Design of Low Complexity Low Delay Short Length Code Based on Convolutional Encoder and LDPC Decoder

Bashar M. Mansoor and Tarik Z. Ismaeel

***Abstract*—**Due to their low delay and complexity, short length codes are attractive for use in wireless communication systems. Turbo and Low-Density Parity Check (LDPC) codes achieve excellent error correction capability and good performance at large block lengths, while the complexity and delay increase exponentially with block length. The main contribution of this work is to design a code with low time delay and low computation complexity, while having a improved BER performance. The proposed code is based on convolutional encoding and LDPC decoding, and can support variable block sizes and multiple code rates. The architecture of this code is investigated for short block length, which aims at optimizing performance by growing the generator and parity-check matrices. The proposed codes are evaluated and simulated over additive white Gaussian noise (AWGN) channel and Quadrature Phase Shift Keying (QPSK) modulation scheme, and then compared with irregular LDPC and turbo codes using a range of rates and block lengths. Theoretical analysis denotes that the structure of this code offers advantages in terms of latency and complexity, while simulation results show that the proposed code outperforms the irregular LDPC code by 0.4 dB and outperforms the turbo code (used in LTE) by 0.2 dB in terms of BER performance.

***Keywords*—**LDPC codes, Convolutional encoder, LTE turbo code, Low-density parity-check matrix (H), Decoding complexity, Decoding delay, the BER performance.

I. Introduction

Interest in short length codes has been rising recently due to their low delay and complexity. In addition, transmission over a multiple code rate system is desirable when the channel is time-varying. A code of short length and multiple rates can be achieved using low-density parity check codes [1]. These codes were first invented by Gallager in the early 1960s, then improved by MacKay and Neal in the late 1990s [2, 3]. It is shown that LDPC codes are able to closely approach the channel capacity and they are very power efficient among short length blocks [4]. LDPC codes are described by a sparse parity-check matrix (H) and called low-density parity check since the density of ones in H matrix is very low. These codes achieve a high bit error rate (BER) performance (low error floor) at long lengths, however, their computation complexity increases linearly with respect to block length. In the last decades, many researchers confirmed that LDPC codes can be decoded in a similar manner to convolutional and turbo codes, and they are actually able to beat the turbo codes in terms of bit error floor, computational complexity, and decoding time [5-8]. A detailed study of modern codes for 4G and future generations of mobile communication systems is presented in [9], where the performance and the stability of LDPC and turbo codes are explored in this study. In [10], researchers showed that LDPC codes work very well at high and low code rates, and due to good performance, they are featured as channel coding for the WiMax standard [11]. An efficient method named convolutional LDPC or recursive LDPC has been invented in order to simplify hardware implementation and reduce computation complexity as well as to reduce the time required for encoder/decoder. These methods employ the convolutional technique to encode information instead of using the encoding of LDPC block code, where the encoder uses only a small number of memory units [12-16]. In the recursive encoding method, complexity and the delay are significantly reduced. However, this method was presented for large code lengths and several constraints must be considered in constructing the H matrix. In this paper, we proposed a novel method to construct a systematic recursive irregular short length LDPC code using a simple implementation of the convolutional encoder and a specific structure of the irregular parity-check matrix. The proposed code is suitable for low delay transmission and can support variable length multi-rate code. The paper is organized as follows; Section II: introduces standard LDPC code. Section III: shows the basic construction of convolutional encoder. An overview of turbo code of LTE is presented in section IV. The proposed model is depicted in section V, and the simulation results with their discussions are detailed in section VI. Section VII shows the conclusion of the paper.

II. LDPC Codes

LDPC codes are characterized by a binary matrix of size (m x n) called parity-check matrix, where m and n represent the number of parity check equations and the code length respectively. This H matrix contains a small number of non-zero elements (ones) and is defined with two main parameters, row weight (r_w) and column weight (c_w),. The r_w corresponds to the number of non-zero elements in a row and the c_w corresponds to the number of non-zero elements in a column.

Bashar M. Mansoor is a Ph.D. student at the Engineering College / Baghdad University – Baghdad / Iraq. (e-mail: bmml77@yahoo.com).

Tarik.Z.Ismaeel is a Professor of Communications Engineering at the Engineering College / Baghdad University – Baghdad / Iraq.

ISSN: 1998-4464

If the r_w and c_w Are uniform with all the rows and columns, then the codes are called regular LDPC codes such as the LDPC codes that were presented by Gallager. With the non-uniform r_w and c_w, the codes are called irregular codes [17]. It has been shown that irregular LDPC codes are able to achieve a better performance compared to the regular LDPC code [18]. The performance of LDPC decoder can be efficiently performed using Tanner graph [19]. This graph represents the H matrix and contains two sets of nodes, variable nodes and check nodes which represent the n bits of a code and the parity constraints respectively. Variable and check nodes are connected by edge connections where the number of edges in the graph is equal to the number of ones in H matrix. Fig. 1 shows an example of the Tanner graph for H matrix given as

$$H = \begin{bmatrix} 1\,1\,0\,0\,0\,1\,0\,1 \\ 0\,1\,1\,0\,1\,0\,1\,0 \\ 0\,1\,1\,1\,0\,0\,1\,1 \\ 1\,0\,0\,1\,1\,1\,0\,0 \end{bmatrix} \quad (1)$$

The columns are represented by eight variable nodes, and the rows are represented four check nodes. As shown in the figure, H has a row weight of 4 and a column weight of 2.

The LDPC code, x is constructed so that

$$Hx^T = 0\ (mod\ 2), \forall\ x\ \in c \quad (2)$$

x^T can be written as

$$x^T = [i \mid c] \quad (3)$$

Correspondingly, H matrix can be split into two matrices as

$$H = [A \mid B] \quad (4)$$

Also, (1) may be written as

$$Ai + Bi = 0 \quad (5)$$

Then, if the matrix B is non-singular, the LDPC code can be found as

$$c = B^{-1}Ai \quad (6)$$

The LDPC code performance is most related to the

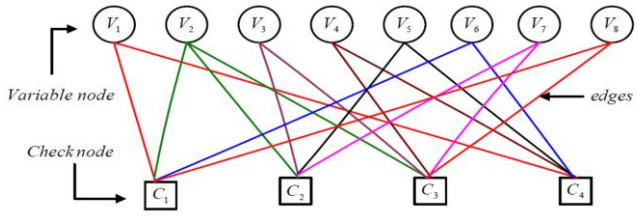

Fig. 1 Tanner Graph Representations of LDPC code

construction of H parameter. There are three significant properties of H matrix: row-column (RC) constraints, girth, and rank. The RC constraints represent the weight of ones in rows and columns represented in H matrix, where the number of RC constraints directly relates to the code length. The girth is the length of shortest cycles of Tanner graphs, while the rank is the maximum number of linearly independent row vectors of the matrix. Tanner graph with larger girth result in LDPC codes with better convergence and performance since the minimum distance is increased by increasing the girth. In the receiver, many algorithms are known to recover information such as [20]: 1) Sum-Product algorithm (SPA), 2) Min-Sum algorithm, 3) Min-Max algorithm, 4) Message-Passing algorithm and 5) Bit-Flipping Decoding algorithm. The SPA is based on belief propagation and has the best decoding performance, but with a high computational complexity [21]. This algorithm can be applied with low complexity and optimal performance using a simplified method presented in [22]. In this paper, all simulation results are obtained using the sum-product algorithm.

III. Convolutional Encoder

By convolutional encoder, codes come in a serial form instead of a block form. This encoder is represented by a set of shift registers and multiplexers. In general, it is specified by three main parameters, (n, k, and K), where n corresponding to the number of output bits, k represents the number of input bits, and K denotes the length constraint. The convolutional encoder combines the bits stored in shift registers to generate the code [20]. It can be recursive or non-recursive, systematic or non-systematic and characterized by different types of diagrams [24]. Fig. 2 shows an example of (2,1,4) convolutional encoder with a code rate of 1/2. It is a model for

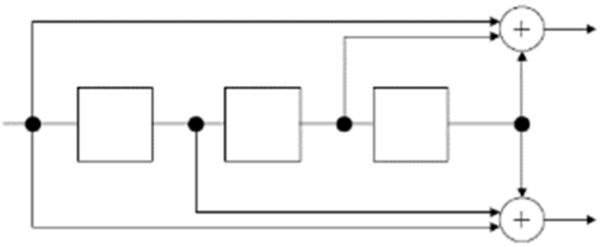

Fig. 2 a model of convolutional encoder.

the systematic and non-recursive encoder and comprises two branches of encoder each with distinct impulse responses.

The impulse response and the constraint length are two important parameters of the convolutional encoder. Parameter 1 specifies the linearity and causality of the system and is called generator sequence or generator polynomial. Parameter 2 refers to the maximum bits that may be affected, by any input bit. For instance, the impulse response streams $g_1(n)$ and $g_2(n)$ for the input $x(n) = (1\,0\,0\,0\,...)$ for the encoder shown in Fig. 2 can be introduced as follows:

$$g_1(n) = x(n) + x(n-2) + x(n-3) \quad (7)$$

$$g_2(n) = x(n) + x(n-1) + x(n-3) \quad (8)$$

In the sequence form, $g_1(n)$ and $g_2(n)$ can be represented as $g_1(n) = [1011]$ and $g_2(n) = [1101]$. Generator sequences are often specified in the octal form. For example g_1 and g_2 can be written as $g_1 = 13$ and $g_2 = 15$, and also can be specified in a polynomial form as

$$g_1(D) = 1 + D^2 + D^3 \quad (9)$$

$$g_2(D) = 1 + D + D^3 \quad (10)$$

where D is an operator related to the Z-transform. The linear combination of the two sequences can be described as

$$G_i = [g_1(D)\ g_2(D)] \quad 0 \le i \le r \quad (11)$$

The generator matrix of the convolutional encoder can be described in a matrix form as

ISSN: 1998-4464

$$G = \begin{bmatrix} G_0 & G_1 & G_2 \dots & G_r & 0 & 0 & \\ & G_0 & G_1 & G_2 \dots & G_r & 0 & \dots \\ & & G_0 & G_1 & G_2 \dots & G_r & \\ & & \vdots & & & & \end{bmatrix} \quad (12)$$

Then, the transform of the encoder output can be expressed as

$$Y(D) = X(D)\, G(D) \quad (13)$$

where $X(D)$ is the input sequence and

$$Y(D) = [Y_1(D)\, Y_2(D)] \quad (14)$$

For $g_1(n)$ = (1 0 1 1) and $g_1(n)$ = (1 1 0 1), the generator matrix is

$$G = \begin{bmatrix} 1\,1 & 0\,1 & 1\,0 & 1\,1 & 0\,0 & 0\,0 & 0\,0 & \\ 0\,0 & 1\,1 & 0\,1 & 1\,0 & 1\,1 & 0\,0 & 0\,0 & \dots \\ 0\,0 & 0\,0 & 1\,1 & 0\,1 & 1\,0 & 1\,1 & 0\,0 & \\ 0\,0 & 0\,0 & 0\,0 & 1\,1 & 0\,1 & 1\,0 & 1\,1 & \\ & & & \vdots & & & & \end{bmatrix} \quad (15)$$

If data sequence $u = [u_0\, u_1 \dots\, u_m]$, then block code can be found as:-

$$c = u\, G \quad (16)$$

To confirm a better degree distribution of ones and girth, and then provides improved BER performance, the length of the generator r must be selected so that it is no less than k/4.

IV. An Overview of Turbo Code for LTE

Turbo codes were invented in 1993 by Berrou [23]. They are the best available error correction technique in the last few years and standardized in the 3G standard because they can achieve near Shannon limit error correction [24]. The turbo code is a combination of two codes that work together. It is formed from a parallel concatenation of two convolutional encoders separated by an interleaver. These encoders are recursive and systematic in practice [20]. Turbo codes achieve a better performance than convolutional codes when the length of the interleaver is very large. Therefore, for improved code performance, a large block size random interleaver is required. However, the delay and the computational complexity of the encoder and decoder are increased when the length of the code is increased. At short block length code, the BER performance of turbo code is worse than that of a convolutional code while the computation complexity is similar. For many applications such as mobile systems, complexity and delay are important issues in choosing the length of block code. Many of interleaver designs are suggested to improve the performance of short length code [25]. Turbo code is the only channel coding used to process data in Long Term Evolution (LTE) and LTE Advanced standards. These standards provide mobile access technology for the latest mobile communication systems which are investigated to increase the capacity and throughput performance for 4G systems [26]. LTE is designed to be scalable, (i.e., it can be updated without disrupting current services) [27]. In LTE, coding is one parameter which is used to choose the order of modulation, where different modulation schemes are used to improve the throughput and achieve high data rates. The modulation schemes used in LTE standards are QPSK, 16QAM, and 64QAM depending on channel quality. The LTE turbo encoder as shown in Fig. 2, consists of three streams. The first stream represents systematic bits (S_k), while the second and third streams (P1 and P2) refer to the two parity bit streams. Each convolutional encoder also contains tail of 4 bits as trellis termination. For an input data bits of size K, the output of each convolutional encoder is a stream of length K+4, and the coding rate of the LTE code slightly less than 1/3. LTE uses 188 values for K. The smallest K is 40 and largest is 6144, where short length codes are specified to allow a better channel estimation [26]. For smallest input block size K=40 bits, the corresponding

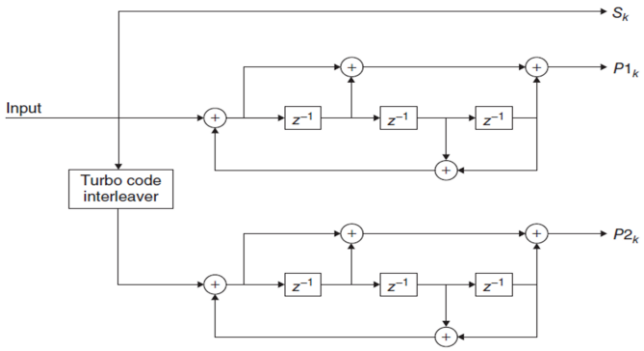

Fig. 3 a block diagram of turbo encoder

code length is 152 bits.

At the decoder, all operations of the encoder are inverted. Two decoders and two interleavers in a feedback loop are performed to decode information in the receiver. The same trellis structure and interleaver performed in the encoder are used in the decoder , but with an iterative operation where BER performance and complexity of the decoder directly relate to the number of iterations.

V. System Model

In communication systems, low BER performance is a necessary but not a sufficient requirement. The delay (time required for encoding and decoding) and hardware implementation of the encoder and decoder also need to be considered. Short length codes are specified in the latest communication systems as they allow better channel estimation and as they achieve improvements in terms of delay and complexity. A lot of studies have been done to achieve these requirements. In [27, 28], researchers propose several methods to reduce the decoding complexity of short length LDPC codes by growing the parity-check matrix and by updating the individual decoder algorithm, while achieving a code with simpler construction is still a challenge. In this work, we attempt to construct a short length code with low complexity and low delay and also having an improved performance when it is compared with LDPC and turbo codes through:

1) Using LDPC decoder at the receiver instead of using a turbo decoder. This has lead to simplifying the structure of the decoder since standard turbo decoder composes of two convolutional decoders and two interleavers while the LDPC decoder is performed by a structure of one component.
2) Using a non-systematic convolutional encoding technique

ISSN: 1998-4464

to encode the information instead of using LDPC block encoding. By standard method (block method), the LDPC encoder uses a large number of addition and multiplication operations to perform the LDPC code. As known, the number of operations is proportional to the size of H matrix and the number of the non-zero element of H matrix, as well as to the method used in encoding. Two encoding methods are known for LDPC block encoding [20]: preprocessing method and efficient encoding method. The two methods have a computational complexity of $O(n^2)$ and $O(n)$ respectively, where n is the code length. However, in the case of convolutional encoding, data bits come in a serial form instead of a block form, and only a small number of memory units, represented by a set of shift registers, are used in the encoding. Clearly, no logical operations are required in the encoder, and the generated code is identical to that code generated by the LDPC block encoder. In such a case, the proposed structure has a clear advantage in terms of hardware implementation and computational complexity.

3) Construct an efficient parity check matrix having a large girth by carefully specifying the parameters of the encoder. This can be accomplished by removing girth 4 when specifying the impulse response of the convolutional encoder. The parity-check matrix used in the decoder is derived from the generator matrix which is constructed from the generator sequence or impulse response of the convolutional encoder. In the SPA, the smaller the cycles the fewer the number of iterations that are correlation free, so maximizing girth of the matrix increases the number of correlation-free iterations and then the convergence of decoding will be enhanced [17, 29]. Clearly, high convergence means low delay decoding.
4) The impulse response of convolutional encoder is to be specified so that it leads to constructing irregular H matrix. This has advantages in terms of decoding complexity and BER performance since the irregular H matrices achieve a performance comparable with that of turbo codes and can be decoded with a very low-complexity iterative decoding scheme [30].
5) Avoiding the variable nodes that having a low number of edge connections in the Tanner graph by adding a sequence of zeroes as inputs at the end of the information. This increase the degree sequence of the node and then enhances the BER performance of the code since the performance of irregular LDPC code improves when the degree sequence has a large value, where the degree represents the number of edges connected to a node [17].
6) Increasing the signal to noise ratio of the received code before decoding and using an H matrix with extended size. At high signal to noise ratio, LDPC decoder can decode the information with the lowest number of average iterations. This certainly reduces the computational complexity and time required for the decoding and achieves optimum performance [31, 32]. As well as, the process of adding zeros of a certain size in the encoder requires extending the H matrix of the same size. This improves decoding performance since the LDPC decoder with large parity matrix achieves a better performance.

As introduced, the proposed model reduces the number of multiplications and additions required in the encoder by replacing the LDPC block encoding with only a set of the shift register, the receiver uses LDPC decoder instead of using a turbo decoder, and its constructed H matrix has an irregular Tanner graph with a large girth. By this model, hardware structure is minimized, the computational complexity and delay are reduced, and the code BER performance is improved compared to the turbo and LDPC codes. The generated code is a short length code and based on convolutional encoder and LDPC decoder, thus named SLC-LDPC code. The two following sections show how encoder and decoder are constructed.

A. Encoder Construction

Here, we attempt to construct a recursive and systematic encoder. The recursive encoding means that the encoder repeats the encoding procedure after a specific length of information. The code is systematic in that, the data bits are a part of the code and the parity-check matrix is a repeat-accumulate matrix like to the matrix of standard WiMAX code [11]. These considerations simplify the structure, implementation and allow the code to be easily encoded and decoded. In the convolutional encoder, the code is generated using a set of shift registers which are implemented according to the impulse response or generator sequence. Generator sequence is a key parameter of the proposed code since it specifies the configuration of the encoder and the decoder. The block diagram depicted in fig. 4 shows the simplest way to generate SLC-LDPC code. Firstly, the information is encoded into two pairs of the parity bit, and secondly, the information is added at the end of the code by the multiplexer. The memory unit is only used to illustrate that the information

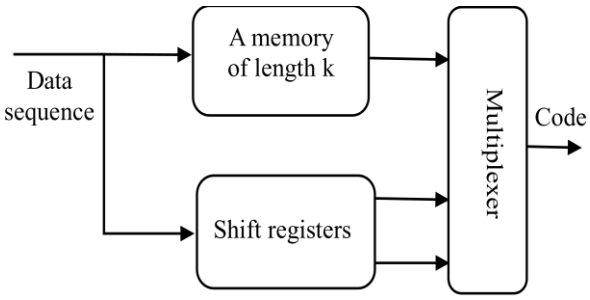

Fig. 4 Block diagram of the SLC-LDPC code generator.

is separated from the parity bits and the code generated is similar to the WiMAX LDPC code.

Number citations consecutively, in square brackets [1]. The sentence In addition to data bits, a sequence of zero bits of length Li is added at the end of the information and provided as input to the shift registers. This addition shifts data sequence by Li times and increases the parity sequence from k

ISSN: 1998-4464

to k+Li, where k is the size of information bits. The most important benefit of adding zero bits is that it allows for providing a code with variable lengths and multi-rates. As well as constructing a code that is compatible with LTE turbo codes in terms of code length and rate. As a result, parity bits are produced separately from information using a convolutional encoder, and then the information is added at the end of parity bits to perform the SLC-LDPC code. Fig. 2 shows an example of a convolutional encoder with a code rate of 1/2 and specific generator sequences. In general, the generator matrix of the proposed code can be obtained using the following procedure.

Let $g^1 = [g_0^1\, g_1^1 \ldots g_r^1]$ and $g^2 = [g_0^2\, g_1^2 \ldots g_r^2]$ then the linear combination of two sequences corresponding to (11) is $G_i = [g_0^1\ g_0^2\ g_1^1\ g_1^2 \ldots g_r^2]$, where r is the degree of generator polynomial and $0 \le i \le r$. The generator matrix for infinite data inputs using (12) can be found as

$$M = \begin{bmatrix} g_0^1 & g_0^2 & g_1^1 & g_1^2 & \ldots & g_r^2 & 0 & \ldots & 0 & 0 & 0 \\ 0 & 0 & g_0^1 & g_0^2 & g_1^1 & g_1^2 & \ldots & g_r^2 & 0 & \ldots & 0 \\ 0 & 0 & 0 & 0 & g_0^1 & g_0^2 & g_1^1 & g_1^2 & \ldots & g_r^2 & \\ & & & & \vdots & & & & & & \end{bmatrix} \ldots \quad (17)$$

The structure of such a matrix provides recursive encoding, where for an infinite number of input symbols, the encoder generates an infinite number of encoding symbols. This method of encoding significantly reduces encoding time and computational complexity compared to block code encoding. For k data inputs and Li zero inputs, the convolutional generator matrix, G_c can be expressed as

$$G_c = M[Li + k, 2(Li + k)] \quad (18)$$

Adding the data sequence to the output of the convolutional encoder, M matrix can be rewritten as

$$M_s = \begin{bmatrix} 1 & 0 & 0 & g_0^1 & g_0^2 & g_1^1 & g_1^2 & \ldots & g_r^2 & 0 & \ldots & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & g_0^1 & g_0^2 & g_1^1 & g_1^2 & \ldots & g_r^2 & 0 & \ldots & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & g_0^1 & g_0^2 & g_1^1 & g_1^2 & \ldots & g_r^2 & \\ & & & & & & & \vdots & & & & & & \end{bmatrix} \ldots \quad (19)$$

Then, the equivalent generator matrix of the encoding can be expressed as

$$G_e = M_s[Li + k, 3(Li + k)] \quad (20)$$

Since the zero sequence is not a part of the sent information, the actual generator matrix of the SLC-LDPC encoder is

$$G_a = M_s[Li + k, 2Li + 3k] \quad (21)$$

Clearly, G matrix directly relates to configuration and length of the impulse response of convolutional encoder, and the generated code bof this method is identical to that code generated by LDPC block code. To optimize code efficiency, the length of generator sequence is selected so that it is not less than k/4 and the Tanner graph is free from girth 4.

B. Decoder Construction

In the receiver, LDPC decoder is employed to decode the information from the received code. The H matrix is the most important parameter in LDPC decoder where constructing an efficient H matrix is the key to designing a good LDPC decoder. The H matrix is constructed using the generator matrix which was extracted from the impulse response of convolutional encoder. Generator matrix was constructed so that it leads to produce an efficient H matrix for the decoder. Using encoder parameters, H matrix can be obtained as follows.

The equivalent generator matrix which is represented by (20) can be written as

$$G_e = [\, I \mid P] \quad (22)$$

where I is identity matrix correspond to the data bits and P is a matrix represents parity-check s.

$$\text{Select } [\, I \mid P] = G_e[k + Li, n] \text{ such that } n \ge 3(k + Li) \quad (23)$$

$$\text{Find } H[\mathrm{n} - \mathrm{k} - \mathrm{Li}, \mathrm{n}] = [\, P^T \mid I] \quad (24)$$

H matrix is derived from the equivalent generator matrix; however, to correctly decode the received SLC-LDPC code, the structure also requires adding a number of zero inputs of size Li to the decoder corresponding to that added through the encoding process. This process improves decoder efficiency and provides codes with variable lengths and multi rates. Clearly, if n = 3k and no zero inputs are considered in the code (Li=0), then the constructed H matrix has a size of (k, 3k) and the code rate exactly equals to 1/3. The length, rate, as well as the performance of the code, will be changed when the Li is changed. For an efficient H matrix, several factors need to be considered through the construction of impulse response or generator sequence for the convolutional encoder.

1) The degree of the impulse response, r must be larger than k/4 to ensure a good degree distribution of ones in H,
2) To remove girth 4 from the Tanner graph, the code that is performed by the linear combinations of the two impulse responses of the two branches must be prefix-free, that is, this code is free from similar individual codes.
3) The sparsity of H matrix is required for the high algorithmic efficiency. So, efficient encoding can be satisfied if ones are sparse in the generated sequence.

Now, we have selected a simple example to verify the theory and to explain how the code is constructed. We consider a convolutional encoder with 1/2 code rate and (n, k, K) = (2, 1, 4). This encoder has one input bit, two output bits, and aconstraint length of four. The impulse response that is selected for the two branches are

$$g^1 = 11 = 1 + D^2 = [1\ 0\ 1\ 0] \quad (25)$$

$$g^2 = 15 = 1 + D^3 = [1\ 0\ 0\ 1] \quad (26)$$

The linear combination of the two encoders is

$$g = [1\ 1\ 0\ 0\ 1\ 0\ 0\ 1] \quad (27)$$

This encoder can be performed by a simple set of shift

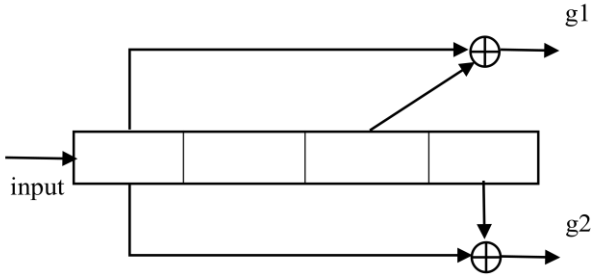

Fig. 5 a block diagram of the convolutional encoder

ISSN: 1998-4464

registers as in fig. 4.

Suppose Li=0, the convolutional generator matrix is

$$G_c = \begin{bmatrix} 1\,1\,0\,0\,1\,0\,0\,1\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0 \\ 0\,0\,1\,1\,0\,0\,1\,0\,0\,1\,0\,0\,0\,0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,1\,1\,0\,0\,1\,0\,0\,1\,0\,0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0\,0\,1\,1\,0\,0\,1\,0\,0\,1\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0\,0\,0\,0\,1\,1\,0\,0\,1\,0\,0\,1\,0\,0 \\ 0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,1\,1\,0\,0\,1\,0\,0\,1 \\ \vdots \end{bmatrix} \quad (28)$$

For input data bits k=6, the equivalent generator matrix for the SLC-LDPC encoder can be calculated according to (20).

$$G_e = \begin{bmatrix} 1\,0\,0\,0\,0\,0\,1\,1\,0\,0\,1\,0\,0\,1\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0 \\ 0\,1\,0\,0\,0\,0\,0\,0\,1\,1\,0\,0\,1\,0\,0\,1\,0\,0\,0\,0\,0\,0\,0 \\ 0\,0\,1\,0\,0\,0\,0\,0\,0\,0\,1\,1\,0\,0\,1\,0\,0\,1\,0\,0\,0\,0\,0\,0 \\ 0\,0\,0\,1\,0\,0\,0\,0\,0\,0\,0\,0\,1\,1\,0\,0\,1\,0\,0\,1\,0\,0\,0\,0 \\ 0\,0\,0\,0\,1\,0\,0\,0\,0\,0\,0\,0\,0\,0\,1\,1\,0\,0\,1\,0\,0\,1\,0\,0 \\ 0\,0\,0\,0\,0\,1\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,1\,1\,0\,0\,1\,0\,0\,1 \end{bmatrix} \quad (29)$$

In this example, we consider k=6 bits and n=3k so that the code rate is 1/3 and the corresponding code length is 18 bits. The systematic parity-check matrix has a size of (12, 18) and it can be performed using (24).

$$H = \begin{bmatrix} 1\,0\,0\,0\,0\,0\,1\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0 \\ 1\,0\,0\,0\,0\,0\,0\,1\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0 \\ 0\,1\,0\,0\,0\,0\,0\,0\,1\,0\,0\,0\,0\,0\,0\,0\,0\,0 \\ 0\,1\,0\,0\,0\,0\,0\,0\,0\,1\,0\,0\,0\,0\,0\,0\,0\,0 \\ 1\,0\,1\,0\,0\,0\,0\,0\,0\,0\,1\,0\,0\,0\,0\,0\,0\,0 \\ 0\,0\,1\,0\,0\,0\,0\,0\,0\,0\,0\,1\,0\,0\,0\,0\,0\,0 \\ 0\,1\,0\,1\,0\,0\,0\,0\,0\,0\,0\,0\,1\,0\,0\,0\,0\,0 \\ 1\,0\,0\,1\,0\,0\,0\,0\,0\,0\,0\,0\,0\,1\,0\,0\,0\,0 \\ 0\,0\,1\,0\,1\,0\,0\,0\,0\,0\,0\,0\,0\,0\,1\,0\,0\,0 \\ 0\,1\,0\,0\,1\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,1\,0\,0 \\ 0\,0\,0\,1\,0\,1\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,1\,0 \\ 0\,0\,1\,0\,0\,1\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,1 \end{bmatrix} \quad (30)$$

The Tanner graph representation of the H matrix is shown in Fig. 6. The rectangles represent check nodes and the circles represent variable nodes. The empty circles correspond to data bits while the filled circles denote parity bits.

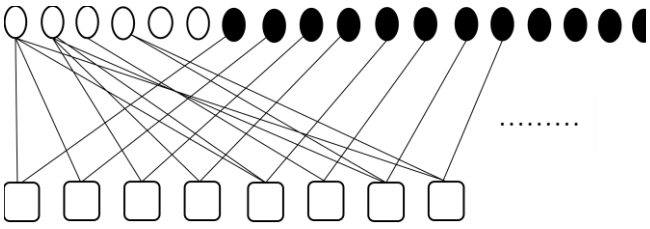

Fig. 6 Tanner graph for the H matrix

From the H matrix and its Tanner graph, we can note that the parity check matrix has a recursive structure, that is, if the code length is increased by the encoder, the decoder requires only a number of edges to be added to the structure in a recursive manner. Such adding is easy to implement since the edge connections of the added nodes are similar to the edge connections that exist in the graph. This can be done in parallel and thereby permitting fully parallel or partially parallel decoder construction. In addition, by this approach, a range of code rates and code lengths can be achieved so that the code proposed can be considered as having variable length and multi-code rates. To illustrate that the proposed H matrix has a recursive structure, the edge connections from the seventh parity node are depicted in Fig. 7. This figure shows how edge connections from parity nodes to variable nodes are repeated over the code length in a recursive manner.

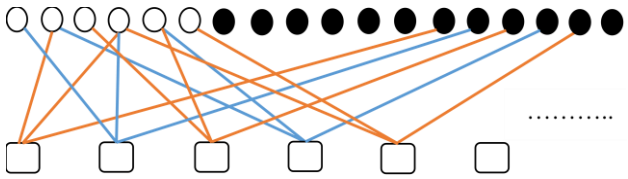

Fig. 7 Tanner graph for the H matrix starting from the seventh parity node

There are six authors or more. Use a space after authors' initials. The characteristics of this structure can be summarized as follows:

1) The code is systematic since the information is a part of it.
2) The decoder has a recursive structure so that the parity bits are determined depending on data bits and encoded parity bits. This provides significant advantages in terms of implementation complexity and delay.
3) H is a random and irregular matrix where the edge connection with the variable node connects to a check node one position away. This reduces delay and improves BER performance.
4) Girth 4 is removed from the H matrix and the Tanner graph has a longer girth. This leads to improve BER performance because minimum weight, distance increases by increasing the girth, and also reduces delay since it reduces the number of iterations required in the decoding.
5) The convolutional encoder can be constructed so that the generated SLC-LDPC codes have a good minimum distance, and also the Tanner graph of these codes contains edge connections of a short cycle.
6) By a process of shifting the information, the proposed structure is capable of providing codes with variable lengths and multi rates.
7) The structure can avoid variable nodes that have a low number of edges in the Tanner graph and can then provide codes with a high BER performance.
8) The process of adding zero bits at the decoder also improves decoding performance and reduces decoding delay because this process increases the signal-to-noise ratio (SNR) of the received code, and also because the added bits represent symbols of a high probability

VI. Simulation and Results

We evaluate the BER performance of the proposed code over a range of code lengths and different code rates using additive white Gaussian noise channel and QPSK modulation scheme. All simulation results are obtained using a specific

ISSN: 1998-4464

generator sequence and a sum-product decoding algorithm. We first simulate the performance of the SLC-LDPC code to prove the idea by comparing a standard SLC-LDPC code (Li=0 and R=1/3) with un-coded QPSK modulation signals and LDPC standard codes. Then, we reconstruct the code to be compared with the turbo codes that are used in the LTE standard. For fair comparisons, all simulations are done using the same code rate and length. The BER performance of the code directly relates to length and configuration of the impulse response. Unlimited configurations of impulse responses can be used through encoder construction. However, the impulse responses considered are selected so that they satisfy the requirements were illustrated in the research. Consider two non-recursive convolutional encoders with two generator sequences g1 = [1 0 0 1 0 1 0 0 0 0 0 0] and g2 = [1 1 0 0 0 0 0 1 0 0 1 1]. The length of each sequence is 12 and the linear combination is g = [1 1 0 1 0 0 1 0 0 0 1 0 0 0 0 1 0 0 0 0 0 1 0 1]. If the number of zero inputs, Li and the data bits, k are set to 0 and 12 respectively, then the corresponding code length is 36 bits and the code rate is exactly 1/3. Fig. 8 shows BER performance when QPSK modulator is used. It can be seen that the BER performance of the proposed code is identical to that achieved by LDPC block encoder (standard encoding method). Also, the curves show that the proposed code achieves much better BER performance than the un-coded signal and the code that is generated using the parity-check matrix considered in the IEEE 802.16 standard. These results indicate that the proposed structure can be applied to provide

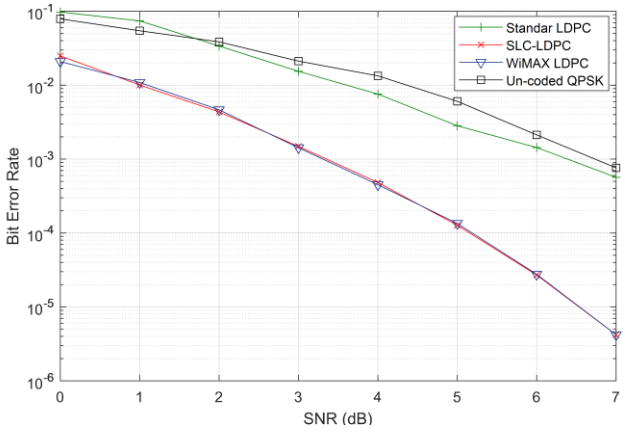

Fig. 8 BER performances using Li=0 and k=12 bits.

improved codes of the LDPC codes.

Another construction of the proposed codes is performed using generator sequences of g1 = [1 0 0 1 0 1 1] and g2 = [1 1 0 0 0 0 1] with a corresponding linear combination of g = [1 1 0 1 0 0 1 0 0 0 1 0 1 1]. The convolutional encoder is equipped with a sequence of zero inputs of 6 bits. So, the generated code has a length of 48 bits, comprising three streams: 12 bits as an information, 24 bits as parity bits corresponding to the information, and 12 bits as parity bits come from using a sequence of zero inputs of 6 bits. The decoding performance of the proposed code with irregular LDPC code are depicted in fig. 9. In this case, the code length is 48 bits and the code rate is exactly 1/4.

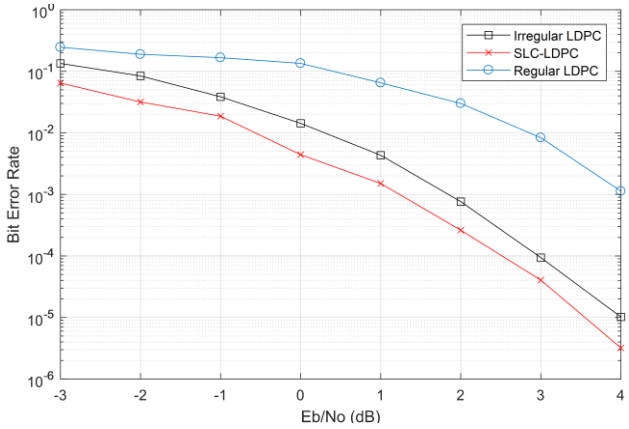

Fig. 9 BER performances with Li=6 bits, k=12 bits and

It has been shown that, among various LDPC codes, irregular DPC codes achieve the best BER performance. However, Fig. 9 shows that the BER performance with SLC-LDPC code is better than irregular LDPC code, as well as regular LDPC code. There are several factors behind this significant improvement in the BER performance: 1) the proposed code has a large minimum distance. 2) the constructed parity-check matrix has good randomness and irregular construction. 3) the Tanner graph is characterized by a large girth. 4) the zero inputs inserted at the decoder significantly increase the SNR of the received code, as they represent a received message with no error probability. 5) By the process of adding zeroes, variable nodes that have a low number of edges are avoided 6) the decoder uses a matrix of size (24 x 48) instead of (24 x 36), this leads to a better decoding performance because the performance of LDPC decoder directly relates to the size of parity-check matrix. In LTE, the turbo encoder adds 12 bits to the code as trellis termination, this means, for the smallest block size of information (k=40 bits), the corresponding turbo code is 132 bits, and for k=12, turbo code has a length of 48 bits.

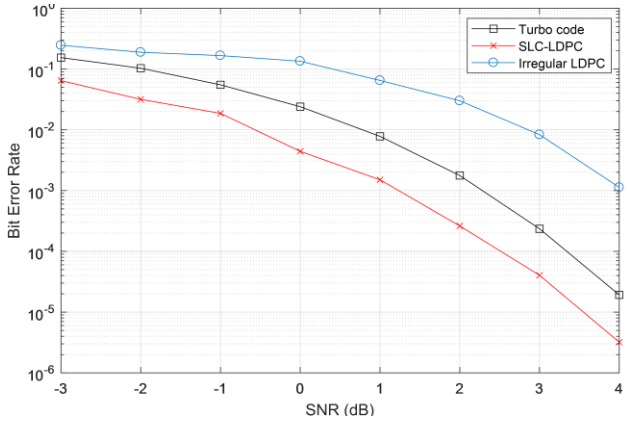

Fig. 10 BER performance of SLC-LDPC code comparing with turbo code used in LTE and irregular code when code length is 48 bits.

ISSN: 1998-4464

One of the main objectives of this research is to find an efficient short length code that can outperform the turbo codes used in LTE in terms of performance and complexity. Therefore, it is necessary to evaluate the performance of the proposed code at the small code lengths and then compare the results with those of turbo codes. If the zero inputs are set to 6 bits, the parity bits added to the proposed code are 12 bits and the generated code will be comparable to that of the turbo code. Fig. 10 shows the BER performances of SLC-LDPC code and turbo code when k=12 bits and the corresponding code length is 48 bits, while impulse responses used are g1 = [1 0 0 1 0 1 1] and g2 = [1 1 0 0 0 0 1]. It can be seen that SLC-LDPC code achieves a good BER performance and also outperforms LTE turbo code as well as irregular code when they are simulated under the same conditions. The BER performances obtained by using impulse responses of g1 = [1 0 0 1 0 1 1 0 0 0] and g2 = [1 1 0 0 0 0 0 1 0 1] when the information bits are set to 24 and 40 bits are shown in fig.'s 11 and 12 respectively. The two figures also show that the SLC-LDPC code is able to provide better performance than a turbo

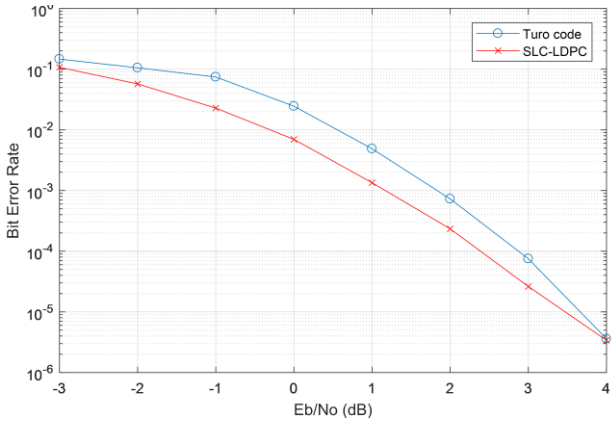

Fig. 11 BER performance of SLC-LDPC code comparing with turbo code when k=24 bits and corresponding code length is 84 bits

and irregular codes.

VII. Conclusion

Despite LDPC codes can provide significant advantages in terms of complexity and delay, achieving LDPC codes with low complexity and low delay at short block lengths remains a challenge. In this work, we proposed a short length code based on a convolutional encoder and LDPC decoder. The parity-check matrix used in the decoder was derived from the generator matrix of the convolutional encoder. Several constraints were considered through designing the generator matrix at the encoder in order to make the parity-check matrix simple and efficient. All simulation results were obtained using an AWGN channel, QPSK modulation schemes, and sum-product decoding algorithm. The proposed code was evaluated and compared with that of regular and irregular LDPC codes as well as turbo codes which are used in the LTE standard. Three significant issues are targeted by this code, BER performance, delay, and complexity. We analytically demonstrated that the proposed structure can significantly reduce computational complexity and delay of encoding by using a convolutional encoder instead of using LDPC block encoder. While delay and complexity of the decoder are reduced by constructing an improved generator matrix so that the parity-check matrix constructed can achieve these requirements. Moreover, BER performances depicted in the figures showed that the proposed code, at a BER of (1 x 10-5), outperforms regular and irregular LDPC codes by 0.4 dB at code rates 1/3 and 1/4 respectively, and outperforms turbo code by 0.2 dB.

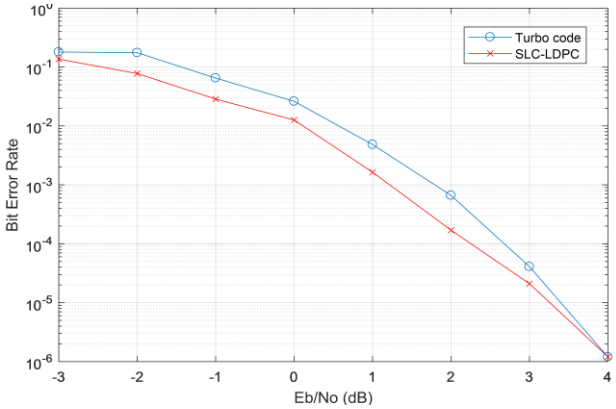

Fig. 12 BER performance of SLC-LDPC code comparing with turbo code when k=40 bits and corresponding code length is 132 bits

References

[1] Y. Sun, M. Karkooti and J. R. Cavallaro, " VLSI Decoder Architecture for High Throughput, Variable Block-size and Multi-rate LDPC Codes," *EEE International Symposium on Circuits and Systems (ISCAS'07)*. May 2007.

[2] R. G. Gallager, "Low-Density Parity-Check Codes," Cambridge, Massachusetts: M.I.T. Press, 1963.

[3] Chia-Yu Lin, "Design of Low-Density Parity Check Coding Systems," Ph.D. Thesis, Department of Computer Science and Information Engineering, College of Electrical Engineering and Computer Science, National Taiwan University, March, 2011.

[4] Ch. Prasartkaew and S. Choomchuay, "A Parity Check Matrix Design for Irregular LDPC Codes with 2K Block Length," *International Symposium on Intelligent Signal Processing and Communication Systems*, pp. 350-355, December, 2009.

[5] A. E.S. Hassan, M. Dessouky, A. A. Elazm and Mona, "Evaluation of Complexity Versus Performance for Turbo Code and LDPC Under Different Code Rates," *The Fourth International Conference on Advances in Satellite and Space Communications*, pp. 98-103, Egypt, 2012.

[6] N. ul Hassan, M. Lentmaier, and G. P. Fettweis, "Comparison of LDPC Block and LDPC Convolutional Codes Based on their Decoding Latency," *7th International Symposium on Turbo Codes & related Topics (ISTC 2012)*, Gotheburg, Schweden, August, 2012, pp. 27-31.

[7] A. H. Khan and Dr K C Roy, " Comparison of Turbo Codes and Low Density Parity Check Codes," *IOSR Journal of Electronics and Communication Engineering (IOSR-JECE)*, vol. 6, no. 6, pp. 11-18, Jul. - Aug. 2013.

[8] K. Narwal, Y. Sharma, " Performance Comparison of Turbo Codes with other Forward Error Correcting Codes,", *International Journal of Electronics and Computer Science Engineering, IJECSE*, vol. 1, no. 2.

[9] P. Shah, P. Vyavahare and A. Jain, " Modern error correcting codes for 4G and beyond: Turbo codes and LDPC codes," *Conference of IEEE Radio and Antenna Days of the Indian Ocean (RADIO)*, vol. 3, Belle Mare, Mouritius, September, 2015.

[10] A. E. Pusane, R. Smarandache, P. O. Vontobel and D. J. Costello, " Deriving Good LDPC Convolutional Codes from LDPC Block

Codes," *IEEE TRANSACTIONS ON INFORMATION THEORY*, September, 2010.
[11] T. Brack, M. Alles, F. Kienle and N. When, " A SYNTHESIZABLE IP CORE FOR WIMAX 802.16E LDPC CODE DECODING," *The 17th Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC'06)*. October 2006.
[12] Zhongwei Si, " Structured LDPC Convolutional Codes," Ph.D Thesis, School of Electrical Engineering Communication Theory Laboratory, Stockholm, Sweden, 2012.
[13] Z. Si, S. Wang, and J. Ma, " An Efficient Method to Construct Parity-Check Matrices for Recursively Encoding Spatially Coupled LDPC Codes," *Extension of the paper published in the 2015 IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, Hong Kong, China, 30 August–2 September 2015., August 2016.
[14] Th. J. Richardson and R. L. Urbanke, " The Capacity of Low-Density Parity-Check Codes Under Message-Passing Decoding," *IEEE TRANSACTIONS ON INFORMATION THEORY*, vol. 47, no. 2, February, 2012.
[15] A. E. Pusane, A. J. Feltstr¨om and A. Sridharan, " Implementation Aspects of LDPC Convolutional Codes," *IEEE TRANSACTIONS ON COMMUNICATIONS*, vol. 56, no. 7, pp. 1060-1068, July, 2008.
[16] Hend A. Orabi, A. Zekry and G. Gomah, "Implementation for Two-Stage Hybrid Decoding for Low Density Parity Check (LDPC) Codes," *International Journal of Computer Applications*, vol. 80, no. 3, October, 2013.
[17] Xiao–Yu Hu, E. Eleftheriou, and D. M. Arnold, "Regular and Irregular Progressive Edge-Growth," *IBM Research*.
[18] T. B. Iliev, G. V. Hristov, P. Z. Zahariev and M. P. Iliev, " Application and Evaluation of the LDPC Codes for the Next Generation Communication Systems,", T. Sobh et al. (eds.), "*Novel Algorithms and Techniques in Telecommunications, Automation and Industrial Electronics*," Springer Science+Business Media, pp. 532–536, 2008.
[19] Andrew James Maier, " Design and Optimization of Decoders for Low-Density Parity Check Codes Synthesized from OpenCL Specifications," M.Sc. Thesis, Department of Electrical and Computer Engineering, University of Alberta, 2016.
[20] K. Deergha Rao, " Channel Coding Techniques for Wireless Communications," Springer India, 2015, ch. 8.
[21] David J. C. MacKay, " Good Error-Correcting Codes Based on Very Sparse Matrices," *IEEE TRANSACTIONS ON INFORMATION THEORY*, vol. 45, no. 2, pp. 399-430, March, 1999.
[22] S. Papaharalabos and P.T. Mathiopoulos, " Simplified sum-product algorithm for decoding LDPC codes with optimal performance," *ELECTRONICS LETTERS*, vol. 45, no. 2, January, 2009.
[23] Mrs.K.M.Bogawar, Assistant Professor, Ms.Sharda Mungale and Dr.Manish Chavan, "Implementation of Turbo Encoder and Decoder," *International Journal of Engineering Trends and Technology (IJETT)*, vol. 8, no. 2, pp. 73-76, February, 2013.
[24] G.gen, "Comparison of simulation results for different Coding Techniques (Uncoded, Reed-Solomon, Reed-Solomon plus Trellis and Reed-Solomon plus Parallel Concatenated Convolutional Codes) for G.992.1.bis and G.992.2.bis," *ITU - Telecommunication Standardization Sector*, Boston, Massachusetts, 10-14 May 1999.
[25] H. R. Sadjadpour , N. J. A. Sloane and G. Nebe," Interleaver Design for Turbo Codes," *IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS*, vol. 19, no. 5, May, 2001.
[26] Dr Houman Zarrinkoub, "UNDERSTANDING LTE WITH MATLAB," John Wiley & Sons, 2014 , ch. 4.
[27] F. A. Newagy, Y. A. Fahmy, and M. M. S. El-Soudani, "NOVEL CONSTRUCTION OF SHORT LENGTH LDPC CODES FOR SIMPLE DECODING," *Journal of Theoretical and Applied Information Technology*, pp. 64-69, Egypt, 2007.
[28] Patel Sneha Bhanubhai, Mary Grace Shajan, Upena D. Dalal, "Performance of Turbo Encoder and Turbo Decoder for LTE," *International Journal of Engineering and Innovative Technology (IJEIT)*, vol. 2, no. 6, October, 2012.
[29] S. J. Johnson, "Introducing Low-Density Parity-Check Codes," Institute for Information Transmission (LIT), School of Electrical Engineering and Computer Science, The University of Newcastle, Australia
[30] Anbuselvi M, "Studies on Anna University, Various Algorithmic improvement in non-Binary LDPC Decoder Design," Ph.D. Thesis, October, 2015.
[31] Zhongxun Wang and Xinglong Gao, " The Min-max Decoding Algorithm of Nonbinary LDPC Codes Based on Early Stopping," *JOURNAL OF COMPUTERS*, vol. 9, no. 10, pp. 2488-2492, October, 2014.
[32] J. Chen, "Reduced-Complexity Decoding of LDPC Codes," *IEEE Transaction on Communication*, vol. 53, No. 8, pp. 1288-1298, August. 2005.

ISSN: 1998-4464