Quantum Inf Process (2009) 8:361–378
DOI 10.1007/s11128-009-0113-1

Fast quantum codes based on Pauli block jacket matrices

Ying Guo · Jun Peng · Moon Ho Lee

Published online: 28 April 2009

Abstract Jacket matrices motivated by the center weight Hadamard matrices have played an important role in signal processing, communications, image compression, cryptography, etc. In this paper, we suggest a design approach for the Pauli block jacket matrix achieved by substituting some Pauli matrices for all elements of common matrices. Since, the well-known Pauli matrices have been widely utilized for quantum information processing, the large-order Pauli block jacket matrix that contains commutative row operations are investigated in detail. After that some special Abelian groups are elegantly generated from any independent rows of the yielded Pauli block jacket matrix. Finally, we show how the Pauli block jacket matrix can simplify the coding theory of quantum error-correction. The quantum codes we provide do not require the dual-containing constraint necessary for the standard quantum error-correction codes, thus allowing us to construct quantum codes of the large codeword length. The proposed codes can be constructed structurally by using the stabilizer formalism of Abelian groups whose generators are selected from the row operations of the Pauli block jacket matrix, and hence have advantages of being fast constructed with the asymptotically good behaviors.

Keywords Pauli block matrix · Block jacket transform · Pauli matrices · Abelian group · Quantum error-correction codes

Y. Guo (✉) · J. Peng
School of Information Science and Engineering, Central South University,
410083 Changsha, China
e-mail: sdguoying@gmail.com

M. H. Lee
Department of Information & Communication Engineering,
Chonbuk National University, Chonju 561-756, Korea

PACS 03.65.Bz · 89.70.+c

1 Introduction

The Hadamard matrix is an orthogonal matrix with highly practical values for signal sequence transforms and data processing [1–4]. Jacket matrices [5–7], which are motivated by the center weight Hadamard matrices, are class of matrices with the inverse being determined by the element-wise of matrices. Mathematically, let $A = (a_{kt})$ be a matrix, if $A^{-1} = (a_{kt}^{-1})^T$, then the matrix A is a jacket matrix, where the superscript symbol 'T' denotes the transpose and '$(\cdot)$' denotes a matrix. Especially, the interesting matrices, such as Hadamard matrices, DFT matrices, belong to the jacket matrix family [8]. Since the inverse of the jacket matrix can be easily calculated due to its elegant characteristics [9,10], it may be elegantly employed in the signal processing, encoding, image compression, etc [11–13].

If each element of the jacket matrix is replaced by a matrix, the resulting matrix is called as the block jacket matrix. Let $[A] = ([a]_{kt})$, if $[A]^{-1} = ([a]_{kt}^{-1})^T$, then the resulting block matrix $[A]$ is the block jacket matrix, where the symbol '$[\cdot]$' denotes the block matrix and the block-wise element $[a]_{kt}$ is a reversible matrix. Since Pauli matrices are complex orthogonal unitary matrices widely exploited in the MIMO code designing [14] and quantum information processing [15], in this paper, we investigate how to fast design the Pauli block matrix, which is available in the coding theory of quantum error-correction codes (QECC).

QECC, as a primary tool for fighting decoherence, demonstrates a formal possibility of storing and manipulating quantum data for the arbitrarily long time in the presence of noise [16–19]. Currently, the striking development of QECC is the employment of the stabilizer formalism, whereby code words are subspaces in Hilbert space specified by an Abelian group. The problem of constructing QECC was reduced to that of searching for the classical dual-containing codes or self-orthogonal codes [16,17]. The virtue of this approach is that QECC can be directly constructed from the classical codes with a certain property, rather than developing a completely new coding theory of QECC from scratch. Unfortunately, the need for the dual-containing code presents a substantial obstacle to the coding theory in its entirety, especially in the context of modern codes, such as Turbo codes and low-density parity-check (LDPC) codes with the large codeword length [18,20]. To resolve this problem, the recursive techniques for the fast block transforms is proposed in this paper to construct the large-order Pauli block jacket matrix, and hence to fast construct quantum code of the large codeword length.

This paper is outlined as follows. In Sect. 2, we describe the standard Pauli block jacket matrix formalism, which is the foundation to construct quantum codes. In Sect. 3, we state the fast construction algorithm for quantum codes in detail. We design Abelian groups based on Pauli block (jacket) matrices, from which the generators of the stabilizer can be fast generated. In Sect. 4, it is shown that the present codes have asymptotically good behaviors. In Sect. 5, the efficiency of the proposed construction approach is concisely analyzed. Finally, conclusions are drawn in Sect. 6.

2 Pauli block jacket matrices

Pauli matrices [15] are defined by $\mathcal{P} = \{\sigma_j : 0 \le j \le 3\}$, for which

$$\begin{aligned} \sigma_0 &= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad \sigma_1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \\ \sigma_2 &= \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad \sigma_3 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \end{aligned} \tag{1}$$

where σ_0 is the 2-order identity matrix I_2, and $i^2 = -1$. For simplicity, we denote I_2 by a block matrix $[I]$ in this paper.

Definition 2.1 Consider two rows of a block matrix $[A]_n = ([a]_{kt})_{n\times n}$, i.e., $[\alpha]_i = ([a]_{i1}, \ldots, [a]_{in})$ and $[\alpha]_j = ([a]_{j1}, \ldots, [a]_{jn})$, where $[a]_{kt}$ is a reversible matrix for any kt. The block matrix $[A]_n$ is called a block jacket matrix if and only if

$$[\alpha]_i \cdot [\alpha]_j^{-1} = \sum_{u=1}^{n} \left\{ [a]_{iu} [a]_{ju}^{-1} \right\} = 0, \tag{2}$$

where $[\alpha]_j^{-1} = \left([a]_{j1}^{-1}, \ldots, [a]_{jn}^{-1} \right)$. Specially, if the element $[a]_{kt} \in \mathcal{P}$ is a Pauli matrix, then $[A]_n$ is called Pauli block jacket matrix, denoted by $[J]_n$.

2.1 Pauli block jacket matrices with size 2 by 2

The 2-order Pauli block jacket matrix $[J]_2$ can be constructed easily as,

$$[J]_2 = \begin{pmatrix} \sigma_i & \sigma_j \\ \sigma_j & \sigma_i \end{pmatrix}, \tag{3}$$

where $\sigma_i, \sigma_j \in \{\sigma_1, \sigma_2, \sigma_3\}$ for $i, j \in \{1, 2, 3\}$. In fact, according to Definition 2.1, one obtains

$$\sigma_i \cdot \sigma_j^{-1} + \sigma_j \cdot \sigma_i^{-1} = \sigma_i \cdot \sigma_j + \sigma_j \cdot \sigma_i = 0.$$

It implies that $[J]_2$ in Eq. 3 is also a Pauli block jacket matrix. However, if $\sigma_i = \sigma_0$ or $\sigma_j = \sigma_0$, then the 2-order block jacket matrix $[J']_2$ can be constructed as,

$$[J']_2 = \begin{pmatrix} \sigma_0 & \sigma_j \\ \sigma_j & -\sigma_0 \end{pmatrix}, \tag{4}$$

which follows the definition of the Pauli block jacket matrix in Definition 2.1.

2.2 Pauli block jacket matrices with higher sizes

To design a Pauli block jacket matrix with the large size, we need to introduce the Kronecker product, $[J]_p \otimes [J]_q$, of two Pauli block jacket matrices $[J]_p$ and $[J]_q$, i.e.,

$$[J]_p \otimes [J]_q = \begin{pmatrix} [a]_{11}[J]_q & \cdots & [a]_{1p}[J]_q \\ [a]_{21}[J]_q & \cdots & [a]_{2p}[J]_q \\ \cdots & \cdots & \cdots \\ [a]_{p1}[J]_q & \cdots & [a]_{pp}[J]_q \end{pmatrix}, \tag{5}$$

where $[J]_{p_1} = ([a]_{ij})_{p\times p}$ and $[J]_q = ([b]_{ij})_{q\times q}$.

Making use of the Kronecker product of several Pauli block jacket matrices, a family of block jacket matrices may be extended by using the following theorem.

Theorem 2.1 *Suppose $[J]_p$ and $[J]_q$ are Pauli block jacket matrices. For any non-negative integer numbers m and s, an n-order Pauli block jacket matrix $[J]_n$ may be constructed as follows*

$$\begin{aligned} [J]_n = &\left\{ [I]_{p^m} \otimes \left(\prod_{i=1}^{s} [I]_{q^{s-i}} \otimes [J]_q \otimes [I]_{q^{i-1}} \right) \right\} \\ &\times \left\{ \left(\prod_{i=1}^{m} [I]_{p^{m-i}} \otimes [J]_p \otimes [I]_{p^{i-1}} \right) \otimes [I]_{q^s} \right\}, \end{aligned} \tag{6}$$

where $n = p^m q^s$ for any two prime numbers p and q.

Proof Based on an arbitrary r-order Pauli block jacket matrix $[J]_r$, the n_0-order block jacket matrices $[J]_{n_0}$ can be obtained by using the recursive relations,

$$\begin{aligned} [J]_{n_0} &= [J]_{r^l} = [J]_{r^{l-1}} \otimes [J]_r \\ &= \prod_{i=0}^{l-1} [I]_{r^{l-i-1}} \otimes [J]_r \otimes [I]_{r^i} \\ &= \prod_{i=1}^{l} [I]_{r^{l-i}} \otimes [J]_r \otimes [I]_{r^{i-1}}, \end{aligned} \tag{7}$$

where $n_0 = r^l$ for $r \in \{p, q\}$ and $l \in \{m, s\}$. Since,

$$[J]_{p^m q^s} = ([I]_{p^m} \otimes [J]_{q^s}) \cdot ([J]_{p^m} \otimes [I]_{q^s}),$$

the proof of the theorem is straightforward. □

Example 2.1 Taking $[J]_p = [J]_2$ for $i = 1$, $j = 2$ and $[J]_q = [J]_2$ for $i = 1$, $j = 3$ in Eq. 3, the 4-order Pauli block jacket matrix $[J]_4$ can be constructed

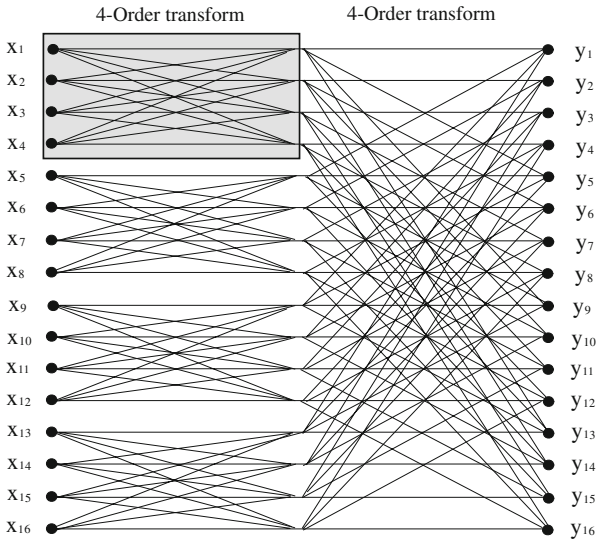

Fig. 1 Signal flow graph for Pauli block transform in Eq. 9

$$[J]_4 = [J]_p \otimes [J]_q = \begin{pmatrix} \sigma_0 & -i\sigma_2 & -i\sigma_3 & i\sigma_1 \\ -i\sigma_2 & \sigma_0 & i\sigma_1 & -i\sigma_3 \\ -i\sigma_3 & i\sigma_1 & \sigma_0 & -i\sigma_2 \\ i\sigma_1 & -i\sigma_3 & -i\sigma_2 & -\sigma_0 \end{pmatrix}. \tag{8}$$

In fact, for any two rows of $[J]_4$, say $[\rho]_2 = (-i\sigma_2, \sigma_0, -i\sigma_1, -i\sigma_3)$ and $[\rho]_3 = (-i\sigma_3, i\sigma_1, \sigma_0, -i\sigma_2)$, it follows

$$[\rho]_1 \cdot [\rho]_2^{-1} = i\sigma_2(i\sigma_3)^{-1} + (i\sigma_1)^{-1} + i\sigma_1 + i\sigma_3(i\sigma_2)^{-1} = 0.$$

Therefore, the block matrix $[J]_4$ is a Pauli block jacket matrix.

Example 2.2 We consider Pauli block jacket matrix $[J]_4$ in Eq. 8 as a basic matrix to generate the large order Pauli block jacket matrices $[J]_{4^m}$ for $m \geq 2$. For example, taking $m = 2$, the 16-order matrix $[J]_{16}$ can be constructed from

$$[J]_{4^2} = ([I]_4 \otimes [J]_4)([J]_4 \otimes [I]_4). \tag{9}$$

The factor graph of above equation can be shown in Fig. 1. It requires 96 additions and 128 multiplications for computation. The computational complexity of the proposed algorithms is shown in Table 1. Comparing it with the direct computation approach, the present algorithms are obviously faster.

Table 1 Computation complexity of the fast algorithms for the construction or decomposing of Pauli block matrices

	Direct computing	Proposed in Eq. 6	Proposed in Eq. 7
Add.	$(n-1)n$	$(p-1)mn+(q-1)sn$	$(r-1)ln_0$
Mult.	n^2	$pmn+qsn$	rln_0

Add. and Mult. denote the number of additions and multiplications

3 Quantum codes based on Pauli block matrices

The above section focuses on the construction of Pauli block jacket matrices. In this section, we continue to investigate applications of the construction or decomposition of Pauli block jacket matrices in the coding theory of QECC. Generally, there are two construction approaches to Pauli block matrices. One is the random construction, and another is the regular method. In practices, Pauli block matrices are always constructed in a regular way. In this section we make the regular constructing approach to the sparse matrix factorization of Pauli block jacket matrices. After that, we employ the proposed approach for the fast constructions of quantum codes based on the stabilizer formalism.

3.1 Abelian groups based on Pauli block matrices

The power of the stabilizer formalism for quantum codes comes from the clever use of the group theory. Let $\mathcal{P}^{\otimes n}$ denote the set of the n-fold tensor product of the single-qubit Pauli operation (matrix) in $\mathcal{P}$. Then $\mathcal{P}^{\otimes n}$ together with possible factors in $\{\pm i, \pm 1\}$ form an n-qubit operation group $\mathcal{G}_n$. An arbitrary operation $\alpha_u \in \mathcal{G}_n$, which can also be regarded as a 2^n by 2^n matrix, is uniquely expressed by the Kronecker product [15]

$$\alpha_u = i^{\lambda}\left(\sigma_1^{x_{u1}}\sigma_3^{z_{u1}}\right)\otimes\left(\sigma_1^{x_{u2}}\sigma_3^{z_{u2}}\right)\otimes\cdots\otimes\left(\sigma_1^{x_{un}}\sigma_3^{z_{un}}\right), \tag{10}$$

where $x_{uj}, z_{uj} \in \{0, 1\}$ for $1 \le j \le n$. Omitting factor i^{λ}, we denote α_u by a concatenated $2n$-dimensional vector $\vec{\alpha}_u$,

$$\vec{\alpha}_u = (\vec{x}_u|\vec{z}_u) = (x_{u1}, x_{u2}, \ldots, x_{un}|z_{u1}, z_{u2}, \ldots, z_{un}). \tag{11}$$

The Hamming weight of $\vec{\alpha}_u$ is the Hamming weight of the bitwise or of $\vec{x}_u$ with $\vec{z}_u$. The symplectic inner product of two vectors $\vec{\alpha}_u = (\vec{x}_u|\vec{z}_u)$ and $\vec{\alpha}_v = (\vec{x}_v|\vec{z}_v)$ is defined by

$$\vec{\alpha}_u \cdot \vec{\alpha}_v = \vec{x}_u \cdot \vec{z}_v + \vec{z}_u \cdot \vec{x}_v = \vec{\alpha}_u R \vec{\alpha}_v^T, \tag{12}$$

where $R = \begin{pmatrix} 0_{n\times n} & I_{n\times n} \\ I_{n\times n} & 0_{n\times n} \end{pmatrix}$. According to ref. [16], two operations α_u and α_v commute if and only if

$$\vec{\alpha}_u \cdot \vec{\alpha}_v = 0. \tag{13}$$

Assume there is no zero-entry in each row $[\rho]_u = ([a]_{u1}, [a]_{u2}, \ldots, [a]_{un})$ of $[J]_n$, from which an n-qubit operation, called the row operation,

$$\alpha_u = [a]_{u1} \otimes [a]_{u1}, \otimes \cdots \otimes [a]_{un},$$

can be generated directly, where $[a]_{uj} = \sigma_1^{x_{uj}} \sigma_3^{z_{uj}}$ for $1 \le u, j \le n$.

To achieve the large-length commutative operations, we consider the q by q permutation matrices $P_q^w \in \{P^w : 1 \le w \le q\}$, where the q by q permutation matrix P is defined by

$$P = \begin{pmatrix} 0\ 1\ 0 \cdots 0 \\ 0\ 0\ 1 \cdots 0 \\ \vdots\ \vdots\ \vdots\ \ \vdots\ \ \vdots \\ 1\ 0\ 0 \cdots 0 \end{pmatrix}.$$

After substituting σ_i and σ_j for the entries '0' and '1' of P_q^w, respectively, we get the q-order Pauli block matrix $[P]_q^w$, where $\sigma_i, \sigma_j \in \mathcal{P}$.

Theorem 3.1 *All row operations of the Pauli block matrix* $[J]_{2q}$ *are commuting, where*

$$[J]_{2q} = [J]_2 \otimes [P]_q^w. \tag{14}$$

Proof Assume any two row operations of the Pauli block jacket matrix $[J]_{2q}$ are respectively denoted by

$$\begin{aligned} \alpha_u &= [a]_{u,1} \otimes [a]_{u,2} \otimes \cdots \otimes [a]_{u,2q}, \\ \alpha_v &= [a]_{v,1} \otimes [a]_{v,2} \otimes \cdots \otimes [a]_{v,2q}, \end{aligned}$$

where $[a]_{u,i}, [a]_{v,j} \in \mathcal{P}$ for $1 \le u, v, i, j \le n$. Based on the properties of Pauli matrices in Sect. 2, one obtains

$$\begin{aligned} \alpha_u \cdot \alpha_v &= ([a]_{u,1} \otimes \cdots \otimes [a]_{u,2q}) \cdot ([a]_{v,1} \otimes \cdots \otimes [a]_{v,2q}) \\ &= ([a]_{u,1} \cdot [a]_{v,1}) \otimes \cdots \otimes ([a]_{u,n} \cdot [a]_{v,2q}) \\ &= (-1)^{2\lambda} ([a]_{v,1} \cdot [a]_{u,1}) \otimes \cdots \otimes ([a]_{v,n} \cdot [a]_{u,2q}) \\ &= ([a]_{u,1} \otimes \cdots \otimes [a]_{u,2q}) \cdot ([a]_{v,1} \otimes \cdots \otimes [a]_{v,2q}) \\ &= \alpha_v \cdot \alpha_u, \end{aligned}$$

where λ is the number of product $[a]_{v,i} \cdot [a]_{u,i}$ such that $[a]_{v,i} \ne \sigma_0$, $[a]_{u,i} \ne \sigma_0$ and $[a]_{v,i} \ne [a]_{u,i}$. This completes the proof of the theorem. □

Corollary 3.1 *All row operations of the Pauli block matrix* $[J]_{p^m q}$ *are commuting, where*

$$[J]_{p^m q} = [J]_{p^m} \otimes [P]_q^w, \tag{15}$$

for which $[J]_{p^m}$ *is a Pauli block jacket matrix in Eq. 6.*

Corollary 3.2 *Given two Pauli block matrices* $[J]_{p^m q_1}$ *and* $[J]_{p^m q_2}$*, any two row operations of the matrix*

$$[J]_{(p^m q_1)^l (p^m q_2)^h} = [J]_{(p^m q_1)^l} \otimes [J]_{(p^m q_2)^h}, \tag{16}$$

are commuting, where l *and* h *are two nonnegative integer numbers.*

Example 3.1 Based on two rows of $[J]_2$ in Eq. 3, one obtains two row operations, $\alpha_1 = \sigma_i \otimes \sigma_j$ and $\alpha_2 = \sigma_j \otimes \sigma_i$, and hence

$$\begin{aligned}\alpha_1 \cdot \alpha_2 &= (\sigma_i \otimes \sigma_j) \cdot (\sigma_j \otimes \sigma_i) = (\sigma_i \cdot \sigma_j) \otimes (\sigma_j \cdot \sigma_i) \\ &= (-1)^2 (\sigma_j \cdot \sigma_i) \otimes (\sigma_i \cdot \sigma_j) = (\sigma_j \otimes \sigma_i) \cdot (\sigma_i \otimes \sigma_j) \\ &= \alpha_2 \cdot \alpha_1.\end{aligned}$$

To confirm the commutativity of α_1 and α_2 without loss of the generality, we let $\sigma_i = \sigma_1$ and $\sigma_j = \sigma_3$. Then two vectors $\vec{\alpha}_1 = (10|01)$ and $\vec{\alpha}_2 = (10|01)$, together with the concatenated matrix

$$A_2 = (A_x^2 | A_z^2) = \left(\begin{array}{cc|cc} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{array}\right), \tag{17}$$

are achieved. It is obvious that

$$A_x^2 \cdot \left(A_z^2\right)^T + A_z^2 \cdot \left(A_x^2\right)^T = 0 \mod 2.$$

It implies that two operations α_1 and α_2 are commuting, and hence span an Abelian group $\langle \{\alpha_1, \alpha_2\} \rangle$.

Furthermore, according to Corollary 3.1, any different row operations of the Kronecker product $[J]_{2^m}$ are commuting. Specially, taking $m = 2$, one gets

$$[J]_4 = [J]_2 \otimes [J]_2.$$

It is easy to check that $[J]_4$ is a Pauli block jacket matrix with two independent commutative row operations.

Example 3.2 We consider $[J]_6 = [J]_2 \otimes [P]_3^2$, where

$$[P]_3^2 = \begin{pmatrix} \sigma_0 & \sigma_0 & \sigma_3 \\ \sigma_3 & \sigma_0 & \sigma_0 \\ \sigma_0 & \sigma_3 & \sigma_0 \end{pmatrix}.$$

Provided $[J]_2$ for $i = 1,\ j = 2$ in Eq. 3, the concatenated matrix H_6 can be constructed

$$H_6 = \left(\begin{array}{c|c} 111111 & 001110 \\ 111111 & 100011 \\ 111111 & 010101 \\ 111111 & 110001 \\ 111111 & 011100 \\ 111111 & 101010 \end{array}\right). \tag{18}$$

It is easy to check that 4 rows of H_6 are independent and commuting.

3.2 Constructions of quantum codes

Given an Abelian operation subgroup $\mathcal{S}$ of $\mathcal{G}_n$, the stabilizer quantum codes $C(\mathcal{S})$ is a set of n-qubit quantum states associated with $\mathcal{S}$, i.e.,

$$C(\mathcal{S}) = \{|\psi\rangle : M|\psi\rangle = |\psi\rangle, \quad \forall M \in \mathcal{S}\}, \tag{19}$$

which is the subspace fixed by $\mathcal{S}$ (called as the stabilizer). For the stabilizer quantum code $[[n, k, d]]$, it encodes k logical qubits into n physics qubits.

Suppose there are $n-k$ generators to span the stabilizer $\mathcal{S}$, from which one obtains the $n-k$ by $2n$ concatenated matrix

$$H = (H_x|H_z).$$

According to the construction of quantum codes in ref.[17], we calculate the generator matrix of quantum codes $G = (G_x|G_z)$ from

$$H_x G_z^T + H_z G_x^T = 0. \tag{20}$$

It can be rewritten as

$$HRG^T = 0. \tag{21}$$

To construct a systemical quantum code, we assume that there exists one unitary matrix U such that

$$U(HR) = (I_{(n-k)\times(n-k)}|\Lambda_{(n-k)\times(n+k)}).$$

According to Eq. 21, the generator matrix G is achieved

$$G = \left(\Lambda^T_{(n-k)\times(n+k)}|I_{(n+k)\times(n+k)}\right). \tag{22}$$

Theorem 3.2 *Given a Pauli block matrix* $[J]_n$ *with at least* $n-k$ *different commutative row operations, the stabilizer quantum code* $C(\mathcal{S})$ *can be constructed with parameters* $[[n,k,d]]$*, where the stabilizer* $\mathcal{S}$ *is a set of n-qubit operations spanned by* $n-k$ *independent row operations of Pauli block matrix* $[J]_n$ *for* $n=(p^m q_1)^l(p^m q_2)^h$ *in Eq. 16.*

Proof Since $[J]_n$ is the Kronecker product of several small size Pauli block matrices $[J]_{p^m q_1}$ and $[J]_{p^m q_2}$, we know that the number of independent row operations of $[J]_n$ depends simultaneously on the independent row operations of $[J]_{pq_1}$ and $[J]_{pq_2}$ [9], which is at most $p^2 q_1 q_2$. Selecting any $n-k$ operations from these independent commutative row operations, an Abelian group, the stabilizer $\mathcal{S}$, can be generated. Namely, any $n-k$ independent row operations $\alpha_{i_1}, \alpha_{i_2}, \ldots, \alpha_{i_{n-k}}$ spans the stabilizer

$$\mathcal{S} = \langle \{\alpha_{i_1}, \alpha_{i_2}, \ldots, \alpha_{i_{n-k}}\} \rangle. \qquad \square$$

Combining Eqs. 19, 20, and 22, the stabilizer quantum code can be structurally constructed.

Example 3.3 Consider the Pauli block matrix $[J]_6$ with the concatenated matrix H_6 described in Eq. 18. There are four independent commutative row operations, which serve as the generators of the stabilizer $\mathcal{S}$. Combining Eqs. 21 and 22, the generator matrix of quantum codes be $G_6 = (G_x|G_z)$ can be calculated as

$$G_6 = \left(\begin{array}{cc} 000000 & 110000 \\ 000000 & 101000 \\ 000000 & 100100 \\ 000000 & 100010 \\ 000000 & 100001 \\ 100100 & 100000 \\ 010010 & 100000 \\ 001001 & 100000 \end{array}\right). \tag{23}$$

Based on Eq. 23, a quantum code $[[6,2,2]]$ can be constructed with the stabilizer $\mathcal{S}$ spanned by the operations with the corresponding concatenated matrix in Eq. 34 shown in Appendix.

4 Asymptotically good behaviors

In the previous section, the stabilizer quantum code $C(\mathcal{S})$ with the parameters $[[n,k,d]]$ can be constructed from Pauli block matrix $[J]_n$ with at least $n-k$ independent commutative row operations. Since the value of the parameter d, Hamming distance of the quantum code, depends on the independent rows of the possible Pauli block matrix $[J]_n$ we now analyze the asymptotic behaviors of the quantum code based on Pauli block jacket matrix $[J]_n$ in Eq. 16 (Fig. 2).

Denote by $\delta = d/n$ and $R = k/n$, respectively. Without loss of generality, we only consider the case of $n = (2q_1)^l$ in this section. Taking the Pauli block matrix $[J]_{2q_1}$ in

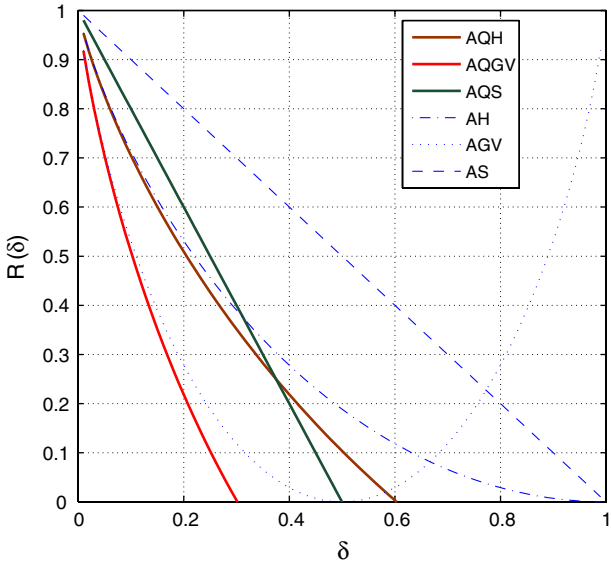

Fig. 2 The relationship of the asymptotic binary quantum singleton bound (*AQS*), the quantum Hamming bound (*AQH*), the quantum Gilbert–Varshamov bound (*AQGV*), and the classical binary singleton bound (*AS*), the classical Hamming bound (*AH*), the classical Gilbert–Varshamov bound (*AGV*)

Eq. 14, it is known that the number of independent row operations of $[J]_{2q_1}$, and hence $[J]_{(2q_1)^l}$, is at most $2q_1$. For some fixed value η we can select the proper parameters n and k such that

$$n - k \leq 2q_1\eta - 1, \tag{24}$$

where $0 \leq \eta \leq 1$. So it follows the lower bound of R,

$$R = \frac{k}{(2q_1)^l} \geq 1 - \frac{\eta}{(2q_1)^{l-1}} + \frac{1}{(2q_1)^l}. \tag{25}$$

By the result of the quantum singleton bound for the binary quantum code [16], we have the upper bound for the pure quantum code of distance d,

$$k \leq n - 2d + 2. \tag{26}$$

Employing Eq. 26, we have

$$\begin{aligned} \delta &= \frac{d}{(2q_1)^l} \leq \frac{\eta}{2(2q_1)^{l-1}} + \frac{1}{(2q_1)^l}, \\ R &= \frac{k}{(2q_1)^l} \leq 1 - 2\delta + \frac{2}{(2q_1)^l}. \end{aligned} \tag{27}$$

Specially taking $l = 1$, one obtains

$$\lim_{q_1\to\infty} \delta \le \frac{\eta}{2} + \lim_{q_1\to\infty} \frac{1}{2q_1} = \frac{\eta}{2},$$
$$\lim_{q_1\to\infty} R \le 1 - 2\delta + \lim_{q_1\to\infty} \frac{2}{2q_1} = 1 - 2\delta. \tag{28}$$

Theorem 4.1 *Given the Pauli block jacket matrix* $[J]_n$ *in Eq. 16, the stabilizer quantum code* $[[n, k, d]]$ *that are generated from the generator matrix* $G = (G_x|G_z)$ *via Eq. 20 are asymptotically good.*

Proof Without loss of the generality, we consider the Pauli block matrix $[J]_n = [J]_{(2q_1)^l}$ with the basic pauli block matrix $[J]_{2q_1}$ in Eq. 14. Since there are three possible non-trivial single-qubit errors in $\{\sigma_1, \sigma_2, \sigma_3\}$, the number of errors of length i on an n-qubit quantum state is $3^i \binom{n}{i}$. For the binary quantum code $[[n, k, d]]$ that corrects errors up to $t = \lfloor (d-1)/2 \rfloor$, we get the quantum Hamming Bound,

$$\sum_{i=0}^{t} 3^i \binom{n}{i} \le 2^{n-k}, \tag{29}$$

and the quantum Gilbert bound

$$\sum_{i=0}^{d-1} 3^i \binom{n}{i} \ge 2^{n-k}. \tag{30}$$

Define the entropy $H(x)$ function [15]

$$H(x) = x \log_4 3 - x \log_4 x - (1-x) \log_4 (1-x), \tag{31}$$

where $0 < x < 1$. Combining Eqs. 24 and 26, one obtains

$$d \le (n-k)/2 + 1 \le q_1 + 1 \le (2q_1)^l/2 = n/2. \tag{32}$$

After employing Stirling's formula for Eq. 29, we get

$$\frac{2}{n} \log_4 \sum_{i=0}^{t} 3^i \binom{n}{i} = 2H(\frac{t}{n}) + o(1) \le 1 - \frac{k}{n}, \tag{33}$$

and hence

$$\lim_{q_1\to\infty} R \le \lim_{q_1\to\infty}\left[1-2H\left(\frac{t}{n}\right)-o(1)\right]$$
$$\le \lim_{q_1\to\infty}\left[1-2H\left(\frac{d}{2n}\right)-o(1)\right]=1-2H\left(\frac{\delta}{2}\right).$$

Thus the proposed quantum code meets the asymptotic quantum Hamming bound [19]. Similarly, through Eqs. 30 and 32, it is obvious that

$$\frac{2}{n}\log_4\sum_{i=0}^{d-1}3^i\binom{n}{i}=2H\left(\frac{d-1}{n}\right)+o(1)\ge 1-\frac{k}{n}.$$

As a consequence, we obtain

$$\lim_{q_1\to\infty} R \ge \lim_{q_1\to\infty}\left[1-2H\left(\frac{d-1}{n}\right)-o(1)\right]$$
$$\ge \lim_{q_1\to\infty}\left[1-2H\left(\frac{d}{n}\right)-o(1)\right]=1-2H(\delta).$$

It means that the proposed quantum code meets the asymptotic quantum Gilbert–Varshamov bound [19]. This completes the proof of the theorem. □

5 Efficiency

To construct the quantum code $[[n, k, d]]$ with the large codeword length, in this paper we do not need to find the classical self-orthogonal (or self-dual) code with respect to a certain trace inner product that is used for the design of quantum code in the constrained range $1 < n < 511$ [16,17]. In fact, we may extend generally the codeword to the large length $n \ge 512$ by making use of the fast construction algorithms with respect to the Kronecker product of the small size codes whether they are self-orthogonal (self-dual) or not. Moreover, we can also construct a quantum code without the need of the self-orthogonal (or self-dual) code. According to Table 2, some quantum codes, which would not be achieved via Calderbank–Shor–Steane's construction, can also be fast constructed from the Kronecker product of several small size Pauli block jacket matrices $\{[J]_2, [J]_3, [J]_4\}$ and Pauli block matrices $\{[P]_2^w, [P]_3^w, [P]_3^w\}$. It is obvious that the yielded quantum codes [[16, 10, 2]] and

Table 2 Quantum codes with parameters $[[n, k, d]]$ by Theorem 3.2

n	6	8	8	12	16	16	16
k	2	4	5	6	10	0	4
d	2	2	2	2	2	8	5
H_n in Eq. $(\cdot)$	34	35	36	37	38	39	40

For the above table, the concatenated matrices H_n of the size $(n-k)\times 2n$ in Eqs. 34–40 corresponding to the stabilizers of $[[n, k, d]]$ quantum codes are listed in Appendix

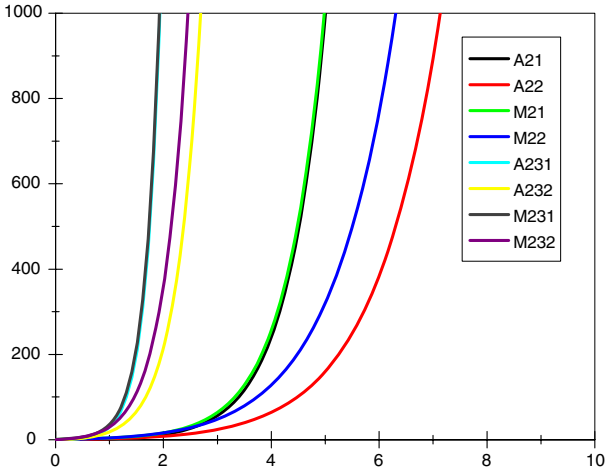

Fig. 3 The encoding computation complexities of the construction algorithms with respect to Table 1 for quantum codes with the respective parameters $n = 2^m$ and $n = 2^m 3^s$. For the simplicity, the notations 'A21' and 'M21' denote the respective number of additions and multiplication calculated with the direct constructions, while 'A22' and 'M22' denote the number of additions and multiplication for the proposed constructions with $n = 2^m$; 'A231' and 'M231' denote the respective number of additions and multiplication for the direct constructions, while 'A232' and 'M232' denote the number of additions and multiplication for the proposed constructions with $n = 2^m 3^s$ and $m = s$

[[16, 4, 5]] have better permitters than the Calderbank–Shor–Steane's code [[16, 4, 3]] in ref. [17].

It is necessary to remark that the parameters of quantum codes in this paper compare well with the most efficient quantum codes known. For an arbitrary number $n = p^m q^s$, the quantum code $[[n, k, d]]$ can be fast constructed from two Pauli block (jacket) matrices $[J]_p$ and $[J]_q$ with the construction algorithm described in Theorem 2.1, where p and q are prime numbers. Through implementing such an algorithm, the amount of the encoding computation complexities is made much smaller than that of the quantum code constructed with the direct computation approach. As an example, taking $p = 2$, $q = 3$ and $m = s$ for the computation complexities in Table 1, the amount of the encoding complexities is sketched in Fig. 3. In particular, let $m = s = 4$, and then the quantum code $[[2^4 3^4, k, d]]$ can be constructed from the recursive relationship of the identity matrix, the 2-order Pauli block jacket matrix $[J]_2$ and the 3-order Pauli block matrix $[P]_3^w$. Comparing to the direct computing approach, which requires 1678320 additions and 1679616 multiplications for the encoding, the proposed construction algorithm requires no more than 15552 additions and 25920 multiplications. It is obvious that this algorithm is much faster than the direct computing approach, which shows the good performance of our construction algorithm.

Furthermore, for any two large positive numbers m and s, the quantum code $[[p^m q^s, k, d]]$ can be fast constructed from the recursive relationship of the corresponding identity matrices, successive Pauli block jacket matrices and Pauli block matrices. It means that the suggested codes are even more suitable when the large codeword lengths are needed. For instance, to construct quantum code with the length

$n = 248832$, all that we need to do is to design the Pauli block matrix $[J]_{2^{10}3^5}$ through selecting $p = 2$, $q = 3$, $m = 10$ and $s = 5$. According to Table 1, it requires no more than 497664 additions and 8709120 multiplications for the proposed algorithm to generate the quantum code $[[248832, k, d]]$.

6 Conclusions

The Pauli block jacket matrices and its applications in coding theory of quantum error-correction codes are investigated in the paper. We investigate constructions of the large order Pauli block jacket matrices. We suggest the fast construction (or decomposition) algorithm for the Pauli block matrix based on the recursive relationship of the identity matrix and successively lower order Pauli block matrices. Since Pauli matrices are all complex orthogonal unitary matrices, we make an instructive approach for the fast constructions of the large order Pauli block matrices, which can be employed to span an Abelian operation group for the generation of the stabilizer of a quantum code. The present stabilizer formalism enables us to structurally construct quantum codes with the efficiency. It also provides the great flexibility in designing quantum codes with large codeword length. These quantum codes have an advantage of being fast constructed with the low complexity and the asymptotically good behaviors.

Appendix: the concatenated matrices of the stabilizers

In this appendix, we list some of the concatenated matrices H_n of the size $(n-k) \times 2n$, which are required for the constructions of the stabilizers of the quantum codes $[[n, k, d]]$ for $n = 6, 8, 8, 12, 16, 16, 16$ in Table 2.

(1) The stabilizer of the $[[6, 2, 2]]$ quantum code:

$$H_6 = \left(\begin{array}{c|c} 111111 & 001110 \\ 111111 & 100011 \\ 111111 & 010101 \\ 111111 & 110001 \\ 111111 & 011100 \\ 111111 & 101010 \end{array}\right). \tag{34}$$

(2) The stabilizer of the $[[8, 4, 2]]$ quantum code:

$$H_8 = \left(\begin{array}{c|c} 11111111 & 00011110 \\ 11111111 & 10000111 \\ 11111111 & 01001011 \\ 11111111 & 00101101 \\ 11111111 & 11100001 \\ 11111111 & 01111000 \\ 11111111 & 10110100 \\ 11111111 & 11010010 \end{array}\right). \tag{35}$$

(3) The stabilizer of the [[8, 5, 2]] quantum code:

$$H_8 = \left(\begin{array}{c|c} 10100101 & 10011001 \\ 01011010 & 01100110 \\ 10100101 & 01100110 \\ 01011010 & 10011001 \end{array}\right). \tag{36}$$

(4) The stabilizer of the [[12, 6, 2]] quantum code:

$$H_{12} = \left(\begin{array}{c|c} 000111000111 & 001110110001 \\ 000111000111 & 100011011100 \\ 000111000111 & 010101101010 \\ 111000111000 & 110001001110 \\ 111000111000 & 011100100011 \\ 111000111000 & 101010010101 \\ 000111000111 & 110001001110 \\ 000111000111 & 011100100011 \\ 000111000111 & 010101101010 \\ 111000111000 & 001110110001 \\ 111000111000 & 100011011100 \\ 111000111000 & 010101101010 \end{array}\right). \tag{37}$$

(5) The stabilizer of the [[16, 10, 2]] quantum code:

$$H_{16} = \left(\begin{array}{c|c} 0000111100001111 & 0001111011100001 \\ 0000111100001111 & 1000011101111000 \\ 0000111100001111 & 0100101110110100 \\ 0000111100001111 & 0010110111010010 \\ 1111000011110000 & 1110000100011110 \\ 1111000011110000 & 0111100010000111 \\ 1111000011110000 & 1011010001001011 \\ 1111000011110000 & 1101001000101101 \\ 0000111100001111 & 0001111000011110 \\ 0000111100001111 & 1000011110000111 \\ 0000111100001111 & 0100101101001011 \\ 0000111100001111 & 0010110100101101 \\ 1111000011110000 & 1110000111100001 \\ 1111000011110000 & 0111100001111000 \\ 1111000011110000 & 1011010010110100 \\ 1111000011110000 & 1101001011010010 \end{array}\right). \tag{38}$$

(6) The stabilizer of the [[16, 0, 8]] quantum code:

$$H_{16} = \left(\begin{array}{c|c} 0001111100011110 & 0000111111110000 \\ 1000111110000111 & 0000111111110000 \\ 0100111101001011 & 0000111111110000 \\ 0010111100101101 & 0000111111110000 \\ 1111000111100001 & 1111000000001111 \\ 1111100001111000 & 1111000000001111 \\ 1111010010110100 & 1111000000001111 \\ 1111001011010010 & 1111000000001111 \\ 0001111000011111 & 1111000000001111 \\ 1000011110001111 & 1111000000001111 \\ 0100101101001111 & 1111000000001111 \\ 0010110100101111 & 1111000000001111 \end{array}\right). \tag{39}$$

(7) The stabilizer of the [[16, 4, 5]] quantum code:

$$H'_{16} = \left(\begin{array}{c|c} 0001111100011110 & 0000111111110000 \\ 1000111110000111 & 0000111111110000 \\ 0100111101001011 & 0000111111110000 \\ 0010111100101101 & 0000111111110000 \\ 1111000111100001 & 1111000000001111 \\ 1111100001111000 & 1111000000001111 \\ 1111010010110100 & 1111000000001111 \\ 1111001011010010 & 1111000000001111 \\ 0001111000011111 & 1111000000001111 \\ 1000011110001111 & 1111000000001111 \\ 0100101101001111 & 1111000000001111 \\ 0010110100101111 & 1111000000001111 \end{array}\right), \tag{40}$$

which is obtained by selecting 12 rows from the full rank matrix H_{16} in Eq. 39.

Acknowledgments The authors are grateful to the anonymous referees for their detailed suggestions. This work was supported by Natural Science Foundation of Hunan Province (Nos. 07JJ3128, 2008RS4016), Postdoctoral Science Foundation of China (Nos. 20070420184, 200801341), and in part by Joint Project KOSEF/NSFC Korea Research Foundation KRF-2007-521-D00330, Chonbuk National University, Korea.

References

1. Ahmed, N., Rao, K.R.: Orthogonal Transforms for Digital Signal Processing. Springer, Berlin (1975)
2. Lee, M.H., Kaveh, M.: Fast Hadamard transform based on a simple matrix factorization. IEEE Trans. Acoust. Speech Signal Process. **34**(6), 1666–1667 (1986)
3. Lee, M.H.: The center weighted Hardamard transform. IEEE Trans. Circuits Syst. **CAS-36**, 1247–1249 (1989)

4. Rao, K.Y., Hershey, J.E.: Hadamard Matrix Analysis and Synthesis. Kluwer, Norwell (1997)
5. Lee, M.H.: A new reverse jacket transform and its fast algorithm. IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process. **47**(1), 39–47 (2000)
6. Lee, M.H., Rajan, B.S., Park, J.Y.: A generalized reverse jacket transform. IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process. **48**(7), 684–691 (2001)
7. Chen, Z., Lee, M.H., Zeng, G.: Fast cocyclic jacket transform. IEEE Trans. Signal Process. **56**(5), 2143–2148 (2008)
8. MacWilliams, F.J., Sloane, N.J.A.: The Theory of Error Correcting Codes. Elsevier, Amsterdam (1988)
9. Zeng, G., Lee, M.H.: A generalized reverse block jacket transform. IEEE Trans. Circuits Syst. I **55**(6), 1589–1600 (2008)
10. Song, W., Lee, M.H.: Orthogonal space-time block codes design using jacket transform for MIMO transmission system. In: IEEE International Conference on Communications (ICC 2008), Beijing, China (2008)
11. Lee, M.H., Hou, J.: Fast block inverse jacket transform. IEEE Signal Process. Lett. **13**(8), 461–464 (2006)
12. Lee, M.H., Finlayson, K.: A Simple Element Inverse Jacket Transform Coding. IEEE ITW, New Zealand (2005)
13. Lee, M.H., Zeng, G.: Family of fast jacket transform algorithms. Elect. Lett. **43**(11), 651 (2007)
14. Hottinen, A., Tirkkonen, O., Wichman, R.: Multi-Antenna Transceiver Techniques for 3G and Beyond. Wiley, New York (2003)
15. Nielsen, M.A., Chuang, I.L.: Quantum computation and quantum information. Cambrige University Press, Cambridge (2002)
16. Calderbank, A.R., Rains, E.M., Shor, P.W., Sloane, N.J.A.: Quantum error-correction via codes over GF(4). IEEE Trans. Inform. Theory **44**, 1369–1387 (1998)
17. Steane, A.M.: Enlagement of Calderbank–Shor–Steane quantum codes. IEEE Trans. Inform. Theory **45**(7), 2492–2495 (1999)
18. MacKay, D.J.C., Mitchison, G.J., McFadden, P.L.: Sparse-graph codes for quantum error correction. IEEE Trans. Inform. Theory **50**, 2315–2330 (2004)
19. Matsumoto, R.: Improvement of Ashikhmin–Litsyn–Tsfasman bound for quantum codes. IEEE Trans. Inform. Theory **48**, 2122–2125 (2002)
20. Djordjevic, I.B.: Quantum LDPC codes from balanced incomplete block designs. IEEE Commun. Lett. **12**(5), 389–391 (2008)