Performance Comparison of Binary LDPC Decoders and FPGA Implementation of Encoder

Jayashree C. Nidagundi and Dr. Siddarama R. Patil

Abstract--- Modern Communication standards deals with high and efficient data transmission and reception. LDPC codes show the error correction performance very close to Shannon's limit. The proposed work deals with performance analysis of both hard and soft decoding methods of LDPC codes and FPGA implementation of encoder and hard decision decoding. Soft decision decoders shows very good performance for rate $1/2$ (8,4) codes.

Index Terms--- Less Density Parity Check Matrix (LDPC), (Field Programmable array logic (FPGA), VHDL, BER(bit error rate).

I. INTRODUCTION

LDPC codes generally outperform compare to other error correction techniques used in all wired and wireless communication standards due to their iterative decoding and inherent parallelism of decoding algorithms. These LDPC codes were started with discovery by Gallager in 1962[1]. His codes performance is very near to Shannon's performance limit, when these codes are decoded with the use of probabilistic methods. Later these codes are reinvented by Neal and Mackay[2] constructing both regular and irregular binary LDPC codes. Due to their very good error correcting capabilities these codes are used as the forward error correction methods for many recent standards such as digital video broadcasting satellite (DVB-S2) [5], the WiMAX standard for microwave communications (802.16e) [3], the G.hn/G.9960 standard for wired home networking [6], and the 10GBASE-T standard for 10 Gigabit Ethernet (802.3an) [4]. The proposed work deals with performance observation of both hard and soft decoding methods of LDPC codes and FPGA implementation of encoding of randomly constructed parity check matrix and bit flipping decoding algorithm. Section II deals with description of LDPC codes, section III with encoding of LDPC codes, section IV with different decoding algorithms, section V with Implementation of LDPC encoder and decoder, section VI results and discussion followed by conclusion.

II. LDPC CODE CONSTRUCTION

Low density parity check (LDPC) codes are one type of forward error correction codes used in most of the recent communication standards like WiMAX, WiFi and

IEEE802.16e etc. The LDPC codes are block codes constructed using very sparse parity check matrix(H). LDPC codes are called as regular if the row weight and column weight of parity matrix is fixed in all rows and all the column of the check matrix otherwise they are called as irregular codes. Parity check matrix is also represented by Tanner graph as shown in figure1.

Figure 1: Parity Check Matrix and TANNER Graph [2]

LDPC parity-check matrix are identified by $(L_c; L_r)$ -regular if number of ones in each column is constant (L_c) and the number of ones in each row will be also as constant (L_r). If this constraints for $(L_c; L_r)$ not considered as constants then the LDPC codes are called as irregular codes. The irregular codes are identified with degree (v, h) , where v indicates the fractional weight row and h indicates the fractional weight of column. Most of the results from the literature shows irregular codes perform better than regular codes with more hardware decoder implementation issues [4]. The parity check matrix H can be implemented using random construction, semi-random construction and also structured construction. Here we have taken irregular and regular $(10,3,5), (10,3,6)$ parity check matrices with the above mentioned random construction. Parity check matrices generated are very sparse. Here the Mackay Neal construction method is used to construct LDPC codes by avoiding 4-cycle free code. The method used to generate the H with given code rate r , codeword length n and degree of row and column v and h as follows:

Algorithm 1: H matrix Construction [2]

```

HMNConstruct(n,r,v,h) --required length,rate,degree
distribution
Initial H[] = all zero
Choose  $\alpha_i$  and  $\beta$  as initial degree
Construct row from 1 to m with degree of  $h$ 
  
```

Jayashree C. Nidagundi, Assistant Professor, Department of Electronics & Communication Engineering, SDM College of Engineering & Technology, Dharwad, India. E-mail: jayaprajwal@rediffmail.com

Dr. Siddarama R. Patil, Professor, Department of Electronics & Communication Engineering, PDA College of Engineering, Kalaburgi, India. E-mail: pdapatil@gmail.com

DOI:10.9756/BIJRCE.8203

Construct row from 1 to n with degree of v
 Fill all nonzero of H matrix with random construction
 Modify the H to avoid 4 cycles
 End;

III. ENCODING OF LDPC CODES

LDPC codes constructed using sparse parity check matrix of fixed row weight and column weight and then encoded then passed through AWGN channel. At the receiver side transmitted information can be decoded to get a original message digits. To encode LDPC codes different types of encoding methods are used as Gauss elimination method, LU factorization method, By using generator matrix $c=u*G$, Approximation linear triangular method (ALT) and Greedy permutation method. Here LU factorization method is used as shown in below algorithm 2.

Algorithm 1: LU Factorization Method

step1: we write H matrix is constructed and linear homogenous equations can be expressed as

$$H_x = 0 \quad (1)$$

Where x = codeword/message bits = c/m

step2: Dividing H matrix into $M \times M$ left part of A, $M \times N/2$ right part of B (where M and N are rows and columns of H matrix)

$$\begin{bmatrix} A \\ B \end{bmatrix} = 0 \quad (2)$$

step3: By putting x value in above equation in above equation (2) then simplifying it

$$A * B + C * m = 0 \quad (3)$$

$$C = A^{-1} * B * m \quad (4)$$

Thus above matrix 'A' can be written by $A=L*U$, where 'L' is lower triangular matrix and 'U' is upper triangular matrix. Substituting above formula in (4) we get

$$C = (LU^{-1}) * B * m \quad (5)$$

IV. DECODING METHODS OF LDPC CODES

LDPC codes are decoded using message passing algorithm as message is passed between check nodes to variable nodes and vice versa and it is iterative method. There are two types of decoding method they are hard decision and soft decision message passing. All algorithms consisting of 3 decoding steps: initialization, horizontal step updating and vertical step updating. In the initialization phase the received channel messages are collected and are initialized to all the variable nodes. In horizontal step updating the received message from variable nodes at the check nodes are computed and updated message sent back to the vertical nodes. In vertical step updated message from horizontal nodes is computed at vertical nodes and syndrome calculation ($mH^T=0$) then process terminates if not process will continue till syndrome is equals to zero or for fixed number of iterations.

A. Hard Decision Technique

The bit-flipping decoding is a hard-decision type of message-passing used in LDPC decoding. It uses binary information in the form of 0 and 1 of the channel considered this information is available at the receiver. This received

information is initialized to the bit nodes in the initial phase. In the check node update phase all the bit nodes sends binary message declaring if it is a one or a zero. All check nodes update the information by XOR received information and sends back the same message to the connected bit nodes. At the bit node update phase all the bit nodes verifies that, if the majority of the messages received by a bit node are different from its received value the bit node changes (flips) its current value. This process is repeated until all of the parity-check equations are satisfied at the bit node.

B. Soft Decision Algorithms

Soft decision algorithms accept the probability of each received bit as input message. The input bit probabilities are called the a priori probabilities for the received bits as they were known in advance before running the LDPC decoder. The bit probabilities returned by the decoder are called the a posteriori probabilities.

a) Sum Product Algorithm

The sum-product decoding works in a similar manner as that of bit flipping but received information here is probability value. Here also there are initial, check node phase and variable node phase steps are also there and the procedure is explained with following steps.

Algorithm 2: Sum Product Decoding

Step1: Initialization step

A priori information $L_n=r_n$

Bit to check node initial message $F_{m,n}=L_n$

Step2: Horizontal Step: Check node processing

$$E_{m,n} = \log \frac{1 + \prod_{n' \in N(m)} \tanh \frac{L_{n'}, m/2}{2}}{1 - \prod_{n' \in N(m)} \tanh \frac{L_{n'}, m/2}{2}} \quad (6)$$

Step3: Vertical step: Variable node processing Let variable node posterior information is:

$$L_n = L_n + \sum_{m \in M(n)} E_{m,n} \quad (7)$$

Variable node update message

$$F_{m,n} = L_n + \sum_{m' \in M(n)} E_{m',n} \quad (8)$$

The procedure will be repeated until $HC^T=0$, Then decoding procedure will stops.

b) Min Sum Algorithm

The min sum decoder further modifies sum product decoding by multiplying the optimization factor (α) [7] to both horizontal and vertical step updation process. The main use of optimization factor (α) is, at particular value it will decrease the bit error rate makes increase in performance of the decoder. But one disadvantage of this decoder is high complexity during the decoding of information received from channel. Generally we consider the ' α ' value as below 0.8db, if we consider the value of ' α ' beyond this limit we won't get any result because when we plot the graph for different values of ' α ' with bit error rate, at 0.8 db curve suddenly increase its bit error rate causing the decreasing in decoder performance.

The BER [8] is a key parameter that is used to check the system that transmits the digital data from source to destination. Generally BER test can be used to check both wired and wireless communication performance. The bit error

rate can be defined as follows:

$BER = \frac{\text{number of errors occur}}{\text{total number of digital data sent}}$

The BER depends on the following parameters.

1. Number of iterations
2. Size of the H matrix
3. Noise factors
4. Number of data sent

V. IMPLEMENTATION OF LDPC ENCODER AND DECODER

The parity check matrix H is generated randomly then the encoding of message bits is carried out using LU factorization technique as described in section II. The codeword generated and tested for the parity check equation then the signal will be passed to the BPSK modulator and passed through AWGN (additive white Gaussian noise) channel for the transmission. At the receiver end, the corrupted data is detected and the decoder is designed for the following parameter: H matrix size, Number of iteration, Noise parameter and Modulated signal. In this paper we have used both hard and soft decision decoding algorithms as discussed in section III are used to decode the original information. The MATLAB coding is used to study the performance of different decoding algorithms. The BER performance of all the decoder is obtained.

After verifying the performance of different decoders the architecture involved to carry out the encoding and decoding computation is understood by writing HDL code. For the encoder design, VHDL code is written considering the same parity check matrix as described in section II, which is converted into systematic generator matrix by row and column transformation, then codeword are generated using the equation

Codeword $C = \text{binary message information } X [G]$, where G generator matrix, The Top level RTL is generated for the same as shown in figure 2 and 3.

Figure 2: RTL Schematic of Encoder

Figure 3: RTL Schematic of Bit Flipping Decoder

VHDL code is also written for the bit flipping hard decision decoding algorithm considering register for data storage and XOR operation at both check node and variable node update steps.

VI. RESULTS AND DISCUSSIONS

The binary LDPC codes are designed according to the following specification as table 1

Table 1: Specification of Designed LDPC Code

LDPC codes	Rate(1/2)
Channel	AWGN
Modulation	BPSK
Encoding	LU Factorization
Decoding	Bit Flipping, Sum-Product, Min-Sum
Max.No of iterations	50

We have taken the H matrix size of 4x8, and generated the random message for encoding and added the noise, transmitted through AWGN channel. At the receiver decoding methods are used then the MATLAB simulation results we got

MATLAB GUI results of the same are as

Figure 4: Simulation of Encoder

ENCODER AND DECODER OF LDPC CODERS

H MATRIX SIZE:

Input Message:

ENCODER:

RECEIVED FROM CHANNEL:

DECODER SUM:

MESSAGE:

Error in DB: 0

Figure 5: Simulation Results of Sum Product Decoder

H MATRIX SIZE:

Input Message:

ENCODER:

RECEIVED FROM CHANNEL:

DECODER BIT FLIP:

MESSAGE:

Error in DB: 3

Figure 6: Simulation Results of Bit Flipping Decoder

H MATRIX SIZE:

Input Message:

ENCODER:

RECEIVED FROM CHANNEL:

DECODER SUM:

MESSAGE:

Error in DB: 0

Figure 7: Simulation Results of Min Sum Decoder

Figure 8: BER Plot of all the Decoding Algorithms

Figure shows BER performance of Hard (BF) vs. Soft Decoders (SPA, MS, OMS). Number of iteration is important consideration in the decoding of LDPC codes, because as with increasing number of iterations, performance of the

decoder will be increases. Sum product decoder shows the behavior as shows in figure the result of decoder for different number of iteration.

Figure 9: BER Performance of Sum Product Algorithm for Iteration i=10

Figure 10: BER Performance of Sum Product Algorithm for Iteration i=30

Figure 11: VHDL Simulation of Encoder

Figure 12: VHDL Simulation of Bit Flipping Decoder

VII. CONCLUSION

In this paper, the simulation performance of different decoding and encoding for LDPC codes is carried out using MATLAB and VHDL implementation. The encoder and hard decision decoder is implemented on Spartan 3E FPGA board. The soft decision Algorithms has good error performance as compared to hard decision algorithms. Further we will try to improve implementation complexity and performance of the encoder and decoders of LDPC codes.

REFERENCES

- [1] R. Gallager, "Low-density parity-check codes", IRE Transactions on information theory, Vol.8, No.1, Pp.21-28, 1962.
- [2] D.J.C. Mackay and R.M. Neal, "Near Shannon limit performance of low density parity check codes," IEE Electron Letters, Vol.32, No.18, 1996.
- [3] IEEE 802.16e. air interface for mobile broadband wireless access systems. ieeep802.16e/d12 draft, oct 2005.
- [4] IEEE P802.3an, 10GBASE-T task force. <http://www.ieee802.org/3/>
- [5] T.T.S.I. digital video broadcasting (DVB) second generation framing structure for broadband satellite applications. <http://www.dvb.org>.
- [6] G.hn/G.9960. next generation standard for wired home network. <http://www.itu.int/ITU-T>.
- [7] M. Rakibul Islam and D. Siam Shafiullah, "Optimized Min-Sum Decoding Algorithm for Low Density Parity Check Codes", International Journal of Advanced Computer Science and Applications, Vol.2, No.12, 2011.
- [8] I. Ali, "Bit-Error-Rate(BER) Simulation Using MATLAB", International Journal of Engineering Research and Applications, Vol.3, No.1, 2013.
- [9] S. Lin and J. Daniel and Jr Costello, "Error Control Coding", Pearson/Prentice Hall, 2nd edition, 2004.
- [10] S. J. Johnson and S. R. Weller, "Resolvable 2-designs for regular low-density parity-check codes", IEEE Trans. Commun., Vol. 51, No. 9, Pp. 1413–1419, 2003.
- [11] S. J. Johnson and S. R. Weller, "Constraining LDPC degree distributions for improved error floor performance", IEEE Commun. Letters, 2006.