EXIT chart analysis of regular and irregular LDPC convolutional codes on AWGN channel

Oulfa Laouar, Imed Amamra, Nadir Derouiche

Laboratory LRES ICT Team, Department of Electrical Engineering, Faculty of Sciences and Technology, University of 20th August 1955, Skikda, Algeria

Article Info

ABSTRACT

Low-density parity-check (LDPC) codes are widely recognized for their excellent forward error correction, near-Shannon-limit performance, and support for high data rates with effective hardware parallelization. Their convolutional counterpart, LDPC convolutional codes (LDPC-CCs), offer additional advantages such as variable codeword lengths, unlimited parity-check matrices, and simpler encoding and decoding. These features make LDPC-CCs particularly suitable for practical implementations with varying channel conditions and data frame sizes. This paper investigates the performance of LDPC-CCs using the extrinsic information transfer (EXIT) chart, a graphical tool for analyzing iterative decoding. EXIT charts visualize mutual information exchange and help predict convergence behavior, estimate performance thresholds, and optimize code design. Starting with the EXIT chart principles for LDPC codes, we derived the mutual information functions for variable and check nodes in regular and irregular LDPC-CC tanner graphs. This involved adapting existing EXIT functions to the periodic parity-check matrix of LDPC-CCs. We compare regular and irregular LDPC-CC constructions, examining the impact of degree distributions and the number of periods in the parity-check matrix on convergence behavior. Our simulations show that irregular LDPC-CCs consistently outperform regular ones, and the EXIT chart analysis confirms that LDPC-CCs demonstrate superior bit error rate (BER) performance compared to equivalent LDPC block codes.

Corresponding Author:

Oulfa Laouar
Laboratory LRES ICT Team, Department of Electrical Engineering, Faculty of Sciences and Technology
University of 20th August 1955
Skikda, Algeria
Email: o.laouar@univ-skikda.dz

1. INTRODUCTION

One of the most extensively used techniques for ensuring reliable data transmission in communication networks is error correction codes [1]. Due to their excellent error-correction performance close to the Shannon limit and low complexity decoding algorithms, an advanced class of channel encoding scheme, is the low-density parity-check (LDPC) code. These codes have gained widespread attention in the 1990s after being overlooked for over 35 years, where the researchers community was attracted to analyze and design these codes [2] using message-passing (MP) decoding algorithm [3]. The improvement in the performance of LDPC codes was observed when irregular codes were derived from regular ones [4], defined in terms of the distribution of variable and check nodes degrees introduced by a graphical representation [5] known as the tanner graph.

LDPC convolutional codes are a type of error-correcting codes used in digital communications to improve the reliability of data transmission over noisy channels. They are a combination of two powerful coding techniques: convolutional coding and LDPC coding, which are designed to work with a continuous stream of data rather than discrete blocks. They are used in many communication systems and are particularly useful for applications with unbounded data sizes [6]. They were first introduced in the early 2000s [7] as a means of achieving higher data rates in satellite and wireless communication systems, where high-speed data transfer was crucial. In addition to their use in communications systems, LDPC-CCs have also been applied in other areas, such as magnetic storage [8] and optical communications [9]. Despite having a shorter convolutional constraint length than the block length, LDPC-CCs have the ability to achieve better performance compared to LDPC codes [10]. This has led to a growing interest in both theoretical research and software development, making LDPC-CCs increasingly popular.

The basic idea behind LDPC-CCs is to use the structure of convolutional codes, which allows for continuous transmission of information, and the error-correcting properties of LDPC codes, which provide strong error-correction capabilities. One of the key challenges in working with LDPC-CCs is finding efficient methods for encoding and decoding algorithms, as the convolutional structure of the code might cause this process more complicated than traditional LDPC codes [7]. Thus, numerous researchers have proposed various techniques for decoding and analyzing LDPC-CCs, including the convergence behavior of iterative belief propagation algorithms [11], [12] on both AWGN and the binary erasure channels (BEC) using the algorithm known as density evolution (DE). It is based on the calculation of the probability density function with different iterations for the threshold analysis of terminated LDPC-CCs in [11], where it has been shown that the use of the DE on the AWGN channel becomes increasingly difficult in terms of performance with large frame lengths. Mitchell *et al.* [12] discusses the construction of protograph-based spatially coupled LDPC (SC-LDPC) codes by coupling together into a single chain of multiple LDPC tanner graphs, resulting in optimized codes with best features in terms of fast convergence rates and close BP thresholds to Shannon limits.

Extrinsic information transfer (EXIT) charts are a powerful tool for analyzing and designing error correction codes, which are first introduced in [13] for analyzing concatenated codes. The use of the EXIT chart was later extended to other types of codes, such as LDPC codes [14] on the additive white Gaussian noise (AWGN) channel, where a method of combination between irregular LDPC codes with coding and modulation was introduced for multiple-input multiple-output (MIMO) fading channels using the EXIT chart tool with different degree distributions. Since then, the EXIT chart has become a widely used tool for analyzing the performance of iterative decoding algorithms for various types of codes, including LDPCs and LDPC-CCs on the BEC [15], [16] respectively. The key benefits of using EXIT charts for error correction codes are: performance prediction, code optimization, computational efficiency by the significant reduction of the computational complexity of designing and optimizing codes, and iterative decoding convergence analysis. There have been some studies on using convergence analysis of LDPC-CCs on the BEC. The work proposed by Sridharan *et al.* [17] have analyzed terminated LDPC-CCs in terms of the iterative convergence of the belief propagation decoding, where they found that irregular LDPC-CCs leads to better thresholds compared to regular ones.

The EXIT chart analysis of protograph (PEXIT) LDPC codes on both AWGN and BEC channels was presented in [18] with general EXIT functions according to those introduced in [14], where the results have shown the possibility of extension to other different types of channels. The PEXIT analysis has taking into consideration punctured variable nodes with different degree distributions and thresholds for protographs with different code rate regimes according to the spatially coupled structure. Based on these analysis techniques, the design of protograph-based LDPC convolutional codes in general [19] and the SC-LDPC convolutional codes in particular [20] have been conducted. Since LDPC-CCs are considered as a terminated type of protograph codes, we focused our goals in this work on the use of the EXIT chart for providing a detailed and intuitive analysis and design of LDPC-CCs iterative decoding on the AWGN channel, making this graphical tool essential for researchers and engineers to evaluate and optimize convergence behavior of such important error-correction codes.

Before analyzing the EXIT technique of LDPC-CCs, the basics of these codes need to be known. Thus, section 2 presents a brief overview of the main parameters of LDPC codes and LDPC-CCs with the Tanner graph details. In addition to, the unwrapping method to obtain the periodically parity-check matrix for LDPC-CCs is explained. Section 3 presents derivations and demonstrations of the EXIT chart obtained from the EXIT functions of LDPC codes. Afterwards, section 4 presents simulation results for the EXIT chart of both LDPCs and LDPC-CCs and compares the threshold convergence of regular and irregular codes. Finally, we summarize and conclude the obtained results in section 5.

2. CONCEPT OF LOW-DENSITY PARITY-CHECK CONVOLUTIONAL CODES

This section of the paper serves as an introduction to the fundamental concepts and notation that will be used in the rest of the paper. Firstly, a comprehensive definition of LDPC codes extended to LDPC-CCs and relevant parameters will be provided. The unwrapping technique will also be discussed as it plays a significant role when examining the EXIT functions of LDPC-CCs. The tanner graph of LDPC codes will be briefly reviewed, and then the tanner graph of LDPC-CCs, which can potentially be infinite, will be derived based on the period of the parity-check matrix.

2.1. Low-density parity-check codes

Definition 1: an $m \times n$ binary matrix H over GF (2), describes a binary (d_v, d_c)-LDPC code of length n with column and row weight d_v and d_c, respectively, and $'0s'$ elsewhere. The parameter R, such that $R = b/c$ $(b<c)$ is an integer, called the code rate, where b and c are obtained by $m/\gcd(m,n)$ and $n/\gcd(m,n)$ respectively. Let (1) be the 5×10 binary parity-check matrix of full-rank constructed from an $1/2$ code rate, with 3 $'1s'$ in each column and 6 $'1s'$ in each row.

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

Definition 2: the parity-check matrix in (1) can be represented by a tanner graph as seen in Figure 1. The graph consists of n variable nodes represented by circles and m check nodes represented by rectangles. Each variable node has three edges ($d_v = 3$) connected to the check nodes, while each check node has six edges ($d_c = 6$) connected to the variable nodes.
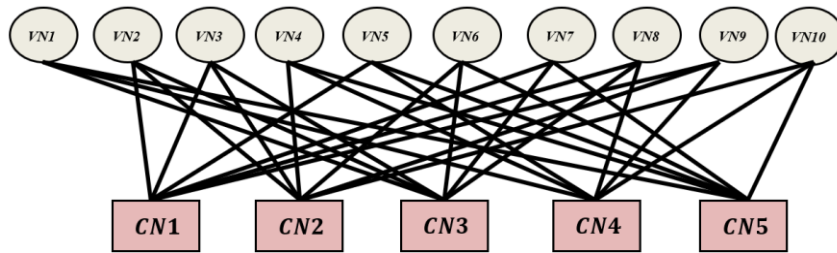

Figure 1. Tanner graph representing the parity-check matrix of (5, 10) LDPC code with the column weight $d_v = 3$ and the row weight $d_c = 6$

2.2. Low-density parity-check convolutional codes construction

Similar to LDPC codes, parity bits are generated by parity check equations in LDPC-CCs. However, the generation process in LDPC-CCs only uses previous systematic bits and parity bits.

Definition 3: let the (m_s, d_v, d_c) regular LDPC-CC be the set of sequences V satisfying the equation $H_{conv} \times V^T = 0$. To describe an LDPC-CC, the period T and the syndrome former memory $m_s = T - 1$ are typically used. The syndrome former memory is determined by the largest width of the non-zero area in the periodically parity-check matrix. H_{conv} which represents the periodically semi-infinite parity-check matrix is defined as (2):

$$H_{conv} = \begin{bmatrix} H_0(0) & & & \\ H_1(1) & H_0(1) & & \\ \vdots & \vdots & \ddots & \\ H_{m_s}(m_s) & \cdots & H_0(m_s) & \\ & \ddots & & \ddots & \\ & & H_{m_s}(t) & \cdots & H_0(t) \\ & & & \ddots & & \ddots \end{bmatrix} \tag{2}$$

where $H_t(0)$ must be of full rank for all possible $t's$.

Starting from the $m_s \times (c - b)$ column and from the $(m_s + 1) \times (c - b)$ row, H_{conv} in (2) has d_v non-zero elements in each column, which is referred to as column weight, and d_c non-zero elements in each row, referred to as row weight. These non-zero elements determine the connections between the corresponding variable and check nodes in the Tanner graph, as each column and row in H_{conv} represent a variable and check node respectivel as shown in Figure 2. The constraint length of H_{conv} is defined as $v_s = (m_s + 1) \times c$ which measures the maximum width of the non-zero bits in H_{conv}.

Figure 2. An example of $H_{conv}(3,6)$ in rate-1/2 LDPC-CC

The elements $H_i(t)$ of H_{conv} defined in (3) are binary $(c - b) \times c$ submatrices, where $(i = 0,1, \dots, m_s)$, $(t = 0,1, \dots, T - 1)$:

$$H_i(t) = \begin{bmatrix} h_i^{(1,1)}(t) & \cdots & h_i^{(1,c)}(t) \\ \vdots & & \vdots \\ h_i^{(c-b,1)}(t) & \cdots & h_i^{(c-b,c)}(t) \end{bmatrix} \tag{3}$$

2.2.1. Unwrapping technique

The process of construction LDPC-CCs from LDPC codes using the unwrapping approach is discussed in this subsection. In this approach, the parity-check matrix of the LDPC code is cut along its diagonal, and the lower-left and upper-right parts are swapped. This resulting matrix is then repeated infinitely, with corresponding vertical and horizontal shifts [7], to form the periodically parity-check matrix for the LDPC-CC.

Pusane *et al.* [6] has discussed different techniques for constructing families of LDPC-CCs, both time-invariant and time-varying, from LDPC codes. They also presented a modification of the method introduced in [7], which involved cutting, exchanging parts of the parity-check matrix and replicating it to form the H_{conv} matrix that defines the LDPC-CC. This modification aims to simplify the matrix construction process and limit the number of codeword bits used in calculating each check sum. This paper will use the same unwrapping method as described in [6], where an ½ code rate LDPC-CC is derived from a block code of the same rate.

For a parity-check matrix with dimensions $m \times n$ of a rate-1/2 block code, as indicated:

$$H = \begin{bmatrix} H_1(1) & H_1(2) & H_1(3) & H_1(4) & \cdots & H_1(n) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ H_m(1) & H_m(2) & H_m(3) & H_m(4) & \cdots & H_m(n) \end{bmatrix} \tag{4}$$

We cut the parity-check matrix moving 2 units to the right, 1 unit down and setting to zero the entire upper part of the matrix:

$$H_1 = \begin{bmatrix} H_1(1) & H_1(2) & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ H_m(1) & H_m(2) & H_m(3) & H_m(4) & \cdots & H_m(n) \end{bmatrix} \tag{5}$$

In the same way, and by setting to zero all the lower part of the matrix, we result:

$$H_2 = \begin{bmatrix} 0 & 0 & H_1(3) & H_1(4) & \cdots & H_1(n) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 0 \end{bmatrix} \tag{6}$$

Associating vertically the two matrices H_1 and H_2 obtained in (5) and (6), the generalization of this described method is straightforward as presented in Figure 3, by which we can construct large ensembles (m_s, d_v, d_c) of LDPC-CCs as seen in Figure 3(a). The finite H_{conv} shown in Figure 3(b) is obtained by using the diagonal cut to split the parity-check matrix presented by (1) into two parts (red and black) following the steps above. This matrix exhibits a pattern where the variable nodes remain constant at d_v over the repeated periods, similar to the original parity-check matrix of the LDPC code as depicted in Figure 2. On the other hand, the check nodes in H_{conv} change from 1 to d_c as opposed to the fixed row weight in the original parity-check matrix.
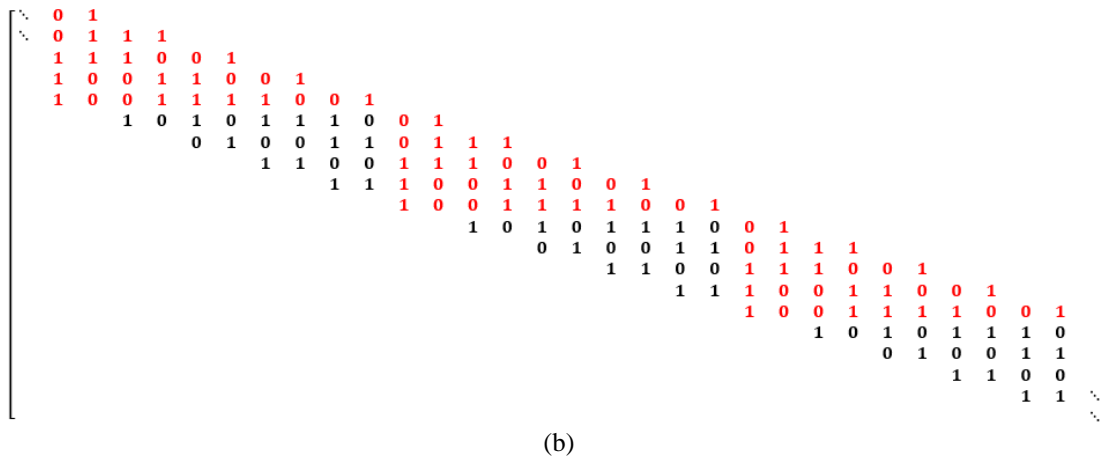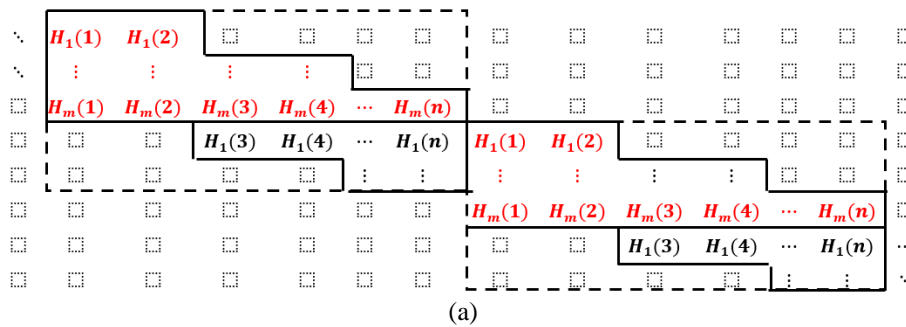

(a)

(b)

Figure 3. Periodically parity-check matrix of LDPC-CCs derived from block code of rate-1/2; (a) general unwrapping method and (b) example on unwrapping method

2.2.2. Tanner graph

The Tanner graph of an LDPC-CC based on its periodically parity-check matrix is infinite due to the semi-infinite nature of the matrix, as depicted in Figure 4. Thus, the distance between two variable nodes that correspond to (c-b) connected to the same check node corresponding to c (as depicted in (3) at each time instant t in the tanner graph) is constrained by the former memory of the code, and short cycles can exist at any point of the graph. As a result, determining the girth of a semi-infinite graph or matrix is impractical.

To determine the girth of an LDPC-CC, it is sufficient to examine one period of H_{conv}, which is possible due to the periodicity of the matrix. Short cycles in the Tanner graph can be avoided through the use of randomly constructed LDPC and LDPC-CCs that are primarily dependent on the sparsity of the parity-check matrix [21], in which, a specific construction for a family of parity-check matrices satisfying the absence of 4-cycles is presented. While the presence of a termination constraint results in a structural irregularity in the tanner graph, better threshold performance can be obtained compared to the corresponding LDPC codes, and we can therefore associate the concept of time with the tanner graph of terminated LDPC-CCs.
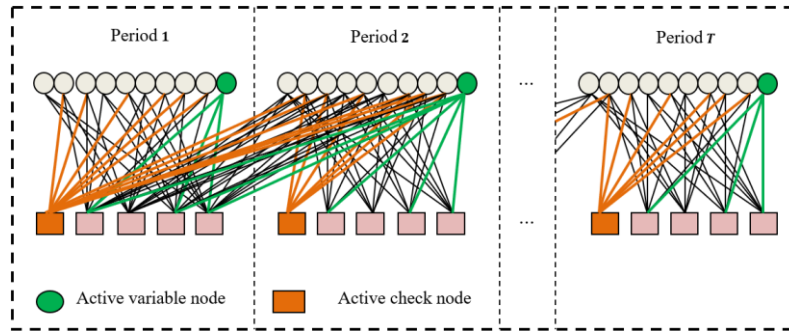
Figure 4. Semi-infinite tanner graph representing an LDPC-CC derived from (5, 10) LDPC code with
$(d_v, d_c) = (3,6)$

3. EXIT CHART GENERATION

Traditionally, the performance evaluation of iterative decoding algorithms for recent forward error correction (FEC) codes was mainly based on the analysis of BER curves, which are obtained by Monte-Carlo simulations. Recently, the convergence behavior of iterative decoding schemes is visuliazed using the EXIT chart's graphical tool. The decoder's EXIT chart includes two curves, one for the variable node (VND) and the other for the check node (CND). Each curve compares the output extrinsic information of a node with the input a priori information to that node.

In the following subsections, we derive the EXIT functions of regular and irregular LDPC-CCs from those of LDPC codes, and we gave examples with different periods of the LDPC-CC parity-check matrix. In addition to, analysis principle of all possible cases of the EXIT curves are detailed to explain the convergence behavior of the iterative decoding in LDPC-CCs.

3.1. EXIT function of regular LDPCs and LDPC-CCs

The regular LDPC parity-check matrix H of a rate $R = 1 - d_v/d_c$ gives a Tanner graph with m check nodes (CNDs) and n variable nodes (VNDs), whereas the regular LDPC-CC parity-check matrix H_{conv} with T periods give $m \times T$ CNDs with d_c degrees, and $n \times T$ VNDs with d_v degrees. Considering now the calculation of EXIT functions for both VNDs and CNDs, which are the two update rules for describing the decoding trajectory and visualizing the convergence of the iterative decoding procedure as following.

3.1.1. Variable node's EXIT functions

The average extrinsic information at the variable node decoder's output for LDPC codes that can be written in the form of $I_{E,V}$ under the symmetric-Gaussian assumption for VND inputs, and parameterized by E_b/N_0 for a given code rate R can be demonstrated from [22] as:

$$I_{E,V} = J(\sqrt{(d_v - 1)\sigma_A^2 + \sigma_{ch}^2}) \tag{7}$$

Since σ_{ch}^2 is the variance of the channel AWGN given by:

$$\sigma_{ch}^2 = 8R \, E_b/N_0 \tag{8}$$

And σ_A^2 is the variance of the a priori information. The log-likelihood ratio (LLR) in this case is Gaussian with variance:

$$\sigma^2 = \sigma_{ch}^2 + (d_v - 1)\sigma_A^2 \tag{9}$$

That results:

$$(d_v - 1)\sigma_A^2 = \sigma^2 - \sigma_{ch}^2 \tag{10}$$

Replacing (10) in $I_{E,V}$ given by (7):

$$\begin{aligned} I_{E,V} &= J\left(\sqrt{\sigma^2 - \sigma_{ch}^2 + \sigma_{ch}^2}\right) \\ &= J(\sqrt{\sigma^2}) \\ &= J(\sigma) \end{aligned} \tag{11}$$

From (7) and (11), $I_{E,V}$ can be written as:

$$I_{E,V} = J(\sigma) = J(\sqrt{(d_v - 1)\sigma_A^2 + \sigma_{ch}^2}) \tag{12}$$

In the general case, $J(x)$ is the mutual information density function of the standard Gaussian distribution given in [22] by:

$$J(x) = 1 - \int_{-\infty}^{+\infty} \frac{1}{\sqrt{2\pi}x} e^{-(l - \frac{x^2}{2})^2/2x^2} \log(1 + e^{-l}) dl \tag{13}$$

Where $x = \sigma$ in this case, and $J(\sigma)$ is the mutual information of the message with a symmetric Gaussian density probability density function (PDF).

For LDPC-CCs, the average extrinsic information at the variable node decoder's output can be written as I_{E,V_T}, and can be obtained using the similar equation as that of LDPC codes as shown in (12), because VND degrees (d_v) are the same in H_{conv} $\forall T$ as seen in Figure 2, and are equal to those of LDPC codes. Then:

$$I_{E,V_T} = J(\sigma) = J(\sqrt{(d_v - 1)\sigma_A^2 + \sigma_{ch}^2} \tag{14}$$

where:

$$\sigma = \sqrt{(d_v - 1)\sigma_A^2 + \sigma_{ch}^2} \tag{15}$$

The mutual information $I_{A,V}$ between the VND inputs (a priori) and the associate bit code with this VND can be demonstrated applying the symmetric-Gaussian assumption as follows.
Starting from (10) we can write:

$$\sigma_A = \sqrt{\frac{\sigma^2 - \sigma_{ch}^2}{(d_v - 1)}} \tag{16}$$

In this case, the a priori information is $x = \sigma_A$ in (13). Thus, $I_{A,V}$ can be written from (16) as:

$$I_{A,V} = J\left(\sqrt{\frac{\sigma^2 - \sigma_{ch}^2}{(d_v - 1)}}\right)$$

By replacing σ^2 with (9) we obtain:

$$I_{A,V} = J\left(\sqrt{\frac{[\sigma_{ch}^2 + (d_v - 1)\sigma_A^2] - \sigma_{ch}^2}{(d_v - 1)}}\right)$$
$$= J(\sigma_A) \tag{17}$$

Since $J(\sigma_A)$ is monotonic in σ_A, the inverse function $J^{-1}(\cdot)$ exists, and the approximation of the inverse function $J^{-1}(\cdot)$ is updated from (17) (see also [14]) and expressed by:

$$\sigma_A = J^{-1}(I_{A,V}) \tag{18}$$

By correspondence between (16) and (18):

$$\sigma_A^2 = [J^{-1}(I_{A,V})]^2$$
$$= \frac{\sigma^2 - \sigma_{ch}^2}{(d_v - 1)} \tag{19}$$

In (12) and (14) can also be written for LDPC and LDPC-CCs respectively as:

$$I_{E,V} = J(\sigma) = J(\sqrt{(d_v - 1)[J^{-1}(I_{A,V})]^2 + \sigma_{ch}^2}) \tag{20}$$

$$I_{E,V_T} = J(\sigma) = J\left(\sqrt{(d_v - 1)[J^{-1}(I_{A,V_T})]^2 + \sigma_{ch}^2}\right) \tag{21}$$

where I_{A,V_T} is the mutual information between the VND inputs (a priori) and the associate bit code is:

$$I_{A,V_T} = J(\sigma_A) \tag{22}$$

3.1.2. Check node's EXIT functions

The average extrinsic information at the check node decoder's output $I_{E,C}$ for LDPC codes can be computed starting with the symmetric-Gaussian assumption. However, the mean and variance of an CND output are not straightforward for determination [14], [22]. For convenience, and according to:

$$\begin{cases} \sigma_{ch} = 0 \\ d_v \leftarrow d_c \\ I_{A,V} \leftarrow 1 - I_{A,C} \end{cases}$$

we can write:

$$I_{E,C} = 1 - I_{E,V}\left(\sigma_{ch} = 0, d_v \leftarrow d_c, I_{A,V} = 1 - I_{A,C}\right) \tag{23}$$

Replacing the given conditions above in (20), $I_{E,C}$ for LDPC codes can be demonstrated as:

$$I_{E,C} = 1 - J\left(\sqrt{(d_c - 1)[J^{-1}(1 - I_{A,C})]^2}\right) \tag{24}$$

From (19) and following the conditions (25):

$$\sigma_A^2 = \frac{\sigma^2}{(d_c - 1)}(\sigma_{ch} = 0, d_v \leftarrow d_c) \tag{25}$$

In additional to:

$$I_{A,V} = 1 - I_{A,C} \tag{26}$$

Combining (18) with (26), we obtain:

$$[J^{-1}(1 - I_{A,C})]^2 = \frac{\sigma^2}{(d_c - 1)} \tag{27}$$

We replace then (27) in (24):

$$\begin{aligned} I_{E,C} &= 1 - J\left(\sqrt{(d_c - 1)\frac{\sigma^2}{(d_c - 1)}}\right) \\ &= 1 - J(\sigma)(\sigma_{ch} = 0, d_v \leftarrow d_c, I_{A,V} = 1 - I_{A,C}) \end{aligned}$$

We have obtained previously in (11) that $J(\sigma) = I_{E,V}$ which confirms the validity of (23).

For LDPC-CCs, CND degrees (d_c) differ in H_{conv} from those in H for LDPC codes with each period T as seen in Figure 3(a) in the general case and in Figure 3(b) as an example. Thus, a polynomial that defines the degree distributions for the check nodes must be generated as (28):

$$\rho(x) = \sum_{i=1}^{d_c} \rho_i x^{i-1} = \rho_1 + \rho_2 x + \rho_3 x^2 + \cdots + \rho_{d_c} x^{d_c - 1} \tag{28}$$

Then, according to (24), the average extrinsic information at the check node decoder's output I_{E,C_T} for LDPC-CCs can be expressed using the polynomial expressed by (28) as:

$$I_{E,C_T} = 1 - \sum_{i=1}^{d_c} \rho_i I_{E,V_T}\left(\sigma_{ch} = 0, d_v \leftarrow d_c, I_{A,V_T} = 1 - I_{A,C_T}\right) \tag{29}$$

Following now the same conditions from (25), the average a priori information $I_{A,C}$ for LDPC codes at the check node decoder's input is further usefully expressed from (26) as:

$$I_{A,C} = 1 - I_{A,V}(\sigma_{ch} = 0, d_v \leftarrow d_c) \tag{30}$$

Obtaining then from (27):

$$I_{A,C} = 1 - J(\frac{\sigma}{\sqrt{d_c-1}}) \qquad (31)$$

Where σ from (11) can be expressed in terms of the inverse function so that:

$$\sigma = J^{-1}(I_{E,V}) \qquad (32)$$

Thus:

$$I_{A,C} = 1 - J(\frac{J^{-1}(I_{E,V})}{\sqrt{d_c-1}}) \qquad (33)$$

To express the average a priori information in terms of the average extrinsic information at the check node decoder's, (23) can be written as:

$$I_{E,V} = 1 - I_{E,C} \qquad (34)$$

By substituting (34) in (33), we obtain:

$$I_{A,C} = 1 - J(\frac{J^{-1}(1-I_{E,C})}{\sqrt{d_c-1}}) \qquad (35)$$

The average a priori information I_{A,C_T} for LDPC-CCs can be obtained using the same approach as for LDPC codes, taking into account the polynomial of the CND degree distributions given by (28) as (36):

$$I_{A,C_T} = 1 - \sum_{i=1}^{d_c} \rho_i J(\frac{J^{-1}(1-I_{E,C_T})}{\sqrt{d_c-1}}) \qquad (36)$$

To summarize, the EXIT functions of regular LDPC and LDPC-CCs are illustrated in Table 1.

Table 1. EXIT functions of regular LDPC and LDPC-CCs

	LDPC	LDPC-CCs
$I_{E,V}, I_{E,V_T}$		$J(\sqrt{(d_v-1)\sigma_A^2 + \sigma_{ch}^2}$
$I_{E,C}, I_{E,C_T}$	$1 - I_{E,V}$	$1 - \sum_{i=1}^{d_c} \rho_i I_{E,V_T}$
$I_{A,V}, I_{A,V_T}$		$J\left(\sqrt{\frac{\sigma^2 - \sigma_{ch}^2}{(d_v-1)}}\right)$
$I_{A,C}, I_{A,C_T}$	$1 - J(\frac{J^{-1}(1-I_{E,C})}{\sqrt{d_c-1}})$	$1 - \sum_{i=1}^{d_c} \rho_i J(\frac{J^{-1}(1-I_{E,C_T})}{\sqrt{d_c-1}})$

3.2. EXIT function of irregular LDPCs and LDPC-CCs

An irregular LDPC code can be depicted as an irregular tanner graph with different degrees of VNDs and CNDs. This graph defines one period of a corresponding LDPC-CC. Variable and check node's EXIT functions for irregular LDPC codes are calculated as weighted averages using two polynomials $\lambda(x)$ and $\rho(x)$ that define the degree distributions of the variable and check nodes respectively [4] as (37) and (38):

$$\lambda(x) = \sum_{i=2}^{d_v} \lambda_i x^{i-1} = \lambda_2 x + \lambda_3 x^2 + \cdots + \lambda_{d_v} x^{d_v-1} \qquad (37)$$

$$\rho(x) = \sum_{i=2}^{d_c} \rho_i x^{i-1} = \rho_2 x + \rho_3 x^2 + \cdots + \rho_{d_c} x^{d_c-1} \qquad (38)$$

The polynomial $\lambda(x)$ is computed by adding up the fraction of edges λ_i that are connected to variable nodes of degree i, from degree 2 to d_v. Similarly, the polynomial $\rho(x)$ is calculated by adding up the fraction of edges ρ_i that are connected to check nodes of degree i, from degree 2 to d_c. The fraction of

edges λ_i and ρ_i are obtained by dividing the number of ones in each column and row, respectively, by the total number of ones (Q) in the parity-check matrix H for LDPC codes and in H_{conv} for LDPC-CCs.

In Figure 5, the parity-check matrix at a rate-1/2 for irregular LDPC (Figure 5(a)) and LDPC-CC (Figure 5(b)) with $T = 1$ is illustrated with the different fraction of edges λ_i and ρ_i.

$$
\begin{array}{c}
\begin{matrix} \lambda_3 & \lambda_2 & \lambda_3 & \lambda_3 & \lambda_2 & \lambda_3 & \lambda_2 & \lambda_3 & \lambda_3 & \lambda_2 \end{matrix} \\
\begin{matrix}
\rho_5 \\ \rho_6 \\ \rho_5 \\ \rho_5 \\ \rho_5
\end{matrix}
\begin{bmatrix}
1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\
0 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \\
1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 \\
0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \\
1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0
\end{bmatrix}
\end{array}
$$

(a)

$$
\begin{array}{c}
\begin{matrix} \lambda_3 & \lambda_2 & \lambda_3 & \lambda_3 & \lambda_2 & \lambda_3 & \lambda_2 & \lambda_3 & \lambda_3 & \lambda_2 \end{matrix} \\
\begin{matrix}
\rho_1 \\ \rho_3 \\ \rho_3 \\ \rho_3 \\ \rho_5 \\ \rho_4 \\ \rho_3 \\ \rho_2 \\ \rho_2
\end{matrix}
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\
1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\
0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1
\end{bmatrix}
\end{array}
$$

(b)

Figure 5. Irregular parity-check matrix at a rate-1/2 for; (a) LDPC code and (b) LDPC-CC with $T = 1$

For the irregular LDPC code, the fraction of edges λ_2 and λ_3 connected to variable nodes of degree 2 and 3 respectively are:

$$\lambda_2 = \frac{8}{26} = 0.3077 \text{ and } \lambda_3 = \frac{18}{26} = 0.6923$$

The fraction of edges ρ_5 and ρ_6 connected to check nodes of degree 5 and 6 respectively are:

$$\rho_5 = \frac{20}{26} = 0.7692 \text{ and } \rho_6 = \frac{6}{26} = 0.2308$$

The irregular LDPC code is then of degree distribution:

$$\lambda(x) = 0.3077x + 0.6923x^2$$

$$\rho(x) = 0.7692x^4 + 0.2308x^5$$

λ_i and ρ_i must respectively satisfy $\sum_{i=2}^{d_v} \lambda_i = 1$ and $\sum_{i=2}^{d_c} \rho_i = 1$. The designed rate of the code in this case is given by $R = 1 - \frac{\int_0^1 \rho(x)dx}{\int_0^1 \lambda(x)dx}$.

From Figure 5, one period $(T = 1)$ of an LDPC-CC is considered, and the total number of ones (Q) in the parity-check matrix H_{conv} is identical to that of the corresponding LDPC code's parity check matrix H as shown in Figure 5(b). Additionally, the fraction of edges λ_2 and λ_3 connected to variable nodes for LDPC-CC $(T = 1)$ is equivalent to that of the LDPC code:

$$\lambda_2 = \frac{8}{26} = 0.3077 \text{ and } \lambda_3 = \frac{18}{26} = 0.6923$$

Nevertheless, the ratio of connections to check nodes in LDPC-CC is distinct from that of LDPC code:

$$\rho_1 = \frac{1}{26} = 0.0385, \rho_2 = \frac{4}{26} = 0.1538, \rho_3 = \frac{12}{26} = 0.4615, \rho_4 = \frac{4}{26} = 0.1538, \rho_5 = \frac{5}{26} = 0.1923$$

The irregular LDPC-CC for $T = 1$ is then of degree distribution:

$$\lambda(x) = 0.3077x + 0.6923x^2$$

$$\rho(x) = 0.0385 + 0.1538x + 0.4615x^2 + 0.1538x^3 + 0.1923x^4$$

For $T = 3$, the degree distributions of the irregular H_{conv} for the same LDPC-CC are obtained as:

$$
\begin{array}{c}
\begin{smallmatrix} \lambda_1 & \lambda_2 & \lambda_3 & \lambda_3 & \lambda_3 & \lambda_2 & \lambda_3 & \lambda_3 & \lambda_3 & \lambda_2 & \lambda_3 & \lambda_3 & \lambda_3 & \lambda_2 & \lambda_3 & \lambda_3 & \lambda_3 & \lambda_2 & \lambda_3 & \lambda_3 & \lambda_3 & \lambda_2 & \lambda_3 & \lambda_3 & \lambda_3 & \lambda_2 \end{smallmatrix} \\
\begin{array}{c} \rho_1 \\ \rho_3 \\ \rho_3 \\ \rho_3 \\ \rho_5 \\ \rho_5 \\ \rho_6 \\ \rho_5 \\ \rho_5 \\ \rho_5 \\ \rho_5 \\ \rho_6 \\ \rho_5 \\ \rho_5 \\ \rho_5 \\ \rho_4 \\ \rho_3 \\ \rho_2 \\ \rho_2 \end{array}
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1
\end{bmatrix}
\end{array}
$$

Even though the total number of ones (Q) in H_{conv} for a period $T = 3$ is three times greater than Q in H_{conv} for $T = 1$ ($Q(T = 3) = 3 \times Q(T = 1)$), the fraction of edges connected to variable nodes for the LDPC-CC with $T = 3$ is the same as that for ($T = 1$):

$$\lambda_2 = \frac{24}{78} = 0.3077 \text{ and } \lambda_3 = \frac{54}{78} = 0.6923$$

However, CNDs degree distributions for LDPC-CC with $T = 3$ are not the same as those of LDPC-CC with $T = 1$:

$$\rho_1 = \frac{1}{78} = 0.0128, \rho_2 = \frac{4}{78} = 0.0513, \rho_3 = \frac{12}{78} = 0.1538, \rho_4 = \frac{4}{78} = 0.0513, \rho_5 = \frac{45}{78} = 0.5770,$$
$$\rho_6 = \frac{12}{78} = 0.1538$$

The degree distributions of the irregular LDPC-CC for $T = 3$ are then given as follows:

$$\lambda(x) = 0.3077x + 0.6923x^2$$

$$\rho(x) = 0.0128 + 0.0513x + 0.1538x^2 + 0.0513x^3 + 0.5770x^4 + 0.1538x^5$$

It can be noticed that for an LDPC-CC $\forall T$, the $CNDs$ degree distributions vary from $i = 1$ to $\max(d_c)$. Degree distributions for irregular LDPC-CCs can be summarized in Table 2.

Table 2. Degree distribution properties of irregular LDPC-CCs

Period (T)	Total number of ones (Q)	VNDs degree distributions (λ)	CNDs degree distributions (ρ)
1	Q	$[2, \max(d_v)]$	$[1, \max(d_c)]$
≥ 2	$T \times Q$		

Then, EXIT functions for irregular codes are also decomposed into variable and check nodes as follows.

3.2.1. Variable Node's EXIT Functions
For an irregular LDPC code's EXIT curve, the variable node function $I_{E,V}$ can be calculated using $J(\sigma)$ as (39):

$$I_{E,V} = \sum_{i=2}^{d_v} \lambda_i J\left(\sqrt{(i-1)\left[J^{-1}(I_{A,V})\right]^2 + \sigma_{ch}^2}\right) \tag{39}$$

Replacing (11) and (19) in (39) for ($d_v \leftarrow i$):

$$I_{E,V} = \sum_{i=2}^{d_v} \lambda_i J\left(\sqrt{(i-1)\left[\frac{\sigma^2 - \sigma_{ch}^2}{(i-1)}\right] + \sigma_{ch}^2}\right)$$
$$= \sum_{i=2}^{d_v} \lambda_i J(\sigma) \tag{40}$$

where $J(\sigma) = I_{E,V}(regular)$. Thus:

$$I_{E,V}(irregular) = \sum_{i=2}^{d_v} \lambda_i I_{E,V}(regular) \tag{41}$$

For irregular LDPC-CCs, since the $VNDs$ degree distributions are equal to those as for irregular LDPC codes, the irregular variable node function I_{E,V_T} can be calculated according to (14) and (37) for $(d_v \leftarrow i)$ as (42):

$$I_{E,V_T}(irregular) = \sum_{i=2}^{d_v} \lambda_i I_{E,V_T}(regular) \tag{42}$$

Under the consistent-Gaussian assumption, the mutual information $I_{A,V}$ between the VND inputs and the channel bits can be computed as [14]:

The following notation given by (43) is introduced for notational precision [23], and to emphasize $I_{A,V}$'s explicit dependence on σ_A:

$$J(\sigma) = I_{A,V}(\sigma_A = \sigma) \tag{43}$$

$$I_{A,V} = J(\sigma) \tag{44}$$
$$= \begin{cases} -\alpha\sigma^3 + \beta\sigma^2 - \theta\sigma & 0 \leq \sigma < 1.6363 \\ 1 - \exp(\alpha\sigma^3 - \beta\sigma^2 - \theta\sigma + \delta) & 1.6363 \leq \sigma < 10 \\ 1 & \sigma \geq 10 \end{cases}$$

3.2.2. Check node's EXIT functions

The EXIT function $I_{E,C}$ of an irregular LDPC code defining the check node function is computed by:

$$I_{E,C} = \sum_{i=2}^{d_c} \rho_i \left[1 - J\left(\sqrt{(i-1)\left[J^{-1}(I_{A,V})\right]^2} \right) \right] \tag{45}$$

By substituting the parameters of (23) in (45), we obtain:

$$\begin{aligned} I_{E,C} &= \sum_{i=2}^{d_c} \rho_i \left[1 - I_{E,V}\left(\sigma_{ch} = 0, d_c \leftarrow i, I_{A,V} = 1 - I_{A,C}\right) \right] \\ &= \sum_{i=2}^{d_c} \rho_i I_{E,C}(regular) \end{aligned} \tag{46}$$

Following the results shown in Table 2, for LDPC-CCs the CND function $\forall T$ is calculated as:

$$I_{E,C_T} = \sum_{i=1}^{d_c} \rho_i \left[1 - J\left(\sqrt{(i-1)\left[J^{-1}(I_{A,V_T})\right]^2} \right) \right] = \sum_{i=1}^{d_c} \rho_i I_{E,C_T}(regular) \tag{47}$$

In (32) can be used to calculate the mutual information $I_{A,C}$ for LDPC codes between the transmitted codeword and the message upon entering the check node. Replacing $I_{A,V}$ of (30) with $J(\sigma)$ from (44):

$$I_{A,C} = 1 - J(\sigma) \tag{48}$$
$$= \begin{cases} \alpha\sigma^3 + \beta\sigma^2 - \theta\sigma & 0 \leq \sigma < 1.3636 \\ 1 + \exp(\alpha\sigma^3 - \beta\sigma^2 - \theta\sigma + \delta) & 1.3636 \leq \sigma < 10 \\ 0 & \sigma \geq 10 \end{cases}$$

Table 3 summarizes the overall metrics of EXIT functions for irregular LDPCs and LDPC-CCs. Tables 4 and 5 define respectively the parameters of the function's approximations expressed by (44) and (48) with their conditions. Since the VNDs degree distributions in irregular LDPC-CCs are the same as those of LDPC codes, then the mutual information I_{A,V_T} can be expressed by (44), where the parameters α, β and θ, as well as their conditions take the same values as those in Table 4 for each period T. In addition to, these parameters and conditions in the case CNDs degree distributions of irregular LDPC-CCs can be expressed by (48) with the same values as those in Table 5 for each period T.

Table 3. EXIT function of irregular LDPCs and LDPC-CCs

Code	$I_{E,C}, I_{E,C_T}$	$I_{A,C}, I_{A,C_T}$	$I_{E,V}, I_{E,V_T}$	$I_{A,V}, I_{A,V_T}$
LDPC	$\sum_{i=2}^{d_c} \rho_i I_{E,C}(regular)$	$\sum_{i=2}^{d_c} \rho_i I_{A,C}(regular)$	$\sum_{i=2}^{d_v} \lambda_i I_{E,V}(regular)$	$\sum_{i=2}^{d_v} \lambda_i I_{A,V}(regular)$
LDPC-CC	$\sum_{i=1}^{d_c} \rho_i I_{E,C_T}(regular)$	$\sum_{i=1}^{d_c} \rho_i I_{A,C_T}(regular)$	$\sum_{i=2}^{d_v} \lambda_i I_{E,V_T}(regular)$	$\sum_{i=2}^{d_v} \lambda_i I_{A,V_T}(regular)$

Table 4. Function approximation $I_{A,V}$, I_{A,V_T} parameters and its conditions

Condition	Parameters			
	α	β	θ	δ
$0 \leq \sigma < 1.6363$	0.0421061	0.209252	0.00640081	0
$1.6363 \leq \sigma < 10$	0.00181491	0.142675	0.0822054	0.0549608
$\sigma \geq 10$	1			

Table 5. Function approximation $I_{A,C}$, I_{A,C_T} parameters and its conditions

Condition	Parameters			
	α	β	θ	δ
$0 \leq \sigma < 1.3636$	1.0421061	0.790748	0.99359919	0
$1.3636 \leq \sigma < 10$	0.00181491	0.142675	0.0822054	0.0822054
$\sigma \geq 10$	0			

3.3. EXIT regions

Once the EXIT chart of a code is obtained, it is important to analyze the EXIT curves and the simulated bit error rate (BER) [24] using various regions to assess the convergence of the iterative decoding process. To achieve the convergence in the iterative decoding process, it is necessary for the transfer characteristics I_{E,C_T}, I_{A,V_T} for LDPC-CCs to be positioned below the combined transfer characteristics I_{E,V_T}, I_{A,C_T}. The ideal case of convergence in the EXIT chart occurs at the point (1.0;1.0). I_{E,C_T} and I_{A,V_T} are equal to 0 at the beginning of the iterative decoding process as shown in Figure 6. Then the decoding trajectory begins at a point $(0; I_{E,V_T}, I_{A,C_T}(0))$ called the start point. The interpretation regions of the EXIT chart are classified into.
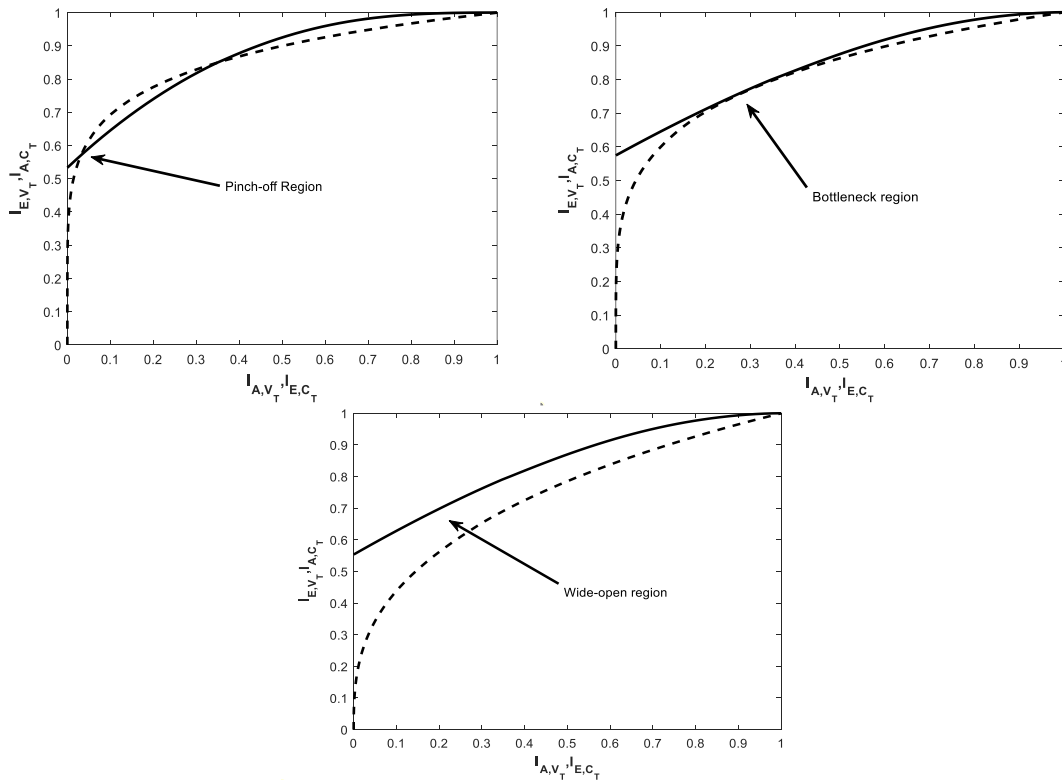

Figure 6. EXIT regions

3.3.1. Pinch-off region

It is the undesired region in the EXIT chart, so the Pinch-off Point occurs at a low signal-to-noise ratio (SNR), where the EXIT curves for the variable and check nodes touch or become extremely close to

each other. Once the curves touch, the decoder gets "stuck" because the exchange of information between variable and check nodes becomes ineffective, and the decoding process can no longer progress toward a successful decoding. It is also called as "cliff region" that means low E_b/N_0 region which results in bad performance and more iterations in this case have no effect. The two decoders' transfer curves cross in this region with low mutual information values, which leads to the absence of convergence.

3.3.2. Bottleneck region

As the iterative decoding progresses, the mutual information increases, and the decoder's state follows a trajectory on the EXIT chart. This trajectory starts near the origin and ideally moves toward the top right corner, where mutual information approaches '1' (indicating successful decoding). The bottleneck region is identified as a narrow gap between the EXIT curves of the variable and check nodes. It occurs when the trajectory of the decoding process passes through an area where the gap between the two curves is small. In this region, the mutual information increases very slowly across iterations, meaning that the decoder struggles to make significant progress in improving the estimate of the transmitted codeword.

3.3.3. Wide-open region

Described also as the waterfall region, which is a region where the iterative decoding process can progress smoothly and efficiently, with a large gap between the EXIT curves of the variable and check nodes. This region is the opposite of the bottleneck or pinch-off regions, as it allows for rapid mutual information exchange and fast convergence during decoding. Due to a high SNR, a wide-open tunnel between the two curves of the transfer characteristics is resulted. This gives a rapid reach to the point (1.0;1.0) in EXIT chart.

The larger the gap, the easier it is for the decoding process to converge, as there is plenty of space for the mutual information to increase from one iteration to the next. This region is critical for ensuring that the decoder can successfully converge on the correct codeword, especially in the early stages of decoding. Maximizing this region through careful code design can lead to better overall performance and error correction capabilities.

4.   SIMULATION RESULTS

This section provides simulation results for LDPC and LDPC-CCs with EXIT charts in order to assess and compare their performance. Firstly, the various (d_v, d_c) values for the different matrices to be used in simulations for a rate-1/2 are defined. Next, a comparison of I_{E,V_T} versus I_{A,V_T} and I_{E,C_T} versus I_{A,C_T} is presented for LDPC-CCs. This subsection will address the impact of the (d_v, d_c) degree distributions as well as the period T on the EXIT curves of LDPC-CCs, as the code's characteristics are affected by this parameter. The second subsection will display a comparison of the performance based on the EXIT charts of regular and irregular LDPCs and LDPC-CCs. Lastly, the third subsection will showcase a comparison of BER with varying iterations and periods for regular and irregular LDPC-CCs.

4.1. Choice of initial parameters by VND/CND EXIT curves

Prior to presenting any outcomes, it is important to establish the chosen degrees, d_v and d_c. In the following simulations, we chose $E_b/N_0 = 1.1\ dB$ for comparison and convenience purpose with the results given in [22], [23]. This choice is motivated such that for lower values we don't reach a convergence of the EXIT chart, and for higher values the convergence is rapidly attained. Figure 7 presents various node degrees at a rate-1/2 for the given number of nodes, which are used in the next simulation analyses.
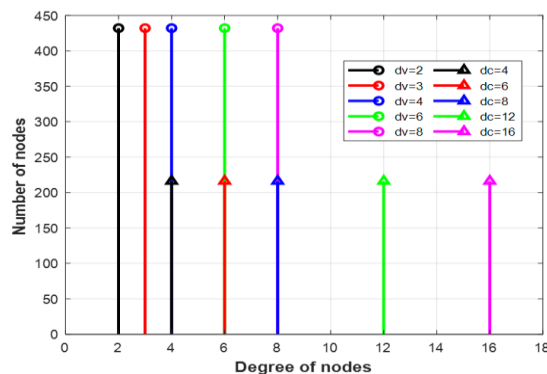

Figure 7. Different (d_v, d_c) degrees for several regular parity-check matrices used in simulations

The EXIT chart of the regular LDPC-CC with different variable/check nodes degrees and periods is depicted in Figure 8. According to Figure 8(a) which represents the EXIT curves for regular LDPC-CCs with $T = 3$, the degrees of VND are varied from 2 to 8 and CND from 4 to 16. These curves are generated using (14), (22) for VNDs and (29), (36) for CNDs. We can clearly observe a decrease in the gap between the VND and CND curves as the values of (d_v, d_c) increase. In Figure 8(b), we study the evolution of EXIT curves as a function of the parameter T with a fixed VND and CND degrees.
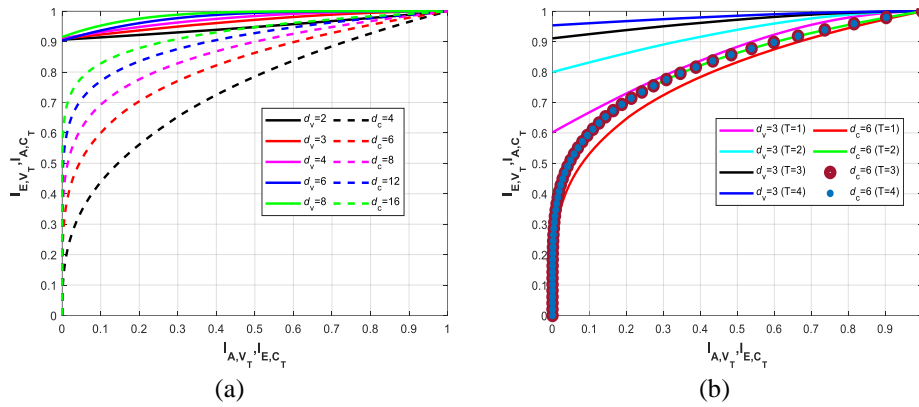
Figure 8. EXIT curves of LDPC-CC with $E_b/N_0 = 1.1\ dB$ for; (a) different (d_v, d_c) degrees with $T = 3$ and (b) different periods with $(d_v, d_c) = (3,6)$

Despite of the growing gap size of EXIT curves with $(d_v, d_c) = (2,4)$ in comparison with $(d_v, d_c) = (3,6)$, we chose (3,6) in our simulations. This choice is motivated by the lower probability of girth occurrences than the choice of $(d_v, d_c) = (2,4)$, such that girth free parity-check matrices exhibit better performances. It is evident that as the number of periods increase, the size of the gap between VND and CND EXIT curves increases, which gives better convergence rate. Also, the VND curves contribute in the enhancement of the gap better than the CND curves, where for $T \geq 2$ these later curves don't exhibit any enhancement and stay stable.

4.2. EXIT Chart analysis of regular and irregular LDPC codes

In this subsection, the focus is on the EXIT chart performance analysis of regular and irregular LDPC codes using the parity-check matrix H. The simulations presented in Figure 9 illustrate the EXIT chart for the regular (432,3,6) and its corresponding irregular LDPC code, with a decoding trajectory shown on the EXIT charts. The convergence of the EXIT chart at the point (1.0;1.0) is observed in Figures 9(a) and (b), indicating the formation of a narrow tunnel for both regular and irregular LDPC codes. The bottleneck region is characterized by a lower BER that may be attainable. However, the region's improvement is more pronounced in the case of irregular LDPC codes as depicted in Figure 9(b).
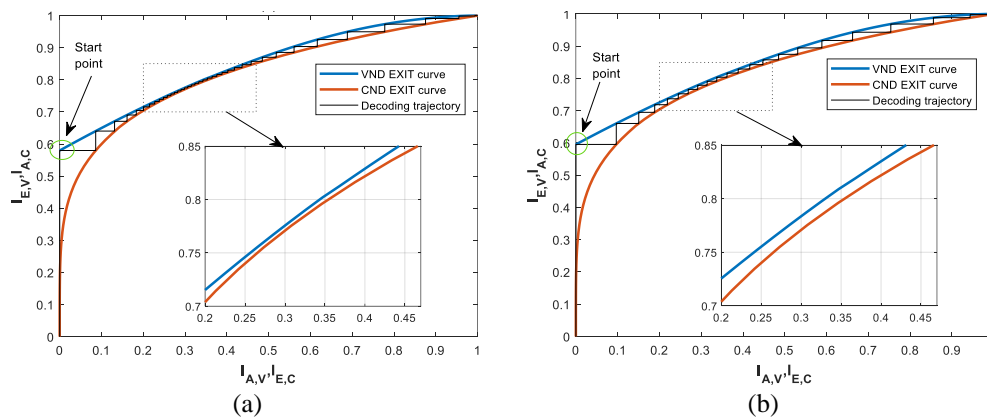

Figure 9. EXIT chart with $E_b/N_0 = 1.1 dB$ for; (a) regular (3,6) LDPC code and (b) irregular LDPC code

4.3. EXIT chart analysis of regular and irregular LDPC-CCs

In this subsection, we present the EXIT chart performance analysis of regular and irregular LDPC-CCs with different periods. Table 6 presents the parameters of LDPC-CC used in IEEE 1901 [25] that we used in our simulations.

Table 6. Main parameters of a rate-1/2 LDPC-CC in IEEE 1901

R	d_v	d_c	m_s	T	v_s
1/2	3	6	215	3	432

The EXIT charts of the regular LDPC-CC (215,3,6) with $T = 1$ and $T = 3$ are shown in Figure 10, using the same initial parameters given in Table 6. We can see that the convergence to the point (1.0;1.0) is reached in for both $T = 1$ and $T = 3$ in Figures 10(a) and (b), respectively. In addition, the convergence speed for $T = 3$ is faster than for $T = 1$. More precisely, the EXIT chart with $T = 1$ spend 11 iterations to intersect at the point (1.0;1.0). However, the EXIT chart with $T = 3$ spend only 3 iterations to intersect at the point (1.0;1.0) explained by a wider gap than the one for $T = 1$.

For irregular LDPC-CCs, the EXIT chart performance analysis with periods $T = 1$ and $T = 3$ is presented in Figures 10(c) and (d), respectively. We notice the same observations as in the case of regular LDPC-CCs with a faster convergence speed. So, the irregular LDPC-CC with $T = 1$ and $T = 3$ needs only 8 and 2.5 iterations rather than 11 and 3 iterations needed in the regular LDPC-CC. Table 7 provides a summary of the results obtained from the EXIT charts for regular and irregular LDPCs and LDPC-CCs.
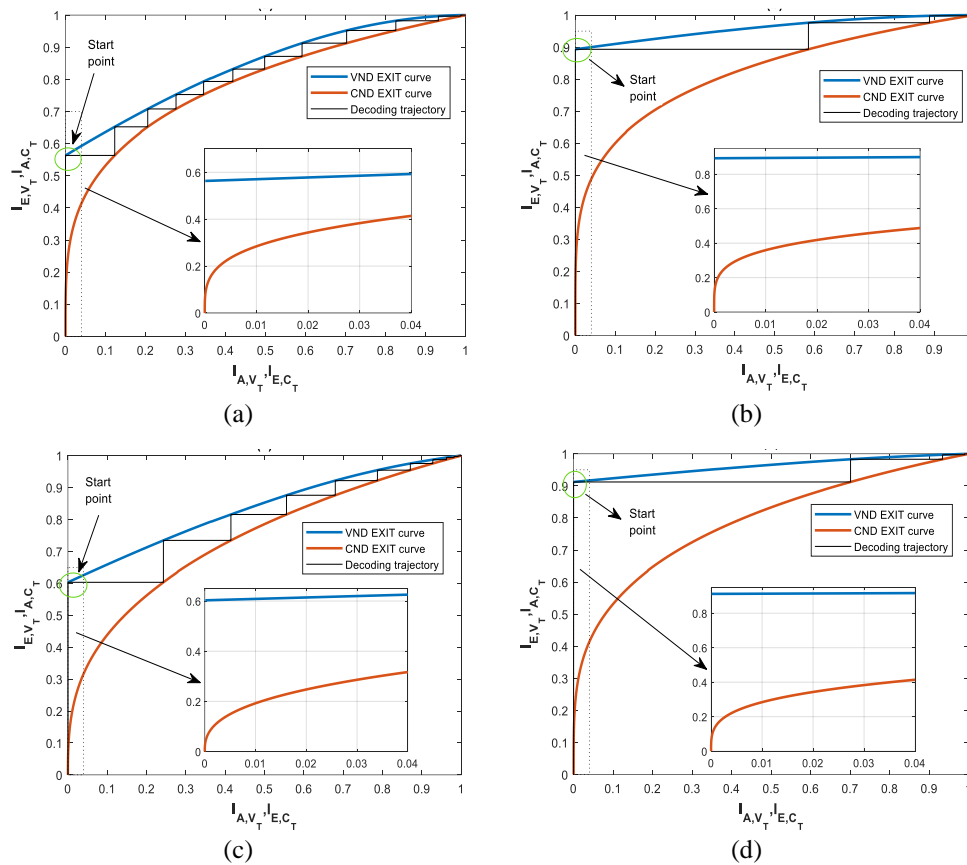

Figure 10. EXIT chart with $E_b/N_0 = 1.1\ dB$ for; (a) regular (3,6) LDPC-CC with $T = 1$, (b) regular (3,6) LDPC-CC with $T = 3$, (c) irregular LDPC-CC with $T = 1$, and (d) irregular LDPC-CC with $T = 3$

4.4. BER performance evaluation with the number of iterations and periods

To confirm the properties obtained with the EXIT chart analysis in the previous subsection, we simulated the performance of regular and irregular LDPC-CCs in Monte-Carlo simulations in terms of the BER, and the obtained results are illustrated in Figure 11.

Table 7. EXIT chart characteristics of regular and irregular LDPCs and LDPC-CCs

R	n	d_v	d_c	E_b/N_0 (dB)	Code	Matrix		Start point	Region	Converging point
1/2	432	3	6	1.1	LDPC	Regular		(0;0.5765)	Bottleneck	(1.0;1.0)
						Irregular		(0;0.6032)		
					LDPC-CC	Regular	$T = 1$	(0;0.5712)	Open	
							$T = 3$	(0;0.8917)	Wide-open	
						Irregular	$T = 1$	(0;0.6030)	Open	
							$T = 3$	(0;0.9077)	Wide-open	

In terms of the number of iterations as shown in Figure 11(a), it can be observed that with an increased number of iterations from 20 to 100, the BER performance improves for both regular and irregular LDPC-CC. particularly, the irregular LDPC-CC achieves a BER of 8×10^{-8} at 2.5 E_b/N_0 after 100 iterations, which is better than the BER of 6×10^{-7} achieved by the regular LDPC-CC at the same E_b/N_0 and with the same number of iterations. This indicates that the irregular LDPC-CCs performs better than regular codes under the same conditions. Moreover, increasing the number of iterations leads to more error-correction performance for both regular and irregular LDPC-CCs.

Since irregular LDPC-CCs have shown better performance in terms of iterations, the impact of increasing the number of periods from $T = 1$ to $T = 10$ is analyzed for irregular codes in Figure 11(b). When comparing the BER for a target value of 10^{-4}, for $T = 3$ we obtain a positive gain of $+1.04$ dB compared to $T = 1$. However, when the target BER is set to 10^{-6}, although when increasing the period from $T = 3$ to $T = 10$, the achieved positive gain exhibit a small improvement ($+0.15$ dB). This suggests that as the number of periods increases, the improvement in BER performance decreases.

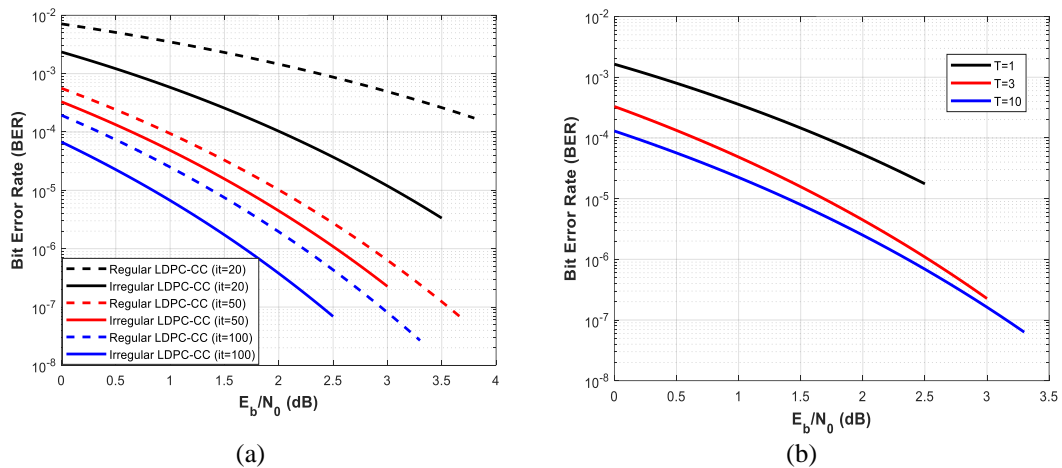(a)   (b)

Figure 11. Performance of LDPC convolutional's BER codes with; (a) different iterations for $T = 3$, and (b) different periods for iterations $= 50$

5.   CONCLUSION

This paper focuses on the performance of LDPC-CCs using the EXIT chart, where we studied the regular and irregular conception and compared the results. We started our study from the case of LDPC codes EXIT chart, then we derived the mutual information functions between variable nodes and check nodes of the LDPC-CCs Tanner graph. This involves applying the EXIT functions used for LDPC codes to the parity-check matrix of LDPC-CCs with the desired number of periods. After that, we simulated the convergence behavior using the obtained EXIT chart functions of the regular and irregular LDPC-CCs.

Firstly, the influence of the parity-check matrix degree distributions on the convergence behavior of the LDPC-CCs was simulated, where the results showed an inverse relation between the gap width of the EXIT chart and the degree distribution parameters. It is obvious that a wider EXIT chart gap means a faster convergence of the iterative decoding, but a lower degree distribution parameters induce highly likely occurrence of short cycles (girth) in the parity-check matrix. Secondly, the influence of the number of periods of the LDPC-CC parity-check matrix on the convergence behavior of the EXIT chart was also simulated. The results showed a faster iterative decoding convergence for a higher number of periods. Next, we compared

the iterative decoding convergence of the regular and irregular cases in LDPC and LDPC-CC codes. As a general results in this simulation, the irregular parity-check matrix gives a better performance than the regular one despite it was an LDPC or LDPC-CC code due to their random nature. In addition to, under the same conditions, LDPC-CCs show a faster iterative decoding convergence than the LDPC codes, as it was proved with other methods for iterative decoding analysis in the literature. Finally, the iterative decoding process evaluation for LDPC-CCs was analyzed using the EXIT chart and the BER curves for comparison purposes. We found that these two charts reflect very closely the iterative decoding process of LDPC-CCs. Mutual information between variable and check nodes for LDPC-CCs, namely the EXIT chart, was found to be a useful tool for studying the convergence behaviour of iterative decoding with low computational complexity and time efficient simulation.

REFERENCES

[1] K. Arora, J. Singh, and Y. S. Randhawa, "A survey on channel coding techniques for 5G wireless networks," *Telecommunication Systems*, vol. 73, no. 4, pp. 637–663, Apr. 2020, doi: 10.1007/s11235-019-00630-3.

[2] H. Zhong and T. Zhang, "Block-LDPC: a practical LDPC coding system design approach," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 52, no. 4, pp. 766–775, Apr. 2005, doi: 10.1109/TCSI.2005.844113.

[3] S. Usman and M. M. Mansour, "Fast column message-passing decoding of low-density parity-check codes," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 68, no. 7, pp. 2389–2393, Jul. 2021, doi: 10.1109/TCSII.2021.3049733.

[4] F. Vatta, A. Soranzo, M. Comisso, G. Buttazzoni, and F. Babich, "Performance study of a class of irregular LDPC codes through low complexity bounds on their belief-propagation decoding thresholds," in *2019 AEIT International Annual Conference (AEIT)*, IEEE, Sep. 2019, pp. 1–6, doi: 10.23919/AEIT.2019.8893306.

[5] R. Tanner, "A recursive approach to low complexity codes," *IEEE Transactions on Information Theory*, vol. 27, no. 5, pp. 533–547, Sep. 1981, doi: 10.1109/TIT.1981.1056404.

[6] 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*, vol. 57, no. 2, pp. 835–857, Feb. 2011, doi: 10.1109/TIT.2010.2095211.

[7] A. Jimenez Felstrom and K. S. Zigangirov, "Time-varying periodic convolutional codes with low-density parity-check matrix," *IEEE Transactions on Information Theory*, vol. 45, no. 6, pp. 2181–2191, 1999, doi: 10.1109/18.782171.

[8] S. Khittiwitchayakul, W. Phakphisut, and P. Supnithi, "Associated sectors of magnetic recording systems using spatially coupled LDPC codes," *ECTI Transactions on Electrical Engineering, Electronics, and Communications*, vol. 20, no. 1, pp. 10–21, Feb. 2022, doi: 10.37936/ecti-eec.2022201.246094.

[9] L. Schmalen, D. Suikat, D. Rosener, and A. Leven, "Evaluation of left-terminated spatially coupled LDPC codes for optical communications," in *2014 The European Conference on Optical Communication (ECOC)*, IEEE, Sep. 2014, pp. 1–3, doi: 10.1109/ECOC.2014.6964006.

[10] K. Zhu and Z. Wu, "Comprehensive study on CC-LDPC, BC-LDPC and polar code," in *2020 IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*, IEEE, Apr. 2020, pp. 1–6, doi: 10.1109/WCNCW48565.2020.9124897.

[11] M. Lentmaier, A. Sridharan, D. J. Costello, and K. S. Zigangirov, "Iterative decoding threshold analysis for LDPC Convolutional Codes," *IEEE Transactions on Information Theory*, vol. 56, no. 10, pp. 5274–5289, Oct. 2010, doi: 10.1109/TIT.2010.2059490.

[12] D. G. M. Mitchell, M. Lentmaier, and D. J. Costello, "Spatially coupled LDPC codes constructed from protographs," *IEEE Transactions on Information Theory*, vol. 61, no. 9, pp. 4866–4889, Sep. 2015, doi: 10.1109/TIT.2015.2453267.

[13] S. ten Brink, "Convergence of iterative decoding," *Electronics Letters*, vol. 35, no. 10, p. 806, 1999, doi: 10.1049/el:19990555.

[14] S. ten Brink, G. Kramer, and A. Ashikhmin, "Design of low-density parity-check codes for modulation and detection," *IEEE Transactions on Communications*, vol. 52, no. 4, pp. 670–678, Apr. 2004, doi: 10.1109/TCOMM.2004.826370.

[15] A. Ashikhmin, G. Kramer, and S. ten Brink, "Extrinsic information transfer functions: model and erasure channel properties," *IEEE Transactions on Information Theory*, vol. 50, no. 11, pp. 2657–2673, Nov. 2004, doi: 10.1109/TIT.2004.836693.

[16] S. Kudekar, T. J. Richardson, and R. L. Urbanke, "Threshold saturation via spatial coupling: why convolutional LDPC ensembles perform so well over the BEC," *IEEE Transactions on Information Theory*, vol. 57, no. 2, pp. 803–834, Feb. 2011, doi: 10.1109/TIT.2010.2095072.

[17] A. Sridharan, M. Lentmaier, and D. J. Costello Jr., "Convergence analysis for a class of LDPC convolutional codes on the erasure channel," in *Annual Allerton Conference on Communications, Control and Computing (Allerton)*, 2004, pp. 953–962.

[18] G. Liva and M. Chiani, "Protograph LDPC codes design based on EXIT analysis," in *IEEE GLOBECOM 2007-2007 IEEE Global Telecommunications Conference*, IEEE, Nov. 2007, pp. 3250–3254, doi: 10.1109/GLOCOM.2007.616.

[19] A. R. Iyengar, M. Papaleo, P. H. Siegel, J. K. Wolf, A. Vanelli-Coralli, and G. E. Corazza, "Windowed decoding of protograph-based LDPC convolutional codes over erasure channels," *IEEE Transactions on Information Theory*, vol. 58, no. 4, pp. 2303–2320, Apr. 2012, doi: 10.1109/TIT.2011.2177439.

[20] H.-Y. Kwak, J.-S. No, and H. Park, "Design of irregular SC-LDPC codes with non-uniform degree distributions by linear programming," *IEEE Transactions on Communications*, vol. 67, no. 4, pp. 2632–2646, Apr. 2019, doi: 10.1109/TCOMM.2018.2889850.

[21] M. Battaglioni, F. Chiaraluce, M. Baldi, and M. Lentmaier, "Girth analysis and design of periodically time-varying SC-LDPC Codes," *IEEE Transactions on Information Theory*, vol. 67, no. 4, pp. 2217–2235, Apr. 2021, doi: 10.1109/TIT.2021.3059414.

[22] G. Liva, S. Song, L. Lan, Y. Zhang, S. Lin, and W. E. Ryan, "Design of LDPC codes: a survey and new results," *Journal of Communications Software and Systems*, vol. 2, no. 3, p. 191, Apr. 2017, doi: 10.24138/jcomss.v2i3.283.

[23] E. Sharon, A. Ashikhmin, and S. Litsyn, "Analysis of low-density parity-check codes based on EXIT functions," *IEEE Transactions on Communications*, vol. 54, no. 8, pp. 1407–1414, Aug. 2006, doi: 10.1109/TCOMM.2006.878828.

[24] M. Ebada, A. Elkelesh, S. Cammerer, and S. ten Brink, "Scattered EXIT charts for finite length LDPC code design," in *2018 IEEE International Conference on Communications (ICC)*, IEEE, May 2018, pp. 1–7, doi: 10.1109/ICC.2018.8422961.

[25] Y. Chen, C. Zhou, Y. Huang, and X. Zeng, "An efficient multi-rate LDPC-CC decoder with layered decoding algorithm," in *2013 IEEE International Conference on Communications (ICC)*, IEEE, Jun. 2013, pp. 5548–5552, doi: 10.1109/ICC.2013.6655475.

BIOGRAPHIES OF AUTHORS

Oulfa Laouar is a Ph.D. student in Electronics of Telecommunications at University of 20th August 1955 of Skikda, Algeria. She received her Master's degree in Telecommunication Engineering from the University of Abbes Laghrour of Khenchela, Algeria in 2019. Her research interests are in channel coding and information theory. She can be contacted at email: o.laouar@univ-skikda.dz.

Imed Amamra is an Associate professor in telecommunications at university of 20th August 1955 of Skikda, Algeria. He received his Master of Science degree (M.Sc.) in Communication engineering from the University of Technology, Iraq in 2002. He received also his Ph.D. in Electronic from the University of 20th August 1955 of Skikda, Algeria in 2018. His research interests are digital signal processing, information and communication technology and coding theory. He can be contacted at email: i.amamra@univ-skikda.dz.

Nadir Derouiche is a Professor in electronic and a director of the electronic research laboratory (LRES) at university of 20th August 1955 of Skikda, Algeria from 2013 to 2020. He received his Ph.D. in Electronic and Systems from the University of Blaise Pascal of Clermont-Ferrand, France in 1990. His research interests are based essentially on channel coding and efficient techniques of iterative decoding, also on cryptography and information security. He can be contacted at email: n.derouiche@univ-skikda.dz.