Channel Coding using Low Density Parity Check Codes in AWGN

Rinu Jose
PG Scholar
Dept of ECESt.
Joseph's College of Engineering
and Technology
Palai, Kerala, India

Ameenudeen P.E
Asst.Professor
Dept of ECESt.
Joseph's College of Engineering
and Technology
Palai, Kerala, India

ABSTRACT
Forward error correction codes (FEC) are used for error detection and correction in communication systems. Low density parity check code (LDPC) is used as a powerful Forward Error Correction code in long distance communication systems which works close to the Shannon limit. Unlike other conventional channel code, the decoding algorithm used for LDPC codes is an iterative message passing algorithm (MPA). They are soft decision and hard decision decoding algorithms. This paper aims at a comparative study between a hard decision algorithm (bit flipping) and a soft decision algorithm (belief propagation). The analysis is based on the Bit Error Rate of decoding outputs. The result shows that the Soft decision decoding gives better performance than the hard decision decoding. LDPC code with soft decision decoding enhances the system performance and makes the long distance communication fast and error free.

Keywords
LDPC, FEC, LBC, iterative decoding, BER, Sum product decoding, Message passing algorithm.

1. INTRODUCTION
A communication is said to be perfect if and only if the transmitted data is received at the receiver end without any error. When the data is transmitted over a communication channel there is transmitted over a communication channel there is a chance to occur an error due to the presence of noise in the channel. The main challenge in the communication system is to achieve a reliable and an error free transmission of data from information source to destination. To increase the reliability of data transmission two basic error control strategies are used. They are Automatic Repeat Request (ARQ) and Forward error correction (FEC) [1]. In ARQ the receiver requests retransmission of unreliable data frames. If an error is detected in a frame that frame can be declared as unreliable. FEC is usually preferred over ARQ schemes since ARQ schemes results in unnecessary wastage of the channel band width due to retransmission. FEC scheme can be used for both error detection and correction. The whole idea of error correction using FEC is the addition of some redundant bits to the message bits of the sender, called as the parity bits. The use of an FEC code in communication systems eliminates theneed of a feedback channel since the retransmission of data can be avoided.

The concept of channel coding was introduced by Claude Shannon in his seminal paper in 1948 [2]. In his work he introduced a new channel parameter called as channel capacity. Shannon theorem showed that "In a noisy channel with capacity C an information is transmitted at a rate R then, if R is less than C there exist a coding technique which allows the probability of error at the receiver to be made arbitrarily small" [2]. This means that it is possible to transmit information with arbitrarily zero probability of error up to nearly a limit of C bits per second, known as the Shannon limit[2].

The Shannon theorem gave rise to the introduction of many error correcting codes called as the channel codes. Mainly there are two types of channel codes, Linear block codes and Convolutional codes. An (n, k) linear block codes accept a block of k information bits called as message and return a block of n coded bits called as the code word. LBCs are used primarily to correct or detect the errors in data transmission. Commonly used block codes are Hamming codes, RS codes, BCH codes, Turbo codes and LDPC codes [3]. The convolutional codes are used primarily for real time error correction and can convert an entire data stream in to one single code word.

Although an LDPC code is a block code it gives an exceptional BER performance nearly close to Shannon limit when compared to other block codes. This is due to the message passing decoding algorithm used in the LDPC decoder. LDPC codes also provide high code gain at low BER. Turbo codes also work close to Shannon limit. But decoding complexity is high in turbo codes than LDPC [4]. And due to this inherent advantage LDPC codes have replaced turbo codes as the error correcting codes in many applicationslike DVB-S2 video standard for the satellite transmission of digital televisions [5]. LDPC is applied to wireless, wired and optical communication system [6] and storage application such as magnetic discs and compact discs. Less computational complexity is the main attraction of the LDPC codes.

There exist mainly two different decoding algorithms for LDPC codes- hard decision decoding and soft decision decoding. In hard decision decoding the message passed contains the actual value of bits. The soft decision decoding is a probabilistic decoding algorithm in which the message passed is the probability value associated with the occurrence of a particular bit [7]. This work aims at the comparison of the decoding performance of the two algorithms in AWGN channel.

The rest of the paper is organized as follows: Section II describes the idea of low density parity check codes. The encoding of LDPC is explained in Section III. Section IV illustrates both hard decision and soft decision decoding algorithm. The system model is explained in section. Section VI discusses the simulation results.

2. LOW DENSITY PARITY CHECK CODES

LDPC is a linear block code in which the parity check matrix has sparse property. The number of 1s in the H matrix is very less compared to number of zeros. LDPC codes were first introduced by Gallager in 1962 [3]. But at that time because of the computational complexity it was largly neglected. In the meantime the field of forward error correction was dominated by highly structured algebraic block and convolutional codes. After the discovery of Turbo codes the LDPC codes were eventually revisited by MacKay, Neal, Sipser and Spielman and Richardson and Urbanke [8]. MacKay and Neal verified the performance of LDPC close to shannon limit[9]. Sipser and Spielman proved that with N tends to ∞ linear decoding complexity was sufficient to decode capacity approaching codes[10]. The mathematical tool to estimate the performance of codes and to build capacity reaching codes was developed by Richardson and Urbanke [11]. The rediscovery of the LDPC gives a drastic change in error correction coding field. LDPC codes are block codes with parity-check matrices thatcontain only a very small number of non-zero entries [8]. It is the sparseness[10] of H which guarantees both a decoding complexity which increases only linearly with the code length and a minimum distance which also increases linearly with thecode length. LDPC codes are represented in two ways. One is matrix form of its H matrix and second is graphical form. The graphical representation of LDPC codes are known as the Tanner graph, which was introduced by Tanner [12]. Tanner graph is a bipartite graph, which means the graph is separated into two partitions. These partitions are called by different names: sub code nodes and digit nodes, variable nodes and check nodes, message nodes and check nodes[8]. The matrix representation and corresponding tanner graph of a sample parity check matrix is shown below.

Here C1, C2, .. C8 are the variable nodes and f1, f2, f3, f4 are the check nodes. Tanner is known as the originator of the codes based on graphs. The tanner graph [12] have an important role in the development of decoding algorithm of LDPC. In decoding of LDPC codes, iterative probabilistic decoding algorithms are widely used. McEliece have shown that these decoding techniques can be derived from Pearls belief propagation algorithm, or message passing algorithm [10].

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

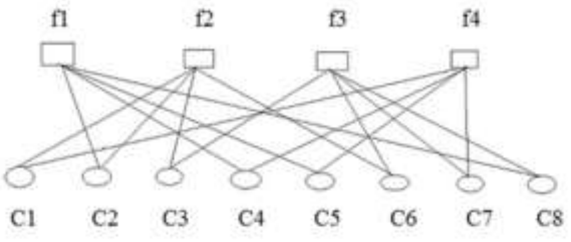

Fig 1: Tanner Graph

The connection between the message node and check node is known as edge. The number of edges in the tanner graph equal to number of ones in parity check matrix. A code is said to be systematic, if and only if the message bit node can be distinguishable from the parity bit nodes by placing them on separate side of the graph. A sequence of connected vertices which start and end at the same vertex in the graph and which contain other vertices no more than once is known as a cycle in a tanner graph. The number of the edges in a tanner graph gives the length of the cycle. Size of the smallest cycle is known as girth of a graph.

3. LDPC ENCODING

Let u be a message block, G is generator matrix, H is parity check matrix. parity-check matrix H can be found by performing Gauss-Jordan elimination on H to obtain it in the form

$$H = [A, I_{n-k}] \quad (1)$$

where A is a (n-k)Xk binary matrix and I_{n-k} is the size (n - k) identity matrix. The generator matrix is then

$$G = [I_k,\ A^k] \quad (2)$$

By applying proper elementary operation and convert H matrix into row reduce echelon from. This gives the sparse property for H matrix. Sparse property means number of 1s is less than the number of 0s. Sparse property make the LDPC less complex. Code word is generated by modulo 2 addition of message bits u with generator matrix G.

$$C = uG \quad (3)$$

C is code word , u is input message bits and G is generatormatrix.

4. LDPC DECODING

LDPC code decoding is performed through iterative processing based on the Tanner graph, to satisfy the parity check conditions. The condition that $CH^T = 0$ is known as parity check condition. If $CH^T = 0$ then the received code word is said to be valid, that is the received code word is similar to the transmitted code word. Iterative decoding has two variations namely hard decision and soft decision decoding algorithms. The decision made by the decoder based on the received information is called a hard-decision if the value of a single bit can either be 0 or 1. Example for hard decision decoding in LDPC is Bit Flipping Algorithm. If the decoder is able to distinguish between a set of quantized values between 0 and 1, then it is called a soft-decision decoder. These values give the probability of a particular bit in a node. The sum product algorithm is a soft decision message-passing algorithm.

A Message Passing Algorithm (MPA)[7] based on Pearls belief algorithm describes the iterative decoding steps. The reason for the name message passing algorithm is that at each round of the algorithms messages are passed from message nodes to check nodes, and from check nodes back to messagenodes in the tanner graph. Different message-passing algorithms are named for the type of messages passed or for the type of operation performed at the nodes. In some algorithms, such as bit-flipping decoding, the messages are binary and in others, such as belief propagation decoding, the messages are probabilities which represent a level of belief about the value of the code word bits [13]. It is often convenient to represent probability values as log likelihood ratios, and when this is done belief propagation decoding is often called sum-product decoding since the use of log likelihood ratios allows the calculations at the bit and check nodes to be computed using sum and product operations [14].

4.1 Hard Decision Decoding

Bit flipping algorithm is the best example for hard decisiondecoding. In the bit-flipping algorithm the messages are passed along the Tanner graph edges. A message node sends a message declaring if it is a one or a zero, and then each check node sends a message to each connected message node by finally declaring that what value the bit is based on the information available to the check node [12]. The algorithm is explained on the basis of an example code word C = $[11001000]^T$. Suppose that the received cord word is Y = $[10001000]^T$. So C_2 was flipped. The figure1 shows the tanner graph which is used for decoding algorithm. The steps involved in the bit flipping algorithm is given below.

TABLE 1.Overview of messages received and sent by the check nodes

Check Nodes	Activities				
f_1	Receive	$C_2 \rightarrow 0$	$C_4 \rightarrow 0$	$C_5 \rightarrow 1$	$C_8 \rightarrow 0$
	Send	$1 \rightarrow C_2$	$1 \rightarrow C_4$	$0 \rightarrow C_5$	$1 \rightarrow C_8$
f_2	Receive	$C_1 \rightarrow 1$	$C_2 \rightarrow 0$	$C_3 \rightarrow 0$	$C_6 \rightarrow 0$
	Send	$0 \rightarrow C_1$	$1 \rightarrow C_2$	$1 \rightarrow C_3$	$1 \rightarrow C_6$
f_3	Receive	$C_3 \rightarrow 0$	$C_6 \rightarrow 0$	$C_7 \rightarrow 0$	$C_8 \rightarrow 0$
	Send	$0 \rightarrow C_3$	$0 \rightarrow C_6$	$0 \rightarrow C_7$	$0 \rightarrow C_8$
f_4	Receive	$C_1 \rightarrow 1$	$C_4 \rightarrow 0$	$C_5 \rightarrow 1$	$C_7 \rightarrow 0$
	Send	$1 \rightarrow C_1$	$0 \rightarrow C_4$	$1 \rightarrow C_5$	$0 \rightarrow C_7$

1) Step 1: All message nodes send a message to their connected check nodes. In this case, the message is the bit they believe to be correct for them. Here C2 receives 0, so C2 send 0 to f1 and f2. Table1 shows the overview over messages received and sent by the check nodes.

Table 2.Message nodes decisions for hard decision decoder

Message Nodes	y_i	Message from check node		Decision
C_1	1	f_2 --> 0	$f_4 \rightarrow 1$	1
C_2	0	$f_2 \rightarrow 1$	f_1 --> 1	1
C_3	0	$f_2 \rightarrow 1$	f_3 --> 0	0
C_4	0	$f_1 \rightarrow 1$	$f_4 \rightarrow 0$	0
C_5	1	$f_1 \rightarrow 0$	$f_4 \rightarrow 1$	1
C_6	0	$f_2 \rightarrow 1$	$f_3 \rightarrow 0$	0
C_7	0	$f_3 \rightarrow 0$	$f_4 \rightarrow 0$	0
C_8	0	$f_1 \rightarrow 1$	$f_3 \rightarrow 0$	0

Step 2: Every check nodes calculate a response to their connected message nodes using the messages they receive from step 1. This response is calculated using the parity-check equations which force all message nodes to connect to a particular check node to sum to 0 (mod 2). If sum is equal to zero then check node send the same bit which they received from the message node. If sum is not equal to zero then the check node flip the bit which they received from the message node and resend it to the message node. Moves to step 3.

Step 3: In this step, the message nodes use the messages they get from the check nodes to decide if the bit at their position is a 0 or a 1 by majority rule. The message nodes then send this hard-decision to their connected check nodes. Table2 illustrate this step.

Step 4: Repeat step 2 until either exit at step 2 or a assigned number of iterations has been passed.

4.2 Soft Decision Decoding

Soft-decision decoding is based on the idea of belief propagation. The sum-product algorithm is a soft decision messagepassing algorithm. In the case of sum product decoding these probabilities are expressed as log-likelihood ratios. Log likelihood ratios (LLR) are used to represent the matrix for a binary variable by a single value. The sum product algorithm is a soft decision message-passing algorithm [15] which is similar to the bit-flipping algorithm. The main difference between SPA and bit flipping algorithm is that each decision is represented with probabilities of the information bits. Important terms in the algorithm are

- $P_i = P_r(C_i = 1\backslash Y_i)$
- q_{ji} is a message sent by the message node C_i to the check node f_j . Every message contains always the pair $q_{ij}(0)$ and $q_{ij}(1)$ which stands for the amount of belief that Y_i is a 0 or a 1.
- R_{ji} is a message sent by the check node f_j to the variable node C_i. Again there is a $r_{ji}(0)$ and $r_{ji}(1)$ that indicates the (current) amount of belief that Yi is a 0 or a 1.

The steps involved in sum product algorithm.

1) Step 1: All message nodes send their q_{ij} messages. $q_{ij}(1) = P_i$ and $q_{ij}(0) = 1 – P_i$
2) Step 2: The check nodes calculate their response messages r_{ji}, Using the formula shown below [10]

$$r_{ji}(0) = \frac{1}{2} + \frac{1}{2}\prod_{(i \in v_j/i)} 1 - 2q_{ij}(1) \quad (4)$$

and

$$r_{ji}(1) = 1 - r_{ji}(0) \quad (5)$$

3) Step 3: The message nodes update their response message to the check nodes using

$$q_{ji}(0) = k_{ij}(1 - p_i)\prod_{(j' \in c_i/j)} (r_{j'i} \ (0)\ ) \quad (6)$$

and

$$q_{ji}(1) = k_{ij}(p_i)\prod_{(j' \in c_i/j)} (r_{j'i} \ (1)\ ) \quad (7)$$

Constant k_{ij} are chosen in such a way to ensure that

$$q_{ij}(0) + q_{ij}(1) = 1 \tag{8}$$

At this point the message nodes also update their current estimation C_i' of their message C_i. This is done by calculating the probabilities for 0 and 1 and voting for the bigger one.

$$Q_i(0) = k_i\ (1 - p_i) \prod_{(j \in C_i)} (r_{ji}\ (0)) \tag{9}$$

and

$$Q_i(1) = k_i\ (p_i) \prod_{(j \in C_i)} (r_{ji}\ (1)) \tag{10}$$

4) Step 4: Ci is 1 if Qi(1) > Qi(0). Otherwise Ci is zero.
5) Step 5: Go to step 2.

SPA algorithm provides low complexity in log domain thanin the probability domain. Using log ratio turns multiplicationinto additions. This makes algorithm simpler [14].

5. SYSTEM MODEL

Fig.2 represents the system model of a basic digital communication system which uses LDPC code as the channel code and BPSK modulation. u is the block of message bits which input to the LDPC encoder and v is the output from the LDPC encoder. After BPSK modulation the codeword C is obtained. This codeword C is transmitted to receiver through communication channel. Channel is modeled as AWGN with noise power density No/2. The transmitted data is passed through the noisy AWGN channel. The noise gets added up with the signal. This effect may cause error in the transmitted data. For the simplicity the model does not account for fading, frequency selectivity, interference, nonlinearity or dispersion. The received codeword r is fed in to a BPSK demodulator to reverse the modulation performed at the transmission section. Then the demodulated codeword y is passed through the LDPC decoder and the estimate of original message u' is recovered at the LDPC decoder.

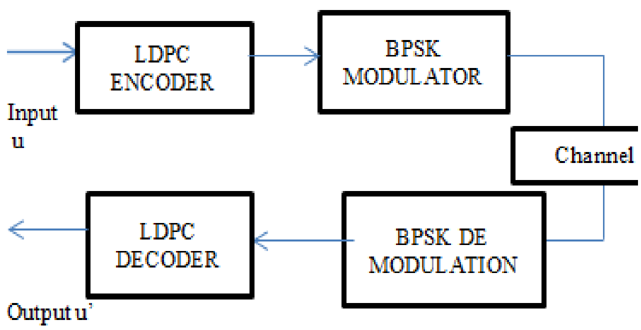

Fig 2: System Model

If the system uses soft decision decoding then the received codeword is first given to the LDPC decoder and then to the BPSK demodulator, ie BPSK demodulator is placed after the LDPC decoder. In this paper work the BER performance of both hard decision and soft decision decoding algorithms are analyzed

6. SIMULATION RESULTS

The simulation results and comparisons of the proposed system were executed and analyzed using MATLAB. In this work, number of codewords, different code rates, number of iterations and decoding schemes are the parameters used for analysis of LDPC codes. The iterations specify the number of times the information is evaluated before making a final decision about the received codeword.

6.1 Analytical study of LDPC with different code rates

In this section, the performance of LDPC code with different code rates isanalysed. Three different code rates are used. The parameters used for the simulation are given in the table shown below..

Table 3.Parameters used

No. of code words	100000
SNR values	0 to 10
Code rates used	1/2 , 1/3, 1/4
No. of iterations	10

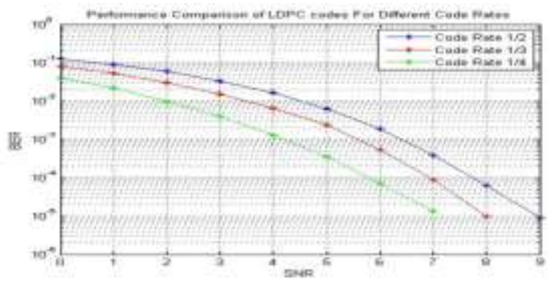

Fig 3: Performance analysis of LDPC codes for different code rates

The ratio between number of message bits and coded bits are known as the code rate. The Code rate can be denoted as k/n. The figure 3 shows the performance analysis of LDPC code for different code rates.From the simulation we can observe that, low code rate gives better BER performance. Which is expected since number of parity bits is increasing. Here at 1/4 code rate LDPC has good BER performance than other code rates.

6.2 Comparison between decoding algorithms

LDPC codes are decoded by using two different decoding algorithms. They are soft decision decoding and hard decision decoding algorithms. Here 100000 code words are coded by using soft decision and hard decision decoding algorithms. These two algorithms are based on iterative decoding technique. In this simulation 50 iterations are used. Table 4 shows the performance analysis of hard decision and soft decision decoding. Number of error decreased with increase in SNR.

Table .4.Performance analysis of hard decision and soft decision decoder

SNR Values	No. of Errors	
	Hard Decision	Soft Decision
0	678778	169167
2	565099	59731
4	385341	9607
6	186396	468
8	53599	7
10	7126	0
12	282	0

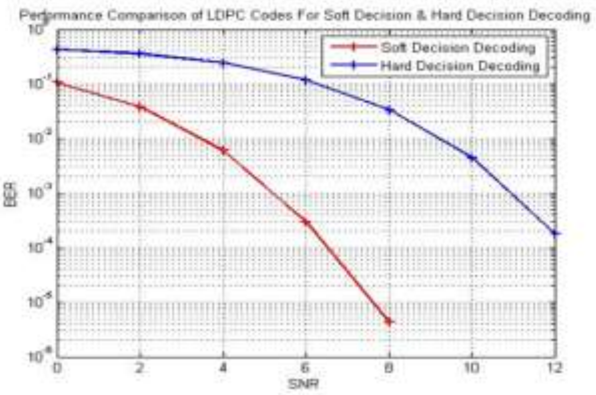

Fig 4: Performance analysis of hard decision and soft decision decoder.

From the table it is clear that, for SNR 10 and 12 the number of error in soft decision decoding is zero. But in hard decision decoding almost all SNR values have error. For the comparative study, consider the SNR=8. In hard decision decoding number of errors for SNR=8 is 53599. But for same SNR number of errors in soft decision decoding is 7. Thus soft decision decoding is giving exceptionally good decoding performance. Soft decision decoding can be done in both probability and log domain. Log domain decoding gives low decoding complexity than the probability domain. In this work log domain soft decision decoding was simulated.Fig.4 shows the comparative results between the hard decision and soft decision decoding based on BER. From the graph we can understand, in soft decision decoding BER 10^{-5} is obtained for a 8 dB SNR. But in the case of hard decision decoding the same BER is obtained at 12 dB SNR. Soft decision decoding allows an additional code gain between 2 or 3 dB when compared to hard decision decoding. Based on the simulation results we can infer that soft decision decoding gives the better performance than the hard decision decoding.

7. CONCLUSIONS

For any communication system our objective is to transmit data without error. For this we have to choose an appropriate forward error correction scheme which gives less probability of error. While considering the BER, LDPC is the most efficient FEC than other codes. LDPC provide high code gain at low BER, and less decoding complexity. LDPC is highly immune to noise. LDPC code can be decoded by using two method. First one is hard decision decoding and other is soft decision decoding. Considering two decoding schemes in LDPC , soft decision decoding gives better performance than the hard decision decoding. Soft decision decoding algorithm works in both log domain and probability domain. Log domain is less complex than probability domain. Considering different code rates, low code rate gives the better BER performance. Based on the simulation results, it is clear that LDPC gives the better performance with low code rates and using soft decision algorithm in log domain as the decoding algorithm. All these advantages of LDPC makes it fit for wireless communication, deep space communication, satellite communication and also for long haul optical communication. A communication system with LDPC can transmit data in long distance with less probability of error.

8. ACKNOWLEDGMENTS

We gratefully acknowledge the support and generosity to our colleagues from St. Joseph’s College of Engineering and Technology, who provided insight and expertise that greatly assisted the research.

9. REFERENCES

[1] Michel Zorzi, Performance of FEC and ARQ Error Control IN Bursty Channel Under Delay Constraints, VTC’98, OTTAWA Canada May 18-21, 1998.

[2] C E Shannon, A Mathematical Theory of Communications, Bell Systems Technology Journal, pages 379-423, 623-657, 1948.

[3] R. G Gallager, Low Density Parity Check Codes, IRE Transactions on Information Theory, vol-8. no.1, pp21-28, January 1962.

[4] Aqiel N. Almaamory, HusamAbduldaem Mohammed, PerformanceEvaluation and Comparison Between LDPC and Turbo Coded MCCDMA, Journal of Engineering, vol. 18, no. 8, April 2012., 2nd ed., January 2002

[5] European telecommunications standards institute, digital video broadcasting (dvb) second generation framing structure for broadband satellite applications, Pages 302-307 v1.1.1., URL: www.dvb.org

[6] B. Vasic , Ivan B. Djordjevic, Low density parity check code and iterative decoding for long haul optical communication system,Journal of light wave technology , vol.21.no.2. February 2003

[7] M. M Jadhav, AnkitPancholi, Dr. A. M Sapkal, Analysis and Implementation of Soft Decision Decoding Algorithm of LDPC ,International Journal of Engineering Trend and Technology(IJETT)- Vol4, Issue6, June 2013

[8] Tuan Ta, A Tutorial on Low Density Parity Check Codes, The University of Texas at Austin

[9] D. Mackay, R. Neal, Near Shannon Limit Performance of Low-Density Parity Check Codes , Electronics Letters, 32(18):1645-1646, August 1996

[10] Sarah. J. Jhonson, Introducing Low-Density Parity Check Codes, The University of Newcastle, Australia

[11] T. J Richardson, M. A Shokrollohi and R. L Urbanke, Design of Capacity Approaching Irregular Low Density Parity Check Codes, IEEE Trans. Inform. Theory, Vol. 47, no 2, pp 619-637, Feb 2001

[12] R. M Tanner, A Recursive Approach to Low Complexity Codes,IEEE Transaction on Information Theory, IT 27(5):533-547, Sep 1981

[13] Namrata P. Bhavsar, BrijeshVala, Design of Hard and Soft Decision Decoding Algorithms of LDPC, International Journal of Computer Applications (0975 8887) Volume 90 No 16, March 2014.

[14] Lakshmi.R, Tilty Tony, Abin Johns Raju, An Analytical Approach to The Performance of Low Density Parity Check Codes, International Conference on Advanced Computing and Communication Systems (ICACCS - 2013), Dec. 19 21, 2013

[15] VikramArkalgudChandrasetty, Syed Mahfuzul Aziz, FPGA Implementation of a LDPC Decoder using a Reduced Complexity Message Passing Algorithm , Journal of Networks, Vol. 6, no. 1, January