Optimization of A New Low Density Parity-Check Decoder using The Syndrome Block

Lagrini Lakbir[1], Sedra Moulay Brahim[2]
[1,2]Laboratoire des Hautes Energies et Sciences de l'Ingénieur et Réacteurs(LHESIR)
The University Ibn Tofail of Kenitra at Morocco

***Abstract* –the Low density parity-check codes are one of the hottest topics in coding theory nowadays equipped with very fast encoding and decoding algorithms, LDPC are very attractive both theoretically and practically.in this article, we present a new algorithm which allows correcting errors quickly and without iterations. We show that the proposed algorithm can be applied for both regular and irregular LDPC codes.**

***Keywords:* Low Density Parity Check decoder, Bit Flipping Algorithm, syndrome block, proper syndrome, Matrix equation.**

I. INTRODUCTION

The Low density parity-check code (LDPC) is an error correcting code used to protect information against noise introduced by the transmission channel and reduce the probability of loss information. Grace Low Density Parity Check, the information transmission rate can be as close to Shannon limit in the same noisy channel. LDPC was developed by Robert Gallager [1] in his doctoral dissertation at MIT in 1960 [2]. It was published 3 years later in MIT Press. Due to the limitation in computational effort in implementing the coder and decoder for such codes and the introduction of Reed-Solomon codes, LDPC was forgotten for almost 31 years. During this period, R. Michael Tanner in 1981 has done important work on this subject, as he generalized LDPC codes with the introduction of the graphical representation of the codes later called Tanner graph. In 1993, with the invention of turbo codes, researchers switched their focus to finding lowcomplexity code which can approach Shannon channel limit. LDPC was reinvented with the work of Mackay, and Luby. Nowadays, LDPC have made its way into some modern applications such as 10GBase-T Ethernet, WiFi, WiMAX, Digital Video Broadcasting (DVB) [3], [4].in contract, the Bit Flip Algorithm(BFA)[5], which isbased on hard-decisiondecoding, has the least complexity but suffers from poor performance of iteration. In this paper, the proposed algorithm also allows for the complete elimination of the decoding iteration compared to the old method, particularly in the second part of the table. At the same time this method reduces the number of iteration by the introduction of a new concept of proper syndromes in part 3.

II. SYSTEM MODEL

The Bit-Flip Algorithm

The Bit-Flip algorithm is based on hard-decision message passing technique. A binary hard-decision is done on the received channel data and then passed to the decoder. The messages passed between the check node and variable nodes are also single-bit hard-decision binary values. The variable node (r) sends the bit information to the connected check nodes (S) over the edges. The check node performs a parity check operation on the bits received from the variable nodes. It sends the message back to the respective variable nodes with a suggestion of the expected bit value for the parity check to be satisfied [6].

The algorithm:

Step 1: We use the equation S = Hr^T to calculate the syndrome with the received vector. If the elements in the set of S are all zeros, it's terminated with the correct vector, otherwise, go to the next step.

Step 2: Calculate the set of {f0, f1, · · ·, fN−1} and find the largest fj .Then transfer the corresponding rj to its opposite number (0 or 1), get a new vector r'.

Step 3: Calculate the vector S = Hr^T with the new vector r'. If the elements of S are all zeros or the iterations reach the maximum number, the decoding is terminated with the current vector, otherwise, the decoding go back to step 2.

Clearly the BFA has simple check node and variable node operations, thus making it a very low complexity decoding algorithm compared to the other algorithms. But this advantage comes with a poor decoding performance.

III. PREVIOUS WORK

Much work has been done to reduce the number of iterations.Vikram Arkalgud Chandrasetty [7] developed a new algorithm using Sum-product algorithm to improve BER and FER performances compared to fully hard-decision based solutions such as those based on the Bit Flip Algorithm (BFA). This article is a continuation of these works in order to reduce the average number of decoding iterations.

IV. PROPOSED METHODOLOGY

The proposed algorithm consists of accomplishing a table, which is divided into three parts or steps.

Step 1 (P1) contain parity check equations.

Step2 (P2), which runs diagonally, includes some possible syndromes from parity syndromes of part 1, without making any calculations and without any iteration. All we need to do is flipping the variable node ri in order to reach a Null Syndrome.

Step 3 we introduce the notion of proper syndrome of the second part, this may we only need to make the addition of the proper syndrome of the second part in order to find the remaining possible syndromes bearing in mind the algorithm of the flowing calculation:

The rules of the addition of proper algorithms

- $S_i \oplus S_j = S_iS_j$ and $S_i \oplus S_i = 0$
- $S_1S_{2..}S_iS_jS_k \oplus S_1S_{2...}S_iS_jS_k = 00..000$
- $S_i \oplus S_j \oplus Sm = S_iS_jS_m$
- $S_iS_jS_m \oplus S_iS_jS_mS_k \oplus S_iS_jS_l = S_iS_jS_kS_l$
- $1 \oplus 1 = 0$ and $1 \oplus 0 = 1$ and $0 \oplus 0 = 0$

ILLUSTRATIVE EXAMPLES:

A/ PARITY CHECK MATRIX H OFF LOW DENSITY PARITY CHECK (LDPC) CODE OF DIMESION 3X7:

Here we will only consider binary messages and so the transmitted messages consist of strings of 0's and 1's.

Example1:

Example parity check matrix

$$H=\begin{bmatrix}1001011\\0101110\\0010111\end{bmatrix}$$

In the first place,

We calculate the syndrome S of the received codeword r such as S=HrT

- If $Hr^T = 0$ (null syndrome) then the received codeword is correct, therefore terminate the algorithm, decoding it with the corresponding message.
- If $Hr^T \neq 0$, the non-zero syndrome, therefore the code word received is incorrect.

The proposed algorithm can determinate the variable node rithat we have toflip in order to find a null syndrome

Suppose the codeword received after the transmission channel is r = r1r2r3r4r5r6r7 Where each ri is either 0 or 1 and HrT = S1S2S3 (the syndrome S= S1S2S3)

$$\begin{pmatrix}1001011\\0101110\\0010111\end{pmatrix}\begin{pmatrix}r1\\r2\\r3\\r4\\r5\\r6\\r7\end{pmatrix} = S_1S_2S_3$$

Each row of H gives a parity check equation:

$$S1 = r1 \oplus r4 \oplus r6 \oplus r7$$

$$S2 = r2 \oplus r4 \oplus r5 \oplus r6$$

$$S2 = r3 \oplus r5 \oplus r6 \oplus r7$$

Equations are called parity-check equations, in which the symbol $\oplus$ represents modulo-2 addition.

Table1.REPRESENTS DIFFERENT CASES OF SYNDROME FOR LDPC (7, 3) CODES:

$S_1S_2S_3$	**r_1**	**r_2**	**r_3**	**r_4**	**r_5**	**r_6**	**r_7**	**Proper Syndrome**	**P**
S_1	X			X		X	X		
S_2		X		X	X	X			P1
S_3			X		X	X	X		
100	X							S_1	
010		X						S_2	
001			X					S_3	
110				X				S_1S_2	P2
011					X			S_2S_3	
111						X		$S_1S_2S_3$	
101							X	S_1S_3	

X in part 2 indicate sr_i which we have to flip in order to annul the syndrome

Example 1:

Explanation of proper syndrome

In column six, the case between line S1 and column r5 is empty, but between line S2 and column r5 , there is an X and between line S3And column r5 there is also an X, which gives the following Expression S2S3 , which is represented in binary by 011.

Example 2:

Supposing after the calculation ofHr^T, we find the syndrome 011 which, according to the Proposed Algorithm, is equivalent to S_2S_3

In this case we only need to flip r_5 to get the null syndrome

If r_5=1 will be changed to 0

If r_5=0 will be changed to 1

We will explain how to read this table:

- The first part (P1) represents parity check equation S1,S_2 and $S_3$and the X represents the r_i forming the Si.
- The second part (P2) which runs diagonally in the table represents the possible syndromes from equations of part 1 without making any calculations. And the X represents the r_i(=1 or 0) which we have to flipping in order to get the syndrome which corresponds to zero.
- The third part (P3) which is in the other tables represents the rest of the possible syndromes which are possible to calculate by introducing the new notion of adding proper syndromes of part 2 (P2).

B/ PARITY CHECK MATRIX H OFF LOW DENSITY PARITY CHECK (LDPC) CODE OF DIMENSION 4X9:

Example2:

Example parity check matrix

$$H=\begin{bmatrix}011011000\\101100100\\111010010\\000110001\end{bmatrix}$$

In the first place,

We calculate the syndrome S of the received codeword r such as S=HrT

- IfHr^T=0(null syndrome) then the received codeword is correct, therefore terminate the algorithm, decoding it with the corresponding message.
- If$Hr^T\neq 0$, the non-zero syndrome, therefore the codeword received is incorrect.

The proposed algorithm can determinate the variable node ri that we have to flip in order to find a null syndrome,

Suppose the codeword received after the transmission channel is r = $r_1r_2r_3r_4r_5r_6r_7r_8r_9$Where eachri is either 0 or 1and HrT = S1S2S3S4(the syndrome S= S1S2S3S4).

$$\begin{pmatrix}011011000\\101100100\\111010010\\000110001\end{pmatrix}\begin{pmatrix}r_1\\r_2\\r_3\\r_4\\r_5\\r_6\\r_7\\r_8\\r_9\end{pmatrix}=S_1S_2S_3S_4$$

Each row of H gives a parity check equation:

$$S_1 = r_2 \oplus r_3 \oplus r_5 \oplus r_6$$
$$S_2 = r_1 \oplus r_3 \oplus r_4 \oplus r_7$$
$$S_3 = r_1 \oplus r_2 \oplus r_3 \oplus r_5 \oplus r_8$$
$$S_4 = r_4 \oplus r_5 \oplus r_9$$

TABLE 2. REPRESENTSALL POSSIBLE SYNDROMES FOR LDPC (9, 4) CODES:

$S_1S_2S_3S_4$	r_1	r_2	r_3	r_4	r_5	r_6	r_7	r_8	r_9	Proper Syndrome	part
S_1		X	X		X	X				Parity check equation	Part 1
S_2	X		X	X			X				
S_3	X	X	X		X			X			
S_4				X	X				X		Part 2
0110	X									S_2S_3	
1010		X								S_1S_3	
1110			X							$S_1S_2S_3$	
0101				X						S_2S_4	
1011					X					$S_1S_3S_4$	
1000						X				S_1	

0100							X			S_2	
0010								X		S_3	
0001									X	S_4	
1111		X		X						$S_1S_2S_3S_4$	Part 3
0111	X								X	$S_2S_3S_4$	
1101				X		X				$S_1S_2S_4$	
1001						X			X	S_1S_4	
0011	X			X						S_3S_4	
1100						X	X			S_1S_2	

Exampleofexplication:

If we make the product of Hr^T and we find for example the syndrome 0111 which is equivalent proper syndrome $S_2S_3S_4$ from the proposed algorithm; we will look in the table, especially the two part (P2), the r_i need to flip to achieve null syndrome.

The proposed algorithm uses the addition of proper syndrome as follows:

- $S_2S_3 \oplus S_4 = S_2S_3S_4$forcing us to flip r_1(case of S_2S_3) etr_9 (case$of S_4$)or
- $S_2 \oplus S_3 \oplus S_4 = S_2S_3S_4$forcing us to flip r_7(case of S_2), r_8 (case of S_3) andr_9(case of S_4) or
- $S_2S_4 \oplus S_3 = S_2S_3S_4$forcing us to flip $r_4(case of S_2S_4$) and$r_8(case of S_3)$.

C/ PARITY CHECK MATRIX H OFF LOW DENSITY PARITY CHECK (LDPC) CODE OF DIMENSION 5X10:

Example 3:

Example parity check matrix

$$H=\begin{bmatrix}1111000000\\1000111000\\0100100110\\0010010101\\0001001011\end{bmatrix}$$

In the first place,

We calculate the syndrome S of the received codeword r such as $S = Hr^T$

- If$Hr^T = 0$ (null syndrome) then the received codeword is correct, therefore terminate the algorithm, decoding it with the corresponding message.
- If $Hr^T \neq 0$,the non-zero syndrome, therefore the codeword received is incorrect.

The proposed algorithm can determinate the variable noder_i that we have to flip in order to find a null syndrome.

Suppose the codeword received after the transmission channel is $r = r_1r_2r_3r_4r_5r_6r_7r_8r_9r_{10}$
Where eachr_i is either 0 or 1and $Hr^T = S_1S_2S_3\,S_4S_5$ (the syndrome$S = S_1S_2S_3S_4S_5$).

$$\begin{pmatrix}1111000000\\1000111000\\0100100110\\0010010101\\0001001011\end{pmatrix}\begin{pmatrix}r_1\\r_2\\r_3\\r_4\\r_5\\r_6\\r_7\\r_8\\r_9\\r_{10}\end{pmatrix} = S_1S_2S_3S_4S_5$$

Each row of H gives a parity check equation:

$$S_1 = r_1 \oplus r_2 \oplus r_3 \oplus r_4$$
$$S_2 = r_1 \oplus r_5 \oplus r_6 \oplus r_7$$
$$S_3 = r_2 \oplus r_5 \oplus r_8 \oplus r_9$$
$$S_4 = r_3 \oplus r_6 \oplus r_8 \oplus r_{10}$$
$$S_5 = r_4 \oplus r_7 \oplus r_9 \oplus r_{10}$$

TABLE 3 REPRESENTS ALL POSSIBLE SYNDROMES FOR LDPC (10, 5) CODES.

$S_1S_2S_3S_4$ S_5	r_1 r_2 $r_3r_4r_5r_6r_7r_8r_9r_{10}$									**Proper syndrome**
S_1	X	X	X	X							
S_2	X				X	X	X				
S_3		X			X			X	X		
S_4			X			X		X		X	
S_5				X			X		X	X	
11000	X										S_1S_2
10100		X									S_1S_3
10010			X								S_1S_4
10001				X							S_1S_5
01100					X						S_2S_3
01010						X					S_2S_4
01001							X				S_2S_5

00110								X			S_3S_4
00101									X		S_3S_5
00011										X	S_4S_5
11110	X							X			$S_1S_2S_3S_4$
11011	X									X	$S_1S_2S_4S_5$
01111							X	X			$S_2S_3S_4S_5$
11101		X					X				$S_1S_2S_3S_5$
10111		X								X	$S_1S_3S_4S_5$

D/ PARITY CHECK MATRIX H OFF LOW DENSITY PARITY CHECK LDPC (12, 6) OF DIMENSION 6X12:

Example 4: Parity-check matrix is:

$$H=\begin{bmatrix}110100001101\\111110010000\\101011000110\\000111101010\\011000110011\\000001111101\end{bmatrix}$$

In the first place,

We calculate the syndrome S of the received codeword r such as $S = Hr^T$

- If $Hr^T = 0$ (null syndrome) then the received codeword is correct, therefore terminate the algorithm, decoding it with the corresponding message.
- If $Hr^T \neq 0$,the non-zero syndrome, therefore the codeword received is incorrect.

The proposed algorithm can determinate the variable node r_i that we have to flip in order to find a null syndrome.

$Hr^T = S_1S_2S_3\,S_4S_5S_6$ (The syndrome $S = S_1S_2S_3S_4S_5S_6$).

$$\begin{pmatrix}110100001101\\111110010000\\101011000110\\000111101010\\011000110011\\000001111101\end{pmatrix}\begin{pmatrix}r_1\\r_2\\r_3\\r_4\\r_5\\r_6\\r_7\\r_8\\r_9\\r_{10}\\r_{11}\\r_{12}\end{pmatrix} = S_1S_2S_3S_4S_5S_6$$

Suppose the codeword received after the transmission channel is $r = r_1r_2r_3r_4r_5r_6r_7r_8r_9r_{10}r_{11}r_{12}$

Where each r_i is either 0 or 1 and

Each row of H gives a parity check equation:

$$S_1 = r_1 \oplus r_2 \oplus r_4 \oplus r_9 \oplus r_{10} \oplus r_{12}$$
$$S_2 = r_1 \oplus r_2 \oplus r_3 \oplus r_4 \oplus r_5 \oplus r_8$$
$$S_3 = r_1 \oplus r_3 \oplus r_5 \oplus r_6 \oplus r_{10} \oplus r_{11}$$
$$S_4 = r_4 \oplus r_5 \oplus r_6 \oplus r_7 \oplus r_9 + r_{11}$$
$$S_5 = r_2 \oplus r_3 \oplus r_7 \oplus r_8 \oplus r_{11} \oplus r_{12}$$
$$S_6 = r_6 \oplus r_7 \oplus r_8 \oplus r_9 \oplus r_{10} \oplus r_{12}$$

TABLE4.SUMMARIES ALL THE POSSIBLE CASE OF THE SYNDROME OF THE PARITY MATRIX TOP OF COLUMN 12 AND 6 ONLINE:

$S_1S_2…S_6$	r_1	r_2	r_3	r_4	r_5	r_6	r_7	r_8	r_9	r_{10}	r_{11}	r_{12}
S_1	X	X		X					X	X		X
S_2	X	X	X	X	X			X				
S_3	X		X		X	X				X	X	
S_4				X	X	X	X		X		X	
S_5		X	X				X	X			X	X
S_6						X	X	X	X	X		X
111000	X											
110010		X										
011010			X									
110100				X								
011100					X							
001101						X						
000111							X					
010011								X				
100101									X			
101001										X		
001110											X	
100011												X
100010	X		X									
001010	X	X										
001100	X			X								
100100	X				X							
110101	X					X						
111111	X						X					
101011	X							X				
011101	X								X			
010001	X									X		
110110	X										X	
011011	X											X
101000		X	X									

000110		X		X								
101110		X			X							
100001		X						X				
010111		X							X			
111100		X									X	
101110			X	X								
010111			X			X						
001001			X					X				
110011			X							X		
010100			X								X	
111001			X									X
100111				X				X				
001111					X			X				
010010					X						X	
011110						X		X				
000011						X					X	
111010								X		X		
110000								X				X
100000			X	X							X	
010000	X	X	X									
001000	X							X				X
000100		X						X	X			
000010									X	X	X	
000001						X			X	X		
000101	X	X			X			X				
001011	X			X			X					
010110	X						X			X		
011000	X			X			X	X				
011001	X								X	X	X	X
011111	X			X				X				
100110		X	X								X	
101010	X				X						X	
111011		X	X					X				
111101		X			X			X				
111110	X	X		X								
101101											X	X
101111		X	X				X					

If for example we find, while calculating the syndromes

- S= 001010 the proper syndrome equivalent, according to the proposed algorithm is S_3S_5 which is equal to $S_3 \oplus S_5$, we go back to the table:

  We have s_3 corresponds to the changeover of $r_1, r_8\ and r_{12}$, and we also have S_5 which corresponds to the changeover of $r_9, r_{10}\ and r_{11}$, so, in the end we only have to flipping $r_1, r_8, r_9 r_{10}, r_{11} and r_{12}$ in order to get a null syndrome.
- If for example we have the syndrome 101101which equals the proper syndrome $S_1S_3S_4S_6$,which in turn, equals $S_3S_4S_5 \oplus S_1S_5S_6$ all we need to do is flipping r_{11} (case of $S_3S_4S_5$) and $\boldsymbol{r_{12}}$ (case of $S_1S_5S_6$) in order to find a Null syndrome.
- Another example if the syndrome is S=000011 and the equivalent proper syndrome is $s_5s_6 = s_5 \oplus s_6$, all we have to do is flipping the variable nodes $r_9, r_{10}\ and r_{11} (case of s_5) and then r_6, r_9 and r_{10}$ but $r_9 and r_{10}$ are respected, all we need to do is flipping $r_6\ and r_{11}$ in order to have a null syndrome.

V. COMPARAISON OF ALGORITHMS

TABLE 5.SHOWS THE COMPARISON BETWEEN THE OLD WAY AND THE NEW ALGORITHM

	Basic algorithm	Modified algorithm
LDPC (7,3)	2^3-1 possible syndromes and 2 possible iterations	Without any iteration for all possible syndromes (2^3-1 possible syndromes)
LDPC (9,4)	2^4-1 possible syndromes, 4 possible iterations. Case ofcodeword received is $r = 101010100$	-Part 2 : without iteration -Part 3 :the additions of proper syndromes of part 2
LDPC (10,5)	15 possibles syndromes possibles iteration	-Part 2 : without iteration -Part 3 : adding proper syndromes of the part 2
LDPC (12,6)	2^6-1 possibles syndromes and 4 possibles iteration	-part 2 : without iteration -part 3 :the addition of proper syndromes of part 2
LDPC (n, k)	if n=2^k -1 and the columns are linearly independent, basic algorithm several iteration	if n=2^k-1 and the columns are linearly independent, we find all directly syndrome without iteration in the part 2 (P2) of the table and permanently delete part 3 (P3).

VI. CONCLUSION

In this paper:

- The proposed algorithm based on the new notion of the syndrome, what we call proper syndrome and the shape of the schedule, which allows us in a simple and easy way to find the variable nodes that we have to flip in order to find a null syndrome instead of the classical iteration method.
- The proposed algorithm allows us as well to completely eliminate and delete the classical decoding

iteration, particularly in relation to a certain number n of the syndrome, such as n the length of the matrix of the parity H.

- The first part in the table gives directly the variable nodes r_i that we have to flip to find a null syndrome, and by the addition of proper syndromes, we can find the r_i that we have to flipping in order to have the reste of the possible null syndromes.

VII. FUTURE SCOPES

The proposed algorithm can be improved in future work about the implementation of carte FPGA, Digital Video Broadcasting-second generation (DVB-S_2).

REFERENCES

[1] R.G.Gallager, Low-Density Parity-Check Codes. IRE Transactions on Information Theory, January 1962,

[2] B. Shin, H. Park, S. Hong, J. Seon and SH. Kim “Quasi-cyclic LDPC codes using overlapping matrices and their layered decoders”, International Journal of Electronics and Communications VOL.68(5), (2014). http://dx.doi.org/10.1016/j.aeue.2013.10.004.

[3] AH. El-Maleh, MA. Landolsi and EA. Alghoneim, “Window-constrained interconnect-efficient progressive edge growth LDPC codes”, international journal of electronics and communications VOL. 67, 2013.

[4] W. Zhenbang, W. Zhenyon, G. Xuemai and G. Qing, “Cross-layer design of LT codes and LDPC codes for satellite multimedia broadcast / multicast services”, china journal of aeronautics (2013).

[5] D.J.C. MacKay, "Good error-correcting codes based on very sparse matrices," *IEEE Transactions on Information Theory*, vol. 45, no. 2, pp. 399-431, March 1999. http://dx.doi.org/10.1016/j.ijleo.2012.09.047.

[6] NP. Bhavsar, B. Vala, “Design of Hard and Soft Decision Decoding Algorithms of LDPC”, International Journal of Computer Applications VOL. 90(4), (2014), http://dx.doi.org/10.5120/15803-4653.

[7] Vikram Arkalgud Chandrasetty and Syed Mahfuzul Aziz, FPGA Implementation of a LDPC Decoder usinga Reduced Complexity Message PassingAlgorithm, School of Electrical & Information Engineering, University of South Australia, JOURNAL OF NETWORKS, VOL. 6, NO. 1, JANUARY 2011