developed with increasing demands for reliable data transmission.

Error correction techniques play a major and important role in communication systems and storage devices to increase the transmission reliability and achieve a better error correction performance with less power. The trends and developments in the area of error correcting codes are shown in the Figure 1.2. Among the error correction coding, LDPC codes, due to its excellent performance, are adopted in several communication standards such as DVB-S2, IEEE802.16e, IEEE802.3an (10BASE-T) and also for inter-satellite communication .Other growing application of LDPC codes are in MIMO and coded cooperative communication.

1.3 Theoretical and Practical LDPC Codes: Systematic Overview

LDPC codes are gaining attention due to its excellent performance but still there are a lot of challenges in its hardware implementation. Due to high complexity, LDPC codes generally offer high processing delays which make it impractical for many applications. Efforts have been made to keep a good tradeoff between the complexity and performance. The LDPC codes that offer good performance but at the same time high complexity, are referred here as theoretical codes in a section below as theoretical approach. Others LDPC codes with less complexity at the cost of small performance loss are referred as practical codes in a section of Practical approach.

1.3.1 High Performance, High Complexity LDPC Codes: Theoretical Approach

LDPC codes construction require parameters such as row and column weights, rate, girth and code length. LDPC codes are classified into two types of constructions. The first one is random construction that offers flexibility in term of design and construction. This method has been used in LDPC codes construction as presented by Macky & Neal and S.Chung et al .The LDPC codes has shown the capacity approaching[6-8, 10-12] high performance but at the cost of high complexity which makes it unsuitable for practical applications and hardware implementation.

1.3.2 Better Performance, Low Complexity LDPC Codes: Practical Approach

LDPC parity check matrix can be obtained by cyclic shift [13] or circular permutation matrices [14] which facilitate the hardware implementation and still offer better performance. The following methods are adopted for getting the Parity Check matrix from circulants.

1.3.2.1 Quasi Cyclic LDPC Code Design: The Quasi cyclic LDPC codes can be obtained by the following methods.

i. **Identity matrix:** The parity check matrix from Identity circulants. This is the usual and easy approach to construct the QC LDPC codes.

ii. **Q matrix :** An $n \times n$ circulant permutation matrix is called Q-matrix[15] if the number of 1's is only one in every column, every row and every diagonal of it.

iii. **D matrix :** An $n \times n$ circulant permutation matrix is called D-matrix if the number of 1's is only one in every column and every row of it **[16]**. The D-vector describing D-matrix is composed of arithmetic progression. The formula of general term of arithmetic progression is $a_n = a_1 + (n-1)b$.

iv. Progressive edge growth (PEG) : In the standard PEG algorithm [17], given the graph parameters, i.e., the number of symbol nodes, the number of check nodes, and the symbol-node-degree sequence, an edge-selection procedure is started such that the placement of a new edge on the graph has as small impact on the girth as possible.

v. Modified PEG to get Q and D matrices : In paper [18] , there are added two new parameters to PEG , i.e., the dimension of a circulant permutation matrix and permutation vector are introduced. Using this algorithm, the QC-LDPC codes based on D-matrix and Q-matrix , outperform the QC-LDPC code based on an identity matrix and PEG random LDPC code, which suggests that D-matrix and Q-matrix are more suitable to be used in QC-LDPC codes than identity matrix as it exhibits better performance and have hardware friendly structures for practical application. The idea of permutation vector is originally introduced. The main principle of this method is to optimize the placement of a new edge to maximize the local girth length under the permutation vector constraint.

vi. Row Division method : This method is to construct large girth QC LDPC[19, 20] codes and cut down the hardware implementation cost . The row groups are paired two times the row weight, which has the complexity as compared to the connection of individual columns and rows. The new codes offer more flexibility in term of girth, code rates and codeword length.

1.3.2.2 Quasi Cyclic LDPC Encoder Implementation:

i. Shift Register based Implementation: QC–LDPC codes have encoding advantage over conventional LDPC codes and their encoding can be carried out by shift register[21] with complexity linearly proportional to the number of parity bits of the code .Additionally QC-LPDC codes require less amount of memory as compared to the general LDPC codes, since their parity check matrices consist of the circulant permutation matrices or the zero matrices.

ii. RAM based Implementation: RAM (instead of shift register) based QC-LDPC encoder[22] for hardware resource saving and high data throughput (in terms of data input to output) .Less XOR & AND gates due to equal or more zeros compared to 1's .Secondly this eliminates the need for additional CLB logic for the parallel to serial circuit , and enhances the data throughput

1.3.3 LDPC Decoding Algorithms

i. Sum Product Algorithm: This is also well know as the belief propagation algorithm invented by Gallager and then re-invented by Mackay & Neal .The sum product algorithm[5, 23-25] has shown best performance with highest complexity.

ii. Logarithmic Sum Product Algorithm: The logarithmic sum-product algorithm[25] is an enhanced version of the sum-product algorithm, introducing LLR (Logarithmic Likelihood Ratio), which reduces most multiplication to addition .

iii. Min Sum Algorithm : To simply belief propagation(BP) algorithm, min-sum algorithm[25-27] is introduced to reduce the complexity of the check node operation . The min-sum algorithm is in effect a simplified version of the logarithmic sum-product algorithm. It trades off precision for speed by eliminating the need for addition in the message update process, resulting in a possible increase in the number of iterations. Several modification has been made to improve the performance and convergence as in references [28-31].

1.3.4 LDPC Architecture

i. Parallel : This Architecture [32] gives extremely low power dissipation and high throughput but is not area efficient.

ii. Serial : In paper [33], the serial approach has adapted , stating that parallel approach leads to an extremely complex interconnect problem. One possibility to avoid the extremely complex interconnect problem is going for a sequential decoding machine that processes the input nodes in a linear order from the first bit-node to the last in every iteration. The results is that the complex interconnect can be solved in random access memory. This is called the *serial approach*. To reach the required throughput performance tens or more decoding machines can be used on one chip.

iii. Semi-Parallel : The semi-parallel decoding architecture[34] offers a better balance between throughput performance and hardware requirements, which is fairly advisable for QC- LDPC codes . This kind of architecture generally achieves a good trade-off between hardware complexity and decoding throughput.

iv. Minimum Semi-Parallel : This architecture[35] that there is a throughput/complexity gap between semi-parallel and serial decoders, which would be efficient and suitable for wireless applications. In order to exploit the gap between semi-parallel and serial decoders, a novel LDPC decoding architecture with a flexible inter-circulant time-sharing scheme of processing units is proposed in this paper. The architecture is advisable and competent for efficiency-demanding applications, such as wireless and mobile systems and portable devices. Practical implementation is one of the major issues of LDPC codes.

1.4 Challenges, Motivations and Research Objective

Due to the near Shannon limit performance[6, 7, 10, 11], LDPC has got popularity among the coding theory researchers. The main drawback of the LDPC code is the hardware complexity[32, 36-38] and long processing delay[39].The tradeoff between hardware complexity and processing delay[34, 40] have been achieved but still need some further improvement for implementing the algorithms more efficiently. While there has been much research on LDPC decoders and their VLSI implementations[41, 42], many difficulties to achieve requirements remain such as lower error floors[43, 44], reduced interconnect complexities[41], smaller die areas, lower power dissipation[45], and design re-configurability to support multiple code lengths and code rates[46] .

The motivation behind the thesis is today's demand for low-cost, low complexity but reliable and high throughput solutions in the digital communication technology. The invention of the re-configurable hardware (FPGA) has triggered the fast prototyping and hardware realization for complex solutions. Together with re-configurable hardware, high level simulation tools (e.g Matlab , C++) give the accurate performance evaluation for the novel algorithms with realistic channel models like AWGN, fading channels etc .

Low density parity check code is a powerful branch of error-correcting coding that has exhibited performances close to the Shannon limit--a theoretical maximum limit for channel capacity whilst offering a reasonable level of computational complexity. Ever since the rediscovery of LDPC codes; design, construction, efficient encoding and decoding, performance analysis, and applications of these powerful error-correction codes in digital communication and storage systems have always become the focal points of research.

Objective of this research can be summarized as follows:

- To investigate the current construction methods of LDPC codes based on both encoding and decoding
- To evaluate the performance and complexity of the LDPC codes and make the analysis and comparison
- To investigate new methods for improving the performance of the LDCPC codes
- To study and work on the QC-LDPC codes for FPGA
- To apply the LDPC codes to MIMO and coded cooperative communication Systems

1.5 Scope and the Thesis Organization

Low density parity check codes are becoming mandatory part of many communications systems and is taking place the old error correction techniques in not only the existing standards but also in newly evolved standards. This thesis covers the LDPC codes overview and its implementation and application. The thesis is organized as follows:

The first chapter gives the overall view of communication technologies and emerging standards. Chapter 2 gives the comprehensive and systematic over view of the communication system and LDPC encoding and decoding methodologies. Chapter 2 covers the channel characteristics, LDPC encoding techniques, LDP C decoding algorithms and its mathematical representation. The chapter also includes the variants of LDPC and the new proposed methods for performance improvement. Chapter 3 gives the hardware efficient encoding and decoding methods know quasi cyclic LDPC (QC LDPC) codes. Chapter 4 summarizes the hardware implementation of the LDPC codes and the efficient methods are investigated. The application of LDPC codes to other communication systems like MIMO and cooperative networks has been studied and also simulated to analyze the performance and get insight of those systems. Chapter 6 is the last one and gives the conclusion and future direction of research.

Chapter 2 Low Density Parity Check Codes

2.1 Information Theory and Digital Communication

In the modern age of science and technology, the demand for efficient and reliable digital data transmission and storage is increasing day by day. The emergence of high speed networks, large scale data processing and information storage in different sectors of life has accelerated the demand of this field. All the today work and research is focused on to control the errors in the data when it is reproduced.

Information theory and digital communication together provide the solutions for many of these problems. Information theory gives solution to maximize the efficiency of measurement, transmission and storage of data in terms of capacity, data reliability and throughput. Digital communication manipulates the data from source to destination over the medium in a desired format.

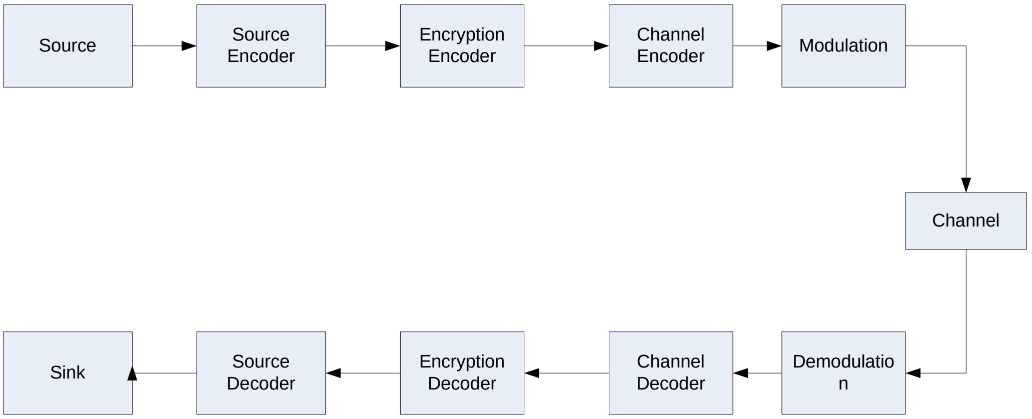


Figure 2.1 Typical Example of Digital Communication System

The typical flow of information from source to destination in a digital communication system[47-50] has been shown in the figure 2.1

The design of any digital communication system begins with a description of the channel (received power , bandwidth limitations, noise statistics, and other impairments such as fading etc) and systems requirement parameters such as data rate and error performance. Digital circuits are less subject to diction and provide better immunity to noise. With digital techniques, extremely low error rates producing high signal fidelity are possible through error detection and correction.

2.1.1 Shannon-Hartley Capacity Theorem

Shannon[2] gave first time the maximum information transfer over a noisy channel. The capacity relationship known as Shannon-Hartley theorem [48] can be given as

$$C = W \log_2 (1 + \frac{S}{N}) \qquad 2.1.1$$

where

C= Channel capacity in bits /sec

W=bandwidth of the channel in Hertz

S=Signal power

N= Noise power

The Shannon-Hartley theorem tells the maximum amount of error-free digital data that can be transmitted over a communications channel (e.g., a copper wire or an optical fiber) with a specified bandwidth in the presence of noise. Shannon's work showed that the values of S, N and W set a limit on transmission rate, not on error probability. Shannon[51] used equation 2.1 to graphically exhibit a bound for the achievable performance of the practical systems.

For $N=N_0W$ and $R=C$, we get the following relationship

$$\frac{S}{N_0 C} = \frac{E_b}{N_0} \qquad 2.1.2$$

By re-arranging the equations we finally get

$$\frac{E_b}{N_0} = \frac{W}{C}(2^{C/W} - 1) \qquad 2.1.3$$

This equation provides the basis for Shannon limits.

2.1.2 Shannon Limit

There exists a limiting value of E_b/N_0 below which there can be not error free communication at any information rate .Using the following identity[48]

$$\lim_{x \to \infty} (1 + x)^{\frac{1}{x}} = e \qquad 2.1.4$$

We can calculate the limiting value of E_b/N_0 as follows

Let $$x = \frac{E_b}{N_0}\frac{C}{W} \qquad 2.1.5$$

$$\frac{C}{W} = x\log_2(1+x)^{\frac{1}{x}} \qquad 2.1.6$$

$$1 = \frac{E_b}{N_0}(1+x)^{\frac{1}{x}} \qquad 2.1.7$$

In the limit as $C/W \to 0$, we get

$$\frac{E_b}{N_0} = \frac{1}{\log_2 e} = 0.693 \qquad 2.1.8$$

Or in decibels

$$\frac{E_b}{N_0} = -1.6\ dB \qquad 2.1.9$$

The value E_b/N_0 is called the Shannon limit. Optimum system designs can be best described as search for trade-offs among various constraints and conflicting goals. The modulation and coding trade-off for best use of transmitter power and channel bandwidth is important.

2.1.3 Entropy

In information theory, entropy is a measure of the uncertainty associated with a random variable. In this context, the term usually refers to the Shannon entropy introduced by Shannon in his landmark paper[2]. Entropy is defined as the average amount of information per source output and is expressed by

$$H = -\sum_{i=1}^{n} p_i \log p_i \qquad 2.1.10$$

where H is in bits/source output and p_i is the probability of *i-th* output and $\sum p_i = 1$.

For the message having two possible outputs (1 or 0) with probabilities p and $q=(p-1)$, the entropy is defined as

$$H = -(p\log_2 + q\log_2 q) \qquad 2.1.11$$

where H can be interpreted in many ways.

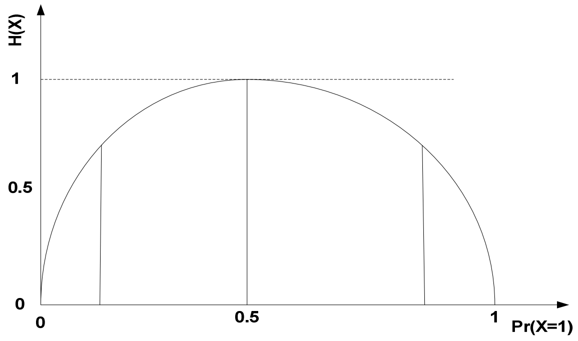


Figure 2.2 Entropy versus Probability

2.1.4 Sampling Theorem

The basic of a digital communication is the Sampling theorem which provides a sufficient condition, but not a necessary one, for perfect reconstruction. This is implemented mostly by a sample and hold operation. In this process, a switch and storage mechanism form a sequence of samples of the continuous input waveform. The output is the pulse amplitude modulation (PAM) which can be retrieved by a simple low pass filter to the original input waveform. For any band-limited signal, the following relation must hold for uniform sampled intervals.

$$T_s \leq \frac{1}{2} f_s \sec \qquad 2.1.12$$

This is known as uniform sampling theorem. The restrictions stated in terms of sampling rate is called Nyquist criterion.

$$f_s \geq 2 f_m \qquad 2.1.13$$

The sampling rate $f_s = 2f_m$ is called the Nyquist rate .This is the theoretical sufficient condition to reconstruct the signal perfectly. Usually to avoid the aliasing effect, the sampling rate is kept a bit higher than the Nyquist rat.

2.1.5 Digital Modulation

Digital modulation is the solution for today need of information capacity, compatibility with digital data services, higher data security, better quality communications(error control coding, equalizer, encryption etc), and quicker system availability. Developers of communications systems face the following constraints:

- bandwidth limited systems
- power limited systems
- inherent noise level of the system

The modulator maps each encoder output to a waveform suitable for transmission over the channel. Digital modulation methods are generally categorized into phase-shift keying (PSK), frequency-shift-keying (FSK), amplitude-shift-keying (ASK) and quadrature-amplitude modulation (QAM). In the former three methods, finite numbers of phases, frequencies and amplitudes are used respectively; while in the latter, a finite number of at least two phases and two amplitudes are used for modulation. Binary-phase-shift-keying (BPSK) is a form of phase modulation whereby the modulator generate the following waveforms s_1(t) and s_2(t) for an encoder output “1” and “0” respectfully.

$$s_1(t) = \sqrt{\frac{2E_s}{T}} \cos(2\pi f_c t), 0 \le t \le T \qquad 2.1.14$$

$$s_2(t) = \sqrt{\frac{2E_s}{T}} \cos(2\pi f_c t + \pi), 0 \le t \le T \qquad 2.1.15$$

E_s is the energy per symbol and f_c is the carrier frequency, T is the time period.

On the other side of the channel, the demodulator maps each o seconds of the waveform to an output. Optimally, the demodulator includes a matched filter or a correlation detector and a sampler. The demodulator output may be soft values (i.e. unquantized real numbers) or may be quantized to finite set of discrete symbols. The sampled output (*y*) of a BPSK demodulator based on coherent detection is a real number. The noisy received signal is denoted as *r(t).*

$$y = \int_0^T r(t) \sqrt{\frac{2E_s}{T}} \cos(2\pi f_c t) dt \qquad 2.1.16$$

For *M-Ary* phase shift keying (MPSK) , the channel signals (*M=2*^{k} , *M* channel signals (S)) are given by the equation

$$S = \begin{cases} s_i = \sqrt{\frac{2E_s}{T} \cos(2\pi f_c t + \theta_i)}, 0 \le t \le T \\ \theta_i = \frac{2\pi(i-1)}{M}, \qquad 0 \le t \le M \end{cases} \qquad 2.1.17$$

For $M = 2^k, k = 2,3,.....$, then MPSK is termed as Quadrature-PSK (4PSK), 8PSK and so on.

2.2 Channel Characterization and Modeling

The medium between the transmitting antenna and the receiving antenna is referred as channel, shown in Figure 2.3

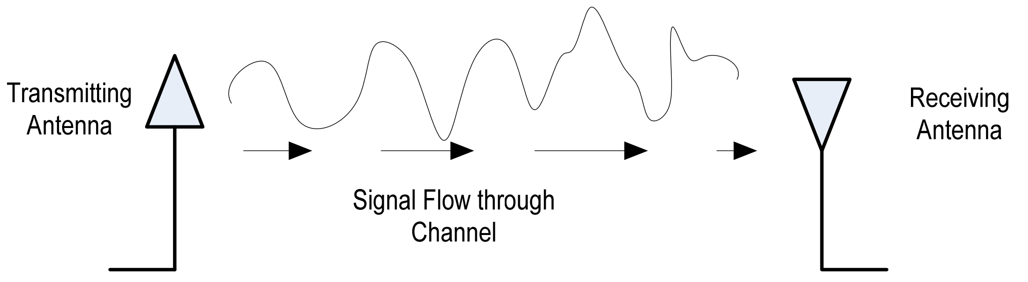


Figure 2.3 Channels between Transmitting and Receiving Antenna

The characteristics of wireless signal[47, 50] changes as it travels from the transmitter antenna to the receiver antenna. These characteristics depend upon the distance between the two antennas, the path(s) taken by the signal, and the environment (buildings, medium characteristics etc) around the path. The profile of received signal can be obtained from that of the transmitted signal if we have a model of the medium between the two. This model of the medium is called channel model.

2.2.1 Additive White Gaussian Noise Channel

Additive white Gaussian noise (AWGN) is used to model line-of-sight (LOS) communication links e.g. satellite or deep space communications. It is also a good model to simulate the background noise in terrestrial multipath links e.g. mobile communications. In the AWGN model, the effects that the channel may have over the transmitted signal *s(t)* is narrowed down to only additive noise

$$r(t) = s(t) + n(t) \qquad 2.2.1$$

Where *n(t)* is a random process with uniformly spread power spectral density (white noise) over a bandwidth much wider than the signal bandwidth and Gaussian distribution of amplitude. The Gaussian probability density function(pdf) with zero mean is given as

$$p(n) = (\frac{1}{\sigma\sqrt{2\pi}}) \exp[-\frac{1}{2}(\frac{n}{\sigma})^2] \qquad 2.2.2$$

where σ^2 is the variance of *n*. The normalized Gaussian density function of a zero-mean process is obtained by assuming that $\sigma = 1$ as shown in the figure 2.4.

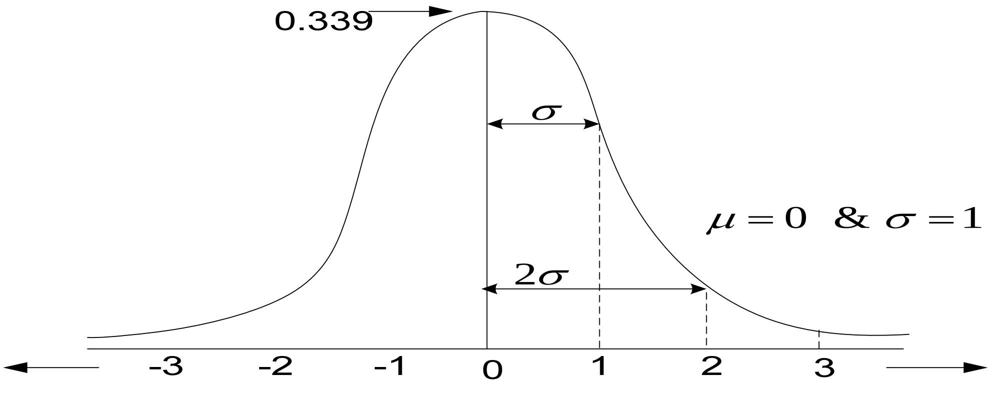


Figure 2.4 Normalized Gaussian pdf

Now probability density function for the received signal *r* is stated as

$$p(r) = (\frac{1}{\sigma\sqrt{2\pi}}) \exp[-\frac{1}{2}\left(\frac{r-s}{\sigma}\right)^2] \qquad 2.2.3$$

The Gaussian distribution is often used as the system noise model because of the Central limit theorem[52] which states that under very general condition the probability distribution of the sum of *k* statistically independent random variables approaches the Gaussian distribution as $k \to \infty$.

In BPSK modulation the set of constellation points contains

$$S = \left\{+\sqrt{E_b}, -\sqrt{E_b}\right\} \qquad 2.2.4$$

where $\pm E_b$ is the energy contained by the signal.

The likelihood function for this is given by

$$\Pr(r \mid s = \pm\sqrt{E_b}) = \left(\frac{1}{\sqrt{2\pi}}\right) \exp(-\frac{1}{2\sigma^2}(r \pm \sqrt{E_b})^2) \qquad 2.2.5$$

where σ^2 is the variance of AWGN[5, 52]. The log likelihood ratio (LLR) for BPSK modulation in AWGN is

$$LLR(r) = \frac{2\sqrt{E_b}}{\sigma^2} r \qquad 2.2.6$$

where the term $\frac{2\sqrt{E_b}}{\sigma^2}$ is called the channel probability factor.

2.2.2 Fading Channel

In a wireless mobile communication system, a signal can travel from transmitter to receiver over multiple reflective paths; this phenomenon is referred to as multipath propagation. The effect can cause fluctuations in the received signal's amplitude, phase, and angle of arrival, giving rise to the terminology multipath fading. Moreover, the relative speed between the transmitter and receiver and the surrounding objects may induce different Doppler shifts on each multipath component. A channel model which consider such effects is called fading channel[53-55].The fading channels are broadly categorized as

a) Small Scale Fading (due to small changes in position)

b) Large Scale Fading (due to motion over large areas)

Small-scale fading is also called Rayleigh fading because if the multiple reflective paths are large in number and there is no line-of-sight signal component, the envelope of the received signal is statistically described by a Rayleigh probability density function. When there is a dominant non-fading signal component present, such as a line-of-sight propagation path, the small scale fading envelope is described by a Rician probability density function. A mobile radio roaming over a large area must process signals that experience both types of fading: small-scale fading superimposed on large-scale fading. As the amplitude of the specular component approaches zero, the Rician pdf approaches a Rayleigh pdf, expressed as

$$p(r) = \begin{cases} \frac{r}{\sigma^2}\exp[-(\frac{r^2}{2\sigma^2}] & \text{for } r \geq 0 \\ 0 & \text{otherwise} \end{cases} \qquad 2.2.7$$

where r is the envelope amplitude of the received signal, and $2\sigma^2$ is the pre-detection mean power of the multipath signal.

2.3 Channel Coding

Channel coding refers to the class of signal transformations designed to improve communications performance by increasing the robustness of the transmitted signal against channel impairments (noise, interference, fading, etc). Channel coding began in 1948 with the publication of a famous paper[2] by C.E.Shannon. He showed that it is possible to design a communication system with any desired small probability of error whenever the rate of transmission is smaller than capacity of the channel.

Channel coding can be divided into two classes; one is the waveform coding and the other is the structured redundancy coding[48] as best known as error control coding. Waveform coding deals with transforming waveforms into better waveforms to make the detection process less subject to errors. Structured sequences deals with transforming data sequences into better sequences having structure redundancy for making the decision process less subject to errors.

2.3.1 Waveform Coding

Waveform coding transform a waveform (representing a message) into an improved waveform set. The improved waveform set then can be used to provide improved bit error probability (P_b) compared to the original set. The most popular of such waveform codes are referred to as orthogonal and bi-orthogonal codes [48]. The common orthogonal signals used in communication systems are *sin(x)* and *cos(x)*. In general, a set of equal signals $s_i(t)$, where $i = 1, 2, ...M$, constitute an orthonormal(orthogonal, normalized to unity) set as

$$z_{ij} = \frac{1}{E}\int_0^T s_i(t)s_j(t)dt = \begin{cases} 1 & \text{for } i = j \\ 0 & \text{otherwise} \end{cases} \qquad 2.3.1$$

where z_{ij} is called the cross correlation coefficient , and where E is signal energy expressed as

$$E = \int_0^T s_i^2(t)dt \qquad 2.3.2$$

In case of a binary orthogonal signal set, the pulse waveform is mathematically described as

$$s_1(t) = p(t) \qquad 0 \le t \le T \qquad 2.3.3$$

$$s_2(t) = p(t - \frac{T}{2}) \quad 0 \le t \le T \qquad 2.3.4$$

The cross correlation between two signals is a measure of the distance between the signal vectors. The smaller the correlation, the more distant are the vectors from each other[48]. This can be verified for antipodal signals ($z_{ij} = -1, s_1(t) = \sin\omega_0 t \;\&\; s_2(t) = -\sin\omega_0 t$) are represented by a vectors that are most distant from each other. For orthogonal signals ($z_{ij}=0$) are represented by vectors that are closer to one another that antipodal vectors.it should be obvious that the distance between two identical waveforms($z_{ij}=1$) is zero.

Since the orthogonality condition in equation 2.23, is represented in terms of waveforms $s_i(\mathrm{t})$ & $s_j(t)$ where *i, j* represent the number of the waveform set. Each waveform consist of the sequence of pulses where each pulse is designated with a level +1 or -1 which in turn represents the binary digits 1 or 0 respectively . Now the equation 2.23 can be show in the form as

$$z_{ij}=\frac{\text{number of digits agreements - number of digits disagreements}}{\text{total number of digits in the sequence}}$$

$$=\begin{cases}1 & \text{for i=j}\\ 0 & \text{otherwise}\end{cases} \tag{2.3.5}$$

2.3.1.1 Orthogonal Codes

Orthogonal coding for a set of data can be represented in a matrix form. A one bit data set can be transformed, using orthogonal code words of two digits each, demonstrated as follows in matrix form:

For data set (0, 1); the orthogonal data codeword set is

$$H_1=\begin{bmatrix}0 & 0\\ 0 & 1\end{bmatrix} \tag{2.3.6}$$

To encode a 2 –bit data set, equation 2.24 is extended both horizontally and vertically, creating a matrix H_2 in the following form:

$$\begin{array}{cc}\underline{\text{data set}} & \underline{\text{Orthogonal codeword set}}\\ \begin{array}{cc}0 & 0\\ 0 & 1\\ \multicolumn{2}{c}{\cdots\cdots}\\ 1 & 0\\ 1 & 1\end{array} & \left[\begin{array}{cc:cc}0 & 0 & 0 & 0\\ 0 & 1 & 0 & 1\\ \multicolumn{4}{c}{\cdots\cdots\cdots\cdots}\\ 0 & 0 & 1 & 1\\ 0 & 1 & 1 & 0\end{array}\right]\end{array} \tag{2.3.6}$$

$$=\begin{bmatrix}\mathrm{H}_1 & \mathrm{H}_1\\ \mathrm{H}_1 & \overline{\mathrm{H}}_1\end{bmatrix}$$

In a general form, for an n-bit data set, we can construct a code word set H_n, of dimension $2^n\times2^n$, known as Hadamard matrix is represented as

$$H_n = \begin{bmatrix} \mathrm{H}_{n-1} & \mathrm{H}_{n-1} \\ \mathrm{H}_{n-1} & \overline{\mathrm{H}}_{n-1} \end{bmatrix} \qquad 2.3.6a$$

2.3.1.2 Bi-orthogonal Codes

Binary orthogonal signal set of n -bits can be obtained from the orthogonal signal set of $n/2$. Orthogonal codes are binary valued and they have equal number of 1's and 0's. There are $n/2$ positions where 1's and 0's differ. Consequently, the distance between two orthogonal codes is also $n/2$ .Since the distance properties are fundamental in error coding, it can be shown that orthogonal codes can be used to detect and correct multiple errors[56].In general, the orthogonal code set is represented as

$$B_n = \begin{bmatrix} H_{n-1} \\ \overline{H}_{n-1} \end{bmatrix} \qquad 2.3.7$$

Where B_n is the bi-orthogonal signals set and consists of a combination of orthogonal(H_{n-1}) and antipodal($\overline{H}_{n-1}$) signals. In terms of auto correlation (z_{ij}), bi-orthogonal codes can be characterized as

$$z_{ij} = \begin{cases} 1 & \text{for i=j} \\ \text{-1} & \text{for i} \neq \text{j, } |i-j| = \dfrac{N}{2} \\ 0 & \text{for i} \neq \text{j, } |i-j| = \dfrac{N}{2} \end{cases} \qquad 2.3.8$$

2.3.2 Error Control Coding

Error control coding uses encoder and decoder for reliable digital communication as close as possible to the Shannon limit of any noisy channel. Error control coding includes the detection of an error or may involve the re-construction of the original data from the data corrupted by noise during storage or transmission.

2.3.2.1 Error Detection Codes

This involves simply detection of error in a data transmission and is generally low complex and easily implemented using simple scheme like repetition codes, parity codes, check sums or cyclic redundancy checks. In repletion codes the transmitter sends the data several times across the channel and the decoder can detect the correct sequence. In the other types of codes like Parity bit codes, a single bit is added to indicate sum of the information bits being even or odd. Parity bit codes were used in the early storage systems. Checksum codes are the same as parity bit codes in which a hash function maps a frame of data to fix-size checksum value which is transmitted with data. Cyclic redundancy codes (CRC) are popular due to implementation simplicity. CRC codes are position dependent checksum in which position of the words as well as value is considered to determine the checksum in a sequence.

2.3.2.2 Error Correction Coding

The error correction coding is categorized as

a) Automatic Repeat Request (ARQ)

b) Forward Error Correction (FEC)

c) Hybrid Automatic repeat request (HARQ)

In error detection coding, the communication system generally needs to provide a means of alerting the transmitter that an error has been detected and a re-transmission is required. Such procedure is called Automatic Repeat Request or Automatic re-transmission query (ARQ). The major advantage of ARQ is that error detection requires much simpler decoding and much less redundancy. ARQ is adaptive [48] in the sense that information is re-transmitted only when an error occurs. Following are the main disadvantages:

i) A reverse channel is not available or delay with ARQ would be excessive

ii) The retransmission strategy may not easily implemented

iii) Excessive number of re-transmission may be required as the errors occurrence not ensured.

On the other hand, in the forward error correction (FEC) scheme, the receiver decodes the codewords and first locates the error and then try to correct it. If the decoder fails to recover the codewords successfully, the erroneous data will be delivered to sink. The decoding process in the FEC is more complex than ARQ system but it offer high throughput and does not require a feedback channel to the receiver.

Hybrid ARQ (HARQ) system combines the traditional ARQ scheme with FEC which corrects some error at the receiver side and sends request for re-transmission as well. Hybrid ARQ are more complex than FEC and ARQ although it provides more reliable transmission together with high throughout. HARQ is used in several wireless applications.

Two main categories of FEC codes are block codes and convolution codes in terms of the code structure. In block codes, the sequence of data is divided into frames of information bits while in convolutional code, the data is a stream of arbitrary length. The most popular types of FEC block codes are Hamming codes, Reed-Solomon coding, BCH, Golay and LDPC. LDPC codes are now used in many recent high-speed communications standard like DVB-S2[57] , WiMax and Wi-Fi[58].

2.4 Graphs

A graph, in general, is an interconnection of objects (some sorts of data having mutual relation) connected by lines or curves. Mathematically, a graph is defined as an ordered pair of sets $G = (V, E)$ where V is the set of nodes (vertices) and E is the set of connections lines (edges). If the is a fix number of vertices and edges , then the graph is called "Finite Graph".

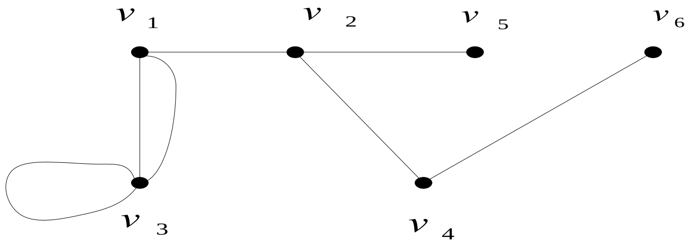


Figure 2.5 Typical Graph Representations

For example, consider a finite graph with vertices $V = \{v_1, v_2, v_3, v_4, v_5\}$ and edges $E = \{(v_1, v_2), (v_1, v_3), (v_3, v_3), (v_3, v_1), (v_2, v_5), (v_2, v_4), (v_5, v_6)\}$ as shown in the figure 2.5.

The edge (v_3, v_3) is called a self loop, and (v_3, v_1) is called the parallel loop. Each connection is called a path. In the figure 2.6 , the longest path connects the vertices v_1, v_2, v_5, v_6 and forms a path of length 4. The path which ends at the same vertex, is called the cycle and all of its vertices form the

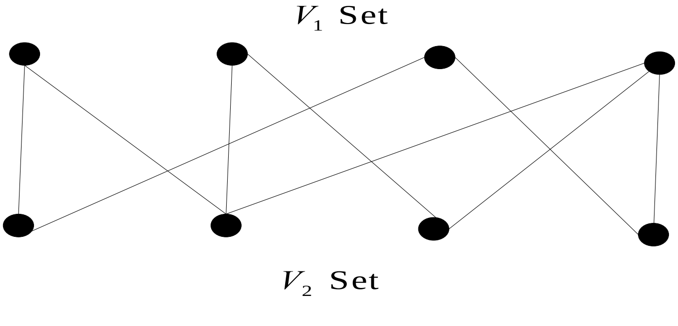


Figure 2.6 Bipartite Graph

cycle length or length of the path of the cycle. The shortest cycle is termed as " Girth" which plays important role in codes on graphs[59].

If the vertices is split into two groups in ordered form and have no closed loop edge (self loop) and vertex of one group connect to the vertex of another group only , then such a graph is termed as " Bipartite graph". For example, $V_1 = \{v_{11}, v_{12}, v_{13}, v_{14}, v_{15}\}$ and $V_2 = \{v_{21}, v_{22}, v_{23}\}$ are the two sets such that $\{V_1, V_2\} \in V$, where figure 2.6 shows some typical connection for the two sets.

2.5 Linear Block Codes

Linear block codes are characterized as (n,k) such that $n > k$ where k shows the message bits and n is the code length. The n codeword bits depend on the source bits k . The k -bit messages form 2^k distinct message sequences, referred to as k-tuples (sequence of k digits). The n -bit block form as many distinct sequences, referred to as n -tuples. A block code is called linear if the modulo-2 sum of each of the two codewords is also a codeword.

2.5.1 Generator Matrix

In general, a generator matrix [48] for $k \times n$ array is defined as

$$G = \begin{bmatrix} V_1 \\ V_2 \\ \vdots \\ V_k \end{bmatrix} = \begin{bmatrix} V_{11} & V_{12} & \cdots V_{1n} \\ V_{21} & V_{22} & \cdots V_{2n} \\ \vdots & \vdots & \vdots \\ V_{k1} & V_{k2} & \cdots V_{kn} \end{bmatrix} \qquad 2.5.1$$

Code vectors, by convention, are usually designated as row vectors. For a message s , a sequence of k message bits is shown below as row vector ($1 \times k$ matrix having one row and k columns):

$$s = s_1, s_2, ...s_k \qquad 2.5.2$$

The codeword C is generated as the matrix product s and G and is written as

$$C = sG \qquad 2.5.3$$

A systematic linear block will have the generator matrix in the following form:

$$G = [P \vdots I_k] \qquad 2.5.4$$

where I_k is a $k \times k$ identity matrix and P is a $k \times (n-k)$ matrix. The codeword generated by the systematic generator matrix can be simply divided into the two parts of *k* message bits and $(n-k)$ parity check bits.

2.5.2 Parity Check Matrix

The parity check matrix is used to decode the receive sequence. For each $(n \times k)$ generator matrix *G*, there exists an $(n \times k) \times n$ matrix H such that each row of *G* is orthogonal to the rows of *H*; i.e. $GH^T = 0$ where H^T is transpose of *H* and 0 is a $k \times (n-k)$ all zero matrix. *H* is called the parity check matrix and for systematic linear block code is written in form:

$$H = \left[I_{n-k} \ \vdots \ P^T \right] \quad 2.5.5$$

H is a parity check matrix because it can be used to test if the codeword is valid or not codeword is valid only if $H.code^T = 0$.

2.6 Introduction to LDPC Codes

Low density Parity check (LDPC) codes, also known Gallager codes are a type of linear block codes , first proposed by Gallager [5] and were scarcely considered in the three decades that followed due to its computational complexity and limited computational ability of the receiver at that time. LDPC codes were reinvented by Mackay and Neal[6, 7] and have taken considerable attention recently due to their Shannon limits performance[6, 8, 10, 11] with belief propagation decoding algorithm. Before Mackay and Neal , a notable work was done by Tanner[60] in which Tanner generalized LDPC codes and introduce a graphical representation of LDPC codes and now called Tanner graph.

2.6.1 Parity Check Matrix and Tanner Graph for LDPC Codes

A low density parity check codes are defined by parity check matrix that is sparse[5, 7].The "low density" terminology comes from the characteristics of their parity check matrix which contains fraction of ones (1s) as compared to the number of zeros(0s). LDPC code can be denoted in general as (N, d_v, d_c) where N is the length of the code equal to the number of the column in a parity check matrix, d_v is the number of ones(1s) in a column of a parity check matrix and d_c is the number ones(1s) is a row and a parity check matrix. LDPC codes can be regular or irregular. If the number of ones (1s) in each row and column of a parity check matrix are the same, then it is called regular and if the number of ones (1s) in row or column are not the same then it is called irregular. The restriction that $d_v < d_c$ is needed to ensure more than just all-zero codeword satisfies all of the constraints or equivalently, to ensure that a nonzero code rate. For regular parity check matrix or regular LDPC code, the number of ones in H satisfies the following condition:

$$M.d_c = N.d_v \qquad 2.6.1$$

where M and N are the number of row and columns of a parity check matrix respectively. The code rate is given as

$$r = 1 - \frac{M}{N} = \frac{N-M}{N} \qquad 2.6.2$$

Or equivalently

$$r = 1 - \frac{d_v}{d_c} = \frac{d_c - d_v}{d_c} \qquad 2.6.3$$

Here M rows are assumed to be linearly independent and $d_v < d_c$.Also for best performance, the number of ones (1s) in a column is set as $d_v \geq 3$.The code is valid only if

$$H.code^T = 0 \qquad 2.6.4$$

where H is the sparse parity check matrix and code is the codeword obtained from the generator matrix(G) and message bits (s).

Consider a parity check matrix H, such that $d_v = 2$ and $d_c = 4$

$$\begin{array}{cc} & \begin{array}{cccccccc} v_1 & v_2 & v_3 & v_4 & v_5 & v_6 & v_7 & v_8 \end{array} \\ H = & \begin{bmatrix} 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \end{bmatrix} \begin{array}{c} c_1 \\ c_2 \\ c_3 \\ c_4 \end{array} \end{array} \qquad 2.6.5$$

The sparse parity check matrix is best represented by a Tanner graph[60, 61].Each row of the parity check matrix represents the variable node and each column of the parity check matrix represents the check node. The one in each row or column shows the connectivity between variable and check nodes. The set of bit nodes connecting to check nodes m is denoted by $N(m) = \{n \mid h_{mn} = 1\}$ and the set of check nodes connecting to bit node n is given by $M(n) = \{m \mid h_{mn} = 1\}$. A typical Tanner graph is shown in the figure 2.7. This graph is for (8, 2, 4) regular LDPC code.

The relationship of check node and variable node in the figure 2.7 is expressed in algebraic form as

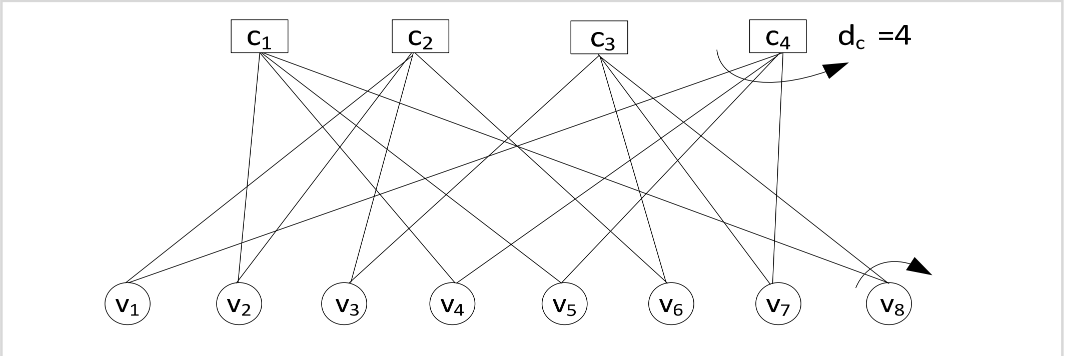


Figure 2.7 Tanner Graph Representation of *H*

$$\begin{aligned} c_1 &: 0 = v_2 + v_4 + v_5 + v_8 \\ c_2 &: 0 = v_1 + v_2 + v_3 + v_6 \\ c_3 &: 0 = v_3 + v_6 + v_7 + v_8 \\ c_4 &: 0 = v_1 + v_4 + v_5 + v_7 \end{aligned} \qquad 2.6.6$$

The work of the Luby et al[9, 62] have demonstrated that irregular parity check matrices generally outperform their regular counterparts but it’s quite difficult to construct an irregular parity check matrix in case when high girth is required. Also irregular codes are difficult to implement and design hardware for it.

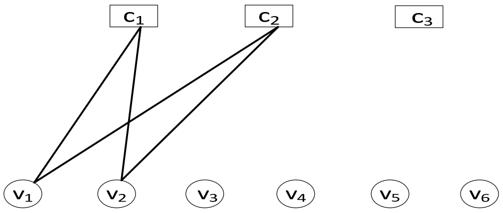


Figure 2.8 Short Cycle Example

Mackay et al[62] discovered that cycles especially short ones , tended to degrade decoding performance of LDPC codes. Therefore it is of high importance that short cycles be avoided in the construction of good LDPC codes. One example of short cycle (four cycles) is shown in the figure 2.8

2.6.2 Encoding of LDPC codes

The encoding of LDPC codes involves two basic tasks before we transmit the data.

a) Constructing the sparse parity check matrix

b) Generate codewords using that matrix.

The method for generating sparse parity check matrix has explained well by Mackay[63] and Neal[64] and has given a library of codes. The straightforward approach for encoding the LDPC code is stated as:

2.6.2.1 Method 1

This method is used for regular LDPC codes[25, 65]. For a given $m \times n$ sparse parity check matrix for the code x holds the following condition

$$H.x^T = 0 \qquad 2.6.$$

Partitioning the parity check matrix H into $m \times m$ matrix A and $m \times (n-m)$ matrix B, after rearranging columns if necessary to make A nonsingular, we can write the H as

$$H = [A \mid B] \qquad 2.6.8$$

Similarly we partitioned the codeword x into information bits s and parity bits p such that

$$x = [p \mid s] \qquad 2.6.9$$

Now equation (2.6.9) becomes

$$[A \mid B][p \mid s]^T = 0 \qquad 2.6.10$$

Thus Equation (2.6.10) becomes

$$Ap + Bs = 0 \qquad 2.6.11$$

Hence

$$p = A^{-1}Bs \qquad 2.6.12$$

It may be faster to compute c into two steps:

1) Compute $z = (Bs)$ in time proportional to $(n-m)$, exploiting the sparseness of B

2) Compute $p = (A^{-1}z)$, in time proportional to $(n-m)^2$

To exploit the sparseness of A with view to optimize step (2), we find the LU decomposition that satisfies

$$A = LU \qquad 2.6.13$$

Where L is sparse lower triangular matrix and U is sparse upper triangular matrix.

The previous process reduced the equation $AC = z$ to $Uc = y$. Now equation (2.6.11) becomes

$$Ly = z \qquad 2.6.14$$

and

$$Up = y \qquad 2.6.15$$

We can solve easily equations (2.6.14) and (2.6.15) by forward substitution and backward substitution respectively. The pivoting and bit reversing (PABR) algorithm[65] can be used to find the non-singular of binary matrix A over GF(2). The PABR first rearranges the column of the Gallager parity-check matrix for pivoting such that A is non-singular. And it then rearranges the row of the Gallager parity-check matrix for pivoting such that A is non-singular.

2.6.2.2 Method 2

The Gaussian elimination can transfer the initial sparse parity check matrix $H_{m\times n}$, which is a full rank, into an equivalent lower triangular form $\tilde{H}_{m\times n}$ as shown in equation (2.6.16)

$$\tilde{H}_{m\times n} = \begin{array}{c} \xleftrightarrow{\quad(n\text{-}m)\quad}\xleftrightarrow{\qquad m \qquad} \\ \begin{bmatrix} h_{11} & \cdots h_{1,(n-m)} & 1 & 0\cdots\cdots & \cdots & 0 \\ h_{21} & \ddots & & \ddots & \ddots & \vdots \\ \vdots & & \ddots & 1 & \ddots & \vdots \\ \vdots & & & \ddots & \ddots & 0 \\ h_{m1} & \cdots\cdots & \cdots & \cdots & h_{m,(n-1)} & 1 \end{bmatrix} \end{array} \Big\updownarrow m \qquad 2.6$$

This is the simple way to get the parity bits but this matrix is no longer sparse. This is a pre-processing before encoding the actual data. To encode the data, the codeword is written in form

$$x = (s, p) \qquad 2.6.17$$

Where $s = (s_1, s_2.........s_{n-m})$ the information is bits and $p = (p_1, p_2.........p_m)$ represents the parity bits which we get by back substitution when the column of the original matrix is shuffled for Gaussian elimination. The parity bits can be obtained in two steps:

i) Initialize

$$p_1 = \sum_{j=1}^{n-m} h_{1,j} s_j \qquad 2.6.18$$

ii) The next $p_2, p_3.........p_m$ parity bits are computed iteratively using the following equation.

$$p_l = \sum_{j=1}^{n-m} h_{l,j} s_j + \sum_{j=1}^{l-1} h_{l,j+(n-m)} p_j \qquad 2.6.19$$

Where $l = 2, 3.....m$. This scheme offers high encoding complexity but is used in very general.

2.6.2.3 Method 3

T.J.Richardson proposed a relatively low complexity efficient encoding[66] scheme. Instead of Gaussian elimination which resulted in a dense lower triangular matrix, the matrix $H_{m\times n}$ is brought to approximate lower triangular matrix $\tilde{H}_{m\times n}$ by row and column shifting only. The Richardson proposed approximate lower triangular matrix is composed of small sub-matrices as

$$\tilde{H}_{m\times n} = \left[\begin{array}{c|c|c} A & \mathrm{B} & \begin{matrix}1 & \ddots & 0 \\ & \mathrm{T} & 1\end{matrix} \\ \hline C & \mathrm{D} & \mathrm{E}\end{array}\right] \qquad 2.6.20$$

Where the dimension of the each matrix is as follow:

$A \rightarrow (m-g)\times(n-m)$

$B \rightarrow (m-g)\times g$

$T \rightarrow (m-g)\times(m-g)$; the lower triangular matrix

$\mathrm{C} \rightarrow \mathrm{g}\times(n-m)$

$D \rightarrow g\times g$

$E \rightarrow g\times(m-g)$

The matrix $\tilde{H}_{m\times n}$ is obtained by the column shifting of the original matrix $H_{m\times n}$.The columns are selected in such order as to give the reasonable depth of matrix E because the number of columns of A and C depends on $n-m$, so the only variable is the dimension g which is kept as small as possible to reduce the complexity and make the encoding efficient. The codeword $x=(s, p_1, p_2)$ is generated in the following way:

$$p_1^T = -U^{-1}(-ET^{-1}A + C)s^T \qquad 2.6.21$$

and

$$p_2^T = -T^{-1}(As^T + Bp_1^T) \qquad 2.6.22$$

where $U = -ET^{-}B + D$ and must be invertible. Once we get the triangular matrix T, then we can interchange the columns of D to make U invertible. Since the resulting matrix $\tilde{H}_{n\times m}$ is obtained by just column shifting of the parity check matrix $H_{n\times m}$ and each of the sub-matrices are sparse, so this method offer a linear complexity in order

2.6.2.4 Method 4

If the parity check matrix has one part regualr systematic and one part is the identity matix such as

$$H_{m\times n'} = \left[\bar{H}_{m\times n} \quad I_{m\times m} \right] \qquad 2.6a$$

The matrix $H_{m\times n'}$ is irregular systematic where $n' = n + m$. The parity equation for this type of the matrix is found simply as

$$p_i = \sum_{j=1}^{N-M} h_{i,j} s_j \qquad 2.6b$$

This need not to find the inverse and offer very low complexity.The codeword is now given by

$$X = [s \;\; p] \qquad 2.6c$$

2.6.3 LDPC Decoding Algorithms

LDPC is an iterative decoding algorithms based on belief propagation which is viewed as message passing algorithm. The message passing algorithm[5, 7, 8, 12, 26, 49] is a decoding algorithm in which messages are passed from node to node through the tanner graph used for complicated calculation using distributed hardware. The nodes act as independent processors, collecting incoming messages and producing outgoing messages. There is no global control over the timing or the content of the messages; bit and check nodes follow the common local rule: “Send a message as soon as all necessary incoming messages have been received”. When the graph is cycle-free, the message passing algorithm is recursive algorithm that always converge, after a finite number of messages have been passed ,to the true a posteriori probability (APP) log-likelihood ratios(LLR) is defined as

$$\lambda_n = \log \frac{\Pr[x_n = 1 \mid y]}{\Pr[x_n = 0 \mid y]} \qquad 2.6.23$$

Where $Y = X + n$ and y is the received sequence, x_n is the transmitted codeword sequence and n is the additive white Gaussian noise. λ_n is the APP log likelihood ratio (LLR).

2.6.3.1 Probability Domain Decoding Algorithm

A codeword $X = \{x_1, x_2 ....x_n\}$ is transmitted after BPSK modulation over a noisy AWGN channel , the received sequence is $Y = X + n$ where $Y = \{y_1, y_2, ...y_n\}$, n is the additive white

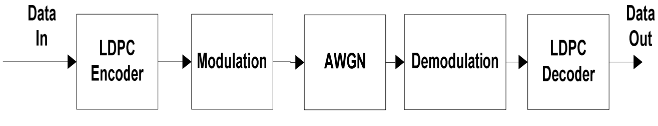


Figure 2.9 LDPC encoder and decoder in a communication system

Gaussian noise with zero mean and variance σ^2 {n~N(0, σ^2)} as shown in figure 2.9.

Now the LDPC decoding algorithm can be stated in the following steps for parity check $H_{m\times n}$ where m is the number of rows and n is the number of columns. The following notation is introduced to represent the decoding algorithms effectively.

$j = n$ =the number of columns in parity check matrix $H_{m\times n}$

$i = m$ = the number of rows in parity check matrix $H_{m\times n}$

$N(j) = \{i : h_{ji} = 1\}$: The set of column locations of the ones (1's) in the ith row of H .

$N(j)\setminus i$: The set of column locations of the ones (1's) in the ith row of H excluding location j .

$M(i) = \{j : h_{ji} = 1\}$: The set of row locations of the ones (1's) in the jth column of H .

$M(i)\setminus j$: The set of row locations of the ones (1's) in the jth column of H excluding location i.

Now the LDPC decoding algorithm can be demonstrated in the following steps.

1) Initialization : The a posteriori probabilities(APP) are initialized to channel output as

For s (i, j) , set

$$q_{ji}^0 = f_i^0 = \frac{1}{1+\exp(\frac{2y_i}{\sigma^2})} \qquad 2.6.24$$

$$q_{ji}^1 = f_i^1 = \frac{1}{1+\exp(\frac{-2y_i}{\sigma^2})} \qquad 2.6.25$$

$$f_i = \frac{f_i^0}{f_i^1} \qquad 2.6.26$$

Set the number of iteration as $I_{\max}$

2) Parity node update(Horizontal processing): Update r_{ji}

$$\delta r_{ji} = \prod_{i' \in N(j)\setminus i} (q_{ji'}^0 - q_{ji'}^1) \qquad 2.6.27$$

$$r_{ji} = \frac{1+\delta r_{ji}/(q_{ji}^0 - q_{ji}^1)}{1-\delta r_{ji}/(q_{ji}^0 + q_{ji}^1)} \qquad 2.6.28$$

3) Bit node update (Vertical processing): Updating the variable node as

$$\overline{R}_i = \prod_{j' \in M(i) \backslash j} r_{j'i} \qquad 2.6.29$$

$$s_{ji} = f_i \frac{\overline{R}_i}{r_{ji}} \qquad 2.6.30$$

Or equivalently $$s_{ji} = \frac{q_{ji}^0}{q_{ji}^1} \qquad 2.6.30a$$

Updating q_{ji}^0 and q_{ji}^0 as follow:

$$q_{ji}^0 = \frac{s_{ji}}{1+s_{ji}} \quad \text{and} \quad q_{ji}^1 = 1 - q_{ji}^0 \qquad 2.6.31$$

4) Calculating estimated codeword: In this step , the APP likelihood ratio R_i is computed and then the estimated codeword is calculated.

$$R_i = f_i \overline{R}_i \qquad 2.6.32$$

$$\hat{X} = \begin{cases} 0 & \text{for } R_i > 0 \\ 1 & \text{else} \end{cases} \qquad 2.6.33$$

R_i comparison depends on the modulation scheme used .Here it is for typical BPSK modulation.

5) Stop condition : If the parity check equation is satisfied

$$H.(\hat{x}_1, \hat{x}_2 \ldots\ldots\ldots\ldots \hat{x}_i)^T = 0 \qquad 2.6.34$$

Or the maximum iteration $(\mathrm{I}_{\max})$ is reached then terminates the decoding or otherwise go to step2.

Alternative way for Step 3 to Step 5: this can also be written in the form:
Step 3: In step 3, equation (2.6.29) and equation (2.6.32) take the following form:

First calculate

$$Q_i(0) = K_i(1-P_i) \prod_{j' \in M(i) \backslash j} r_{j'i}(0) \qquad 2.6.29a$$

$$Q_i(1) = K_i P_i \prod_{j' \in M(i) \backslash j} r_{j'i}(1) \qquad 2.6.29b$$

Now update

$$q_{ji}(0) = K_{ji}(1-P_i) \prod_{i' \in N(j) \backslash i} r_{i'j}(0) \qquad 2.6.31a$$

$$q_{ji}(1) = K_{ji}(P_i) \prod_{i' \in N(j) \backslash i} r_{i'j}(1) \qquad 2.6.31b$$

The constant K has the value such that $Q_i(0) + Q_i(1) = 1$ and $P_i = Pr\left(x_i = 1 | y_i\right)$ is the probability of $x_i = 1$ under that the knowledge of the received signal y is known, while the code structure is not considered. Set $q_{ji}(0) = 1 - P_i$ and $q_{ji}(1) = P_i$ for all i, j for which $h_{ji} = 1$

Step 4: For every column index i , compute

$$\hat{X} = \begin{cases} 1, \text{if } \ \mathrm{Q}_i(1) \geq 0.5(or > \mathrm{Q}_i(0)) \\ 0, \text{else} \end{cases} \qquad 2.6.33a$$

If $rem(\hat{X}.H^T, 2) = 0$, or if the maximum number of iterations is reached, then stop, else, continue iteration from Step 2.

2.6.3.2 Log Domain Decoding Algorithm

The probability domain sum product algorithm (SPA) suffers as many multiplications are involved although additions are less costly to implement. Due to many multiplications involved in BP SPA which can make it numerically unstable, therefore log domain SPA is preferred. We define the following notations:

$$L(f_i) = L(\frac{f_i^0}{f_i^1}) = \log\left(\frac{\Pr(x_i = 0 \mid y_i)}{\Pr(x_i = 1 \mid y_i)}\right) = \log\left(\frac{1/1 + \exp\left(2y_i / \sigma^2\right)}{1/1 + \exp\left(-2y_i / \sigma^2\right)}\right) = \frac{2y_i}{\sigma^2}$$

$$L(r_{ji}) = \log\left(\frac{r_{ji}(0)}{r_{ji}(1)}\right)$$

$$L(q_{ji}) = \log\left(\frac{q_{ji}(0)}{q_{ji}(1)}\right)$$

$$L(R_i) = \log\left(\frac{Q_i(0)}{Q_i(1)}\right)$$

1) Initialization : Set the maximum number of iterations $(I_{\max})$ and initialize as follows

$$L(q_{ji}) = L(f_i) = \frac{2y_i}{\sigma^2} \qquad 2.6.34$$

2) Parity node update(Horizontal process): For initial derivation purpose for log-domain SPA ,

As $r_{ji}(0) = 1 - r_{ji}(1)$ and $\tanh\left[\frac{1}{2}\log(\frac{p_0}{p_1}\right] = p_0 - p_1 = 1 - 2p_1$, we can write

$$\tanh\left(\frac{1}{2}L(r_{ji})\right) = \prod_{i' \in N(j) \backslash i} \tanh\left(\frac{1}{2}L(q_{ji'})\right) \qquad 2.6.35$$

Now factorizing $L(q_{ij})$ into sign and magnitude components as follow:

$L(q_{ij}) = \alpha_{ji}\beta_{ji}$, where $\alpha_{ji} = sign\left[L(q_{ji})\right]$ and $\beta_{ji} = \mid L(q_{ji}) \mid$

Re-writing equation (2.6.35) as

$$\tanh\left(\frac{1}{2}L(r_{ji})\right) = \prod_{i' \in N(j) \backslash i} \alpha_{ji'} . \prod_{i' \in N(j) \backslash i} \tanh\left(\frac{1}{2}\beta_{ji'}\right)$$

$$L(r_{ji}) = \prod_{i'} \alpha_{ji'} .2\tanh^{-1}\left(\prod_{i'} \tanh\left(\frac{1}{2}\beta_{i'j}\right)\right)$$

$$=\prod_{i'} \alpha_{ji'} .2\tanh^{-1}\log^{-1}\log\left(\prod_{i'} \tanh\left(\frac{1}{2}\beta_{i'j}\right)\right)$$

$$=\prod_{i'} \alpha_{ji'} .2\tanh^{-1}\log^{-1}\sum_{i'}\log\left(\tanh\left(\frac{1}{2}\beta_{i'j}\right)\right)$$

$$= \prod_{i'\in N(j)\backslash j} \alpha_{ji'} .\Phi\left(\sum_{i'\in N(j)\backslash j} \Phi\left(\beta_{ji'}\right)\right) \tag{2.6.36}$$

Where

$\Phi(x) = \Phi(x)^{-1} = -\log[\tanh(x/2) = \log\left[\frac{e^x+1}{e^x-1}\right]$ such that $x > 0$. This function can be implemented as look up table.

3) Bit node update (Vertical processing): Updating the variable node

Dividing equation (2.6.31a) by (2.6.31b) and then taking the logarithm of both sides, we get

$$L(q_{ji}) = L(f_i) + \sum_{j'\in M(i)\backslash j} L(r_{j'i}) \tag{2.6.37}$$

4) Updating the final Log-likelihood ratio(LLR):

Similarly we get from equation (2.6.29a) and (2.6.29b)

$$L(R_i) = L(f_i) + \sum_{j\in M(i)} L(r_{ji}) \tag{2.6.38}$$

5) Calculating estimated codeword and Stop condition :

For every column index i, we calculate

$$\hat{X} = \begin{cases} 0 & \text{for L}(R_i) > 0 \\ 1 & \text{else} \end{cases} \tag{2.6.38a}$$

If $rem(\hat{X}.H^T, 2) = rem(\hat{x}_1, \hat{x}_2, ...\hat{x}_n.H^T, 2) = 0$, or if the maximum number of iterations is reached, then stop, else, continue iteration from Step 2.

2.6.3.3 Approximation to Log-domain SPA algorithm

The approximation to Log-domain SPA knows as min-sum algorithm (MSA)[26], greatly reduces the complexity but with reduced performance. In the log-domain SPA, the complexity is at the step 2. Min sum decoding algorithm reduces the complexity by approximating the magnitude of the initial LLR ($L(q_{ji})$). From equation 2.6.36, we get the approximation as:

$$L(r_{ji}) = \prod_{i' \in N(j)\backslash j} \alpha_{ji'}.\Phi\left(\sum_{i' \in N(j)\backslash j} \Phi\left(\beta_{ji'}\right) \right) \cong \prod_{i' \in N(j)\backslash i} \alpha_{ji'}.Min_{i' \in N(j)\backslash i} \left|\left(\beta_{ji'}\right)\right| \qquad 2.6.39$$

After this modification, all the steps are repeated in the same way in as in Log-domain SPA in section (2.6.3.2).

2.6.3.4 Procedure of Min-Sum decoding algorithm

Before stating the min-sum decoding algorithm, we define some notations as

X = the transmitted actual sequence

Y = the received sequence

$L_i = L(f_i)$ = initial channel LLR

$V_{ji} = L(q_{ji})$ =variable node message

$C_{ji} = L(r_{ji})$ = check node message

$\hat{L}_i^k$ =The final LLR used for calculating the estimated codeword ($\hat{X}$)

The min-sum decoding algorithm is now stated in the steps below for parity check matrix

1) Initialization : Set $L_i = Y$ as the initial log likelihood ratio (LLR) as no priori information [12, 26] about the AWGN is required. and for each $(j,i) \in \{(m,n) \mid h_{mn} = 1\}$, we initialize the variable node as

$$V_{ji} = L_i \qquad 2.6.40$$

Set the maximum number of iterations ($I_{\max}$) as $k = 0$ to $I_{\max}$

2) Parity node update(Horizontal processing): Message passing from variable to parity(check) node for $j = 0$ to $M - 1(d_v)$ and C_{ji}^k is updated for each $i \in N(j)$ as follows

$$C_{ji}^k = \prod_{i' \in N(j)\backslash i} sign(V_{ji'}^{k-1}).Min_{\ i' \in N(j)\backslash i} |V_{ji'}^{k-1}| \qquad 2.6.41$$

3) Bit node updates (Vertical processing): Check node to variable node

For $i = 0$ to $N - 1(d_c)$, calculate the final LLR

$$\hat{L}_i^k = L_i + \sum_{j \in M(i)} C_{ji}^k \qquad 2.6.42$$

Updating the variable (bit) node message for each $j \in M(i)$

$$V_{ji}^k = \hat{L}_i^k - C_{ji}^k \text{ or } (V_{ji}^k = L_i + \sum_{j' \in M(i)\backslash j} C_{j'i}^k) \qquad 2.6.43$$

4) Hard decision : Estimating the codeword by hard decision as follows

$$\hat{X} = \begin{cases} 0 & \text{for } \hat{L}_i > 0 \\ 1 & \text{else} \end{cases} \qquad 2.6.44$$

5) Stop condition :If the parity check equation is satisfied i.e.

$$H.\hat{X}^T = 0 \qquad 2.6.45$$

Or the maximum iteration $I_{\max}$ is reached then terminates the decoding or otherwise go back to step 2.

The message passing between check node and variable node in step 2 and step 3 can also be represented in a graphical way as sown in figure 2.10 and figure 2.11

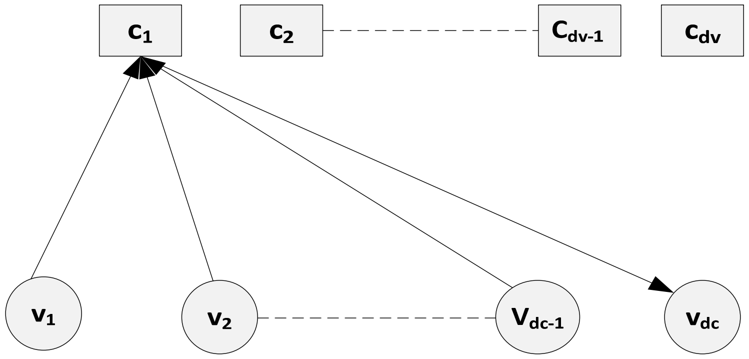


Figure 2.10 Horizontal processing: bit nodes to check nodes

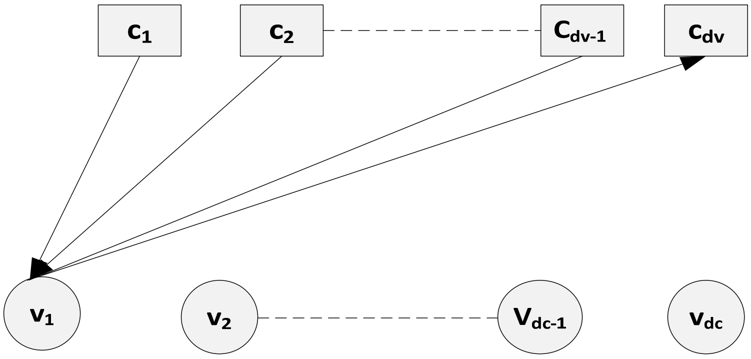


Figure 2.11 Vertical Processing: check nodes to bit nodes

2.6.3.5 Performance Improvement to Min-Sum Decoding Algorithm

MSA gives a significant decrease in decoding complexity and is also independent of the noise variance, so no a priori information of the AWGN channel is required. But MSA causes performance degradation reasonable because of check node messages overestimation in comparison to SPA.

For a certain edges v and c denote L_1 and L_2 as the values of $L(r_{ji})$ and C_{ji}^{k} computed by SPA and MSA respectively for the same messages during the previous iteration. Then the following statement holds about the relationship[12] of L_1 and L_2.

i) The values L_1 and L_2 have the same sign i.e. $sign(L_1) = sign(L_2)$

ii) The magnitude of L_2 is always greater than L_1 i.e. $L_2 > L_1$

The normalization factor is required to correct the magnitude overestimation and bring L_2 close to the value of L_1 which significantly improve the decoding performance. Several modification algorithms have been proposed in order to improve the performance of reduced complexity min-sum decoding algorithm(also known as universal most powerful (UMP) algorithms[26].

Two approaches[67, 68] are most popular for the check node message update in the equation (2.6.41) as under:

a) First Approach :Normalized Min-Sum Decoding Algorithm

$$C_{ji}^{k} = sf.\prod_{i'\in N(j)\backslash i} sign(V_{ji'}^{k-1}).Min_{\ i'\in N(j)\backslash i}|V_{ji'}^{k-1}| \qquad 2.6.41a$$

The scaling factor (sf) is used as normalization factor to correcting the variable message overestimation during the check node update in step 2. The value of scaling factor is $0 < sf \le 1$.

b) Second Approach :Offset Min-Sum Decoding Algorithm

$$C_{ji}^{k} = \prod_{i'\in N(j)\backslash i} sign(V_{ji'}^{k-1}).Max_{\ i'\in N(j)\backslash i}|V_{ji'}^{k-1} - f, 0| \qquad 2.6.41b$$

All the extrinsic messages with reliability values smaller than the offset factor f , are set to 0, such that they have no contribution to the preceding bit node processing.

In both the approaches, the normalization factor and the offset values should vary with each iteration number for achieving better performance but for making the processing complexity simple, it is kept constant. All other steps are repeated in the same way as in section 2.6.3.4.

2.7 Proposed Improved Method and Performance Analysis

Min-sum algorithm (MSA) minimizes the complexity of SPA but suffers from performance degradation and the bit error ratio is significantly higher than SPA. All improvements are made to make the MSA close to SPA in performance. The Offset and the Normalized MSA alter the inaccurate magnitude for the check node update calculated in step 2. The performance improvements by offset values and normalization factor are significant at low hardware complexity.

The modified min-sum algorithm[28] is selected for the new proposed technique[69] which is based on the following conditions for any two consecutive iterations.

a) When signs of the present and the previous variable messages are the same then the increase in the magnitude is comparatively small.

b) .If sign of the present message and the previous variable messages are different then the increase in the magnitude is large.

To slow down the sign change and to avoid overestimation for the variable message magnitude in all iterations, the signs of the present and the previous variable message during the vertical process are compared. If signs are different then first add the present and the previous message and normalize it by a smaller factor and when there is no sign change, the variable message is normalized with a relatively greater factor. The choice of the scaling factors is obviously dependent on the magnitude increase and its hardware implementation complexity. The range for both the scaling factors is $0 < s < 1$. In general, a scaling factor less than 0.5 and close to 0 is used for correction when there is sign change, and when no sign change a scaling factor greater than or equal to 0.5 is used .

In Equation (2.6.43) the variable message is stored after calculated at the kth iteration before using for updating as $V_{ji}^{k,tmp}$.

$$V_{ji}^{k,tmp} = R_i^k - C_{ji}^k \qquad 2.7.$$

Comparing the signs of the present message $V_{ji}^{k,tmp}$ and previous message V_{ji}^{k-1} .

If $\mathrm{sign}\left(V_{ji}^{k,tmp}\right) \neq sign\left(V_{ji}^{k-1}\right)$

Then update the message as:

$$V_{ji}^k = sf_2\,(V_{ji}^{k,tmp} + V_{ji}^{k-1}) \qquad 2.7.2$$

Else if $\mathrm{sign}\left(V_{ji}^{k,tmp}\right) == sign\left(V_{ji}^{k-1}\right)$

Then update the message as:

$$V_{ji}^k = sf_1\,(V_{ji}^{k,tmp}) \qquad 2.7.3$$

The temporary stored message and the previous message are added and multiplied with a scaling factor which greatly reduces the effect of overestimation in magnitude.

The scaling factors sf_1 and sf_2 are chosen such as it can be conveniently implemented in hardware and at the same time provide good approximation to the error performance. Now if the signs are different then the change in magnitude is large and is modified with small factor to reduce the overestimation effect. The scaling factors set for the simulation are sf_1=0.5 and sf_2=0.25.

Now Eq.7 and Eq.8 can be re-written as;

$$V_{ji}^{k} = 0.25(V_{ji}^{k,tmp} + V_{ji}^{k-1}) \qquad 2.7.2a$$

$$V_{ji}^{k} = 0.5(V_{ji}^{k,tmp}) \qquad 2.7.3a$$

Equation (2.7.2a) and equation (2.7.3a) gives good performance achievement while the cost for hardware is very low. This brings further improvement to the MSA in both lower and upper region of SNR by using two scaling factors. This algorithm can be further improved[70] by writing the equation (2.7.2a) as:

$$V_{ji}^{k} = 0.25(V_{ji}^{k,tmp}) \qquad 2.7.2b$$

In this the previous message is not required to add which simplify the hardware.

2.7.1 Hardware Implementation of the Proposed Scheme

The scaling factors chosen are easily implemented in hardware as the shift registers. The scaling factors chosen are the divided by 2 and 4 which are simply implemented in the hardware as the shift registers as shown in the figure 2.12. The first factor sf_2 is implemented in the shift register-1 and the second factor sf_1 is implemented in the shift register-2. Comparing to the algorithm in [28] , there is an additional shift register. The hardware complexity increased by one additional shift register, but it contributes greatly to the error performance. The sign comparator decides which input is to be selected for assigning to the current message V_{ji}^{k} through multiplexer (Mux) unit. The shift register is fast and easy to implement. So the hardware complexity does not increase reasonably while the performance

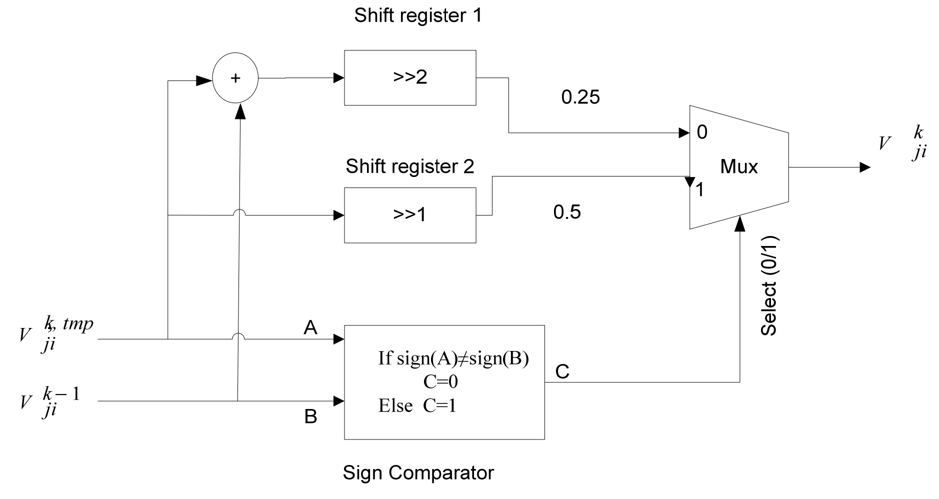


Figure 2.12 Implementation for variable message correction

achievement is better. Instead of uniform modification to all the variable messages, it gives the flexibility to update the messages in two ways which is significant in terms of error performance.

2.7.2 Analysis of Normalization Factor on Performance

In equation (2.6.41a), the check node message is updated as follow:

$$C_{ji}^{k} = sf.\prod_{\substack{i^{'}\in N(j)\\ i^{'}\neq i}} sign(V_{ji^{'}}^{k-1}).\min_{\substack{i^{'}\in N(j)\\ i^{'}\neq i}} |V_{ji^{'}}^{k-1}| \qquad 2.6.41b$$

The range for scaling factor (*sf*) value is *0< sf <1*. This is called the single factor or single way normalized min-sum decoding algorithm. It uses one scaling factor for updating the check node message during the row processing. The normalized min-sum algorithm has been simulated here to show the scaling factor effect on error performance when either code rate or length is changed. Three types of codes have been selected for the results validation and comparative analysis. Regular medium length codes (1024, 512) and (1296, 864) are chosen to show the effect for code rate change and a short length code (684,324) is chosen to show effect when the code length is changed. The simulation results in figures 2.13, 2.14 and 2.15 clearly show that error performance is affected significantly with scaling factor and it varies with code rate and length. Actually adaptive normalization can greatly

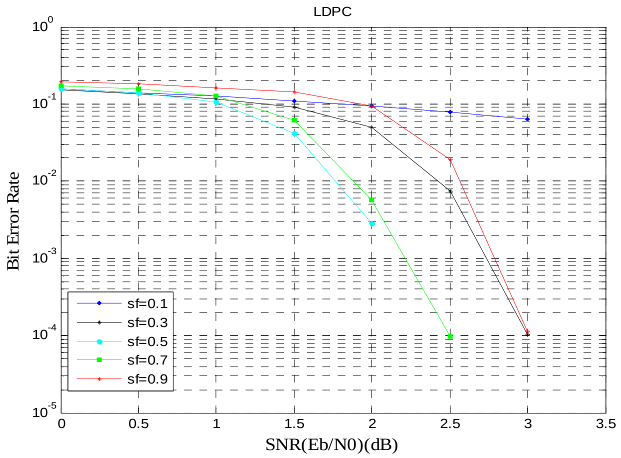


Figure 2.13 performance of one-way normalized min-sum for (1296, 864) LDPC code for various sf values ,iterations=10

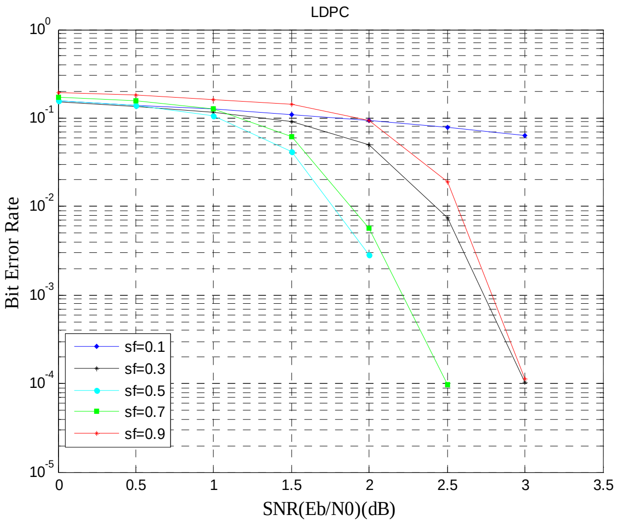


Figure 2.14 . BER performance of one-way normalized min-sum for (648, 324) LDPC code for various *sf* values, iteration=10

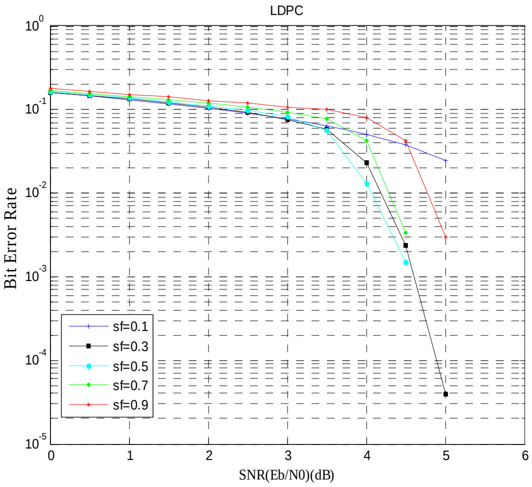


Figure 2.15 BER performance of one-way normalized min-sum for (1024, 512) LDPC

improve the performance but then it becomes very complex for real time applications and makes it practically impossible although in theory it can shows good results.

2.7.3 Two Way Normalized Min-Sum Simulation & Analysis

Equation (2.7.2) and equation (2.7.3) give the two-way normalized min-sum decoding algorithm. The two-way normalized min-sum algorithm (called hereafter New MSA), is validated by the codes (1024, 512), (1296, 864) and (684,324). The maximum allowable number of iterations is kept as 10. In the Figures 2.17, 2.18 and 2.19 we clearly observe that the improved min-sum decoding algorithm outperform than the MMSA, and even from SPA for medium and short length codes. The

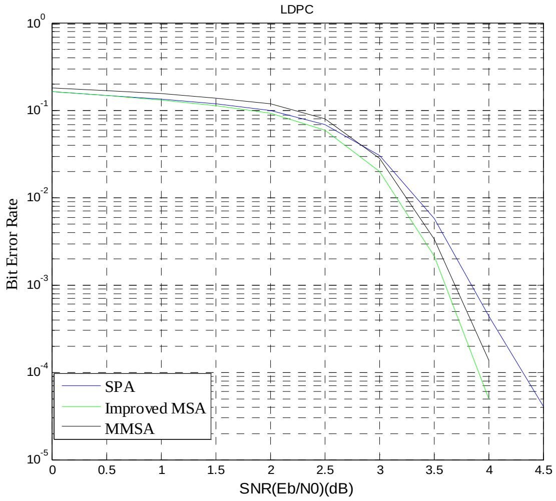


Figure 2.16 BER performance of two-way normalized min-sum for (324,648) LDPC code, (sf1=0.5; sf2=0.25) ,iterations=10

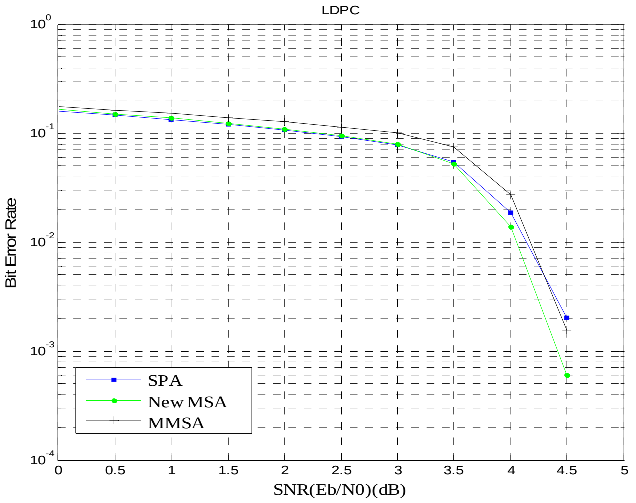


Figure 2.17 BER performance of two-way normalized min-sum for (1024, 512) LDPC code (sf1=0.5; sf2=0.25), iterations=10

outperformance of an improved min-sum decoding algorithm than the standard sum product algorithms is due to the fact that SPA depends on large sparse parity check matrix while the selected parity check matrix is of medium and short length. Simulations are performed for validating the performance of this two-way normalization technique and single-way normalization technique for medium and short length LDPC codes which best suits for most of the practical applications. The performance is tested at allowable maximum number of iteration as 10. The results obtained for two-way normalized min-sum (new MSA) are compared with MMSA and SPA.

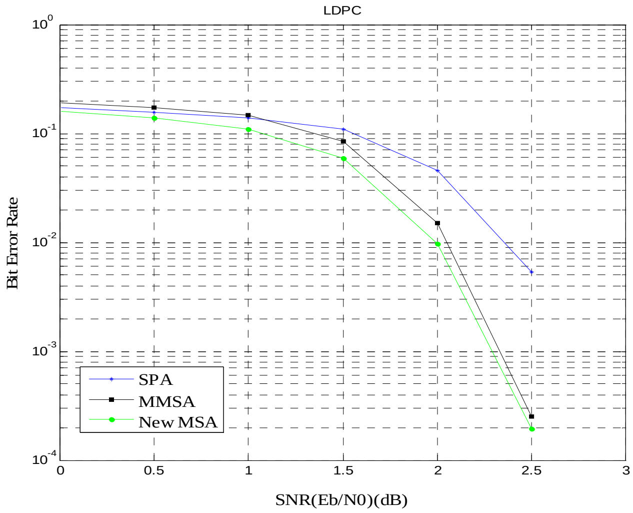


Figure 2.18 BER performance of two-way normalized min-sum for (1296, 864) LDPC code (sf1=0.5; sf2=0.25)

The comparison shows that for the same scaling factors, the two-way normalized min-sum decoding algorithm outperforms even the code rate and length is varied. Furthermore, it is better in error performance, so it can be adopted for many practical applications.

Table 2. 1Performance Analyses and Comparison

Algorithm	BER Performance for (code length ,rate, SNR db)		
	(1024 , ½, 4.5)	(1296, 2/3, 2.5)	(648, ½, 4.0)
SPA	0.002029	0.005335	0.0004375
MMSA	0.001553	0.0002492	0.000137
New MSA	0.0005937	0.0001659	.00004969

2.8 Chapter Summary

This chapter discusses the basics of LDPC codes and its historical development. This also discusses in detail the mathematical improvements to the LDPC coded information. The SPA algorithm after its simplification to the Min-Sum algorithm (MSA) made it possible to use and implement it in practical applications. The tanner graph representation demonstrates well the good performance LDPC codes with high girth. The last section is the contribution to this chapter to improve further the LDPC codes performance.

Chapter 3 Quasi Cyclic LDPC Codes

3.1 Introduction

Although low density parity check codes have gotten tremendous popularity among the scientists and engineers but the problem of encoding complexity is also one of the obstacles for their wide range commercial applications. Another disadvantage of general LDPC codes is that of inefficient large memory requirement for storing their parity check matrices. Quasi- cyclic (QC) LDPC codes are the good candidate to solve the memory problem as their parity check matrices consists of circulant permutation matrices[71] or the zero matrix. QC LDPC codes have also shown performance[72] close to Shannon limit. QC LDPC is becoming popular among the hardware implementation related researchers and many efficient encoding[16, 21, 22, 73, 74] methods for implementation has been proposed.

3.2 Constructing Quasi Cyclic Matrices

Several methods have been suggested for constructing QC-LDPC codes. The structure of the codes depends on the arrangement of the constituents' sub-matrices and their shift values. Two constraints are always kept in mind while designing LDPC codes; 1) High girth to improve the performance, 2) low complexity in implementation. Some of the techniques include finite geometry[75, 76] , algebraic construction [77] , finite field approach[78].

The QC-LDPC parity check matrix has the general structure as

$$H_{QC} = [C_1, C_2.....C_i] \qquad 3.2.1$$

Where $C_1, C_2.....C_i$ are all circular shifted matrices or circulants in such a way that the parity check matrix is full rank. The easiest approach is to find each circulant by identity sub-matrices. Shifted identity matrices are easily obtained by shifting the row of an identity matrix to the right or left by some amount. Some arrangement of the identity matrices are shown as

$$\begin{pmatrix} I_{11} & I_{12} & I_{13} & I_{14} \\ I_{21} & I_{22} & I_{23} & I_{24} \\ I_{31} & I_{32} & I_{33} & I_{34} \end{pmatrix} \qquad 3.2.2a$$

$$\begin{pmatrix} I_{11} & O & I_{13} & I_{14} \\ O & I_{22} & I_{23} & I_{24} \\ I_{31} & I_{32} & O & I_{34} \\ I_{41} & I_{42} & I_{43} & O \end{pmatrix} \qquad 3.2.2b$$

Equation 3.2.2a is with all non-zero sub-matrices and Equation 3.2.2b is with zero sub-matrices. Each sub-matrix in a row shows the weight one and similarly for each column. In the equation 3.2.2a, the row and column weights are 3 and 4 respectively while in the Equation 3.2.2b, the row and column weights are 3 and 3 respectively due to the placement of the zero sub-matrices.

The circulant matrix (or may be the identity matrix) of size $m \times m$ can be shown as

$$C = \begin{pmatrix} c_0 & c_1 & \cdots & c_{m-1} \\ c_{m-1} & c_0 & \cdots & c_{m-2} \\ \vdots & \vdots & \cdots & \vdots \\ c_1 & c_2 & \cdots & c_0 \end{pmatrix} \qquad 3.2.2c$$

A circulant matrix is uniquely specified by a polynomial formed by the entries of the first row

$$c(x) = c_0 + c_1 x + c_2 x^2 + \cdots\cdots c_{m-1} x^{m-1} \qquad 3.2.2d$$

For a rate $1/k$ systematic QC code has the $m \times mp$ generator matrix of the form

$$G_{QC} = [I_m, p_1, p_2 \cdots p_{k-1}] = [I_{m\times m} \quad P_{m\times mp}] \qquad 3.2.2e$$

$$p_j = s_1 G_{1,j} + s_2 G_{2,j} + \cdots\cdots s_m G_{m,j} \qquad 3.2.2f$$

Where I_m is the $m \times m$ identity matrix and C_i is the $m \times m$ binary circulant matrices. After getting the generator matrix, the encoding is simply done as

$$X = s.G_{QC} = s.[I \quad P] = [s \;\; p\;] \qquad 3.2.2g$$

Where s contains the information bits.

3.3 QC LDPC Encoder Design

The parity check matrix for QC LDPC codes[21] is represented as

$$H_{QC} = [C_1, C_2 .....C_i] \qquad 3.3.1$$

This matrix can be decomposed into two parts

$$H_{QC} = \begin{bmatrix} M & D \end{bmatrix} \qquad 3.3.2$$

Where D is a square matrix and must be invertible. D and M both are quasi cyclic, composed of circulant. The desired generator matrix G has the following form

$$G_{QC} = \begin{bmatrix} I & P \end{bmatrix} \qquad 3.3.3$$

The necessary and sufficient condition for the generator matrix is that

$$G_{QC} H_{QC} = [0] \qquad 3.3.4$$

The codeword $X = [s \;\; p]$ has two parts ; the information bits s and the parity bits p such that

Thus X is a codeword if and only if

$$\begin{bmatrix} I \\ P \end{bmatrix} \begin{bmatrix} M & D \end{bmatrix} = 0 \qquad 3.3.5$$

Or equivalently

$$MI^T + DP^T = 0 \qquad 3.3.6$$

$$\Rightarrow P^T = D^{-1}MI = D^{-1}M \quad \text{(Mod 2)} \qquad 3.3.7$$

If a matrix is circular or composed of circular permutation matrices, the inverse is also a circular matrix[79] which can be obtained as follows:

The cyclic matrix in equation 3.2.2c is such that $C^T = C$ and $X^T = (x_1, x_2, \cdots, x_n)^T$ is a matrix of polynomial such that

$$CX^T = (1, 0, 0, \cdots\cdots, 0)^T \qquad 3.3.8$$

This can be written in the polynomial form as

$$\begin{aligned} &c_0 x_1 + c_2 x_2 + \cdots\cdots + c_{m-1} x_n = 1 \\ &c_{m-1} x_1 + c_0 x_2 + \cdots + c_{m-2} x_n = 0 \\ &\vdots \;\; \vdots \;\; \vdots \;\; \vdots \;\; \vdots \;\; \vdots \;\; \vdots \;\; \vdots \\ &c_1 x_1 + c_2 x_2 + \cdots\cdots\cdots + c_0 x_n = 0 \end{aligned} \qquad 3.3.9$$

Solving the equation (3.3.9), we obtain the inverse C^{-1} of C written in the form

$$C^{-1} = \begin{pmatrix} x_1 & x_2 & \cdots & x_n \\ x_2 & x_3 & \cdots & x_1 \\ \vdots & \vdots & & \vdots \\ x_n & x_1 & \cdots & x_{n-1} \end{pmatrix} \qquad 3.3.10$$

This can be verified by multiplying the equations (3.2.2c) and (3.3.10) such that

$$C^{-1}C = I(indentity\ matrix) \qquad 3.3.11$$

$$\begin{pmatrix} c_0 & c_1 & \cdots & c_{m-1} \\ c_{m-1} & c_0 & \cdots & c_{m-2} \\ \vdots & \vdots & \cdots & \vdots \\ c_1 & c_2 & \cdots & c_0 \end{pmatrix} \begin{pmatrix} x_1 & x_2 & \cdots & x_n \\ x_2 & x_3 & \cdots & x_1 \\ \vdots & \vdots & & \vdots \\ x_n & x_1 & \cdots & x_{n-1} \end{pmatrix} = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & 0 \end{pmatrix} \quad 3.3.12$$

This method can be used for obtaining the D^{-1} in equation (3.3.7).

The structure of the D matrix plays an important role in the QC LDPC parity check matrix. There are several methods to construct regular and irregular quasi-cyclic LDPC codes. In[80] it was shown that QC LDPC codes based on D-matrix and Q-matrix ,designed by modified PEG, are more suitable to be used than identity matrix as they outperform that of identity matrix .

3.3.1 QC LDPC Encoder Implementation

QC–LDPC codes have encoding advantage over conventional LDPC codes and their encoding can be carried out by shift register[21, 81] called shift register adder accumulate(SRAA) with complexity linearly proportional to the number of parity bits as shown in the equation(3.2.2e) of the code .Additionally QC-LPDC codes require less amount of memory as compared to the general LDPC codes, since their parity check matrices consist of the circulant permutation matrices or the zero matrices.

3.4 QC LDPC Decoder

QC LDPC codes are the family of implementation oriented codes. QC LDPC codes have not only efficient encoding characteristic but has also solved the problem of decoding complexity. QC LDP codes have got the following advantages;

1) Memory efficient
2) Less Hardware Complexity
3) High Convergence Speed through Layered decoding
4) High Throughput

3.4.1 Layered and Non-layered LDPC Codes

All the decoding algorithms described in chapter 2 are non-layered LDPC codes. In those algorithms (BP, SPA and MSA) at the kth iteration ,the check-to-bit(check node) messages are updated by using the values of bit-to-check(variable node) messages obtained at the $(k-1)th$ iteration during horizontal processing .Then all the values of the bit-to-check messages are updated by using the values of the check-to-bit newly calculated at the kth iteration. Layered schedule considers the parity check matrix as layers of check equations and update the variable node information right after check node information of current layer. Two types of layer decoding has been considered in [82]. Many different approaches has been made to improve it further in terms of memory , energy(power) and throughput[83-88].

The straightforward horizontal layered min-sum decoding algorithm is given below:

1) Initialization: $L_{ji}^{0} = L_{i}^{0} = Y$ & $\mathrm{C}_{ji}^{0} = 0$ (as stated in chapter 2) where $i = 0,1,2 \cdots N-1$

2) Set maximum iteration $k = 1$ to $I_{\max}$

a) Layer initialization: (Here, in general each row of a parity check matrix is considered as one layer). Set $t = 1$ to $\mathrm{t}_{\max}$ (No. of Rows or Row Blocks)

$$V_{ji}^{t} = L_{i}^{t-1} - C_{ji}^{t-1} \qquad 3.4.1$$

b) Check Node Update:

$$C_{ji}^{t} = s.f \prod_{i' \in N(j)\backslash i} sign(V_{Vji'}^{t}) \min_{i' \in N(j)\backslash i} |V_{Vji'}^{t}| \qquad 3.4.2$$

c) Variable Node update :

$$L_{i}^{t} = Y + \sum C_{ji}^{t} \qquad 3.4.3$$

Go to step a until M reaches.

$$\hat{X} = \begin{cases} 1 & \mathrm{L}_{i}^{t_{\max}} < 0 \\ 0 & \mathrm{L}_{i}^{t_{\max}} \geq 0 \end{cases} \qquad 3.4.4$$

If $rem(\hat{X}.H^{T},2) = rem(\hat{x}_1, \hat{x}_2, ...\hat{x}_n.H^{T},2) = 0$, or if the maximum number of iterations is reached, then stop, else, continue iteration from Step 2.

3.4.2 Memory Efficient Layered LDPC Decoding

To minimize the interconnect complexity and memory size at the check node update (horizontal processing), the MSA can be formulated in such a way not to store all the messages but only the following four elements[89, 90].

i) The 1st smallest magnitude

ii) The 2nd smallest magnitude

iii) The index of the 1st smallest magnitude

iv) The signs of all the soft message of the row(variable messages).Note: This may not require in software simulation only in case quantized values are not used.

In the check to variable message passing phase, a check node c sends only the 1st smallest magnitude (*min1*), the 2nd smallest magnitude (*min2*) and the index of the 1st smallest magnitude (*index*) where *min1*$\leq$ *min2*.

Now the check node (step 2) is updated in the layered decoding in section (3.4.1) as

$$C_i^t = \begin{cases} sf.sign(V_i^t).\min 1 & if\ \ i == index \\ sf.sign(V_i^t).\min 2 & ohterwise \end{cases} \qquad 3.4.5$$

3.5 QC LDPC Performance Analysis

Consider a quasi cyclic parity check matrix $H_{520\times1040}$ which has the each sub-matrices (circulant) with $size(m',n') = 130$ and LDPC code length $L = 1500$, $d_v = 4$ and $d_c = 10$. The code has been simulated for SPA , MSA with normalized and offset values and with the new improved MSA[70].

$H = [M \mid D]$ where D is square matrix such that it is full rank and is represented as

$D = \begin{pmatrix} D_1 & \cdots & 0 \\ \vdots & D_2 & \vdots \\ 0 & \cdots & D_i \end{pmatrix}$, the diagonal elements are designed such that the parity check matrix H is regular and the upper and lower triangular values are zero matrices.

$$H = \begin{bmatrix} C_{11} & C_{12} & C_{13} & C_{14} & C_{15} & C_{16} & D_1 & 0 & 0 & 0 \\ C_{21} & C_{22} & C_{23} & C_{24} & C_{25} & C_{26} & 0 & D_2 & 0 & 0 \\ C_{31} & C_{32} & C_{33} & C_{34} & C_{35} & C_{36} & 0 & 0 & D_3 & 0 \\ C_{41} & C_{42} & C_{43} & C_{44} & C_{45} & C_{46} & 0 & 0 & 0 & D_4 \end{bmatrix} \qquad 3.3.1$$

Where $C_{150,150}$ is a circulant permutation matrix.

The QC LDPC code is simulated for SPA, simple MSA , layered normalized MSA which is based on the memory efficient MSA as stated in section 3.4..2 , offset MSA and new improved MSA. The value of the offset is chosen by search and is considered here as the approximately good one and there may be exist some other optimum or near optimum values. Also the normalized values for MSA are chosen such that it give good performance as well as the hardware implementation is also simple abut may have some other more optimum values exists.

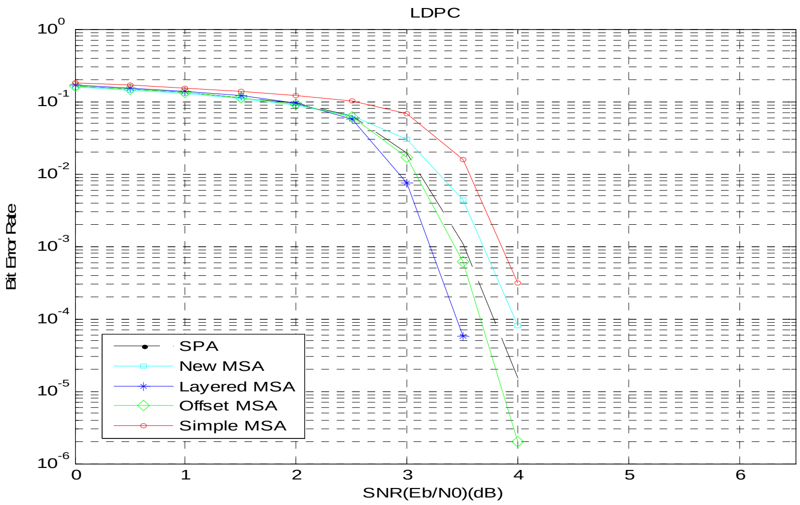


Figure 3.1 Simulations for QC LDPC code (1500, 4, 10), iterations=10

3.6 Chapter Summary

The main objective of this chapter is to introduce practical LDPC codes which are easily implementable and have good performance. This also mentions the encoding for QC LDPC which is different than conventional LDPC in finding the generator matrix for which we need to get the inverse matrix. This summarizes the QC LDPC as both row by row layered or block by block layered for fast convergence. The last section of the chapter gives the performance graphs for QC LDPC for SPA ,MSA , New MSA ,simple MSA and memory efficient layered MSA decoder.

Chapter 4 Practical LDPC Codes

4.1 Introduction

Quasi cyclic LDPC codes have the advantage of both encoding and decoding over the conventional LDPC (non-structured) as it reduce greatly the hardware complexity as it has simplified the check and variable node interconnection. QC LDPC has limited the decoding performance compared to unstructured random LDPC codes. This is due to the limitation in constructing the QC matrices with large girth although it can be improved by PEG based QC matrices. Different architecture have been proposed to improve the throughput as well reduce the complexity.

4.2 LDPC Decoder Architecture

Different approaches are made to implement the LDPC codes in hardware looking at requirements of the market. Some researchers emphasized the high throughput but overlooked the complexity while others proposed the very low area, energy efficient but with inefficient low throughput. The following graph[35] in figure 4.2 best defines the architectural and throughput along with complexity trade off. Parallel architecture gives extremely high power dissipation and high throughput and need large hardware area due to complex interconnection. On the other side serial architecture offers very low power dissipation, require less hardware. QC LDPC makes the tradeoff between serial and parallel architecture.

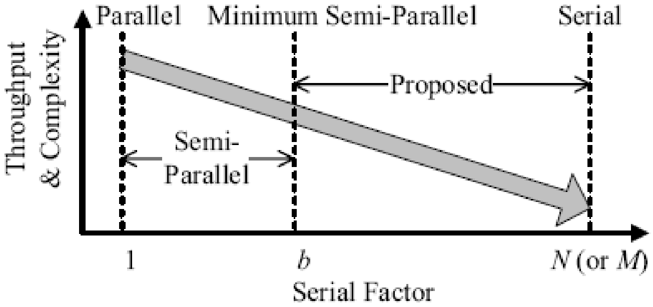
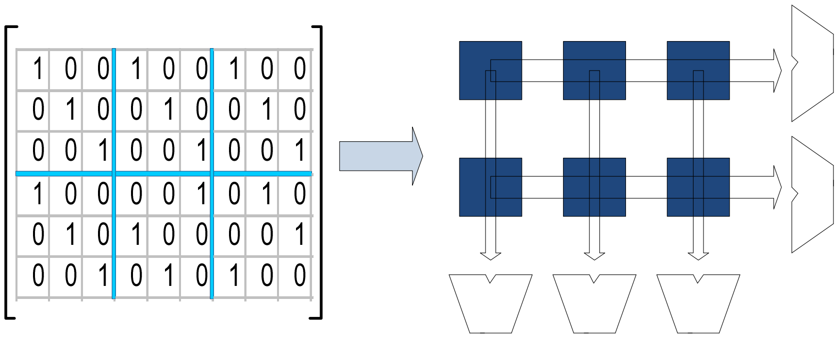


Figure 4.1 Parallel to Serial vs complexity & throughput: b is the sub-matrix size; M and N are the row and column of parity check matrix.

4.2.1 Semi Parallel Architecture

Semi parallel architecture [84, 86, 89, 91] are often designed for structured codes. Structured codes have the inherit advantage that could be used to reduce hardware implementation. The structure of a code affects the interconnection network between variable and check nodes. Quasi cyclic LDPC codes are the type of structured codes that have less hardware complexity and cost of both the encoder and decoder. The cyclic shifting of the identity sub-matrices of the QC LDPC parity check matrix, simplifies routing and addressing of messages within processing nodes. Decoder architecture mainly differ in processing node inter-connection, communication scheduling and node implementations.

Semi parallel decoding architecture employs a time sharing scheme of processing units to reduce hardware complexity. To demonstrate the parallelism between the block of the QC LDPC ,the parity check matrix is shown as an example . In this example, the parity check matrix is divided into two horizontal layers and three vertical layers. There is a serial processing inside each block and a parallel processing between blocks. In figure 4.2 , the serial factor of the parity check matrix is $b = 3$ which is the size of the circulant .This type of architecture is called semi-parallel.



$$\left[\begin{array}{ccc|ccc|ccc}
1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\
0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\
0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\
\hline
1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\
0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0
\end{array}\right]$$

Figure 4.2 QC LDPC Parity check matrix and its semi-parallel processing hardware

When the serial factor becomes equal to *M (or N)* then it is totally serial architecture and if it becomes equal to 1, then the architecture is fully parallel. The complexity can be reduced further if the gap between serial and semi-parallel architecture is efficiently utilized.

4.2.2 Hardware Realization of LDPC Decoder

Layered decoder is one of the practical decoder used for fast convergence and is also memory efficient. The flow for the layered decoder is show in the figure 4.3. V2C is the message from variable to check node, C2V is check to variable node message passing and Est.code is the estimated codeword

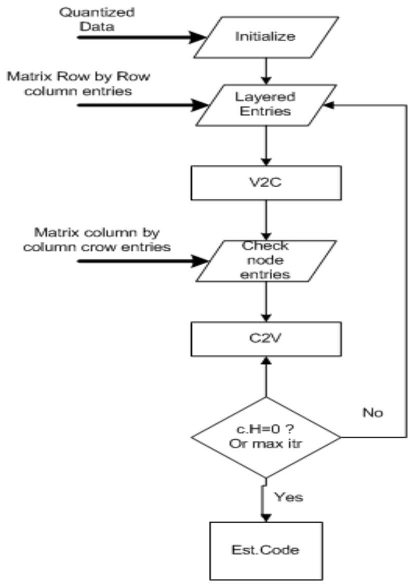


Figure 4.3 Data flow of a typical layered decoder

There is some other more efficient technique in which both row and column offsets are not required, only one shift matrix and the shifting information[41] is required.

The offset matrix in figure 4.2 can be written as the offset entries for each of the circulant matrices

$$offsets = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 & 1 \end{bmatrix} \Leftrightarrow H = \begin{bmatrix} I_0 & I_0 & I_0 \\ I_0 & I_2 & I_1 \end{bmatrix} \qquad 4.2.1$$

Now the figure 4.3 can also be shown with this offset control matrix in the figure 4.4.

4.2.2.1 Clipping and Quantization :The Finite word length of the soft information[92] as one of the major factors affects the size of the memory, the complexity of computation logic, routing complexity, and the decoding performance of an LDPC code. It also decides the size of memory to store the intrinsic and extrinsic messages and determines the overall implementation area in the partially parallel (semi-parallel) LDPC decoder. Therefore, the reduction of the finite word length without significant performance loss can decrease the hardware size which includes the computation logic and memory banks.

For a binary BPSK modulated data $\{\pm 1\}$, transmission over AWGN corrupts this data and then the data is no more integer and shaped as floating point. The received values are clipped symmetrically[93] at a certain threshold $\{a\}$ and then uniformly quantized in the range $\{a_{th}, -a_{th}\}$. There are $2^q - 1$quantization intervals, symmetric with respect to the origin and each represented by q quantization bits. Integer numbers $-(2^q - 1)$ to $+(2^q - 1)$ are assigned and at bit nodes, an outgoing message is clipped to $\pm(2^q - 1)$ if it exceeds the threshold.

Table 4. 1: Comparison of uniform and non-uniform quantization for LDPC decoder

Decimal values (After AWGN)	(6:3) Uniform Quantization	5-bits Non-uniform Quantization	(5:3) Uniform Quantization	4-bits Non-uniform quantization
± 0.000	s00000	s0000	s0000	s000
± 0.125	s00001	s0001	s0001	s001
± 0.250	s00010	s0010	s0010	s010
± 0.375	s00011	s0011	s0011	s011
± 0.500	s00100	s0100	s0100	
± 0.625	s00101	s0101	s0101	s100
± 0.750	s00110	s0110	s0110	
± 0.875	s00111	s0111	s0111	s101
± 1.000	s01000		s1000	
± 1.125	s01001	s1000	s1001	
± 1.250	s01010		s1010	s101
± 1.375	s01011	s1001	s1011	
± 1.500	s01100		s1100	
± 1.625	s01101	s1010	s1101	
± 1.750	s01110		s1110	s111
± 1.875	s01111	s1011	s1111	
± 2.000	s10000			
± 2.125	s10001			
± 2.250	s10010	s1100		
± 2.375	s10011			
± 2.500	s10100			
± 2.625	s10101			
± 2.750	s10110	s1101		
± 2.875	s10111			
± 3.000	s11000			
± 3.125	s11001			
± 3.250	s11010	s1110		
± 3.375	s11011			
± 3.500	s11100			
± 3.625	s11101			
± 3.750	s11110	s1111		
± 3.875	s11111			

A good tradeoff between hardware complexity and decoding performance is given with 6-bits quantization scheme for both intrinsic and extrinsic messages, which are uniformly quantized with 1 sign bit, 2 integer bits and 3 fractional bits.

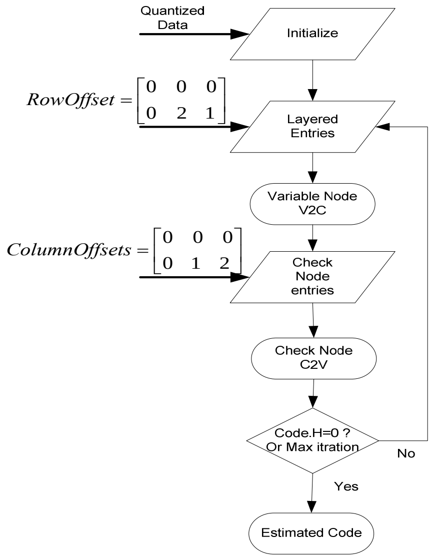

Quantized Data
Initialize
$RowOffset = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 & 1 \end{bmatrix}$
Layered Entries
Variable Node V2C
$ColumnOffsets = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 2 \end{bmatrix}$
Check Node entries
Check Node C2V
Code.H=0 ? Or Max itration
No
Yes
Estimated Code

Figure 4.4 Layered Decoder with Offsets entries

This fixed size quantization of 6-bits shows good performance. Some papers have suggested some adaptive quantization[94] techniques for either intrinsic or extrinsic messages. In an LDPC decoder both intrinsic (channel values) and extrinsic information (C2V messages) has to be stored. If we can use less number of quantization bits, the storage complexity will be significantly reduced. In fact, using less quantization bits can greatly simplify the hardware implementation but will degrade the error performance of LDPC decoder. Adaptive quantization can be used to get better BER performance.

4.2.2.2 Variable and Check Node Hardware Units

At each iteration, the check update node unit (CNU) is computing the sign and the absolute min values. It finds the smallest two inputs and the index of the smallest one. CNU function is to find the first minimum, 2nd minimum and the index of the 1st minimum for each row process. The hardware for the CNU process is shown in the figure 4.1. After finding that values , these values are stored to be used in variable node update unit (VNU). In VNU the input messages are firstly transferred to two complement format and then do the add operation. Finally they are transferred back to sign and magnitude format and is scaled.

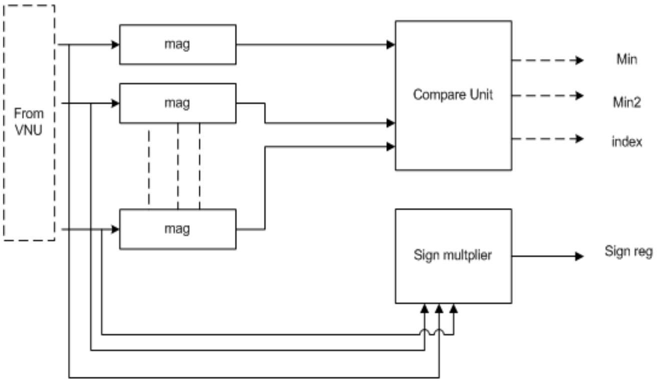


Figure 4.5 CNU Architecture to find the minimum values and the signs

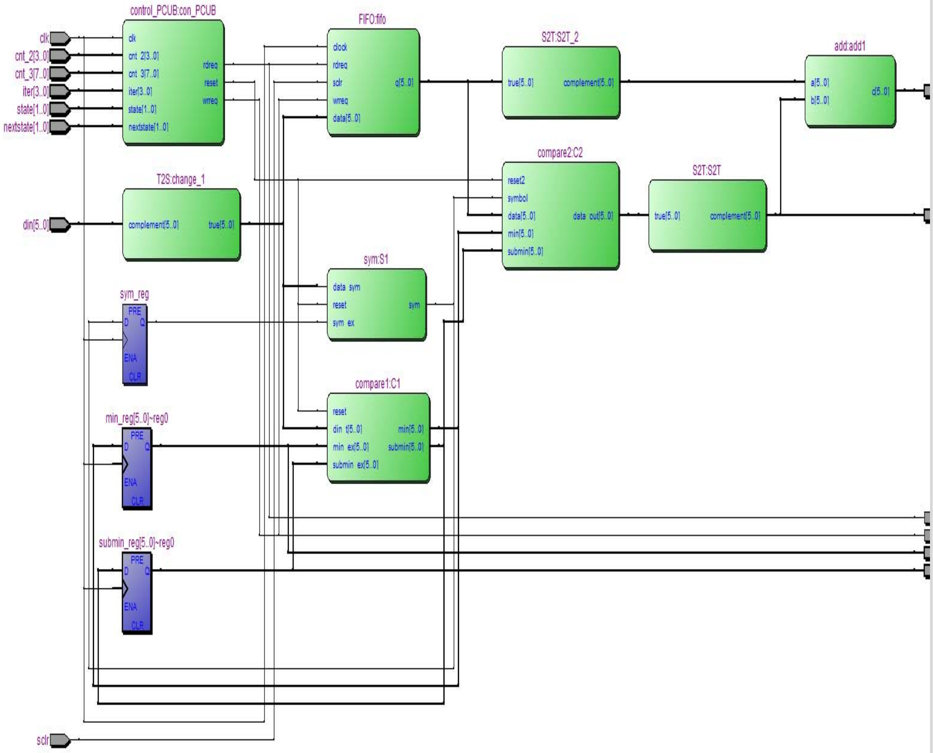


Figure 4.5a. Verilog RTL view of CNU

Figure 4.5a is the RTL view of CNU. The data out of CNU goes to sign comparator and the data input is CNU is stored in FIFO.

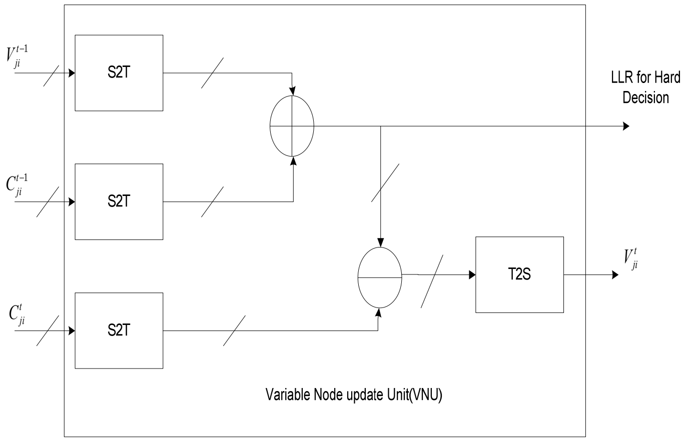


Figure 4.6 VNU detail for updating the variable message and data out for hard decision

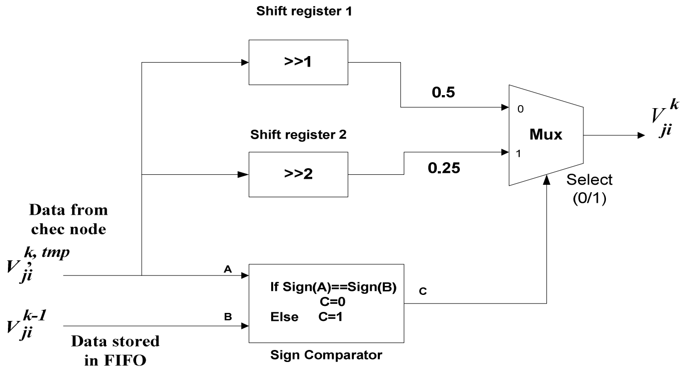


Figure 4.7 Hardware implementation for normalized values based on sign comparator

4.3 Case Study

In this section discuss the QC LDPC implementation study overview. The section introduce some concepts although which can be implemented on hardware (FPGA). A new method[70] is suggested here for LDPC decoder on FPGA and other hardware. In figure 4.7, a new hardware to scale and update the variable message is shown. The scaling factor multiplication complexity is the same as used in already existing hardware LDPC decoders. It uses two different scaling factors which give better approximation in decoding .The increase in hardware complexity is the sign comparator , FIFO and

multiplexer. This circuitry do not add time latency as sign comparator implemented as exclusive OR gate and multiplexer delay is not that significant. The scaling factor chosen are according to the message overestimation and hardware implementation.

We see that the signs are compared as XOR gate and the scaling factors are basically implemented as data bus shifting .The only problem is with FIFO as it is required to temporarily store the message and then utilized for comparison. FIFO need to be synchronized properly as the date to VNU comes from CNU is fast .Some alternate solutions can be found to store the date temporarily if FIFO seems not a good solution. One module is required for each row block of a QC LDPC parity check matrix used in decoding. Figure 4.9 shows the FIFO in connection with data1 in for sing

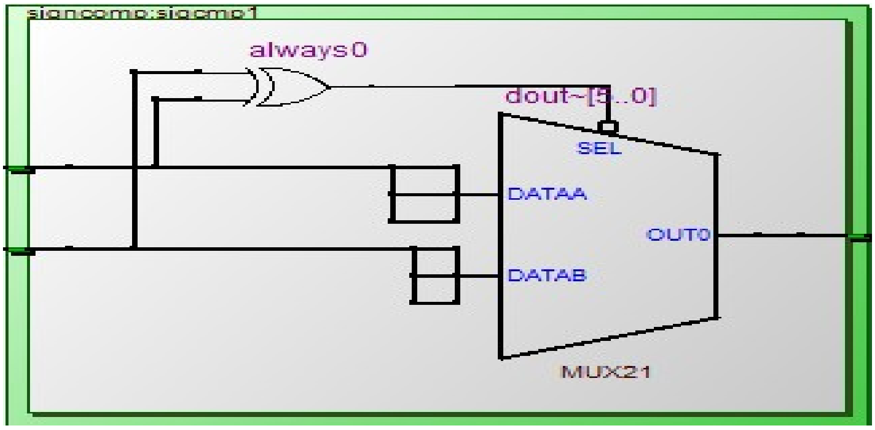


Figure 4.8 RTL Verilog view of sign comparator

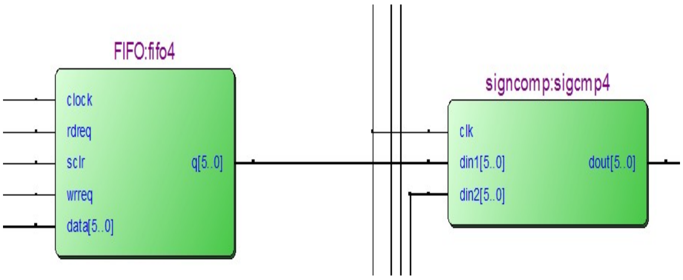


Figure 4.9 FIFO for temporarily storing data for sign comparison

comparison and then scaled accordingly.

4.4 Chapter Summary

This chapter is about the hardware implantation of the LDPC decoder only. This gives the insight into the existing methods for requirements such has energy and area efficient, fast proceeding, less complexity and good performance.

Chapter 5 Application to Communication Systems

5.1 MIMO Communication

With the advent of wireless communication, efforts have always been made to transmit maximum data with maximum reliability. To achieve the maximum data rate, MIMO wireless systems have gained popularity as its theoretic-capacity increase linearly with increase in the number of antennas. The error performance of MIMO system can be greatly improved by error correction code[95]. The multiple-input and multiple-output (MIMO) is the use of multiple antennas at both the transmitter and receiver to improve communication performance. It is one of several forms of smart antenna technology. MIMO technology has attracted attention in wireless communications, because it offers significant increases in data throughput and link range without additional bandwidth or transmit power. Because of these properties, MIMO is an important part of modern wireless communication standards such as IEEE 802.11n (Wifi), WiMAX etc.

Consider a flat fading MIMO system model with N_t transmit and N_r receive antennas .The received signal vector at each instant of time is given by:

$$r = Hx + n \qquad 5.1.1$$

.Where r is $N_r \times 1$ received signal vector, H is a $N_r \times N_t$ channel response matrix, x is a $N_t \times 1$ transmitted signal vector and n is the additive white Gaussian noise (AWGN).Typical MIMO communication system is show in Figure 4.1 and figure 4.1a shows a typical LDPC coded MIMO communication system.

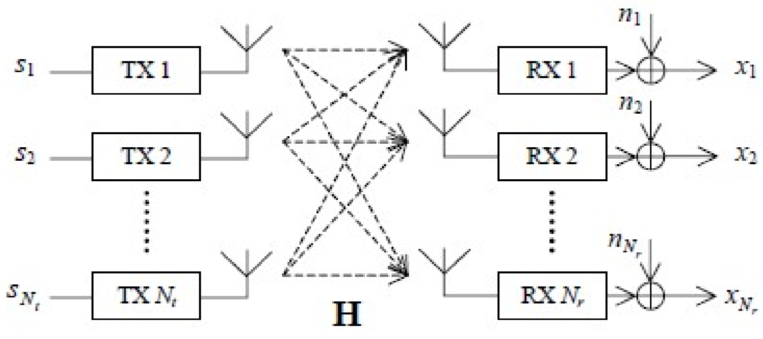


Figure 5.1 MIMO communication system

Figure 4.2 represents a 2×2 MIMO system[96] with 2 allocated transmit antenna and 2 allocated receive antenna. Consider that we have a transmission sequence, for example. $x_1, x_2 \cdots x_n$. In normal transmission, we will be sending x_1 in the first time slot, x_2 in the second time slot, x_3 and so on. However, as we now have 2 transmit antennas, we may group the symbols into groups of two. In the first time slot, send x_1 and x_2 from the first and second antenna. In second time slot, send x_3 and x_4 from the first and second antenna, send x_5 and x_3 in the third time slot and so on. Notice that as we are grouping two symbols and sending them in one time slot, we need only $n/2$ time slots to complete the transmission – **data rate is doubled**! This forms the simple explanation of a probable MIMO transmission scheme with 2 transmit antennas and 2 receive antennas. The two transmitted symbols interfered with each other called inter channel interference (ICI). The channel is flat fading – In simple terms, it means that the multipath channel has only one tap. So, the convolution operation reduces to a simple multiplication and the channel experience by each transmit antenna is independent from the channel experienced by other transmit antennas. For the ith transmit antenna to jth receive antenna, each transmitted symbol gets multiplied by a randomly varying complex number h_{ji} .As the channel under consideration is a Rayleigh channel, the real and imaginary parts of h_{ji} are Gaussian distributed having mean $\mu_{h_{ji}} = 0$ and variance $\sigma^2_{h_{ji}} = 1/2$ .The channel experienced between each transmit to the receive antenna is independent and randomly varying in time.

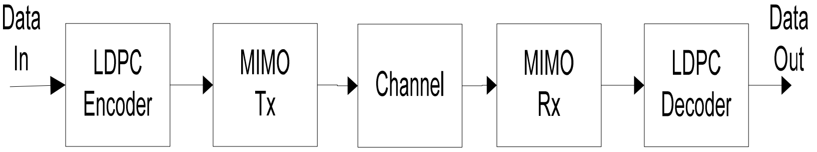


Figure 5.1a Typical LDPC code MIMO communications

On the receive antenna, the noise n has the Gaussian probability density function with

$$p(n) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp^{\frac{-(n-\mu)^2}{2\sigma^2}} \qquad 5.1.2$$

Where $\mu = 0$ and $\sigma^2 = \frac{N_0}{2}$.

The channel h_{ji} is known at the receiver.

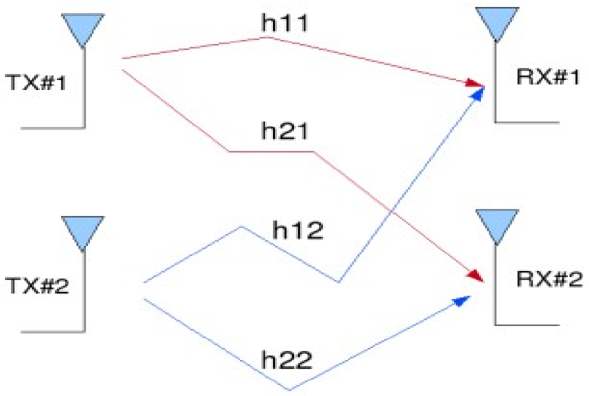


Figure 5.2 2 Transmitter, 2 receiver MIMO system (2x2 channel MIMO)

5.1.1 Zero forcing (ZF) equalizer for 2×2 MIMO channel

Let us now try to understand the math for extracting the two symbols which interfered with each other. In the first time slot, the received signal on the first receive antenna is,

$$y_1 = h_{11}x_1 + h_{12}x_2 + n_1 = [h_{11}, h_{12}]\begin{bmatrix} x_1 \\ x_1 \end{bmatrix} + n_1 \qquad 5.13$$

The received signal on the second receive antenna is,

$$y_2 = h_{21}x_1 + h_{22}x_2 + n_2 = [h_{21}, h_{22}]\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + n_2 \qquad 5.14$$

where

y_1, y_2, are the received symbol on the first and second antenna respectively,

h_{11} is the channel from 1st transmit antenna to 1st receive antenna,

h_{12} is the channel from 2nd transmit antenna to 1st receive antenna,

h_{21} is the channel from 1st transmit antenna to 2nd receive antenna,

h_{22} is the channel from 2nd transmit antenna to 2nd receive antenna,

x_1, x_2 are the transmitted symbols and

n_1, n_2 is the noise on 1st and 2nd receive antennas.

We assume that the receiver knows h_{11} , h_{12} , h_{21} and h_{22} . The receiver also knows y_1 & y_2 . The unknowns are x_1 & x_2 . So we have two equations and two unknowns. For convenience, the above equation can be represented in matrix notation as follows:

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} n_1 \\ n_2 \end{bmatrix} \qquad 5.1.5$$

Equivalently,

$$r = Hx + n \qquad 5.1.1a$$

To solve for x , we know that we need to find a matrix W which satisfies $WH = I$. The Zero Forcing (ZF) linear detector for meeting this constraint is given by,

$$W = (H^H H)^{-1} H^H \qquad 5.1.6$$

This matrix is also known as the pseudo inverse for a general m x n matrix.

The term,

$$\boldsymbol{H}^H \boldsymbol{H} = \begin{bmatrix} h_{1,1}^* & h_{2,1}^* \\ h_{1,2}^* & h_{2,2}^* \end{bmatrix} \begin{bmatrix} h_{1,1} & h_{1,2} \\ h_{2,1} & h_{2,2} \end{bmatrix} = \begin{bmatrix} |h_{1,1}|^2 + |h_{2,1}|^2 & h_{1,1}^* h_{1,2} + h_{2,1}^* h_{2,2} \\ h_{1,2}^* h_{1,1} + h_{2,2}^* h_{2,1} & |h_{1,2}|^2 + |h_{2,2}|^2 \end{bmatrix}. \qquad 5.1.7$$

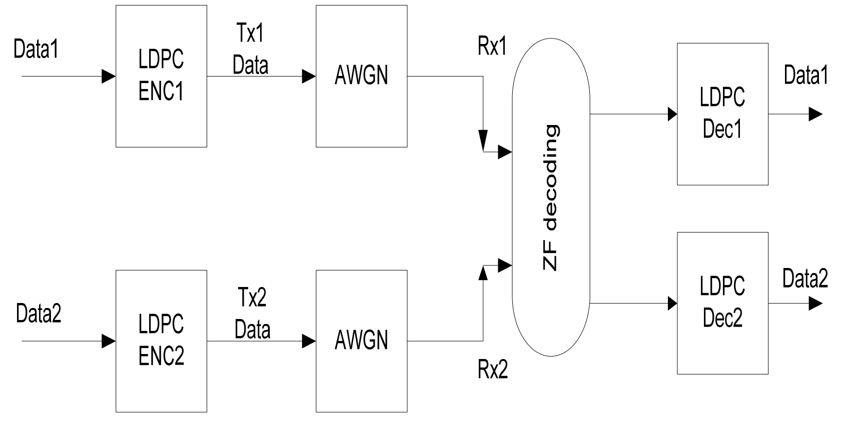


Figure 5.3 . 2x2 LDPC coded MIMO System with ZF equalizer

5.1.2 LDPC coded 2×2 MIMO channel with Zero forcing (ZF) Equalizer

The performance of the MIMO system can be enhanced by using state of the art error correction technique like QC LDPC as shown in figure 5.3.

The system has been simulated for a QC LDPC parity check matrix $H_{261\times522}$ such that code length=522 and each sub-matrix size is 87.

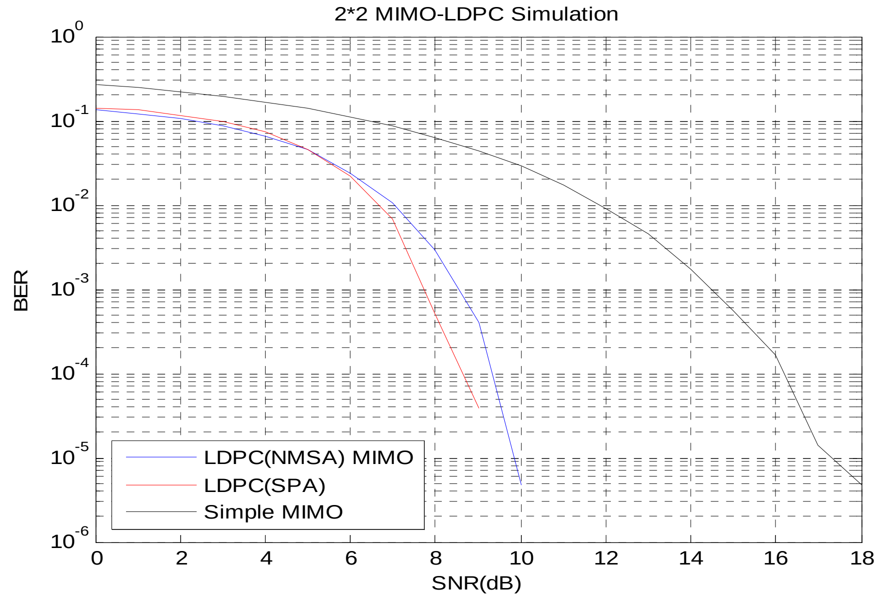


Figure 5.4 Performance comparison of LDPC coded MIMO and simple MIMO with ZF

The graph in figure 5.4 shows the improved performance for LDPC code MIMO in comparison to simple MIMO with ZF decoding. The LDPC decoding algorithm used are the standard SPA and the new MSA [70] as the new MSA or the improved MSA shows better performance for medium and short length codes and are suitable for the MIMO channels to split data in small packets and transmit independently.

5.2 Coded Cooperative Communication

Cooperative communication is one of the fastest growing areas of research, and it is likely to be a key enabling technology for efficient spectrum use in future. The key idea in user-cooperation is that of resource-sharing among multiple nodes in a network. The reason behind the exploration of user-cooperation is that willingness to share power and computation with neighboring nodes can lead to savings of overall network resources. Mesh networks provide an enormous application space for user-cooperation strategies to be implemented. In traditional communication networks, the physical layer is only responsible for communicating information from one node to another. In contrast, user-cooperation implies a paradigm shift, where the channel is not just one link but the network itself. The current chapter summarizes the fundamental limits achievable by cooperative communication, and also discusses practical code constructions that carry the potential to reach these limits. Cooperation is possible whenever the number of communicating terminals exceeds two. Therefore, a three-terminal network is a fundamental unit in user-cooperation. Two features differentiate cooperative transmission[97] schemes from conventional non-cooperative systems: 1) the use of multiple users' resources to transmit the data of a single source; and 2) a proper combination of signals from multiple cooperating users at the destination, where we have two users transmitting their local messages to the destination over independent fading channels. Suppose that the transmission fails when the channel enters a deep fade, i.e., when the signal-to-noise ratio(SNR) of the received signal falls below a certain threshold, as indicated with the grey region in Figure 5.5. If the two users cooperate by relaying each others' messages and the inter-user channel is sufficiently reliable, the communication outage occurs only when both users experience poor channels simultaneously.

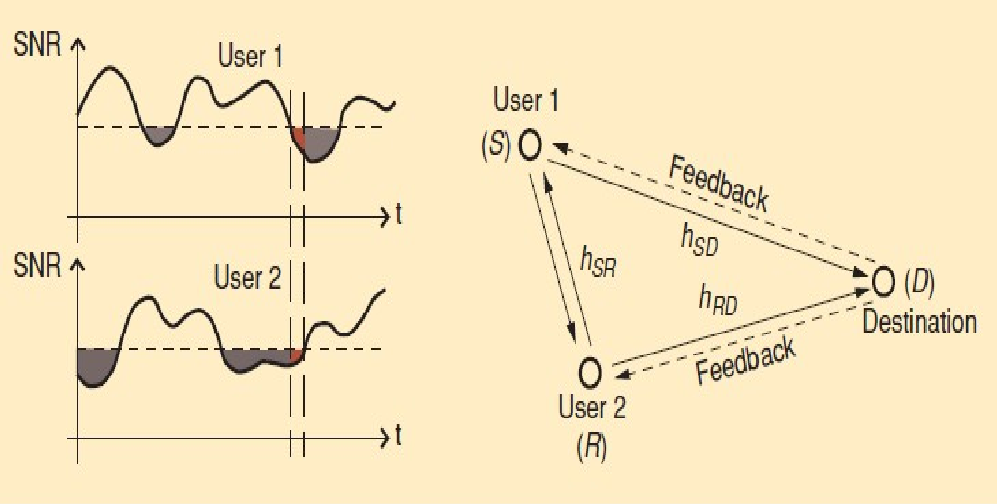


Figure 5.5 A basic cooperative network model

5.2.1 LDPC Coded Cooperative Communication

Consider a coded one-relay cooperation[98] shown in Figure 5.6, where the codeword bits encoded by the first encoder(Enc-1), and is sent to the relay (S-R) and the destination (S-D), respectively. The decoder, denoted by Dec-2, in the relay first decodes the incoming signal corrupted by noise in S-R channel; then the encoder, denoted by Enc-2, in the relay encodes the estimated message bits and resends them to the destination in relay-to-destination (R-D) channel. The coded sequence from the encoder in the relay is correlated with that from the encoder in the source, i.e. both parity-check bits from Enc-2 and Enc-1 are dependent on their common message bits. The two check sequences are completely correlated if the decoded bits by Dec-2 are error-free over S-R channel; otherwise, they are partially correlated if some errors occur in the decoded bits by Dec-2. The above two cases are called ideal and non-ideal coded relay cooperation, respectively.

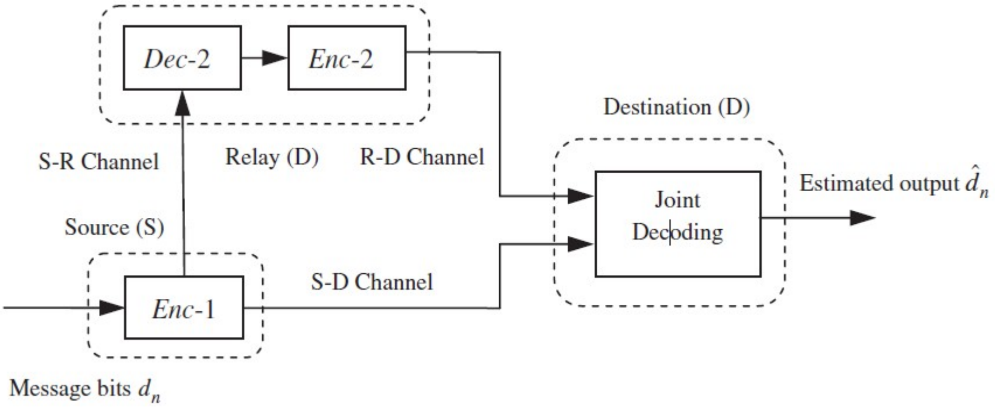


Figure 5.6 LDPC Coded relay cooperation and joint decoding

The paper [98] introduces a new iterative joint irregular systmatic Tanner graph for cooperative relay network. This is a three layer Tanner graph in which the 1st and 3rd layer check nodes share some of the common variable nodes and the remaining variable nodes are connected to the respective check nodes. This gives a single parity check matrix which is applied at the destination to jointly decode the information bits.

We consider a one relay cooperative communication systems[98,99] as shown in figure 5.7. The typical distance for the coded copperative system are mentioned in [99,100]. Here the distances between source , relay and destination are such that the distance between source to destination is normaliezed to 1.

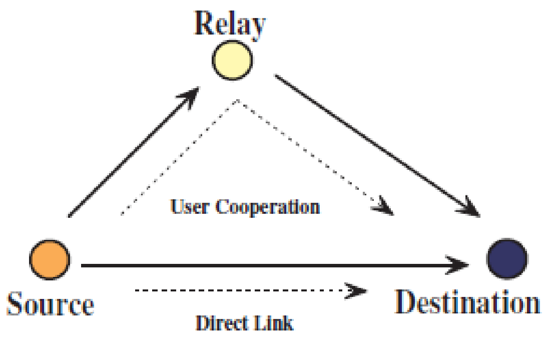


Figure 5.7 One relay Cooperative communication system

$$\gamma_1 = 1, \quad \gamma_0 = \frac{1}{d^\alpha} \quad , \text{ and } \gamma_2 = \frac{1}{(1-d)^\alpha} \qquad 5.2.1$$

Where α is the path loss and is usually taken in the range 2~3.The graph in the figure 6.8 is simulated for ideal and non-ideal cooperative communication. The distance between R-D in an non-ideal cooperation is such that it receives 4 db more power. The distance between R-D is such that it receives power 1 db more than S-D for both ideal and non-ideal situation. A parity check matrix for the source encoder is H_1 & H_2 is selected with 250 rows and 500 columns such that

$$H_S = \left[H^1_{250\times500} \quad I_{250\times500} \right] \& H_R = \left[H^2_{250\times500} \quad I_{250\times500} \right] \qquad 5.2.2$$

Such that $H_1 \neq H_2$ and H_2 is the row permutation matix obtained from H_1 and both are regular matrices. The number of ones in row and columns are equal in both the matrices $d_v = 4$ and $d_c = 8$. I is the identitity matrix , H_R is the irregular systematic pairty check matrix at the relay encoder , H_S is the irregular systematic pairty check matrix at thesource encoder. The final matrix at destination D is H_D and is given by

$$H_D = \begin{bmatrix} H_1 & I_{250\times250} & 0_{250\times250} \\ H_2 & 0_{250\times250} & I_{250\times250} \end{bmatrix} \qquad 5.2.3$$

A decode /re-encode/forward strategy has been adopted at the relay channel. At the relay a message is deocoded and then re-encode the parity only and then transmit the pairty bits to the destination where it is combined with the message from the sources such that the p_r is sent by relay R:

$$code = \left[s \quad p_s \quad p_r \right] \qquad 5.2.4$$

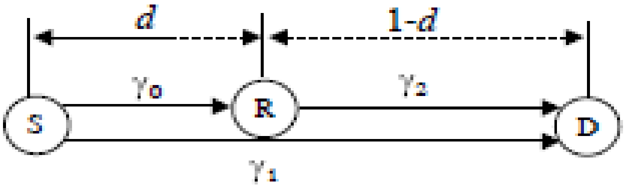


Figure 5.8 Typical way for showing the distances between sources(S), relay(R) and destination (D)

This code is decoded by parity check matrix in equation 5.2.3 by LDPC logrithmic sum product algorithm(SPA).The channels are simulated for the rayleigh fading coefficients such that

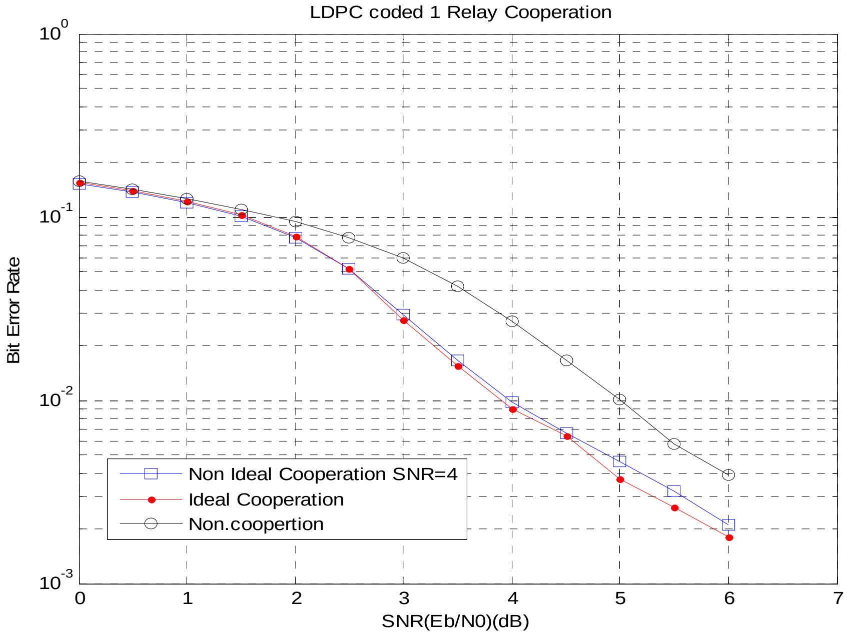


Figure 5.9 BER performance for one relay-cooperation and non-cooperation systems for code length=750, rate (at S and R)=2/3, iterations=10

$$y = hx + n \tag{5.2.5}$$

Where h is the rayleigh fading channel coefficient , n is the additive white guassian noise , x is the information bits. The simulation results shows the comparison for the coded cooperation with ideal , non-ideal and non-cooperative communication(deirect source to destination) under the same channel conditions. This same code has been simulated under the same channel conditions for layered min sum decoding algorithm which has fast convergence and better results as this is free of noise variance .So prior channel information is not required to initialize the information bits.The graph in figure 5.10 shows the performance comparison for this pratical type of LDPC deocder.

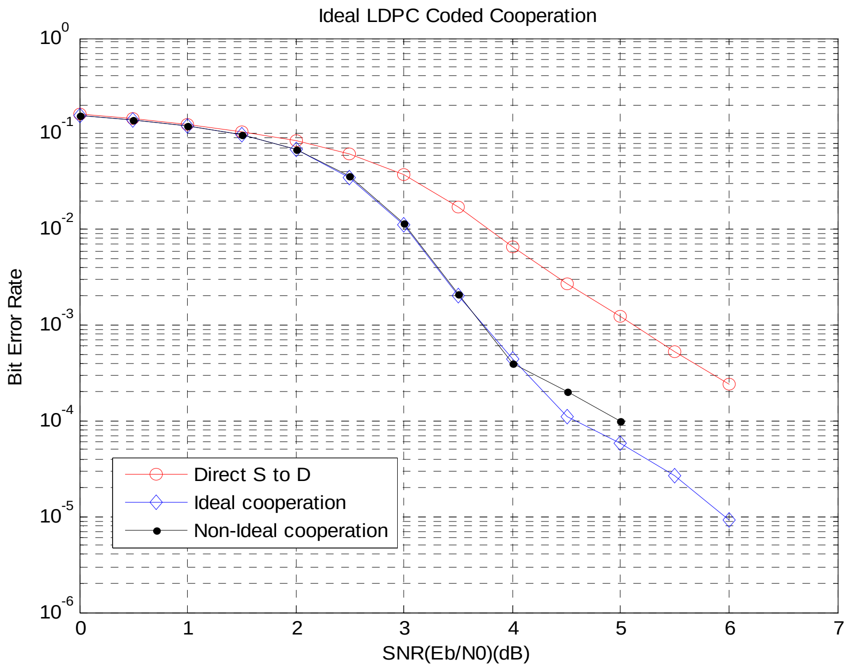


Figure 5.10 BER performance for one relay-cooperation and non-cooperation systems for code length=750, rate (at S and R)=2/3, iterations=10

5.3 Chapter Summary

This chapter gives the idea of using the LDPC codes in communication systems. We see that for MIMO as well as cooperative networks, LDPC gives a good performance. In cooperative communication, the joint Tanner graph for encoding irregular systematic LDPC codes and its joint decoding make it more suitable for future applications in such systems.

Chapter 6 Conclusion and Future Work

6.1 Main Work and Conclusion

In this thesis mainly the LDPC decoding algorithm are studied in theoretical and practical perspectives. A new improved min-sum LDPC algorithm has been proposed and has been simulated and applied to other communication systems. The work can be concluded as

1) Studying and analysis of the existing LDPC algorithms
2) Proposed a new min-sum algorithms which showed improved performance
3) LDPC decoder for hardware has been studied for good performance and high throughput
4) LDPC coded MIMO and single relay cooperative communication systems have also been simulated for performance analysis

6.2 Future Work

1) There is a vast scope for hardware implementation of better performance LDPC decoder and encoder
2) LDPC codes can be extended efficiently to use in application like inter-satellite communication, WiMax, MIMO cooperative coded communication and new application can be investigated.
3) Due to its better performance at low SNR, its applications can be extended to all communication systems where power is the main concerned.

References

[1] http://www.acorn.net.au/telecoms/coding/coding.cfm.

[2] C.E.Shannon, "A mathematical theory of communication." Bell System Technical Journal, 1948. vol. 27: p. 379-423 and 623-656.

[3] A.J.Viterbi, "Error Bounds For Convolutional Codes And An Asymptotically Optimum Decoding Algorithm." IEEE Trans. Inf. Theory, 260-269. vol. IT13: p. April 1967.

[4] C.Berrou, A.G., P.Thitimajshima, “Near Shannon limit error-correcting coding and decoding: Turbo-codes.” Int. Conf.Comm., Pro. 1993: p. 1064-1070.

[5] R.G.Gallager, "Low-Density Parity-Check Code. Cambridge." MA: MIT Press, 1963.

[6] D.J.Mackay, R.N., “Near Shannon Limit Performance of Low Density Parity Check Codes.” Electronic Letters, 1996. vol. 32, no.18: p. 1645-1646.

[7] D.J.MacKay, "Good error-correcting codes based on very sparse matrices." IEEE Trans. Infor. Theory, 1999. vol. 45: p. 399-431.

[8] T.J.Richardson, R.L.U., "The Capacity of Low-Density Parity-Check Codes Under Message-Passing Decoding." IEEE Trans.Information Theory, 2001. Vol.47: p. 599-618.

[9] M.G.Luby, M.M., M.A.Shokrollahi,D.A.Spielman, "Improved Low Density Parity Check Codes Using Irregular Graphs." IEEE Trans. Information Theory, 2001. vol.47: p. 585-598.

[10] S. Chung, G.D.F., J. J. Richardson, and R. Urbanke, “On the design of low-density parity-check codes within 0.0045db of the Shannon limit.” IEEE Communication Letters, 2001. vol. 5: p. 58-60.

[11] T.J.Richardson, A.S., R.Urbanke, "Design of capacity-approaching low-density parity-check codes." IEEE Trans. Inform. Theory, 2001. vol. 47.

[12] J. Chen, M.P.C.F., “Near optimum universal belief propagation based decoding of low-density parity check codes.” IEEE Transactions on Communication, 2002. vol.50: p. 406-414.

[13] Okamura, T.K., “Designing LDPC codes using cyclic shifts." Proceedings. IEEE International Symposium on Information Theory, 2003.

[14] Fossorier, M.P.C., “Quasi-Cyclic Low-Density Parity-Check Codes From Circulant Permutation Matrices." IEEE TRANSACTIONS ON INFORMATION THEORY, 2004. vol. 50, No.8.

[15] L. Peng, G.Z., X. Liu, “The Low-Density Parity-Check Codes Based on The n-Queen Problem.” Proceeding. ACM Conference. NRBC, New York, 2004: p. 37-41.

[16] Li Peng, G.Z., Xiaoxiao Wu, Xi Yan, “The Efficient D-LDPC Encoder based on arithmetical progression." Proceeding. IEEE GMC, Shanghai, 2007.

[17] X. Hu, E.E., D. M. Arnold, “Regular and irregular progressive edge-growth Tanner graph.” IEEE Trans. Inform. Theory, 2005. vol.51,No.1: p. 386-398.

[18] Wei Zhan, G.Z., Li Peng, Xi Yan, “Qausi-Cyclic LDPC Code based on D and Q matrices through progressive edge growth." International Symposium on Intelligent Signal Processing and Communication Systems. (ISPACS 2007). 2008.

[19] A. Yahya, O.S., M. F. M. Salleh,Farid Ghani, "A new Quasi-Cyclic low density parity check codes." IEEE Symposium on Industrial Electronics & Applications (ISIEA 2009), 2009. vol.1: p. 239 - 242.

[20] A. Yahya, O.S., M. F. M. Salleh, Farid Ghani, "Row Division Method to Generate QC-LDPC Codes." Fifth Advanced International Conference on Telecommunications. AICT '09., 2009: p. 183 - 187.

[21] Zongwang Li, L.C., Lingqi Zeng, Shu Lin,Wai H. Fong, "Efficient Encoding of QC LDPC code." IEEE Transactions on Communications, 2006. 54: p. 71-81.

[22] Zhang Wenjun, L.C., "A Novel Encoding Architecure of QC LDPC codes based on RAMs." Information Science and System Science:www.paper.edu.cn, 2007.

[23] D.J. Mackay, R.N., “Near Shannon Limit Performance of Low Density Parity Check Codes.” Electronic Letters, 1996. vol. 32, no.18: p. 1645-1646.

[24] Changzheng Ma, B.P.N., "LDPC DECODING ALGORITHM WITH ESTIMATION OF NOISE VARIANCE." 8th International Conference on Signal Processing, 2006. vol.3.

[25] Zongjie Tu, S.Z., "Overview of LDPC Codes." 7th IEEE International Conference on Computer and Information Technology(CIT 2007), 2007: p. 469 - 474.

[26] M.Fossorier, M.M., "Reduced Complexity Iterative Decoding of Low-Density Parity Check Codes Based on Belief Propagation. " IEEE Transactions on communications, 1999. vol.47.

[27] J. Chen, M.P.C.F., "Density evolution for two improved BP-Based decoding algorithms of LDPC codes." IEEE Communications Letters, 2002. vol.6: p. 208 - 210.

[28] LIU Hai-yang, Q.W.-z., LIU Bin, LI Jiang-peng, LUO Shi-dong, CHEN Jie, "Novel modified min-sum decoding algorithm for low-density parity-check codes." "The Journal

of China Universities of Posts and Telecommunications," 2010. vol.17: p. 1-5.

[29] Meng Xu, J.W., Meng Zhang, "A modified Offset Min-Sum decoding algorithm for LDPC codes." 3rd IEEE International Conference on Computer Science and Information Technology (ICCSIT2010). 2010. vol.3: p. 19 - 22.

[30] Xiaofu Wu, Y.S., Ming Jiang,Chunming Zhao, "Adaptive-Normalized/Offset Min-Sum Algorithm." IEEE Communications Letters, 2010. vol.14: p. 667 - 669.

[31] Xiaofu Wu, Y.S., Long Cui,Ming Jiang,Chunming Zhao, "Adaptive-normalized min-sum algorithm." 2nd International Conference on Future Computer and Communication (ICFCC2010), 2010. vol.2: p. 661 -663.

[32] C.Howland, A.B., “Parallel decoding architectures for low density parity check codes.” The 2001 IEEE International Symposium on Circuits and Systems - ISCAS 2001, 2001. vol.4: p. 742-745.

[33] M. Cocco, J.D., M. Heijligers, A. Hekstra, Jos Huisken, "A scalable architecture for LDPC decoding." Proceedings: Design, Automation and Test in Europe Conference and Exhibition, 2004. vol.3: p. 88 - 93.

[34] Zhongfeng Wang, Q.-w.J., "Low complexity, high speed decoder architecture for quasi-cyclic LDPC codes." Proc. ISCAS (6), 2005: p. 5786-5789.

[35] Nan Jiang, K.P., Zhixing Yang, “Flexible Low-Complexity Decoding Architecture for QC-LDPC Codes." 11th IEEE Singapore International Conference on Communication Systems.(ICCS 2008), 2008: p. 1316 - 1320.

[36] A J Blanksby, C.J.H., “A 690-mW 1-Gb/s 1024-b, rate 1/2 low-density parity-check code decoder.” JSSC, March, 2002. vol. 37: p. 404-412.

[37] A. Darabiha, A.C.C., F. R. Kschischang, “Power reduction techniques for LDPC decoders.” JSSC, Aug. 2008. vol.43: p. 1835-1845.

[38] T. Mohsenin, B.B., “High-throughput LDPC decoders using a multiple Split-Row method.” ICASSP, 2007. vol.2: p. 13-16.

[39] M.Cocco, J.D., M.Heijligers, A.Hekstra,J.Huisken, Silicon Hive, Eindhoven, “A scalable architecture for LDPC decoding.” proceeding:Design, Automation and Test in Europe Conference and Exhibition, 2004. vol.3: p. 88-93.

[40] M. Karkooti, J.R.C., "Semi-parallel reconfigurable architectures for real-time LDPC decoding." ITCC'2004, April 2004. vol.1: p. 579-585.

[41] Sangmin Kim, G.E.S., Hanho Lee, "A Reduced-Complexity Architecture for LDPC

Layered Decoding Schemes." IEEE Transaction on VLSI Systems, 2011. vol.19: p. 1099-1103.

[42] C. Roth, A.C., C. Studer, Y. Leblebiciz, A. Burgz, "Area, throughput, and energy-efficiency trade-offs in the VLSI implementation of LDPC decoders." 2011 IEEE International Symposium on Circuits and Systems (ISCAS), 2011: p. 1772 - 1775.

[43] Richardson, T., “Error floors of LDPC codes.” Proceeding of the 41st Allerton Conference., Oct.2003.

[44] Yang Han, W.E.R., "LDPC decoder strategies for achieving low error floors." Information Theory and Applications Workshop, 2008: p. 277 - 286.

[45] Kai He, J.S., Li Li,Zhongfeng Wang, "Low Power Decoder Design for QC LDPC Codes." Proceedings of 2010 IEEE International Symposinm on Circuirts and Systems (ISCAS), August, 2010: p. 3937-3940.

[46] T.Mohsenin, B.B., "Trends and challenges in LDPC hardware decoders." Asilomar Conference on Signals, Systems, and Computers, 2009: p. 1273-1277.

[47] A.Goldsmith, "Wireless Communications," 2005. Cambridge University Press.

[48] B.Sklar, "Digital Communication: Fundamentals and Applications ", 2nd Edition.

[49] J.R.Barry, E.A.L., D.G.Messserschmitt, "Digital Communication " 3rd Edition.

[50] T.S.Rappaport, "Wireless Communications: Principles and Practice," 2nd ed. Prentice Hall, Englewood Cliffs,NJ.

[51] C.E.Shannon., p. and J. 1949., " Communication in the presence of noise." Proceedings: IRE, 1949: p. 10-12.

[52] V.Krishnan, "PROBABILITY AND RANDOM PROCESSES," A John Wiley & Sons, Inc.

[53] B.Sklar, "Rayleigh Fading Channels in Mobile Digital Communication Systems, " Part II: Mitigation. IEEE Communications Magazine, 1997.

[54] B.Sklar, "Rayleigh Fading Channels in Mobile Digital Communication Systems," Part I: Characterization. IEEE Communications Magazine, 1997.

[55] M.K.Simon, M.S.A., "Digital Communication over Fading Channels: A Unified Approach to Performance Analysis," John Wiley & Sons, Inc.

[56] S.Faruque, "Orthogonal Coding and Iterative Decoding Improves Coding Gain." IEEE International Conference on Electro/Information Technology(EIT 2008), 2008: p. 276 - 279.

[57] M.Eroz, F.W.S., L.N.Lee, "DVB-S2 low density parity check codes with near Shannon limit performance." Int. J. Satell. Commun. Network, 2004. vol.22: p. 269-279.

[58] S.H.Gupta, B.V., "LDPC for Wi-Fi and WiMAX Technologies." 2009 International Conference on Emerging Trends in Electronic and Photonic Devices & Systems (ELECTRO-2009), 2010: p. 262 - 265.

[59] G.D.Forney, "Codes on Graphs: Normal Realizations." "IEEE Transactions on Information Theory," 2001. vol.47: p. 520 - 548.

[60] M.Tanner, R., "A recursive approach to low complexity codes.” IEEE Transactions on Information Theory, 1981. vol. IT-27: p. 533-547.

[61] F.R.Kschischang, B.J.F., H.A.Loeliger, "Factor graphs and the sum-product algorithm." IEEE Transactions on Information Theory. vol. 47: p. 498 - 519.

[62] D.J.C.MacKay, S.T.W., M.C.Davey, "Comparison of construction of irreugualr Gallager Codes." IEEE Trans. Communincation, 1999. vol.47: p. 1449-1454.

[63] http://www.inference.phy.cam.ac.uk/mackay/codes/.

[64] http://www.cs.utoronto.ca/~radford/ldpc.software.html.

[65] S.Chae, Y.P., "Low Complexity Encoding of Regular Low Density Parity Check Codes." IEEE 58th Vehicular Technology Conference(VTC 2003-Fall), 2003. vol: p. 1822-1826.

[66] T. J. Richardson, R.L.U., "Efficient Encoding of Low-Density Parity-Check Codes." IEEE TRANSACTIONS ON INFORMATION THEORY, 2001. vol. 47, No. 2.

[67] J.Chen, M.P.C.F., "Density evolution for two improved BP-Based decoding algorithms of LDPC codes." IEEE Communications Letters, 2002. vol.6: p. 208 - 210.

[68] J.Chen, A.D., E.Eleftheriou, M.Fossorier, X.–Y.Hu, "Reduced-Complexity Decoding of LDPC Codes." IEEE Transactions on Communications, 2005. vol. 53: p. 1288-1299.

[69] Waheed Ullah, J., Fengfan Yang, S.M.Aziz, "Two-Way Normalization of Min-Sum Decoding Algorithm for Medium and Short Length Low Density Parity Check Codes." 2011 7th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM), 2011: p. 1-5.

[70] Waheed ullah, J., Yang Fengfang, “Improved min-sum decoding algorithm for moderate length low density parity check codes." International conference on Computer, Informatics, Cybernetics and Application(ICIA-2011), LNEE Springer(ISSN: 1876-1100), 2011: p. 939-944.

[71] M.P.C.Fossorier, "Quasi-Cyclic Low-Density Parity-Check Codes From Circulant

Permutation Matrices." IEEE Transactions on Information theory, 2004. vol. 50,NO:8: p. 1788-1793.

[72] L.Chen, J.X., I.Djurdjevic, S.Lin, "Near Shannon Limit Quasi-Cyclic Low Density Parity- Check Codes." IEEE Trans. on Communications, 2004.

[73] S.Myung, K.Y., J.Kim, "Quasi-Cyclic LDPC Codes for Fast Encoding." IEEE transaction on Information Theory, 2005. vol. 51, NO.8: p. 2894-2901.

[74] M.Baldi, F.B., F.Chiaraluce, "On a Family of Circulant Matrices for Quasi-Cyclic Low-Density Generator Matrix Codes." IEEE Transactions on Information Theory, 2011. vol.57, issue:9: p. 6052-6067.

[75] Y.Kou, S.L., M.P.C Fossorier, " LDPC codes based on Finite Geomatry." IEEE Transaction on Information Theory, 2001. vol.47: p. 2711-2736.

[76] H.Tang, J.X., Y.Kou,S.Lin,K.A.Ghaffar, "Codes on Finite Geometries." IEEE transaction on Information Theory, 2005. vol.51: p. 572-596.

[77] H.Tang, J.X., Y.Kou,S.Lin,K.A.Ghaffar, "On algebraic construction of Gallager and circulant low-density parity-check codes." IEEE transaction on Information Theory, 2004. vol. 50,NO.6: p. 1269-1279.

[78] L. Lan, L.Z., Y.Y.Tai, L.Chen, S.Lin; K.Abdel-Ghaffar, "Construction of Quasi-Cyclic LDPC Codes for AWGN and Binary Erasure Channels: A Finite Field Approach." 2007. vol.53: p. 2429-2458.

[79] Nian-Liang, W., "The simple method of cyclic matrix Inversion." Journal of Shangluo Teacher College, 2002. vol. 16, NO.4.

[80] W.Zhan, G.Z., L.Peng, X.Yan, "Quasi-cyclic LDPC codes based on D and Q matrices through progressive edge growth." International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS 2007), 2007: p. 12-15.

[81] Z.Yang, Q.X., K.Peng, J.Fu, "A Fast and Efficient Encoding Structure for QC-LDPC Codes." 2008: p. 16-20.

[82] J.Zhang, M.F., "Shuffled iterative decoding." IEEE Transactions on Communications, 2005. vol.53: p. 209-213.

[83] M.M.Mansour, N.R.S., "High-Throughput LDPC Decoders." IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION SYSTEMS, 2003. Vol. 11, NO.6: p. 976-996.

[84] T.Mohsenin, B.B., “High-throughput LDPC decoders using a multiple Split-Row

method.” ICASSP, 2007. vol.2: p. 13-16.

[85] Zhiqiang Cui, Z.W., Youjian Liu, "High-Throughput Layered LDPC Decoding Architecture." IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2009. vol.17: p. 582 - 587.

[86] Kai Zhang, X.H., Zhongfeng Wang, "High-throughput layered decoder implementation for quasi-cyclic LDPC codes." IEEE Journal on Selected Areas in Communications, 2009. vol.17(6): p. 985 - 994.

[87] Jie Jin, C.-y.T., "An Energy Efficient Layered Decoding Architecture for LDPC Decoder." 2010. Vol.18, NO. 8: p. 1185-1195.

[88] Kai He, J.S., Li Li,Zhongfeng Wang, "Low Power Decoder Design for QC LDPC Codes." Proceedings of 2010 IEEE International Symposinm on Circuirts and Systems (ISCAS), 2010: p. 3937-3940.

[89] Nan Jiang, K.P., Jian Song,Chanyong Pan, Zhixing Yang, "High-Throughput QC-LDPC Decoders." IEEE TRANSACTIONS ON BROADCASTING, 2009. Vol.55, NO.2: p. 251-259.

[90] Hao Zhong, W.X., Ningde Xie,Tong Zhang, "Area-Efficient Min-Sum Decoder Design for High-Rate Quasi-Cyclic Low-Density Parity-Check Codes in Magnetic Recording." IEEE Transactions on Magnetics, 2007. vol.43: p. 4117 - 4122.

[91] M. Karkooti, J.R.C., "Semi-parallel reconfigurable architectures for real-time LDPC decoding." ITCC'2004, 2004. vol.1: p. 579-585.

[92] D.Oh, K.K.P., "Min-Sum Decoder Architectures With Reduced Word Length for LDPC Codes." IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS, 2010. vol.57,No.1: p. 105-115.

[93] Jianguang Zhao, F.Z., Amir H. Banihashemi, "On Implementation of Min-Sum Algorithm and Its Modifications for Decoding Low-Density Parity-Check (LDPC) Codes." IEEE Transations on COmmunications, 2005. vol. 53, No.4: p. 549-554.

[94] Cha-Hao Chung, Y.-L.U., Ming-Che Lu, Mao-Chao Lin, "Adaptive quantization for low-density-parity-check decoders." 2010 International Symposium on Information Theory and its Applications (ISITA), 2010: p. 13-18.

[95] Ben Lu, G.Y., Xiaodong Wang, "Performance Analysis and Design Optimization of LDPC-Coded MIMO OFDM Systems." IEEE TRANSACTIONS ON SIGNAL PROCESSING, 2004. 52: p. 148-361.

[96] http://www.dsplog.com/2008/10/24/mimo-zero-forcing/.

[97] Yao-Win Hong, W.-J.H., Fu-Hsuan Chiu, C.-C. Jay Kuo, "Cooperative Communications in Resource-Constrained Wireless Networks." IEEE Signal Processing Magazine, 2007. vol.24: p. 47-57.

[98] Fengfan Yang, J.C., Peng Zong, Shunwai Zhang,Qiuxia Zhang, "Joint iterative decoding for pragmatic irregular LDPC-coded multi-relay cooperations." International Journal of Electronics, 2011. vol.98,No.10: p. 1383-1397.

[99] Marjan Karkooti, J.R.C., "Cooperative Communications Using Scalable,Medium Block-length LDPC Codes." IEEE Wireless Communications and Networking Conference,(WCNC 2008), 2008: p. 88 - 93.

[100] M.A.Khojastepour, N.A., B.Aazhang, "Code design for the relay channel and factor graph decoding." Conference Record of the Thirty-Eighth Asilomar Conference on Signals, Systems and Computers., 2004: p. 2000 - 2004.

Acknowledgement

All praise and thanks are due to the Almighty Allah (God), the most gracious and merciful. There is a long list of people whom I would like to say thanks. Those are my family, my teachers, colleagues and friends who always there for me to help. I would like to say special thanks to:

- My supervisor Prof. Yang Fengfan for giving me the opportunity to work under his auspices, for his guidance, support and always encouragement.
- All my professors but especially to Prof. Zong Peng for his fruitful discussion, providing joint forum and recreation activities.
- Prof.Dr. B.T.Maharaj, Prof.Dr.S.M.Aziz and Dr.Abid Yahya for their valuable suggestions, reviewing and helping in my research papers.
- All my friends for helping me directly or indirectly in all matters. Especially Zhang, Jiangtao, Bruce (Wang), Jing Jing.
- My friends Ding, Zhang (William) and Hou Wen for bringing pleasant moments in life here.
- My friend Wu Xiao Hua for her help especially in my studies and research.
- All other Chinese friends not mentioned their names here.
- My foreign friends Darwin , Estebene , Giovany and Marx for nice time together
- My Pakistani friends in Nanjing : Ansar, Jawad , Badar ,Masood, Ali Tirrmizi ,Rizwan , Ghafoor Shah, Shehzad ,Ashhad ,Ahmad, Adeel, Bilal and others
- All my friends and colleagues in Pakistan especially Mr.Ayaz Ayub, Dr.Asif Hameed , Dr.Rizwan Asghar, Saqib Masood, Farooq Bhatti,Farooq Yaseen ,Cameron, Tahir , Nawaz ,Azeem , Mushtaq ,Azam, Dr.Saqib Ali and to all others not mentioned here.
- And to my family for their love and support, especially my mom to be very kind and loving.

Publications

1.) Waheed Ullah, Jiangtao, Fengfan Yang, S.M.Aziz, "Two-Way Normalization of Min-Sum Decoding Algorithm for Medium and Short Length Low Density Parity Check Codes." 2011 7th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM),2011: p. 1-5.

2.) Waheed Ullah, Jiangtao, Yang Fengfang, “Improved min-sum decoding algorithm for moderate length low density parity check codes." International conference on Computer, Informatics, Cybernetics and Application (ICIA-2011), LNEE Springer (ISSN: 1876-1100), 2011: p. 939-944.

3) Waheed Ullah ,Yang Fengfang , B.T. Maharaj ,Abid Yahya , “Performance of improved min-sum LDPC decoding algorithm for AWGN channel ,” Submitted to “ Journals of Academy Publisher”