

# Efficient Design to Error Detection in EG-LDPC Codes



Anvesh Thatikonda, C Chitra

**Abstract:** Capability to correct large number of errors Majority logic decodable codes is suitable for memory applications. The memory access time as well as area of utilization and the decoding time is reducing using Majority Logic Decoder. In Euclidean Geometry Low-Density Parity-Check (EG-LDPC) codes there exists fault secure detector which are used for error correction. Because EG-LDPC codes are widely used it will avoid high complexity in decoding. Majority logic decodable which is a one step is a technique similar to a class of Euclidean geometry low density parity check (EG-LDPC) codes. This method is very effective for EG-LDPC codes as shown in the result. The number of errors are detected by the majority logic decode further it can be used for correct the errors which occurred.

**Keyword:** EG-LDPC, decoding, encoding

## I. INTRODUCTION

The review codes are usually used to affirm recollections from so these are called fragile Errors, which change the reasonable estimation of memory cells without harming the circuit. As progress scales, memory contraptions convince the opportunity to be progressively unmistakable and altogether progressively fit slip-up change codes are required. These codes can fix a progressively significant number of Errors, yet generally, require complex decoders. To keep away from a high decoding complex nature, the usage of one phase larger part strategy for deduction decidable codes was from the outset proposed in for memory applications. Further work on this point at that point showed up. One phase bigger part justification interpreting can be recognized progressively with astoundingly direct equipment, regardless, requires long decoding events. In memory, this would build the way time which is a key framework parameter. Only a couple of classes of codes can be decoded using one phase lion's offer strategy for deduction unraveling. Among those are some EG low thickness consistency check (EG-LDPC) codes which were used as a segment, and multifaceted nature set low thickness correspondence check codes. A methodology was beginning late proposed to invigorate a successive utilization of larger part rationale unwinding of DS-LDPC codes.

The thought behind the procedure is to utilize the foremost cycles of greater part rationale unraveling to perceive whether the term being decoded covers Errors. On the off chance that there are no Errors, by then deciphering can be halted without finishing the remainder of the complements, along these lines basically diminishing the disentangling time.

For a code with square length  $N$ , greater part rationale deciphering requires  $N$  (when acknowledged successively) cycles, the code estimate develops, so does the translating time. In this procedure, essentially the fundamental three cycles are utilized to see Errors, thusly accomplishing a general speed increment precisely  $N$  is critical. It has been shown that for DS-LDPC codes, all blunder blends of up to five Errors can be perceived in the underlying three cycles. Additionally, Errors influencing more than five bits were perceived with a likelihood near one. The likelihood of undetected Errors was in like way found to lessen as the code square length extended. For a billion blunder structures just a few Errors (or from time to time none) were undetected. This may be tasteful for a few employments.

Another focal point of the proposed procedure is that it requires astoundingly insignificant extra hardware as the interpreting gear is in like way utilized for blunder affirmation. For instance, it was gave the possibility that the extra region required to understand the course of action was essentially around 1% for enormous word sizes. The procedure depends upon the DS-LDPC codes properties and subsequently it isn't unmistakably fitting to other code classes. In the going with, a practically identical framework for EG-LDPC codes is shown. Nano development gives littler, speedier, and lower imperativeness contraptions which concede considerably progressively phenomenal and limited hardware; on the other hand, these focal points go with an expense—the nano-scale devices may be less solid. Warm and shot-mayhem estimates alone recommend that the transient weakness rate of an individual nano-scale contraption (e.g., transistor or nano-wire) may be requesting of size higher than the present devices. In like manner, we can imagine that combinational reason will be helpless to transient defects notwithstanding limit cells and correspondence channels. In that capacity, the viewpoint of ensuring just memory cells and enduring the wrapping hardware will never indicate messes up isn't any progressively critical.

Manuscript published on January 30, 2020.

\* Correspondence Author

**Anvesh Thatikonda\***, Research Scholar, Dept. of E.C.E, Sri Satya Sai University of Technology & Medical Sciences, Sehore, Bhopal-Indore Road, Madhya Pradesh, India

**Dr. C Chitra**, Research Guide, Dept. of E.C.E, Sri Satya Sai University of Technology & Medical Sciences, Sehore, Bhopal Indore Road, Madhya Pradesh, India

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an [open access](http://creativecommons.org/licenses/by-nc-nd/4.0/) article under the CC-BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>)

II. MAJORITY LOGIC DECODING (MLD)

An n-bit code c, which k-bit encodes information vector I is made through copying the k-bit information vector with k × n bit generator structure G, i.e.,  $c = I \cdot G$ . Figure displays the generator system of (15, 7) EG-LDPC code the entirety of the lines of the grid is cyclic developments of the key line. This cyclic code age doesn't make a systematic code and the information bits must be decoded from the encoded vector, which isn't beguiling for our imperfection tolerant framework in perspective on the further perplexity and yield the action. The network of any code cyclic could be changed over into methodical structure ( $G = [I : X]$ ).

	C <sub>0</sub>	C <sub>1</sub>	C <sub>2</sub>	C <sub>3</sub>	C <sub>4</sub>	C <sub>5</sub>	C <sub>6</sub>	C <sub>7</sub>	C <sub>8</sub>	C <sub>9</sub>	C <sub>10</sub>	C <sub>11</sub>	C <sub>12</sub>	C <sub>13</sub>	C <sub>14</sub>
i <sub>0</sub>	1	0	0	0	1	0	1	1	1	0	0	0	0	0	0
i <sub>1</sub>	0	1	0	0	0	1	0	1	1	1	0	0	0	0	0
i <sub>2</sub>	0	0	1	0	0	0	1	0	1	1	1	0	0	0	0
i <sub>3</sub>	0	0	0	1	0	0	0	1	0	1	1	1	0	0	0
i <sub>4</sub>	0	0	0	0	1	0	0	0	1	0	1	1	1	0	0
i <sub>5</sub>	0	0	0	0	0	1	0	0	0	1	0	1	1	1	0
i <sub>6</sub>	0	0	0	0	0	0	1	0	0	0	1	0	1	1	1

Figure 1 Matrix generation of EG-LDPC (15, 7) code in cyclic format

The figure displays the systematic generator system to make EG-LDPC (15, 7) code. The encoded vector, which is made by the interior consequence of the data vector and the generator grid, contains data bits looked for after by uniformity bits, where every fairness bit is fundamentally an internal aftereffect of data vector and an area of X, from  $G = [I : X]$ . The decoder's structure is shown in Figure 2: The r(X) is the received vector in polynomial form. CS1, CS2,.....CSJ are the J check sums.

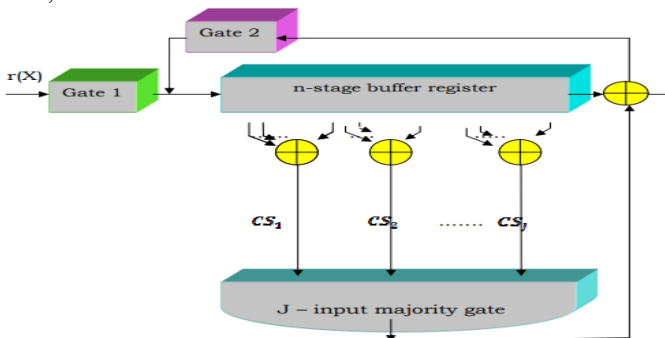


Figure 2. The structure of Majority logic decoder

III. THE ERROR CORRECTION PROCEDURE

1. Turn on gate1 while gate2 is turned off. The buffer register is used to store the received vector r(X). This vector is fed into the buffer register having the length equal to the total length of the message vector. The first received digit is r(n-1)
2. Form the parity J check sums which are orthogonal on e (n-1) by adding the relevant received digits over GF(2).
3. The orthogonal check sums (J) are applied to the gate of majority logic scheme. The digit r(n-1) emerging out from the buffer is corrected based on the resultant output of the majority logic gate.
4. At the end of the above step, the contents of the buffer register have been shifted by one place to the rightside with gate 2 'on'. The buffer register at this point, contains the second received digit at the appropriate

i.e., rightmost stage of the buffer register. This is corrected in exactly the same way as the first received digit was. The steps two and three are repeated by the decoder.

5. Decode digit by digit until the n shifts are over

EG-LDPC TWO-DIMENSIONAL CODES

A special subclass of EG-LDPC codes is the class of EG-LDPC codes over PG(2,2s) for various s and m=2. For any s >= 2 the, 2-D EGLDPC code's parameters are:

Length =  $n = 22s + 2s + 1$

Number of parity bits =  $(n-k) = 3s + 1$

Column weight of H matrix =  $2s + 1$

Number of information bits =  $k = n - (3s + 1)$

Row weight of H matrix =  $2s + 1$

Minimum distance =  $2s + 2$

The H matrix is of dimension  $(22s + 2s + 1)$ -by-  $(22s + 2s + 1)$ . It could be obtained by compelling the frequency vector of a line in PG (2, 2s) &  $22s + 2s$ .

IV. IMPLEMENTATION ALGORITHM FOR ENCODER AND DECODER

The encoder and decoder of the EG-LDPC codes is described. The encoder design is a standard method available in the literature. This is briefly covered for completeness, but as such the decoder design is the main focus of the thesis which is a novel approach.

Encoder Procedure

The encoder implements the code vector  $C_X = I_X * G\_MATRIX$  where  $I_X$  is the length k information vector over  $Z_q$ . The encoder can be designed using k or n-k shift registers over  $Z_q$  with feedback determined by the generator polynomial g(X) of EG(2, 2<sup>s</sup>, 2<sup>s</sup>) over  $Z_q$ . Since the cyclic code is any serial scheme based on shift registers that are applicable for encoding binary/symbol based cyclic codes can be employed for the encoder design. Instead of Galois field additions and multiplications, p-adic integer addition and multiplication are used. Also shift registers store p-adic integers. Figure 4.3 depicts the generic structure of the encoder. Let the feedback coefficients be  $a_0$  to  $a_{n-k}$ . The additions and multiplications are over modulo  $2^l$ . In the figure,  $I_X$  and  $C_X$  are the information and the encoded vectors/polynomials.

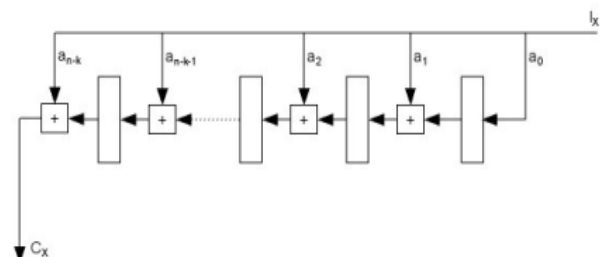


Figure 3 Encoder structure

Strategic Decoding Procedure

For the application, an appropriate simple majority logic decoding is used to simplify the complexity. Since majority logic is applied in binary domain, one can implement a sequential method to decode the received vector R(X) over  $Z_q$ .

## Design Parameters Verification

The parameters, for example, the quantity of mistakes and the comparing data bits, the equality bits required according to the table 4.1 are utilized in the table 5.1 to test the blunder location and the remedy capacity of the 2D EG LDPC code. In every one of the cases since just two dimensional p-adic (2-adic) is considered. Hence,  $p=2$ . The densities of the zeros and non-zeros are determined showing that the non-zeros passages become more as the type 't' is increased (for 2,3,4,5 mistakes Figs. 5.1,5.2,5.3,5.4), i.e., as the ring size builds, the non-zeros sections increment. The ring size really shows the arbitrary burst mistake length. The proposed design deals with the higher burst length blunder location and remedy by unfurling the layer which generally is changed off to spare the power. Subsequently, according to the interest prerequisite, the interpreting engineering reaches out into multilayered engineering. The structure of the design is made to suite VLSI with the end goal that the particular methodology is done, a prime necessity for this decoder.

## V. THE COMPLEXITY ANALYSIS

RS code Uses *Galois Field* arithmetic. The Addition operation involves EX-OR operation. The computational complexity is economical whereas the computational complexity of multiplication is expensive as detailed below. Lookup tables of log and anti-log are expensive for multiplication and division which is computationally expensive. It's Encoding is of the order of  $O(mn)$ . The requires decoding for  $n \times n$  inversion matrix of the order of:  $O(n^3)$  and then  $O(n^2)$  to re-calculate data words. However, Decoding is  $O(n^3) + O(n^2x)$  with  $x$  words per block, Encoding is  $O(mnx)$ .

## VI. RESULTS

It is like a travelling bag with zip folded layers to expand when required. As the load increases, the bag's capacity can be increased by unzipping the bottom layer of the bag. In the current context of EG-LDPC decoding, the switching of the layers of decoding in the decoding architecture is based on the user's requirement of quality of the content. This feature does not exist currently in decoding operation. It also caters to the burst errors of variable length dynamically. The complexity of the EG LDPC codes is only linear as against non-linear complexity of RS codes. An application of EG LDPC code decoder in fault tolerance of a single node/link failure using the new concept of 0-free PDN is proposed. This application has lot of commercial applications in medical/aerospace/critical systems.

## VII. CONCLUSION

The EG LDPC is made an image decoder instead of a bit decoder like in RS codes with the assistance of p-adic idea. Lifting plan represents the various terms of burst length. This offers a one of a kind component. The p-adic ring size can be powerfully changed according to the need of the application which in the remote condition has turned out to be critical. For the vitality constrained versatile terminals, this can be a valuable alternative, in spite of the fact that

there is an exchange off with the presentation. Likewise, when the SNR/SIR is great, one can change from conviction spread to dominant part rationale disentangling without trading off the presentation to lessen the handling power.

## REFERENCES

- Bai, X. Wang, and M. Xu, C. Chen, B. "Non-binary LDPC codes built on the basis of a cyclic MDS code and a low-complexity non-binary message transmission decoding algorithm" IEEE Commun. Lett., vol. 14, no. 3, pp. 239-241, Mar. 2010.
- Bo Fu Ampadu, P. "Hybrid ARQ Fault Detection with Reduced Cross-talk Delay for Reliable Chip Connections", 24th International IEEE Fault Tolerance Symposium in VLSI Systems, PP 440-448, Chicago, October 2009. DFT '09.
- Hsing-Min Chen, Pei-Lun Diao, Chung-Hsuan Wang, John K. Zao, Chih-Ming Chiu, "A free convolution UEP code for robust SVC / MGS wireless transmission", Multimedia (ISM), 2011 IEEE International Symposium on Page(s): 279 - 284, Dec. 2011.
- Jiang Y, Yang M, Balakrishnan G and Kim Y., "Performance Analysis of Wireless Sensor Network Error Codes", papers at the 4th International Conference on Information Technology (ITNG) 2007, 2007, pp. 876-879.
- M. Rakov and B. Parhami, "Applying Perfect Divergences for Designing Efficient and Robust Connection Networks," Proc. Conf. International Communications in Data Processing, Las Vegas, NV, 27.-30. 207-213
- Mengqi Zhou W. Ross Stone, "Microwave, Antenna, Propagation and EMC Technologies for Wireless Communications", MAPE, IEEE 5th International Symposium Conference, 2013
- Mohamed H. Omar, Ahmed El-Mahmoudy, Karim G. Seddik, Ayman Elezabi, "On the Tail-Biting Convolutional Code Decoder for the LTE and LTE-A standards", Asilomar Conference on Signals, Systems and Computers, Date: 3-6 Nov. 2013, Page no. 510 - 514.
- Mohammad M. Mansour, Naresh R. Shanbhag, "A Novel Design Methodology for High- Performance Programmable Decoder Cores for AA-LDPC Codes", Journal of VLSI Signal Processing Systems, vol 40, Iss 3, pages 371-382, July 2005.
- Rafid Mahmood Ahmed Badr ; Ashish Khisti "Convolutional codes with maximum column sum rank for network streaming", IEEE International Symposium on Information Theory (ISIT), 14-19 June 2015, pp 2271 - 2275.
- Rolf Johannesson, Irina E. Bocharova, Boris D. Kudryashov, "LDPC convolutional codes and QDP-LDPC block codes in standard communication scenarios", IEEE International Symposium on Information Theory, Date June 29 2014-July 4 2014, Page No. 2774 - 2778.
- T. Zhang, and F. Sun "Temporary and Incorrect Fault Tolerant System Design for CMOS / Nanodevice Hybrid Digital Storage," IEEE Trans. Nanotechnol., vol. 6, no. 3, pp. 341-351, Jun. 2007
- Tarakpraveen Uppalapati, Harihara S.G, B.Suryanarayana Adiga, M. Girish Chandra, "Decoding Architectures for Projective Geometry Based LDPC Codes", IFIP Wireless Days, Dubai, Nov.2008.
- Wonyong Sung, Junho Cho, Jonghong, and Kim, Hyunwoo Ji "VLSI Implementation of a Soft Bit-Flipping Decoder for PG-LDPC Codes", IEEE, pp. 908-911, 2009.

## AUTHORS PROFILE

**Anvesh Thatikonda**, is currently Pursuing Ph.D in Sri Satya Sai University of Technological & Medical Science, Bhopal. He has completed his M. Tech from JNTUH.

**Dr. C. Chitra**, Research Guide, Dept. of E.C.E, Sri Satya Sai University of Technology & Medical Sciences, Sehore, Bhopal Indore Road, Madhya Pradesh, India.