entropy

MDPI

Article

Quantum Coding via Quasi-Cyclic Block Matrix

Yuan Li [1,*] and Jin-Yang Li [2]

[1] School of Electronic Information Engineering, Shanghai Dianji University, Shanghai 200240, China
[2] College of Information Engineering, NorthWest Agriculture and Forestry University, Xi'an 712100, China
[*] Correspondence: yli@sdju.edu.cn

Abstract: An effective construction method for long-length quantum code has important applications in the field based on large-scale data. With the rapid development of quantum computing, how to construct this class of quantum coding has become one of the key research fields in quantum information theory. Motivated by the block jacket matrix and its circulant permutation, we proposed a construction method for quantum quasi-cyclic (QC) codes with two classical codes. This simplifies the coding process for long-length quantum error-correction code (QECC) using number decomposition. The obtained code length N can achieve $O(n^2)$ if an appropriate prime number n is taken. Furthermore, with a suitable parameter in the construction method, the obtained codes have four cycles in their generator matrices and show good performance for low density codes.

Keywords: long-length quantum codes; stabilizer codes; jacket matrix; quasi-cyclic codes

check for updates

Citation: Li, Y.; Li, J.-Y. Quantum Coding via Quasi-Cyclic Block Matrix. *Entropy* **2023**, *25*, 537. https://doi.org/10.3390/e25030537

Academic Editor: Giuliano Benenti

Received: 11 January 2023
Revised: 17 March 2023
Accepted: 18 March 2023
Published: 21 March 2023

1. Introduction

Quantum communication requires that environmental effects and decoherence should be reduced with reliable quantum information processing. In practice, information reconciliation in quantum key distribution (QKD) is important for the secret key rate and also affects the maximum transmission distance [1–4]. Error-correcting code generally may be applied in a quantum channel to correct errors caused by channel noise and possible interventions from eavesdroppers. To obtain an acceptable level, QECC is an essential method because of its robustness and efficiency in quantum computation [5–10]. One of the advantages of quantum computation is that its high efficiency compares favorably to classical computation; it is able to handle large-scale data that classical computation cannot. Previously, most of the construction methods for QECC focused on the generation of stabilizers, and there was little research on the long code type. Furthermore, due to some fields in quantum information becoming gradually more practical, this has prompted researchers to identify a good coding method for quantum error-correcting codes of long length. For example, encoding large-scale data has potential applications in the field of machine learning (ML) with respect to big data [11–14]. Therefore, how to efficiently express classical massive data with physics-based codes is also an important research field. To obtain a general quantum code, the question is usually converted to a problem of stabilizer generation. One typical method is to obtain the generation matrix of a quantum code based on two classical codes, which are called Calderbank–Shor–Steane (CSS) codes [6,7]. By resorting to the generalization of cyclic codes, a class of classical codes called quasi-cyclic (QC) codes can build linear codes based on algebraic structure, which has improved some fundamental minimum distances [15–18]. For instance, it can satisfy the modified version of the Gilbert–Vashamov (GV) bound [19–21]. It has shown good performance when applied to form quantum codes.

In the construction process for long-sequence QECCs, how to design the generators of stabilizers based on the block matrix is key. Methods from QC low density parity check (LDPC) codes have been studied by Hagiwara et al. in terms of a probabilistic method [22,23]. In 2018, Galindo et al. applied two generators to construct a quantum version with dual-containing QC [24]. Some QC short-length codes were obtained with good parameters. Not long after that, Ezerman et al., in 2019, used QC codes with large Hermitian hulls to form QECCs over fields F_4 and F_9 [25], so that a record-breaking binary QECC was obtained. Furthermore, J. Lv et al. proposed some new binary quantum codes derived from one generator quasi-cyclic codes with a stabilizer [26,27]. Recently, some researchers have considered the application of this kind of quantum code in quantum key distribution [4]. However, few researchers have paid attention to the construction of long-length quantum QC code involved with massive data. Inspired by the previous work, we presented QC code constructions of long length to generate QECCs with a family of orthogonal jacket matrices, the main property of which are that the inverse matrix can be obtained by its element-wise inverse or block-wise inverse [28,29]. Therefore, it can be realized relatively easily with a physical circuit. Furthermore, since Gallager first proposed LDPC codes in the 1960s [30], this class of classical code has shown good performance approaching the channel capacity [31–35]. Subsequently, its quantum versions has been investigated [22,23]. However, the achievements in this field have been explored far less than their classical counterparts. The constructed quantum codes have shown good performance for low density codes based on iterative coding in our proposed construction method.

In this paper, with the advantage of the convenient implementation of the jacket transform, we applied a quasi-cyclic method via a low-density block matrix to gain long-length quantum codes. If a prime number in the proposed construction method is taken to properly choose the jacket matrix with a basic matrix and block circular matrix combined together, then a longer length of matrix to encode classical data can be obtained.

This paper is arranged as follows: In Section 2, we present some preliminary information which is necessary for QECCs. Then, in Sections 3 and 4, we investigate the construction of long-length quasi-cyclic quantum codes which are generated from block jacket matrices based on a basic matrix and circulant permutation matrices. Furthermore, the construction conditions with low density are also analyzed. Finally, conclusions are drawn in Section 5.

2. Preliminaries

Some relevant notation and basic construction methods for quantum error correction codes are first briefly reviewed below.

2.1. General Construction Methods of QECCs

A linear binary quantum error-correcting code $[[N,k,d]]$ denotes that k-data vectors are encoded to N-dimensional vectors in space $\mathcal{P}_N$, where d is the minimum distance. Consider a 2^N-dimensional Hilbert space based on a complex field $\mathbb{C}$

$$\mathcal{P}_N = \mathbb{C}^{2^N} = (\mathbb{C}^2)^{\otimes N}, \tag{1}$$

where every 2^N standard basis vector in space $\mathcal{P}_N$ is indexed by a classical binary vector $u \in F_2^N$ and denoted by $|u\rangle$. It is generally called a N-qubit state space, for which each component in the tensor product corresponds to one qubit. Similar to classical coding, a fundamental problem in quantum error correction is to generate quantum codes based on the best possible minimum distance. The general construction method for QECCs usually relies on a so-called stabilizer. A stabilizer quantum code $\mathcal{C}[[N,k,d]]$ can be gained from the stabilizer denoted as

$$S = \prod_{i-1}^{N-k} (I + M_i^{m_i}) : m_i \in \{0,1\}, \tag{2}$$

where $M_1, M_2 \cdots M_{N-k}$ are $N-k$ commuting generators of the stabilizer, which is the collection of orthogonal quantum state eigenvectors that refer to code words. Therefore, to form a stabilizer quantum code, $N-k$ generators of stabilizer S should first be designed, which are expressed by the following generator matrix

$$\mathcal{G} = (\mathcal{G}^x \quad \mathcal{G}^z)_{(N-k)\times 2N} = (M_1, M_2, \dots, M_{N-k})^T, \tag{3}$$

where $\mathcal{G}^x = (g^x_{ij})_{(N-k)\times N}$, $\mathcal{G}^z = (g^z_{ij})_{(N-k)\times N}$ for $1 \le i \le N-k, 1 \le j \le N$, and x, z are the Pauli transformation. According to the property of commuting generators, the elements of its row vector satisfy that the symplectic inner product is equal to zero.

Another well known construction method for quantum codes is the CSS code which is generated by a pair of classical codes $C_1[n, k_1, d_1]$ and $C_2[n, k_2, d_2]$, where k_i and d_i are the information code length and minimum distance, respectively. This means that this class of quantum error code is a complex vector space characterized by a pair of classical binary linear codes C_1 and C_2. Here, the parity-check matrices H_1 and H_2 of the two classical codes are required to satisfy $H_1 \cdot H_2^T = 0$, i.e., every row of H_1 is orthogonal to every row of H_2. Hence, define a CSS code as a complex linear combination of vectors:

$$\sum_{d' \in C_2^{\perp}} |c + d'\rangle \quad \text{for} \quad c \in C \tag{4}$$

where $C_2^{\perp}$ is the dual code of the classical code C_2. According to its definition, the obtained quantum CSS code $\mathcal{C}[[N, k_1 + k_2 - N, \min\{d_1, d_2\}]]$ is a class of special stabilizer code, of which the generator matrix is

$$\mathcal{G}_c = \begin{pmatrix} H_1 & O \\ O & H_2 \end{pmatrix}. \tag{5}$$

In light of the characteristics of big data processed in artificial intelligence and other fields, we shall apply the recursive relationship of a quasi-cyclic block jacket matrix to easily obtain long-length QECCs.

2.2. Error of Quantum Error-Correction Code and Bound

As with different quantum codes, errors are also labeled with strings in field F_2. This will occur when the states are transmitted through quantum channels. The traditional approach to error correction for quantum codes is to consider the single qubit flip error, the phase error or the phase-flip error, which can be described with three Pauli operators σ_1, σ_2 and $\sigma_3 = \omega\sigma_1\sigma_2$, respectively. Here, ω is the primitive root of unity. Every error on N qubits can be denoted as $e = \sigma_1^X \sigma_2^Z$, for $X = (x_1, x_2, \dots, x_n)$ and $Z = (z_1, z_2, \dots, z_n) \in F_2^N$. Reflexive stabilizer codes and CSS codes have a one-to-one correspondence by choosing a basis for the error group. In terms of the Pauli matrices, the single qubit quantum error can be described as $X(a)$ and $Z(a)$ for $a \in F_2$, whose action on $|x\rangle \in C$ is given by

$$X(a)|x\rangle = |x + a\rangle \quad \text{and} \quad Z(a)|x\rangle = \omega^{ax}|x\rangle \tag{6}$$

Therefore, it acts on an N-qubit basis state $|Q\rangle = (q_1, q_2, \dots, q_N)$ in F_2^N as follows

$$\begin{aligned} e|Q\rangle &= (-1)^{Z\cdot Q}|X + Q\rangle \\ &= (-1)^{z_1\cdot q_1 + \dots + z_N \cdot q_N}|x_1 + q_1, \dots, x_N + q_N\rangle. \end{aligned} \tag{7}$$

Furthermore, the quantum Hamming bound of a general quantum code $\mathcal{C}[[N, k, d]]$ is required to satisfy the following inequality condition [36]

$$\sum_{l=0}^{t} 3^l \binom{N}{l} \le 2^{N-k}, \tag{8}$$

which may correct up to $t = [(d-1)/2]$ quantum error bits.

3. Long-Length Coding Constructions Based on Jacket Matrix

Quasi-cyclic codes are the generalization of cyclic codes, which can generate a class of linear codes of algebraic structure. Assume C is a classical code of length N over field F_2, which is closed with a cyclic-shift operator γ. This means that any codeword $\vec{C} = (c_0, c_1, \ldots, c_{N-1}) \in C$ satisfies $\gamma\vec{C} = \gamma(c_0, c_1, \ldots, c_{N-1}) = (c_{N-1}, c_0, \ldots, c_{N-2}) \in C$. The circuit of the cyclic shift can be represented as in Figure 1. In this circuit, $\vec{C}$ as an input code vector can generate an output code vector with some integer s times of γ operation, $1 \leq s \leq N-1$.

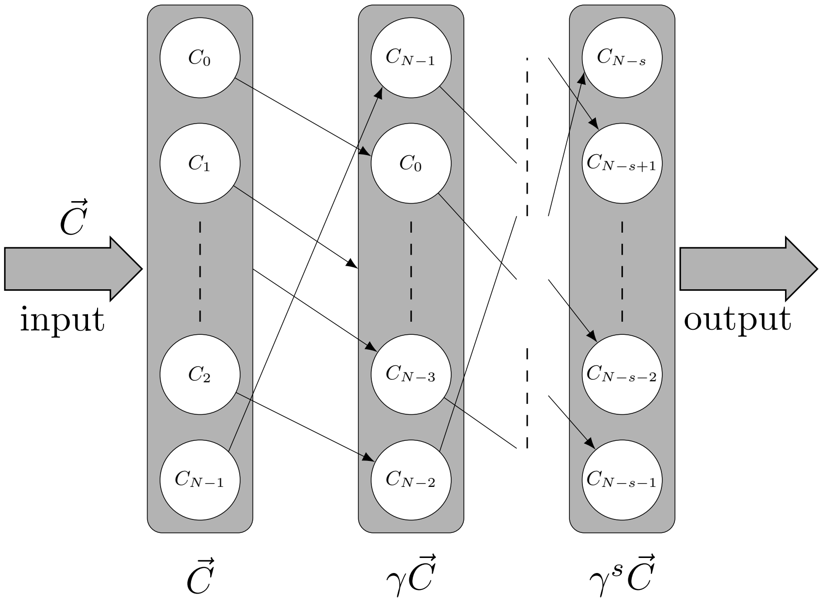

Figure 1. Schematic of classical cyclic code vector. Here, according to s times of cyclic-shift operator γ, the classical code $\vec{C}$ as an input code can generate its cyclic vectors, $\gamma\vec{C}, \gamma^2\vec{C}, \ldots, \gamma^s\vec{C}$. Here, s is an integer number, so that the generated vectors belong to a quotient ring.

More generally, if there is a smallest positive integer ℓ such that

$$\gamma^{\ell}(c) = (c_{N-\ell}, c_{N-\ell+1}, \ldots, c_{N-1}, c_0, \ldots, c_{N-\ell-1}), \tag{9}$$

also belongs to C, the linear code C is called a quasi-cyclic code of index ℓ. Furthermore, given $C^{\perp} = \{\vec{v} \in F_2^N | \langle \vec{u}, \vec{v} \rangle = 0, \forall \vec{u} \in C\}$ is called its dual code. If $C \subseteq C^{\perp}$, the code C is self-orthogonal. In quantum theory, if any quantum code word $|c\rangle \in \mathcal{P}_N$ is still a quantum code state in this space after several cyclic shifts, the quantum codes will be correspondingly called quasi-cyclic quantum codes.

In mathematical theory, a cyclic code is an ideal in the quotient ring R; hence, it can be gained by a single polynomial. Here, the quotient ring is defined as $R = F_2[x]/\langle x^n - 1 \rangle$ for a prime number n. It is an isomorphic $\gamma : F_2^n \to R$ between the ring formed by all $n \times n$ circulant matrices and the ring R formed by the polynomial $x^n - 1$. As a consequence, the circulant matrix corresponds to the following polynomial

$$a = a_0 + a_1 x + \ldots + a_{n-1} x^{n-1}. \tag{10}$$

Under the isomorphism γ, it has $\gamma\vec{a} = a_{n-1} + a_0 x + \ldots + a_{n-2}x^{n-1}$, for $\vec{a} = (a_0, a_1, \ldots, a_{n-1}) \in F_2^n$. If we define $K^n = I$, such as

$$K = \begin{pmatrix} 0 & 0 & \ldots & 1 \\ 1 & 0 & \ldots & 0 \\ 0 & 1 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & 0 \end{pmatrix} \tag{11}$$

is a $n \times n$ matrix representing the right cyclic shift by one position, it is readily seen that a circulant matrix A over F_2 can be represented in the form

$$A = a_0 I + a_1 K + \ldots + a_{n-1}K^{n-1}, \tag{12}$$

where I is the $n \times n$ identity matrix. More generally, the $n \times n$ matrix $K = (k_{ij})$ may be taken as

$$k_{ij} = \begin{cases} 1 & \text{if } i = (j+h) \bmod n \\ 0 & \text{otherwise,} \end{cases} \tag{13}$$

where, h is an integer for $1 \leq h \leq n-1$. It is obvious that the matrix K can form an Abelian operator group $\mathcal{K} = \{I = K^0, K^1, \ldots, K^{n-1}\}$. The index i of $K^i \in \mathcal{K}$ is an exponent of matrix K, which is called a *basic matrix*. More generally, the matrix G_e formed by the index of the basic matrix is defined as an *exponent matrix*. On the basis of the two matrices, we also introduce a *circulant permutation matrix* Q obtained by an operation "$\wedge$" between the matrices K and G_e as

$$\begin{aligned} Q &= (K^{a_{ij}})_{mn \times nn} = K \wedge G_e \\ &= \begin{pmatrix} K^{a_{11}} & K^{a_{12}} & \ldots & K^{a_{1n}} \\ K^{a_{21}} & K^{a_{22}} & \ldots & K^{a_{2n}} \\ \vdots & \vdots & \ddots & \vdots \\ K^{a_{m1}} & K^{a_{m2}} & \ldots & K^{a_{mn}} \end{pmatrix}, \end{aligned} \tag{14}$$

where $a_{ij}, 1 \leq i \leq m, 1 \leq j \leq n$, is the exponent of the basic matrix K, and

$$G_e = (a_{ij})_{m \times n} = \begin{pmatrix} a_{11} & a_{12} & \ldots & a_{1n} \\ a_{21} & a_{22} & \ldots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \ldots & a_{mn} \end{pmatrix}. \tag{15}$$

It is easily seen that Q is essentially an array of cyclic-shift operators.

On the other hand, a class of matrix $J_{n \times n} = (a_{ij})_{n \times n}$ is called a *jacket matrix*, which could lead to a simple encoding algorithm [28,29], if it satisfies

$$a_{ij}^{-1} = \begin{cases} (a_{ij})/a & \text{if } a_{ij} \neq 0, \\ 0 & a_{ij} = 0, \end{cases} \tag{16}$$

i.e., $J_{n \times n}^{-1} = (a_{ij}^{-1})^T / a$, where a is the normalized constant. It is obvious that Pauli matrices and the Hadamard matrix belong to a jacket matrix.

In addition, we also define an operation '$\hat{\otimes}$' between the Jacket matrix J and the cyclic-shift operator vector $\vec{K} = (I, K, \ldots, K^{n-1})$ as

$$J \hat{\otimes} \vec{K} = J \hat{\otimes} (I, K, \ldots, K^{n-1}) = (J, JK, \ldots, JK^{n-1}). \tag{17}$$

Then, on the basis of the jacket matrix J and the circulant permutation matrix Q, we design a matrix G with length $N = n(n-1)$ constructed by the following circuit

$$\begin{aligned} G &= J\hat{\otimes}(K \wedge G_e) = J\hat{\otimes}Q \\ &= J\hat{\otimes}\begin{pmatrix} K^{a_{11}} & K^{a_{12}} & \dots & K^{a_{1n}} \\ K^{a_{21}} & K^{a_{22}} & \dots & K^{a_{2n}} \\ \vdots & \vdots & \ddots & \vdots \\ K^{a_{m1}} & K^{a_{m2}} & \dots & K^{a_{mn}} \end{pmatrix} \\ &= \begin{pmatrix} JK^{a_{11}} & JK^{a_{12}} & \dots & JK^{a_{1n}} \\ JK^{a_{21}} & JK^{a_{22}} & \dots & JK^{a_{2n}} \\ \vdots & \vdots & \ddots & \vdots \\ JK^{a_{m1}} & JK^{a_{m2}} & \dots & JK^{a_{mn}} \end{pmatrix}, \end{aligned} \tag{18}$$

where, $G_e = (a_{ij})_{mn}$ is the exponent matrix of the basic matrix K in Equation (13).

4. Long-Length Quantum Coding via Quasi-Cyclic Jacket Matrix

Based on the above mathematical definitions, we mainly consider how to obtain the exponent matrix G_e and the jacket matrix J as follows.

4.1. The Construction Method of Jacket Matrix

Firstly, the construction manner of the low-parity jacket matrix J in Equation (18) is investigated. According to their definition, it is obvious that the Hadamard matrix and the Pauli matrices are special jacket matrices. Consider the 2-order Hadamard matrix

$$H_2 = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \tag{19}$$

as the smallest basic matrix. A binary n_1-size Hadamard matrix $H = (h_{ij})_{n_1 \times n_1}$ for $n_1 = 2^m$, $m \geq 3$, under mapping: $1 \to 1, -1 \to 0$, may be obtained with the following recursive relation

$$H_{n_1} = H_{n_1/2} \otimes \mathrm{H}_2. \tag{20}$$

Generally, on the basis of the tensor product, the $n_1 = 2^m$-order jacket matrix can be shown as

$$J_{n_1} = \prod_{i=1}^{m} I_{2^{m-i}} \otimes J_2 \otimes I_{2^{i-1}}, \tag{21}$$

where J_2 may also be chosen as Pauli matrices except for the Hadamard matrix,

$$\sigma_0 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad \text{or} \quad \sigma_1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}. \tag{22}$$

Correspondingly, one obtains

$$J_{n_1} J_{n_1}^T = (\prod_{i=3}^{m} I_{2^{m-i}} \otimes (J_2 J_2^T) \otimes I_{2^{i-1}}) \otimes (J_{2^3}^T J_{2^3}). \tag{23}$$

Similarly, we also consider a jacket matrix with size $n_2 = 3^m$ based on the fundamental matrix J_3 as follows: Denoting a 1×1 Jacket matrix as $J_1 = 1$, the direct sum J_3 may be written as

$$J_3 = J_1 \oplus J_2 = \begin{pmatrix} 1 & 0 \\ 0 & J_2 \end{pmatrix}. \tag{24}$$

It is easy to check that

$$J_3 J_3^{-1} \quad = \quad \begin{pmatrix} 1 & O \\ O & J_2 \end{pmatrix} \begin{pmatrix} 1 & O \\ O^T & J_2^{-1} \end{pmatrix} = I_3, \tag{25}$$

Then, one may obtain

$$J_{n_2} = J_{n_2/3} \otimes J_3, \tag{26}$$

for $n_2 = 3^m, m \in \{1, 2, \cdots\}$.

In terms of number theory, any finite prime number n may be decomposed into a Kronecker product and the direct sum of 2 and 3 resort to the following form

$$n = 2^{n_1'} + 2^{n_2'} + \cdots + 2^{n_r'} + 3^{m_1'} + \cdots + 3^{m_s'}, \tag{27}$$

where n_μ', m_ν' are integers for $1 \leq \mu \leq r, 1 \leq \nu \leq s$. Therefore, to obtain any prime size of quantum code, we first construct generator matrices with size 2^m and 3^m. In view of the recursive relation, the jacket matrix J_n of large-scale order and its transpose matrix J_n^T satisfy

$$\begin{aligned} J_n J_n^T = {} & (J_2^{\otimes n_1'} J_2^{\otimes n_1' T}) \oplus \cdots (J_2^{\otimes n_r'} J_2^{\otimes n_1' T}) \oplus (J_3^{\otimes m_1'} J_3^{\otimes m_1' T}) \\ & \oplus \cdots (J_3^{\otimes m_s'} J_3^{\otimes m_s' T}). \end{aligned} \tag{28}$$

It is obvious that the weight of the gained jacket matrix J_{n_2} is equal to 1 when it involves J_2 as one of the Pauli matrices. For clarity, some decomposition methods of the jacket matrix are presented based on 2-order and 3-order *fundamental jacket matrices* J_2 (Pauli matrices or the Hadamard matrix) and J_3 in Table 1.

Table 1. The construction of jacket matrices.

Jacket Matrix	Decompositions of Prime Number
J_5	$J_2 \oplus J_3$
J_7	$J_2^{\otimes 2} \oplus J_3$
J_{29}	$J_3^{\otimes 3} \oplus J_2 = J_2^{\otimes 4} \oplus J_{13}$
J_{31}	$J_3^{\otimes 3} \oplus J_2^{\otimes 2} = J_2^{\otimes 3} \oplus J_{23}$
J_{37}	$J_2^{\otimes 5} \oplus J_5 = J_2^{\otimes 4} \oplus J_3 \otimes J_7$

As can be seen from the above table, any large-scale jacket matrix of prime order can be decomposed by these two kinds of matrices.

Example 1. *Take parameter $n = 59$ that is decomposed with a Kronecker product and the direct sum of the fundamental Jacket matrices, then the concatenated matrix may be expressed as different methods*

$$\begin{aligned} J_{59} = J_2^{\otimes 5} \oplus J_3^{\otimes 3} = J_{32} \oplus J_{27} = \left(\begin{array}{c|c} J_{32} & O \\ \hline O & J_{27} \end{array} \right) \\ = J_7^{\otimes 2} \oplus J_{10} = J_{49} \oplus J_{10} = \left(\begin{array}{c|c} J_{49} & O \\ \hline O & J_{10} \end{array} \right) \end{aligned} \tag{29}$$

Therefore, the same jacket matrix may be described with different forms in the light of its decomposition methods. Apparently, both a jacket matrix J and a K^i matrix have row weight 1 with any positive integer $1 \leq i \leq l$, so that the row weight of G is exactly $n - 1$, for a given prime $n > 2$, where the block length of the constructed code is $N = n(n - 1)$.

4.2. The Construction Method of the Exponent Matrix

Next, based on the obtained basic matrix K, to generate the stabilizer with two circulant permutation matrices, we consider further the construction algorithm of the exponent matrix G_e in Equation (18) as follows:

Assume an Abelian group $Z_n = \{0, 1, \ldots, n-1\}$ for a prime number $n > 2$, and its subset Z_n^* obtained by non-zero elements, i.e., $Z_n^* = Z_n / \{0\}$. As n is a prime number, the size of Z_n^* is even. Hence, the set Z_n^* can be divided on average into two subsets

$Z_{l_1}^* = \{2k+1, 0 \leq k \leq l-1\}$ and $Z_{l_2}^* = \{2k, 1 \leq k \leq l\}$ of no common element. As can be seen from the group, it is clear that any $a \in Z_{l_1}^*, b \in Z_{l_2}^*, a \neq b$ and the orders of both the subsets are half of $n-1$, i.e., $|Z_{l_1}^*| = |Z_{l_2}^*| = (n-1)/2 = l$. Denote all the elements in $Z_{n_i}^*$ to be arranged as the first row vector $\vec{e}_{ij}$ of an obtained matrix E_i, where $i = 1, 2$ is the ith subset and $j = 1, 2, \ldots,$ and l is the jth permutation, so the number of the maximum sort order is l.

To the first vector $\vec{e}_{11} = (e_{11}, e_{12}, \ldots, e_{1l})$, we take the described anticlockwise (or clockwise) cyclic shift K in (13), i.e., $\vec{e}_{11}K = (e_{1l}, e_{11}, \ldots, e_{1,l-1})$. After $l-1$ times of operation K to vector $\vec{e}_{11}$, l vectors may be generated as the matrix E_1. With a similar rule, we take the sequential arrangement $\vec{e}_{21} = (e_{21}, e_{22}, \ldots, e_{2l})$ as the first vector of another matrix E_2. It is required that $\vec{e}_{21}$ is a different sequence from $\vec{e}_{11}$. Subsequently, the elements in the two subsets can be constructed as two exponent block matrices E_1 and E_2. The exponent matrix is generated by combining them. Namely, define an operator '$\odot$' for combining the two sub-matrices as

$$G_e = E_1 \odot E_2 = (E_1 | E_2) = \left(\begin{array}{c|c} \vec{e}_{11} & \vec{e}_{21} \\ \vec{e}_{12} & \vec{e}_{22} \\ \vdots & \vdots \\ \vec{e}_{1l} & \vec{e}_{2l} \end{array} \right). \tag{30}$$

Here, E_i is an exponent sub-matrix for obtaining the exponent matrix G_e, and $\vec{e}_i$ is the row vector of the exponent sub-matrix E_i, $i = 1, 2$. On the basis of the same rule, two exponent matrices G_{e_1} and G_{e_2} may be gained from the elements in $Z_{l_1}^*$ and $Z_{l_2}^*$, respectively. After constructing the exponent matrix G_e and the basic matrix K, the circulant permutation Q may be obtained. By combining it with the n-order jacket matrix, the $nl \times n(n-1)$ generator matrix G is finally obtained.

Furthermore, if a greater code rate is required, we also apply the described method again to add the rows. This purpose will be achieved to make the first vector begin with a different cyclic-shifting permutation vector with opposite direction, i.e., clockwise (or anticlockwise). As a result, it will obtain a $n(n-1) \times n(n-1)$ generator matrix G as

$$G_e = \left(\begin{array}{c|c} E_1 & E_2 \\ E_1' & E_2' \end{array} \right), \tag{31}$$

where E_1' and E_2' are newly obtained matrices with the similar method. It is obvious that the row weight of the obtained matrix is equal to $n-1$. In fact, if we wish to further reduce the weight of the row matrix, a zero block matrix can be used during the design process. If we denote "τ_0" as the exponent of the $n \times n$ zero matrix, just taking "τ_0" instead of $1 \leq t \leq l-1$ elements in $Z_{l_1}^*$ and $Z_{l_2}^*$, the row weight will reduce $2t$ in the obtained matrix.

The whole physical circuit of the construction process can be described as in Figure 2. In the process, a prime number n will be divided into two branches. According to the designed approach, two matrices G_{e_1} and G_{e_2} that generated similarly to G_e will be obtained by operating a cycle-shift operation γ on two groups of vectors $\vec{e}_{a_i}$ and $\vec{e}_{b_j}$ $(i, j = 1, 2)$. Finally, we take the obtained matrix G_i (similar matrix G in Equation (18)) as the parity-check matrix H_i of a classical QC code by resort to the algorithm.

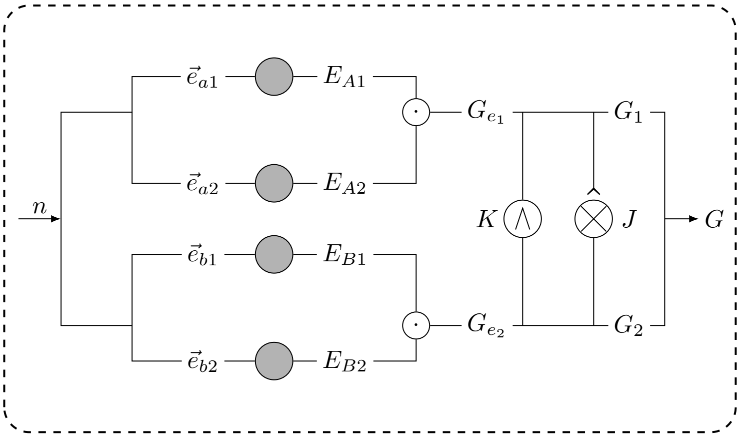

Figure 2. The circuit of the constructing generator matrix G. After taking the proper prime number n, two exponent matrices G_{e_1} and G_{e_2} may be obtained by resort to the operator '$\odot$' on matrices E_{A_i} and E_{B_j} using a cyclic shift to vectors $\vec{e}_{a_i}$ and $\vec{e}_{b_j}$, respectively. Correspondingly, the quasi-cyclic matrix H of the quantum codes can be obtained based on the two classical quasi-cyclic matrices G_1 and G_2, which are yielded by the same jacket matrix and two circulant permutation matrices by resort to the operation '$\hat{\otimes}$'.

However, two conditions of coding construction should be satisfied for the generated matrix G. On the one hand, any two rows in G should be orthogonal, i.e., it is self-orthogonal. In fact, according to the proposed construction algorithm, take a cyclic-shift operator σ in the no-unit operator group $\mathcal{K}/\{I\}$, where the group $\mathcal{K}$ is formed by K introduced before. Without losing generality, denote $\vec{v}_\sigma = (\sigma^{i_1}, \sigma^{i_2}, \ldots, \sigma^{i_l})$ and $\vec{v'}_\sigma = (\sigma^{i'_1}, \sigma^{i'_2}, \ldots, \sigma^{i'_l})$ two row operator vectors in the circulant permutation matrix Q, $1 \leq i, i' \leq l = (n-1)/2$. Namely, exponent vectors $\vec{e}_i = (i_1, i_2, \ldots, i_l)$ and $\vec{e'}_i = (i'_1, i'_2, \ldots, i'_l)$ are both in the same matrix E_1 or E_2. It is obvious that, if any binary row vector $\vec{c}$ in the jacket matrix J is both operated by $\vec{v}_\sigma$ and $\vec{v'}_\sigma$ with operation $\hat{\otimes}$, their inner product is equal to 0 or 1, i.e., $(\vec{c}\hat{\otimes}\vec{v}_\sigma) \cdot (\vec{c}\hat{\otimes}\vec{v'}_\sigma)^T = 0$ or 1. Therefore, if any two row vectors $\vec{x}$ and $\vec{y}$ in matrix J are operated by the concatenated vector $\vec{v} = (\vec{v}_\sigma|\vec{v'}_\sigma)$ with size $2l$ and another vector $\gamma\vec{v}$, their inner product will always be 0 for mode 2 with one-to-one mapping, i.e., $(\vec{x}\hat{\otimes}\vec{v}) \cdot (\vec{x}\hat{\otimes}\gamma\vec{v})^T = 0$; here, γ is a cyclic-shift operator. This means that G is a self-orthogonal matrix.

On the other hand, it should also be proved that any codewords c_1 from $C_2^\perp$ are orthogonal to all codewords c_2 from $C_1^\perp$, i.e., $G_1 \cdot G_2^\perp = \mathbf{0}$. In fact, for the basic matrix K, assume its cyclic-shift time of operating γ on it in $Z_{l_1}^*$ and $Z_{l_2}^*$, respectively. According to the described algorithm, the cyclic-shift steps in the two sets are both 2, i.e., with the same operation γ^2. Therefore, any two matrices K_1 and K_2 of which exponents from $Z_{l_1}^*$ and $Z_{l_2}^*$, can be described as forms like Equation (12), i.e.,

$$\begin{aligned} K_1 &= a_1K + a_3K^3 + \ldots + a_{n-1}K^{n-1}, \\ K_2 &= a_2K^2 + a_4K^4 + \ldots + a_nK^n. \end{aligned} \tag{32}$$

It is obvious that K_1 and K_2 are linearly independent, i.e., the two generated Abelian spaces are orthogonal.

As a result, if we take G_{e_1} and G_{e_2} as the exponent matrices constructed by Equation (18), we can obtain two N-length matrices G_1 and G_2. Then, the $N - k_i$ rows of the two matrices are randomly chosen as the parity-check matrix H_i of classical code $C_i[N, k_i, d_i]$, $(i = 1, 2)$, so a generator matrix $\mathcal{G}_c$ of a QC quantum code $\mathcal{C}[[N, k_1 + k_2 - N, \min\{d_1, d_2\}]]$ of the form of Equation (5) is obtained.

Example 2. *Let prime number* $n = 11, Z_1^* = \{1, 2, \cdots, 10\}$*, so* $l = (n-1)/2 = 5$*, and sets* $Z_{l_1}^* = \{1, 3, 5, 7, 9\}$, $Z_{l_2}^* = \{2, 4, 6, 8, 10\}$. *Take* $\vec{e}_{11} = (1, 3, 5, 7, 9)$ *and* $\vec{e}_{21} = (3, 5, 7, 9, 1)$ *as the* *first vectors to construct the sub-matrices E_{A1} and E_{A2}, respectively, so that G_{e1} is obtained by combining the two sub-matrices. Similarly, also take $\vec{e}_{22} = (2,4,6,8,10)$ and $\vec{e}_{22} = (4,6,8,10,2)$ as the first vectors, then G_{e2} will be constructed by coupling E_{B1} and E_{B2}. Correspondingly, in terms of the basic matrix K with parameter $h = 1$ in Equation (13), the exponent matrices G_{e_1} and G_{e_2} may be generated as*

$$\begin{aligned} G_{e_1} &= \left(\; E_{A1} \odot E_{A2} \; \right) = \left(\; E_{A1} \;\middle|\; E_{A2} \; \right) \\ &= \left(\begin{array}{ccccc|ccccc} 1 & 3 & 5 & 7 & 9 & 3 & 5 & 7 & 9 & 1 \\ 9 & 1 & 3 & 5 & 7 & 1 & 3 & 5 & 7 & 9 \\ 7 & 9 & 1 & 3 & 5 & 9 & 1 & 3 & 5 & 7 \\ 5 & 7 & 9 & 1 & 3 & 7 & 9 & 1 & 3 & 5 \\ 3 & 5 & 7 & 9 & 1 & 5 & 7 & 9 & 1 & 3 \end{array} \right). \end{aligned} \tag{33}$$

Similarly, by making use of a similar method, matrix G_{e_2} may be obtained as follows:

$$\begin{aligned} G_{e_2} &= E_{B1} \odot E_{B2} = \left(\; E_{B1} \;\middle|\; E_{B2} \; \right) \\ &= \left(\begin{array}{ccccc|ccccc} 2 & 4 & 6 & 8 & 10 & 4 & 6 & 8 & 10 & 2 \\ 10 & 2 & 4 & 6 & 8 & 2 & 4 & 6 & 8 & 10 \\ 8 & 10 & 2 & 4 & 6 & 10 & 2 & 4 & 6 & 8 \\ 6 & 8 & 10 & 2 & 4 & 8 & 10 & 2 & 4 & 6 \\ 4 & 6 & 8 & 10 & 2 & 6 & 8 & 10 & 2 & 4 \end{array} \right). \end{aligned} \tag{34}$$

We also take the 11 by 11 basic-matrix $J_{11} = J_6 \oplus J_5$. Then, the matrices G_1 and G_2 with size 55×110 are obtained by combining the two exponent matrices and the basic-matrix. It is obvious that the obtained matrix can encode 55 information bits at most. To increase the coding rate k/N, we can also add the rows E'_{A1} and E'_{A2} (E'_{B1} and E'_{B2}) of the exponent matrix in terms of the algorithm described before, such as

$$(E'_{A1}|E'_{A2}) = \left(\begin{array}{ccccc|ccccc} 5 & 7 & 9 & 1 & 3 & 3 & 5 & 7 & 9 & 1 \\ 7 & 9 & 1 & 3 & 5 & 5 & 7 & 9 & 1 & 3 \\ 9 & 1 & 3 & 5 & 7 & 7 & 9 & 1 & 3 & 5 \\ 1 & 3 & 5 & 7 & 9 & 9 & 1 & 3 & 5 & 7 \\ 3 & 5 & 7 & 9 & 1 & 1 & 3 & 5 & 7 & 9 \end{array} \right) \tag{35}$$

and

$$\begin{aligned} &(E'_{B1}|E'_{B2}) \\ &= \left(\begin{array}{ccccc|ccccc} 6 & 8 & 10 & 2 & 4 & 4 & 6 & 8 & 10 & 2 \\ 8 & 10 & 2 & 4 & 6 & 6 & 8 & 10 & 2 & 4 \\ 10 & 2 & 4 & 6 & 8 & 8 & 10 & 2 & 4 & 6 \\ 2 & 4 & 6 & 8 & 10 & 10 & 2 & 4 & 6 & 8 \\ 4 & 6 & 8 & 10 & 2 & 2 & 4 & 6 & 8 & 10 \end{array} \right). \end{aligned} \tag{36}$$

As the result, the exponent matrices G_{e_1} and G_{e_2} can also be shown as the following 110×110 matrix

$$G_{e_1} = \left(\begin{array}{c|c} E_{A1} & E_{A2} \\ E'_{A1} & E'_{A2} \end{array} \right) \text{ and } G_{e_2} = \left(\begin{array}{c|c} E_{B1} & E_{B2} \\ E'_{B1} & E'_{B2} \end{array} \right). \tag{37}$$

Therefore, the two matrices G_1 and G_2 are used to form the generator matrix $\mathcal{G}_c$ in Equation (5).

For example, if we wish to obtain a quantum code with a coding rate of 0.4, by applying the proposed approach, the two classical codes $C_1[110, 80, 12]$ and $C_2[110, 74, 14]$ can be constructed, respectively. Therefore, according to the construction method of CSS-type, a quantum code $\mathcal{C}[110, 44, 12]$ is obtained. With greatly increasing code length of QC codes, some methods of computing the minimum distance are required [37,38].

In the following, we use different parameters to analysis the bit error ratio (BER) of the presented QC codes with a jacket matrix of 0.5 code rate. We first consider the coding performance of one of the pair of classical codes C_1 and C_2 in Figure 3. Here, if different prime numbers are taken $n = 29, 23, 19$ in the construction method, their corresponding code lengths are $N = n(n-1) = 812, 506, 342$, respectively. With a 0.5 coding rate, we compare their corresponding QC codes. In the signal-to-noise ratio (SNR) range, it is shown that the two shorter lengths of the taken QC quantum codes show better performance from 0.2 to 0.4 dB than the longer one. However, owing to the close cycles and their bit-flip error correction capability, a greater length of code shows its superiority with increasing length.

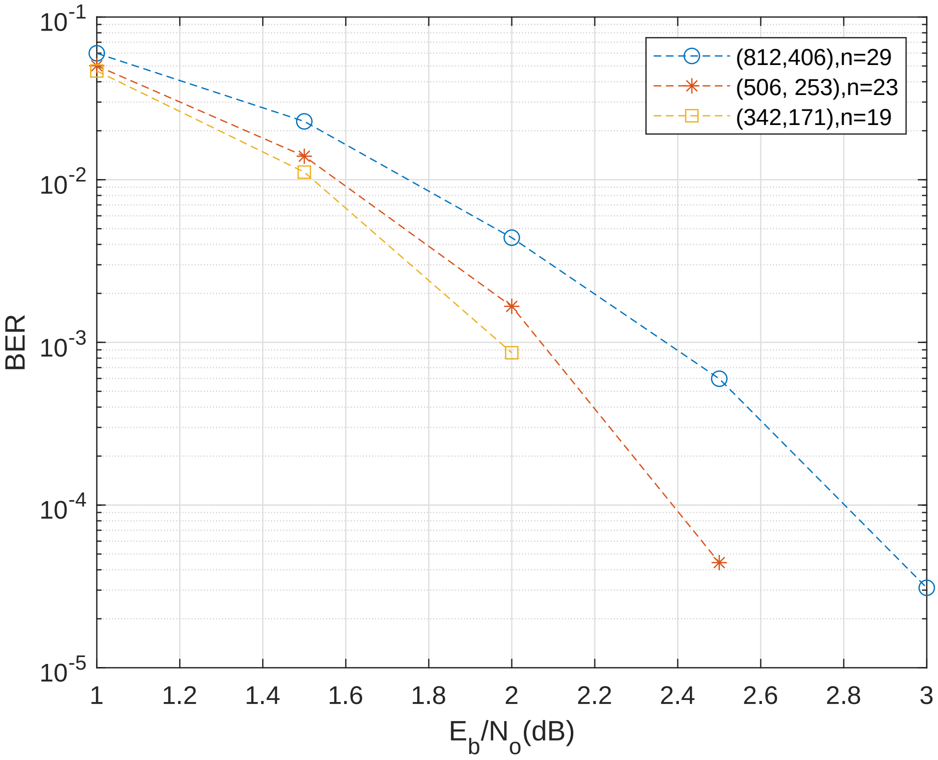

Figure 3. BER performance comparison of code rate $R = 0.5$ classical QC code with $N = 812$, $N = 506$ and $N = 342$. To achieve a greater length of cycle, the decomposition numbers of prime n more than 4 (to avoid $4-$cycles) are used in our construction method. Assuming the input is a Gaussian white noise channel model, we take the prime numbers $n = 29, 23, 19$ and the maximum iterations of operator γ as $n-1$ to construct the jacket matrixes; then the code lengths are $812, 506, 342$, respectively.

Furthermore, we consider the probability of three classes of Pauli errors in the communication channel. Assuming the channel is viewed as a possibly more realistic depolarizing channel, it can independently generate a list of X bit-flip errors, Z phase-flip errors and Y errors which are a combination of bit and phase flip with equal probability $f/3$ for a total probability f. Hence, the marginal flip probability distributed identically is $f_m = 2f/3$. Owing to the construction methods of all CSS-type quantum codes obtained by the two classical codes C_1 and C_2, the X-and-Z errors may be separately decoded and corrected with a standard classical correction algorithm [39]. In Figure 4, we compare the proposed QC quantum codes with the conventional quantum error-correction codes proposed in [22]. Here, we take, respectively, the quantum coding rate of these codes as 0.50 and 0.48. The proposed codes of lengths $N = 7832$ and $N = 5256$ are generated by jacket matrices with parameters $n = 89$ and $n = 73$, respectively. According to the construction method in [22], two codes with similar lengths $N = 6806$ and $N = 4970$ are compared. Because the pair of classical codes C_1 and C_2 are isomorphic, the average decoding performances are similar in simulation. Therefore, we just focus on analysis of the single classical code in Figure 4. According to the presented coding method, the obtained generator matrix has sparse properties in terms of the character of the jacket block matrix; hence, a larger minimum distance for the generator matrix can be obtained. As a result, the bit error rate of our gained codes is higher than for the conventional codes, although the floor of no error for both codes is down to about 10^{-6}. However, with increasing f_m, their BERs tend to the same level. Therefore, the proposed QC quantum codes show better performance for the coding of large-scale data.

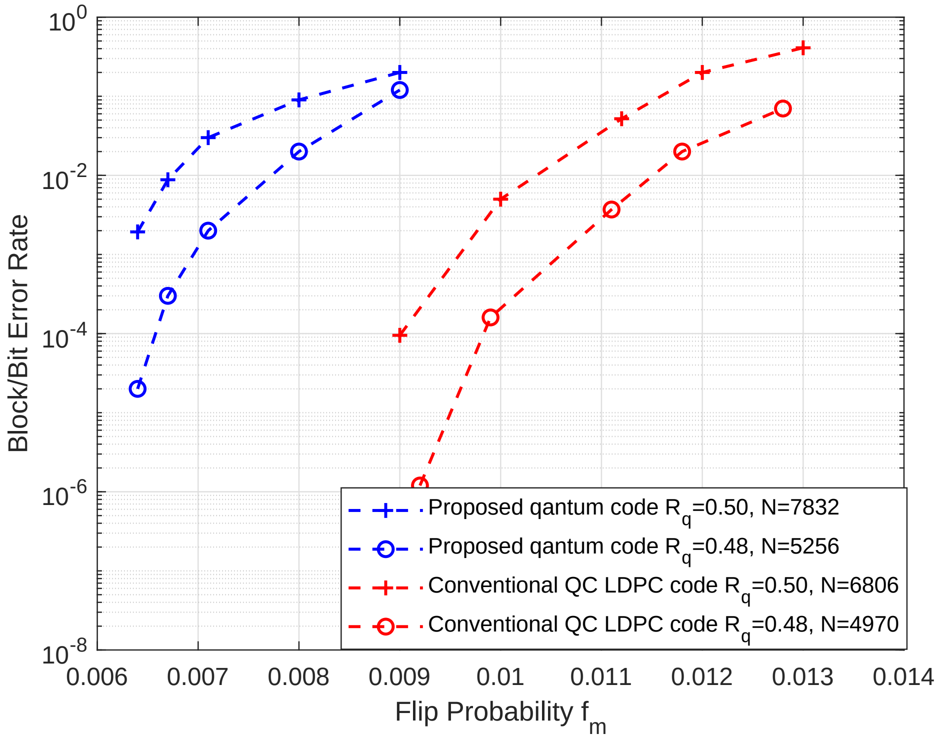

Figure 4. Comparison of decoding error rate of the proposed QC quantum codes (bule) and conventional QC LDPC CSS codes (red). Here, the proposed QC quantum codes with lengths $N = 7832$ and $N = 5256$ are generated by $n = 89$ and $n = 73$ for $n - 1$ times of iterations on a binary symmetric channel, and the conventional QC LDPC codes are obtained with similar lengths with rates R_q = 0.50, 0.48, respectively. Based on the equal error rate of one of the classical pair codes C_1 and C_2, the bit error rate of the marginal flip probability f_m of X and Z errors is shown.

Assume J_ι is a $\iota \times \iota$ jacket in Equation (23) constructed with Pauli matrices σ_0 and σ_1 (not including the Hadamard matrix) in the described algorithm. To analysis the computation complexity, we first consider the 0 and 1 densities in the binary matrix denoted by

$$d_0 = \frac{\iota_1}{\iota_0 + \iota_1}, \tag{38}$$

where ι_0 and ι_1 are the 0 and 1 numbers respectively. Therefore, the density d_0 of the jacket matrix equals $1/\iota$ and becomes lower with increasing length ι. According to the described algorithm, there are two operators, i.e., the Kronecker-product operator '$\otimes$' and the direct-sum operator '$\oplus$', involved. It is obvious that the calculation complexity of the direct sum is $O(1)$ that is less than the Kronecker-product operator's $O(\iota)$.

Therefore, the low complexity of the coding method should obey the following rules:

(1) the length of the taken matrices between the Kronecker-product operator should be larger;

(2) to obtain the long QC quantum code, the number in the decomposition method should include more direct-sum operators than Kronecker-product operators. As a result, the constructed quantum codes show good performance for low density code.

In addition, we consider the low-density properties of the constructed matrix. After first being proposed by Gallager, LDPC codes were rediscovered several decades later [31,32]. Classical LDPC codes have good performance, which approach the channel capacity in the limit of large block size. There has been considerable interest in finding quantum versions of these codes [33–35]. However, quantum LDPC codes are far less studied than their classical counterparts. In spite of its good performance, one main obstacle is how to obtain a highly efficient algorithm for iterative coding. In our paper, we have tentatively presented a method of constructing quantum LDPC codes and the results further enrich the theoretical achievements in this field. To ensure that the Tanner graph of the LDPC codes is free of 4-cycles, the girth may be at least 6 because the cycle of short length will reduce the performance of the LDPC codes. To meet the requirement, we just take the parameter n of the basic jacket matrix to be no less than 7, i.e., $l \geq 3$, in our construction algorithm.

In classical codes, a bipartite Tanner graph consists of check nodes and variable nodes [40]. The variable and check node degrees are denoted as η and ρ, respectively, which correspond to the row and column minimum distances of the sparse parity-check matrix H. Assume the length of the shortest cycle on the graph, and that the number of independent iterations are referred to as the girth g and T, respectively [41,42]. Correspondingly, a useful iteration gain is generally bounded by

$$T < g/4 \leq T + 1. \tag{39}$$

Obviously, the gained block length should be sufficient while the girth is given, so that the bound satisfies [42]

$$N \geq 1 + \sum_{i=2}^{x+1} \eta(\eta-1)^{i-2}(\rho-1)^{i-1} \tag{40}$$

for a specific girth $g = 4x + 2$, where x is an integer. According to the bound (40), the relation of the code length N and the weights η and ρ is shown in Figure 5. Here, as x is taken as 2, 3 and 4 with hypothesis $\eta = \rho$, then the cycle-10, cycle-14 and cycle-18 are gained, respectively.

In this figure, it is shown that the length increases exponentially with the cycle length and its minimum weight η. Provided that the required cycle is determined, i.e., the cycle is bigger than a 4-cycle, for balancing the weights η, ρ and the code length N, it is necessary to consider the fundamental construction of the jacket matrix. In terms of the characters of the proposed method, the minimum distance of the obtained codes will grow linearly with increase in the parameter n. Thereby, we can apply the taken prime number to generate jacket matrices to adjust the relation between η and N. The main rule for reducing the density and increasing the cycle of the parity check matrix H is that n in the decomposition method should be larger. For example, if we decompose the length $n = 156$ of the QC code as $J_{156} = J_{13} \otimes J_3 \otimes J_2^{\otimes 2}$ with the most basic method, the cycles will be smaller. However, if we take another decomposition method $J_{156} = J_7 \otimes J_{11} \oplus J_{43} \oplus J_{23} \oplus J_{13}$, the cycle length is obviously increased, and, hence, the obtained quantum codes have the advantage of simple implementation of an iterative decoder and low-complexity encoding.

According to information coding theory, the sparse parity-check matrix H of code $[N, k]$ for decoding can be gained from the generator matrix. Based on the character of our constructed matrix with prime number n, it is easy to check that the variable and check node degrees in H or the corresponding bipartite Tanner graph at most are $\eta = n - 1$ and $\rho = [k/n] + 1$, where $[\cdot]$ is an integral function. For Example 2, the row and column weights of the parity-check matrix for constructed codes $[110, 80]$ are $\eta = 9$ and $\rho = 8$, respectively. It is obvious that the bound can be met with its 6-cycle for $x = 1$.

Furthermore, according to the family of proposed codes from the circulant permutation matrix in Equation (14), the obtained quantum QC code of dimension $O(N)$ contains its upper bound on the minimum distance $O(N/\log N)$ as the code length $N \to \infty$. This means that, the adopted parameter n in the proposed coding method should correspondingly approach

$$n \approx \sqrt{\sum_{i=2}^{x+1} \eta(\eta-1)^{2(i-2)}}, \quad 1 < x \in \mathbb{Z} \tag{41}$$

with the long-code length, where $\mathbb{Z}$ is an integer set.

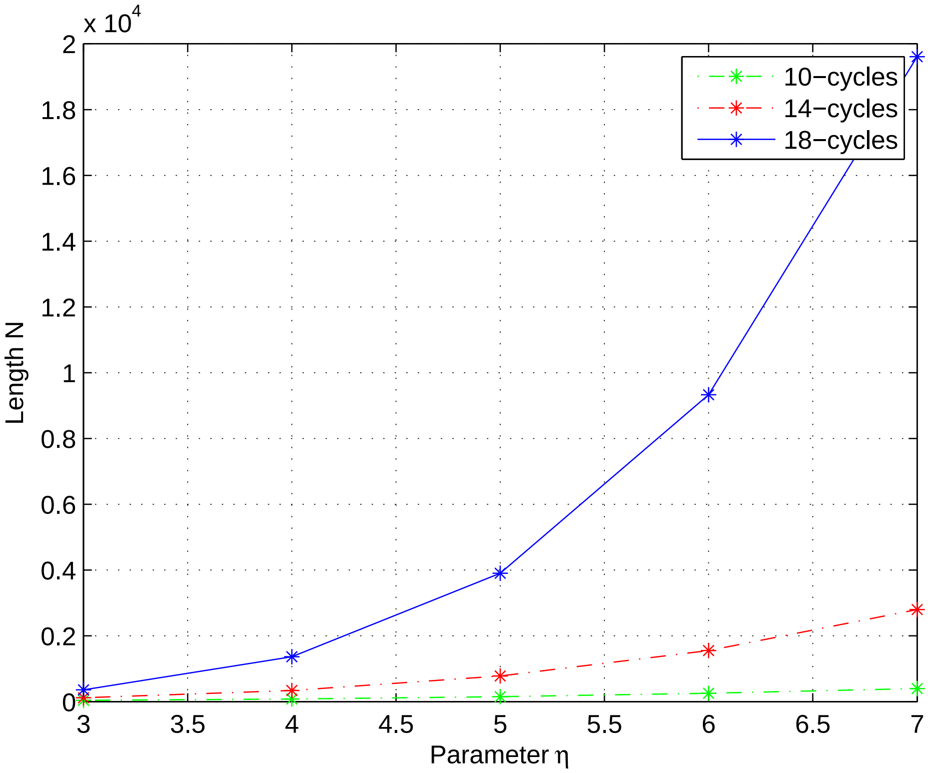

Figure 5. The performance of required code length N with different parameters. Here, to reflect the relation of several parameters, it takes $x = 2, 3, 4$ so that 10-cycles, 14-cycles and 18-cycles are obtained with hypothesis $\eta = \rho$. To satisfy the bound, the prime number n in the proposed method should also meet the condition of length N and cycles.

In practical applications, quantum codes with long length may potentially be used in some future quantum information fields involved in large-scale data, such as quantum machine learning, which uses quantum computers and aims to enhance the power of machine learning. However, some major obstacles still limit the use of quantum hardware for practical applications of machine learning. One of the bottlenecks is that the data has to be encoded into the quantum computer in an efficient manner to generate its useful quantum kernel. Although some encoding methods have been investigated [43–45], the features are limited by the number of qubits [46]. Additionally, the inherent noise of quantum computers also influences the quality of the experimental results. Generally, rapid processing of the large datasets is vital because the operation timescale relies linearly on the size of the dataset. Therefore, to provide a useful kernel for machine learning, a good coding manner with long code length can efficiently load a high-dimensional feature vector into a quantum computer. As a consequence, the proposed coding method in this paper provides a tentative coding scheme for ML based on large-scale data.

5. Conclusions

The coding method for massive data has important applications in the field of information transmission and big-data processing. With the rapid development of quantum computing, this class of coding construction has become one of the research fields in quantum information. In this paper, based on a class of circulant permutation matrices, we presented quantum quasi-cyclic CSS codes in terms of two long-length classical codes based on block matrices. Motivated by the advantages of convenient and low-complexity implementation of an iterative decoder for proposed code, a family of jacket block matrices were applied to quantum coding construction. The presented method can efficiently construct long-length quantum QC codes based on the proposed method of an iterative coding process. If the parameter is selected appropriately, the obtained quantum codes are number 4-cycles in their generators and have good performance for LDPC codes.

Furthermore, we also analysed the marginal flip probability f_m with respect to our proposed codes; that a good performance was obtained for the bit error rate is shown in Section 4. Therefore, the obtained QC quantum codes that benefit from their semi-parallel architectures can potentially be applied to the future quantum information field in relation to massive data.

Author Contributions: Formal analysis, Investigation, Writing–original draft, Y.L.; Software, J.-Y.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the Natural Science Foundation of Shanghai (No.19ZR1420000).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data used to support the findings of this study are included within the article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Gisin, N.; Ribordy, G.; Tittel, W.; Zbinden, H. Quantum cryptography. *Rev. Mod. Phys.* **2002**, *74*, 145. [CrossRef]
2. Assche, G.V.; Cardinal, J.; Cerf, N.J. Reconciliation of a quantum distributed Gaussian key. *IEEE Trans. Inf. Theory* **2004**, *50*, 394–400.
3. Scarani, V.; Pasquinucci, H.B.; Cerf, N.J. The security of practical quantum key distribution. *Rev. Mod. Phys.* **2009**, *81*, 301. [CrossRef]
4. Guo, Y.; Wang, K.S.; Huang, D.; Jiang, X.Q. High efficiency continuous-variable quantum key distribution based on qc-ldpc codes. *Chin. Opt. Lett.* **2019**, *11*, 112701. [CrossRef]
5. Calderbank, A.R.; Rains, E.M.; Shor, P.W. Quantum Error Correction and Orthogonal Geometry. *Phys. Rev. Lett.* **1997**, *76*, 405. [CrossRef]
6. Calderbank, A.R.; Shor, P. Good quantum error-correcting codes exist. *Phys. Rev. A* **1996**, *54*, 1098. [CrossRef]
7. Steane, A. Multiple particle inference and quantum error correction. *Proc. R. Soc. A* **1996**, *452*, 2551.
8. Shor, P.W. Scheme for reducing decoherence in quantum computer memory. *Phys. Rev. A* **1995**, *52*, 2493. [CrossRef] [PubMed]
9. Steane, A.M. Error correcting codes in quantum theory. *Phys. Rev. Lett.* **1996**, *77*, 793. [CrossRef]
10. Nielsen, M.A.; Chuang, I.S. *Quantum Computation and Quantum Information*; Cambridge University Press: Cambridge, UK, 2000.
11. Rebentrost, P.; Mohseni, M.; Lloyd, S. Quantum support vector machine for big data classification. *Phys. Rev. Lett.* **2014**, *113*, 130503. [CrossRef]
12. Vandermolen, R.; Wright, D. Graph-Theoretic Approach to Quantum Error Correction. *arXiv* **2021**, arXiv:2110.08414.
13. Zhou, N.R.; Zhang, T.F.; Xie, X.W.; Wu, J.Y. Hybrid quantum-classical generative adversarial networks for image generation via learning discrete distribution. *Signal Process. Image Commun.* **2023**, *110*, 116891. [CrossRef]
14. Cao, C.; Zhang, C.; Wu, Z.; Grassl, M.; Zeng, B. Quantum variational learning for quantum error-correcting codes. *arXiv* **2022**, arXiv:2204.03560.
15. Aydin, N.; Ray-Chaudhuri, D.K. Quasi-cyclic codes over Z4 and some new binary codes. *IEEE Trans. Inf. Theory* **2002**, *48*, 2065–2069. [CrossRef]
16. Daskalov, R.; Hristov, P. New binary one-generator quasi-cyclic codes. *IEEE Trans. Inf. Theory* **2003**, *49*, 3001–3005. [CrossRef]
17. Kapshikar, U.; Kundu, S. Diagonal distance of quantum codes and hardness of the minimum distance problem. *arXiv* **2022**, arXiv:2203.04262.
18. Panteleev, P.; Kalachev, G. Quantum LDPC Codes with Almost Linear Minimum Distance. *IEEE Trans. Inf. Theory* **2022**, *1*, 68. [CrossRef]
19. Ling, S.; Sol'e, P. Good self-dual qausi-cyclic codes exist. *IEEE Trans. Inf. Theory* **2003**, *49*, 1052–1053. [CrossRef]
20. Kasami, T. A Gilbert-Varshamov bound for quasi-cyclic codes of rate1/2. *IEEE Trans. Inf. Theory* **2008**, *20*, 679. [CrossRef]

21. Loubenets, E.R. General lower and upper bounds under minimum-error quantum state discrimination. *Phys. Rev. A* **2022**, *105*, 032410. [CrossRef]
22. Hagiwara, M.; Imai, H. Quantum quasi-cyclic LDPC codes. In Proceedings of the 2007 IEEE International Symposium on Information Theory, Nice, France, 24–29 June 2007; pp. 806–810.
23. Hagiwara, M.; Kasai, K.; Imai, H.; Sakaniwa, K. Spatially Coupled Quasi-Cyclic Quantum LDPC Codes. In Proceedings of the 2011 IEEE International Symposium on Information Theory Proceedings, ISIT 2011, St. Petersburg, Russia, 31 July–5 August 2011.
24. Galindo, C.; Hernando, F.; Mastsumoto, R. Quasi-cyclic constructions of quantum codes. *Finite Fields Appl.* **2018**, *52*, 261–280. [CrossRef]
25. Ezerman, M.F.; Ling, S.; Ozkaya, B.; Sol'e, P. Good stabilizer codes from quasi-cyclic codes over F_4 and F_9. *arXiv* **2019**, arXiv:1906.04964v1.
26. Lv, J.; Li, R.; Wang, J. New binary quantum codes derived from one generator quasi-cyclic codes. *IEEE Access* **2019**, *7*, 85782–85785. [CrossRef]
27. Lv, J.; Li, R.; Wang, J. An explicit construction of quantum stabilizer codes from quasi-cyclic codes. *IEEE Commun. Lett.* **2020**, *24*, 1067–1071. [CrossRef]
28. Lee, M.H. The center weighted Hardamard transform. *IEEE Trans. Circuits Syst.* **1989**, *36*, 1247. [CrossRef]
29. Lee, M.H.; Hou, J. Fast Block Inverse Jacket Transform. *IEEE Signal Process. Lett.* **2006**, *13*, 461. [CrossRef]
30. Gallager, R.G. Low-Density Parity-Check Codes. Ph.D. Thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 1963.
31. Davey, M.C.; MacKay, D.J.C. Low density parity check codes over GF(q). *IEEE Commun. Lett.* **1998**, *2*, 165–167. [CrossRef]
32. MacKay, D.J.C. Good error-correcting codes based on very sparse matrices. *IEEE Trans. Inf. Theory* 1999, *45*, 399–432. [CrossRef]
33. Kovalev, A.A.; Pryadko, L.P. Fault tolerance of quantum low-density parity check codes with sublinear distance scaling. *Phys. Rev. A* **2013**, *87*, 020304. [CrossRef]
34. Fawzi, O.; Grospellier, A.; Leverrier, A. Constant overhead quantum fault-tolerance with quantum expander codes. In Proceedings of the 2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS), Paris, France, 7–9 October 2018; pp. 743–754.
35. Tremblay, M.A.; Delfosse, N.; Beverland, M.E. Constant-overhead quantum error correction with thin planar connectivity. *arXiv* **2021**, arXiv:2109.14609.
36. Feng, K.; Ma, Z. A finite Gilbert-Varshamov bound for pure stabilizer quantum codes. *IEEE Trans Inf. Theory* **2004**, *50*, 3323. [CrossRef]
37. Kou, Y.; Lin, S.; Fossorier, M. Low-density parity-check codes based on finite geometries: A rediscovery and new results. *IEEE Trans. Inf. Theory* **2001**, *47*, 2711–2736. [CrossRef]
38. Berrou, C.; Vaton, S. Computing the minimum distance of linear codes by the error impulse method. In Proceedings of the IEEE International Symposium on Information Theory, Taipei, Taiwan, 17–21 November 2002.
39. MacKay, D.J.C.; Mitchison, G.; McFadden, P.L. Sparse-graph codes for quantum error correction. *IEEE Trans. Inf. Theory* **2004**, *50*, 2315–2330. [CrossRef]
40. Hu, X.Y.; Eleftheriou, E.; Arnold, D.M. Regular and Irregular Progressive EdgeGrowth Tanner Graphs. *IEEE Trans. Commun.* **2005**, *51*, 386–388.
41. Tanner, R.M. A recursive approach to low complexity codes. *IEEE Trans. Inf. Theory* **1981**, *27*, 533. [CrossRef]
42. Gallager, R.G. Low-density parity-check codes. *IEEE Trans. Inf. Theory* **1962**, *8*, 21. [CrossRef]
43. Perez-Salinas, A.; Cervera-Lierta, A.; Gil-Fuster, E.; Latorre, J.I. Data re-uploading for a universal quantum classifier. *Quantum* **2020**, *4*, 226. [CrossRef]
44. Schuld, M. Quantum machine learning models are kernel methods. *arXiv* **2021**, arXiv:2101.11020.
45. Haug, T.; Self, C.N.; Kim, M.S. Large-scale quantum machine learning. *arXiv* **2021**, arXiv:2108.01039.
46. Havlcek, V.; Corcoles, A.D.; Temme, K.; Harrow, A.W.; Kandala, A.; Chow, J.M.; Gambetta, J.M. Supervised learning with quantum-enhanced feature spaces. *Nature* **2019**, *567*, 209. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.