Performance Enhancement of LDPC Codes Based on Protograph Construction in 5G-NR Standard

Aymen M. Al-Kadhimi a*, A. E. Abdelkareem a, Charalampos C. Tsimenidis b

a College of Information Engineering/ Al-Nahrain University/ Baghdad/ Iraq.
b Department of Engineering/ Nottingham Trent University/ Nottingham/ UK.

Abstract: To meet the high throughput demands, the 3rd Generation Partnership Project has specified the low-density parity check (LDPC) codes in the fifth generation-new radio 5G-NR standard with rate and length compatibility and scalability. This paper presents an extensive performance evaluation and enhancement of LPDC using the protograph-based construction defined in the 5G-NR standard. Firstly, the protograph-LDPC with layered offset min-sum (OMS) decoding, polar with successive cancellation list (SCL), and block turbo code are implemented and compared. Puncturing and shortening are applied to maintain block length at 1024 and code rate at 1/2 for all codes for comparison fairness. The results showed that P-LDPC outperforms its counterparts in terms of bit/frame error rate (BER/FER) behavior for given signal-to-noise ratios. Then, different P-LDPC settings were realized to study the effects of base graph selection (Graph1 or Graph2), code rate change (1/3 - 2/3), and block lengths increase (260 – 4160 bits). The simulation outcomes proved that BER performed better for lower coding rates or higher block lengths. Furthermore, P-LDPC behavior was examined over a Rayleigh flat-fading channel to achieve a 12.5 dB coding gain at 0.001 BER compared with uncoded transmission.

Keywords:
Base Graph, Check/ Variable Nodes, Code Rate, LLR.

ARTICLE INFO

Article history:
Received 01 Dec 2022
Received in revised form 25 June 2023
Accepted 03 July 2023
Final Proofreading 12 July 2023
Available online 01 Nov. 2023

© THIS IS AN OPEN ACCESS ARTICLE UNDER THE CC BY LICENSE
http://creativecommons.org/licenses/by/4.0/


*Corresponding author:
Aymen M. Al-Kadhimi
College of Information Engineering/ Al-Nahrain University/ Baghdad/ Iraq.
1. INTRODUCTION

One of the ultimate objectives of communication is to have a reliable connection. Forward error correction (FEC) techniques are implemented to improve data transmission reliability, which is achieved by adding extra redundant bits to the data (information) bits before being sent via a channel. Then, the coded bits can be uncoded on the receiver side to retrieve the original source bits. Consequently, the channel impairments are mitigated or even canceled due to applying well-designed codes [1]. According to Shannon’s theory in 1948 [2], non-erroneous data transmission via noisy channels is achievable by appropriately designing FEC codes with a rate not exceeding the channel’s capacity. Afterward, many FEC codes were investigated and developed, such as hamming code [3], LDPC codes [4], turbo codes [5], and polar code [6].

Particularly, the LDPC code has attracted researchers to investigate and develop them due to its capacity-approaching performance. However, one of the practical challenges of LDPC code is the high encoding complexity. To tackle this weakness, a structured LDPC class, called multi-edge type (MET) codes, has been presented [7]. P-LDPC codes have been proposed as a sub-class of MET codes offering low coding complexity and high error-resilience performance [8-12].

A protograph is a relatively minimized-number-of-nodes Tanner graph. Copying and permutation are the two main operations that derive larger sizes of a protograph construction. Whenever a protograph is copied M times, each edge of the protograph grows into a package of M edges that connect M check nodes (CNs) to M variable nodes (VNs). These replicas of the protograph are connected by performing permutation of the CN-to-VN pairs within each package. Consequently, the resultant code graph is M times larger than the code corresponding to the protograph with a similar rate and CN and VN distribution. Furthermore, by ensuring the permuting operation is circularly applied to every protograph edge, the derived code is adjustable to high decoding speeds. For example, Fig. 1 depicts a protograph with 3 VNs, 2 CNs, and five edges with the copy-and-permute operation.

Fig. 1: Example of a Protograph Copying and Permutation.

The P-LDPC codes have attracted many research societies to adopt and develop these codes. The 3rd Generation Partnership Project (3GPP) is one of those communities that has deployed P-LDPC in practice as one of the main channel coding schemes in the 5G-NR standard [13]. This paper compares the performance of P-LDPC codes with other coding schemes and extensively evaluates different base graphs, code lengths, and code rates employed in the 5G-NR. This study considers the scenario of transmitting binary phase shift-keying (BPSK) symbols over additive white Gaussian noise (AWGN) and Rayleigh flat-fading channels.

In the past three decades, LDPC codes have been developed in literature by designers and researchers [14-16]. The channel coding has been the main scope of deployment of those LDPC codes for transmission over an AWGN channel [10, 17], a fading channel [18], and a relay-aided cooperative channel [19]. Hadamard-LDPC with a protograph structure...
can be designed using Hadamard codes instead of single parity check (SPC) codes [20]. This designed code showed a capacity approaching 0.16 dB apart from the Shannon limit. On the other hand, LDPC and P-LDPC codes can be deployed as source coding schemes as in [21, 22], respectively. Nevertheless, LDPC can be concatenated with Luby-transform codes [23] to produce Raptor codes. These latter codes are applicable in transmitting video via wireless communications [24, 25], i.e., distributed source codes [26]. Moreover, LDPC and protograph-based LDPC codes can be implemented for both source and channel coding for multimedia applications. In addition, LDPC with non-binary codes has been well analyzed and designed in [30, 31]. These codes are seamlessly creating a combination with higher-order modulation schemes. Similarly, the non-binary codes were extended to the protograph structure design [32]. Other variants of protograph structure were developed for convolutional-based LDPC codes [33] and raptor-based codes [34]. In addition, further minimizes the realization complexity, quasi-cyclic P-LDPC codes were proposed by circular-permute regulations [35]. The P-LDPC codes have attracted many research societies to adopt and develop these codes. The 3rd Generation Partnership Project (3GPP) is one of those communities that has deployed P-LDPC in practice as one of the main channel coding schemes in the 5G-NR standard [13]. This paper compares the performance of P-LDPC codes with other coding schemes and extensively evaluates different base graphs, code lengths, and code rates employed in the 5G-NR. This study considers the scenario of transmitting binary phase shift-keying (BPSK) symbols over additive white Gaussian noise (AWGN) and Rayleigh flat-fading channels.

3.PROTOGRAPH CODE CONSTRUCTION

The LDPC code construction is mostly implemented in two stages: by optimizing a protograph that performs well in decoding; and by applying a progressive edge growth (PEG) algorithm [36]. This algorithm expands the chosen protograph for constructing the parity check matrix (H) with a minimized number of short cycles [37]. Otherwise, the resultant Tanner graph is unsatisfied with the protograph base constraints, and the high number of short cycles may cause performance degradation in the decoding phase [38].

In the 5G standard, one of two base graphs (BG) matrices (B₁ and B₂) is selected for constructing the LDPC H matrix. The matrices B₁ and B₂ are of sizes 46 \times 68 and 42 \times 52, respectively. The matrix H is obtained by replacing each element of B with a matrix with size Z_c \times Z_c, where Z_c is an expansion (or lifting) factor specified in the standard according to a set index (i_{S}), as shown in Table 1. The B’s elements can be valued as \(-1, 0, 1, \ldots, Z_c - 1\), where each value is expanded as follows:

- Every entry of value \(-1\) in B is replaced by an all-zero matrix of size Z_c \times Z_c;
- Each entry of value \(i = 0, 1, \ldots, Z_c - 1\) in B is replaced by identity matrix \(I\) of size Z_c \times Z_c circularly shifted right \(i\) times.

For demonstration purposes, a mock example shows the paragraph construction for an LDPC code. Suppose a 3 \times 6 base graph matrix B is defined in Eq. (1), and an expansion factor \(Z_c = 5\) is selected. Each entry of B may have any integer value of -1 up to \(Z_c - 1\). Each value of B is expanded into a 5 \times 5 matrix, as shown in Eq. (2). For example, the value -1 is expanded into a 5 \times 5 all-zero matrix, whereas 1 is expanded into a 5 \times 5 identity matrix \(I\) circularly shifted right 1 time.

\[
B = \begin{bmatrix}
1 & -1 & 3 & 1 & 0 & -1 \\
2 & 0 & -1 & 0 & 0 & 0 \\
-1 & 4 & 2 & 1 & -1 & 0
\end{bmatrix}
\]  

(1)

\[
H = \begin{bmatrix}
\end{bmatrix}
\]  

(2)

For LDPC in the 5G-NR standard, the base matrix B is constructed first in a protograph. Then, the matrix B is expanded to build the parity-check matrix H. The matrix B is structured in blocks defined below:

\[
B = \begin{bmatrix}
A & E & O \\
B & C & I
\end{bmatrix}
\]  

(3)

where the blocks’ sizes are given below to build 

- \(B_1\rightarrow A: 4 \times 22, E: 4 \times 4, O: 4 \times 42\) all-zero;
- \(B: 42 \times 22, C: 42 \times 4, I: 42 \times 42\) identity
For $B_2 \rightarrow A: 4 \times 10, E: 4 \times 4, O: 4 \times 38$ all-zero; $B: 38 \times 10, C: 38 \times 4, I: 38 \times 38$ identity

To further comprehend the construction process, an example of $B_2$ is shown in Fig. 2. According to Table 1, the selected parameters are $I_{LS} = 0$ and expansion factor $Z_c = 128$. The dots shown in the figure are drawn only for none -1 elements. In other words, those elements which expanded to zeros are ignored to visualize the non-zero effective elements better. The blocks A, E, B, C, and I are depicted in colors in Fig. 2. Each dot drawn in the figure will be expanded into 128. For instance, every blue dot appearing in the $l$ block has a value of 0 in $B_2$ and will expand into a $128 \times 128$ identity matrix in $H$.  

![Fig. 2 Example of B2 Matrix Drawn with Only None -1 Elements.](Image)

**4. ENCODING IN THE 5G STANDARD**

For the sake of simplicity, an example of a $(4 \times 8)$ $B$ matrix is considered, as depicted in Eq. (4). Again, each element in $B$ is expanded (lifted) by a factor of $Z_c = 5$.

\[
B = \begin{bmatrix}
1 & -1 & 3 & 1 & 2 & 0 & -1 & -1 \\
2 & 0 & -1 & 3 & -1 & 0 & 0 & 0 \\
-1 & 4 & 2 & 0 & 1 & -1 & 0 & 0 \\
4 & 1 & 0 & -1 & 2 & -1 & 0 & 0
\end{bmatrix}
\]

(4)

Equation (4) can be rewritten with another notation for a more appropriate mathematical representation shown afterward. The resultant graph matrix becomes as follows:

\[
B = \begin{bmatrix}
l_1 & 0 & l_3 & l_1 & l_2 & l_1 & 0 & 0 \\
l_2 & l_1 & l_3 & l_1 & l_2 & l_1 & 0 & 0 \\
0 & l_4 & l_1 & l_3 & l_1 & l_2 & l_1 & 0 \\
l_4 & l_1 & l_3 & l_1 & l_2 & l_1 & 0 & 0
\end{bmatrix}
\]

(5)

where $l_i$ is a $5 \times 5$ identity matrix circularly shifted right $i$ times;

The $B$ matrix in Eq. (5) has $(n - k) \times n$ dimension. Hence, in the systematic form:

- The message is $[m_1 \; m_2 \; ... \; m_{10}]$, where $m_i$ is 128 bits.
- The codeword is represented as $[m_1 \; m_2 \; ... \; m_{10} \; p_1 \; p_2 \; p_3 \; p_4 \; ... \; p_{12}]$, where $p_i$ is 128 bits parity-check part.
- Using Eq. (6), the double-diagonal encoding uses the first four rows of $B_2$ to determine the parity checks $p_3$, $p_2$, $p_3$, and $p_4$.
- The 6th row is used to find $p_6$. The 7th row is used to find $p_7$, and so on through the last row to find $p_{12}$.

\[
\begin{bmatrix}
l_1 & 0 & l_3 & l_1 & l_2 & l_1 & 0 & 0 \\
l_2 & l_1 & l_3 & l_1 & l_2 & l_1 & 0 & 0 \\
0 & l_4 & l_1 & l_3 & l_1 & l_2 & l_1 & 0 \\
l_4 & l_1 & l_3 & l_1 & l_2 & l_1 & 0 & 0
\end{bmatrix}
\]

(12)

5. DECODING OF P-LDPC

The decoding of LDPC codes is done in a soft-in-soft-output (SISO) iterative manner. Although the iterative decoding based on the sum-product (SP) algorithm can excellently perform, the high complexity is an effective

(6) to reveal the four equations in Eq. (7) through Eq. (10).

\[
B C^T = 0 
\]

(6)

\[
l_1 m_1 + l_3 m_3 + l_1 m_4 + l_2 p_1 + l_2 p_2 = 0 
\]

(7)

\[
l_2 m_1 + l_1 m_2 + l_3 m_4 + l_2 p_2 + l_3 p_3 = 0 
\]

(8)

\[
l_4 m_2 + l_3 m_3 + l_1 m_4 + l_1 p_1 + l_3 p_4 = 0 
\]

(9)

\[
l_4 m_1 + l_3 m_2 + l_3 m_3 + l_3 p_1 + l_4 p_4 = 0 
\]

(10)

By adding the four preceding formulæ, they yield:

\[
l_1 p_1 = l_1 m_1 + l_3 m_3 + l_1 m_4 + l_2 p_1 + l_3 m_2 + l_2 p_2 + l_4 p_3 + l_1 m_4 + l_4 m_2 + l_3 m_3 + l_3 m_4 + l_4 m_1 + l_3 m_2 + l_3 m_3 \]

(11)

Hence, the first parity-check $p_1$ is determined. Then, $p_1$ is substituted in Eq. (7) to find $p_2$, $p_2$ is substituted in Eq. (8) to find $p_3$, and $p_1$ and $p_3$ are substituted in Eq. (9) to find $p_4$.

The way the $B$ matrix is structured is called double-diagonal. That structure, shown in blue in Eq. (5), is well designed to help the encoding phase in P-LDPC codes in the 5G standard. After laying the ground in the preceding example, it is straightforward to understand the double-diagonal structure in $B_1$ and $B_2$ matrices in 5G. Again, $B_2$ with index $I_{LS} = 0$ and expansion factor $Z_c = 128$ is considered for demonstration, as appears in Eq. (12). The double-diagonal structure is surrounded and highlighted in blue color, which is block $E$ in the structure in Eq. (3).
concern in the implementation stage. Min-sum (MS) algorithm [39] has been proposed to reduce complexity by utilizing approximation in check nodes (CN) updates of log-likelihood ratio (LLR) calculations. However, that complexity reduction is considered a trade-off with decoding performance. To compensate for the decoding performance degradation of the MS algorithm, the offset MS (OMS) algorithm [40] has been presented. That compensation is achieved by applying a correction factor ($\alpha$) added to the CN’s output LLR directly.

In this paper, the following notation and assumption are set. The codeword $C = \{c_1, c_2, \ldots, c_i\}$ is characterized by a $J \times I$ parity-check matrix of LDPC code that is mapped using the BPSK scheme to produce $S = \{s_1, s_2, \ldots, s_i\}$ symbols by $s_i = 1 - 2c_i, i \in [1, I]$. Afterward, the symbol vector passes throughout an AWGN channel to result in a received vector $r_i = s_i + g_i, i \in [1, I]$, where $g_i$ is a random variable that is zero-mean independent Gaussian with variance $\sigma^2$. Then, the SISO decoding is done based on LLRs $\text{LLR}_{ij}$ of check nodes (CN) updates. The channel LLR is used by the decoder of $C_1$ in the first iteration; only then LLR$_{21}$ is applied. That layering decoding would result in a reduction in the number of iterations for the decoder to converge.

In terms of $B$ base graph matrices in 5G standard, each row block is treated as a layer for decoding. In other words, in the case of $B_2$, it is being processed as 42 layers since it has 42-row blocks. $H = [H_1 H_2]$ (layer 1

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

(13)

Fig. 3: Example of Layered Decoding Structure

6. PUNCTURING FOR RATE MATCHING

The method of eliminating some parity check bits from the codewords is called puncturing. This process increases the code rate, which in turn involves sending more information message bits. In the 5G NR standard, different code rates are achievable via puncturing parts of the base graph matrix. If the base graph 1 (i.e., $B_1$) is considered, the 46 × 68 base matrix will expand to 46$Z_c \times 66Z_c$ parity check matrix $H$ with $68Z_c - 46Z_c = 22Z_c$ message bits. The first two column message blocks of $B_1$, or equivalently first 2$Z_c$ message bits, are punctured in 5G NR before transmission. Fig. 4 shows how puncturing of $B_1$ is employed to offer different code rates. If all remaining 66 blocks, or equivalently 66$Z_c$ bits, are considered for transmission, the resultant code rate is $R = 22Z_c/66Z_c = 1/3$, as shown in blue color in Fig. 4.
To increase the rate, some parts of the $B_1$, or equivalently $H$, are removed or punctured. In the case of transmitting the first $44Z_c$ bits, i.e., $20Z_c$ message bits $+24Z_c$ parity bits, and puncturing the last $22Z_c$ bits that result in $R = 22Z_c/44Z_c = 1/2$. That means only the $24 \times 46$ or $24Z_c \times 46Z_c$ upper-left part of $B_1$ or $H$, respectively, is used as appeared in red in the figure. To further increase the rate, only the first $33Z_c$ bits, i.e., $20Z_c$ message bits $+13Z_c$ parity bits, are sent to result in $R = 22Z_c/33Z_c = 2/3$. This rate is achieved by considering only the $13 \times 35$ upper-left part of $B_1$, as represented in green in the figure. Other variants of rates are allowed, and the same puncturing mechanism is correspondingly applicable for $B_2$. Bit interleaving is the second stage used in standard after puncturing for rate matching in higher-order modulation. However, this study considers only puncturing since the BPSK scheme is applied.

7. RESULTS

7.1. Results of P-LDPC vs. Other Coding Schemes

First, the P-LDPC code’s performance is compared with polar codes [6] and block turbo codes [42]. For the P-LDPC code, base graph 2 (BG2), expansion factor $Z_c = 52$, and layered OMS decoding were chosen for evaluation purposes. For polar codes, the selected decoding algorithm is the successive cancellation list (SCL) [43] with list sizes ($L=4$ and $L=8$) aided by a cyclic redundancy check (CRC) with 11 bits long [13, 44]. For block turbo codes, a 2-dimension turbo product code (2-D TPC) is chosen by serially concatenating an extended Bose–Chaudhuri–Hocquenghem (BCH) code and an extended Hamming code. For all codes above, the codeword length was $n \approx 1024$, and puncturing/shortening was performed to offer a code rate of $R = 1/2$ for comparison fairness. The number of transmitted blocks was kept at 50000 at its peak with a maximum number of iterations of 20. The scenario of transmitting coded BPSK symbols over an AWGN was assumed.

The frame (or block) error rate (FER) and bit error rate (BER) were calculated for different values of bit energy to noise spectral density ratio ($E_b/N_0$), as depicted in Fig. 5.

For BER, shown in solid lines, all codes’ curves are growing progressively as the $E_b/N_0$ increases but with different rates of change. They show convergence in performance for low values of $E_b/N_0$ and start to diverge for higher dB values. The Polar SCList8-CRC offers better BER and FER behavior than Polar SCList4-CRC since the list size is double, which allows for better bit-wise decoding. On the other hand, 2D-TPC proposes a better BER than Polar codes for almost all dB values. Even though each of Polar SCList4-CRC, Polar SCList8-CRC, and 2D-TPC codes gave a reasonably good performance, it is explicitly shown that P-LDPC offers the best behavior in terms of both FER and BER compared with its counterparts.

7.2. Results of P-LDPC with Different Settings

After getting the P-LDPC evaluated and compared with other coding schemes, this section discusses some simulation results of P-LDPC with various selections of base graphs, block lengths, and coding rates. The layered OMS decoding and BPSK over AWGN were adopted.

For base graphs, according to Table 1, index $i_{LS} = 1$ with expansion factor $Z_c = 24$ and $i_{LS} = 6$ with $Z_c = 52$ were selected for BG1 and BG2, respectively. The two matrices, $B_1$ and $B_2$, were punctured to produce a code rate $R = 1/2$. After puncturing, each row of BG1, i.e., in $B_1$, became 44 long and expanded to $44 \times 24 = 1056$ bits as codeword length ($n$). Similarly, BG2 produced $20 \times 52 = 1040$ bits as codeword length ($n$). Consequently, the two base graphs were expanded to those specific expansion factor values to guarantee each generates $n \approx 1024$ for comparison fairness. Fig. 6 shows that the two codes behave well with enhancement as $E_b/N_0$ increases. Both codes perform steadily,
almost similar for the region between $E_b/N_0 = 1$ and 1.75 dB, with a small divergence above that value. In general, it is explicit that the code generated by BG2 outperforms that produced by BG1 in terms of FER and BER for higher $E_b/N_0$. That is because the recommended selection of base graphs, especially for $R = 1/2$, is BG2 [13].

As the number of iterations contributes to decoding complexity, it is beneficial to consider that in this context. Fig. 7 shows how many iterations the same two base graphs with $R = 1/2$ require to achieve a certain FER for different values of $E_b/N_0$. As depicted in Fig. 7, the number of iterations is inversely proportional to $E_b/N_0$. Even though BG1 requires two more iterations than BG2 at the launch of the graph, both codes converge and perform similarly for higher $E_b/N_0$ to achieve only four iterations for each code at $E_b/N_0 = 2.5$ dB.

To investigate the impact of increasing/decreasing the block lengths on the code’s performance, BG2 with $R = 1/2$ was applied with different values of expansion factors. All available factors at index $i_L = 6$ were selected, i.e., $Z_e = 13, 26, 52, 104,$ and 208, which produce $n = 260, 520, 1040, 2080,$ and 4160 bits, respectively. In terms of BER performance, Fig. 8 exposes the poorest behavior, presented by the shortest length at $n = 260$. In contrast, the sharpest decline that indicates the best performance is noted at the longest length, $n = 4160$, to offer a BER of 0.00001 at 1.5 dB of $E_b/N_0$. For FER performance, Fig. 9 depicts the same behavior that explicitly shows that the codes’ performance is excellently improved as their block lengths get longer. Nevertheless, the code complexity and block length are a matter of compromise.

The coding rate at all the preceding scenarios was fixed at $R = 1/2$. However, variants of the rates are important to study. BG1 and BG2 construction allows for several coding rates via puncturing. BG2 was chosen to evaluate the performance at $R = 1/3, 2/5, 1/2,$ and 2/3. Fig. 10 shows the BER performance of those different coding rates in BG2. It is distinct that as the coding rate increased, the BER curve was shifted to the right for higher $E_b/N_0$. It indicates that the puncturing part of a codeword is beneficial for transmitting more information bits but at the expense of performance degradation. For the highest rate at $R = 2/3$, a 3.25 dB of $E_b/N_0$ is required to achieve a BER of 0.00003. This BER is effectively offered by the
code with $R = 1/3$ and only 1.55 dB. In general, the less the coding rate, the sharper fall (better) the graph. Consequently, there is a compromise in terms of coding rate and performance for P-LDPC codes in the 5G NR standard and for other codes in general.

7.3. Results of P-LDPC Over a Rayleigh Channel

Unlike the previous two sections, the P-LDPC evaluation was performed by transmitting data via a Rayleigh flat-fading channel instead of an AWGN channel. The block length and code rate were maintained at 1024 and 1/2, respectively. Fig. 11 depicts that the P-LDPC behavior is reasonably good since it offers a BER of 0.00003 at $E_b/N_0 = 11.5$ dB. Moreover, it is shown that the uncoded BPSK requires 24 dB to achieve a BER of 0.001, while it takes only 11.5 dB for P-LDPC-coded BPSK to offer the same level of BER. Consequently, the P-LDPC code proposes a 12.5 dB coding gain.

Furthermore, the performance of P-LDPC will undoubtedly be improved for longer block lengths and/or lower coding rates.

8. CONCLUSION

This study enhances the P-LDPC code performance based on base graph construction specified in the 5G NR standard. The BER/FER behavior was evaluated using the OMS algorithm with layering for P-LDPC. The simulation results showed that this constructed code excellently behaved well as $E_b/N_0$ increased and outperformed its counterparts, such as CRC-aided polar SCL and TPC. The variant expansion factors were adopted to achieve different codeword lengths, and puncturing was exploited to raise the code rate from 1/3 through 2/3. Based on the results, it is explicit how an increase in coding rate and a decrease in code word length would lead to performance degradation. Furthermore, P-LDPC performed well when coded-BPSK symbols transmitted over a Rayleigh flat-fading channel to deliver 12.5 dB coding gain at 0.001 BER compared with the uncoded transmission.

On the other hand, the presented P-LDPC code can be further enhanced for future work. Firstly, more efficient mechanisms of the extrinsic LLR exchange among CNs and VCs can be adopted to improve code performance. Secondly, the OMS algorithm might be implemented with an early stopping criterion. That means the decoding process would end once a certain threshold of message beliefs is achieved. Consequently, instead of running 20 iterations (used in this paper), an approach similar to BER will be offered by fewer iterations. Thirdly, the implemented code performance in this study is bad over the frequency-selective channel. Hence, further consideration of the high Doppler effect might be examined. The system can perform better by realizing an equalizer over MIMO-OFDM channels [45]. Fourthly, higher-order modulation schemes, such as 16-QAM and 64-QAM, can be investigated to examine the code’s behavior [46], especially with CRC attachment. Lastly, this study is initially intended to be applied over an AWGN channel, and then, it will be extended to realize P-LDPC in the Internet of underwater things (IoUT). Underwater channels are very harsh environments where the Doppler shift is extremely high, and special adaptive compensation algorithms are crucial to implement to deal with such severe conditions [47].

REFERENCES


