International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459, Volume 2, Issue 6, June 2012)

# Error Performance of Decoding of LDPC over AWGN Channel.

G. Renuka

*Electronic and Communication Engineering Department*
*S.R Engineering College, Warangal. A.P, INDIA*

***Abstract*—The design of future communication systems with high throughput demands will become a critical task, especially when sophisticated channel coding schemes have to be applied. Low-Density Parity-Check(LDPC) codes are among the most powerful forward error correcting codes, since they enable to get as close as a fraction of a dB from the Shannon limit. A simplified decoding algorithms for decoding Low-Density Parity-Check (LDPC)codes is proposed with a view to reduce the implementation complexity. The algorithm is based on simple hard-decisiondecoding techniques while utilizing the advantages of soft channel information to improve decoder performance . This astonishing performance combined with their relatively simple decoding algorithm makes these codes very attractive for the next digital transmission system generations.The parity check matrix used for the LDPC codes is sparse. The complexity and performance of codes are improved by the modifications in the decoding algorithm. This work deals with study and implementation of LDPC coding system over AWGN Channel.**

***Keywords*— LDPC Low Density Parity Check, AWGN Additive White Gaussian Niose, BER Bit error Rate, SPA Sum Product Algorithm,MPA Message Passing Algorithm, BP Belief Propagation, MS Min-sum Algorithm**

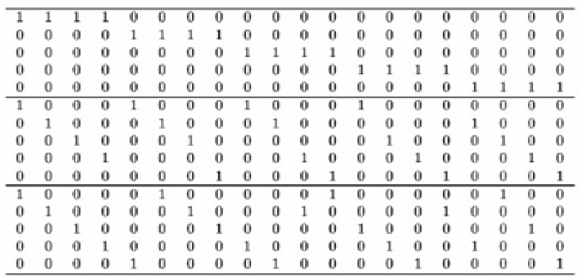
## I. INTRODUCTION

From the perspective of the receiver there are many ways to combat with the disturbance caused by the channel. In short, coded transmission permits two measures :error detection and correction. In coding theory by adding redundancy bits to the transmission, it is possible to detect the presence of errors. If the error can not be located, then the message is retransmitted until the error free message is received by the receiver. This method is called ARQ (automatic repeat and request). In forward error correction redundancy bits are added to information bits and used to reconstruct the original information without errors. These are represented by their sparse parity check matrix. The matrix constructed over the field GF(2) contains only 0s and 1s. According to Gallager, the matrix contains fixed number of ones in the columns and fixed number of ones in the rows. These codes are called regular LDPC codes.

If the matrix does not contain the fixed number of ones in columns and rows then those codes are called irregular LDPC codes.Galler defined the(n,j,k)matrix as n columns,j number of ones in each column and k number of ones in each row,(n,j,k) defines that the number of rows=n.j/k.thus the code rate R=1-j/k;

Low-density parity-check codes offer performances spectacularly close to theoretical limits when decoded using iterative soft-decision decoding algorithms based on factor graphs. Indeed, a rate R = 1/2 LDPC code with a block length of 107 bits has been shown to perform within 0.04 dB of the Shannon limit for the binary input additive white Gaussian noise channel at a bit error rate of 10−6, and has a threshold only 0.0045 dB away from that limit. Both LDPC codes and turbo codes are capacity-approaching codes in that sense. They are both theoretically and practically capable of solving Shannon's channel coding problem. LDPC codes are mainly used for error detection and correction in communication systems. Performance of irregular codes is better than regular codes. It is already the case for the next digital satellite broadcasting standard (DVB-S2), where an irregular LDPC code has been chosen to protect the downlink information. The other applications are Digital Subscriber Line(DSL), WiMAX (IEEE 802.16e standard for microwave communications), 10GBase-T Ethernet(802.3an)

Examlple of low density parity check code matrix for n=20,j=3,k=4.



| | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |
| 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 |

International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459, Volume 2, Issue 6, June 2012)

## II. Representation Of LDPC Codes

LDPC is defined by a sparse paritycheck matrix H.Suppose matix H has n columns and m rows,and then the codeword consists of n bits which satisfy m parity check.1 in the matrix indicates that the bit is involved in a parity check.the total length of the codeword is n bits,the number of message bits are k=n-m,the rate of the code is k/n.Example for the parity check matrix and parity check equations are shown below

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

The parity check equations are

$f_0=c_0 \oplus c_1 \oplus c_2 \oplus c_3$

$f_1=c_0 \oplus c_4 \oplus c_5 \oplus c_6$

$f_2=c_1 \oplus c_4 \oplus c_7 \oplus c_7$

$f_3=c_2 \oplus c_5 \oplus c_7 \oplus c_9$

$f_4=c_3 \oplus c_6 \oplus c_8 \oplus c_9$

An important advance in the theory of LDPC codes is made possible by Tanner's method of using bipartite graphs to provide a graphical representation of the parity check matrix. As a consequence, the bipartite graph of a LDPC code is sometimes referred to as a Tanner graph. In graphs edge is the connection between the two nodes. A bipartite graph is a graph in which the nodes may be partitioned into two subsets such that there are no edges connecting nodes within a subset. In the context of LDPC codes, the two subsets of nodes are referred to as variable nodes and check nodes. There is one variable node for each of the n bits in the code and there is one check node for each of the m rows of H. An edge exists between the ith variable node and the jth check node if and only if hij = 1. Here, hij is an element in the H matrix at the ith row and jth column. An example of a bipartite graph for the above matrix is shown in the Figure 3.3. In a graph, the number of edges incident upon a node is called the degree of the node.

A cycle of length l is the path comprising of l edges which closes back itself. The solid line shown in the Figure.2.3 indicates the cycle in the graph. The length of the short cycle in the bipartite graphs is called girth of the graph. The performance of these codes will be improved by increasing the girth of the graph

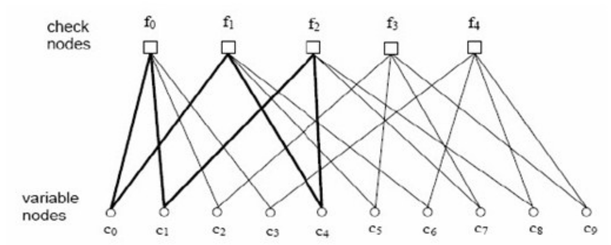


**Fig: Bipartite graph for the LDPC code of length 10**

*2.1 Types of LDPC:*

1.Regular LDPC codes

2.Irregular LDPC codes

*2.2 Construction of the Parity Check Matrix (H)*

The H matrix is to be constructed by using the following step

i. The Matrix H of size m x n is generated, which contains all elements zeroes in it

ii. In the matrix H, in each column wc (column weight) numbers of bits are flipped in distinct rows.

iii. The ones in the H matrix are rearranged between rows to make row weight wr as uniform as possible.

iv.If any row contains 1's less than two, the corresponding code bit will be zero. There should be at least two 1's in each row.

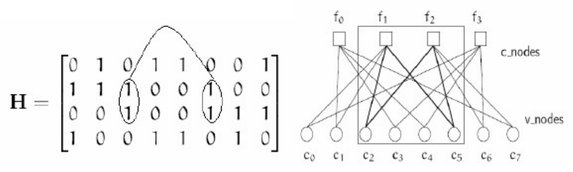


**Fig.2.8 overlapping of ones in H matrix and Corresponding short cycle in graph**

International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459, Volume 2, Issue 6, June 2012)

v. Figure 2.8 shows the overlapping of two ones in two columns. The ones are rearranged between rows and columns to avoid these overlapping, then H matrix will have girth-6 in bipartite graph.

*2.3.Parity check matrix in Systematic form:*

The H matrix can be composed of two sparse matrices C1 , C2 [2] .

$$H = [C_1 \mid C_2]_{m \times n}$$

$$H_{sys} = [\, I_m \mid P \,]_{m \times n}$$

*2.4. Generator Matrix:*

$$G = [\, P^T \mid I_k \,]_{k \times n}$$

The relation between the G and H matrix is

$$G.H^T = 0 .$$

*2.5 .Encoding:*
The randomly generated multiplying the message bits are encoded, by message bits with the Generator matrix,

$$c = u .G$$

Here c is a codeword of length n. Here the encoder is systematic so we maygenerate only parity check bits [2]. The parity check bits are computed as p = c1−1c2u .

*2.6 .Channel:*

Channels are the medium through which the data is transferred from source to the destination.some noise may be added to the data passed through the channels because of imperfect nature of the materials in the channels or by the effects of the environment.This corrupted data is received at the decoder.The message decoded by the decoder will be send to destination.

*Additive White Gaussian Noise(AWGN)Channel:*

The encoded code word is transmitted through the AWGN (additive whiteGaussian noise) channel [4]. In the channel, the binary vector c is first mapped intotransmitted signal vector t.

The binary phase-shift keyed (BPSK) signal constellation is employed, so that the signal a = ES represents the bit 1 and the signal − a representsthe bit 0. The energy per message bit Eb is related to transmitted code bit ES byES =R Eb . Here, R = k n is the rate of the code. The transmitted signal vector t has elements tn = (2cn − 1)a . The signal vector passes through channel which adds a Gaussian random noise vector n , where each element of n is independent, identically distributed

r =t+n.

Given the received data, the posterior probability of detection can be computed as

$$\frac{E_b}{N_o} = \frac{E_s}{2R\sigma 2}$$

$$SNR(dB) = 10\log_{10}\left[\frac{E_b}{N_o}\right]$$

$$\sigma = \sqrt{\frac{1}{2R\frac{E_b}{N_o}}} \qquad n = mean + \sigma \times \bigcup$$

The received signal is $r = t + n$

$$p\left(c_n = {}^{1}\!/_{rn}\right) = \frac{1}{1 + e^{-2arn/\sigma 2}}$$

## III. Implementation Of The Decoding System

In this chapter, implementation of decoding system will be discussed. The system mainly contains encoder, channel and decoder. In the encoder implementation, parity check matrix is constructed and a standard encoding method used to encode the message. Implementation of the AWGN and Rayleigh channel is discussed. And finally the implementation of decoder by using bit flip algorithm, Belief propagation algorithm, sum product algorithm in log domain, min-sum algorithm and min-sum algorithm with correction factor decoding is discussed.

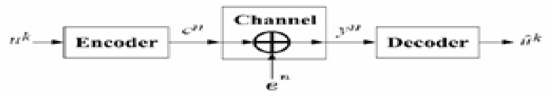


**Fig.1 Block Diagram of an error correct coding communication system with an additive white Gaussian noise**

The error correcting coding scheme shown in Fig.3.1 comprises of two components: the encoder and the decoder. For low density parity check codes (and linear block codes in general) the encoder takes binary sequences of length k and produces encoded binary sequences of length n > k which contain m = n -k parity-check bits [4]. And the noise is added to this binary sequence by the Additive White Gaussian Noise process [5]. The decoder receives the encoded data after it has been affected by the channel noise and attempts to correctly determine what was sent from encoder by correcting the errors in the receiving data.

*Encoder:*

The channel encoder adds the redundancy bits to the message by using some constraints according to the message bits [4]. These extra bits are used at the channel decoder to estimate the transmitted code. The main part of encoding is generating the parity bits. The generator matrix is used to generate parity bits. The implementation of encoder [2] is explained below. Implementation of the encoder

i. Construction of the parity check matrix.

ii. Rearrangement of the parity check matrix into systematic form.

iii. Generating the Generator matrix from rearranged H matrix.

iv. Multiplying the message word with generator matrix to get encoded codeword.

*Construction of the Parity Check Matrix (H)*

The H matrix is to be constructed by using the following procedure [2]

i. For the given size (m x n), an all zero matrix is generated. In this example m=5 n=10.

$$H=\begin{bmatrix} 0\,0\,0\,0\,0\,0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0\,0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0\,0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0\,0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0\,0\,0\,0\,0\,0 \end{bmatrix}$$

ii. In this implementation column weight wc =3. So 3 bits each column are flipped to 1 in distinct rows. The total number of 1's in the matrix is n wc . In this example wc =2

$$H=\begin{bmatrix} 1\,1\,1\,1\,0\,0\,0\,0\,1\,0 \\ 0\,0\,1\,0\,1\,1\,0\,0\,1\,1 \\ 0\,0\,0\,1\,1\,0\,1\,1\,0\,0 \\ 1\,0\,0\,0\,0\,1\,0\,1\,0\,1 \\ 0\,1\,0\,0\,0\,0\,1\,0\,0\,0 \end{bmatrix}$$

iii. If any row contains 1's less than two, the corresponding code bit will be zero. Atleast two 1's in each row.

iv. The number of 1's are rearranged between different rows to make row weight as uniform as possible. The row weight wr=nwc/m. In this example wr=4.

$$H=\begin{bmatrix} 1\,1\,1\,1\,0\,0\,0\,0\,1\,0 \\ 0\,0\,0\,0\,1\,1\,0\,0\,1\,1 \\ 0\,0\,0\,1\,1\,0\,1\,1\,0\,0 \\ 1\,0\,0\,0\,0\,1\,0\,1\,0\,1 \\ 0\,1\,1\,0\,0\,0\,1\,0\,1\,0 \end{bmatrix}$$

v. In the matrix H no two column should have overlap greater than one, then it will have girth of 6 in bipartite graph. In the following matrix rows 1 and 5 shares 1's with the columns 2 and 3.

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

These are removed by row and column arrangements. After elimination of overlapping of 1's the matrix will be as below.

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

*Encoding:*

The message bits are randomly generated by using uniform distribution and these bits are multiplied with the Generator matrix to get the encoded code word [4].

c = u .G

For example u=[0 1 0 1 0] then c=[1 1 0 0 0 0 1 0 1 0].

In this implementation, only parity bits are generated and message bits are appended to the parity bits. In systematic encoding message bits are directly available.

*Decoder:*

The decoder receives the code from the channel which contains errors. It uses the parity bits information and H matrix to correct the errors in the code word. The decoder can be implemented by using Hard Decision Decoding (HDD) and Soft Decision Decoding (SDD) methods.

1. Hard Decision Decoding (Bit Flip Algorithm)

2. Soft Decision Decoding

i. Belief Propagation Algorithm

$$P_t=\Pr\left(ci={1}/{Y_t}\right)=\frac{1}{\left(1+\exp-2aYi/\sigma 2\right)}$$

$\sigma$ =standard deviation

$y_t$ =float value received from the channel

ii. Log-domain Sum Product Algorithm

$$L(C)=\log\left(\frac{P(C=0)}{P(C=1)}\right)$$

initialization $L(C_i)=\log\left\{P(C_i={0}/{Y_i})/P(C_i={1}/{Y_i})\right\}$

$$L(qi_j)=L(c_i)={-2ay_i}/{\sigma 2}$$

Horizontal step

$$L(qi_j)=\alpha_j\beta i_j$$

Vertical step

$$L(qi_j)=L(c_i)+\sum_{j'\in C(i)\backslash j}L(r_{j'i})$$

iii. Min-Sum Algorithm

$$L(q_{ij})=L(C_i)=-ay_i$$

$$L(q_{ij})=L(C_i)+\sum_{j'\in C(i)\backslash j}L(r_{ij})$$

iv. Min-Sum algorithm with correction factor

For each edge e, the magnitude of the outgoing message is $|m_{e,out}|$

$$|m_{e,out}|=\max\left(\min\beta_{rj}-Y_c,0\right)$$

## IV.RESULTS

The performance of different block length codes and different algorithms over additive white Gaussian noise are evaluated. The coding scheme is implemented in c programming language and the figures are drawn in Matlab. In the simulation, source information is randomly generated by the uniform distribution. Column weight is 3 and row weight is 4 in parity check matrix. The encoder encodes the message bits block by block. In this work ten million message bits are transmitted through the channel with block lengths (100, 500 and 1000) at the rate of 0.5. the performance plots are drawn for bit error rate Versus Eb/No..

These codes are simulated for the range 0dB to 5dB.

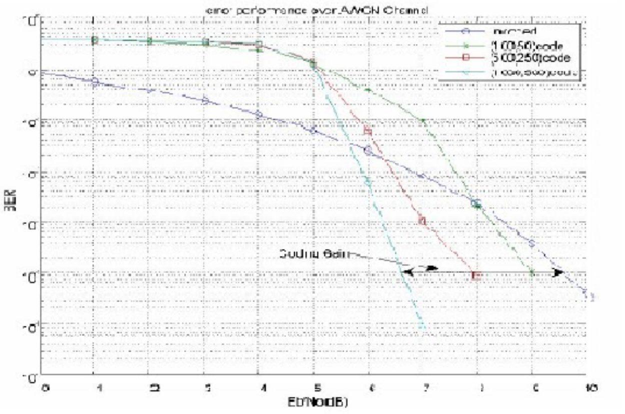


**Fig.4.1 Error performance of different block length codes with bit-flip algorithm decoding over AWGN channel.**

| Eb/No (in dB) n=1000 | uncoded (R=1) | n=100, R=0.5 | n=500, R=0.5 | n=1000, R=0.5 |
|---|---|---|---|---|
| 0.0 | $7.9\times 10^{-2}$ | $3.67\times 10^{-1}$ | $3.69\times 10^{-1}$ | $3.68\times 10^{-1}$ |
| 1.0 | $5.61\times 10^{-2}$ | $3.55\times 10^{-1}$ | $3.57\times 10^{-1}$ | $3.58\times 10^{-1}$ |
| 2.0 | $3.70\times 10^{-2}$ | $3.36\times 10^{-1}$ | $3.44\times 10^{-1}$ | $3.46\times 10^{-1}$ |
| 3.0 | $2.29\times 10^{-2}$ | $3.03\times 10^{-1}$ | $3.34\times 10^{-1}$ | $3.34\times 10^{-1}$ |
| 4.0 | $1.26\times 10^{-2}$ | $2.32\times 10^{-1}$ | $3.04\times 10^{-1}$ | $3.16\times 10^{-1}$ |
| 5.0 | $6.02\times 10^{-3}$ | $1.26\times 10^{-2}$ | $1.29\times 10^{-1}$ | $1.23\times 10^{-1}$ |
| 6.0 | $2.42\times 10^{-3}$ | $3.83\times 10^{-2}$ | $5.90\times 10^{-3}$ | $5.60\times 10^{-4}$ |

**TABLE.4.1**

**BER Vs Eb/N0 for different block length codes with bit-flip algorithm decoding over AWGN channel**

| Eb/No (in dB) n=1000 | n=100, R=0.5 | n=500, R=0.5 | n=1000, R=0.5 |
|---|---|---|---|
| 0.0 | $1.24\times 10^{-1}$ | $1.29\times 10^{-1}$ | $1.29\times 10^{-1}$ |
| 0.5 | $9.76\times 10^{-2}$ | $1.00\times 10^{-1}$ | $1.03\times 10^{-1}$ |
| 1.0 | $6.98\times 10^{-2}$ | $6.13\times 10^{-2}$ | $6.01\times 10^{-2}$ |
| 1.5 | $4.45\times 10^{-2}$ | $2.46\times 10^{-2}$ | $1.65\times 10^{-2}$ |
| 2.0 | $2.46\times 10^{-2}$ | $5.94\times 10^{-3}$ | $2.23\times 10^{-3}$ |
| 2.5 | $1.14\times 10^{-2}$ | $1.53\times 10^{-3}$ | $6.85\times 10^{-4}$ |
| 3.0 | $4.53\times 10^{-3}$ | $7.54\times 10^{-4}$ | $2.27\times 10^{-4}$ |

**TABLE4.2**

**BER Vs Eb/N0 for different block length codes with Belief propagation algorithm over AWGN channel**

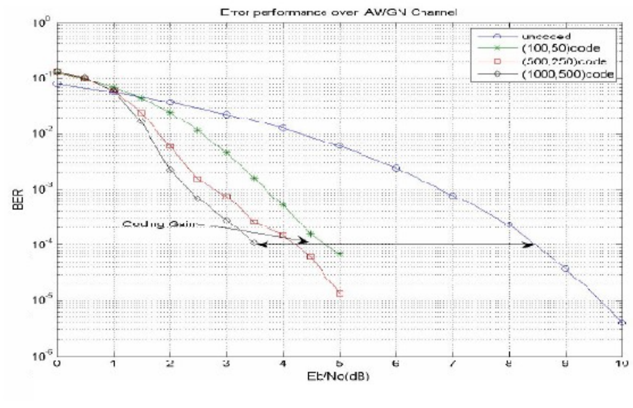


Fig.4.2 Error performance of different block length codes with Belief propagation decoding over AWGN channel.

| Eb/No (in dB) n=1000 | n=100, R=0.5 | n=500, R=0.5 | n=1000, R=0.5 |
|---|---|---|---|
| 0.0 | $1.24\times 10^{-1}$ | $1.29\times 10^{-1}$ | $1.30\times 10^{-1}$ |
| 0.5 | $9.76\times 10^{-2}$ | $1.00\times 10^{-1}$ | $1.03\times 10^{-1}$ |
| 1.0 | $6.98\times 10^{-2}$ | $6.09\times 10^{-2}$ | $5.98\times 10^{-2}$ |
| 1.5 | $4.44\times 10^{-2}$ | $2.36\times 10^{-2}$ | $1.51\times 10^{-2}$ |
| 2.0 | $2.44\times 10^{-2}$ | $4.92\times 10^{-3}$ | $1.34\times 10^{-3}$ |
| 2.5 | $1.11\times 10^{-2}$ | $5.31\times 10^{-3}$ | $2.50\times 10^{-4}$ |
| 3.0 | $4.29\times 10^{-3}$ | $4.20\times 10^{-4}$ | $0.00\times 10^{-4}$ |

TABLE.4.3

BER Vs Eb/N0 for different block length codes with log- Sum Product Algorithm over AWGN channel

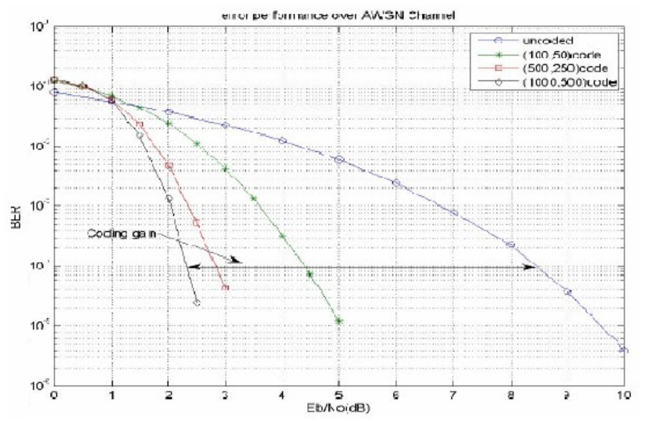


Fig.4.3 Error performance of different block length codes with log-sum product algorithm decoding over AWGN channel.

| Eb/No (in dB) | n=100, R=0.5 | n=500, R=0.5 | n=1000, R=0.5 |
|---|---|---|---|
| 0.0 | $1.53\times 10^{-1}$ | $1.66\times 10^{-1}$ | $1.66\times 10^{-1}$ |
| 0.5 | $1.24\times 10^{-1}$ | $1.41\times 10^{-1}$ | $1.43\times 10^{-1}$ |
| 1.0 | $9.18\times 10^{-2}$ | $1.06\times 10^{-1}$ | $1.13\times 10^{-1}$ |
| 1.5 | $6.07\times 10^{-2}$ | $5.91\times 10^{-2}$ | $6.29\times 10^{-2}$ |
| 2.0 | $3.49\times 10^{-2}$ | $1.96\times 10^{-2}$ | $1.35\times 10^{-2}$ |
| 2.5 | $1.67\times 10^{-2}$ | $3.09\times 10^{-3}$ | $6.75\times 10^{-4}$ |
| 3.0 | $6.50\times 10^{-3}$ | $1.82\times 10^{-4}$ | $9.00\times 10^{-6}$ |
| 3.5 | $2.08\times 10^{-3}$ | $1.30\times 10^{-6}$ | $0.00\times 10^{-6}$ |
| 4.0 | $5.05\times 10^{-4}$ | $5.00\times 10^{-6}$ | $0.00\times 10^{-6}$ |
| 4.5 | $1.19\times 10^{-4}$ | $0.00\times 10^{-6}$ | $0.00\times 10^{-6}$ |
| 5.0 | $1.80\times 10^{-5}$ | $0.00\times 10^{-6}$ | $0.00\times 10^{-6}$ |

TABLE.4.4

BER Vs Eb/N0 of different block length codes in min-sum decoding over AWGN channel

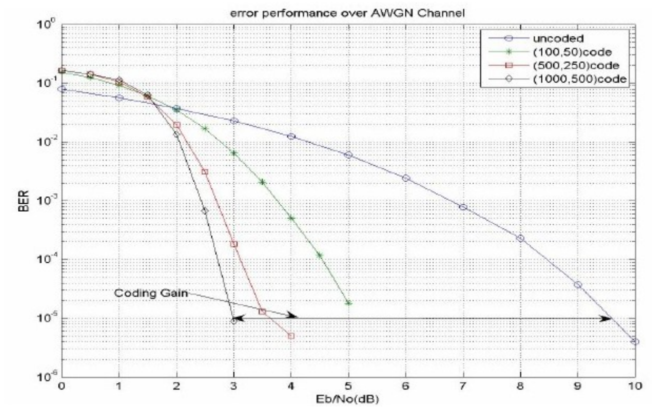


Fig.4.4. Error performance of different block length codes with min-sum decoding over AWGN channel

| Eb/No (in dB) | n=100, R=0.5 | n=500, R=0.5 | n=1000, R=0.5 |
|---|---|---|---|
| 0.0 | $1.29\times 10^{-1}$ | $1.70\times 10^{-6}$ | $1.36\times 10^{-1}$ |
| 0.5 | $1.02\times 10^{-1}$ | $1.06\times 10^{-1}$ | $1.09\times 10^{-1}$ |
| 1.0 | $7.52\times 10^{-2}$ | $1.06\times 10^{-1}$ | $7.30\times 10^{-2}$ |
| 1.5 | $4.98\times 10^{-2}$ | $3.29\times 10^{-2}$ | $2.75\times 10^{-2}$ |
| 2.0 | $2.86\times 10^{-2}$ | $8.52\times 10^{-3}$ | $3.13\times 10^{-3}$ |
| 2.5 | $1.39\times 10^{-2}$ | $9.21\times 10^{-4}$ | $6.40\times 10^{-6}$ |
| 3.0 | $5.59\times 10^{-3}$ | $6.40\times 10^{-6}$ | $0.00\times 10^{-6}$ |
| 3.5 | $1.80\times 10^{-3}$ | $1.40\times 10^{-6}$ | $0.00\times 10^{-6}$ |
| 4.0 | $4.63\times 10^{-4}$ | $1.00\times 10^{-6}$ | $0.00\times 10^{-6}$ |
| 4.5 | $1.04\times 10^{-4}$ | $0.00\times 10^{-6}$ | $0.00\times 10^{-6}$ |
| 5.0 | $1.70\times 10^{-5}$ | $0.00\times 10^{-6}$ | $0.00\times 10^{-6}$ |

TABLE.4.5

BER Vs Eb/N0 of different block length codes with min-sum with correction factor decoding over AWGN channel

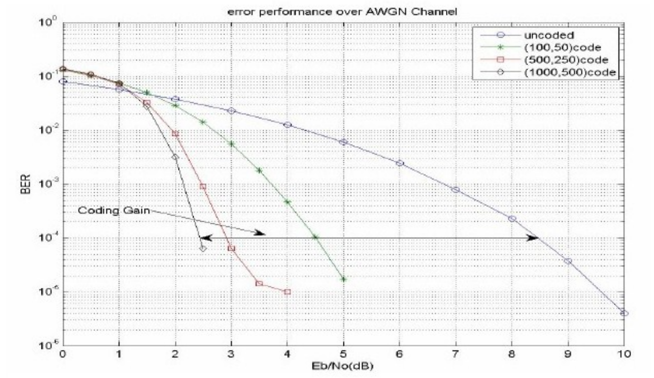


Fig.4.5.Error performance of different block length codes with min-sum algorithm with correction factor decoding over AWGN channel

| Algorithm | BER | Coding gain(in dB) |
|---|---|---|
| Bit flip algorithm | $10^{-4}$ | 2.25 |
| Belief Propagation | $10^{-4}$ | 5.00 |
| Log sum product | $10^{-4}$ | 6.25 |
| Min sum algorithm | $10^{-4}$ | 5.75 |
| Min sum with correction | $10^{-4}$ | 6.00 |

TABLE.4.5

Coding gain for AWGN channel. Block length=1000

The following termination constraint has been used.
1.$CH^T=0$ is satisfied in the decoding algorithm.
2.The decoding algorithm computes for 50 iterations.

## V. CONCLUSION

This work presents an implementation of a channel coding scheme. It contains encoder, channel and a decoder. Encoder is implemented using standard encoding method. Parity check matrix is constructed and generator matrix is implemented from parity check matrix. Additive white Gaussian noise channel model is implemented. Decoder is implemented using both soft decision and hard decision decoding algorithms. The performance of different iterative algorithms is compared.

The performance of the modified min-sum is nearer to the log-sum product algorithm decoding with low complexity.
Comparison between hard and soft decision decoding is carried out. Simulations are carried out for the large and small block length codes. Simulation results show that performance of log-domain sum product algorithm is best among the soft decision decoding algorithms. The performance is better for large block length codes. The min-sum with correction factor algorithm performs better than original min-sum algorithm.

## REFERENCES

[1] J.. Chen and M. Fossorier, "Near optimum universal belief propagation based decoding of low density parity check codes",vol.50,pp.406-414,Mar,2002.

[2] Torben Brack, Frank Kienle, Norbert Wehn "Disclosing the LDPC Code Decoder Design Space" in 2006.

[3] Mohamed.Shaqfeh and Norbert Goertz. "Systematic modification of Parity CheckMatrices for efficient encoding of LDPC codes" ICC proceedings 2007 IEEE Communication society.

[4] Kiran K.G, G.S.Choi, Mark.B.Yeary and Mohammed Atiquzzaman"VLSI Architecture for Layered Decoding for Irregular LDPC Codes of WiMax" ICC proceedings 2007 IEEE Communication society.

[5] Bernhard M.J.Leiner " LDPC codes-a brief tutorial" in April 2005.

[6] D. J. C. MacKay, "Good error-correcting codes based on very sparse matrices,"IEEE Trans. Inform. Theory, vol. 45, no. 2, pp. 399.431, March 1999

[7] N. Axvig, K. Morrison, E. Psota, D. Dreher, L.C. P´erez, and J. L. Walker Towards a universal theory of decoding and pseudocodewords. SGER Technical Report 0801, University of Nebraska-Lincoln, March 2008.

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

[9] http://www.csee.wvu.edu/wcrl/ldpc.html.