Application of LDPC Codes to Networks

G.V.Joseph and M.A.Ambroze

Centre for Security, Communications and Network Research,
Plymouth University, Plymouth, UK
e-mail: info@cscan.org

Abstract

Internet is no longer just about emails and websites. It has become the critical medium of communication and entertainment and keeps on growing larger and larger day by day. Internet serves the society with the entire major needs like education, finance and business. The network performance has become a vital ingredient in businesses. Such rapidly growing internet make the network management increasingly important. The more and more emphasis is actually held on speed, connectivity and reliability. The dependence on the network connection by individuals or the enterprises will break down catastrophically if any problem occurs with the internet. The major problems with the internet these days are network efficiency and congestion. Because of such network congestion, the quality of service to the users gets deteriorated. Congestion leads to retransmissions which are considered as high overload. Such increase in transmission results in the packet loss or packet delay causing the applications to retransmit the data and thereby adding more and more traffic which results in further congestion. In order to reduce such retransmissions, the error correction codes are implemented. This research will investigate the ways to reduce the need for retransmission by using more efficient error correction codes. The most efficient form of error correction codes is the Low Density Parity-Check (LDPC) codes. This project involves the implementation of the LDPC codes which will be able to reconstruct a certain number of lost packets at the receiver end without the need to retransmit them again.

Keywords

Error correction codes, LDPC, Encoding, Decoding

1 Introduction

Computer networks are used for various purposes serving the companies and individuals. The digital means of communication has become an essential tool in a technological society. The network congestion is one of the major problems in data networking. The congestion affects the overall efficiency of the network. Because of the reduction in the network efficiency, there are variations in data rate and data delay that might alters the throughput of the network. Throughput is defined as the number of packets sent at a particular time. Reduction in throughput degrades the quality of service. Congestion occurs because of the increase in the number of network and associated devices and also the increase in the transmission rate which causes buffer overflow. This will lead to the loss in the transmitted packets making the application to retransmit the lost packets and adding further increase in congestion. Finally the congestion will get increased more and more and results in a very low level of throughput hence making the communication less useful. The

concept of the network coding was first introduced for satellite communication networks (Yeung *et al.* 1999). The first concept was a fundamental one and not thoroughly examined. After continuous investigations, the network coding concept was again fully developed (Ahlswede *et al.* 2000). The secondly introduced concept presented the advantages of the network coding method over the store-and-forward method. Later, the network coding found itself useful in many applications like information theory and coding, networking, wireless communication, etc. The concept of network coding is thoroughly investigated on both single information source and multiple information sources. Due to continuous development, many applications based on network coding have emerged. Thus, the network coding has placed them at a prominent position in the communication technology.

2 Literature Review

Low Density Parity Check (LDPC) codes are the set of linear block codes. Their name comes from the properties of the parity check matrix which contains only few numbers of ones when compared with the number of zeros. They are suitable for implementations that make heavy use of parallelism. The LDPC codes provide the channel-capacity performance on a large collection of data transmission and storage channels while simultaneously admitting implementable decoders. The LDPC codes have underwent rapid progress from the time they have been introduced. Such codes are now been used in many applications. The LDPC codes are applied in satellite-based Digital Video Broadcasting (DVB) and also in optical communication. They are highly adopted in IEEE wireless local area network standard. They are also in the consideration to be employed in third generation mobile telephony.

2.1 Background of LDPC codes

Low Density Parity Check (LDPC) codes were first proposed by Gallager in his PhD thesis in the year 1962. (Gallager, 1962 and 1963). Though the proposal was made, the code was scarcely used for 35 years that followed. The need for high complexity computation and introduction of the Reed-Solomon codes made the LDPC codes non-usable for such long gap. During that period, the concatenated codes were found appropriate for error control coding and hence this was also a reason. Also, the hardware of that time was not suitable to implement an effective decoder for LDPC codes. Hence because of these reasons, the forward error correction was dominated by the convolutional codes and structured block codes. Eventhough they dominated, their performance was well below the limit described by Shannon in his seminal paper (Shannon, 1948). Then the introduction of turbo codes made a revolutionary change in the coding theory which was found to be the best of all the error correction codes. The turbo codes were proposed by Berrou, Glavieux and Thitimajshima in 1993 (Berrou *et al.* 1993).

During the mid 90s, the research on LDPC codes commenced again by MacKay and Luby who introduced a new set of block codes which resembled the same features of the turbo codes (MacKay, 1999). Also, many new generalisations for LDPC codes were given by Richardson and Urbanke (Richardson and Urbanke, 2001). They introduced a set of irregular LDPC codes which outperformed the turbo codes. LDPC codes found to be more effective than the turbo codes. The decoder of the

LDPC codes used to declare if there is any decoding failure whereas decoder in turbo codes has to perform many computations to halt the decoding process. In LDPC codes, the shape of the parity check matrix specifies the creation of any rate and block length LDPC codes. The validity of the codeword is validated even when the error occurs. Moreover, the LDPC codes are not copyright protected and this made them more useful commercially.

LDPC codes are the set of linear block codes with the sparse parity check matrix. As the name suggests, it has very small number of non-zero elements in the parity check matrix. This guarantees both the decoding complexity and the minimum distance that linearly increases with the code length. Except the sparseness in the parity check matrix, there is no other difference between the LDPC codes and other block codes. Even all the other set of block codes can be represented as the LDPC codes by using the sparse parity check matrix. However finding the sparse parity check matrix for the present set of block codes is not so easy and it finds difficulties in practical cases. In LDPC codes, the generator matrix is determined only after constructing a sparse parity-check matrix. Gauss-Jordan elimination (Gaussian reduction) method is used to find the non-sparse generator matrix from the standard parity check matrix. Hence the encoding complexity can become quadratic in the code length. Using appropriate column permutations and back substitution methods, a linear-time encoding is processed. Encoding of LDPC codes and other classical block codes has some similarities. But the difference between them is how they are decoded. Classical block codes are decoded using the maximum likelihood decoding algorithms whereas the LDPC codes are decoded by the iterative algorithms using the graphical representation of the parity check matrix and thereby focussing more on the properties of the parity check matrix.

2.2 Construction of LDPC codes

There are different algorithms present for the construction of the LDPC codes. Those different algorithms are based upon different design approaches aiming different design criterion. It also depends upon the efficient encoding and decoding method. The most obvious method to construct the LDPC codes is through the construction of parity check matrix with a low density and with other suitable characteristics. The original LDPC codes were described by Gallager (Gallager, 1962). He used the regular LDPC codes and defined in H matrix form. In Gallager parity check matrix, the rows are divided into W_c sets with M/ W_c rows in each set. The first set of rows contains W_r consecutive ones ordering from left to right across the columns. All the other set of rows are formed by the column permutation of the first set. The Gallager codes were generalised by Tanner in 1981 (Tanner, 1981). That generalised LDPC codes were used for the study in CDMA (Code Division Multiple Access) communication channel. Gallager codes were extended by MacKay and others (MacKay, 1999).

Another form of constructing LDPC code was given by MacKay and Neal (MacKay and Neal, 2005). They suggested a way in which one column is added with another column positioning from left to right in the parity check matrix. So, the column weight can be chosen for reaching the right bit degree distribution. The location of ones in each column is chosen from the rows that are not full. If there are any

unfilled positions, the remaining columns are added. The row degree distribution may vary because of this process. Hence, by staring the process again, the correct row degree distributions are obtained. The only drawback in MacKay codes is that they do not have adequate structure to enable low-complexity encoding.

Richardson and Luby defined the ensembles of the irregular LDPC codes (Richardson *et al.* 2001) (Luby *et al.* 2001). Those codes are parameterised by the degree distribution polynomials. And also they explained how to optimise the polynomials for different communication channels. But the irregular codes do not essentially useful for efficient encoding. However Richardson and Urbanke proposed methods for achieving the linear-time encoding the codes.

Another form of LDPC codes called Repeat Accumulate (RA) codes has been proposed (Divsalar *et al.* 1998). This code has the characteristics of both the turbo codes and the LDPC codes. The RA codes have weight 2 columns in a step prototype for the last m columns of the parity check matrix. This form is of systematic block code and they are efficient and easily encoded. They are capable of operating at capacity limits, but they have low rate. The bits are repeated more than others yielding irregular repeat-accumulate (IRA) codes (Jin *et al.* 2000). The IRA encoder has a low density generator matrix, permuter and accumulator. The IRA codes are capable of operating close to the limits than the RA codes. The difference between them is IRA codes are non-systematic whereas the RA codes are systematic codes.

2.3 Representation of LDPC codes

There are two ways in which the LDPC codes can be represented. The two ways are matrix representation and graphical representation. The matrix representation is similar to the representation of other classical linear block codes.

2.3.1 Matrix Representation

The following is the example of a parity check matrix H represented by the matrix form.

$$\begin{bmatrix} 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \end{bmatrix}$$

This is a parity check matrix with dimensions n × m for a (8, 4) code. In the matrix W_r denotes the number of ones in each row and W_c denotes the number of ones in each columns. W_r denotes the number of ones in each row. W_c denotes the number of ones in each column. For the matrix to be of low density then the following two conditions must be satisfied: $W_c << n$ and $W_r << m$.

2.3.2 Graphical Representation

The second way of representation of LDPC codes is the graphical representation that was introduced by Tanner. They are the graphical depiction of the parity check matrix. This form not only provides the representation but also helps to describe the decoding algorithm. The tanner graph always contains set of nodes. The nodes of the graph are of two different types. They are variable nodes (v-nodes) or Bit nodes and Check nodes (c-nodes)

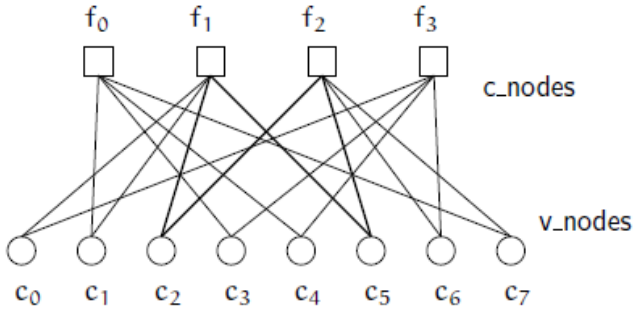

Figure 1: Tanner graph associated with the parity check matrix (Source: www.engr.uvic.ca/~masoudg/upload/ldpc-a%20brief%20tutorial.pdf)

3 Implementation

The following section will clearly explain the methods of encoding and decoding algorithms for the LDPC codes.

3.1 Encoding LDPC codes

The encoding of LDPC codes has two main functions involved. They are (a) construction of parity check matrix that is sparse and (b) generation of codeword with the matrix. Sparsity in the parity check matrix means each symbol node very few connections to the check nodes in the tanner graph. The parity check matrix always contains fewer number of 1's when compared to the number of 0's. In order to reduce the number of 1's in the parity check matrix; there are many algorithms and methods. One such method is gauss elimination that reduces the matrix H by employing elementary row and column operations. Often the generator matrix is not sparse only the parity check matrix is sparse which leads to the complexity in the encoding. The encoding efficiency is quadratic in block length. This is the only contrast with the turbo codes which has linear encode complexity. However it is possible to encode with minimum complexity by performing some process prior to encoding (Richardson and Urbanke, 2001). Some of the methods and algorithms for reducing the number of 1's in the parity check matrix are discussed below.

The LU factorisation was the first encoding method with the linear complexity introduced by Neal (Neal, 1999). This method was used in order to reduce the dense inverse operation that is involved in the encoding process. This method is applicable

for both the regular and irregular form of LDPC codes. In the systematic encoding process of the (n, k) LDPC encoder, the parity check matrix H can be divided into two sub groups. Let the two sub groups denoted by A and B. The matrix A is given by [(n-k) × (n-k)] and the matrix B is given by [(n-k) × k]. The codeword can also be split into systematic form that contains two categories of bits. The first k bits are the source message bits and they are denoted by s. The remaining (n, k) bits are the parity check bits and they are denoted by c. Hence the codeword is given by [s, c]. The encoding algorithm should satisfy the following condition, H × codeword ^{T} = 0. The LU factorisation is applicable for any kind of matrix as it finds solutions for all. This method is easy to program and it is very fast. The only disadvantage in this method is that it is difficult to find a good sparse LU decomposition for arbitrary H matrix (Su *et al.* 2005).

The Approximate Linear Triangulation (ALT) algorithm contains the parity check matrix of the LDPC code which is very sparse. But that sparsity is not present in the generator matrix of the LDPC code. The encoding of the LDPC codes is based upon the approximate lower triangular (ALT) form which is given by [L × N] where N - the block length of the code and L - the number of parity check equations. The complexity of encoding is very high because of the absence of sparsity in the generator matrix. Richardson and Urbanke developed an algorithm called RU algorithm that was widely used. The encoding is done with a specified parity check matrix H with a low triangular shape. The complexity in the algorithm is given by $O(n+g^2)$. The only disadvantage with the ALT algorithm is that there is no exact programmable step by step algorithm (Qi and Goertz, 2007)

The Greedy permutation algorithm is used to transform the parity check matrix into an approximate lower triangular form with minimum gap. The gap refers to the number of rows of the parity check matrix that cannot be brought into triangular form by row and column permutation. The complexity of the algorithm is given by $O(n^3)$. This algorithm involves mathematical calculations only and there is no exact programmable step by step algorithm. This method concentrates only with the reduction of the parity check matrix. The method provides the tradeoffs between the gap size and the performance for any given block length.

The Gaussian elimination is the most conventional method of encoding the LDPC codes. This method involves the systematic encoding with the generator matrix derived from the parity check matrix. The Gaussian elimination method is applicable for any type of block code and it does not deploy the sparseness of the LDPC codes. The complexity is $O(n^2)$ where n denotes the length of the codeword. This method is used to determine the generator matrix from the parity check matrix. The generator matrix is determined by performing the row permutations, modulo-2 operation on any two rows and some column permutation. The form of the generator matrix and the parity check matrix are given. Generator matrix is given by $G = [I_k \quad A^T]$. Parity check matrix by $H = [A \quad I_k]$. The parity check matrix is reduced into row echelon form by employing elementary row operations. The codeword after encoding is called the encoder codeword and it is denoted by c. The encoded codeword is obtained by multiplying the generator matrix with the information bits given by c = I × G. Thus the resultant codeword is the output of the encoder and with the help of the codeword, the information bits are received at the decoder.

3.2 Decoding LPDC codes

Many algorithms were developed for the decoding of LDPC codes. These algorithms were discovered independently several times. When Gallager introduced the LDPC codes in 1960s (Gallager, 1962), he also provided a decoding algorithm that is typically near optimal. The decoding algorithms iteratively compute the distribution of variables in graph-based models. Those algorithms were used to serve different purposes and hence they come under different names depending on the context. The commonly employed decoding algorithms are the message passing algorithm, belief propagation algorithm and the sum product algorithm. The term 'message passing' relatively represents all the iterative algorithms including the sum product and belief propagation algorithm and their approximations.

In order to explain the decoding algorithms, the simple variant which works on the platform must be explained. They are hard decision decoding and soft decision decoding. Hard decision decoding is easier to implement than the soft decision decoding. However, soft decision decoding offer better performance and decoding results when compared to the hard decision decoding.

When binary codes are used i.e. 0s and 1s, the digital modulator has only binary inputs. If digital demodulator output quantization is used, the decoder has only binary inputs. In this case, the demodulator is said to make hard decisions. Decoding based on hard decision made by the digital demodulator is called "hard decision decoding". If the output of digital demodulator consists of more than two quantization levels or without any quantization, the digital demodulator is said to make soft decisions. Decoding based on the soft decision made by digital demodulator is called soft-decision decoding. This method does not employ any flipping up of bits as in the hard decoder. The evidence that the checks provide about the bits are accumulated and the probabilities are propagated through the tanner graph. This method of decoding offers a means of bridging the performance gap between the system that uses hard-decision decoding and the system that uses maximum-likelihood decoding. The confidence information can then be used to improve the decoding process in such a way that the probability of decoding error and the decoding delay can be reduced. Hence the performance of the soft decision decoding is far better than the hard decision decoding.

3.3 Iterative Decoding algorithms

The set of decoding algorithms for decoding LDPC codes are collectively called as the message passing algorithms. Their operation is based upon passing the information along the edges of the tanner graph. These message passing algorithms are called as the iterative decoding algorithms. The messages are passed front and back between the variable nodes and the check nodes iteratively till a result is obtained. Consider the binary erasure channel where the transmitted bits are received correctly or received as erased with the erasure probability ε. In the erasure channel, the received bits are always correct and hence there is no need for the decoder to check the received bits. The main task of the decoder is to determine the value of the unknown bits. The parity-check equations are formed that includes only one erased bit, the correct value for the unknown (erased) bit can be determined by choosing the

value which satisfies the even parity. In this decoding method, the check node determines the value of an erased bit if it is the only erased bit in its parity-check equation that is framed already. The messages are passed along the edges of the tanner graph and the process is straightforward. Each bit node transmits the same outgoing message to each of its connected check nodes. The outgoing message is denoted by M. The message is declared as 1 or 0 or x if it is erased. If there is only one error x in the received message, the value of x can be calculated by choosing the suitable parity. The check node returns the message to the bit nodes. This message is labelled as $E_{j,i}$ where the j denotes the j^{th} check node and i denotes the i^{th} variable node. If the bit node of the erased bit receives 1 or 0 then the bit node changes the value to the incoming message. This process is repeated till all the erased bits are identified or the maximum number of iterations are performed.

The bit flipping algorithm is a message passing algorithm that is based upon the hard decision decoding of the LDPC codes. A hard decision is made on all the incoming bits and the result is passed to the decoder. The binary messages are passed onto the edges of the tanner graph. One of the bit nodes sends the message to check node containing the value one or zero. Each check node is connected to other bit nodes. The message received by the check node is forwarded to all the bit nodes that are directly connected to them. The check node determines the parity check equations. It also checks if the modulo-2 sum of the incoming message is zero. If messages received by a bit node are different from its received value; the bit node flips the current value. The process is repeated till all the parity check equations are satisfied.

The sum product algorithm is a type of message passing technique based upon the soft decision decoding. The sum product algorithm is similar to the bit flipping algorithm but the only difference is that in sum product algorithm, the message represents the probability. The bit flipping algorithm decoder accepts the initial hard decision on the received bits as input. But the sum product algorithm accepts the probability on the received bit as input. The incoming bit probabilities are called as the priori probabilities for the received bits. The input bit probabilities are known in advance before running the LDPC decoder. The bit probabilities that are returned by the decoder are called as posterior probabilities. The probabilities are expressed as log-likelihood ratios. The aim of this algorithm is to calculate the maximum a posterior probability (MAP) for each codeword.

4 Results and Discussions

The following section contains the explanation of the results obtained in the project. There are two phases in the project. They are encoding and decoding part. The encoding of LDPC codes involves the generation of the encoded codeword that is used to transmit the information from the sender to the receiver. The codeword is generated by appending the information bits with the generator matrix. Hence the codeword is used at the decoder as well in order to determine the exact information transmitted. The steps involved in the encoding are explained below.

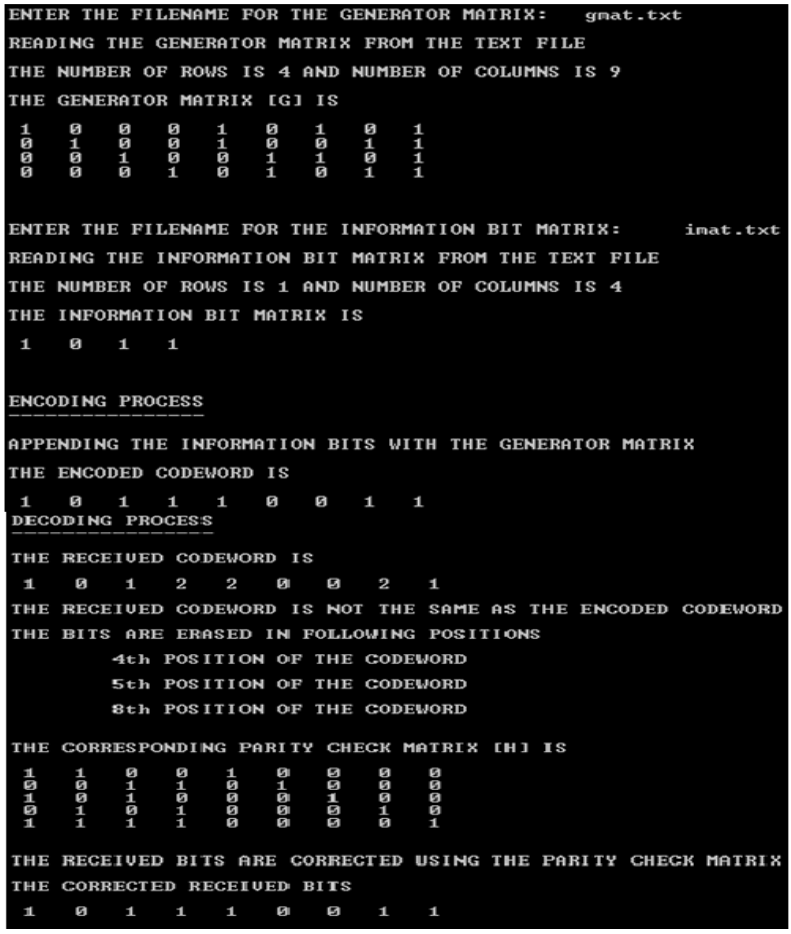

Figure 2: Output of the LDPC encoder and decoder

The generator matrix is given as the input. The generator matrix is already created by the user and stored in a text file. on. The name of the text file is *gmat.txt* and the matrix stored in the generator matrix is given below. The generator matrix is of the form G = [I_k P] where I_k denotes the identity matrix and P denotes the k × (n - k)

$$G = \begin{vmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \end{vmatrix}$$

The information bits are stored in the text file and it is also called through a function written in the program code. The name of the text file is *imat.txt* in which the information bits are stored. The contents of the text file are retrieved.

$$I = [\ 1\ 0\ 1\ 1\ ]$$

Using the generator matrix and the information bits, the codeword is generated that is used to transmit over the communication channel.

$$\text{Codeword, } c = I \times G = [1\ 0\ 1\ 1] \times \begin{vmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \end{vmatrix}$$

$$\text{Codeword} = [1\ 0\ 1\ 1\ 1\ 0\ 0\ 1\ 1]$$

Hence this is the encoded codeword generated from the information bits and the generator matrix. This codeword is passed to the receiver through the communication channel.

The decoding phase involves the restoration of the original message that is transmitted. The transmission is done via the binary erasure channel. Binary erasure channel has only two output probabilities. One probability is that the information received is correct and the other probability is that the information bit is erased. There is no chance of getting a bit as incorrect as this is not a characteristic of the binary erasure channel. In this example, such a transmission in the erasure communication channel has caused errors. This can be identified from the received codeword.

From the figure 4.1, it is noted that the received codeword is not the same as that of the encoded codeword before it enter the erasure channel.

$$\text{Received codeword} = [1\ 0\ 1\ x\ x\ 0\ 0\ x\ 1]$$
$$\text{Actual encoded codeword} = [1\ 0\ 1\ 1\ 1\ 0\ 0\ 1\ 1]$$

From the comparison of the encoded codeword and the received codeword, it can be inferred that there is error in the 4^{th}, 5^{th} and 8^{th} position of the codeword. The three codeword in the positions 4^{th}, 5^{th} and 8^{th} are erased due to the nature of the transmitting medium.

The erased bits have to be determined and done through the parity check matrix. The parity check matrix can be determined from the generator matrix. For the generator matrix that is used in the encoder phase, the corresponding parity check matrix is given below. Usually the parity check matrix is in the form of $H = [P^T\ I_r]$. P^T is the transpose of the parity matrix. The parity matrix is found from the generation matrix and the I_r is the identity matrix. Thus the parity check matrix can be easily constructed from the generator matrix and the vice-versa.

$$H = \begin{vmatrix} 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \end{vmatrix}$$

The decoding algorithm used in this project is based upon the simple algebra. The received codeword is determined as correct without any erasure if and only if it

satisfies the following condition, $c \times H^T = 0$. Hence if the product of the transpose of parity check matrix with the received codeword has no value, then the transmission is said to reliable. If not, there is some error in transmission which is nothing but the erasures.The transpose of the parity check matrix is given below

$$\begin{vmatrix} 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \end{vmatrix} = \begin{vmatrix} 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{vmatrix}$$

Thus the transpose of the parity check matrix is determined.

It is necessary to find out the erasures i.e. calculate and find the bit values in 4^{th}, 5^{th} and 8^{th} positions. The erased values can be found out with the help of the parity check matrix. The parity check matrix is compared with the received codeword which finally gives the erased bits. The method to find is given below.Compare the first row of the parity check matrix with the received codeword.

Received codeword ⟶ 1 0 1 x x 0 0 x 1
H matrix ⟶ 1 1 0 0 1 0 0 0 0

The bits in the received codeword that corresponds to the positions of the 1's in the H matrix are noted down. The bits are 1, 0 and x. The total of all the values must be 0 which notifies the condition. Hence, for the whole sum to be 0, the value of x should be 1. Finally the value of x is found as 1. The value of 1 must be written in the 5^{th} positions of the codeword. Hence the received codeword is given below.

Received codeword ⟶ 1 0 1 x (1) 0 0 x 1

Similarly the 4^{th} and 8^{th} positions of the codeword are determined using the next successive rows of the parity check matrix. Finally the 4^{th} and the 8^{th} position are found to be 1 and 1 respectively.

Received codeword ⟶ 1 0 1 (1) (1) 0 0 (1) 1

Now all the erased positions are determined using the parity check matrix. In order to determine the calculated erased bits are right, the decoding condition has to be satisfied. $c \times H^T = 0$

$$[\,1\ 0\ 1\ 1\ 1\ 0\ 0\ 1\ 1\,] \times \begin{vmatrix} 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{vmatrix} = 0$$

Hence the product of the corrected received codeword and the transpose of the parity check matrix is 0 and the decoding condition satisfies. Hence the corrected codeword bits are same as that of the encoded codeword. Thus the decoding process is executed.

5 Conclusion and Future Works

During data transmission there are various factors that affect the transmission of the data which in turn affects the performance of the network. This project involves the role of error correction codes in the context of digital communication system. The encoder and decoder are the important blocks of any communication system. The data is encoded before transmission through the channel and it is decoded when it passes out from the channel. Error correction code is implemented in the encoders and decoders in order to maintain the reliability of the transmission. The most efficient one is the LDPC code which is investigated in the project. During encoding, the information bit is encoded with the generator matrix of the LDPC code and finally the encoded codeword is generated. The encoded codeword is transmitted through the binary erasure channel which contains the information bits. Due to the nature of the channel, some bits are erased during the transmission. But the decoder uses the parity check matrix which is closely associated with the generator matrix and compares it with the received codeword. Finally, the correct codeword is determined by the decoder.This project can be implemented in the network layer or transport layer or the application layer of the OSI model. By implementing so, the error correction codes especially LDPC codes reduce the need of retransmitting the lost packets and will be able to reconstruct the certain number of lost packets at the receiver end. From this point of view, the error correction codes (LDPC codes) play a key role in a digital communication system as well as in the data transmission.

Low density-parity-check codes have been studied a lot in the last years and huge progresses have been made in the understanding and ability to design iterative coding systems. The performance in the LDPC codes is better than the turbo codes. The LDPC codes make the possibility to implement the parallelizable decoders. There is a drastic increase in the data transmission technology over the past few years. Many new techniques are provided to shape the data traffic in order to maximise the efficiency of the bandwidth reservation scheme whilst guaranteeing a defined quality of service in terms of data loss and delay. These new techniques come with the challenges of processing more and more bits which requires the powerful code design implementation. More efficient classes of codes that suit the developing data transmission field must be developed and their performance has to be examined. This

project involves the LDPC codes which are found to be the most efficient form of error correction codes. The encoding and the decoding algorithm used here are the conventional way of implementation in the communication system. Because of the improvements if communication field, the encoder has to be designed with more linear complexity. Also, the decoding method should use the iterative algorithms and the performance has to be evaluated. The most important consideration in the next generation communication system is to develop the LDPC decoder that enables a close integration between the codes and the hardware architecture designs. The FPGA result shows promise for future ASIC implementation for the use in next generation communication systems. Another consideration is that it is necessary to design new LDPC codes which will not only provide near-capacity performance and also will have efficient structure for low power implementations. If the internet is used for simulation purposes, then it implemented in network/transport level and application level interfaces.

6 References

Ahlswede, R., Cai, N., Li, S.Y.R., and Yeung, R.W. (2000) "*Network information flow*," IEEE Transactions on Information Theory, Vol. IT-46, pp. 1204–1216

Berrou, C., Glavieux, A and Thitimajshima, (1993) P. "*Near Shannon Limit! 3ror-Correcthg Coding and Decoding: Turbo Codes*," in Proc. 1993 IEEE International Conference on Communications, Geneva, Switzerland, pp. 1064-1070

Divsalar, D., Jin, H. And McEliece, R. (1998) Proc. 36th Annual Allerton Conference on Communication, Control and Computing, "*Coding Theorems for Turbo-like codes*", pp.201-210

Gallager, R.G. (1962) "*Low-Density Parity-Check Codes*", IRE Transactions on Information Theory, vol. IT-8, pp. 21-28

Gallager, R.G. (1963) "*Low-Density Parity-Check Codes*", Cambridge, MA: M.I.T. Press

Jin, H., Khandekar, A. and McEliece, R. (2000) "*Irregular repeat-accumulate codes*" Proc. 2nd. International Symposium on Turbo Codes and Related Topics, France, pp. 1-8s

"LDPC codes – a brief tutorial" [Online] Available at:www.engr.uvic.ca/~masoudg/upload/ldpc-a%20brief%20tutorial.pdf (Accessed on 27/12/2009)

Luby, M., Mitzenmacher, M. Shokrollahi, M. and Spielman, D. (2001) "*Improved low-density parity check codes using irregular graphs*", IEEE Transactions on Information Theory, pp. 585-598

MacKay, D.J. (1999) "*Good Error-Correcting Codes Based on Very Sparse Matrices*," IEEE Trans. Znfo. Theory, vol. 45, no. 2, pp. 399-43

MacKay, D. and Neal, R. (2005) "*Good codes based on very sparse matrices*", in Cryptography and Coding, 5th IMA Conference, C.Boyd, Ed., Lecture Notes in Computer Science, Germany, pp. 100-111

Neal, R.M. (1999) "*Sparse matrix methods and probabilistic algorithm*", IMA Program On Codes, Systems, and Graphic Models, 1999

Qi, H. and Goertz, N. (2007) "*Low-Complexity Encoding of LDPC Codes: A New Algorithm and its Performance*", Joint Research Institute for Signal & Image Processing, School of Engineering and Electronics, The University of Edinburgh, UK

Richardson, T.J and Urbanke, R.L. (2001) "*Efficient encoding of low-density parity-check codes*," IEEE Transactions on Information Theory, vol. 47, no. 2, pp. 638456

Richardson, T., Shokrollahi, A. and Urbanke, R. (2001), *"Design of capacity-approaching irregular low-density parity-check codes"*, IEEE Transactions on Information Theory, vol. 47, pp. 619-637

Shannon, C. (1948) "*A Mathematical Theory of Communication*," Bell Syst. Tech. Journal, vol. 27, pp. 623-656

Su, J., Liu, Z., Liu, K., Shen, B. and Min, H. (2005) "An efficient low complexity LDPC encoder based on LU factorization with pivoting", 6th International conference on ASICON, Shanghai, vol. 1, pp. 107-110

Yeung, R.W., Zhang, Z. (1999) "*Distributed source coding for satellite communications*," IEEE Transactions on Information Theory, Vol. IT-45, pp. 1111–1120