

Cellular Automata-Based LDPC Decoder

C. Abisha Queen, M. Anbuselvi and S. Salivahanan

Abstract Low-density parity check (LDPC) codes are the capacity approaching codes having better decoding performance closer to Shannon's limit. The performance of the LDPC codes depends on the block length, code rate, structure of parity check matrix (H-matrix), and on the decoding process. Various code construction methods are structured including Quasi-cyclic-irregular parity check matrix to improve the performance of the LDPC codes. Cellular Automata are a computational method that realizes the complex computational blocks into simple, regular, and modular structures. In this paper, the cellular automata-based LDPC parity check matrix with hierarchical diagonal parity check matrix structure has been incorporated in the decoder design. Error performance improvement of 0.0417 dB is obtained with HDPCM-based LDPC decoder using cellular automata. Performance analysis on increased code length validates the proposed LDPC decoder with the improved decoding performance.

Keywords LDPC codes · Decoding algorithm · Cellular automata · Parity check matrix · Density · Girth

C. Abisha Queen (✉) · M. Anbuselvi
ECE Department, SSN College of Engineering, Kalavakkam, India
e-mail: abiqueenrec@gmail.com

M. Anbuselvi
e-mail: anbuselvim@ssn.edu.in

S. Salivahanan
SSN College of Engineering, Kalavakkam, India
e-mail: salivahanans@ssn.edu.in

1 Introduction

Low-density parity check codes are the class of linear block codes with sparse parity check matrix (H-matrix) [1, 2]. The advantages of LDPC codes are highly parallelizable in hardware implementation; codes are capacity approaching, efficiently decoded by parallel iterative decoding algorithm with low latency and these advantages allow LDPC codes to real-time and high-throughput application [3]. LDPC codes find application in standards such as 10GBase-T Ethernet, IEEE 802.11(WiFi), WiMAX, data storage in flash memory and Digital Video Broadcasting (DVB). LDPC decoding algorithms include functions in computation which gives complex computational blocks structures in VLSI implementation [4]. The parity check matrix gives the parity check constraints for the LDPC codes. The structure of the parity check matrix is one of the metric which determines the performance of the LDPC codes. Quasi-cyclic LDPC codes provides decoding with fast convergence rate, low error-floor, and better performance in the AWGN and BEC channel and it is close to Shannon limit [5]. Various algorithmic modifications are provided to reduce the complexity in the implementation at the cost of degradation in the performance [4, 6, 7].

The cellular automata implements the complex computational blocks into simple modular logic structures [8]. Cellular Automata (CA) is the computational method for various VLSI applications which includes error correcting codes [9, 10], design of cipher system [11], testing of circuits, authentication scheme, and compression. Cellular automata implementation provides better parallelism and it is also cost effective compared to DSP processor. Cellular automata-based LDPC structure is obtained using the cellular automata rules, which includes the modular expression for generation of the hierarchical diagonal parity check matrix.

In this paper, cellular automata-based LDPC codes have been proposed to provide better performance and reduced computational complexity. This paper includes basic concepts of cellular automata, classification of rules, and cellular automata rules for HDPCM generation.

2 Cellular Automata Concept and Classification

Cellular automata are a computational method which implements complex computational blocks into simple modular logic structures. Cellular automata (CA) are attractive due to their fine grain parallelism, simple computational structures, and local communication patterns [12]. CA consist of an infinite number of finite cells arranged on a regular lattice. Each cell in the lattice is identical and works simultaneously. The value in each cell in time $t + 1$ is a function of the value of cell in time t , the value of neighboring cells in time t , and rule associated with the cell. One-dimensional cellular automata consist of a row of cells with $r = n$ in general representation, where r is the distance from the present cell to the neighborhood

cells to which the value of the current cells depend on and n is the number of cells in the row, $n = 1$ gives $y_{i-1}(t)$, $y_i(t)$, $y_{i+1}(t)$ as the neighborhood cells. In one-dimensional cellular automata [13], the value of the cell $y_i(t+1)$ depends on the neighborhood cells $y_{i-1}(t)$, $y_i(t)$, and $y_{i+1}(t)$. $y_{i-1}(t)$ corresponds to the cell to the left of the current cell and $y_{i+1}(t)$ corresponds to the cell to the right of the current cell. The value of the present cell depends on the neighborhood cells previous values that results in 8 (i.e., 2^3) possible neighborhood combinations. The possible number of rules is 256 (i.e., 2^8).

3 Cellular Automata Rules for LDPC Decoding

Parity check matrix (PCM) is the random matrix defined in the Galois Field which determines the strength of the LDPC codes. The complexity in the computations of LDPC codes depend on the number of nonzero numbers and structure of the PCM. By increasing the sparsity of the PCM, the computational complexity is reduced significantly. The increased sparsity in the PCM is achieved by implementing the cellular automata rules. The increased sparsity of the matrix is required to reduce the complexity in the computation of LDPC codes in the implementation of the decoding algorithm. The CA-based generated PCM is sparser compared to the random PCM. PCM obtained by rule 90 is sparser compared to the random or other two CA rules. The rule 90 has a less complex function compared to all the other random CA rules.

Rule 90: A simple linear CA rule generates sparse matrix by the exclusive-or of its two neighbors [13].

$$y_i(t+1) = y_{i-1}(t) \oplus y_{i+1}(t)$$

Rule 240: Shift-right operations are performed [13].

$$y_i(t+1) = y_{i-1}(t)$$

3.1 Comparison of PCM Generated by Using Cellular Automata Rules

The sparsity comparison between the random LDPC PCM and PCM generated by the cellular automata rules are shown in Table 1. The sparse arity check matrix PCM was obtained by the Rule 90 with simple function for implementation.

The random number generation using nonuniform CA randomizer has been found to be superior to that of Rule 30 [14] and takes the same time as that of uniform cellular automata rules. The hierarchical diagonal parity check matrix is

Table 1 Comparison of CA rules

<p>Random LDPC PCM</p> $H = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \end{bmatrix}$	<p>PCM using CA Rule 30</p> $H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 \end{bmatrix}$
<p>PCM using CA Rule 90</p> $H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}$	<p>PCM using CA Rule 150</p> $H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 \end{bmatrix}$

implemented in cellular automata by concatenating Rule 240 and Rule 90, and are represented by

$$H_{HDPCM} = [RM_{S,H,L}|IM]$$

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

where $RM_{S,H,L}$, the single diagonal elements along with the diagonal elements both in the upper right half and lower left half of the matrix and IM, the identity matrix.

3.2 Algorithm to Generate CA-Based LDPC Codes

See Table 2.

4 LDPC Decoding Algorithm

Sum-Product decoding algorithm (SPA) of LDPC codes approach the channel capacity in the additive white Gaussian noise channel. The computational complexity of the SPA is given by $O(q)$ as it has sum and product operations for each

Table 2 Algorithm to generate CA based LDPC codes

<p>Initially the $H(m,n)$ matrix is generated by rule 240 to the first half matrix and second half matrix by specifying the initial conditions as $H(1,1)=H(1,(n/2+1))=1$.</p> <p>STEP: 1.a, Initialization for first row to apply CA rule</p> <pre> i=1 for j=1:1:n/2 If j==1 H(1,1)=1; else H(i,j)=0; end end b, Rule 240 for i=2:1:m for j=1:1:n/2 if j==1 H(i,j)=H(i-1,n/2); else H(i,j)=H(i-1,j-1); end end c, similar is done for j=n/2+1:1:n by applying the rule 240. STEP: 2. Splitting the H-matrix as H_1-matrix and H_2-matrix. STEP: 3. For H_1-matrix the rule 240 is applied by specifying $H_1(1,(n/4+1))=H_1((m/2+1),1)=1$. </pre>	<p>STEP: 4. Random matrix $H_R(m,n/2)$ is generated by using Rule 90.</p> <pre> i=1; j=1; for j=2:1:n/2 H_R(1,1)=1; H_R(i,j)=0; end for i=2:1:m for j=1:1:n/2 if j==1 H_R(1,j)=xor((H_R(i-1,n),H_R(i-1,j+1))); elseif j==n H_R(1,j)=xor(H_R(i-1,j-1),H_R(i-1,1)); else H_R(1,j)=xor(H_R(i-1,j-1),xor(H_R(i-1,j+1))); end end end STEP: 5. Matrix H_{12} is obtained by OR-operating the matrix H_R and H_1. $H_{12}=H_R \text{ or } H_1$; STEP: 6. Final HDM matrix is obtained by concatenating H_{12} and H_2. </pre>
---	--

check node processing, where q represents the cardinality of the Galois field. The SPA can also be implemented in the probability domain using fast fourier transforms (FFT) [15] and the complexity will be dominated by $O(q \log_2 q)$ as it includes sum and product operations. The FFT-based SPA (FFT-SPA) needs complicated multiplications. The hardware expensive multiplications are removed by the exponential and logarithm operation with look-up tables during the check node and variable node processing. In the max-log-SPA, the computational

complexity is given by $O(q^2)$ with sum and comparisons. Variations in the algorithm is done to reduce the complexity that includes min-sum decoding algorithm which provides reduced memory requirement and with little performance loss. Min-max decoding algorithm gives reduced number of comparisons and it degrades the performance. To provide better trade-off between computational complexity and performance, the proposed method of cellular automata-based LDPC parity check matrix is generated for the FFT-SPA decoding using the cellular automata rules. The computational complexity is reduced by the cellular automata as it implements the complex computational block to simple modular logic structures.

5 Hierarchical Diagonal Parity Check Matrix in CA-Based LDPC Codes

In the cellular automata-based LDPC codes, the hierarchical diagonal parity check matrix (HDPCM) is derived using CA rules. The information bits are encoded by using the generation matrix and modulated by BPSK modulation. The modulated information is transmitted through the channel. The Additive White Gaussian Noise (AWGN) is added to the transmitted information. The channel probability for the channel and the received bits are calculated and decoded using FFT-SPA. FFT-SPA decoding algorithm was implemented with the HDPCM derived by the CA rules as shown in Table 2. The computational complexity of the sum-product algorithm (SPA) is $O(q^2)$ and by the FFT-SPA it scales down to $O(q \log q)$.

For the code length 648, the BER for different SNR, noise variance (σ), and comparison between random LDPC codes HDPCM matrix and CA-based LDPC codes HDPCM matrix is plotted.

5.1 Analysis of HDPCM

Sparsity of the PCM is the ratio of total number of zero elements to the total number of elements in the parity check matrix structure. The sparsity of the HDPCM matrix for code length $n = 648$ is 0.9698 and for $n = 1008$ the sparsity is 0.9729.

Density of the PCM is the ratio of total numbers of nonzero elements to the total number of elements in the parity check matrix structure. The density of the HDPCM matrix for code length $n = 648$ is 0.0302 and for $n = 1008$ the density is 0.0271.

A cycle of length l is a path between the variable and check nodes of l distinct edges which closes on it. The shortest possible cycle has the length of four. The minimum cycle length is girth. Short cycles of Tanner graph have a negative influence on the performance of iterative decoding [16]. So, short cycles should be avoided when designing good LDPC codes.

5.2 Performance Analysis

BER is the ratio of number of erroneous bits to the total number of bits. Cellular automata concept is introduced in the parity check matrix generation by using the nonuniform cellular automata rules. For the code length of 648, the BER curve is plotted for different SNR dB values in Fig. 1. The BER for noise variance $\sigma = 0.8$ is obtained for different SNRs from -1 to 5 dB for 25 iterations.

The CA-based LDPC decoder error performance is improved for the increasing SNR values. The error performance is similar to the other decoder performance for the different noise variance is analyzed.

5.3 Comparison Analysis

The hybrid CA rules are used to obtain the HDM structure. The HDM structure provides better error performance compared to the random parity check matrix. For $\sigma = 0.8$ with 30 iterations, the LDPC codes comparison plot between CA derived HDPCM and random HDPCM for different SNRs is analyzed and plotted in Fig. 2. The SNR of 3 dB gives a BER of 0.0849 dB for random HDPCM and 0.0432 dB for CA-derived HDPCM. The improvement in the BER is obtained as 0.0417 dB.

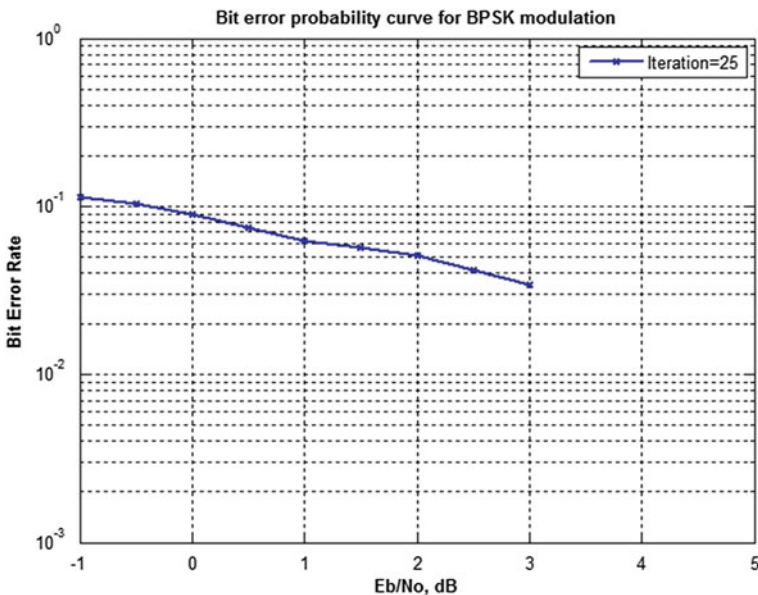


Fig. 1 BER curve for CA based LDPC codes

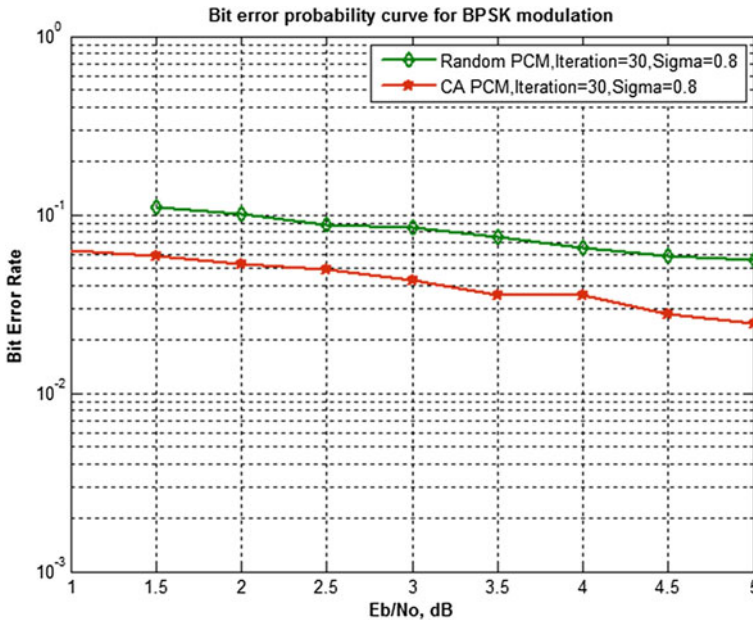


Fig. 2 Comparison between CA based and random LDPC codes

This analysis shows the error performance improvement by using the HDPCM structure in comparison with the random PCM in the LDPC decoding. The HDPCM matrix is a sparse matrix that reduces the computational complexity with the increase in the performance.

5.4 CA-Based LDPC Codes for Different Code Length

In LDPC decoding, by increasing the code length the error performance is improved. By the proposed cellular automata-based parity check matrix, the analysis for the increased code length has been done and plotted in Fig. 3. BER for code lengths 648 and 1008 was plotted for various SNR dB values with sigma = 0.8 and 25 iterations.

It has been observed that by increasing the code length, the BER performance improvement was observed. The complexity in the computation was also minimized by the increased sparsity in the cellular automata-based HDPCM. Comparing the code length $n = 648$ in the Wifi application and the underwater communication with $n = 1008$, the code length $n = 1008$ provides better error performance.

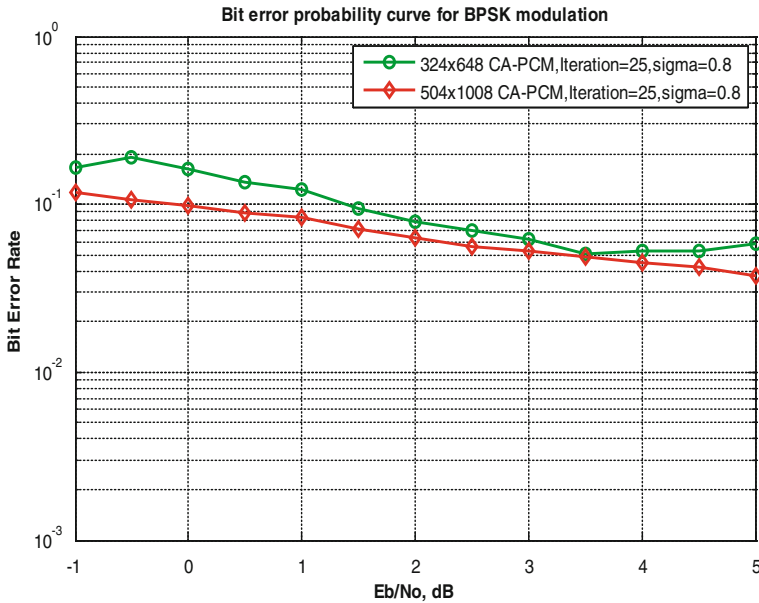


Fig. 3 BER for different code length

6 Conclusion

The cellular automata-based HDPCM is constructed and increased error performance is obtained by satisfying the girth and $\frac{1}{2}$ code rate for the decoder design. The performance analysis of the CA-based LDPC codes and the random LDPC codes provides improved error performance in the CA-based LDPC codes. Performance prompts to be increased for the increased code length. The designed decoder is evaluated for decoder parameters like BER, girth, and convergence rate. The CA-based LDPC codes provides a convergence rate of 0.0417 dB for code length $n = 324$ on comparing with random LDPC codes. The cellular automata-based LDPC decoder can be used for different channel modulations and different channels. The implication of the CA-based LDPC decoder can be used for higher order Galois Field.

References

1. Gallager RG. Low density parity check codes. IRE Trans Inf Theory 1962;IT-8:21–28.
2. Davey MC, Mackay DJC. Low density parity check codes GF (q). IEEE Commun Lett. 1998;2(6):165–7.
3. Ma X. On constructing low density parity-check codes. PhD thesis, University of Waterloo, Ontario, Canada; 2007.

4. Barnault L, Declercq D. Fast decoding algorithm for LDPC over $GF(2^q)$. *Inf Theory Workshop 2003*;2003:70–73.
5. Zhou B, Kang J, Song S, Lin S, Abdel-Ghaffar K, Xu M. Construction of non-binary quasi-cyclic LDPC codes by arrays and array dispersions. *IEEE Trans Commun* 2009;57(6):1652, 1662.
6. Song H, Cruz JR. Reduced-complexity decoding of Q-ary LDPC codes for magnetic recording. *IEEE Trans Magn*. 2003;39(2):1081–1087.
7. Hong H, Sun Z. An improved decoding algorithm of non binary LDPC code. In: 2011 International Conference on Information Technology, Computer Engineering and Management Sciences (ICM), vol 3; 2011. p. 104,107, 24–25.
8. Koroglu ME, Şiap I, Akın H. Cellular automata based byte error correcting codes over finite fields. In: American Institute of Physics Conference Proceedings, 2012, p. 183.
9. Cho S-J, Choi U-S, Heo S-H. Design of double error correcting codes based on cellular automata. *J Appl Math Comput*. 2006;21(1–2):545–53.
10. Sasidhar K, Chattopadhyay S, Chaudhuri PP. CAA decoder for cellular automata based byte error correcting code. *IEEE Trans Comput*. 1996;45(9):1003–16.
11. Das Debasis, Ray Abhishek. A parallel encryption algorithm for block ciphers based on reversible programmable cellular automata. *J Comput Sci Eng*. 2010;1(1):82–90.
12. Corsonello P, Spezzano G, Staino G, Talia D. Efficient implementation of cellular algorithms on reconfigurable hardware. In: Proceedings 10th Euromicro Workshop on Parallel, Distributed and Network-based Processing, 2002, p. 211, 218.
13. Wolfram. Theory and applications of cellular automata. In: Advanced series on complex systems, vol 1. Singapore: World Scientific, 1986. p. 485–557.
14. Tomassini M, Sipper M, Zolla M, Perrenoud M. Generating high-quality random numbers in parallel by cellular automata. *Future Gener Comput Syst*. 1999;16:291–305.
15. Carrasco RA, Johnston M. Non-binary error control coding for wireless communication and data storage. London: Wiley; 2008.
16. Tanner RM. A recursive approach to low complexity codes. *IEEE Trans Inf Theory* 1981;533–547.