Encoding entanglement-assisted quantum stabilizer codes*

Wang Yun-Jiang(王云江)[a)†], Bai Bao-Ming(白宝明)[a)], Li Zhuo(李 卓)[a)], Peng Jin-Ye(彭进业)[b)c)], and Xiao He-Ling(肖鹤玲)[a)]

[a)] *State Key Laboratory of ISN, Department of Telecommunications Engineering, Xidian University, Xi'an* 710071, *China*

[b)] *School of Electronics and Information, Northwestern Polytechnical University, Xi'an* 710069, *China*

[c)] *The Key Laboratory of AIPPC of Ministry of Education, Northwestern Polytechnical University, Xi'an* 710069, *China*

(Received 21 April 2010; revised manuscript received 20 April 2011)

We address the problem of encoding entanglement-assisted (EA) quantum error-correcting codes (QECCs) and of the corresponding complexity. We present an iterative algorithm from which a quantum circuit composed of $CNOT$, H, and S gates can be derived directly with complexity $O(n^2)$ to encode the qubits being sent. Moreover, we derive the number of each gate consumed in our algorithm according to which we can design EA QECCs with low encoding complexity. Another advantage brought by our algorithm is the easiness and efficiency of programming on classical computers.

Keywords: quantum error correction, entanglement-assisted quantum stabilizer codes, encoding complexity

PACS: 03.67.–a, 03.67.Hk, 03.67.Pp **DOI:** 10.1088/1674-1056/21/2/020304

1. Introduction

Information processing based on quantum mechanics (so-called quantum information processing), which is viewed as the key technic for the next information revolution, has attracted more and more attention.[1−3] Noise is a great bane of information processing, in order to implement quantum information processing reliably, we have to protect against the effects of noise. As the role played by error-correcting codes in classical settings, quantum error-correcting codes are essential in quantum settings.[4−8]

The most important contributions to the theory of constructing quantum error-correcting codes (QECCs) from classical codes are the establishment of stabilizer formalism[6] and the presentation of Calderbank–Shor–Steane (CSS) codes.[7,8] According to these two theories, we can import classical dual-containing binary or quaternary codes into quantum codes directly, since a commuting subgroup of G_n[1)] (so-called stabilizer) can be obtained from a dual-containing classical code.[9−12] Unfortunately, in general, most modern codes do not satisfy the dual-containing constraint. Thus it is hard to import those codes, such as low-density parity check (LDPC) codes, into quantum domain.[13] However, if pre-shared entanglement is available, entanglement-assisted (EA) stabilizer formalism proposed in Ref. [14], will allow us to import an arbitrary classical binary or quarternary code into quantum codes by embedding a noncommuting group into a larger stabilizer which associates with a well-defined quantum code, named EAQECC. Thus, we have more freedom to design sparse quantum codes, and in general, EAQECCs have superior performances over conventional QECCs.[15−17]

The encoding operations of conventional quantum stabilizer codes can be immediately obtained from their stabilizer. However, it is not the case for EAQECCs. Considering that encoding complexity is one of the most important properties for a code in practical applications, we ask the following ques-

*Project supported by the National Basic Research Program of China (Grant No. 2010CB328300), the National Natural Science Foundation of China (Grant Nos. 60972046 and 60902030), the Program for Changjiang Scholars and Innovative Research Team in University (Grant No. IRT0852), the Natural Science Foundation of Shaanxi Province (Grant No. 2010JQ8025), the Specialized Research Fund for the Doctoral Program of Higher Education of China (Grant No. 20100203120004), the 111 Program (Grant No. B08038), and the China Scholarship Council (Grant No. [2008]3019).

†Corresponding author. E-mail: yunjiang.w@gmail.com

© 2012 Chinese Physical Society and IOP Publishing Ltd

1) G_n is a group of n-fold tensor products of single-qubit Pauli operators with the possible overall factors ± 1, $\pm i$ under multiplication.

tions for EAQECCs: How about the encoding complexity of EAQECCs? In other words, could we encode EAQECCs with polynomial complexity? Below, we will present an iterative algorithm to address this problem. As what will be shown, an iterative encoding algorithm is presented in this paper. Using this algorithm, a quantum circuit composed of $CNOT$, H, and S gates can be derived directly, with complexity $O(n^2)$, to encode the qubits being sent based on the EA stabilizer fomalism, where n is the code length.

2. Encoding algorithm of EA quantum stabilizer codes

The operations required in the encoding of EAQECCs can be described as follows: suppose Alice wants to encode her quantum state $|\psi\rangle$ according to the structure of a classical binary or quarternary code whose check matrix is H and, initially, necessary entanglement has been shared between Alice and Bob. Then Alice needs to perform operations on her $|\psi\rangle$, her half of the entangled pairs and some necessary ancilla qubits[2] such that after these operations, the quantum state being sent associates with M, where M is the group generated by the set of generators $\{M_i, i = 1, \ldots, r\}$ which are mapped from H (see Ref. [14] for details).

The organization of this section is as follows. In Subsection 2.1, we review a strategy to transform M into its canonical form which could be decomposed into an "isotropic" subgroup generated by a set of commuting generators and a "symplectic" subgroup generated by a set of anti-commuting generator pairs. In Subsection 2.2, we present an iterative algorithm to show how we could encode k logical qubits into n physical qubits by using s ancilla bits, c ebits based on EA stabilizer formalism, where s is the number of independent generators in the isotropic subgroup, c is the number of anti-commuting pairs of generators in the symplectic subgroup, and n is equal to the sum of s, c, and k. Finally, the complexity analysis of our algorithm is given in Subsection 2.3.

2.1. Transforming a noncommuting set of generators to its canonical form

According to the EA stabilizer formalism, it is easy to construct a quantum code from a classical binary or quarternary code as follows. Without loss of generality, suppose that the classical code is quarternary[3] and the check matrix is

$$\boldsymbol{H_c} = \begin{pmatrix} 1 & \omega & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 \\ \omega & 0 & \omega & 0 & 1 & \omega \end{pmatrix}. \tag{1}$$

Here, as is customary, we take $GF(4)$ to consist of the elements $0, 1, \omega, \bar{\omega}$, with $\omega^2 = \omega + 1, \omega^3 = 1$, and conjugation defined by $\bar{x} = x^2$. The isomorphic relation between Pauli operator and $GF(4)$ is as follows, $I \leftrightarrow 0$, $X \leftrightarrow 1$, $Z \leftrightarrow \omega$, and $Y \leftrightarrow \bar{\omega}(\omega^2)$; *multiplication* $\leftrightarrow$ *addition* and *commutativity* $\leftrightarrow$ *trace inner product*.

In order to construct a quantum code, we first transform H_c to $\widehat{H}_c$, where

$$\widehat{\boldsymbol{H_c}} = \begin{pmatrix} H_c \\ \omega H_c \end{pmatrix} = \begin{pmatrix} 1 & \omega & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 \\ \omega & 0 & \omega & 0 & 1 & \omega \\ \omega & \bar{\omega} & \omega & 0 & \omega & 0 \\ \omega & \omega & 0 & \omega & 0 & \omega \\ \bar{\omega} & 0 & \bar{\omega} & 0 & \omega & \bar{\omega} \end{pmatrix}. \tag{2}$$

Then using the isomorphic relation between Pauli operator and $GF(4)$, a set of Pauli generators M could be obtained, where

$$\begin{aligned} M_1 &= X\ \ Z\ \ X\ \ I\ \ X\ \ I, \\ M_2 &= X\ \ X\ \ I\ \ X\ \ I\ \ X, \\ M_3 &= Z\ \ I\ \ Z\ \ I\ \ X\ \ Z, \\ M_4 &= Z\ \ Y\ \ Z\ \ I\ \ Z\ \ I, \\ M_5 &= Z\ \ Z\ \ I\ \ Z\ \ I\ \ Z, \\ M_6 &= Y\ \ I\ \ Y\ \ I\ \ Z\ \ Y. \end{aligned} \tag{3}$$

Obviously, M is noncommuting. In order to encode quantum state according to M, we need to transform M to its canonical form first, then we will know how many entangled pairs should be pre-shared between the sender and the receiver (say Alice and Bob), how many ancilla bits Alice should prepare, and how many bits could be encoded. The following strategy mentioned in Ref. [18] can give us an iterative procedure to achieve this transformation.

Without loss of generality, suppose M has r independent generators $\{M_i, i = 1, \ldots, r\}$. Then we can choose products of M_i such that the products commute and anti-commute in pairs. This can be achieved by using the following strategy[4].[18]

[2] These encoded qubits plus the other half of the pre-shared entangled pairs held by Bob associates with a stabilizer.[14]

[3] Binary codes can be viewed as a subclass of quaternary codes.

[4] We set a variable c, initially being 0, to indicate the number of anti-commuting pairs in M.

(i) If all M_i commute to begin with, they are already in the required form with $c = 0$, end. Otherwise, go to (ii).

(ii) Let $c = c + 1$, then identify a pair M_{i_1} and M_{i_2} that are anti-commuting and make them M'_c and M'_{c+1}.

(iii) If any other M_i anti-commutes with M_{i_1}, then multiply it by M_{i_2}; if, on the other hand, M_i anti-commutes with M_{i_2}, then multiply it by M_{i_1}; and if both, by both.

(iv) Remove M_{i_1} and M_{i_2}. For the remaining modified M_i's, if they commute with each other, end; or else, go to (ii).

Consequently, we can transform M to its canonical form M' so that $[M'_i, M'_{i'}] = 0$, unless $i = 2j-1, i' = 2j$ $(j \leq c)$ when $\{M'_i, M'_{i'}\} = 0$. Moreover, from this mathematic transformation, the number of entangled pairs (c) that should be pre-shared between Alice and Bob, the number of ancilla qubits (s) will be consumed in the encoding process, and the number of logical qubits (k) can be encoded in a block can be obtained: $s = r - 2c$, $k = n + c - r$.

2.2. Iterative encoding algorithm and complexity analysis

Now suppose the canonical noncommuting group M' with six generators ($r = 6$) has been obtained by using the strategy mentioned in the last subsection, for

$$\begin{aligned} M'_1 &= X\ \ Z\ \ X\ \ I\ \ X\ \ I, \\ M'_2 &= X\ \ X\ \ I\ \ X\ \ I\ \ X, \\ M'_3 &= Z\ \ I\ \ Z\ \ I\ \ X\ \ Z, \\ M'_4 &= Z\ \ Y\ \ Z\ \ I\ \ Z\ \ I, \\ M'_5 &= Z\ \ X\ \ Y\ \ X\ \ Z\ \ Z, \\ M'_6 &= X\ \ Y\ \ Z\ \ Y\ \ X\ \ X. \end{aligned} \tag{4}$$

It is easy to check that there are two anti-commuting pairs (M'_1, M'_2) and (M'_3, M'_4), thus $c = 2$, $s = r - 2c = 6 - 2 \times 2 = 2$, $k = n + c - r = 6 + 2 - 6 = 2$. Suppose Alice and Bob have two pre-shared entangled pairs $(|00\rangle + |11\rangle)/\sqrt{2}$, two ancillia qubits $|0\rangle$ and two logical qubits to be sent. Then the stabilizer with which these initial qubits associate is generated by

$$\begin{array}{l|ll} \widehat{M}_1 = X\ \ I\ \ I\ \ I\ \ I\ \ I & X & I \\ \widehat{M}_2 = Z\ \ I\ \ I\ \ I\ \ I\ \ I & Z & I \\ \widehat{M}_3 = I\ \ X\ \ I\ \ I\ \ I\ \ I & I & X \\ \widehat{M}_4 = I\ \ Z\ \ I\ \ I\ \ I\ \ I & I & Z \\ \widehat{M}_5 = I\ \ I\ \ Z\ \ I\ \ I\ \ I & I & I \\ \widehat{M}_6 = I\ \ I\ \ I\ \ Z\ \ I\ \ I & I & I \end{array}, \tag{5}$$

where the left part of $\widehat{M}$ (denoted by $\widehat{M_{\rm L}}$) is associated with the qubits possessed by Alice, and the right part $\widehat{M_{\rm R}}$ is associated with the other half of the two entangled pairs possessed by Bob. Because the goal of Alice is to encode her qubits according to M', the task left for her is to perform encoding operations on her qubits so that the set of operators her qubits is associated with will be transformed from $\widehat{M_{\rm L}}$ to M'. $(M'|\widehat{M_{\rm R}})$ can give a well-defined EAQECCs space.

Although the isomorphic relation between $\widehat{M_{\rm L}}$ and M' in the viewpoint of commutation does allow us to design a quantum circuit composed of $CNOT$, H and S gates to achieve this transformation,[18−20] it appears a little difficult to design it out directly. In light of the property of $CNOT$, H, and S gates, if we can design a quantum circuit transforming M' to $\widehat{M_{\rm L}}$, the reverse circuit will give us what we desire.

Next, we show how we can transform M' to $\widehat{M_{\rm L}}$ using $CNOT$, H, and S gates but begin with two important theorems that our algorithm based on.[18]

Theorem 1 The encoded operators should obey the same commutation relations as their unencoded counterpart, or else, we violate the uncertainty principle.

Theorem 2 Any operation taking element of G_n to element of G_n under conjugation can be composed of the $CNOT$, H, and S gates. These gates will preserve the commutation relations between the elements of G_n.

In order to present our algorithm clearly, we first need to represent Pauli operators as symplectic binary vectors $(X|Z)$ (the left is the X part, the right the Z part). For example, $I \Leftrightarrow (0|0)$, $X \Leftrightarrow (1|0)$, $Z \Leftrightarrow (0|1)$, and $Y \Leftrightarrow (1|1)$. Thus,

$$\boldsymbol{M'} = \left(\begin{array}{cccccc|cccccc} 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \end{array}\right). \tag{6}$$

Then according to the transformation functions of $CNOT$, H, and S (Table 1),[21] the transformation matrices of these quantum gates can be easily obtained when right multiplied on pauli operators, for

$$CNOT = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix}, \tag{7}$$

$$H = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \tag{8}$$

$$S = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}. \tag{9}$$

Table 1. Transformation functions of $CNOT$, H, and S.

Operation	Input	Output
$CNOT$	X_1	X_1X_2
	X_2	X_2
	Z_1	Z_1
	Z_2	Z_1Z_2
H	X	Z
	Z	X
S	X	Y
	Z	Z

That is, when right multiplied on Pauli operator, $CNOT(i \to j)$[5] adds the i-th column in the X part to the j-th column in the X part and adds the j-th column in the Z part to the i-th column in the Z part. $H(i)$ switches the i-th column in the X part with the i-th column in the Z part. $S(i)$ adds the i-th column in the X part to the i-th column in the Z part. Due to these operations, we can transform M' to $\widehat{M_{\rm L}}$ as follows.

(I) If c is equal to 0 go to (V); else go to (II).

(II) *Transforming the first generator of the concerning anti-commuting pair to* X_1[6]: Arrange the first row so that its x_1[7] is 1. Then clear all other entries in the first row by the following steps. First, except for x_1, for all other nonzero entries (say i-th) in the X part perform $CNOT(1 \to i)$ on qubit one and the i-th qubit. After that, if z_1 is equal to 0, perform S on qubit one. If not, do nothing. Second, except for z_1, for all other nonzero entries (say j-th) in the Z part, perform $CNOT(1 \to j)$ on qubit one and the j-th qubit. Finally, perform S on qubit one, go to (III).

(III) *Transforming the second generator of the concerning anti-commuting pair to* Z_1: First, except for z_1, for all other nonzero entries (say i-th) in the Z part in the second row perform $CNOT(1 \to i)$ on qubit one and the i-th qubit. After that, set a local variable f (initially being 0), and let $f = 1$ when x_1 is equal to 1. Then perform H on qubit one. Second, for all nonzero entries (say j-th) in the X part in the second row, perform $CNOT(1 \to j)$ on qubit one and the j-th qubit. Next, if f is equal to 1, perform S on qubit one and then H on qubit one, or else perform H on qubit one only. Finally, let $c = c - 1$, go to (IV).

(IV) Firstly, remove the first two rows (X_1 and Z_1) entirely, and for all other rows, remove the first column in their X part and the first column in their Z part. We name the remaining matrix M_s, and for this M_s go to (I).

(V) *Transforming the commuting generator into* Z_1: If s is equal to 0, end; or else, arrange the first row so that its z_1 is equal to 1. Then transform this row to Z_1 according to the strategies mentioned in (III). Next remove this row entirely and for the remaining matrix, remove the first column both in the X part and Z part. Finally, name the new matrix M_i, assign $s = s - 1$, and for this M_i go to (V).

As for our example, after the implementations of steps (I), (II), and (III), M' in Eq. (4) will transform to

$$\boldsymbol{M'} = \left(\begin{array}{c|ccccc} X & I & I & I & I & I \\ Z & I & I & I & I & I \\ \hline I & I & Z & I & X & Z \\ I & X & Z & I & Z & I \\ I & Y & Y & X & Z & Z \\ I & Y & Y & Y & I & X \end{array}\right), \tag{10}$$

[5] The i-th qubit is the control qubit and the j-th is the target one.

[6] X_i means that except for the i-th entry being X, all other entries are I. The definition of Z_i is similar.

[7] We use x_i to denote the i-th entry in the X part of the concerning generator, and similarly for z_i.

and c becomes 1. Thus for the bottom right part of Eq. (10) named M_s in our algorithm, steps (II) and (III) will be implemented giving $c = 0$ and

$$\boldsymbol{M'} = \left(\begin{array}{cc|cccc} X & I & I & I & I & I \\ Z & I & I & I & I & I \\ I & X & I & I & I & I \\ I & Z & I & I & I & I \\ \hline I & I & Y & X & X & Z \\ I & I & Y & Y & X & X \end{array}\right). \tag{11}$$

Now c is equal to 0 and s is equal to 2, thus for the bottom right part of M' in Eq. (11), step (V) will be implemented giving $s = 1$ and

$$\boldsymbol{M'} = \left(\begin{array}{ccc|ccc} X & I & I & I & I & I \\ Z & I & I & I & I & I \\ I & X & I & I & I & I \\ I & Z & I & I & I & I \\ I & I & Z & I & I & I \\ \hline I & I & Z(I) & Z & Z & X \end{array}\right). \tag{12}$$

Finally, step (V) is implemented again for the bottom right part of Eq. (12), named M_i in our algorithm, yielding the final required matrix (the left part of Eq. (5)) and $s = 0$, end.

Consequently, the following quantum circuit (Fig. 1) can be obtained according to the procedure mentioned above.

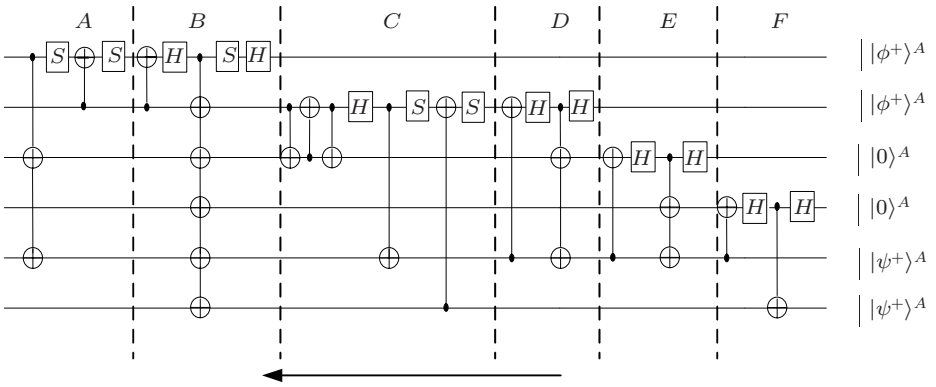


Fig. 1. Quantum circuit directly derived from the algorithm mentioned in the text to encode the sent quantum code which is associated with the noncommuting group $\widehat{M'}$ (Eq. (4) (the left part of Eq. (13)). $|\phi^+\rangle^A$ indicates the pre-shared entanglement pair possessed by Alice, $|0\rangle$ indicates the ancilla qubit, and $|\psi\rangle^A$ indicates the qubit being sent by Alice. After the implementation of A section, the first row of M' will be transformed to X_1. After the implementation of B section, the second row of M' will be transformed to Z_1, similarly for C, D, E, F sections. After the implementation of C and D, M will be transformed into Eq. (11). After the implementation of E, M will be transformed into Eq. (12). After the implementation of the last section F, M will be transformed into the required matrix (the left part of Eq. (5)). Thus it is obvious that the reverse of this quantum circuit will transform the left part of $\widehat{M}$ (Eq. (5)) into the left part of $\widehat{M'}$ (Eq. (13)), which is just the encoding procedure for an EAQECC, where the extra qubit is assumed to be possessed by Bob and to be error-free.

If Alice applies the quantum circuit reversely on her sending qubits, the corresponding stabilizer will be transformed from $\widehat{M}$ (Eq. (5)) to to $\widehat{M'}$ (Eq. (13)), for

$$\begin{array}{lcccccc|cc} \widehat{M'}_1 = & X & Z & X & I & X & I & X & I \\ \widehat{M'}_2 = & X & X & I & X & I & X & Z & I \\ \widehat{M'}_3 = & Z & I & Z & I & X & Z & I & X \\ \widehat{M'}_4 = & Z & Y & Z & I & Z & I & I & Z \\ \widehat{M'}_5 = & Z & X & Y & X & Z & Z & I & I \\ \widehat{M'}_6 = & X & Y & Z & Y & X & X & I & I \end{array}. \tag{13}$$

If the qubits held by Bob are assumed to be error-free, the error-correcting capacity of this EA quantum code will rely on the structure of the left part of $\widehat{M'}$, which is exactly the quantum counterpart of the check matrix of our concerning classical quarternary code. As pointed out in Ref. [14], the better the classical quarternary code is, the better the corresponding EA quantum code will be.

Next, we show some key points in our iterative algorithm based on the two theorems. First, in step (III), we need not arrange z_1 to be equal to 1. This is because after the implementation of step (II), z_1 is already 1. If not, it violates the two theorems. For the same reason, after the implementations of steps (II) and (III), besides the first two generators

which are transformed to X_1 and Z_1 respectively, the first entries of all other generators must be I (see Eq. (10)). Whereby, the operations on the new matrix will not destroy the transformations we have done. This is the key of our iterative strategy being feasible. Second, after the implementation of step (V), the first generator of the new matrix is Z_1, however, for the remaining generators, their first entries could be Z but not I (see Eq. (12)). In light of the properties of generators, $\{g_1, g_2, \ldots, g_i, \ldots, g_j, \ldots, g_r\} = \{g_1, g_2, \ldots, g_i, \ldots, g_i g_j, \ldots, g_r\}$, we can transform these entries to I by multiplying the first generator of the new matrix.[8)] This is why we can remove the first entries of the remaining generator in step (V), and also the reason why our iterative idea is applied in step (V).

2.3. Encoding complexity analysis

Considering the importance of encoding complexity, it is interesting to analyse the complexity yielded by our algorithm. According to our algorithm, we first choose an anti-commuting pair in the symplectic group, and transform the two concerning generators into X_1 (step (II)) and Z_1 (step (III)), respectively. Obviously, the initially arranged procedure in step (II) will consume three $CNOT$ gates and one H gate at most, which corresponds to the case where there is no X entry in this generator and the first entry is I. The following clearance of other entries in step (II) will consume $2(n-1)$ $CNOT$ gates and two S gates at most. Similarly, when we transform a generator with n entries into Z_1, $2(n-1)$ $CNOT$ gates, two H gate and one S gate will be consumed. Thus transforming a pair of anti-commuting generators, each with n entries, into X_1 and Z_1, respectively, $(4n-1)$ $CNOT$ gates, three S gates, and three H gates will be consumed. Because the transformation from a generator in the isotropic subgroup into Z_1 (step (V)) is similar to the step (III)), except an extra required procedure to arrange the Z_1 of this row is 1. Therefore this transformation will consume $(2n+1)$ $CNOT$ gates, three H gates, and one S gate totally.

Now suppose Alice encodes k qubits in n physical bits using c pre-shared entangled pairs and s ancilla qubits. The encoding circuit yielded by our algorithm will, at most, consume $[(4c+2s)n - 2c^2 - 2s \times c + c - s^2 + 2s]$ $CNOT$ gates, $3(c+s)$ H gates, and $(3c+s)$ S gates. Because $c+s+k=n$, we say that the encoding complexity of EAQECCs is $O(n^2)$.

3. Conclusion

In this paper, we present an iterative algorithm to address the encoding of EAQECCs, from which a quantum circuit composed of $CNOT$, H, and S gates can be derived directly with complexity $O(n^2)$. In addition, we demonstrate the feasibility of our iterative encoding algorithm.

References

[1] Liu W J, Chen H W, Ma T H, Li Z Q, Liu Z H and Hu W B 2009 *Chin. Phys.* B **18** 4105

[2] Zhang S, Wang J, Zhang Q and Tang C J 2009 *Acta Phys. Sin.* **57** 73 (in Chinese)

[3] Chen P, Deng F G and Long G L 2006 *Chin. Phys.* **15** 2228

[4] Wang Y J, Bai B M and Wang X M 2010 *Acta Phys. Sin.* **59** 7591 (in Chinese)

[5] Wang Y J, Bai B M, Peng J Y and Wang X M 2011 *Acta Phys. Sin.* **60** 030306 (in Chinese)

[6] Gottesman D 1997 *Stabilizer Codes and Quantum Error Correction* (Ph. D. thesis) (Los Angeles: Caltech)

[7] Calderbank A R and Shor P W 1996 *Phys. Rev.* A **54** 1098

[8] Steane A M 1996 *Phys. Rev. Lett.* **77** 793

[9] Wen K and Long G L 2010 *Int. J. Quantum Inform.* **18** 697

[10] Li Y, Zeng G H and Moon H L 2009 *Chin. Phys.* B **18** 4154

[11] Li Z and Xing L J 2008 *Chin. Phys.* B **17** 28

[12] Calderbank A R, Rains E M, Shor P W and Sloane N J A 1998 *IEEE Trans. Inform. Theory* **47** 1369

[13] MacKay D J C, Mitchison G J and McFadden P L 2004 *IEEE Trans. Inform. Theory* **50** 2315

[14] Brun T, Devetak I and Hsieh M H 2006 *Science* **314** 436

[15] Hsieh M H, Brun T and Devetak I 2009 *Phys. Rev.* A **79** 032304

[16] Wang Y J, Bai B M, Zhao W B and Wang X M 2009 *Int. J. Quantum Inform.* **7** 1373

[17] Wang Y J, Sanders B C, Bai B M and Wang X M 2009 *arXiv*: 0912.4546 [quant-ph]

[18] Gottesman D 2007 http://www.perimeterinstitute.ca /personal/dgottesman/QECC2007 [2007]

[19] Fattal D, Cubitt T S, Yamamoto Y, Bravyi S and Chuang I L 2004 *arXiv*: 0406168 [quant-ph]

[20] Cao H X, Li L, Chen Z L, Zhang Y and Guo Z H 2010 *Chin. Sci. Bull.* **55** 2122

[21] Nielsen M A and Chuang I L 2000 *Quantum Computation and Quantum Information* (Cambridge: Cambridge University Press) pp. 459–462

[8)] This is an equivalent mathematic transformation, no physical operation is needed, which means in practice we do not need to care whether it is I or Z.