Leveraging Automorphisms of Quantum Codes for Fault-Tolerant Quantum Computation

Markus Grassl
Centre for Quantum Technologies
National University of Singapore
Email: Markus.Grassl@nus.edu.sg

Martin Roetteler
NEC Laboratories America
Princeton, NJ, U.S.A.
Email: mroetteler@nec-labs.com

***Abstract*—Fault-tolerant quantum computation is a technique that is necessary to build a scalable quantum computer from noisy physical building blocks. Key for the implementation of fault-tolerant computations is the ability to perform a universal set of quantum gates that act on the code space of an underlying quantum code. To implement such a universal gate set fault-tolerantly is an expensive task in terms of physical operations, and any possible shortcut to save operations is potentially beneficial and might lead to a reduction in overhead for fault-tolerant computations. We show how the automorphism group of a quantum code can be used to implement some operators on the encoded quantum states in a fault-tolerant way by merely permuting the physical qubits. We derive conditions that a code has to satisfy in order to have a large group of operations that can be implemented transversally when combining transversal CNOT with automorphisms. We give several examples for quantum codes with large groups, including codes with parameters $[\![8,3,3]\!]$, $[\![15,7,3]\!]$, $[\![22,8,4]\!]$, and $[\![31,11,5]\!]$.**

I. Introduction

Quantum error-correcting codes (QECC) are essential ingredients for the realization of quantum computing devices. In addition to the mere error correction, it is also important that quantum operations can be implemented in a fault-tolerant way, i. e., the operations preserve the code space and if an operation fails, the errors remain local [1], [2]. Several schemes are known for universal fault-tolerant quantum computing, including schemes that are based on distance-three codes [3] such as for instance the concatenated Steane code [4], [5], concatenated error detecting codes [6], or the Bacon-Shor codes [7]. Quite recently, the surface code—a stabilizer code that exhibits one of the highest reported thresholds that exceed 1% for a standard 2D lattice of physical qubits and independent depolarizing noise—has gained a lot of attention [8], [9]. So far, most of the schemes for fault-tolerant quantum computing encode very few qubits per code block; in the case of concatenated codes, typically QECCs are chosen that encode only a single qubit per code block. In [10], however, the implementation of fault-tolerant gates for codes encoding $k>1$ based on teleportation and using an auxiliary code block is discussed.

In this paper we present a general method that allows the implementation of operations in a fault-tolerant manner for codes encoding several qubits without using an auxiliary code block. Like in the single-qubit case, CSS codes appear to be well suited for our methods, but they can be applied to any stabilizer code. The basic idea is that code automorphisms can give rise to non-trivial logical operations on the encoded quantum information that can be executed by merely permuting, or what arguably is simpler in a practical implementation, simply relabeling of the physical qubits. While such operations cannot *per se* give rise to a universal gate set, for which additional techniques such as state distillation are essential, our construction can nevertheless lead to operations that can be performed at basically zero cost. This might lead to overhead reductions, in particular for fault-tolerant quantum computations on long block codes, provided they exhibit large automorphism groups or automorphism groups with suitable structure.

II. CSS Codes and their Automorphism Group

First we consider the special case of CSS codes based on a classical linear $C=[n,k_1,d_1]$ which is contained in its dual code $C^\perp=[n,n-k_1,d_2]$. The (permutation) automorphism group $\mathrm{Aut}(C)$ of C is the set of all permutations $\pi\in S_n$ that preserve the code, i. e., (see also [11])

$$\forall \boldsymbol{c}\in C\colon \boldsymbol{c}^\pi=(c_{\pi(1)},\ldots,c_{\pi(n)})\in C. \tag{1}$$

It turns out that $\mathrm{Aut}(C)=\mathrm{Aut}(C^\perp)$.

Lemma 1. *Let $B=\{\boldsymbol{b}_1,\ldots,\boldsymbol{b}_{n-2k_1},\ldots,\boldsymbol{b}_{n-k_1}\}$ be a basis of $C^\perp$ such that $B_0=\{\boldsymbol{b}_{n-2k_1+1},\ldots,\boldsymbol{b}_{n-k_1}\}$ is a basis of C. With respect to the basis B, the automorphism group $\mathrm{Aut}(C)$ has a linear representation in the block-triangular form*

$$\begin{aligned}\mathrm{Aut}(C)&\to\mathrm{GL}(n-k_1,2)\\ \pi&\mapsto T(\pi)=\left(\begin{array}{c|c}T_1(\pi)&T_2(\pi)\\\hline 0&T_3(\pi)\end{array}\right).\end{aligned}\tag{2}$$

Recall that the basis states of the CSS code $\mathcal{C}=[\![n,k,d]\!]$, where $k=n-2k_1$, based on the code C are given by

$$|\psi_{\boldsymbol{v}}\rangle=\frac{1}{\sqrt{|C|}}\sum_{\boldsymbol{c}\in C}|\boldsymbol{c}+\boldsymbol{v}\rangle,\tag{3}$$

where the vectors $\boldsymbol{v}=\sum_{i=1}^k\beta_i\boldsymbol{b}_i$ are representatives of the cosets of C in $C^\perp$. If we apply a permutation $\pi\in\mathrm{Aut}(C)$ to the qudits of a basis state of the CSS code, from eq. (2) it follows that

$$|\psi_{\boldsymbol{v}}\rangle^\pi = \frac{1}{\sqrt{|C|}}\sum_{\boldsymbol{c}\in C}|\boldsymbol{c}^\pi + \boldsymbol{v}^\pi\rangle = \frac{1}{\sqrt{|C|}}\sum_{\boldsymbol{c}\in C}|\boldsymbol{c} + \boldsymbol{v}'\rangle = |\psi_{\boldsymbol{v}'}\rangle, \tag{4}$$

where

$$\boldsymbol{v}' = \sum_{i=1}^{k}\beta_i'\boldsymbol{b}_i \quad\text{and}\quad \beta_i' = \sum_{j=1}^{k}\big(T_1(\pi)\big)_{ij}\beta_j. \tag{5}$$

Note that the basis state $|\psi_{\boldsymbol{v}}\rangle$ corresponds to the encoding of the computational basis state $|\boldsymbol{\beta}\rangle$. Hence we can label the basis states of the CSS code $\mathcal{C}$ by the vector $\boldsymbol{\beta} = (\beta_1,\ldots,\beta_k)^T$. Then we have

$$|\boldsymbol{\beta}\rangle^\pi = |T_1(\pi)\boldsymbol{\beta}\rangle, \tag{6}$$

i. e., the automorphism π of the classical code C gives rise to a permutation of the basis states of the CSS code $\mathcal{C}$ corresponding to the linear transformation $T_1(\pi)$. In summary we have:

Theorem 1. *Let $\mathcal{C}$ be a CSS code based on the classical code $C \le C^\perp$. Then the automorphism $\pi \in \operatorname{Aut}(C)$ corresponds to the linear operation $T_1(\pi)$ defined in eq. (2) on the logical basis states of $\mathcal{C}$.*

In the general situation, a CSS code $\mathcal{C}$ is based on nested classical codes $C_2 \subset C_1$, and the basis states of $\mathcal{C}$ correspond to the cosets of C_2 in C_1. In general, the automorphism groups $\operatorname{Aut}(C_1)$ and $\operatorname{Aut}(C_2)$ need not be equal. However, when we consider their intersection, we obtain the following result:

Theorem 2. *Let $\mathcal{C}$ be a CSS code based on nested classical codes $C_2 \le C_1$. Then a joint automorphism $\pi \in \operatorname{Aut}(C_1)\cap\operatorname{Aut}(C_2)$ corresponds to a linear operation $T_1(\pi)$ on the logical basis states of $\mathcal{C}$, defined analogously to eq. (2).*

Note that these operations can be implemented by permuting the qubits or just by relabeling them. Below we will show that by a similar argument, the (permutation) automorphism group of an additive code corresponding to a stabilizer code gives rise to symplectic operations on the logical operators of the stabilizer code. We would also like to point out that while automorphism groups of additive codes have been investigated before, see e. g., [12], [13], [14], the idea to leverage automorphisms to perform large sets of encoded logical operations does not seem to have been investigated much.[1]

III. Combining Automorphisms and Transversal Operations

For CSS codes, applying the controlled-NOT (CNOT) operation transversally is an operation preserving the space of two copies of the code. More precisely, we have

$$\mathrm{CNOT}^{\otimes n}\big(|\psi_{\boldsymbol{v}_1}\rangle|\psi_{\boldsymbol{v}_2}\rangle\big) = |\psi_{\boldsymbol{v}_1}\rangle|\psi_{\boldsymbol{v}_1+\boldsymbol{v}_2}\rangle, \tag{7}$$

where $\mathrm{CNOT}^{\otimes n}$ should be understood as applying CNOT-gates to the corresponding qudits in both code blocks. In terms of the encoded basis states, we have

$$\mathrm{CNOT}^{\otimes n}\big(|\boldsymbol{\beta_1}\rangle|\boldsymbol{\beta_2}\rangle\big) = |\boldsymbol{\beta_1}\rangle|\boldsymbol{\beta_1}+\boldsymbol{\beta_2}\rangle, \tag{8}$$

i. e., the transversal CNOT corresponds to the linear $2k \times 2k$ matrix

$$\left(\begin{array}{c|c} I & 0 \\ \hline I & I \end{array}\right). \tag{9}$$

In the following we assume that the CNOT-gates can not only be applied to the corresponding pairs of qudits in each code block, but between any pair of qudits. Then we can combine the operations on the code arising from the automorphism group of the underlying classical code and the transversal CNOT.

Theorem 3. *Given a CSS code $\mathcal{C} = [\![n,k,d]\!]$ derived from a linear code $C \le C^\perp$ with automorphism group $\operatorname{Aut}(C)$, one can realize the following group G_{12} of linear transformations on $2k$ encoded qudits in a fault-tolerant manner:*

$$G_{12} = \left\langle \left(\begin{array}{c|c} I & 0 \\ \hline I & I \end{array}\right), \left(\begin{array}{c|c} I & I \\ \hline 0 & I \end{array}\right), \right. \tag{10}$$

$$\left. \left(\begin{array}{c|c} T_1(\pi_1) & 0 \\ \hline 0 & T_1(\pi_2) \end{array}\right) : \pi_1,\pi_2 \in \operatorname{Aut}(C) \right\rangle.$$

The first two generators of G_{12} are the transversal CNOT with all controls in the first or second code block, respectively. While we cannot make a general statement about the relation between the automorphism group $\operatorname{Aut}(C)$ and the group G_{12}, we have the following observation.

Lemma 2. *The group G_{12} contains all matrices of the form*

$$\left(\begin{array}{c|c} I & A \\ \hline 0 & I \end{array}\right) \quad\text{and}\quad \left(\begin{array}{c|c} I & 0 \\ \hline A & I \end{array}\right), \tag{11}$$

where A is an arbitrary element of the $\mathbb{Z}$-algebra generated by the matrices $T_1(\pi_j)$, i. e.,

$$A = \sum_{\pi\in\operatorname{Aut}(C)} \alpha_\pi T_1(\pi), \qquad \alpha_\pi \in \mathbb{Z}. \tag{12}$$

Hence we can in particular realize transformations of the form

$$|\boldsymbol{\beta}_1\rangle|\boldsymbol{\beta}_2\rangle \mapsto |\boldsymbol{\beta}_1\rangle|A\boldsymbol{\beta}_1+\boldsymbol{\beta_2}\rangle. \tag{13}$$

Proof: First, note that

$$\left(\begin{array}{c|c} T_1(\pi) & 0 \\ \hline 0 & I \end{array}\right)\left(\begin{array}{c|c} I & I \\ \hline 0 & I \end{array}\right)\left(\begin{array}{c|c} T_1(\pi) & 0 \\ \hline 0 & I \end{array}\right)^{-1} \tag{14}$$

$$= \left(\begin{array}{c|c} I & T_1(\pi) \\ \hline 0 & I \end{array}\right).$$

The products of these matrices and their inverses yield arbitrary integer linear combinations of the matrices $T_1(\pi_j)$ in the upper right block. The result for lower-triangular block matrices follows analogously. ■

[1] However, we would like to point out that the automorphism group of the quantum Hamming code of length 15 was used to aid fault-tolerant quantum computation in a talk given by J. Harrington at the QEC 2011 conference.

Theorem 4. *Assume that the group G_{12} contains all matrices of the form*

$$\left\{\left(\begin{array}{c|c} I & A \\ \hline 0 & I \end{array}\right) : A \in M_{n\times n}(\mathbb{F}_q)\right\} \quad \textit{and} \qquad (15)$$

$$\left\{\left(\begin{array}{c|c} I & 0 \\ \hline B & I \end{array}\right) : B \in M_{n\times n}(\mathbb{F}_q)\right\},$$

where $A, B \in M_{n\times n}(\mathbb{F}_q)$ are arbitrary matrices of the algebra of $n \times n$ matrices over the field $\mathbb{F}_q$. Then $G_{12} = \mathrm{SL}_{2n}(\mathbb{F}_q)$.

Proof: Let $E_{i,j}$ denote the $n \times n$ matrix which has the entry 1 in row i and column j, and is zero elsewhere. By assumption, the group G_{12} contains the following two matrices:

$$M_1 = \left(\begin{array}{c|c} I & \alpha E_{i,j} \\ \hline 0 & I \end{array}\right) \quad \text{and} \quad M_2 = \left(\begin{array}{c|c} I & 0 \\ \hline \beta E_{j,k} & I \end{array}\right) \qquad (16)$$

with $i \neq k$. We compute

$$M_2^{-1} M_1 M_2 M_1^{-1} = \left(\begin{array}{c|c} I + \alpha\beta E_{i,k} & 0 \\ \hline 0 & I \end{array}\right). \qquad (17)$$

By symmetry, we also get the same type of matrices in the lower right block, and in summary all elementary transvections with identity on the diagonal and a single non-zero off-diagonal entry. Furthermore, for $t \neq 0$ we obtain the following factorizations of diagonal matrices:

$$\left(\begin{array}{cc|cc} t & 0 & 0 & 0 \\ 0 & 1/t & 0 & 0 \\ \hline 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{array}\right) =$$

$$\left(\begin{array}{cc|cc} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \hline 0 & 0 & 1 & 0 \\ 0 & (t-1)/t & 0 & 1 \end{array}\right) \left(\begin{array}{cc|cc} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & -1 \\ \hline 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{array}\right)$$

$$\times \left(\begin{array}{cc|cc} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \hline 0 & 0 & 1 & 0 \\ (1-t)/t & 0 & 0 & 1 \end{array}\right) \left(\begin{array}{cc|cc} 1 & 0 & 0 & -t \\ 0 & 1 & 0 & 0 \\ \hline 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{array}\right)$$

$$\times \left(\begin{array}{cc|cc} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \hline 0 & 0 & 1 & 0 \\ (t-1)/t^2 & (1-t)/t & 0 & 1 \end{array}\right) \left(\begin{array}{cc|cc} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ \hline 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{array}\right) \qquad (18)$$

and

$$\left(\begin{array}{cc|cc} t & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \hline 0 & 0 & 1/t & 0 \\ 0 & 0 & 0 & 1 \end{array}\right) = \left(\begin{array}{cc|cc} 1 & 0 & t-1 & 0 \\ 0 & 1 & 0 & 0 \\ \hline 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{array}\right) \left(\begin{array}{cc|cc} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \hline 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{array}\right)$$

$$\times \left(\begin{array}{cc|cc} 1 & 0 & (1-t)/t & 0 \\ 0 & 1 & 0 & 0 \\ \hline 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{array}\right) \left(\begin{array}{cc|cc} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \hline -t & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{array}\right). \qquad (19)$$

The matrices of the form (18) and (19) generate all diagonal matrices with unit determinant. Together with the transvections in (16) and (17), they generate the full special linear group $\mathrm{SL}_2(\mathbb{F}_q)$. ■

IV. Examples

Good candidates for our construction are codes with large automorphism group or automorphism groups for which the representation given by $T_1(\pi)$ is irreducible or has only a few irreducible components of large dimension. Among those, Reed-Muller codes and cyclic codes are promising candidates.

A. CSS code $[\![15, 7, 3]\!]$

The 4th-order binary Hamming code is a linear code with parameters $[15, 11, 3]$ that contains its dual code $C = [15, 4, 8]$. The automorphism group of C is isomorphic to the alternating group A_8 of order 21600. The CSS construction yields a QECC $\mathcal{C} = [\![15, 7, 3]\!]$.

The linear action on the 7 logical qubits is given by the group

$$G_1 = \left\langle \begin{pmatrix} 1&0&0&1&1&0&1 \\ 1&1&0&0&1&0&0 \\ 1&1&1&0&1&1&1 \\ 1&1&0&0&0&1&0 \\ 0&1&0&0&1&0&1 \\ 0&0&0&1&1&0&1 \\ 1&1&0&0&1&1&0 \end{pmatrix}, \begin{pmatrix} 1&0&1&0&0&1&0 \\ 1&1&1&1&1&0&0 \\ 0&1&1&0&1&1&0 \\ 0&1&0&1&0&1&1 \\ 1&0&0&1&0&0&0 \\ 0&1&0&0&1&1&0 \\ 0&1&1&1&1&0&1 \end{pmatrix} \right\rangle. \qquad (20)$$

Combining the group $G_1 \times G_1$ with the transversal CNOTs, we get the group $G_{12} \cong \mathrm{SL}(12, 2) \times \mathrm{SL}(2, 2)$ with more than 2^{144} elements. What is even more, the block-diagonal subgroup $\tilde{G}_1$ of G_{12} that acts trivially on the second code block is isomorphic to the group $\mathrm{SL}(6, 2)$.

A closer inspection shows that the Hamming code contains the all-one vector which corresponds to the logical operator $X^{\otimes 15}$ on the code. Both are invariant under permutations. Hence on the subcode $[\![15, 6, 3]\!]$ of the original code, which is obtained by removing the all-one vector from the Hamming code, we can realize the full linear group $\mathrm{SL}(6, 2)$ on the encoded states as well as the full linear group $\mathrm{SL}(12, 2)$ on pairs of encoded states.

B. CSS code $[\![31, 11, 5]\!]$

The BCH code with parameters $[31, 21, 5]$ contains its dual $C = [31, 10, 12]$. The resulting CSS code has parameters $\mathcal{C} = [\![31, 11, 5]\!]$. The automorphism group of C is a group G_1 of order 155 isomorphic to $C_{31} \rtimes C_5$. However, when combining $G_1 \times G_1$ with the transversal CNOTs, we obtain the group G_{12} isomorphic to $\mathrm{SL}(10, 2) \times \mathrm{SL}(10, 2) \times \mathrm{SL}(2, 2)$ with more than 2^{199} elements. Restricted to one code block, we get the group $\tilde{G}_1 \cong \mathrm{SL}(5, 2) \times \mathrm{SL}(5, 2)$. Similar as for the CSS code $[\![15, 7, 3]\!]$, we find that the spaces of dimension 5, 5, and 1 stabilized by the automorphism group correspond to cyclic subcodes lying between the code C and $C^{\perp}$. On each of these subspaces, we can realize the full linear group, despite the fact that the automorphism group $\mathrm{Aut}(C)$ is relatively small.

C. CSS code $[\![22,8,4]\!]$

The classical self-orthogonal code $C=[22,7,8]$ generated by

$$G=\begin{pmatrix}
1&0&0&0&1&0&0&0&0&1&1&0&0&0&1&0&1&0&1&0&1&0\\
0&1&0&0&1&0&0&0&0&1&1&0&1&0&0&1&0&1&0&1&1&1\\
0&0&1&0&1&0&0&0&0&1&0&1&0&0&1&1&0&1&1&0&0&0\\
0&0&0&1&0&0&0&1&0&0&1&0&0&0&0&1&1&0&1&0&1&1\\
0&0&0&0&0&1&0&1&0&1&0&1&1&1&1&1&1&0&0&0&1&0\\
0&0&0&0&0&0&1&1&0&0&1&0&0&1&1&0&1&0&1&1&0&0\\
0&0&0&0&0&0&0&0&1&1&1&1&1&1&1&1&1&1&1&1&1&1
\end{pmatrix}\tag{21}$$

is contained in its dual $C^{\perp}=[22,15,4]$. Hence we obtain a CSS code $\mathcal{C}=[\![22,8,4]\!]$. The automorphism group of C has order 336 and is isomorphic to a semi-direct product of $\mathrm{PSL}(2,7)$ and Z_2. Although the group is relatively small, the action on the space of 8 logical qubits is an irreducible matrix group

$$G_1=\left\langle\begin{pmatrix}
1&1&0&1&0&1&1&0\\
0&1&1&1&1&1&0&0\\
0&1&1&0&1&1&0&1\\
1&1&1&0&0&0&0&0\\
1&0&1&0&1&1&0&0\\
1&1&0&1&1&1&0&1\\
0&0&1&0&0&1&0&0\\
1&0&1&0&0&1&1&0
\end{pmatrix},\begin{pmatrix}
1&1&1&0&0&0&0&1\\
0&1&0&1&1&0&1&0\\
0&1&0&0&1&0&1&1\\
1&0&1&0&1&0&0&1\\
1&1&0&0&0&0&1&1\\
1&1&1&0&1&1&0&1\\
0&0&0&0&0&0&1&0\\
1&0&1&1&0&0&0&0
\end{pmatrix}\right\rangle.\tag{22}$$

The matrices in G_1 span the full space of binary 8×8 matrices. Hence by Theorem 4, combining the group $G_1\times G_1$ with the transversal CNOTs we get the maximal possible group $\mathrm{SL}(16,2)$.

D. Stabilizer code $[\![8,3,3]\!]$

In the following we illustrate that our method does not only apply to CSS codes, but can be extended to general stabilizer codes, albeit with some restrictions.

There is a stabilizer code $\mathcal{C}=[\![8,3,3]\!]$ whose five generators of the stabilizer, the three logical Z-operators, and the three logical X-operators correspond to the following vectors (top to down, respectively) over $GF(4)$:

$$\left(\begin{array}{c}
g_1\\ g_2\\ g_3\\ g_4\\ g_5\\ \hline \overline{Z}_1\\ \overline{Z}_2\\ \overline{Z}_3\\ \hline \overline{X}_1\\ \overline{X}_2\\ \overline{X}_3
\end{array}\right)=\left(\begin{array}{cccccccc}
1&0&\omega&0&\omega^2&\omega&1&\omega^2\\
\omega&0&\omega&1&0&\omega^2&\omega^2&1\\
0&1&\omega&\omega&\omega^2&1&\omega^2&0\\
0&\omega&0&\omega^2&\omega&1&1&\omega^2\\
0&0&1&\omega^2&1&\omega&\omega^2&\omega\\ \hline
0&0&\omega&1&\omega&\omega^2&1&\omega^2\\
0&0&0&\omega&0&\omega&\omega&\omega\\
0&0&\omega&0&\omega&\omega&0&\omega\\ \hline
0&0&0&1&\omega&0&\omega^2&0\\
0&0&0&0&1&0&\omega&\omega^2\\
0&0&\omega&0&0&0&\omega^2&1
\end{array}\right)\tag{23}$$

Here $\omega\in GF(4)$ obeys the relation $\omega^2=\omega+1$, and Pauli matrices X, Y, and Z correspond to 1, ω^2, and ω, respectively. The permutation automorphism group of $\mathcal{C}$ is isomorphic to the group $\mathrm{AGL}(1,8)$ of order 56. It preserves the space corresponding to the generators g_ν of the stabilizer. On the six-dimensional symplectic space corresponding to logical operators $\overline{Z}_i$ and $\overline{X}_j$ of $\mathcal{C}$, we have the following matrix representation of $\mathrm{Aut}(\mathcal{C})$:

$$G_1=\left\langle\begin{pmatrix}
0&1&0&0&0&0\\
0&0&1&0&0&0\\
1&0&1&0&0&0\\
\\
1&0&0&0&1&0\\
0&1&1&1&0&1\\
0&1&0&1&0&0
\end{pmatrix},\begin{pmatrix}
1&0&0&0&0&0\\
0&1&0&0&0&0\\
0&0&1&0&0&0\\
\\
0&1&0&1&0&0\\
1&0&0&0&1&0\\
0&0&0&0&0&1
\end{pmatrix}\right\rangle$$

Here we used extra space to indicate the blocks corresponding to the logical Z- and X-operators. Note that with this choice of logical operators, the generators of G_1 are lower block-triangular matrices and hence the space corresponding to the logical Z-operators is preserved.

Unlike the situation for CSS codes, the transversal CNOT-gate does not preserve stabilizer codes in general. Therefore, Theorems 3 and 4, which play an instrumental role in obtaining a large group of fault-tolerant operations, do not apply. So we have to look for stabilizer codes which have a larger symmetry group to start with, or codes which support some transversal operations on two code blocks.

V. Code Families

We briefly discuss the situation for CSS codes based on Reed-Muller codes or cyclic codes.

A. Reed-Muller codes

The r-th order binary Reed-Muller code $\mathrm{RM}(r,m)$ of length $n=2^m$, for $0\le r\le m$ is obtained by the evaluation of all Boolean functions in m variables of maximal degree r (see, e. g., [11]). The automorphism group of $\mathrm{RM}(r,m)$ contains the group $\mathrm{AGL}(m,2)$ of all affine transformations on $\mathbb{F}_2^m$. As affine transformations preserve the degree of Boolean functions, it follows that the automorphism group also preserves the cosets of $\mathrm{RM}(r,m)$ in $\mathrm{RM}(r+1,m)$. Hence, if a CSS code is based on the nested codes $\mathrm{RM}(r,m)\subset RM(r+s,m)$, the action of $\mathrm{AGL}(m,2)$ on the CSS code will not mix the blocks of logical qubits corresponding to homogeneous Boolean functions of fixed degree. Additional automorphisms or other techniques are needed to implement operations between the blocks.

B. Cyclic codes

Recall that every linear binary cyclic code of odd length n can be uniquely described by a generator polynomial $g(X)$ that divides X^n-1. Given two nested cyclic codes $C_2=[n,k_2]\subset C_1=[n,k_1]$, their generator polynomials obey the relation $g_2(X)=g_1(X)h(X)$, where $h(X)$ is some factor of X^n-1 of degree k_1-k_2. Assume that the polynomial $h(X)$ has irreducible factors $h_i(X)$ of degree δ_i, respectively. Then the coset space C_1/C_2 can be decomposed into spaces of dimension δ_i which are preserved by the action of the cyclic group Z_n of order n. In turn, for a cyclic CSS code based on

$C_2 \subset C_1$, the cyclic shift gives rise to operations on blocks with δ_i logical qubits. If $n < \delta_i^2$, the matrices corresponding to the action on these blocks do not generate the full algebra of $\delta_i \times \delta_i$ matrices. Hence we cannot apply Theorem 4, and it is not clear whether we can implement the full group of linear transformations on that block with δ_i logical qubits. The situation changes when there are more automorphisms than just the cyclic shift, as we have seen in the examples in Sections IV-A and IV-B.

VI. Towards the full Clifford Group

When the conditions in Theorem 4 are met, we can implement all linear transformations on a single block of k logical qudits as well as on any number of such blocks. Using tensor products of local X-operations corresponding to coset representatives $\boldsymbol{v}$, we can implement affine shifts on the logical qudits, and hence all affine transformations.

If a CSS code is based on a classical self-orthogonal code $C \leq C^\perp$, we can apply a local Fourier transformation transversally on all qudits, resulting in a simultaneous Fourier transformation on all logical qudits. This operation will interchange the role of the logical X- and Z-operations. In order to implement all Clifford operations on the logical qudits, additional transformations that mix X- and Z-operations are required.

If a CSS code is based on a self-orthogonal doubly-even binary code, applying the local transformation $P = \operatorname{diag}(1, i)$, where $i^2 = -1$, transversally induces an operation on the code states (3) given by

$$P^{\otimes n}|\psi_{\boldsymbol{v}}\rangle = i^{\operatorname{wgt}(\boldsymbol{v})}|\psi_{\boldsymbol{v}}\rangle. \tag{24}$$

Hence depending on $\operatorname{wgt}(\boldsymbol{v}) \bmod 4$, different powers of P are applied to the corresponding logical state. The favorable situation is when we indeed have a different action on the logical qubits. In that case, the combination with permutations of the logical qubits (which are in particular linear transformations) yields a larger group of transformations on the logical qudits. The very group, however, depends on the particular code.

VII. Conclusions

We proposed a general method that allows the implementation of operations in a fault-tolerant manner for codes encoding several qubits. In Theorem 4 we presented a sufficient condition on the automorphism group of a quantum code such that all linear transformations on the logical qubits can be implemented by permutations of the qubits and transversal CNOT operations. We applied this to a set of examples, including quantum codes with parameters $[\![8,3,3]\!]$, $[\![15,7,3]\!]$, $[\![22,8,4]\!]$, and $[\![31,11,5]\!]$. Furthermore, we discussed the prospects for applying this framework to infinite families of quantum block codes, such as the Reed-Muller codes and cyclic codes. There are several open questions that are implied by these observations: (i) Can we find more examples of quantum codes for which the complete set of linear transformations can be implemented following Theorems 3 and 4? In particular, it would be interesting to know if code families with this property exist that are asymptotically good. (ii) Can we find codes—or families of codes—for which we can implement not only all linear transformations, but the full Clifford group on k logical qudits extending the results shown here?

Acknowledgments

Supported in part by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior National Business Center Contract number DllPC20l66. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoI/NBC or the U.S. Government.

The Centre for Quantum Technologies (CQT) is a Research Centre of Excellence funded by the Ministry of Education and the National Research Foundation of Singapore.

References

[1] D. Gottesman, "Stabilizer codes and quantum error correction," Ph.D. dissertation, Caltech, 1997, see also: arXiv preprint quant-ph/9705052.

[2] M. Nielsen and I. Chuang, *Quantum Computation and Quantum Information*. Cambridge, UK: Cambridge University Press, 2000.

[3] P. Aliferis, D. Gottesman, and J. Preskill, "Quantum accuracy threshold for concatenated distance-3 codes," *Quant. Information and Computation*, vol. 6, no. 2, pp. 97–165, 2006.

[4] A. Steane, "Overhead and noise threshold of fault-tolerant quantum error correction," *Phys. Rev. A*, vol. 68, p. 042322, 2003.

[5] K. M. Svore, D. P. DiVincenzo, and B. M. Terhal, "Noise threshold for a fault-tolerant two-dimensional lattice architecture," *Quant. Information and Computation*, vol. 7, no. 4, pp. 297–318, 2007.

[6] E. Knill, "Quantum computing with realistically noisy devices," *Nature*, vol. 434, pp. 39–44, 2005.

[7] P. Aliferis and A. Cross, "Subsystem fault tolerance with the Bacon-Shor code," *Phys. Rev. Lett.*, vol. 98, p. 220502, 2006.

[8] E. Dennis, A. Alexei Kitaev, A. Landahl, and J. Preskill, "Topological quantum memory," *J. Math. Phys.*, vol. 43, pp. 4452–4505, 2002.

[9] A. G. Fowler, M. Mariantoni, J. M. Martinis, and A. N. Cleland, "Surface codes: Towards practical large-scale quantum computation," *Phys. Rev. A*, vol. 86, p. 032324, 2012.

[10] A. M. Steane and B. Ibinson, "Fault-tolerant logical gate networks for Calderbank-Shor-Steane codes," *Phys. Rev. A*, vol. 72, p. 052335, 2005.

[11] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error–Correcting Codes*. Amsterdam: North–Holland, 1977.

[12] A. R. Calderbank, E. M. Rains, P. W. Shor, and N. J. A. Sloane, "Quantum error correction via codes over GF(4)," *IEEE Trans. Inform. Theory*, vol. 44, pp. 1369–1387, 1998.

[13] E. M. Rains, "Quantum codes of minimum distance two," *IEEE Trans. Inform. Theory*, vol. 45, no. 1, pp. 266–271, 1999.

[14] B. Zeng, A. Cross, and I. L. Chuang, "Transversality versus universality for additive quantum codes," *IEEE Trans. Inform. Theory*, vol. 57, no. 9, pp. 6272–6284, 2011.