Fifteen Years of Quantum LDPC Coding and Improved Decoding Strategies

ZUNAIRA BABAR, PANAGIOTIS BOTSINIS, (Student Member, IEEE), DIMITRIOS ALANIS, (Student Member, IEEE), SOON XIN NG, (Senior Member, IEEE), AND LAJOS HANZO, (Fellow, IEEE)

School of Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, U.K.
Corresponding author: L. Hanzo (lh@ecs.soton.ac.uk)

This work was supported by the European Research Council Advanced Fellow Grant through the Engineering and Physical Sciences Research Council U.K. under Grant EP/L018659/1.

ABSTRACT The near-capacity performance of classical low-density parity check (LDPC) codes and their efficient iterative decoding makes quantum LDPC (QLPDC) codes a promising candidate for quantum error correction. In this paper, we present a comprehensive survey of QLDPC codes from the perspective of code design as well as in terms of their decoding algorithms. We also conceive a modified non-binary decoding algorithm for homogeneous Calderbank–Shor–Steane-type QLDPC codes, which is capable of alleviating the problems imposed by the unavoidable length-four cycles. Our modified decoder outperforms the state-of-the-art decoders in terms of their word error rate performance, despite imposing a reduced decoding complexity. Finally, we intricately amalgamate our modified decoder with the classic uniformly reweighted belief propagation for the sake of achieving an improved performance.

INDEX TERMS Quantum error correction, low density parity check codes, quantum low density parity check codes, iterative decoding.

I. INTRODUCTION Operating close to Shannon’s channel capacity limit is only feasible under the idealized conditions of perfect synchronization, perfect channel estimation and in case of potentially infinite delay/complexity channel codes. It was demonstrated in [1] that only a fraction of the theoretical limit is achievable in realistic scenarios. The logarithmic increase in the idealized Shannonian capacity with the transmit power imposes another limitation. Nevertheless, provided that we can create a sufficiently high number of parallel streams and that we have a low-complexity full-search-based multi-stream detector, the throughput of the wireless system may be increased linearly with the transmit power. Unfortunately, the associated optimal full-search-based multi-stream detectors have an excessive complexity, which increases exponentially both with the number of users as well as with that of the antennas. Since quantum-based parallel computation is capable of...
solving certain complex problems at a substantially lower complexity than its classical counterpart, quantum parallel processing techniques may be invoked [2]–[6]. The peculiar laws of quantum mechanics have also spurred interest in quantum-based communication systems, which have given rise to a new range of security paradigms in the context of quantum key distribution techniques [7], [8], quantum secure direct communication [9], [10] and unconditional quantum location verification [11].

Unfortunately, quantum noise, conventionally termed as ‘decoherence’, imposes a hitherto insurmountable impairment on the practical implementation of quantum computation as well as on quantum communication systems. More precisely, decoherence is the undesirable interaction of the constituent qubits\(^1\) with the environment, which perturbs the superposition of states [13], [14]. For the sake of having a reliable quantum computation or communication system, it is desired to counteract the above-mentioned decoherence so that the qubits retain their coherent quantum states for practical durations. Since decoherence may be characterized either by bit-flips or phase-flips or in fact possibly by both, Quantum Error Correction Codes (QECCs), designed for correcting both bit-flips as well as phase-flips, may be invoked for correcting the errors inflicted on the qubits [13].

The inception of QECCs dates back to 1995 when Shor [13] conceived the first quantum code, which was however only capable of correcting a single error. Since then the quest for approaching the quantum capacity bounds at an affordable encoding and decoding complexity has continued. In this context, the astounding performance of the classical Low Density Parity Check (LDPC) codes [15]–[19], which exhibit a near-capacity performance at an affordable decoding complexity, has inspired the community to design Quantum Low Density Parity Check (QLDPC) codes. The sparseness of the QLDPc matrix is of particular interest in the quantum domain, because it requires only a small number of interactions per qubit during the error correction procedure, thus facilitating fault-tolerant decoding.

QLDPC codes belong to the family of Quantum Stabilizer Codes (QSCs) [20], [21], which is a generalized formalism for designing quantum codes from any arbitrary classical binary and quaternary codes\(^2\). However, this transfiguration from the classical to the quantum domain imposes a stringent symplectic criterion on the parent classical codes, which brings with it various design challenges. Against this backdrop, in this paper we survey the evolution of QLDPC code designs, focusing on the various code constructions to conceive powerful QLDPC codes from the known families of classical LDPC codes. We also review the syndrome-based iterative decoding algorithms invoked for QLDPC codes. Finally, we conceive a modified non-binary decoding algorithm for homogeneous Calderbank-Shor-Steane (CSS)-type QLDPC codes, which is capable of mitigating the impact of the unavoidable length-4 cycles. Our modified decoder exhibits a superior Word Error Rate (WER) performance, despite its reduced decoding complexity, when compared to the state-of-the-art decoding techniques. We demonstrate furthermore that the Uniformly-Reweighted Belief Propagation (URW-BP) technique of [23] and [24] may also be invoked for further improving the attainable performance.

\(1\)In contrast to a classical bit, which can either assume a value of 0 or 1, a qubit can exist in a superposition of the two states represented as \(|\psi\rangle = \alpha|0\rangle + \beta|1\rangle\). Here, \(\{\\rangle\) is called Dirac notation or Ket [12], while \(\alpha\) and \(\beta\) are complex numbers with \(|\alpha|^2 + |\beta|^2 = 1\). More specifically, a qubit exists in a continuum of states between \(|0\rangle\) and \(|1\rangle\) until it is `measured` or ‘observed’. Upon ‘measurement’ it collapses to the state \(|0\rangle\) with a probability of \(|\alpha|^2\) and \(|1\rangle\) with a probability of \(|\beta|^2\).

\(2\)For a detailed description of the transition from the classical codes to the quantum codes, readers are referred to [22].

This paper is organized as depicted in Fig. 1. We commence with a summary of the stabilizer code design formalism in Section II. We then proceed with a review of QLDPC code designs in Section III, while a range of powerful decoding techniques are discussed in Section IV. Finally, we present our proposed decoding algorithm in Section V, while in Section VI we detail the reweighted belief propagation philosophy. Our simulation results are presented in Section VII, while Section VIII concludes our discourse.

**II. STABILIZER FORMALISM**

Let us first state the important definitions used for describing the stabilizer code formalism [25].

**A. PAULI OPERATORS**

The I, X, Y and Z Pauli operators are defined by the following matrices:

\[
I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}
\]  \hspace{1cm} (1)
where the X, Y and Z operators anti-commute with each other.

**B. PAULI GROUP**

A single qubit Pauli group \(G_1\) is a group formed by the Pauli matrices \(I, X, Y\) and \(Z\), which is closed under multiplication. Therefore, it consists of all the Pauli matrices together with the multiplicative factors \(\pm 1\) and \(\pm i\), i.e. we have:

\[
G_1 \equiv \{\pm 1, \pm i, \pm X, \pm iX, \pm Y, \pm iY, \pm Z, \pm iZ\}. \tag{2}
\]

The general Pauli group \(G_n\) is an \(n\)-fold tensor product of \(G_1\).

**C. DEPOLARIZING CHANNEL**

A depolarizing channel, which is characterized by the probability \(p\), inflicts an error \(\mathcal{P} \in G_n\) on \(n\) qubits, where each qubit may independently experience either a bit-flip (X), a phase-flip (Z) or both (Y) with a probability of \(p/3\) each, when considering the default symmetric depolarizing channel.

Fig. 2 shows the general schematic of a quantum communication system. At the transmitter TX, an \([n, k]\) QSC, constructed over a code space \(C\), maps the information word (logical qubits) \(|\psi\rangle \in C^{2^n}\) onto the codeword (physical qubits) \(|\tilde{\psi}\rangle \in C^{2^n}\) with the aid of \((n-k)\) auxiliary (also called ancilla) qubits initialized to the state \(|0\rangle\). Here \(C^d\) denotes the \(d\)-dimensional Hilbert space. Furthermore, let \(\mathcal{P} \in G_n\) be the channel error inflicted on the transmitted codewords. Consequently, \(|\psi\rangle = \mathcal{P}|\tilde{\psi}\rangle\) is the noisy codeword received at the receiver RX, which invokes a 3-step decoding procedure for recovering the intended transmitted information \(|\tilde{\psi}\rangle\).

Unlike a classical decoder, which measures the received bit values, a quantum decoder cannot measure the received qubits without perturbing their superimposed quantum states. More specifically, qubits collapse to classical bits upon their measurement/observation. Therefore, inspired by the Parity Check Matrix (PCM)-based syndrome decoding of classical codes [26], a quantum decoder circumvents the associated measurement operation by observing the error syndromes without reading the actual quantum information. In the context of an \([n, k]\) QSC, this is achieved by a set of \((n-k)\) independent commuting Pauli generators \(g_i \in G_n\), for \(1 \leq i \leq (n-k)\). The corresponding stabilizer group \(\mathcal{H}\) contains both \(g_i\) and all the products of \(g_i\) for \(1 \leq i \leq (n-k)\) and forms an Abelian subgroup of \(G_n\). A unique feature of these generators is that they do not change the state of valid codewords, while yielding an eigenvalue of \(-1\) for the corrupted states. Consequently, the eigenvalue is \(-1\) if \(\mathcal{P}\) anti-commutes with the stabilizer \(g_i\) and it is \(+1\) if \(\mathcal{P}\) commutes with \(g_i\), which can be formulated as:

\[
g_i|\tilde{\psi}\rangle = \begin{cases} |\tilde{\psi}\rangle & \text{if } g_i \mathcal{P} = \mathcal{P} g_i \\ -|\tilde{\psi}\rangle & \text{if } g_i \mathcal{P} = -\mathcal{P} g_i \end{cases} \tag{3}
\]

The resultant \(\pm 1\) eigenvalue gives the corresponding error syndrome, which is 0 for an eigenvalue of \(+1\) and 1 for an eigenvalue of \(-1\). Hence, within the ‘syndrome processing’ block of Fig. 2, the receiver RX computes the syndrome of the received sequence \(|\tilde{\psi}\rangle\) and uses it to estimate the channel-induced error pattern \(\mathcal{P}\). The recovery operator \(\mathcal{R}\) then uses the estimated error pattern \(\mathcal{P}\) to restore the potentially error-free transmitted coded stream. It must be mentioned here that those channel errors, which differ only by the stabilizer group, have the same impact on all the codewords and therefore can be corrected by the same recovery operation. This equips quantum codes with the intrinsic property of degeneracy [27]. More explicitly, the error patterns \(\mathcal{P}\) and \(\mathcal{P}' = g_i \mathcal{P}\) have the same impact on the transmitted codeword and therefore can be corrected by the same recovery operation. Finally, the ‘inverse encoder’ shown in Fig. 2, processes the recovered coded sequence \(|\tilde{\psi}\rangle\), yielding the estimated transmitted information qubits \(|\tilde{\tilde{\psi}}\rangle\).

QSCs may be characterized in terms of an equivalent binary parity check matrix notation satisfying the commutativity constraint of the stabilizer generators [28], [29]. This is achieved by mapping the \(I, X, Y\) and \(Z\) Pauli operators onto \((\mathbb{F}_2)^2\) as follows:

\[
\begin{array}{cccc}
I & \rightarrow (0, 0), & X & \rightarrow (0, 1), & Y & \rightarrow (1, 1), & Z & \rightarrow (1, 0). \tag{4}
\end{array}
\]

\[3\]For example, consider a 3-qubit bit-flip repetition code, which encodes \(|\psi\rangle = \alpha |0\rangle + \beta |1\rangle\) into \(|\tilde{\psi}\rangle = \alpha |000\rangle + \beta |111\rangle\), and has the generators \(g_1 = ZZI\) and \(g_2 = ZIZ\). Both \(g_1\) as well as \(g_2\) do not affect a valid codeword, i.e. \(g_1 |\tilde{\psi}\rangle = g_2 |\tilde{\psi}\rangle = |\tilde{\psi}\rangle\). On the other hand, let the received codeword be \(|\bar{\psi}\rangle = \alpha |010\rangle + \beta |101\rangle\), then we get \(g_1 |\bar{\psi}\rangle = -|\bar{\psi}\rangle\), while \(g_2 |\bar{\psi}\rangle = |\bar{\psi}\rangle\). Therefore, the resulting syndromes are 1 and 0, respectively.

\[4\]For the 3-qubit bit-flip repetition code, let \(\mathcal{P} = IIX\) and \(\mathcal{P}' = g_1\mathcal{P} = ZZX\) be the errors inflicted by the channel. Both \(\mathcal{P}\) as well as \(\mathcal{P}'\) corrupt the transmitted codeword \(|\tilde{\psi}\rangle = \alpha |000\rangle + \beta |111\rangle\) to \(|\bar{\psi}\rangle = \alpha |001\rangle + \beta |110\rangle\). Consequently, the errors \(\mathcal{P}\) and \(\mathcal{P}'\) need not be differentiated and are therefore classified as degenerate errors.
Based on this Pauli-to-binary isomorphism, the \((n-k)\) stabilizers of an \([n,k]\) stabilizer code constitute the rows of the binary PCM \(H\), which is a concatenation of a pair of \((n-k)\times n\) binary matrices \(H_z\) and \(H_x\), as given below:

\[
H = (H_z|H_x).
\]  

(5)

Each row of \(H\) corresponds to a stabilizer of \(\mathcal{H}\), so that the \(i\)th columns of \(H_z\) and \(H_x\) are used to compute the error imposed on the \(i\)th qubit. More specifically, a binary 1 in the \(i\)th column of \(H_z\) (or \(H_x\)) represents a \(Z\) (or \(X\)) Pauli operator in the corresponding stabilizer. Furthermore, given the matrix notation of Eq. (5), the commutative property of stabilizer generators is transformed into the orthogonality of rows with respect to the symplectic product (also referred to as a twisted product). If row \(i\) is \(H_i = (H_{i,z}, H_{i,x})\), where \(H_{i,z}\) and \(H_{i,x}\) are the binary strings for \(Z\) and \(X\) respectively, then the symplectic product of rows \(i\) and \(j\) is given by,

\[
H_i \cdot H_j = (H_{i,z}' \cdot H_{j,z} + H_{i,x}' \cdot H_{j,x}) \mod 2.
\]  

(6)

The symplectic product of Eq. (6) is zero if there are an even number of places in the generators \(g_i\) and \(g_j\) with different non-Identity (i.e. \(X\), \(Y\), or \(Z\)) operators; thus meeting the commutativity requirement.\(^5\) We may further deduce from Eq. (6) if \(H\) is expressed as \(H = (H_z|H_x)\), then the symplectic product is satisfied for all the rows of \(H\) if and only if we have:

\[
H_z H_x^T + H_x H_z^T = 0.
\]  

(7)

Consequently, any classical code satisfying Eq. (7) may be used for constructing QSCs.

A special class of stabilizer codes is constituted by the family of CSS codes, invented independently by Calderbank and Shor [30] as well as by Steane [31, 32], which facilitates the design of high-performance quantum codes from the known family of classical binary linear codes. More explicitly, a \([n,k_1 - k_2]\) CSS code, which is capable of correcting \(t\) bit-flips as well as phase-flips, can be constructed from the classical linear block codes \(C_1(n,k_1)\) and \(C_2(n,k_2)\), if we have \(C_2 \subseteq C_1\), and both \(C_1\) and the dual of \(C_2\), i.e. \(C_2^\perp\), can correct \(t\) errors. In CSS construction, the PCM \(H_z'\) of \(C_1\) is used for correcting bit-flips, while the PCM \(H_x'\) of \(C_2^\perp\) is used for phase-flip correction. Consequently, the PCM of the resultant CSS code assumes the following form:

\[
H = \begin{pmatrix}
H_z' & 0 \\
0 & H_x'
\end{pmatrix}.
\]  

(8)

\(^5\)For example, let \(g_1 = \text{XZI}\) and \(g_2 = \text{ZII}\), which have different non-Identity Pauli operators only at the first index. Then the generators \(g_1\) and \(g_2\) anti-commute. Alternatively, according to the binary mapping of Eq. (4), we have \(g_1 \equiv (010)|100\), while \(g_2 \equiv (100)|000\). Therefore, the symplectic product of Eq. (6) yields a value of 1. By contrast, if the generators are \(g_1 = \text{XZI}\) and \(g_2 = \text{ZII}\), then they commute. Consequently, the symplectic product of Eq. (6) gives a value of zero.

where we have \(H_z = \begin{pmatrix} H_z' & 0 \\ 0 & H_x' \end{pmatrix}\), \(H_x = \begin{pmatrix} 0 & H_x' \\ H_z' & 0 \end{pmatrix}\) and both \(H_z'\) and \(H_x'\) are \((n - k_1) \times n\) and \(k_2 \times n\) binary matrices, respectively. Furthermore, since we have \(C_2 \subseteq C_1\), the symplectic condition of Eq. (7) is reduced to \(H_z H_x^T = 0\). For the specific case where \(H_z' = H_x'^\perp\), the resultant structure is termed as a dual-containing (or self-orthogonal) code because \(H_z'^\perp H_z^T = 0\), which is equivalent to \(C_1^\perp \subseteq C_1\).

**TABLE 1.** GF(4) addition.

<table>
<thead>
<tr>
<th>+</th>
<th>0</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>

**TABLE 2.** GF(4) multiplication.

<table>
<thead>
<tr>
<th>×</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Since the \(I, X, Y\) and \(Z\) Pauli operators have the equivalent 2-bit representation of Eq. (4), they may also be expressed in the Galois Field GF(4) by the equivalent 4-ary symbols. More specifically, the Pauli-to-GF(4) isomorphism may be encapsulated as:

\[
\begin{align*}
I & \rightarrow 0, & X & \rightarrow 1, & Y & \rightarrow \omega, & Z & \rightarrow \bar{\omega},
\end{align*}
\]  

(9)

where 0, 1, \(\omega\) and \(\bar{\omega}\) are the elements of GF(4), which conform to the additive and multiplicative rules of Table 1 and Table 2, respectively. According to the Pauli-to-GF(4) isomorphism, the multiplication of Pauli operators is transformed into the addition of the corresponding elements in GF(4), while the commutativity (symplectic product) criterion is mapped onto the trace\(^6\) inner product\([21]\). For example, multiplying the set of Pauli operators \([I, X, Z, Y]\) with Pauli-\(X\) is equivalent to the second column of Table 1. Furthermore, the commutative relationship between \(A\) and \(B\) in GF(4) is computed using the trace inner product as follows\(^7\):

\[
\text{Tr}(A \times B) = \text{Tr}(\bar{A} \times \bar{B}) = 0.
\]  

(10)

where \(|,\rangle\rangle\) represents the Hermitian inner product and \(\bar{B}\) denotes the conjugate of \(B\). Furthermore, \(\text{Tr}(0) = \text{Tr}(1) = 0\), while \(\text{Tr}(\omega) = \text{Tr}(\bar{\omega}) = 1\). Consequently, based on Eq. (10), the symmetric product of Eq. (6) is transformed into the trace inner product in GF(4). For example, the symplectic product of the \(i\)th and \(j\)th row of \(H\), which is defined in GF(4), is formulated as:

\[
\bar{H}_i \cdot \bar{H}_j = \text{Tr}(\bar{H}_i \cdot \bar{H}_j) = \text{Tr}\left(\sum_{t=1}^{n} \bar{H}_{it} \times \bar{H}_{jt}\right).
\]  

(11)

\(^6\)In GF(4), the trace operator maps \(x\) to \((x + \bar{x})\), where \(\bar{x}\) is the conjugate of \(x\) [33].

\(^7\)We denote GF(4) variables with a * on top, e.g. \(\hat{x}\).
where $\hat{H}_{it}$ denotes the element in the $i$th row and $r$th column of $\hat{H}$.

**TABLE 3. Quantum-to-classical isomorphism.**

<table>
<thead>
<tr>
<th>Pauli</th>
<th>$(\mathbb{F}_2)^2$</th>
<th>GF(4)</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>(0, 0)</td>
<td>0</td>
</tr>
<tr>
<td>X</td>
<td>(0, 1)</td>
<td>1</td>
</tr>
<tr>
<td>Z</td>
<td>(1, 0)</td>
<td>$\omega$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Multiplicativity</th>
<th>Bit-wise Addition</th>
<th>Symplectic Product</th>
<th>Trace Inner Product</th>
</tr>
</thead>
</table>

Based on the above discussions, a Pauli operator may be expressed in terms of the equivalent binary or quaternary representation, which is summarized in Table 3. This in turn facilitates the design of quantum codes from the known classical codes. More explicitly, arbitrary classical binary and quaternary codes may only be used for constructing QSCs if they satisfy the commutativity criterion of Eq. (7). Consequently, some of the good classical codes cannot be exploited in the quantum domain. The entanglement-assisted stabilizer formalism of [34]–[37] overcomes this limitation by using pre-shared entanglement\(^9\) between the transmitter and receiver to embed a set of non-commuting stabilizer generators into a larger set of commuting generators.

Fig. 3 shows the general schematic of a quantum communication system, which incorporates an Entanglement-Assisted Quantum Stabilizer Code (EA-QSC). An $[n, k, c]$ EA-QSC encodes the information qubits $|\psi\rangle$ into the coded sequence $|\tilde{\psi}\rangle$ with the aid of $(n-k-c)$ auxiliary qubits, which are initialized to the state $|0\rangle$. Furthermore, the transmitter and receiver share $c$ entangled qubits (ebits) before actual transmission takes place. This may be carried out outside the busy hour, when the channel is under-utilized, thus efficiently distributing the transmission requirements in time. More specifically, the state $|\phi^+\rangle$ of an ebit is given by the following Bell state:

$$|\phi^+\rangle = \frac{|00\rangle_{T_X R_X} + |11\rangle_{T_X R_X}}{\sqrt{2}},$$

where $T_X$ and $R_X$ denotes the transmitter’s and receiver’s half of the ebit, respectively. Similar to the superdense coding protocol of [38], it is assumed that the receiver’s half of the c ebits are transmitted over a noiseless quantum channel, while the transmitter’s half of the c ebits together with the $(n - k - c)$ auxiliary qubits are used for encoding the intended $k$ information qubits into $n$ coded qubits. The resultant $n$-qubit codewords $|\tilde{\psi}\rangle$ are transmitted over a noisy quantum channel. The receiver then combines his half of the $c$ noiseless ebits with the received $n$-qubit noisy codewords $|\psi\rangle$ to compute the syndrome, which is used for estimating the error pattern $\mathcal{P}$ incurred on the $n$-qubit transmitted codewords. The rest of the processing at the receiver is the same as that in Fig. 2.

The entangled state of Eq. (12) has unique commutativity properties, which assist us in transforming a set of non-Abelian generators into an Abelian set. The state $|\phi^+\rangle$ is stabilized by the operators $X^{T_X}X^{R_X}$ and $Z^{T_X}Z^{R_X}$, which commute with each other. Therefore, we have\(^10\):

$$[X^{T_X}X^{R_X}, Z^{T_X}Z^{R_X}] = 0.\quad (13)$$

However, local operators acting on either of the qubits anti-commute, i.e. we have:

$$[X^{T_X}, Z^{T_X}] = [X^{R_X}, Z^{R_X}] = 0.\quad (14)$$

Therefore, if we have two single qubit operators $X^{T_X}$ and $Z^{T_X}$, which anti-commute with each other, then we can resolve the anti-commutativity by entangling another qubit and choosing the local operators on this additional qubit such that the resultant two-qubit generators $X^{T_X}X^{R_X}$ and $Z^{T_X}Z^{R_X}$ (for this case) commute. This additional qubit constitutes the receiver half of the ebit. In other words, we entangle an additional qubit for the sake of ensuring that the resultant two-qubit operators have an even number of places with different non-identity operators, which in turn ensures commutativity.\(^11\)

**III. QUANTUM LDPC CODE DESIGNS**

Analogous to classical LDPC codes, which belong to the family of linear block codes, QLDPC codes are inherently stabilizer codes, which may be characterized using an equivalent classical Parity Check Matrix (PCM) $H$ of Eq. (5).

\(^9\)Two qubits are said to be entangled if they cannot be decomposed into the tensor product of the constituent qubits. Consequently, a peculiar link exists between the two qubits such that measuring one qubit also collapses the other, despite their spatial separation [25].

\(^10\)[$a, b]$ represents the commutative relation between $a$ and $b$, while $[a, b]$ denotes the anti-commutative relation.

\(^11\)For example, if $g_1 = XZI$ and $g_2 = ZII$, which anti-commute, then we can resolve the anti-commutativity by using an additional entangled qubit for extending the generators $g_1$ and $g_2$ to $g_1' = XZIX$ and $g_2' = ZIZZ$, respectively, where the Pauli operators to the left of the vertical bar (\(\vert\)) act on the $n$-qubit codeword, while that to the right of the vertical bar acts on the receiver’s half of the ebit.
More specifically, an \([n, k]\) QLDPC code having a coding rate of \(R_Q = k/n\) is equivalent to a \((2n, n + k)\) binary LDPC code having a coding rate of \(R_c = (n + k)/2n\). We may divide the QLDPC codes into three main categories on the basis of the general global structure of the associated PCM \(H\), namely Calderbank-Shor-Steane (CSS) codes, non-CSS codes and Entanglement-Assisted (EA) codes, as summarized in Fig. 4.

The CSS-type constructions may also be classified as dual-containing and as non-dual-containing codes. Let us now take a look at each of these categories individually.

![FIGURE 4. Classification of QLDPC codes.](image)

**A. CALDERBANK-SHOR-STEANE CODES**

Ideally, any two classical binary LDPC codes, which meet the symplectic criterion, may be used for constructing a CSS-based QLDPC code. However, randomly choosing the constituent pair of classical codes is not feasible, because finding two sparse codes, which satisfy the stringent symplectic constraint, is highly unlikely. This motivated Postol [39] to conceive the first example of a CSS-based non-dual-containing QLDPC code from a small \((15, 7)\) finite geometry based classical LDPC code in 2001. More specifically, in Postol’s code, the PCM of a finite geometry based cyclic classical LDPC code constitutes the \(H^T\) of Eq. (8), while \(H^T_\eta\) is derived from \(H_\eta\), so that the symplectic criterion is satisfied, i.e. we have \(H_\eta^T H_\eta^T = 0\). Since both the constituent PCMs, i.e. \(H^T_\eta\) and \(H^T_\epsilon\), are cyclic, this facilitates the implementation of the encoder. However, Postol did not develop a generalized method for his proposed design, which could facilitate the construction of QLDPC codes from any arbitrary finite geometry based classical LDPC codes. This gap was filled by Mackay et al. in [29], where several systematic constructions were developed for the CSS-based QLDPC codes by restricting the designs to the dual-containing structure.

Before proceeding with the constructions of [29], let us take a look at the symplectic condition of Eq. (7) in the context of the dual-containing QLDPC codes. Recall from Section II that the symplectic criterion of Eq. (7) reduces to \(H^T_\eta H^T_\eta = 0\) for the dual-containing QLDPC codes, which have \(H^T_\eta = H^T_\eta\). This in turn implies that the PCM of a classical LDPC code may only be used for constructing a dual-containing QLDPC code if:

1. it has an even row weight; and
2. every pair of rows has an even number of overlapping 1’s, which we may term as an ‘even overlap’.

By contrast, good classical LDPC codes must have at most a single overlapping 1 between every pair of rows for the sake of avoiding length-4 cycles because short cycles of length-4 impair the performance of the associated decoding algorithm. Consequently, the ‘even overlap’ condition results in unavoidable cycles of length 4 in the resultant PCM, as depicted in Fig. 5 for a random binary PCM \(H^T_\eta\) given by\(^ {12}\):

\[
H^T_\eta = \begin{pmatrix}
1 & 0 & 1 & 1 \\
1 & 1 & 1 & 0
\end{pmatrix}
\]

Furthermore, the constraint \(H^T_\eta H^T_\eta = 0\) also implies that the code-space of the underlying classical code must contain its dual. Hence, the resultant code contains codewords having a weight equal to the row weight \(\rho\). Therefore, the minimum distance of the classical dual-containing code is upper-bounded by \(\rho\). Surprisingly, this upper-bound does not exist for quantum codes due to the degenerate nature of quantum errors. More specifically, the \(n\)-tuple channel error pattern acting on the codewords of a QSC, may be classified as:

1. **Detected Error Patterns:** These error patterns anticommute with the stabilizers of the code, yielding a non-trivial syndrome.
2. **Harmful Undetected Error Patterns:** This class of error patterns commutes with the stabilizers. Consequently, these error patterns are harmful, because they map one valid codeword onto another; thus, corrupting the codeword without triggering a non-trivial syndrome. Harmful undetected error patterns are attributed to the small minimum distance of the code.

\(^{12}\) This is a random example for illustrating the impact of an even number of overlaps. The \(H^T_\eta\) of Eq. (15) may not be a good classical code.
3) Harmless Undetected Error Patterns: This is a unique class of error patterns, which do not have a classical analogue. Similar to the ‘harmful undetected error patterns’, these error patterns also commute with the stabilizers, but they are harmless in the context of quantum codes. This is because these are the degenerate errors, which belong to the stabilizer group, and therefore do not corrupt the state of the valid codewords. More explicitly, for dual-containing CSS codes, the harmless undetected error patterns lie in the code-space of the dual code $C_m^\perp$, as depicted in Fig. 6. It must be mentioned here that although the harmless undetected errors do not affect the minimum distance of the resultant quantum code, they lead to the ‘symmetric degeneracy error’ in the iterative decoding procedure of QLDPC codes, which will be discussed in Section IV-C.

Bicycle codes, which were proposed by Mackay et al. in [29], marked the first major breakthrough towards the realization of CSS-based dual-containing quantum LDPC codes. The proposed code design relies on a semi-random/semi-structured construction, which satisfies the dual-containing constraint by deliberately imposing a global structure on the constituent PCM. A bicycle code having a row weight of $\rho$, a block length of $n$ and $(n-k)$ stabilizers is constructed using a random sparse $n/2 \times n/2$ cyclic matrix $C_m$, which has a row weight of $\rho/2$. The non-zero entries in $C_m$ can be chosen either randomly or using a difference set satisfying the property that every difference (modulo $n/2$) occurs at most once in the set. This matrix $C_m$ is then used for constructing a base matrix $H_0$, which is a concatenation of $C_m$ and its transpose, i.e. we have:

$$H_0 = (C_m, C_m^T).$$

Consequently, $H_0$ is a dual-containing code satisfying the ‘even overlap’ constraint, because every overlap that occurs in $C_m$ may also be found in $C_m^T$. Furthermore, since $H_0$ is an $n/2 \times n$ matrix, the resultant dual-containing quantum LDPC code has a coding rate $R_Q = 0$ (or equivalently $R_c = 1/2$). To achieve a non-zero coding rate, $k$ rows of $H_0$ are discarded, so that the column weights of the resultant $(n-k) \times n$ PCM $H'_Q$ are as uniform as possible. This code design offers flexibility in choosing the code parameters, i.e. $\rho$, $n$ and $k$. However, the minimum distance of the resultant code is upper-bounded by $\rho$. This is because the discarded rows of $H_0$ are all codewords of weight $\rho$, which are not contained in the dual, and therefore contribute to the harmful undetected error patterns.

Mackay et al. also proposed unicycle codes in [29], which are derived from perfect difference sets. The perfect difference set property implies that all pairs of rows of the PCM must have a single overlapping 1. Since we need an ‘even overlap’ to achieve a dual-containing structure, the PCM is extended by adding an extra column having all logical ones. Hence, every pair of rows in the resultant PCM have two overlapping 1’s, which result in a single length-4 cycle between every pair of rows. Thus, an $(n, k)$ PCM is transformed into a dual-containing $(n+1, k+1)$ PCM, which has a row weight of $(\rho + 1)$ (where $\rho$ is the row weight of the initial matrix and must be odd) and whose column weights are all $\rho$, except for the last ‘all-one’ column. Mackay et al. also suggested that the unique structure of unicycle codes may be exploited for avoiding the length-4 cycles during the decoding procedure [29]. More explicitly, a unicycle code may be viewed as a superposition of two codes, i.e. one having an ‘all-zero’ column at the end and the other having an ‘all-one’ column. For the sake of avoiding the short cycles, each of the two codes is decoded separately using the sum product algorithm [29]. If both decoders return a valid codeword, the codeword which has the maximum likelihood is chosen. Hence, an improved decoding procedure is conceived at the cost of an increased decoding complexity. Furthermore, the minimum distance of the unicycle codes constructed using difference sets is upper-bounded by the row weight, because the resultant code has codewords of weight $\rho$, which do not lie in the dual. Since the choice of $n$, $k$ and $\rho$ for perfect difference sets is limited, this design does not offer much flexibility in choosing the code parameters. By contrast, bicycle codes can be constructed from any arbitrary cyclic classical LDPC.

To extend the application of Mackay’s unicycle codes to a wider range of code parameters, Aly [40] exploited the classical type-II Euclidean Geometry (EG) LDPC codes of [41]. Similar to the perfect difference sets, a classical type-II EG LDPC code having a PCM $H_{EG-II}$ has the unique characteristic that all pairs of rows have a single overlapping value of 1. Consequently, Aly suggested that the code characterized by

$$\begin{align*}
(1 - 2) \mod 7 &= 6, \\
(1 - 4) \mod 7 &= 4, \\
(2 - 1) \mod 7 &= 1, \\
(2 - 4) \mod 7 &= 5, \\
(4 - 1) \mod 7 &= 3, \\
(4 - 2) \mod 7 &= 2.
\end{align*}$$
an \((n-k) \times n\) matrix \(H_{\text{EG-II}}\) may be converted into a dual-containing code in the following two ways:

1) If the row weight of \(H_{\text{EG-II}}\) is odd, then similar to the unicycle codes, an ‘all-one’ column \(I\) is appended to \(H_{\text{EG-II}}\), i.e. we have:

\[
H'_r = (H_{\text{EG-II}} | I).
\] (17)

2) If the row weight of \(H_{\text{EG-II}}\) is even, then \(I\) is appended to \(H_{\text{EG-II}}\) for the sake of ensuring an ‘even overlap’, while an identity matrix \(I\) of size \((n-k) \times (n-k)\) is appended to make the row weight even, i.e. we have:

\[
H'_z = (H_{\text{EG-II}} | I | I).
\] (18)

The resultant codes offer beneficial high coding rates. However, they have an upper-bounded minimum distance of at least \((\gamma+1)\), where \(\gamma\) denotes the column weight.

Unicycle code construction was further explored by Djordjevic [42] for designing Quasi-Cyclic (QC) high-rate dual-containing QLDPC codes from the BalancedIncomplete Block Design (BIBD) based classical LDPC codes [43], [44], which have a minimum distance of at least \((\gamma+1)\), where \(\gamma\) denotes the column weight. More specifically, the BIBD is characterized by the parameter \(\lambda\). A BIBD-based LDPC code has exactly \(\lambda\) overlaps between every pair of rows. Since good classical LDPC codes must have at most a single row overlap, \(\lambda\) is set to 1 for designing classical LDPC codes with a girth of at least 6. Consequently, analogous to the perfect difference set based classical LDPC codes, each pair of rows has a single overlapping value of \(1\), which can be made even by imposing the unicycle code structure on the PCM. Djordjevic also designed dual-containing LDPC codes by using BIBDs associated with an even \(\lambda\). Unfortunately, the even \(\lambda\) based QLDPC codes failed to outperform the unicycle based BIBD constructions [42].

Since all the aforementioned dual-containing constructions resulted in an upper-bounded minimum distance, the quest for the construction of unbounded QLDPC codes continued. Pursuing this objective, another non-trivial class of dual-containing QLDPC codes was proposed by Mackay et al. in [45], which was derived from Cayley graphs. These codes were further investigated by Couvreur et al. in [46] and [47], where it was formally shown that the lower bound on the minimum distance of the resultant code is a logarithmic function of the code length, thus the minimum distance can be improved by extending the code-word (or block) length, albeit again, only logarithmically.

\[14\] BIBD\((v, b, r, k, \lambda)\) distributes all the \(v\) elements (or points) of a set \(V\) into \(b\) subsets (or blocks) of size \(k\) such that:

- each pair of elements occurs in exactly \(\lambda\) of the blocks,
- every element occurs in exactly \(r\) blocks, and
- the number of elements in each block \(k\) is small as compared to the size \(v\) of set \(V\); thus, giving it the name "incomplete."

Let us consider a set \(V\) of seven numbers, which is given by \(V = \{1, 2, 3, 4, 5, 6, 7\}\). Then, the blocks \(1, 2, 4\), \(2, 6, 5\), \(3, 4, 6\), \(4, 5, 7\), \(1, 5, 6\), \(2, 6, 7\) and \(1, 3, 7\) constitute the BIBD since there are 7 elements \((v)\) in the set \(V\) which are distributed among 7 blocks \((b)\), each element appears in 3 blocks \((r)\), each block has 3 elements \((k)\) and each pair of elements occur in 1 block \((\lambda)\).

However, this is achieved at the cost of an increased decoding complexity imposed by the escalating row weight, which also increases logarithmically with the block length. Furthermore, Cayley graph based designs may be viewed as a special class of the topological codes [48]–[50],\(^{15}\) which are already known to have growing minimum distances.

Let us recall that the dual-containing QLDPC codes have unavoidable short cycles, which impair the performance of the decoding algorithm. Hence, even if dual-containing QLDPC codes having an unbounded minimum distance are designed, they are unlikely to surpass the performance of their non-dual-containing counterparts. Therefore, in the midst of these activities, Lou and Garcia-Frias [55], [56] rekindled the interest in CSS-based non-dual-containing QLDPC codes by invoking the classical Low Density Generator Matrix (LDGM) codes for code construction. More specifically, since both the generator matrix and the PCM of an LDGM code are sparse, they can be used as the components of a CSS code. Let \(\tilde{G}\) and \(\tilde{H}\) be the generator and matrix, respectively, of an \((n, k)\) LDGM code. Then the resultant CSS code may be formulated as follows:

\[
H = \begin{pmatrix}
\tilde{H} & 0 \\
0 & \tilde{G}
\end{pmatrix}.
\] (19)

Since \(\tilde{H}\) is an \((n-k) \times n\) matrix, while \(\tilde{G}\) is a \(k \times n\) matrix, the resultant PCM \(H\) is an \(n \times 2n\) matrix. Consequently, the corresponding QLDPC code has a coding rate of zero. Lou and Garcia-Frias [55], [56] suggested that this may be avoided by applying linear row operations both to \(\tilde{G}\) as well as to \(\tilde{H}\) for the sake of reducing their number of rows. Unfortunately, this row-reduction may in turn create short cycles in the resultant PCM. For the sake of avoiding the adverse impact of these short cycles, Lou and Garcia-Frias [55], [56] also conceived a modified Tanner graph, which requires code doping [57] for pushing the iterative decoding process towards convergence. Hence, an improved performance is achieved at the cost of an increased decoding complexity.

Unfortunately, the constituent codes of all the aforementioned CSS constructions, both those of the dual-containing as well as of the non-dual-containing codes, suffer from the presence of length-4 cycles. To dispense with these short cycles, Hagiwara and Imai [58], [59] conceived a unique class of non-dual-containing QC-QLDPC codes, which have a girth of at least 6. More specifically, let us consider a circulant matrix \(T\) having a size of \(LP/2 \times LP/2\), \(\rho = L/2\) and \(\gamma = L\), which is given by [59]:

\[
T = \begin{pmatrix}
i_0 & i_1 & \cdots & i_{L/2-1} \\
i_{L/2-1} & i_0 & \cdots & i_{L/2-2} \\
\vdots & \vdots & \ddots & \vdots \\
i_1 & i_2 & \cdots & i_0
\end{pmatrix},
\] (20)

\(^{15}\)The family of topological codes, e.g. [48]–[54], is beyond the scope of this work.
where \( t_i \) denotes the index of the circulant permutation matrix\(^{16} \) of size \( P \) and \( t_i \in [P_\infty] := \{0, 1, \ldots, P-1\} \cup \{\infty\}.\) Hagiwara et al. have shown that \( H'_{x} \) and \( H'_{x} \) derived from the matrix \( T \) of Eq. (20) satisfy the symplectic criterion, if they have the form:

\[
H'_{x} = (T_1, T_2) \quad \text{and} \quad H'_{x} = \left( -T_1^T, -T_1^T \right).
\]

Furthermore, since row deletion does not perturb the symplectic criterion, rows may be deleted from \( H'_{x} \) and \( H'_{x} \) in order to achieve the desired coding rate. For the sake of ensuring a girth of 6, Hagiwara et al. relied upon algebraic combinatorics for designing the constituent circulant matrices \( T_1 \) and \( T_2 \), so that all the rows of \( H'_{x} \) as well as of \( H'_{x} \) have at most a single overlap. The binary codes of [29] may be viewed as a special case of this construction, i.e. when \( P = 1 \) and \( T_2 = T_1^T \). Unfortunately, the resultant codes failed to outperform MacKay’s binary codes [29] and their minimum distance is upper-bounded by the row weight.

Among all the dual-containing codes discussed above, Mackay’s binary construction [29] offers the best performance at an affordable decoding complexity. However, the resultant performance is still not on par with that of the classical LDPC codes. For example, the rate-1/4 binary code of [29], having \( n = 19, 014 \), operates within about 5.5 dB of the Hashing limit at a Word Error Rate (WER) of \( 10^{-3} \). Furthermore, all the aforementioned codes have an upper-bounded minimum distance except for the Cayley graph based designs. In the quest for increasing the minimum distance and hence to approach the capacity, Hagiwara et al. extended the QC design of [58] and [59] to Spatially-Coupled (SC) codes in [60], which outperformed their corresponding ‘non-coupled’ counterparts at the cost of a small coding rate loss. However, the performance still remained relatively far from the capacity. More specifically, the SC QC-QLDPC of [60], having a coding rate of 0.49 and a length of \( n = 1, 81, 000 \), operates within about 3.8 dB of the Hashing limit at a WER of \( 10^{-3} \). Kasai et al. further contributed to these developments by deriving non-binary QC-QLDPC codes in [61] and [62] from the design of [58] and [59]. The resultant codes outperformed their binary counterparts at the cost of an increased decoding complexity. A rate-1/2 code, having a length of \( n = 20, 560 \) and a Galois field of \( \text{GF}(2^{10}) \), was shown to operate within about 1.9 dB of the Hashing limit at a WER of \( 10^{-3} \). The SC codes were further investigated by Andriyanova et al.,

\[ I(1) = \begin{pmatrix} 0 & 1 & 0 & \ldots & 0 \\ 0 & 0 & 1 & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \ldots & 0 \\ 1 & 0 & 0 & \ldots & 0 \end{pmatrix}. \]

More explicitly, \( I(1) \) is a \( P \times P \) identity matrix shifted to the right by one position. Therefore, \( I(x) \) may be defined as a \( P \times P \) identity matrix shifted to the right by \( x \) positions, where \( x \) is known as the index of the permutation matrix. Moreover, \( x = 0 \) defines an unshifted identity matrix, while \( x = \infty \) is specially used to denote a zero matrix of size \( P \times P \).

in [63], where the constituent codes were derived from the classical LDGM codes as in [55] and [56]. Analogous to the EA quantum codes, Andriyanova et al. assumed that some qubits are transmitted over a noiseless channel. Consequently, the resultant rate-1/4 LDGM-based SC-QLDPC codes, having a length of \( n = 76, 800 \), succeeded in operating within about 1.7 dB of the Hashing limit at a WER of \( 10^{-3} \). The assumption of having noiseless qubits was later eliminated in [64], whereby these qubits were protected by the error reducing Quantum Turbo Code (QTC) of [64], which resulted in a modest coding rate loss and in a moderately increased complexity for the overall code. It was shown that the performance of the resultant rate-1/2 QTC-assisted LDGM-based SC-QLDPC code, having a length of \( n = 8, 21, 760 \), is within about 0.7 dB of the Hashing limit at a WER of \( 10^{-3} \). Fig. 7 compares the achievable performance of the aforementioned codes, namely of the ‘bicycle’ code of [29], ‘SC QC-QLDPC’ code of [60], ‘non-binary QC-QLDPC’ code of [61] and [62], ‘LDGM-based SC-QLDPC’ code of [63] and the ‘QTC-assisted LDGM-based SC-QLDPC’ code of [64], at a WER of \( 10^{-3} \), which is benchmarked against the Hashing bound.

All the main contributions pertaining to CSS-based QLDPD codes are summarized in Fig. 8.

---

\(^{16}\) A circulant permutation matrix \( I(1) \) of size \( P \) is given by:
Postol [39] conceived the prototype of a non-dual-containing QLDPC code from a small finite geometry based classical LDPC codes. No generalized code construction formalism was proposed.

Mackay et al. [29] proposed a generalized construction referred to as 'bicycle' for designing dual-containing QLDPC codes from any arbitrary cyclic classical LDPC code. The minimum distance was upper-bounded by the row weight and an improved decoding algorithm was required for tackling length-4 cycles.

Mackay et al. [29] introduced a generalized dual-containing code structure termed as 'unicycle' derived from perfect difference sets and developed an improved decoding algorithm for unicycle codes to overcome the problems of length-4 cycles. The minimum distance was upper-bounded by the row weight, the range of possible code parameters remained limited and the associated decoding complexity was increased.

Loa et al. [55], [56] exploited the generator and PCM of classical LDGM codes for constructing non-dual-containing QLDPC codes and invoked code doping based improved decoding. The minimum distance was upper-bounded and the decoding complexity was increased.

Camara et al. [65], [66] exploited a group-theoretical approach for constructing self-orthogonal quaternary PCMs for non-CSS QLDPC codes. Failed to outperform Mackay’s codes of [29].

Hagiwara et al. [58] constructed non-dual-containing QC-QLDPC codes having a girth of 6 using a pair of classical QC-LDPC codes, which was found with the aid of algebraic combinatorics. The minimum distance was upper-bounded by the row weight and the proposed code failed to outperform Mackay's bicycle codes [29].

Aly et al. [40] constructed dual-containing QLPCs from finite geometry based classical LDPCs by exploiting the unicycle code design. The minimum distance was upper-bounded and the decoding complexity was increased.

Djordjevic [42] derived dual-containing QLPCs from even index BIBDs as well as BIBD-based unicycle codes. The minimum distance was upper-bounded. Even index BIBD codes failed to outperform the BIBD-based unicycle codes.

Hsieh et al. [67] conceived the first EA QC-QLDPC codes, which outperformed their un-assisted counterparts. Despite their efforts to minimize the number of ebits, significant fractions of ebits were required, which grew with the code length.

Hsieh et al. [68], [69] proposed finite geometry based EA-QLDPCs. Two of the proposed constructions required only a single ebit, while the entanglement consumption rate was a decreasing function of the code length for the remaining designs.

Tan et al. [70] conceived several systematic non-CSS constructions by exploiting simple yet powerful coding techniques, e.g., concatenation, rotation and scrambling. Failed to outperform Mackay's codes of [29].

Djordjevic [71] introduced BIBD based EA-QLDPC codes requiring only a single ebit.

Fujiiwara et al. [72] conceived a general framework for designing EA-QLDPCs having a prescribed number of ebits. Some designs required only a single ebit.

Couveur et al. [46], [47] further investigated the Cayley graph based dual-containing QLDPC codes of [45] to resolve the problem of upper-bounded minimum distance. The row weight increased logarithmically with the block length, imposing an increased decoding complexity.

Kasai [61], [62] extended the non-dual-containing QC-QLDPC codes of [58] to non-binary constructions. The minimum distance was upper-bounded and the decoding complexity was increased.

Hagiwara et al. [60] proposed spatially-coupled non-dual-containing QC-QLDPC codes, which outperformed the 'non-coupled' design of [58] at the cost of a small coding rate loss. The minimum distance was upper-bounded.

Andryianova et al. [63] derived spatially coupled non-dual-containing QLDPCs from LDGM-based codes of [55], [56], resulting in a performance close to the Hashing limit. Noiseless transmission of some qubits was assumed and the minimum distance was upper-bounded.

Fujiiwara et al. [73] further investigated EA-QLDPCs requiring a single ebit.

Maurice et al. [64] improved the non-dual-containing design of [63] by protecting the noiseless qubits using the error reducing turbo code of [64]. The performance was arbitrarily close to the Hashing limit at the cost of a small coding rate loss. The minimum distance was upper-bounded and the encoding/decoding complexity was increased.

Fujiiwara et al. [74] conceived EA-QLDPC codes relying on 'less noisy' qubits, which assume a phase-flip channel model for the ebits.

FIGURE 8. Major contributions to the development of QLDPC codes. The 'code type' for each contribution is highlighted in bold, while the associated 'demerits' are marked in italics.
structure, the design conceived by Camara et al. aims at building the symplectic constraint into the local code structure. More specifically, since the PCM of a classical quaternary LDPC code can be mapped onto the generators of a QSC based on Eq. (9), Camara et al. developed a group theoretical approach for constructing self-orthogonal quaternary LDPC codes satisfying the symplectic criterion of Eq. (11). It was found that the Tanner graph of the resultant self-orthogonal quaternary PCM has cycles of length 4. However, these short cycles are imposed by the commutativity constraint. More specifically, every column of a quaternary PCM must contain at least two different non-zero entries, i.e., Pauli-$X$, Pauli-$Z$, or Pauli-$Y$, so that it can correct both phase-flips as well as bit-flips occurring on that qubit. On the other hand, any two rows of the PCM must have an even number of positions with different non-zero elements (or non-Identity Pauli operators). For example, let us consider a weight-2 column of a PCM, which is involved in two rows with a value of 1 and $\omega$, respectively. Now to meet the commutativity constraint, these two rows must have another overlapping column having different non-zero entries; thus, creating cycles of length-4. Intuitively, these short cycles are also present in the PCM $H$ of the CSS codes, when they are viewed in the quaternary domain. In fact, these cycles are excessive in the dual-containing CSS codes, which also have the additional cycles resulting from the dual-containing constraint.¹⁷ The proposed non-CSS QLDPD codes of [65] and [66] outperformed the bicycle codes in the waterfall region of their performance curve, while yielding a higher error floor due to their small minimum distance. It is expected that this non-CSS construction may have an unbounded minimum distance, thus yielding lower error floors, when the block length is sufficiently large. However, this was not explicitly proven in [65] and [66].

Pursuing the same line of research, Tan and Li [70] were the first researchers to design the constituent PMs $H_x$ and $H_z$ of a non-CSS code by invoking classical binary codes. More specifically, they conceived several systematic constructions for non-CSS QLDPD codes, which imposed both global as well as local structures on the underlying binary codes for the sake of satisfying the symplectic criterion. This is achieved by exploiting simple yet powerful coding techniques, which include concatenation, rotation and scrambling. The designed codes exhibit a better performance than the non-CSS codes of [65] and [66]. However, they still failed to outperform Mackay’s codes of [29]. In conclusion, the major milestones achieved in the domain of non-CSS QLDPD codes are summarized in Fig. 8.

C. ENTANGLEMENT-ASSISTED QLDPD CODES

Efficient classical LDPC codes exist, which are known to approach the Shannon capacity for a large block size. For example, the optimized 1/2-rate classical LDPC code of [76] operates within 0.13 dB of the capacity limit for transmission over an Additive White Gaussian Noise (AWGN) channel at a Bit Error Rate (BER) of $10^{-6}$ using a code length of $10^6$. More specifically, the turbo cliff of this LDPC code is merely 0.06 dB away from the Shannon capacity. This inspired researchers to achieve a comparable performance for QLDPD codes. Unfortunately, the symplectic criterion, or more specifically the commutativity requirement of the stabilizers, limits the direct application of such efficient classical codes in the quantum domain. As discussed in Sections III-A and III-B, only a limited class of classical codes, which conform to stringent local or global structural constraints, may be used as the constituents of a quantum code. This obstacle may be overcome by exploiting the EA quantum code designs of [35]–[37], which assist us in importing any classical code into the quantum domain. However, the pre-shared noiseless entangled qubits (ebits) of an EA code constitute a valuable resource, because maintaining a noiseless entangled state is not a trivial task. Consequently, a practically realizable code design should aim for minimizing the number of pre-shared noiseless ebits.

The first EA-QLDPC codes were conceived by Hsieh et al. in [67], whereby EA CSS-based QC-QLDPD codes were designed from their classical counterparts. Hsieh et al. chose the constituent circulant matrices of the classical QC code by ensuring that the number of ebits required is minimized. Despite their efforts, a significant number of these ebits was required, which grew with the code length. More importantly, these designs supported the conjecture that the high efficiency of EA codes should be attributed to the large fractions of pre-shared ebits. On a positive note, since the EA quantum codes of [67] shared the same attributes as the classical parent code, especially in terms of the girth and the minimum distance, these EA-QLDPC codes outperformed the state-of-the-art unassisted QLDPD codes. Working further in the direction of minimizing the number of pre-shared ebits, in [68] and [69], Hsieh et al. conceived finite-geometry based EA-QLDPCs, whose ‘entanglement consumption rate’ decreases with the code length. Furthermore, two of these constructions required only a single ebit regardless of the code length; thus dispensing with the then prevailing apprehensions surrounding the family of EA-codes. It must be emphasized here that the proposed design does not impose any restrictions on the underlying finite geometry based classical LDPC codes of [41]. A more general framework conceived for designing EA-QLDPCs, having a prescribed number of ebits, was presented in [72], which was derived from combinatorial design theory. Some of these designs required only a single ebit, despite having a high performance, a high coding rate and a low complexity. The necessary and sufficient conditions for designing single-ebit based EA-QLDPD codes were further investigated in [73]. Moreover, BIBD based EA-QLDPC codes requiring only a single ebit were also identified in [71]. Recently, Fujiwara [77] introduced the notion of quantum codes relying on ‘less noisy’ (or ‘reliable’) qubits. More explicitly, unlike the EA formalism, which requires completely noiseless ebits, the framework of [77] assumes that these auxiliary qubits are subjected to a phase-flip channel.

¹⁷This is further discussed in Section IV-C.
which is a more realistic noise model. In this spirit, Fujiwara et al. [74] conceived QLDPC codes relying on ‘less noisy’ qubits. The major contributions made in the domain of EA-QLDPC codes are discussed in Sections IV-A and IV-B, respectively.

IV. ITERATIVE DECODING OF QUANTUM LDPC CODES

Analogous to the classical LDPC codes, QLDPC codes invoke the classic Belief Propagation (BP) based decoding, also referred to as the Sum-Product Algorithm (SPA), which operates over the Tanner graph of the corresponding PCM. However, let us recall from Section II that qubits collapse upon measurement. Therefore, the syndrome-based version [78] of the classic codeword decoding has to be used for QLDPC codes. The underlying BP can be implemented both in the binary as well as in the quaternary domain, which are discussed in Sections IV-A and IV-B, respectively.

A. BINARY DECODING

A quantum depolarizing channel characterized by the depolarizing probability \( p \) is isomorphic to two independent Binary Symmetric Channels (BSCs) [29], i.e. one for phase-flips and the other for bit-flips, each having a cross-over probability of \( 2p/3 \). More explicitly, based on the Pauli-to-binary isomorphism encapsulated in Eq. (4), a Pauli error \( \mathcal{P} \in \mathcal{G}_n \) experienced by an \( n \)-qubit block transmitted over a depolarizing channel can be modeled by an effective error-vector \( P \), which is a binary vector of length \( 2n \). The effective error \( P \) may be represented as \( P = (P_x, P_z) \), where both \( P_x \) and \( P_z \) are \( n \)-bit long and represent \( \mathbf{Z} \) and \( \mathbf{X} \) errors, respectively. This implies that an \( \mathbf{X} \) error imposed on the \( r \)-th qubit will yield a 0 and a 1 at the \( r \)-th and \( (n + t) \)-th index of \( P \), respectively. Similarly, a \( \mathbf{Z} \) error imposed on the \( r \)-th qubit will give a 1 and a 0 at the \( r \)-th and \( (n + t) \)-th index of \( P \), respectively. While a \( \mathbf{Y} \) error on the \( r \)-th qubit will result in a 1 at both the 1st as well as \( (n + r) \)-th index of \( P \). Since a depolarizing channel characterized by the probability \( p \) incurs \( \mathbf{X}, \mathbf{Y} \) and \( \mathbf{Z} \) errors with an equal probability of \( p/3 \), the effective error-vector \( P \) reduces to two BSCs having a crossover probability of \( 2p/3 \), where we have one channel for the \( \mathbf{Z} \) errors and the other for the \( \mathbf{X} \) errors.

Based on the aforementioned simplified notion, which ignores the correlation between the \( \mathbf{X} \) and \( \mathbf{Z} \) errors, QLDPC codes can be decoded by running the syndrome-based BP over the Tanner graph of the equivalent binary code having \( H = (H_x, H_z) \) [70]. More explicitly, let \( S \) be the observed syndrome sequence, which is given by the symplectic product of \( H \) and \( P \), as formulated below:

\[
S = H \cdot P^T = H_x P_x^T + H_z P_z^T. \tag{22}
\]

The observed syndrome \( S \) of Eq. (22) is fed to a classical syndrome-based LDPC decoder to estimate the most likely inflicted channel error \( \tilde{P} \), as depicted in Fig. 9. For an \( H \) of size \( m \times 2n \), where we have \( m = (n - k) \), the resultant estimated error vector \( \tilde{P} \) is of length \( 2n \), whose first \( n \) bits are for the estimated phase errors \( \tilde{P}_x \), while the other \( n \) bits indicate the estimated bit errors \( \tilde{P}_z \). Finally, the \( 2n \)-bit binary vector is mapped onto the \( n \)-qubit Pauli error \( \tilde{P} \) based on the mapping encapsulated in Eq. (4). More explicitly, the \( r \)-th and \((n + t)\)-th value of \( \tilde{P} \) are combined based on Eq. (4) to estimate the error inflicted on the \( r \)-th qubit.

For CSS codes, we have \( H_z = \begin{pmatrix} H'_z \end{pmatrix} \) and \( H_x = \begin{pmatrix} 0 \\ H'_x \end{pmatrix} \).

Consequently, the Tanner graph of the matrix \( H \) consists of two independent Tanner graphs corresponding to the matrices \( H'_x \) and \( H'_z \). This in turn implies that \( \mathbf{X} \) and \( \mathbf{Z} \) errors can be decoded independently using the matrices \( H'_x \) and \( H'_z \), respectively [29]. Hence, the Qubit Error Rate (QBER) of a CSS QLDPC code may be approximated by the sum of the BER of the constituent classical codes. More explicitly, if \( p'_{ex} \) and \( p'_{ez} \) are the classical BERs for \( H'_x \) and \( H'_z \), respectively, then the overall QBER is equivalent to \( (p'_{ex} + p'_{ez} - p'_{ex} p'_{ez}) \approx (p'_{ex} + p'_{ez}) \), which reduces to \( 2p'_{ez} \) for a dual-containing CSS code having \( H'_z = H'_x^* \).

For a binary \( m \times 2n \) LDPC matrix \( H \), the classical LDPC decoder of Fig. 9 aims for finding the most likely error \( P \) of length \( 2n \) given the observed syndrome \( S \), i.e. we have:

\[
\tilde{P} = \arg \max_{P \in \mathbb{F}_2^{2n}} P(S|P), \tag{23}
\]

where \( P(S|P) \) is the probability of experiencing the error \( P \in \mathbb{F}_2^{2n} \) imposed on the transmitted codewords, given that the syndrome of the received qubits \( |\psi\rangle \) is \( S \in \mathbb{F}_2^m \). Unfortunately, Eq. (23) defines an NP-complete problem [79]. A sub-optimal algorithm for solving Eq. (23) is constituted by the classic BP, which finds the element-wise optimum value rather than the global optimum. More explicitly, for \( P = (P_0, P_1, \ldots , P_{2n-1}) \), BP finds \( P_t \) such that:

\[
\tilde{P}_t = \arg \max_{P_t \in \mathbb{F}_2} P_t(S|P_t), \tag{24}
\]
where \( P(P_t|S) \) is the marginalized probability of the \( t \)th bit. The BP operates by exchanging messages over the Tanner graph of \( H \) having check nodes \( c_i \) for \( i \in \{0, m-1\} \) and variable nodes \( v_t \) for \( t \in \{0,2n-1\} \). The messages sent by the \( t \)th check node \( c_i \) to the \( t \)th variable node are denoted by \( m_{c_i}^{a} \rightarrow v_{t} \), while the messages directed from the \( t \)th variable node to the \( t \)th check node are given by \( m_{v_t}^{a} \rightarrow c_i \), where \( P_t \) is the error imposed on the \( t \)th variable node.

The overall syndrome-based message exchange procedure is summarized in Algorithm 1, which proceeds as follows [78]:

- **Initialization:** The algorithm begins by initializing the messages \( m_{v_t}^{0} \rightarrow c_i \) according to the channel model \( P_{ch}(P_t) \). For a BSC having a crossover probability of 2\( p/3 \), we have:

\[
\begin{align*}
m_{v_t}^{0} \rightarrow c_i &= 1 - 2p/3, \\
m_{v_t}^{1} \rightarrow c_i &= 2p/3.
\end{align*}
\]  

(25)

- **Horizontal message exchange:** Let \( V(c_i) \) be the set of variable nodes connected to the check node \( c_i \), i.e. \( V(c_i) \equiv \{ v_t : H_{c_t} = 1 \} \), and \( V(c_i) \setminus v_t \) be the set \( V(c_i) \) excluding the variable node \( v_t \). As depicted in Fig. 10a, in this step the algorithm runs through the rows of \( H \) (checks) and computes the message \( m_{c_i}^{a} \rightarrow v_{t} \) for each \( v_t \in V(c_i) \) and \( P_t \in \mathbb{F}_2 \). The message \( m_{c_i}^{a} \rightarrow v_{t} \) represents the probability that the syndrome value observed for the check \( c_i \) is \( S_{t} \) given that the \( t \)th variable node has the error \((P_t = a)\), where \( a \in \{0,1\} \). This can be mathematically formulated as:

\[
m_{c_i}^{a} \rightarrow v_{t} = K \sum_{P \in P_{ch}(P_t = a)} P(S_{t}|P) \prod_{v_j \in V(c_i) \setminus v_t} m_{v_j}^{P} \rightarrow c_i,
\]  

(26)

where \( K \) is the normalization constant invoked for ensuring \( \sum_{a \in \{0,1\}} m_{c_i}^{a} \rightarrow v_{t} = 1 \), while \( P(S_{t}|P) \) is a binary function, which is equal to 1 only when the check \( c_i \) is satisfied, i.e. when the value of the check node \( c_i \) computed using the error vector \( P \) matches the measured syndrome value \( S_{t} \), otherwise it is 0. Furthermore, according to Eq. (26), the messages \( m_{c_i}^{a} \rightarrow v_{t} \) destined for the \( t \)th variable node do not take into account the messages flowing in the opposite direction along the same edge, i.e. \( m_{v_t}^{P} \rightarrow c_i \). Consequently, \( m_{c_i}^{a} \rightarrow v_{t} \) only contains the new information gleaned from the messages sent by the other variable nodes and it is therefore termed as ‘extrinsic’. This ensures that the successive iterations of this iterative algorithm are independent.

- **Vertical message exchange:** Let \( C(v_t) \) be the set of check nodes connected to the variable node \( v_t \), i.e. \( C(v_t) \equiv \{ c_i : H_{c_t} = 1 \} \), and \( C(v_t) \setminus c_i \) be the set \( C(v_t) \) excluding the check node \( c_i \). As shown in Fig. 10b, for each column of \( H \) (hence called ‘vertical’), the BP computes the message \( m_{v_t}^{P} \rightarrow c_i \) for all \( c_i \in V(v_t) \) and \( P_t \in \mathbb{F}_2 \). More explicitly, the messages \( m_{v_t}^{P} \rightarrow c_i \) are computed by evaluating the product of the channel information \( P_{ch}(P_t = a) \) and the messages \( m_{c_i}^{a} \rightarrow v_{t} \) flowing into the variable node \( v_t \) along all the edges connected to it, but excluding \( m_{c_i}^{a} \rightarrow v_{t} \), which is received along the same edge. Hence, the extrinsic message is computed as:

\[
m_{v_t}^{a} \rightarrow c_i = K P_{ch}(P_t = a) \prod_{c_j \in C(v_t) \setminus c_i} m_{c_j}^{a} \rightarrow v_{t},
\]  

(27)

where \( k \) is the normalization constant, which ensures that \( \sum_{a \in \{0,1\}} m_{v_t}^{a} \rightarrow c_i = 1 \).

- **Element-wise marginal probability:** Finally, the element-wise marginal probability \( P(P_t|S) \) for \( P_t \in \mathbb{F}_2 \) is calculated as follows:

\[
P(P_t = a|S) = KP_{ch}(P_t = a) \prod_{c_i \in C(v_t)} m_{c_i}^{a} \rightarrow v_{t},
\]  

(28)

which takes into account all the messages flowing into the variable node \( v_t \).

- **Hard decision & syndrome check:** As previously portrayed in Eq. (24), a hard decision is made by finding the most likely error \( \hat{P}_t \), which maximizes the marginal probability computed in Eq. (28). Based on the estimated error vector \( \hat{P} \), the syndrome \( \hat{S} = H(\hat{P}_x : \hat{P}_z)^T \) is computed. If the syndrome \( \hat{S} \) of the estimated error \( \hat{P} \) is the same as the observed syndrome \( S \), the process halts, indicating that the correct solution is found. Otherwise, the algorithm repeats itself from the horizontal message exchange step onwards. This iterative procedure continues, until either \( \hat{S} = S \) or the maximum number of iterations \( I_{\text{max}} \) is reached.

**B. NON-BINARY DECODING**

Based on the Pauli-to-GF(4) formalism of Eq. (9), QLDPC codes can be decoded by invoking the non-binary BP, which takes into account the correlation between the phase-flips and bit-flips. The syndrome-based non-binary BP is similar to the binary BP of Algorithm 1, with the following two major modifications:

- Non-binary BP exploits the depolarizing channel model, which does not ignore the correlation between the bit and
Algorithm 1 Syndrome-Based BP

1: Set \( P_{ch}(0) \leftarrow (1 - 2p/3) \) and \( P_{ch}(1) \leftarrow 2p/3 \).
2: Initialize \( m_{ci}^{a} \rightarrow c_i \) \( P_{ch}(a), \forall v_i, c_i \in C(\gamma_i) \) and \( a \in \{0, 1\} \).
3: for iter \( \leftarrow 0 \) to \( \max \) do
4:   for all \( i \in \{0, (m - 1)\}, v_i \in V(\gamma_i) \) and \( a \in \{0, 1\} \) do
5:      \( m_{ci}^{a} \rightarrow v_i \leftarrow k \sum_{P_i = a} P(S)\prod_{v_i' \in V(\gamma_i) / v_i} m_{ci}^{a} \rightarrow v_i' \).
6:   end for
7: for \( t \leftarrow 0 \) to \((2n - 1)\) do
8:   for all \( c_i \in C(\gamma_i) \) and \( a \in \{0, 1\} \) do
9:      \( m_{ci}^{a} \rightarrow \leftarrow kP_{ch}(P_i = a) \prod_{c_i' \in C(\gamma_i) / c_i} m_{ci}^{a} \rightarrow v_i' \).
10: end for
11: for all \( a \in \{0, 1\} \) do
12:      \( P_i \leftarrow \arg\max P(P_i | S) \).
13: end for
14: end for
15: end for
16: \( \hat{S} \leftarrow H(\hat{P}_s : P_i)^T \).
17: if \( \hat{S} = S \) then
18:   return \( \hat{P} \).
19: end if
20: end for

Phase errors. The equivalent 4-ary channel model has the following probability distribution:
\[
P_{ch}(\hat{P} = \hat{a}) = \begin{cases} 1 - p, & \text{if } \hat{a} = 0 \\ p/3, & \text{if } \hat{a} \in \{1, \omega, \overline{\omega}\}, \end{cases} \tag{29}
\]
where we have \( \hat{P} = (\hat{P}_0, \hat{P}_1, \ldots, \hat{P}_i, \ldots, \hat{P}_{n-1}) \) and \( \hat{P}_i \) denotes the error inflicted on the \( i \)th qubit.

The syndrome \( S_i \), which was computed as \( H(\hat{P}_s : P_i)^T \) in the binary scenario, is now given by the trace inner product of \( \hat{H} \) and \( \hat{P} \) (see Eq. (11)):
\[
S_i = Tr(\hat{H}_i \cdot \hat{P}), \tag{30}
\]
where \( \hat{H}_i \) is the \( i \)th row of \( H \) in GF(4) and \( i \in \{0, m - 1\} \).

As compared to the binary BP, non-binary decoding imposes an increased complexity, specifically on the horizontal message exchange step. More explicitly, since the summation in Eq. (26) runs for all possible error sequences \( \{\hat{P} : \hat{P}_i = \hat{a}\} \), which yield the syndrome \( S_i \) for the \( i \)th check node, the complexity increases both with the row weight as well as with the dimensionality of the Galois field. For classical non-binary LDPC codes, this increased complexity is alleviated by invoking the Fast Fourier Transform (FFT) based decoding of [80], which can be conveniently adapted to the syndrome-based decoding of QLDPC codes.

Based on the notion of the trace inner product of Eq. (11), Eq. (30) can be expanded as:
\[
S_i = Tr(\hat{S}_i) = Tr \left( \sum_{t \in V(\gamma_i)} \hat{H}_t \times \hat{P}_i \right), \tag{31}
\]
where we have \( \hat{S}_i \in \{0, 1, \omega, \overline{\omega}\} \), which can also be expressed as:
\[
\hat{S}_i = \hat{H}_t \times \hat{P}_i + \sum_{t' \in V(\gamma_i) / \gamma_i} \hat{H}_t' \times \hat{P}_{i'}'. \tag{32}
\]
Unlike in the binary scenario, where we have \( H_t \in \{0, 1\} \), here we have \( \hat{H}_t \in \{1, \omega, \overline{\omega}\} \) in Eq. (32). Therefore, given the messages \( m_{ci}^{a} \rightarrow v_i \) and \( m_{ci}^{a} \rightarrow \overline{c}_i \) exchanged between the check node \( c_i \) and the variable node \( v_i \) for \( \hat{P} = \hat{a} \), we denote the equivalent messages for \( (\hat{H}_t \times \hat{P}_i) \) as \( \hat{m}_{ci}^{a} \rightarrow v_i \) and \( \hat{m}_{ci}^{a} \rightarrow \overline{c}_i \), respectively, where we have \( (\hat{H}_t \times \hat{a}) = \hat{a} \cdot \hat{a} \). Based on this notation, we may infer from Eq. (31) and Eq. (32) that the Probability Density Function (PDF) of the horizontal message \( \hat{m}_{ci}^{a} \rightarrow v_i \) can be obtained by convolving the PDFs of the messages \( \hat{m}_{ci}^{a} \rightarrow v_i \) for \( v_i \in V(\gamma_i) \). We may further notice in Eq. (31) that for a given \( S_i, \hat{S}_i \) can have two possible values. More explicitly, for GF(4), we have \( Tr(0) = Tr(1) = 0 \), while \( Tr(\omega) = Tr(\overline{\omega}) = 1 \). Consequently, for \( S_i = Tr(\hat{S}_i) = 0 \) = \( Tr(\hat{S}_i) = 1 \), we have:
\[
PDF[\hat{m}_{ci}^{a} \rightarrow v_i] = PDF[\hat{m}_{ci}^{1} \rightarrow v_i] = \frac{1}{2} \left( \otimes PDF[\hat{m}_{ci}^{0} \rightarrow v_i] + \otimes PDF[\hat{m}_{ci}^{1} \rightarrow v_i] \right). \tag{33}
\]
where \( \otimes \) represents the convolution process and \( v_i \in V(\gamma_i) \). Similarly, for \( S_i = Tr(\hat{S}_i) = \omega = Tr(\overline{\omega}) = 1 \), we have:
\[
PDF[\hat{m}_{ci}^{a} \rightarrow v_i] = PDF[\hat{m}_{ci}^{\omega} \rightarrow v_i] = \frac{1}{2} \left( \otimes PDF[\hat{m}_{ci}^{0} \rightarrow v_i] + \otimes PDF[\hat{m}_{ci}^{\omega} \rightarrow v_i] \right). \tag{34}
\]
The complex convolution operation required in Eq. (33) and (34) can be efficiently implemented by multiplying the corresponding PDFs in the frequency domain with the aid of the FFT-based algorithm of [80].

C. DECODING ISSUES & HEURISTIC METHODS FOR IMPROVEMENT

Belief propagation invoked for decoding LDPC codes gives the exact solution only when the underlying Tanner graph is
a tree. Nonetheless, it yields reasonably good approximations even in the presence of cycles, provided that the girth of the associated LDPC matrix is sufficiently large, at least 6. This has been proven by the capacity approaching classical LDPC codes, for example in [16] and [17]. Unfortunately, short cycles of length 4 are unavoidable in the construction of QLDPC codes, which in turn impair the iterative decoding procedure.

The unavoidable cycles of length 4 found in QLDPC codes are the result of the commutativity property of the stabilizers. More explicitly, the constituent stabilizer generators of a stabilizer code must commute, i.e. they should have even number of places with different non-Identity Pauli operators. In other words, if an anti-commuting pair of Pauli operators acts on the \( r \)th variable node in a pair of stabilizer generators, then there should be another anti-commuting pair of Pauli operators acting on the \( r \)th variable node in the same pair of generators for the sake of ensuring that the generators commute with each other. For example, the generators:

\[
g_0 = XIYZ, \\
g_1 = ZYXI, 
\]

commute\(^{18}\) because there are two pairs of anti-commuting Pauli operators acting on the first and third qubits, respectively. This in turn implies that the corresponding rows in the resultant PCM have even number of overlaps, which give rise to short cycles in the Tanner graph, as illustrated in Fig. 11. Since here the key point is to have “different non-Identity operators”, a possible option could be to assign only a single type of non-Identity operator to each variable node of the Tanner graph. If we only assign Pauli-\(X\) to the variable node \( v_i \) so that it does not anti-commute in any pair of generators, then we will be unable to detect both Pauli-\(X\) as well as Pauli-\(Y\) errors acting on \( v_i \). This would yield an undesirable code, which has a minimum distance of one. We may conclude that:

1) each column of a QLDPC matrix must have at least two different non-Identity Pauli operators, and
2) every pair of rows must have even number of places with different non-Identity Pauli operators.

Consequently, all CSS as well as non-CSS QLDPC constructions have a Tanner graph of girth-4. It is interesting to observe here that these short cycles may be avoided in the corresponding binary formalism. Let us consider the example given in Eq. (35), which can be expressed in the binary form as follows:

\[
g_0 \rightarrow (1 \ 0 \ 1 \ 0 \ | \ 0 \ 0 \ 1 \ 1), \\
g_1 \rightarrow (0 \ 1 \ 1 \ 0 \ | \ 1 \ 1 \ 0 \ 0). 
\]

Since these binary\(^{18}\) generators only have a single overlapping 1, the length 4 cycle no longer exits. However, let us recall from Section IV-B that binary decoding ignores the correlation between the \( X \) and \( Z \) errors, which degrades the performance. Hence, a compromise must be struck between these two conflicting aspects.

\[
\hat{H} = \begin{pmatrix} \omega H_z^1 \\ \omega H_z^2 \end{pmatrix}. 
\]

Consequently, the \( i \)th and \( (i + m/2) \)th rows completely overlap, resulting in numerous cycles of length 4. Furthermore, the dual-containing code construction also has the additional short cycles within the matrix \( H_z^i \), as discussed in Section III-A, which exist even in the binary formalism. Table 4 summarizes the presence of unavoidable short cycles in various code structures, while Fig. 12 captures the merits and demerits of GF(4) decoding as compared to its binary counterpart.

---

\(^{18}\)This is just a random example to illustrate the concept of commutativity and the resulting short cycles. The generators \( g_0 \) and \( g_1 \) of this example may not constitute a good stabilizer code.
More explicitly, rather than finding the most likely error, as in Eq. (23), the decoding algorithm should find the most likely error coset by summing the probabilities of all degenerate errors [81], [82]. Furthermore, QLDPC codes are highly degenerate as compared to the other families of quantum codes. This is because the generators of a QLDPC code are sparse in nature. Consequently, it has many low-weight degenerate errors, which dominate the probability of the error coset. It is therefore more likely that the most probable error \( \tilde{P} \) of Eq. (23) may not coincide with the most probable error coset for QLDPC codes. However, rather than exploiting the benefits of high degeneracy associated with sparse codes, the marginalized iterative decoding invoked for QLDPC codes is impaired by degeneracy [81], [82]. This is because degenerate errors of equal weight have the same marginalized probability distribution, which can be attributed to the symmetry of the probability distribution of the channel depicted in Eq. (29).

Let us review the case study given in [81]. Consider a 2-qubit stabilizer code having the generators \( XX \) and \( ZZ \). Assume furthermore that \( IX \) is the channel error encountered during transmission over a depolarizing channel, whose PDF is given in Eq. (29). The resultant syndrome is \( S = (01) \) and the corresponding set of degenerate errors is \([XI, IX, YZ, ZY]\). Consequently, the marginalized conditional probability of the error on each of the two qubits is given by:

\[
P(\tilde{P}_i = a | S) = \begin{cases} 1 - p, & \text{if } a = 0 \\ p/3, & \text{if } a \in \{1, \omega, \bar{\omega}\}, \end{cases} \tag{38}
\]

where \( t = \{0, 1\} \). Hence, the marginalized probability is identical for both the qubits. This symmetry forces the decoder to detect the same error on both the qubits. However, none of the associated errors, i.e. \([XI, IX, YZ, ZY]\), exhibit this symmetry, hence leading to the ‘symmetric degeneracy error’ concept of [81]. Moreover, since the channel profile of Eq. (29) is biased towards the Identity operator, the probability of ‘no-error’ dominates at low noise levels.

Poulin and Chung investigated various heuristic methods in [81] to break the symmetry exhibited by the marginalized probabilities of Eq. (38). Among the investigated methods, “random perturbation” provides the best performance. It aims for breaking the degenerate symmetry by randomly perturbing the channel PDF of Eq. (29) for the qubits involved in the frustrated checks,\(^\text{19}\), thus putting an end to the decoding impasse. Random perturbation begins with the standard non-binary BP, which gives the estimated channel error \( \hat{P} \). If the syndrome computed for \( \hat{P} \) is not the same as the observed channel syndrome \( S \), the channel probabilities of all variable nodes \( v_i \) connected to a randomly chosen frustrated check \( c_i \) are perturbed (up to a normalization) as follows:

\[
P_{ch}(\hat{P}_t = 0) \rightarrow P_{ch}(\hat{P}_t = 0),
\]
\[
P_{ch}(\hat{P}_t = 1) \rightarrow (1 + \delta_1)P_{ch}(\hat{P}_t = 1),
\]
\[
P_{ch}(\hat{P}_t = \omega) \rightarrow (1 + \delta_0)P_{ch}(\hat{P}_t = \omega),
\]
\[
P_{ch}(\hat{P}_t = 0) \rightarrow (1 + \delta_{\bar{\omega}})P_{ch}(\hat{P}_t = 0), \tag{39}
\]

where \( \delta_1, \delta_0, \delta_{\bar{\omega}} \) are random variables in the range \([0, \delta]\).

Non-binary BP is re-run with these modified channel probabilities for \( T_{\text{pert}} \) iterations and \( \hat{P} \) is estimated again. If all the check nodes are satisfied now, the process terminates. Otherwise, the channel probabilities perturbed in Eq. (39) are restored and the process is repeated with another randomly chosen frustrated check.

Another heuristic method of alleviating the symmetric degeneracy problem was conceived in [83], which exploits an enhanced feedback procedure. More specifically, Wang et al. [83] proposed an “enhanced feedback” strategy for perturbing the channel probabilities similar to the random perturbation, but this perturbation is based both on the stabilizer generators involved in the frustrated checks as well as on the channel model. Similar to the random perturbation method, the enhanced feedback algorithm randomly selects a frustrated check \( c_i \). It also selects a variable node \( v_i \) connected to \( c_i \). Let \( S_i \) be the value of the \( i \)th check node for the estimated error \( \hat{P} \), while \( S_i \) be the \( i \)th observed channel syndrome. The channel probability for \( v_i \) is then perturbed as follows:

- If \( S_i = 0 \) and \( S_i = 1 \), then:

\[
P_{ch}(\hat{P}_t = a) = \begin{cases} p/2, & \text{if } a = 0 \text{ or } \hat{H}_i, \\ (1 - p)/2, & \text{otherwise}. \end{cases} \tag{40}
\]

- If \( S_i = 1 \) and \( S_i = 0 \), then:

\[
P_{ch}(\hat{P}_t = a) = \begin{cases} (1 - p)/2, & \text{if } a = 0 \text{ or } \hat{H}_i, \\ p/2, & \text{otherwise}. \end{cases} \tag{41}
\]

The perturbed values are fed to the standard non-binary BP decoder, which provides a new estimate of the channel error. The perturbation process is repeated, until all the checks are satisfied or the maximum number of feedbacks \( n_f \) is reached. Since these perturbations are more reliable than random perturbations, this method outperforms the random perturbation based heuristic method of [29].

V. MODIFIED NON-BINARY DECODING FOR HOMOGENEOUS CSS-TYPE QLDPC CODES

Let us recall from Section IV-C that homogeneous CSS-type QLDPC codes, which include both the dual-containing

\(^{19}\) Check nodes for which the computed syndrome does not match the observed syndrome are known as frustrated checks [81].
construction as well as the EA-QLDPC codes, have an excessive number of short cycles. The $i$th and $(i + m/2)$th rows of the associated PCM $\hat{H}$ are related by a multiple of $\omega$, i.e. we have $\hat{H}_i = \omega \hat{H}_{i+m/2}$, as seen in Eq. (37). For example, consider the 7-qubit Steane code [32], which is derived from the (7, 4) Hamming code. The PCM of a classical (7, 4) Hamming code is given by:

$$H'_z = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{pmatrix}. \quad (42)$$

Consequently, according to Eq. (37), the corresponding PCM of the 7-qubit Steane code is:

$$\hat{H} = \begin{pmatrix} \omega & 0 & 0 & \omega & 0 & \omega & \omega \\ 0 & \omega & 0 & \omega & 0 & \omega & 0 \\ 0 & 0 & \omega & 0 & \omega & \omega & \omega \\ 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{pmatrix}. \quad (43)$$

whose Tanner graph is plotted in Fig. 13. As gleaned from Fig. 13, cycles of length 4 exist between all the variable nodes connected to the checks $c_i$ and $c_{i+3}$. The dual-containing nature of Steane code also results in some additional short cycles. However, here we focus our attention only on the cycles resulting from the homogeneous CSS structure.

To alleviate the impact of these short cycles, we propose a modified Tanner graph, which amalgamates the check nodes $c_i$ and $c_{i+m/2}$ into a supernode, thereby eliminating the cycles. The resultant modified Tanner graph is given in Fig. 14. Based on the modified Tanner graph of Fig. 14, the horizontal messages exchanged between the supernodes $(c_i, c_{i+m/2})$ and the variable nodes $v_i$ aim for satisfying both the checks $c_i$ and $c_{i+m/2}$ simultaneously. Therefore, we have to modify Eq. (33) and (34) of the non-binary BP accordingly.

![Tanner graph of the 7-qubit Steane code.](image)

![Modified Tanner graph of 7-qubit Steane code. Check nodes $c_i$ and $c_{i+m/2}$ are combined to form a supernode.](image)

**TABLE 5.** List of all possible values of $\hat{S}_i$ and the corresponding values of $\hat{S}_{i+m/2}$ and the binary syndromes $S_i = \text{Tr}(\hat{S}_i)$ and $S_{i+m/2} = \text{Tr}(\hat{S}_{i+m/2})$.

<table>
<thead>
<tr>
<th>$S_i$</th>
<th>$S_{i+m/2}$</th>
<th>$\hat{S}_i$</th>
<th>$\hat{S}_{i+m/2}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$\omega$</td>
<td>0</td>
<td>1</td>
<td>$\omega$</td>
</tr>
<tr>
<td>$\omega$</td>
<td>1</td>
<td>$\omega$</td>
<td>1</td>
</tr>
</tbody>
</table>

Consequently, for the supernode $C_i = (c_i, c_{i+m/2})$, the PDFs of Eq. (33) and (34) may be modified as follows:

- If the observed channel syndromes are $(S_i, S_{i+m/2}) = (0, 0)$, then:

$$PDF[m_{i}^{\hat{S}_i}_{v_{i}}] \propto PDF[m_{i+1}^{\hat{S}_{i+m/2}}_{v_{i}}]. \quad (44)$$

- If the observed channel syndromes obey $(S_i, S_{i+m/2}) = (0, 1)$, then we have:

$$PDF[m_{i}^{\hat{S}_i}_{v_{i}}] \propto PDF[m_{i+1}^{\hat{S}_{i+m/2}}_{v_{i}}] \propto PDF[m_{i}^{\hat{S}_i}_{v_{i}}]. \quad (45)$$

- If the observed channel syndromes satisfy $(S_i, S_{i+m/2}) = (1, 0)$, then we arrive at:

$$PDF[m_{i}^{\hat{S}_i}_{v_{i}}] \propto PDF[m_{i+1}^{\hat{S}_{i+m/2}}_{v_{i}}] \propto PDF[m_{i}^{\hat{S}_i}_{v_{i}}]. \quad (46)$$

- If the observed channel syndromes are $(S_i, S_{i+m/2}) = (1, 1)$, then:

$$PDF[m_{i}^{\hat{S}_i}_{v_{i}}] \propto PDF[m_{i+1}^{\hat{S}_{i+m/2}}_{v_{i}}] \propto PDF[m_{i}^{\hat{S}_i}_{v_{i}}]. \quad (47)$$

Here $\hat{S}_i = (\hat{H}_i \times \tilde{a})$ for $\tilde{a} \in \{0, 1, \omega, \bar{\omega}\}$.

Hence, Eq. (44) to (47) ensure that both the constituent check nodes $c_i$ and $c_{i+m/2}$ of the supernode $C_i$ are satisfied simultaneously. This is achieved without any additional complexity overhead. In fact, our proposed method requires less computations than the standard non-binary BP, because the number of check nodes is reduced to half.

Let us consider the Steane code of Eq. (43) for explaining the decoding procedure. Assume that when the 7-qubit codeword is transmitted over a quantum depolarizing channel having a depolarizing probability of $p = 0.26$, an X error is inflicted on the first qubit, i.e.
BP algorithm proceeds as follows: We first run the standard non-binary BP on the Tanner graph of Fig. 13 for estimating the channel error. The non-binary BP algorithm proceeds as follows:

- **Initialization:** The messages $\tilde{m}^{\hat{a}}_{c_i \rightarrow v_j}$, which are sent from the variable nodes $v_j \in \{v_0, v_1, \ldots, v_6\}$ to the check nodes $c_i \in \{c_0, c_1, \ldots, c_5\}$ for $\hat{a} \in \{0, 1, \omega, \bar{\omega}\}$, are initialized according to the channel depolarizing probability of $p = 0.26$, i.e. we have:

$$
\tilde{m}^{\hat{a}}_{c_i \rightarrow v_j} = \begin{cases} 0.74, & \text{if } \hat{a} = 0 \\ 0.0867, & \text{if } \hat{a} \in \{1, \omega, \bar{\omega}\}.
\end{cases} \quad (49)
$$

- **Horizontal message exchange:** The horizontal messages $m^{\hat{a}}_{c_i \rightarrow v_j}$, equivalent to Eq. (26), which are sent from the check nodes $c_i$ to the variable nodes $v_j$, may be computed using the FFT-based algorithm of [80]. The algorithm is briefly outlined below:

**Step 1 (PDF of $\tilde{m}^{\hat{a}}_{c_i \rightarrow v_j}$):** Recall from Section IV-B that we have:

$$
\hat{a}_s = \tilde{H}_d \times \bar{\tilde{a}}. \quad (50)
$$

Consequently, the PDF of $\tilde{m}^{\hat{a}}_{c_i \rightarrow v_j}$ can be obtained by permuting the corresponding PDF of $m^{\hat{a}}_{c_i \rightarrow v_j}$ according to the value of $\tilde{H}_d$ using Eq. (50). Let us consider the PDF of the message $m^{\hat{a}}_{c_i \rightarrow v_j}$, which is equivalent to $(m^{0}_{v_0 \rightarrow c_0}, m^{1}_{v_0 \rightarrow c_0}, m^{\omega}_{v_0 \rightarrow c_0}, m^{\bar{\omega}}_{v_0 \rightarrow c_0})$. The corresponding entry in $\tilde{H}$ is $\tilde{H}_{00} = \omega$. Hence, using Eq. (50) and Table 2, we get $\hat{a}_s = (0, \omega, 1, \bar{\omega})$ for $\hat{a} = (0, 1, \omega, \bar{\omega})$. This implies that the PDF of $\tilde{m}^{\hat{a}}_{c_i \rightarrow v_j}$ is equivalent to $(m^{0}_{v_0 \rightarrow c_0}, m^{1}_{v_0 \rightarrow c_0}, m^{\omega}_{v_0 \rightarrow c_0}, m^{\bar{\omega}}_{v_0 \rightarrow c_0})$. For the $\tilde{H}$ of Eq. (43), we may generalize the computation of $m^{\hat{a}}_{c_i \rightarrow v_j}$ as follows:

$$
\text{PDF} \{\tilde{m}^{\hat{a}}_{c_i \rightarrow v_j}\} = (m^{0}_{c_0 \rightarrow v_j}, m^{1}_{c_0 \rightarrow v_j}, m^{\omega}_{c_0 \rightarrow v_j}, m^{\bar{\omega}}_{c_0 \rightarrow v_j}). \quad (51)
$$

if $c_i \in \{c_0, c_1, c_2\}$, while we have:

$$
\text{PDF} \{\tilde{m}^{\hat{a}}_{c_i \rightarrow v_j}\} = (m^{0}_{c_4 \rightarrow v_j}, m^{1}_{c_4 \rightarrow v_j}, m^{\omega}_{c_4 \rightarrow v_j}, m^{\bar{\omega}}_{c_4 \rightarrow v_j}), \quad (52)
$$

if $c_i \in \{c_3, c_4, c_5\}$. Furthermore, given the initial PDF of Eq. (49), Eq. (51) and Eq. (52) reduces to:

$$
\tilde{m}^{\hat{a}}_{c_i \rightarrow v_j} = \begin{cases} 0.74, & \text{if } \hat{a} = 0 \\ 0.0867, & \text{if } \hat{a} \in \{1, \omega, \bar{\omega}\}, \quad (53)
\end{cases}
$$

for $c_i \in \{c_0, c_1, \ldots, c_5\}$.

**Step 2 (FFT of the PDF of $\tilde{m}^{\hat{a}}_{c_i \rightarrow v_j}$):** Recall from Section IV-B that the convolution operation required in Eq. (33) and Eq. (34) is equivalent to the multiplication of the corresponding PDFs in the frequency domain. The FFT of the PDF of Eq. (53) can be computed using the FFT matrix as follows:

$$
\mathcal{F} \{\tilde{m}^{\hat{a}}_{c_i \rightarrow v_j}\} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & -1 & 1 \\ 1 & -1 & -1 & 1 & 1 & -1 \\ -1 & 1 & 1 & -1 & -1 & 1 \\ -1 & -1 & 1 & 1 & -1 & 1 \\ \end{pmatrix} \begin{pmatrix} m^{0}_{v_0 \rightarrow c_0} \\ m^{1}_{v_0 \rightarrow c_0} \\ m^{\omega}_{v_0 \rightarrow c_0} \\ m^{\bar{\omega}}_{v_0 \rightarrow c_0} \end{pmatrix}, \quad (54)
$$

where $\mathcal{F}$ denotes the FFT operation. Hence, the FFT of the PDF of Eq. (53) is equivalent to:

$$
\mathcal{F} \{\tilde{m}^{\hat{a}}_{c_i \rightarrow v_j}\} = \begin{pmatrix} 0.6533 \\ 0.6533 \\ 0.6533 \end{pmatrix}. \quad (55)
$$

**Step 3 (Convolution of PDFs):** The convolution operations of Eq. (33) and Eq. (34), which are invoked for computing the horizontal messages related to the variable node $v_j$, can be carried out using the FFT as follows:

$$
\bigotimes_{v_j \prime} \text{PDF} \{\tilde{m}^{\hat{a}}_{c_i \rightarrow v_j}\} \equiv \mathcal{F}^{-1} \left\{ \prod_{v_j \prime} \mathcal{F} \{\tilde{m}^{\hat{a}}_{c_i \rightarrow v_j}\} \right\}. \quad (56)
$$

where $\mathcal{F}^{-1}$ denotes the Inverse FFT (IFFT) operation and $v_j \prime \in V(c) \setminus v_j$. Given the $H$ of Eq. (43) and the FFT of Eq. (55), we get:

$$
\prod_{v_j \prime} \mathcal{F} \{\tilde{m}^{\hat{a}}_{c_i \rightarrow v_j}\} = \begin{pmatrix} 1 \\ 0.2788 \\ 0.2788 \\ 0.2788 \end{pmatrix}. \quad (57)
$$

Then, the inverse FFT of Eq. (57) is computed by multiplying it with the FFT matrix, which is the same as that in Eq. (54). More explicitly, we have:

$$
\mathcal{F}^{-1} \left\{ \prod_{v_j \prime} \mathcal{F} \{\tilde{m}^{\hat{a}}_{c_i \rightarrow v_j}\} \right\} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & 1 \\ 1 & -1 & -1 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 0.2788 \\ 0.2788 \\ 0.2788 \end{pmatrix} = \begin{pmatrix} 1.8364 \\ 0.7212 \\ 0.7212 \end{pmatrix}. \quad (58)
$$
Finally, the PDF of Eq. (58) is normalized to yield the output of Eq. (56), i.e., we get:

$$\begin{align*}
\mathbb{P}(\hat{m}_{c_i\rightarrow v_j}) = \begin{pmatrix}
0.4591 & 0.1803 \\
0.1803 & 0.1803 \\
0.3197 & 0.3197
\end{pmatrix}.
\end{align*}$$

(59)

Step 4 (PDF of $\hat{m}_{c_i\rightarrow v_j}$): The PDF of the messages $\hat{m}_{c_i\rightarrow v_j}$ may be computed using Eq. (33) or Eq. (34) depending on the value of the syndrome observed, which was computed in Eq. (48). Since the syndrome of Eq. (48) is 1 for the first check node $c_0$, we use Eq. (34) for computing the PDF of the messages emerging from the check node $c_0$. Therefore, we get:

$$\begin{align*}
\hat{m}_{c_0\rightarrow v_j} &= \begin{pmatrix}
0.1803 \\
0.1803 \\
0.3197 \\
0.3197
\end{pmatrix}.
\end{align*}$$

(60)

Furthermore, the syndrome of Eq. (48) has a value of 0 for all other check nodes. Therefore, we use Eq. (33) for $c_i \neq c_0$, which yields the following PDF:

$$\begin{align*}
\hat{m}_{c_i\rightarrow v_j} &= \begin{pmatrix}
0.3197 \\
0.1803 \\
0.1803
\end{pmatrix}.
\end{align*}$$

(61)

Step 5 (PDF of $m_{c_i\rightarrow v_j}$): For the sake of retrieving the messages $m_{c_i\rightarrow v_j}$ from the PDF of $\hat{m}_{c_i\rightarrow v_j}$, the resultant PDFs of Eq. (60) and Eq. (61) have to be permuted, as we did in Step 1. More specifically, the permutation operation, which is required for this step, is the reverse of the permutation operation carried out in Step 1. Let us consider the check nodes $c_i \in \{c_0, c_1, c_2\}$, for which the non-zero values of $\hat{H}_{it}$ are always equal to $\omega$ (or equivalently all the branches emerging from these check nodes in the Tanner graph of Fig. 13 are labeled with the Pauli-$Z$ operator). Furthermore, recall from Step 1 that

$$\hat{a}_i = (0, \omega, 1, \bar{\omega})$$

for $\hat{a}_i = (0, 1, \omega, \bar{\omega})$, when $\hat{H}_{it} = \omega$. This implies that the PDF of $m_{c_i\rightarrow v_j}$ is equivalent to

$$\begin{pmatrix}
\hat{m}_{c_0\rightarrow v_j} \\
\hat{m}_{c_1\rightarrow v_j} \\
\hat{m}_{c_2\rightarrow v_j}
\end{pmatrix}$$

for $c_i \in \{c_0, c_1, c_2\}$. For all other check nodes, the PDF of $m_{c_i\rightarrow v_j}$ is the same as that of $\hat{m}_{c_i\rightarrow v_j}$, because we have $\hat{H}_{it} = 1$. Therefore, the resultant PDFs are as follows:

$$\begin{align*}
m_{c_0\rightarrow v_j} &= \begin{pmatrix}
0.1803 \\
0.3197 \\
0.1803 \\
0.3197
\end{pmatrix},
\end{align*}$$

(62)

for $c_i = c_0$, while we have:

$$\begin{align*}
m_{c_i\rightarrow v_j} &= \begin{pmatrix}
0.3197 \\
0.1803 \\
0.3197 \\
0.1803
\end{pmatrix},
\end{align*}$$

(63)

TABLE 6. Marginal probability $P(P_t = \hat{a})$ after the first iteration, when the standard non-binary BP decoding algorithm is invoked over the Tanner graph of the 7-qubit Steane code for transmission through a depolarizing channel having $p = 0.26$, which inflicts an X error on the first qubit, i.e., we have $P = \Xi_{111}$. For $c_i \in \{c_1, c_2\}$, and we have:

$$m_{c_i\rightarrow v_j} = \begin{pmatrix}
0.3197 \\
0.3197 \\
0.1803 \\
0.1803
\end{pmatrix}. $$

(64)

for the remaining check nodes $c_i \in \{c_3, c_4, c_5\}$.

- **Vertical message exchange:** We next compute the vertical messages $m_{v_j\rightarrow c_i}$ using Eq. (27). For example, consider the message $m_{v_0\rightarrow c_0}$, which is destined from the variable node $v_0$ to the check node $c_0$. Since the variable node $v_0$ is only connected to $c_0$ and $c_3$ in the Tanner graph of Fig. 13, the message $m_{v_0\rightarrow c_0}$ may be computed as:

$$m_{v_0\rightarrow c_0} = K P_{ch}(P_0 = \hat{a}) \times m_{c_1\rightarrow v_0} = \begin{pmatrix}
0.8005 \\
0.0937 \\
0.0529 \\
0.0529
\end{pmatrix}.$$  

(65)

- **Element-wise marginal probability:** The element-wise marginal probabilities of the error on the variable node $v_i$, given the observed syndrome $S$, may be computed using Eq. (28). Let us consider again the variable node $v_0$, which is connected to check nodes $c_0$ and $c_3$. Consequently, the resultant marginal distribution of the error $P_t$ inflicted on the variable node $v_i$ may be computed as:

$$P(P_0 = \hat{a} | S) = K P_{ch}(P_0 = \hat{a}) \times m_{c_0\rightarrow v_0} \times m_{c_3\rightarrow v_0} = \begin{pmatrix}
0.7189 \\
0.1493 \\
0.0475 \\
0.0842
\end{pmatrix}.$$  

(66)

The process is repeated for all the variable nodes and the resultant marginalized probabilities are tabulated in Table 6.

- **Hard decision & syndrome check:** Finally, a hard decision is made for the sake of finding the most likely error $P_t$, which maximizes the marginal probability computed in the previous step. The resultant values of $P_t$ are listed in the last column of Table 6. More specifically, the probability of ‘no-error’ dominates for all the variable nodes. The specific syndrome corresponding to the resultant
estimated error $\hat{P}_t$ does not match the observed syndrome $S$ of Eq. (48). Hence, the algorithm repeats itself from the horizontal message exchange step onwards.

Fig. 15a plots the resultant marginal probability $P(P_t = \hat{a}|S)$ for the first qubit, as the iterations proceed. As seen from Fig. 15a, the standard decoding algorithm fails to converge. We next invoke our modified non-binary BP algorithm for the sake of analyzing the impact of the proposed algorithm.

Recall from Fig. 14 that the check nodes $c_i$ and $c_{i+3}$ are amalgamated into a single supernode $C_i$. The corresponding observed syndrome values of Eq. (48) are also amalgamated, which yields $(S_0, S_3) = (1, 0), (S_1, S_4) = (0, 0)$ and $(S_2, S_5) = (0, 0)$. Consequently, the modified BP differs from the standard non-binary BP in Step 4 of the ‘horizontal message exchange’, since it takes into account the amalgamated supernodes, rather than the individual check nodes.

Using Eq. (44) to Eq. (47), Step 4 of the ‘horizontal message exchange’ may be carried out as follows:

1. Step 4 (PDF of $\hat{m}_{C_i \rightarrow v_t}$): Since the syndrome observed for the supernode $C_0$ is $(S_0, S_3) = (1, 0)$, we use Eq. (46) for computing the PDF of the messages emerging from this supernode. Consequently, we arrive at:

$$\hat{m}_{C_0 \rightarrow v_t} = \bigotimes_{v_t} PDF\{\hat{m}_{v_t \rightarrow C_0}\} = \begin{pmatrix} 0.1803 \\ 0.1803 \\ 0.4591 \\ 0.1803 \end{pmatrix}. \quad (67)$$

Furthermore, since the syndrome is $(S_i, S_{i+3}) = (0, 0)$ for all other supernodes, we use Eq. (44) for computing the corresponding PDFs. Hence, we get:

$$\hat{m}_{C_i \rightarrow v_t} = \bigotimes_{v_t} PDF\{\hat{m}_{v_t \rightarrow C_i}\} = \begin{pmatrix} 0.4591 \\ 0.1803 \\ 0.1803 \end{pmatrix}. \quad (68)$$

for $C_i \in \{C_1, C_2\}$.

The rest of the decoding algorithm is the same as the standard non-binary BP, except that we only have three supernodes in the modified Tanner graph of Fig. 14 in contrast to the six check nodes of Fig. 13. The resultant marginalized probabilities are tabulated in Table 7, while Fig. 15b plots the marginal probability for the first qubit, as the iterations proceed. We may observe in Fig. 15b that our modified BP algorithm converges to the correct estimate in as few as two iterations.

![FIGURE 15. Evolution of the marginal probability for the first qubit of the 7-qubit Steane code for transmission through a depolarizing channel having $p = 0.26$, which inflicts an X error on the first qubit, i.e. we have $\mathcal{P} = \Xi\mathrm{III}$. Standard BP fails to converge, while our modified BP converges to the correct solution in two iterations. (a) Standard non-binary BP. (b) Modified non-binary BP.](image)

**Table 7.** Marginal probability $P(P_t = \hat{a}|S)$ after the first iteration, when the modified non-binary BP decoding algorithm is invoked over the Tanner graph of the 7-qubit Steane code for transmission through a depolarizing channel having $p = 0.26$, which inflicts an X error on the first qubit, i.e. we have $\mathcal{P} = \Xi\mathrm{III}$. 

<table>
<thead>
<tr>
<th>$t$</th>
<th>$\delta = 0$</th>
<th>$\delta = 1$</th>
<th>$\delta = \omega$</th>
<th>$\delta = \Xi$</th>
<th>$P_t$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.1946</td>
<td>0.6525</td>
<td>0.0764</td>
<td>0.0764</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0.8788</td>
<td>0.0404</td>
<td>0.0404</td>
<td>0.0404</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0.8788</td>
<td>0.0404</td>
<td>0.0404</td>
<td>0.0404</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0.8271</td>
<td>0.0969</td>
<td>0.0380</td>
<td>0.0380</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>0.8271</td>
<td>0.0969</td>
<td>0.0380</td>
<td>0.0380</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>0.8271</td>
<td>0.0969</td>
<td>0.0380</td>
<td>0.0380</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>0.8271</td>
<td>0.0969</td>
<td>0.0380</td>
<td>0.0380</td>
<td>0</td>
</tr>
</tbody>
</table>

**VI. REWEIGHTED BP FOR GRAPHS EXHIBITING CYCLES**

Belief propagation is capable of providing a reasonably good approximation to the optimization problem of Eq. (24), provided that the underlying Tanner graph has a sufficiently high girth. However, it is not guaranteed to converge or may converge onto an incorrect solution in the presence of cycles [84], [85]. Furthermore, it may require a large number of iterations for achieving convergence, especially in the high noise regime, thereby imposing a higher complexity. These shortcomings of the classic BP algorithm are primarily due to the fact that the BP messages become dependent with time when short cycles exist in the ‘Tanner graph’. Alternatively, we may refer to the messages as being ‘over-confident’ or ‘over-estimated’. To alleviate the impact of this over-confidence, Wainwright et al. [84] conceived
the Tree-Reweighted Belief Propagation (TRW-BP) method for pair-wise interactions, which improves the convergence of the classic BP by reweighting the edges of the underlying graph with their Edge Appearance Probabilities (EAP). The TRW-BP algorithm was extended to higher-order interactions in [23] and [24], whereby EAPs were replaced by the Factor Appearance Probabilities (FAPs) of the nodes. Based on this extended TRW-BP, Wymeersch et al. re-formulated the vertical message exchange step of the classic BP (Eq. (27)) as [23], [24]:

\[ m^a_{v_i \rightarrow c_j} = K P_{\text{ch}}(P_t = a) \left( m^a_{c_j \rightarrow v_t} \right)^{\rho_i - 1} \prod_{c_i' \in C(v_t) \setminus c_j} \left( m^a_{c_i' \rightarrow v_t} \right)^{\rho_{c_i'}}, \]  

where \( \rho_i \) is the FAP of the \( i \)th check node. Similarly, the computation of the element-wise marginal probability (Eq. (28)) was modified as:

\[ P(P_t = a|S) = K P_{\text{ch}}(P_t = a) \prod_{c_j \in C(v_t)} \left( m^a_{c_j \rightarrow v_t} \right)^{\rho_{c_j}}. \]  

Both Eq. (69) and (70) reduces to the classic BP for \( \rho_i = 1 \forall i \). The TRW-BP technique requires the optimization of \( \rho_i \) for all nodes. To reduce this optimization task, Wymeersch et al. [23], [24] also proposed the URW-BP, which invokes a uniform FAP value for all the nodes, where we have \( \rho_i = \rho \forall i \). Various other variations of TRW-BP have been investigated in [86]-[89] for classical binary LDPC codes, which demonstrate that the TRW-BP effectively improves the convergence of binary LDPC codes, when the number of iterations is not too high. Inspired by these results, in Section VII we also analyze the impact of URW-BP on the non-binary decoding of quantum LDPC codes, which are known to have unavoidable short cycles.

### Table 8. System I - simulation parameters.

<table>
<thead>
<tr>
<th>QLDP C Matrix</th>
<th>Mackay’s bicycle code</th>
</tr>
</thead>
<tbody>
<tr>
<td>Code Construction</td>
<td>Mackay’s bicycle code</td>
</tr>
<tr>
<td>Coded qubits</td>
<td>( n = 800 )</td>
</tr>
<tr>
<td>Information qubits</td>
<td>( k = 400 )</td>
</tr>
<tr>
<td>E-bits</td>
<td>( c = 0 )</td>
</tr>
<tr>
<td>Row weight</td>
<td>30</td>
</tr>
</tbody>
</table>

### VII. RESULTS AND DISCUSSIONS

#### A. MODIFIED NON-BINARY DECODING

For the sake of quantifying the attainable performance gain of our modified non-binary BP of Section V, in this section we compare its performance in conjunction with the decoding algorithms of Section IV. Our first system of Table 8 relies on Mackay’s 1/2-rate [800, 400] bicycle code having a row weight of 30. The corresponding WER performance recorded for various channel depolarizing probabilities is plotted in Fig. 16, where we have considered the following decoders:

1. **Binary**: the binary BP decoding algorithm of Section IV-A.
2. **Standard Non-Binary**: the non-binary BP decoding algorithm of Section IV-B.
3. **Random Perturbation**: the random perturbation technique [81] of Section IV-C.
4. **Enhanced Feedback**: the enhanced feedback method [83] of Section IV-C.
5. **Modified Non-Binary**: our modified non-binary BP of Section V.
6. **Modified & Enhanced Feedback**: our modified non-binary BP of Section V amalgamated with the enhanced feedback method [83] of Section IV-C.

For all the decoding schemes, we have used a maximum of \( \tau_{\text{max}} = 90 \) iterations. Furthermore, for both the ‘Random Perturbation’ as well as for the ‘Enhanced Feedback’, we set \( \gamma_{\text{pert}} = 40 \), while the random perturbation strength was set to \( \delta = 0.1 \) and the maximum number of feedbacks to \( n_{\alpha} = 40 \) for the ‘Enhanced Feedback’. These simulation parameters are tabulated in Table 8. Each decoding algorithm iterates until either a valid error is found or the maximum number of iterations is reached. Furthermore, the WER metric here counts both the detected as well as the undetected block errors.

We may observe in Fig. 16 that the ‘Binary’ decoder exhibits the worse performance. Using the ‘Binary’ decoder, we achieve a WER of \( 10^{-4} \) at a channel depolarizing probability of \( p = 0.0085 \), which increases to \( p = 0.01075 \) with the ‘Standard Non-Binary’ decoder. This is equivalent to a \((0.0075-0.0085) \times 100\) = 26% depolarizing probability increase that the decoder can cope with. Furthermore, the ‘Random Perturbation’, the ‘Enhanced Feedback’ and the

---

20 EAP of an edge represents the probability of appearance of that edge in a randomly chosen spanning tree.

21 FAP denotes the appearance probability of a node in the collection of trees [23], [24].

22 We have used the decoding parameters of [83].
‘Modified Non-Binary’ decoders have a similar performance at low noise levels, increasing the tolerable depolarizing probability to \( p = 0.014 \) at a WER of \( 10^{-4} \), which corresponds to a \( (0.014 - 0.0175) \times 100 ) = 30\% \) increase of \( p \) at WER \( = 10^{-4} \) with respect to the ‘Standard Non-Binary’ decoder. Furthermore, with the ‘Modified & Enhanced Feedback’ configuration, the tolerable depolarizing probability increases to \( p = 0.017 \) at a WER of \( 10^{-4} \), which is equivalent to about \( (0.014 - 0.014) \times 100 ) = 21\% \) increase with respect to \( p = 0.014 \). Table 9 summarizes these results.

### Table 9. Achievable depolarizing probability (p) at a WER of \( 10^{-4} \), based on Fig. 16.

<table>
<thead>
<tr>
<th>Dec. No.</th>
<th>Decoding Method</th>
<th>( p )</th>
<th>Improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Binary</td>
<td>0.0085</td>
<td>-</td>
</tr>
<tr>
<td>2</td>
<td>Standard Non-Binary</td>
<td>0.01075</td>
<td>26% w.r.t. Dec. 1</td>
</tr>
<tr>
<td>3</td>
<td>Random Perturbation</td>
<td>0.014</td>
<td>30% w.r.t. Dec. 2</td>
</tr>
<tr>
<td>4</td>
<td>Enhanced Feedback</td>
<td>0.014</td>
<td>30% w.r.t. Dec. 2</td>
</tr>
<tr>
<td>5</td>
<td>Modified Non-Binary</td>
<td>0.014</td>
<td>30% w.r.t. Dec. 2</td>
</tr>
<tr>
<td>6</td>
<td>Modified &amp; Enhanced Feedback</td>
<td>0.017</td>
<td>21% w.r.t. Dec. 5</td>
</tr>
</tbody>
</table>

The performance of our ‘Modified Non-Binary’ BP at a WER of \( 10^{-4} \) is similar to that of the heuristic methods, namely ‘Random Perturbation’ and ‘Enhanced Feedback’. However, the ‘Modified Non-Binary’ technique imposes a lower decoding complexity in terms of the average number of decoding iterations, which is evidenced in Fig. 17. Consequently, our ‘Modified Non-Binary’ BP converges faster than the existing decoding schemes. In particular, in the high-noise regime, our ‘Modified Non-Binary’ decoder outperforms both the ‘Random Perturbation’ and the ‘Enhanced Feedback’ in terms of its WER performance recorded in Fig. 16 as well as in terms of the average number of iterations seen in Fig. 17. As compared to the ‘Standard Non-Binary’ decoder, the ‘Modified Non-Binary’ algorithm always yields a lower WER and invokes on average less decoding iterations. We may observe furthermore in Fig. 17 that the amalgamated ‘Modified & Enhanced Feedback’ invokes less iterations as compared to the ‘Enhanced Feedback’, while the performance of the former is also superior in terms of the WER curve of Fig. 16. This is again due to the fact that the modified BP of Section V facilitates faster convergence as compared to the standard non-binary decoding. More specifically, in the region of interest, i.e. for \( p \leq 0.017 \) corresponding to the desired WER of \( \leq 10^{-4} \), the combination of the enhanced feedback method with our modified BP, namely ‘Modified & Enhanced Feedback’, imposes almost the same complexity as that imposed by the ‘Modified Non-Binary’ BP, when used on its own. However, the former exhibits a much lower WER than the latter. We compare furthermore the performance of all the decoding schemes at a depolarizing probability of \( p = 0.016 \) in Table 10.

### Table 10. Performance comparison in terms of the achievable WER and the average number of decoding iterations (\( I_{avg} \)) invoked at a depolarizing probability of \( p = 0.016 \), based on Fig. 16 and Fig. 17.

<table>
<thead>
<tr>
<th>Dec. No.</th>
<th>Decoding Method</th>
<th>WER</th>
<th>( I_{avg} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Binary</td>
<td>1.5710 ( -4 )</td>
<td>3.98</td>
</tr>
<tr>
<td>2</td>
<td>Standard Non-Binary</td>
<td>1.4710 ( -4 )</td>
<td>4.28</td>
</tr>
<tr>
<td>3</td>
<td>Random Perturbation</td>
<td>4.5710 ( -4 )</td>
<td>7.52</td>
</tr>
<tr>
<td>4</td>
<td>Enhanced Feedback</td>
<td>4.5710 ( -4 )</td>
<td>5.08</td>
</tr>
<tr>
<td>5</td>
<td>Modified Non-Binary</td>
<td>3.6710 ( -4 )</td>
<td>2.81</td>
</tr>
<tr>
<td>6</td>
<td>Modified &amp; Enhanced Feedback</td>
<td>5.2710 ( -4 )</td>
<td>2.96</td>
</tr>
</tbody>
</table>

Let us now compare the performance of the different decoding schemes in the context of our second system of Table 11, relying on the homogeneous EA-LDPC code of [83] having \( n = 816 \), \( k = 404 \) and \( e = 404 \), which is derived from the Mackay’s classical (816, 408) LDPC, having a row weight of 6 and a column weight of 3. For all the decoding schemes, we have used a maximum of \( I_{max} = 90 \) iterations. Furthermore, for both the ‘Random Perturbation’ as well as for the ‘Enhanced Feedback’ methods, we set \( T_{pert} = 40 \), while the random perturbation strength was set to \( \delta = 0.1 \) and the maximum number of feedbacks \( n_f = 81 \) was used for the ‘Enhanced Feedback’ decoder. These simulation parameters are summarized in Table 11. The resultant WER performance curves are compared in Fig. 18, while the average number of decoding iterations invoked for varying channel depolarizing probabilities are compared in Fig. 19. As observed from Fig. 18, the ‘Binary’ decoder achieves a WER of \( 10^{-4} \) at \( p = 0.057 \), which increases to \( p = 0.069 \) when the ‘Standard Non-Binary’ decoder is invoked. Consequently, the ‘Standard Non-Binary’ increases the tolerable depolarizing...
probability by \((0.069 - 0.057) \times 100\) = 21\% as compared to the ‘binary’ decoder. This is further increased to \(p = 0.076\) in conjunction with the ‘Random Perturbation’, which corresponds to about \((0.076 - 0.069) \times 100\) = 10\% increase and to \(p = 0.082\) for the ‘Enhanced Feedback’, which represents a \((0.082 - 0.069) \times 100\) = 19\% increase. By contrast, our ‘Modified Non-Binary’ BP exhibits a WER of \(10^{-4}\) around \(p = 0.085\), which corresponds to a \((0.085 - 0.069) \times 100\) = 23\% increase as compared to the ‘Standard Non-Binary’ decoder. Using the heuristic enhanced feedback approach with our modified BP, namely ‘Modified & Enhanced Feedback’ provides a further increase to \(p = 0.0945\), which represents a \((0.0945 - 0.085) \times 100\) = 11\% increase. These results are tabulated in Table 12. In terms of the average number of decoding iterations, our ‘Modified Non-Binary’ BP always outperforms both the ‘Standard Non-Binary’ decoder as well as the ‘Random Perturbation’ and the ‘Enhanced Feedback’ solutions, as depicted in Fig. 19.

### B. UNIFORMLY-REWEIGHTED BP

Since bicycle codes exhibit numerous short cycles, we use our first system of Table 8 for the analysis of the URW-BP of Section VI, which is combined with our modified nonbinary decoder of Section V. More precisely, we amalgamate the horizontal exchange step of our modified non-binary BP with the vertical exchange step of the URW-BP.

We commence by heuristically determining the optimum value \(\rho\) of the FAP, which varies with both the channel depolarizing probability as well as with the maximum number of decoding iterations. Fig. 20 shows the impact of \(\rho\) on the WER performance at varying channel depolarizing probabilities \(\rho\) for \(I_{\text{max}} = 10, 20\) and 90 iterations. We may observe in Fig. 20 that the WER varies with the value of \(\rho\), attaining a minimum value at the optimum \(\rho\). This optimum \(\rho\) is different for each \(p\) value, tending to move towards \(\rho = 1\) as the value of \(p\) increases or as the maximum affordable number of iterations increases. The resultant values of \(\rho\) optimized for different channel depolarizing probabilities \(p\) and for different maximum number of iterations are summarized in Table 13.

### TABLE 12. Achievable depolarizing probability (\(p\)) at a WER of \(10^{-4}\), based on Fig. 18.

<table>
<thead>
<tr>
<th>Dec. No.</th>
<th>Decoding Method</th>
<th>(p)</th>
<th>Improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Binary</td>
<td>0.057</td>
<td>-</td>
</tr>
<tr>
<td>2</td>
<td>Standard Non-Binary</td>
<td>0.069</td>
<td>21% w.r.t. Dec. 1</td>
</tr>
<tr>
<td>3</td>
<td>Random Perturbation</td>
<td>0.076</td>
<td>10% w.r.t. Dec. 2</td>
</tr>
<tr>
<td>4</td>
<td>Enhanced Feedback</td>
<td>0.082</td>
<td>19% w.r.t. Dec. 2</td>
</tr>
<tr>
<td>5</td>
<td>Modified Non-Binary</td>
<td>0.085</td>
<td>25% w.r.t. Dec. 2</td>
</tr>
<tr>
<td>6</td>
<td>Modified &amp; Enhanced Feedback</td>
<td>0.0945</td>
<td>11% w.r.t. Dec. 2</td>
</tr>
</tbody>
</table>

To quantify the performance gain achieved with the aid of the URW-BP, we compare the performance of the optimized URW-BP to our modified non-binary BP in Fig. 21 for \(I_{\text{max}} = 10, 20\) and 90 iteration. Here the optimized URW-BP is based on the best values of \(\rho\) listed in Table 13. The performance curves of Fig. 21 reveal that the improvement in WER is lower for higher values of \(p\) as well as for larger values of the maximum number of affordable iterations. For example, when a maximum of \(I_{\text{max}} = 10\) decoding iterations are invoked at a WER of \(10^{-3}\), the URW-BP scheme increases \(p = 0.0125\) to \(p = 0.0155\), which is around a \((0.0155 - 0.0125) \times 100\) = 24\% increase. By contrast, for a maximum of \(I_{\text{max}} = 20\) iterations, URW-BP increases from \(p = 0.015\) to \(p = 0.017\) at a WER of \(10^{-3}\). This is equivalent to an increase of \((0.017 - 0.015) \times 100\) = 13\%. Furthermore, at an even higher maximum number of iterations of \(I_{\text{max}} = 90\), URW-BP achieves a WER of \(10^{-3}\) at \(p = 0.0185\), which is only a \((0.0185 - 0.018) \times 100\) = 3\% increase as compared to the modified non-binary algorithm.
URW-BP is amalgamated with the modified non-binary decoder and the performance is analyzed for the System I of Table 8. (a) $I_{\text{max}} = 10$ iterations. (b) $I_{\text{max}} = 20$ iterations. (c) $I_{\text{max}} = 90$ iterations.

These values are summarized in Table 14. Hence, the notion of reweighting the message probabilities is more beneficial at low depolarizing probabilities and for smaller values of the maximum affordable number of iterations. This is because at higher depolarizing probabilities (and similarly larger values of the maximum number of iterations), the messages are highly correlated.

VIII. CONCLUSIONS AND DESIGN GUIDELINES
QLDPC codes may be constructed from the classical binary as well as quaternary codes by imposing the stringent symplectic criterion on the resultant PCM, which ensures that the stabilizer generators commute with each other. The design guidelines of constructing QLDPC codes may be summarized as follows:

- An $[n, k]$ QLDPC code, having a coding rate of $R_Q = k/n$, may be constructed from a classical $(2n, n + k)$ binary LDPC code, having a coding rate of $R_c = (n + k)/2n$, if the associated PCM $H$ satisfies the stringent symplectic criterion.
- Ideally, the rows of the PCM $H$ should have at most a single overlapping value of 1 (or non-zero value in the GF(4) formalism) for the sake of avoiding length-4 cycles in the Tanner graph, which degrade the performance of the iterative decoding algorithm. Unfortunately, the symplectic criterion requires ‘even overlaps’ between the rows of $H$, thus resulting in unavoidable length-4 cycles. A major design challenge is therefore to construct good QLDPC codes in the wake of the unavoidable length-4 cycles.
- We may exploit four main global structures of the PCM $H$ for designing QLDPC codes, namely dual-containing CSS, non-dual-containing CSS, non-CSS...
From the perspective of decoding, the challenges may be summarized as follows:

- **Degeneracy**: Quantum codes are inherently degenerate in nature. This may improve the associated decoding performance if the decoder takes this degeneracy into account. Unfortunately, the BP algorithm does not exploit this degeneracy. In fact, since BP is based on marginalized probabilities, the presence of degenerate errors impairs its performance.

- **Short cycles**: Unavoidable length-4 cycles found in QLDPC codes degrade the performance of BP. This gets even worse for the homogeneous CSS codes, when they are decoded in the non-binary domain.

Heuristic methods, namely random perturbation and enhanced feedback, are known to alleviate both these issues to some extent. However, this is achieved at the cost of an increased decoding complexity. Therefore, we conceive a modified non-binary decoding algorithm for homogeneous CSS-type QLDPC codes, which successfully alleviates the issue of unavoidable length-4 cycles. Our modified decoder exhibits a superior WER performance, despite its lower decoding complexity as compared to the state-of-the-art decoding techniques. It may also be amalgamated with heuristic methods for attaining additional performance gains. We also demonstrated that URW-BP can be exploited for counteracting the issues of short-cycles.

**REFERENCES**


and the entanglement-assisted solutions of Fig. 4. The design challenges associated with each of these structures are summarized below:

- **Dual-containing CSS (Section III-A)**: Mackay’s bicycle codes are so far the best amongst the dual-containing CSS codes, but their performance is still not on par with the classical LDPC codes. This is because this construction suffers the most from having short cycles, which exist both in the binary as well as in the GF(4) formalism.

- **Non-dual-containing CSS codes (Section III-A)**: It is difficult to find a pair of sparse binary PCMs satisfying the symplectic criterion, which constitute good QLDPC codes. At the time of writing, only the SC QC-LDPC codes and the non-binary QC-QLDPC codes are known to perform close to the Hashing bound. But this comes either at the cost of pre-shared noiseless ebits or at an increased complexity.

- **Non-CSS codes (Section III-B)**: Ideally, non-CSS constructions are preferred over the CSS codes because they exploit the redundant qubits more efficiently. However, finding good non-CSS QLDPC codes remains an open challenge at the time of writing.

- **EA codes (Section III-C)**: Entanglement-assistance may assist us in achieving a performance comparable to that of the classical LDPC codes. However, this requires pre-shared ebits, which constitute a valuable resource gleaned at the cost of a transmission overhead. Therefore, efforts must be made to minimize the number of required ebits.

- Additionally, it is desirable that the resultant QLDPC code has the following attributes:
  - A structured PCM, for example a cyclic or quasi-cyclic structure, for facilitating its implementation; and
  - An unbounded minimum distance or at least a sufficiently high minimum distance for long block lengths.

QLDPC codes may be decoded using syndrome-based BP either in the binary domain or in the non-binary domain. Besides the obvious lower complexity of the binary decoding, the two main differences between these decoding regimes are:

- In contrast to the binary decoding, which assumes that the bit-flips and phase-flips are independent, non-binary decoding takes into account the correlation between them, which improves their performance.

- The number of length-4 cycles is higher in the non-binary formalism of the PCM as compared to the binary one. This tends to degrade the performance of the non-binary decoder.

Hence, we have a pair of conflicting attributes. However, the non-binary BP outperformed the binary BP for both QLDPC codes, which we investigated in this paper.


ZUNAIRA BABAR received the B.Eng. degree in electrical engineering from the National University of Science and Technology, Islamabad, Pakistan, in 2008, and the M.Sc. (Hons.) and Ph.D. degrees in wireless communications from the University of Southampton, U.K., in 2011 and 2015, respectively. Her research interests include quantum error correction codes, channel coding, coded modulation, iterative detection, and cooperative communications.

PANAGIOTIS BOTSINIS (S’12) received the M.Eng. degree from the School of Electrical and Computer Engineering, National Technical University of Athens, Greece, in 2010, and the M.Sc. (Hons.) and Ph.D. degrees in wireless communications from the University of Southampton, U.K., in 2011 and 2015, respectively. Since 2010, he has been a member of the Technical Chamber of Greece. He is currently a Research Fellow with the Southampton Wireless Group, School of Electronics and Computer Science, University of Southampton. His research interests include quantum-assisted communications, quantum computation, iterative detection, OFDM, MIMO, multiple access systems, coded modulation, channel coding, cooperative communications, and combinatorial optimization.

DIMITRIOS ALANIS (S’13) received the M.Eng. degree in electrical and computer engineering from the Aristotle University of Thessaloniki, in 2011, and the M.Sc. degree in wireless communications from the University of Southampton, in 2012, where he is currently pursuing the Ph.D. degree with the Southampton Wireless Group, School of Electronics and Computer Science. His research interests include quantum computation and quantum information processing, quantum search algorithms, cooperative communications, resource allocation for self-organizing networks, bioinspired optimization algorithms, and classical and quantum game theory.
SOON XIN NG (S’99–M’03–SM’08) received the B.Eng. (Hons.) degree in electronics engineering and the Ph.D. degree in wireless communications from the University of Southampton, Southampton, U.K., in 1999 and 2002, respectively. From 2003 to 2006, he was a Post-Doctoral Research Fellow working on collaborative European research projects known as SCOUT, NEWCOM, and PHOENIX. Since 2006, he has been a member of the Academic Staff with the School of Electronics and Computer Science, University of Southampton. He is currently involved in the OPTIMIX and CONCERTO European projects and the IUATC and UC4G projects. He is an Associate Professor of Telecommunications with the University of Southampton. He has authored over 180 papers and co-authored two John Wiley/IEEE Press books in his research field. His research interests include adaptive coded modulation, coded modulation, channel coding, space-time coding, joint source and channel coding, iterative detection, OFDM, MIMO, cooperative communications, distributed coding, quantum error correction codes, and joint wireless-and-optical-fiber communications. He is a Chartered Engineer and fellow of the Higher Education Academy, U.K.

LAJOS HANZO (M’91–SM’92–F’04) received the degree in electronics in 1976, the Ph.D. degree in 1983, and the Doctor Honoris Causa degree from the Technical University of Budapest, in 2009. During his 38-year career in telecommunications, he has held various research and academic positions in Hungary, Germany, and the U.K. Since 1986, he has been with the School of Electronics and Computer Science, University of Southampton, U.K., as the Chair in Telecommunications. He has successfully supervised 100 Ph.D. students, co-authored 20 John Wiley/IEEE Press books in mobile radio communications totaling in excess of 10 000 pages, authored over 1500 research entries at the IEEE Xplore, acted as the TPC Chair and General Chair of the IEEE conferences, presented keynote lectures, and received a number of distinctions. He is directing 100 strong academic research teams, working on a range of research projects in the field of wireless multimedia communications sponsored by the industry, the Engineering and Physical Sciences Research Council, U.K., the European Research Council’s Advanced Fellow Grant, and the Royal Society’s Wolfson Research Merit Award. He is an enthusiastic supporter of industrial and academic liaison and offers a range of industrial courses. He is a fellow of the Royal Academy of Engineering, the Institution of Engineering and Technology, and the European Association for Signal Processing. He is also a Governor of the IEEE VTS. From 2008 to 2012, he was the Editor-in-Chief of the IEEE Press and a Chaired Professor with Tsinghua University, Beijing. He has over 22 000 citations.
AUTHOR QUERIES

AQ:1 = Please check whether the edits made in the financial section are OK.
AQ:2 = Table 1 and its citation are missed in the provided manuscript. So other tables and their corresponding citations have been renumbered in the proof. Please check and confirm that they are fine as set.
AQ:3 = Please provide the issue no. for ref. [1].
AQ:4 = Please provide the issue no. and month for refs. [2]–[4] and [22].
AQ:5 = Please provide the month. Also confirm the issue no. for ref. [5].
AQ:6 = Please provide the month for ref. [18].
AQ:7 = Please provide the department name for ref. [21].
AQ:8 = Please confirm the author name “V. Savić” is correct in throughout references.
AQ:9 = Please provide the expansion for the acronyms “OFDM and MIMO.”