International Journal of Speech Technology

Performance analysis of short length Low density parity check codes

--Manuscript Draft--

Manuscript Number:	IJST-D-20-00116
Full Title:	Performance analysis of short length Low density parity check codes
Article Type:	Manuscript
Keywords:	LPDC Encoder, PEG, SPEC, EXIT chart
Corresponding Author:	ANAND ANBALAGAN, Ph.D St.Martin's Engineering College,Secunderabad Madurai, N/A INDIA
Corresponding Author Secondary Information:	
Corresponding Author's Institution:	St.Martin's Engineering College,Secunderabad
Corresponding Author's Secondary Institution:	
First Author:	ANAND ANBALAGAN, Ph.D
First Author Secondary Information:	
Order of Authors:	ANAND ANBALAGAN, Ph.D
	SIVAKANNAN SUBRAMANI, Ph.D.,
	C Kamalanathan
	Sunita Panda
Order of Authors Secondary Information:	
Funding Information:	
Abstract:	Outstanding bit error rate LDPC design in waterfall region and error floor region is one of the challenging tasks for the past decade. This chapter, focuses on the design of LDPC encoder with the low error floor and waterfall region of BER with minimum trapping set. Scheduled Progressive Edge-Growth (SPEG) LDPC encoder is used, and the simulation result of density evolution and exit chart are giving the better convergence of LDPC encoder. BER performance in error floor can controlled by minimum trapping set and waterfall region controlled by scheduled PEG LDPC encoder (1000, 500) with code length (n) is less than 600. The girth of the SPEG encoder is 8. SPEG with minimum trapping set will perform well for short length code also and it converges faster than the other PEG encoder.

Performance analysis of short length Low density parity check codes

Dr.Anand Anbalagan[1],

[1]Department of ECE,

St.Martin's Engineering College,

Secundrabad, India

anandtce@gmail.com

Sivakanann Subramani[2],

[2]Department of ECE,

MVJ College of Engineering,

Bangalore,India

sivakannan87@gmail.com

Kamalanathan C[3]

[3]Department of ECE,
GITAM School of Technology,
GITAM University Bangalore,India
,kamalanadhan@gmail.com

Sunita Panda[4]

[4]Department of ECE,
GITAM School of Technology,
GITAM University Bangalore,India
sunita.panda@gitam.edu

Performance analysis of short length Low density parity check codes

Abstract

Outstanding bit error rate LDPC design in waterfall region and error floor region is one of the challenging tasks for the past decade. This chapter, focuses on the design of LDPC encoder with the low error floor and waterfall region of BER with minimum trapping set. Scheduled Progressive Edge-Growth (SPEG) LDPC encoder is used, and the simulation result of density evolution and exit chart are giving the better convergence of LDPC encoder. BER performance in error floor can controlled by minimum trapping set and waterfall region controlled by scheduled PEG LDPC encoder (1000, 500) with code length (n) is less than 600. The girth of the SPEG encoder is 8. SPEG with minimum trapping set will perform well for short length code also and it converges faster than the other PEG encoder.

***Keyword*s**: LPDC Encoder, PEG, SPEC, EXIT chart

INTRODUCTION

LDPC codes are celebrated for its Shannon capacity and easy implementation in ASIC and FPGA. Formerly, the BER performance of LDPC code depended only on the decoding mechanism. But now the implementation of the proper encoder will increase the performance of BER in both error floor and waterfall region. Most of the researches (Richardson 2003, Tao Tian, Chris Jones et al. 2010, Tao Tian et al 2010, Zheng et al. 2010) have focused only on error floor performance. LDPC ensembles design has lot of techniques such as Quasi-Cyclic (QC) LPDC, Prototype, and PEG with regular and irregular constructions. QCLDPC construction with easy implementation with a shift register and its permutation matrix (Z) plays a vital role in the design of QCLDPC (Wang et al. 2013). Prototype and lifting prototype mechanism yields good ensemble construction with fast convergence even if better in irregular LDPC. But PEG technique has a lot of flexibility of designing of ensembles in regular and irregular with high girth. If the girth of the LDPC is maximum, then its stopping set size will decrease (Gholami&Esmaeili 2012). Hence PEG with large girth will eliminate the problem of stopping set in the tanner graph. In this research work will enrich the ensemble design of PEG LDPC is enhanced in two ways. (i) Scheduled PEG directed towards fast convergence (ii) PEG mechanism growth in the fashion of minimum trapping set.

This paper is organized as follows: In section 1, basic construction about PEG and then section 2 describes the modification of PEG algorithm in a different way. Section 3 describes the minimal trapping set. Section 4 explains the proposed SPEG with minimum trapping set algorithm and section 5 and 6 are dedicated to optimization of degree of distribution using density evolution and EXIT chart analysis, respectively. Section 7 discusses the simulation results in AWGN with various PEG LDPC constructions and then conclusion.

1.PEG LDPC ENSEMBLE CONSTRUCTION

LDPC ensembles can be represented by tanner graph edges and nodes as (V, C, E), where V= $\{v_1, v_2, .. v_n\}$ is the set of variable nodes(VN), C= $\{c_1, c_2, .. c_n\}$ is the set of check nodes (CN) and E⊆ VXC is the set of edges. The edges are placed in the graph one by one, by processing one VN socket at a time. At the end of the process, a bi-section is established between the VN sockets and CN (check node) sockets. This class of algorithms is known as the class of progressive edge-growth (PEG) algorithms. The PEG algorithm is suited to construct the unstructured finite length LDPC code with large girth. (Khazraie et al. 2012)

The motivation behind the PEG algorithm is to tackle the problem of increasing the girth of a Tanner graph by maximizing the local girth of a VN whenever a new edge is drawn from this VN toward the CN set. The PEG algorithm works for any number of VNs and CNs, and for any VN degree distribution. Therefore, it is extremely flexible. For an irregular VN degree profile, ordering the VNs according to their degrees from the smallest to the largest and processing the VNs according to this ordering is in general beneficial.

PEG algorithm node by node manner is summarized as follows (DejanVukobratovi&VojinSenk 2009)

Algorithm: 1

For j= 1 to n do

For k=1 to d_{vj} do

Determine$\boldsymbol{C_{vj}} \in E$

$C_i \leftarrow \{\boldsymbol{C_{vj}} |\ \text{mindeg}\}$

Add edge (V_j,C_j) to E

End for

End for

D_s= $\{d_{v1}, d_{v2,} \ldots\ldots d_{vn} | d_{v1} \leq d_{v1} \leq \cdots \leq d_{sn}\}$, D_s the target sequence of the variable node degrees is sorted in non-decreasing order. It denotes that $\boldsymbol{C_{vj}}$the set of check node, whose distance V_j is maximum. If $E_{\boldsymbol{vj}} \neq \emptyset$, $\boldsymbol{C_{vj}}$it can be determined by expanding a sub graph from variable node V_j up to maximal length. Finally, it is observed that the check node degree distribution of the constructed Tanner graph is almost uniform. Finite length LDPC codes are characterized by a good compromise between waterfall and error floor performances. But finite length codes are not providing good error floor. Hence some modifications are made in PEG algorithm for good error floor without sacrificing waterfall region.

Hence the improvisation of PEG done by (i) degree-by-degree manner (ii) minimizing the number cycles created (Aditya Ramamoorthy& Richard Wesel 2004);

(DejanVukobratovi´C&VojinSenk 2009) (iii) minimizing the approximate cycle extrinsic(ACE) message degree (iv) PEG produce LDPC code graphs with significantly larger minimal stopping set compared with random construction algorithm. Comparing the method (iii) and (iv) the minimal trapping set, performs well in error floor region. But finding trapping set from Tanner graph is NP-hard problem. The modified PEG by ACE (Xiao &Banihashemi 2004) and degree-by-degree with scheduling method (Lam Pham Sy et al. 2011) will be discussed in section 2

2.MODIFIED PEG ALGORITHM

One of the key metrics that have been successfully adopted to improve the original PEG is referred as the approximated cycle extrinsic message degree (ACE) of cycles of Tanner graph. The edges of VN $V_j$are indexed from 0 to d_{vj} -1, and the Kth edge of VN V_j is denoted by $\boldsymbol{e}_{Vj}^{k}$ where k ∈{0,….d_{vj} -1}. Moreover, the neighborhood of VN V_j within the depth *l* is denoted by, $\boldsymbol{N}_{Vj}^{l}$. Denoting by $\boldsymbol{p}_{Vj,c}^{l}$ is the set of paths of length 2 *l*+1 from V_j to c ∈ $\boldsymbol{C}_{Vj}^{l}$

Algorithm:2 Modified PEG with ACE

1: If $\left|\boldsymbol{N}_{Vj}^{lmax+1}\right| = \left|\boldsymbol{N}_{Vj}^{lmax}\right| < m$ then

2: set$\boldsymbol{e}_{Vj}^{k}$=(C_i, V_j)

End

Else

3: determine the ACE of $\boldsymbol{p}_{Vj,c}^{lmax+1}$

Do: 2 until lowest degree of $\boldsymbol{p}_{Vj,c}^{lmax+1}$

End

The above PEG with ACE algorithm gives better error floor performance by giving a penalty of waterfall region. Therefore, scheduled PEG degree-by-degree (Sharon &Litsyn 2008) (DejanVukobratovi´C&VojinSenk 2009) is formed to overcome the tradeoff between waterfall region and error floor region.

Scheduled Progressive Edge growth algorithm is proposed to improve the average inefficiency ($\bar{\mu}$) of irregular LDPC. The ensemble of irregular LDPC can be represented by fraction on node and edge. Let δ_d and γ_d are the fraction on variable node and check node of degree d. Let also λ_d and ρ_d are the fraction of edges connected to variable and check node of degree d. π is the random permutation matrix.

$$\lambda(x) = \sum_d \lambda_d \, x^{d-1}, \rho(x) = \sum_d \rho_d \, x^{d-1} \quad (1)$$

$$\delta(x) = \sum_d \delta_d \, x^d, \lambda(x) = \sum_d \gamma_d \, x^d \quad (2)$$

Code rate (r) =1-$\frac{\int_0^1 \rho(x)dx}{\int_0^1 \lambda\,(x)dx}$ (3)

And coding inefficiency $\mu(\pi)$=$\frac{K_\pi}{K}$ (4)

It is assumed that the girth of the graph goes infinity with the codeword n, which actually happens for almost all the codes of irregular ensemble E(λ, ρ). It follows the decoding efficiency, which can be expressed as $\frac{1-p_e}{r}$ also goes to threshold value:

μ_{th}=$\frac{1-p_{th}}{r}$ which will be referred as inefficiency threshold. The density evolution of the irregular LDPC can be found by tracking the threshold with the ensemble E (λ, ρ). We consider the collection of discrete variable node subset $\boldsymbol{v}_d^{(t)}$ ⊆ V is indexed by

t ∈ {1, 2,….T} and d ∈ {1,2 ... d_{max}}

$$\boldsymbol{v}_d^{(t)} \subseteq V_d \quad (5)$$

$$V=\bigcup_{d=1}^{d_{max}} \bigcup_{t=1}^{T} \boldsymbol{v}_d^{(t)} \quad (6)$$

and $\boldsymbol{n}_d^{(t)}$ the number of variable node in $\boldsymbol{v}_d^{(t)}$

$$n = \sum_{d=1}^{d_{max}} \sum_{t=0}^{T} \boldsymbol{n}_d^{(t)} \quad (7)$$

Algorithm: 3 Scheduled PEG

For t=1 to T do

For d=1 to d_{max} do

For k=1 to d do

For $V_j \in \boldsymbol{v}_d^{(t)}$ do

Determine$\boldsymbol{C}_{\boldsymbol{vj}} \in E$

$C_i \leftarrow \{\boldsymbol{C}_{\boldsymbol{vj}} |$ mindeg$\}$

Add edge (V_j,C_j) to E

End for

End for

End for

End for

Different choice of scheduling subset $\{\boldsymbol{v}_d^{(t)}\}$ might lead to codes with different performance. Even though the set optimized there should be the penalty for waterfall region at some extent. SPEG providing good performance with error floor and waterfall region compared with ACE, but exact calculation of scheduling subset is random distribution, so that, it focused on the scheduled PEG with avoiding minimal trapping set which yields good result in both domain. Next section 3 the trapping set calculation under AWGN channel will be discussed

3. MINIMAL TRAPPING SET OF IRREGULAR CODE

A trapping set for an iterative decoding algorithm is defined as a non-empty set of variable nodes that are not eventually correct by the decoder. (Nguyen et al. 2012). A trapping set is called an T(a,b) trapping set if it contains variable nodes and the subgraph induced by these nodes has b odd degree check nodes. T (a,b) is the subset of V, the set of variable nodes in T are connected to T at least twice .

The size of stopping set T is defined as the cardinality of T. From the Figure 1 set $\{v_2, v_6, V_9\}$ is a stopping set. It shown in Figure 1 that the set of erasures are remaining when the iterative erasure decoding algorithm stops until the erroneous value is equal to the unique maximum stopping set.

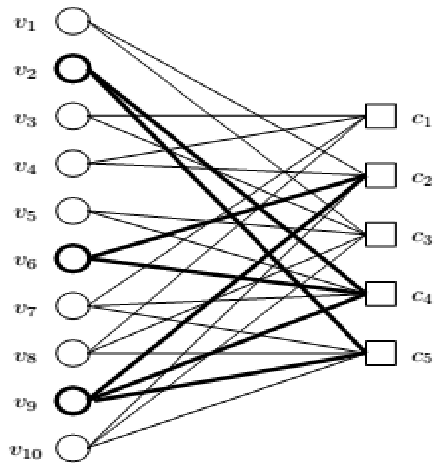

Figure 1 An irregular LDPC code

$$d_c(\mathrm{i,j}) = \begin{cases} |i-j| for\ i,j \leq n_2 + 1 \\ \infty\ Otherwise \end{cases} \quad (8)$$

n_2 is the number of degree-2 variable nodes.

In order to identify the non-selectable CNs a subgraph from VN $V_j$should be spread up to depth 2 (Richter & Hof 2006). The modification on algorithm 1 with minimal trapping set is reinforced by further condition of non-selectability on the surviving CNs in $\boldsymbol{N}_{Vj}^{lmax}$. By satisfying the Equation (8) it can build PEG avoidance of even small trapping set. This will provide great performance in error floor region. Next, section 4 will give the notion of proposed work performance in waterfall region and error floor region.

4. PROPOSED SPEG WITH AVOIDANCE OF SMALL TRAPPING SET

Algorithm 3 and 4 gives the idea of SPEG and small trapping set condition. In SPEG the scheduled parameter $\boldsymbol{v}_d^{(t)}$ calculation is trial and error problem. Then it is optimized by differential evolution method. But still calculation of scheduling parameter is an exhaustive search, due to that the performance of error floor falls with some extent, hence this can be overcome by adapting the idea of avoidance of minimal trapping set with SPEG which will resulting outstanding performance in error floor region without the sacrificing of waterfall region. This notion can be used for regular and irregular PEG LPDC construction. PEG with minimal trapping set algorithm is applicable for BSC and AWGN also. Hence the proposed work gives the universal use of algorithm with various family of PEG.

Algorithm: 4 SPEG with avoidance of minimal trapping set:

For t=1 to T do

    For d=1 to d_{max} do

        For k=1 to d do

For $V_j \in \boldsymbol{v}_d^{(t)}$ do

Determine$\boldsymbol{C_{vj}} \in E$

If

{

Det.distance:$d_c(\mathrm{i},\mathrm{j}) = |i - j| for\ i, j \leq n_2 + 1$

Along the selectable survival path $\boldsymbol{N}_{Vj}^{lmax}$

$C_i \leftarrow \{\boldsymbol{C_{vj}}|\ \mathrm{mindeg}\}$

Add edge (V_j,C_j) to E

}

Else

}

Reject the survival path $\boldsymbol{N}_{Vj}^{lmax}$

$\boldsymbol{N}_{Vj}^{lmax} = \boldsymbol{N}_{Vj}^{lmax} + 1$

}

End for

End for

End for

End for

Algorithm: 4 (Anand&P.Senthil Kumar) degree-by-degree manner is scheduled progressive check node and variable node, but the elimination of trapping set from the small subset level will reduce the untraceable erasure when decoding process. Hence the combination of these two techniques together yields good error floor performance. Coming Simulation results in section 7supports the proposed work

5. DENSITY EVOLUTION

Density evolution(DE) methods are used to find the threshold of the channel for an erased bit. It can be used in Binary Erasure channel (V. Savin, 2008) BEC and Binary input additive white Gaussian (BI-AWGN). The aim of DE to find the erasure probability of decoder which corrects the all erased bit.

5.1Prosperities of Density Evolution (DE)

(i) Symmetric distribution
(ii) All-zeros codeword
(iii) Concentration
(iv) Cycle-free graph

If the channel is symmetric then LLR output by iterative decoder is also symmetric. Due to the symmetric property, independent codeword is transmitted and modeledby all zero codeword. To choose high probability of random code for transmission will increase the ensemble average. If there is no cycle in tanner graph, then the messages are independent of density evolution.

5.2Density Evolution on BEC

For MP in BEC the error bit can correlate only when the messages satisfy the parity check equation.

$$\varepsilon = (1 - p_i) \quad (9)$$

If there is probability of one more message then its probability:

$$q_i = 1 - (1 - p_i)^{w_r - 1} \quad (10)$$

$w_r - 1$is the edge of the check node

w_r &w_c is weight of variable and check node

$$p_{i=}\varepsilon(q_i - 1)^{w_c - 1} \quad (11)$$

$$p_{l=}\ \varepsilon(1 - (1 - p_{l-1})^{w_r - 1})^{w_c - 1} \quad (12)$$

Prior to decoding process the probability of erased bit

$$P_o = \ \varepsilon \quad (13)$$

$$P_1 = \ \varepsilon(q_i - 1)^{w_c - 1} \quad (14)$$

Recursive equation finds the message from the erased bit on BEC.

5.3Ensemble Threshold

For $\varepsilon \in [0,1]$ then the lower bound

$$f(0, \varepsilon) = \varepsilon\lambda(1 - \rho(1) \quad (15)$$

Upper bound

$$f(1, \varepsilon) = \varepsilon\lambda(1 - \rho(1 - 1) = \varepsilon \quad (16)$$

$$0 \leq \ f(\rho, \varepsilon) \leq \varepsilon \quad (17)$$

$P_1(\varepsilon) \rightarrow 0$ then

$$P_1(\varepsilon') \rightarrow 0 \forall\, \varepsilon < \varepsilon' \quad (18)$$

The value of ε is called threshold. So (λ, ρ) can be represented as the supernum of ε for with $p_l(\varepsilon) \rightarrow 0$

$$\varepsilon^*(\lambda, \rho) = \sup \{ \varepsilon \in [0,1] : p_{1_{l \rightarrow \infty}}(\varepsilon) \rightarrow 0 \} \quad (19)$$

The equation (19) ε^* eliminate all the erased probability as zero. Threshold calculation of channel is one of the important tasks when designing the $(\lambda, \rho\ )$ degree distribution of variable and check node.

5.4Density Evolution on BI-AWGN

In BI-AWGN all LLR values are depending upon the probability density function (pdf),

$$P(M_1) = P(R) \otimes \sum_i \lambda_i\, p(E_1)^{\otimes(i-1)} \quad (20)$$

$\otimes$ Convolution operator

$$z = log\frac{f(z)}{f(-z)} \quad (21)$$

$$f(z) = \iint f(x, y) dx\, dy \quad (22)$$

$$\text{If } P\left({}^{L}\!/_{x} = -1\right) = P\left(-{}^{L}\!/_{x} = 1\right) \quad (23)$$

$$P_c = P(L < 0) \quad (24)$$

Equation (23) is used for condition 1 and equation (24) is used for condition zero with the distribution of LLR Vs Density is a Gaussian function with different variance values.

6.EXIT CHART

Exit information transfer (EXIT) gives the visual interpretation of decoder with the help of mutual information of check node and variable node (Schmalen et al. 2011). This can give the information of decoder with in less number of iteration.

EXIT chart can be used as

(i) EXIT can find the threshold whenever the two curves link each other

(ii) It can find the speed of decoding process using the optimized $(\boldsymbol{\lambda}(\boldsymbol{x}), \boldsymbol{\rho}(\boldsymbol{x}))$

(iii) It can be used to generate the ensemble for capacity approaching code.

7 RESULTS AND DISCUSSION

Figure 2 shows that IEEE 802.11 recent exponential non-binary matrix is developed and this matrix is used as the seed matrix of regular quasi-cyclic LDPC code with the each entry corresponding to 27.

Code rate =2/3

Code word length= 608

Circulant matrix size =27.

$$\begin{bmatrix}
0 & -1 & -1 & -1 & 0 & 0 & -1 & -1 & 0 & -1 & -1 & 0 & 1 & 0 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
22 & 0 & -1 & -1 & 17 & -1 & 0 & 0 & 12 & -1 & -1 & -1 & -1 & 0 & 0 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
6 & -1 & 0 & -1 & 10 & -1 & -1 & -1 & 24 & -1 & 0 & -1 & -1 & -1 & 0 & 0 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
2 & -1 & -1 & 0 & 20 & -1 & -1 & -1 & 25 & 0 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
23 & -1 & -1 & -1 & 3 & -1 & -1 & -1 & 0 & -1 & 9 & 11 & -1 & -1 & -1 & -1 & 0 & 0 & -1 & -1 & -1 & -1 & -1 & -1 \\
24 & -1 & 23 & 1 & 17 & -1 & 3 & -1 & 10 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & -1 & -1 & -1 & -1 & -1 \\
25 & -1 & -1 & -1 & 8 & -1 & -1 & -1 & 7 & 18 & -1 & -1 & 0 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & -1 & -1 & -1 & -1 \\
13 & 24 & -1 & -1 & 0 & -1 & 8 & -1 & 6 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & -1 & -1 & -1 \\
7 & 20 & -1 & 16 & 22 & 10 & -1 & -1 & 23 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & -1 & -1 \\
11 & -1 & -1 & -1 & 19 & -1 & -1 & -1 & 13 & -1 & 3 & 17 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & -1 \\
25 & -1 & 8 & -1 & 23 & 18 & -1 & 14 & 9 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 0 & 0 \\
3 & -1 & -1 & -1 & 16 & -1 & -1 & 2 & 25 & 5 & -1 & -1 & 1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 0
\end{bmatrix}$$

Figure 2 802.11-2012 exponential H matrix

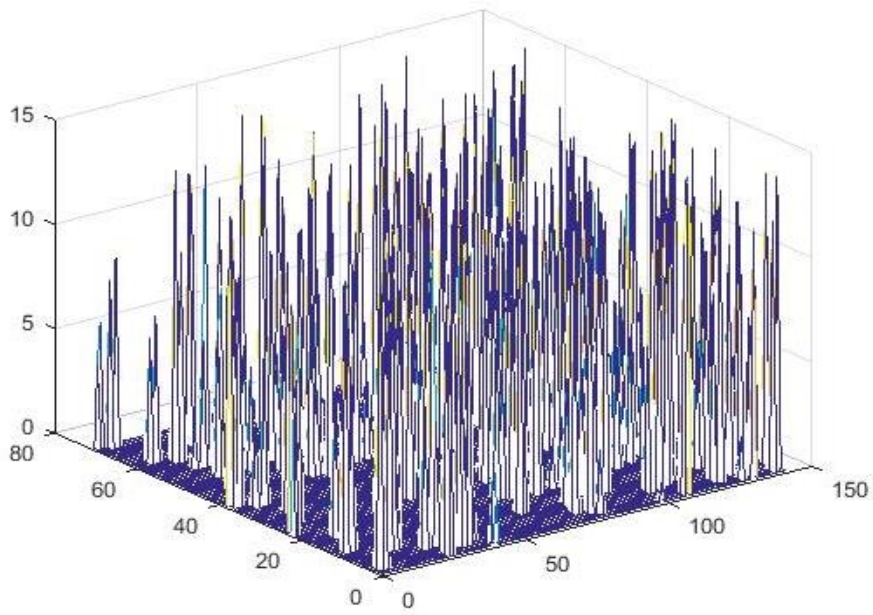

Figure 3 Non-binary PEG Seed matrix size of (150, 78) with the girth of 8

The Non-binary seed matrix PEG is shown in Figure 3 non-binary values are connected as most as 15 number and this irregular non-binary matrix obtained when the 35 th bit node is progressive. The non-zero entries are shown a mesh command in MATLAB.

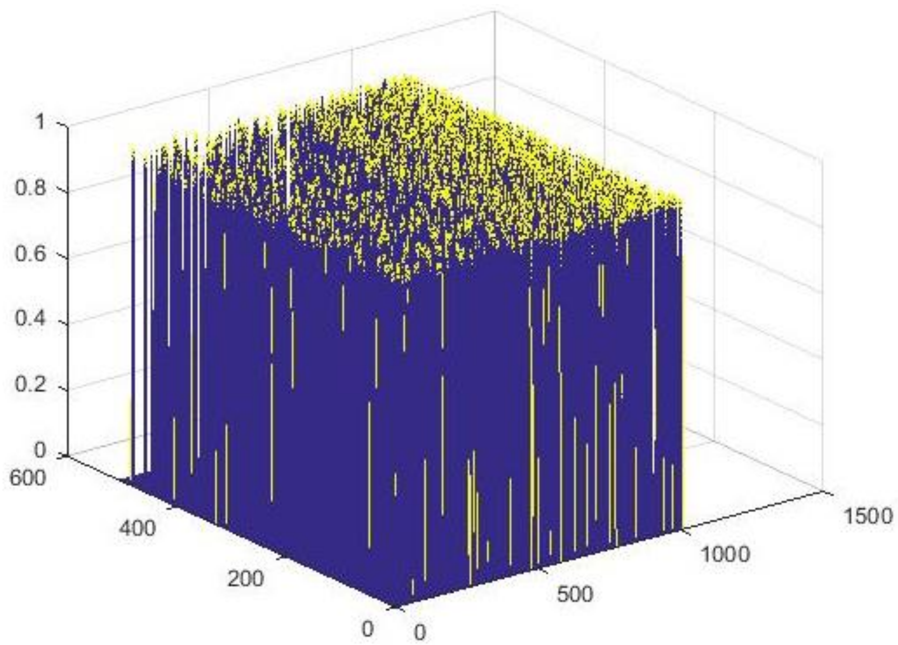

Figure 4 Binary irregular LDPC H matrix by PEG (524,1000)

To differentiate the regular and irregular binary and non-binary Figure 4 shows that the irregular PEG with input codeword size is as 500. The girth of the encoder matrix is 5 only. So, the Non-zero elements are distributed randomly. This encoder matrix illustrates the sparseness of the matrix in the after column of 250 onwards. It gives the information of increasing the matrix size distribution of non-zero which will be sparse.

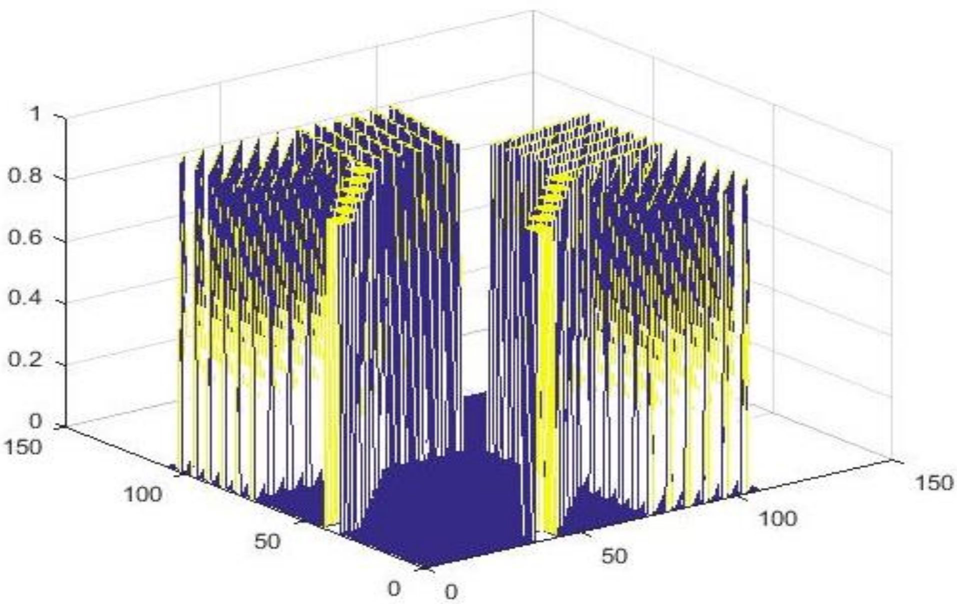

Figure 5 Encoder seed matrix of PEG with Girth of 8 (100,150)

The PEG LDPC encoder matrix with a girth of 8 is simulated and plotted with the size of (100,50) this is starting seed matrix of the PEG at the 50 th node connection progressive finished, and it is grown progressively with the node by node basis upto (560,1000) as in IEEE standard 802.11-2012 standard. The Figure 5 gives the construction of LDPC PEG with a girth of illustrating the layer performance in Encoder matrix. The corresponding H matrix also is shown in Figure 6.

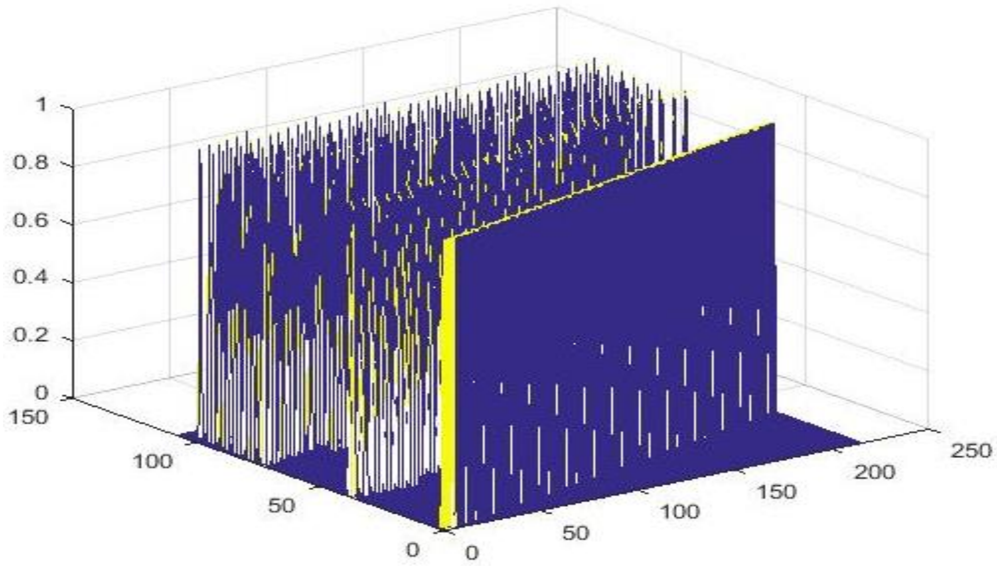

Figure 6 H matrix of PEG with girth of 8 (100,50)

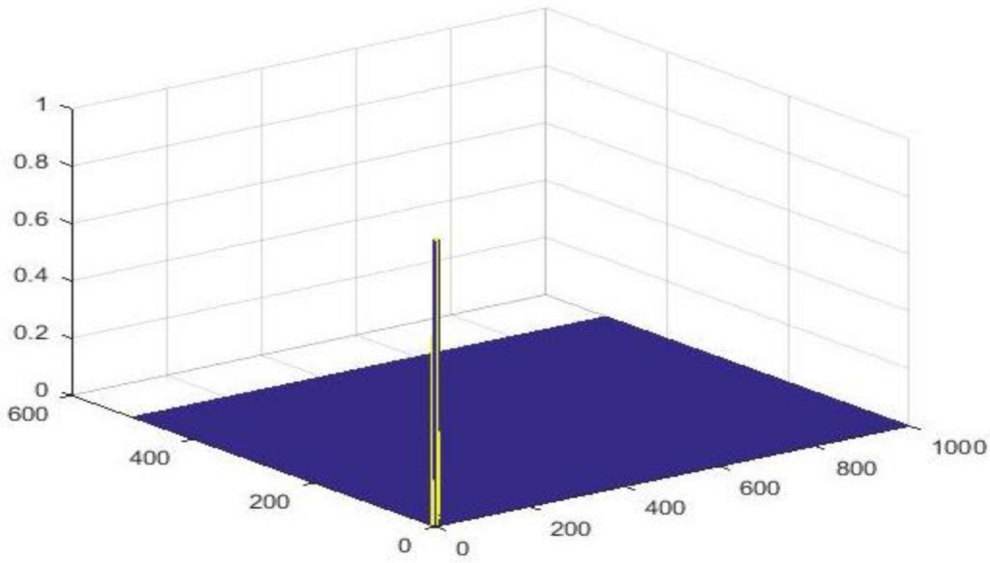

Figure 7 regular QCLDPC by improved PEG (500,1000)

Figure 7 shows that regular PEG LDPC size(500,1000) with the girth of 8 is completed with the full size. The complete matrix has less number of one's near to zero, is illustrated. Only 648 non-zero values are there in the construction which is shown in Figure 8. The main reduction of non-zero values by ACE LDPC encoder (improved PEGLDPC) is also shown here.

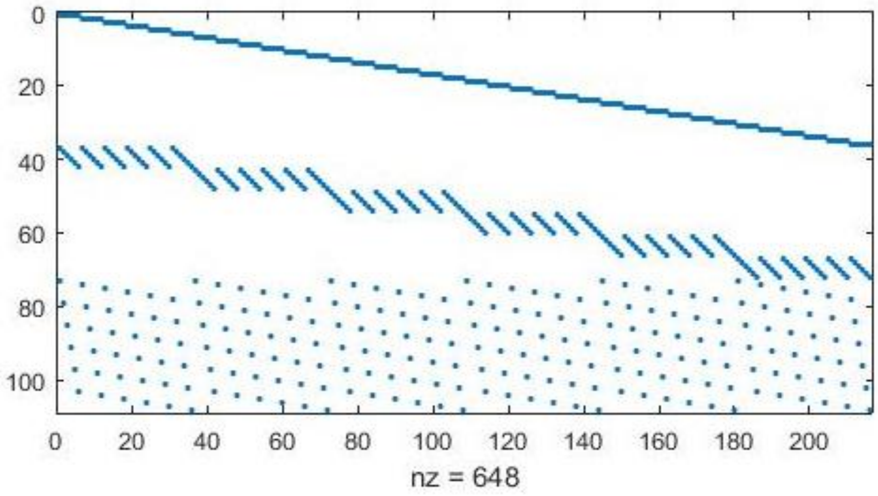

Figure 8 Spy representation of improved PEGLDPC (500, 1000)

Channel asymptotic behaviour can be calculated by density evolution technology. DE is used to calculate the probability of error by MATLAB and the result is shown.

```
chan = zeros(1,6001);
chan(3240) = 91.606;
chan(2762) = 8.394;
ext = [-30 0.01 6001];
mapping = [-10 0.0002 50000];
dv = 3;
dc = 6;
iter = 50;
stop_pe = 1e-5;
```

result of probability error is tabulated corresponding to the iteration.

Table 1 Density evolution analysis

Degree distribution	Threshold	Itertion10	Iteration 20	Iteration 30	Iteration 40	Iteration 50
Variable node degree [0 0.2895 0.3158 0 0 0.3947] Check node degree	0. 023	0.0839	0.0788	0.0792	0.0796	0.0845
	0.45	0.0631	0.0580	0.0543	0.0520	0.0592

[0 0 0 0 0 0.9032 0.0968]						

The degree of distribution of check node and the variable node from the IEEE 802.11-2012 taken into account for calculating threshold of the channel. The threshold value of the channel is the information capacity to the upper bound. But the value of probability error is not converging even when the number of iteration is increased. So it is clear that the threshold value and probability of error depend up on degree of distribution of irregular code check node and variable node. The Density evolution with a posteriori probability will not give a good result for the specific encoding process. This is shown in Table 1 from density evaluation analysis the threshold value is calculated form different iteration. But a priori probability is required information before transmission in many iterative decoding algorithms. So, EXIT (Extrinsic information) of check node, and EXIT chart also give the information of channel threshold by using Intrinsic and Extrinsic information calculation with the logarithmic domain and this will show the clear boundary of channel threshold,

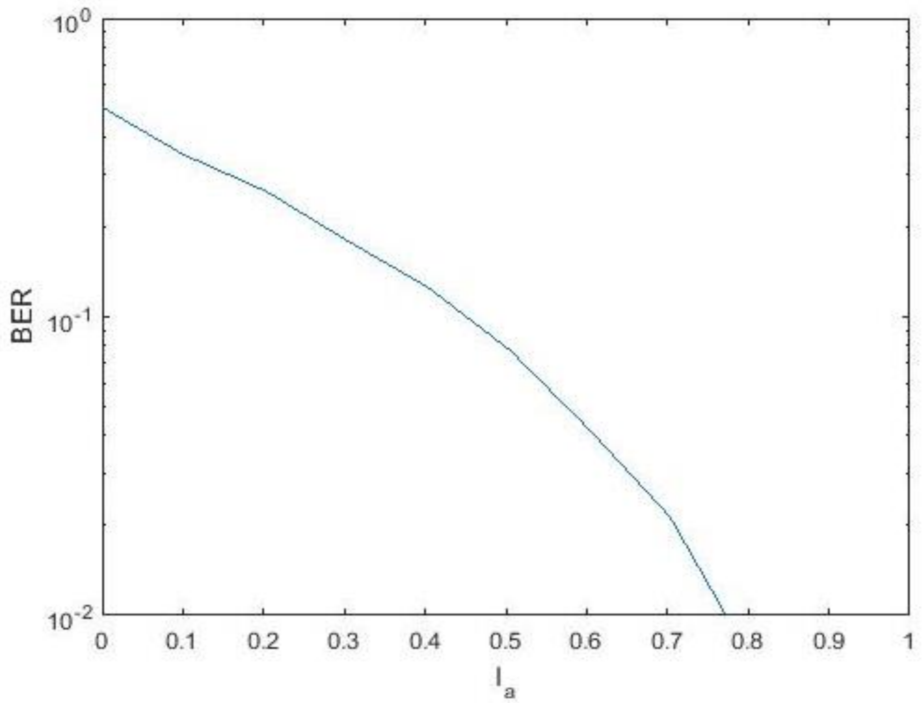

Figure 9 Exit chart for binary LDPC

EXIT chart in Figure 9 gives the information of channel threshold as 0.768 for the BER of 10^-2. This simulation is for codeword length as 500. If we increase the codewordlength upto 1000, it will not take a tremendous changes in the channel threshold values. So, the effect of codeword length is not that much an issues in channel threshold.

So, even the short length also has the same performance as large codeword length. Hence, reducing the codewordlength is not the influence the threshold. From the analysis channel threshold, it is depending upon the probability in density evolution, but in EXIT chart the value of threshold mainly depends upon the Extrinsic and Intrinsic information of check node and a variable node of this system in logarithm domain.

The EXIT chart values are tabulated in Table 1 is shown as NBLDPC extrinsic information transfer for the codeword length as 500 with 11^{th} order polynomial to get better understanding of bit/channel use and the illustration is also shown in Figure 9.

CONCLUSION:

The PEG encoder are already achieve good performance and its produce near to channel capacity. The EXIT chart analysis is supporting that the PEG and SPEG are fast in convergence for short length codes. This because the PEG ensembles are obtained from density evolutions. The behaviour of SPEG encoder is attributed by the trapping set elimination algorithm is added advantage to the encoding process and eliminate infinite loop when it will be decoded. So the SPEG encoder with trapping set elimination process leads good in encoder performance.

Reference

Richardson, TJ, Shokrollahi, MA &Urbanke, RL 2001,'Design of capacity-approaching irregular low-density parity-check codes', IEEE Trans. Inf. Theory, vol. 47, no. 2, pp. 619-637

Tao Tian, Chris Jones, John D Villasenor & Richard D Wesel 2010,'Constructing short-length irregular LDPC codes with low error floor' Communications, IEEE Transactions on vol. 58, no. 10.

Zheng, X, Lau, F &Tse, C 2010,'Constructing short-length irregular LDPC codes with low error floor', IEEE Trans. Commun. vol. 58, no. 10, pp. 2823-2834

Wang, C, Chen, X, Li, Z & Yang, S 2013,'A simplified min-sum decoding algorithm for non-binary LDPC codes', IEEE Trans. Commun. vol. 61, no. 7, pp. 24-32.

Gholami, M &Esmaeili, M 2012, 'Maximum-girth cylinder-type block-circulant LDPC codes', IEEE Trans. Commun. vol. 60, no. 4, pp. 952-962.

Khazraie, S, Asvaldi, R &Banihashemi, A 2012, 'A PEG construction of finite-length LDPC codes with low error floor, IEEE Commun. Lett. vol. 16, no. 8, pp. 1288-1291

(DejanVukobratovi&VojinSenk 2009)

Aditya Ramamoorthy& Richard Wesel, 2004, 'Construction of short block length irregularlow-density parity-check codes', 2004 IEEE International Conference on Communications, Paris, France, vol. 1.

Xiao, H &Banihashemi, A 2004, ‘Improved progressive-edge-growth (PEG) construction of irregular LDPC codes, IEEE Commun. Lett. vol. 8, no. 12, pp. 715-717

Sharon, E &Litsyn, S 2008, ‘Constructing LDPC codes by error minimization progressive edge growth’, IEEE Trans. Commun., vol. 56, no. 3, pp. 359-368

Savin, V 2007, ‘Iterative LDPC decoding using neighborhood reliabilities’, in: IEEE International Symposium on Information Theory (ISIT), pp. 221-225.

Schmalen, L, Adrat, M, Clevorn, T & Vary, P 2011, ‘EXIT chart based system design for iterative source-channel decoding with fixed-length codes’, IEEE Trans. Commun. vol. 59, no. 9.

Sharon, E, Ashikhmin, A &Litsyn, S 2006, ‘EXIT functions for binary input memoryless symmetric channels’, IEEE Trans. Commun. vol. 54, no. 7, pp. 1207-1214.

Anand, A&Senthil Kumar, P 2015, ‘An efficient non-binary LDPC decoder with layer dynamic scheduling’, J. Comput. Theor. Nanosci, vol. 12, no. 12, pp. 5066-5070

Anand, A&Senthil Kumar, P 2016, ‘Scheduled progressive edge growth ldpc encoder with mimimal trapping set’, Asian journal of information technology, vol. 15, no. 13, pp. 2162-2166

Healy, C &Lamare, R 2012, ‘Decoder-optimized progressive edge growth algorithms for the design of LDPC codes with low error floors’, IEEE Commun. Lett., vol. 16, no. 6, pp. 889-892.

Jesus Martines-Mateo, David Elkouss&Vicenetemartin 2011, ‘Improved construction of irregular progressive edge growth tanner graph’, arXiv:1007.3588v2[cs.IT]

Declercq, D &Fossorier, M 2016, ‘Channel coding theory, Algorithms and applications’.

Tian, T, Jones, C, Villasenor, J & Wesel, R 2004, ‘Selective avoidance of cycles in irregular LDPC code constructions’, IEEE Trans. Commun. vol. 52, no. 8, pp. 1242-1247

Nguyen, DV &Vasic, B 2014‘Two-bit bit flipping algorithms for LDPC codes and collective error correction’, IEEE Trans. Comm.

Performance analysis of short length Low density parity check codes

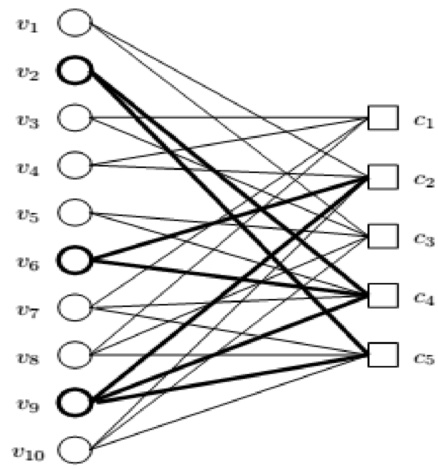

Figure 1 An irregular LDPC code

$$\begin{bmatrix}
0 & -1 & -1 & -1 & 0 & 0 & -1 & -1 & 0 & -1 & -1 & 0 & 1 & 0 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
22 & 0 & -1 & -1 & 17 & -1 & 0 & 0 & 12 & -1 & -1 & -1 & -1 & 0 & 0 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
6 & -1 & 0 & -1 & 10 & -1 & -1 & -1 & 24 & -1 & 0 & -1 & -1 & -1 & 0 & 0 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
2 & -1 & -1 & 0 & 20 & -1 & -1 & -1 & 25 & 0 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
23 & -1 & -1 & -1 & 3 & -1 & -1 & -1 & 0 & -1 & 9 & 11 & -1 & -1 & -1 & -1 & 0 & 0 & -1 & -1 & -1 & -1 & -1 & -1 \\
24 & -1 & 23 & 1 & 17 & -1 & 3 & -1 & 10 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & -1 & -1 & -1 & -1 & -1 \\
25 & -1 & -1 & -1 & 8 & -1 & -1 & -1 & 7 & 18 & -1 & -1 & 0 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & -1 & -1 & -1 & -1 \\
13 & 24 & -1 & -1 & 0 & -1 & 8 & -1 & 6 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & -1 & -1 & -1 \\
7 & 20 & -1 & 16 & 22 & 10 & -1 & -1 & 23 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & -1 & -1 \\
11 & -1 & -1 & -1 & 19 & -1 & -1 & -1 & 13 & -1 & 3 & 17 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & -1 \\
25 & -1 & 8 & -1 & 23 & 18 & -1 & 14 & 9 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 0 & 0 \\
3 & -1 & -1 & -1 & 16 & -1 & -1 & 2 & 25 & 5 & -1 & -1 & 1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 0
\end{bmatrix}$$

Figure 2 802.11-2012 exponential H matrix

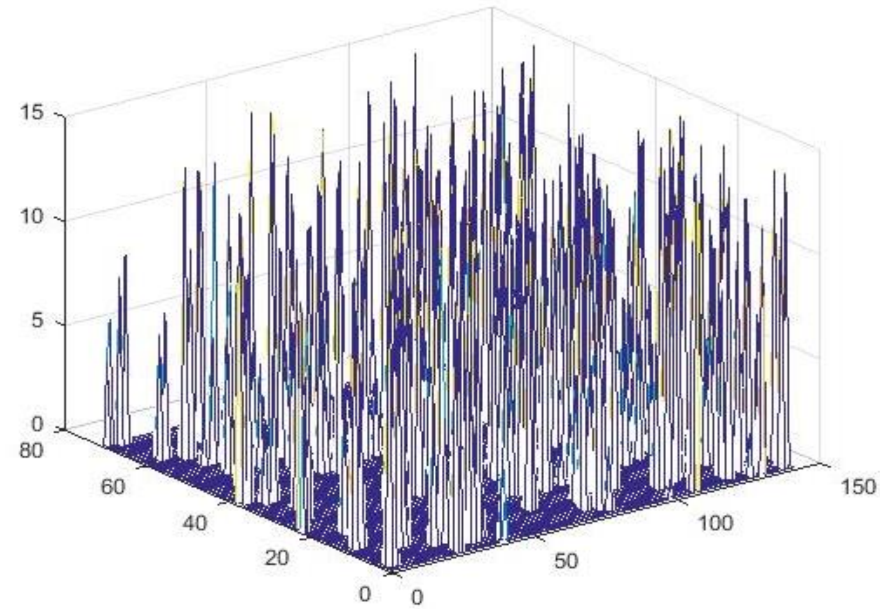

Figure 3 Non-binary PEG Seed matrix size of (150, 78) with the girth of 8

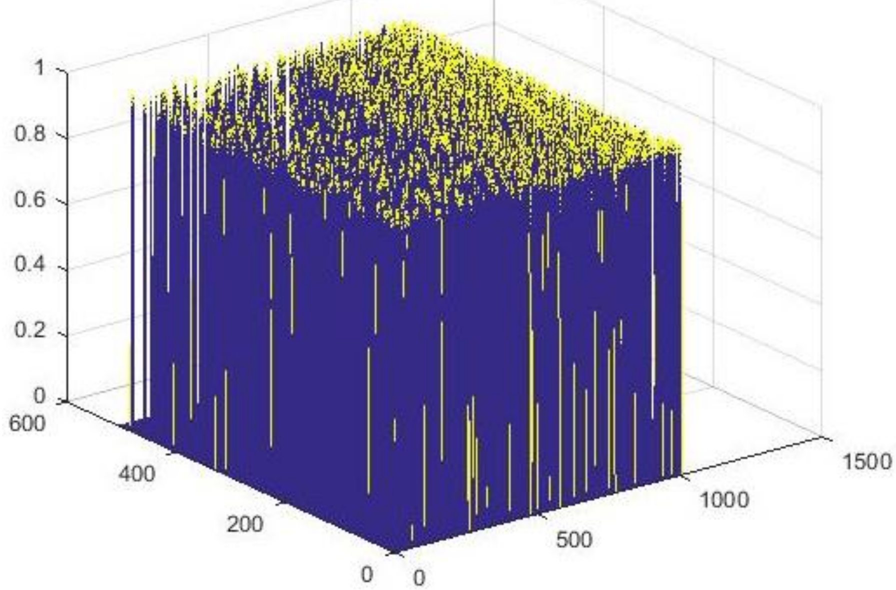

Figure 4 Binary irregular LDPC H matrix by PEG (524,1000)

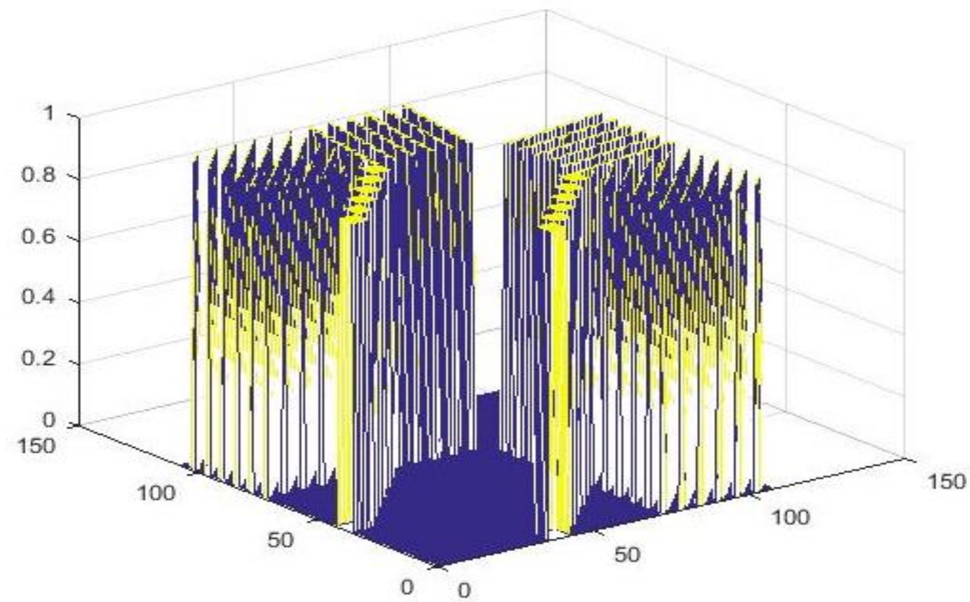

Figure 5 Encoder seed matrix of PEG with Girth of 8 (100,150)

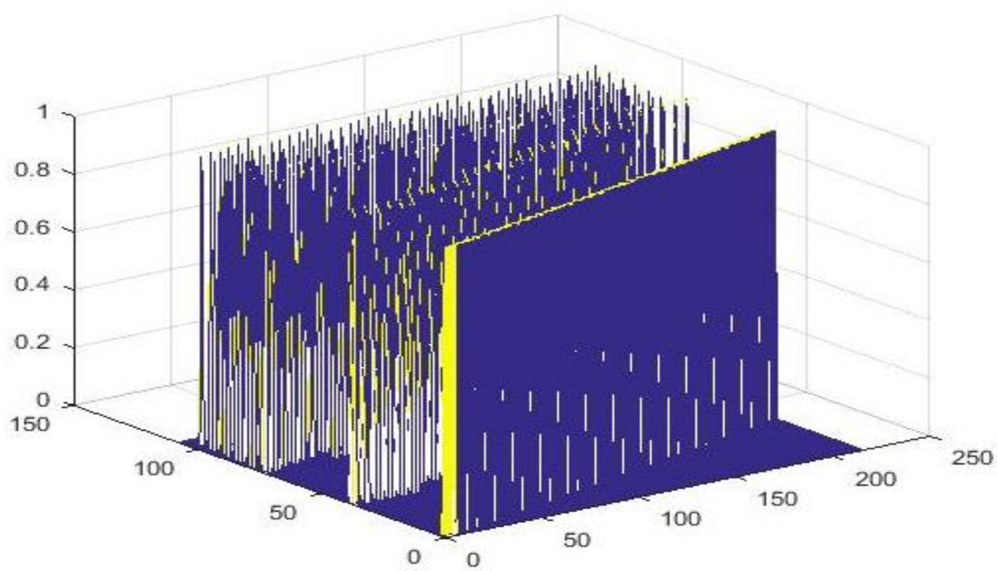

Figure 6 H matrix of PEG with girth of 8 (100,50)

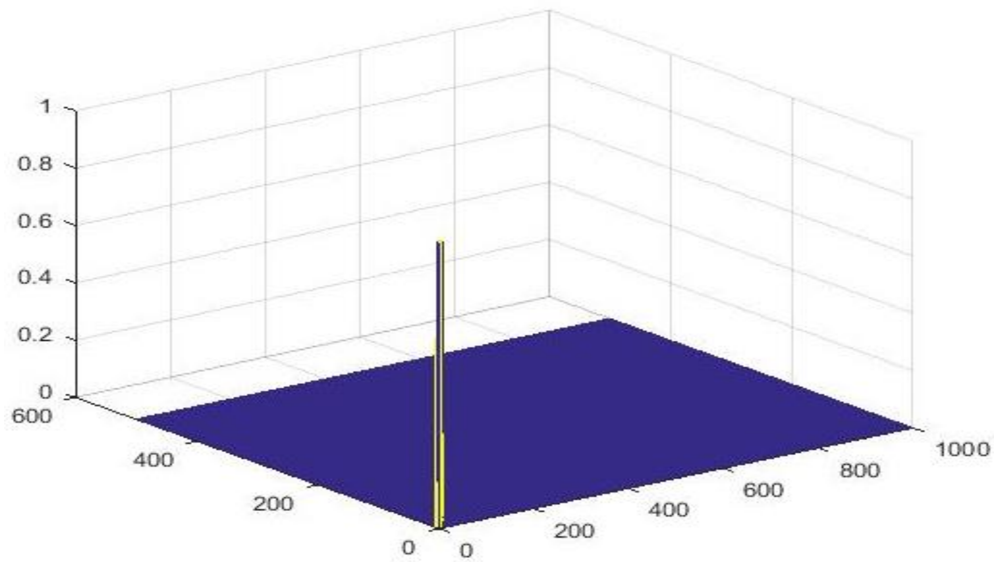

Figure 7 regular QCLDPC by improved PEG (500,1000)

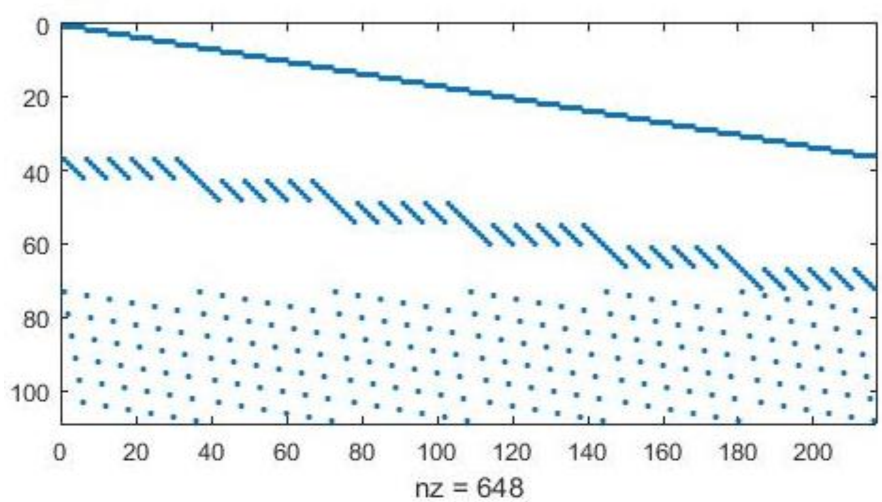

Figure 8 Spy representation of improved PEGLDPC (500, 1000)

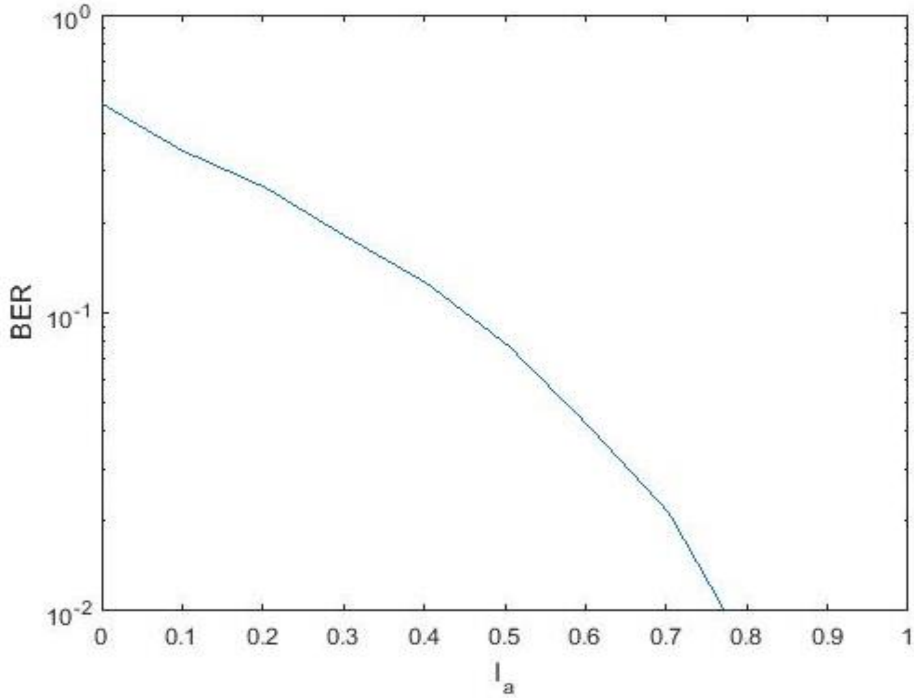

Figure 9 Exit chart for binary LDPC

Performance analysis of short length Low density parity check codes

Table 1 Density evolution analysis

Degree distribution	Threshold	Itertion10	Iteration 20	Iteration 30	Iteration 40	Iteration 50
Variable node degree [0 0.2895 0.3158 0 0 0.3947] Check node degree [0 0 0 0 0 0.9032 0.0968]	0. 023	0.0839	0.0788	0.0792	0.0796	0.0845
	0.45	0.0631	0.0580	0.0543	0.0520	0.0592