THE TORIC CODE

ALICE WANG

Abstract. The toric code is a fundamental model for topologically-encoded error correction in quantum computation. This paper aims to build up the mathematical formalisms describing the toric code from both an algebraic and topological perspective, following the structure of [1]. First we describe the stabilizer formalism and general principles of homology. Then we discuss the topological operators that construct the toric code and the key concept of abelian anyons, quasiparticles which can be used for physically implementing the toric code. This paper assumes knowledge of linear algebra, as well as some basic knowledge of logic and abstract algebra.

Contents

1. Introduction

Quantum error correction is a vital step in the realization of quantum computing. Inherent systematic faults such as decoherence cause unwanted changes in the physical quantum state and hence alter the stored bits of information. Topological quantum codes (first described in [6]), which implement quantum bits through operators on a lattice, can detect and correct errors through their inherent topological properties. Thus, topological codes allow for the realization of universal quantum computing with inherent fault tolerance [6].

In this paper, we first describe the stabilizer formalism for quantum error correction. Following this and a discussion of basic homology, we introduce the toric code, the fundamental topological error-correcting code, and define the anyon, a quasiparticle critical for physical implementation of the toric code.

Date: September 17, 2024.

As in the broader field of quantum computing, we use Dirac or "bra-ket" notation. A mathematically-rigorous look into Dirac notation is beyond the scope of this paper.[1]

1.1. **Preliminary definitions.** In classical computing, information is constructed out of *bits*, which have a value of either 1 or 0. Quantum physics allows us to describe *qubits*, or quantum bits, as a superposition of orthornormal basis vectors $|0\rangle$ and $|1\rangle$. These vectors are called *code basis states* and correspond physically to the ground and excited states, respectively [9].

Definition 1.1. The *qubit* $|\psi\rangle$ is described by the vector $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle = \begin{bmatrix}\alpha\\ \beta\end{bmatrix}$, where $\alpha, \beta \in \mathbb{Z}$ such that $|\alpha|^2 + |\beta|^2 = 1$.

Logical qubits should be distinguished from *physical qubits*. Physical qubits derive from the physical system and are therefore vulnerable to alteration by errors.[2] Logical qubits $|\psi\rangle_L$ are constructed using a repetition code of physical qubits. For example, the three-qubit repetition code is given by

$$|0\rangle_L = |000\rangle, |1\rangle_L = |111\rangle, |\psi\rangle_L = \alpha|0\rangle_L + \beta|1\rangle_L.$$

The value of a logical qubit is equal to the value of the majority of physical qubits within it. Hence, logical qubits ignore errors on individual physical qubits (i.e. under the three-qubit repetition code, $|110\rangle$ would be corrected to $|111\rangle$) [1].

Definition 1.2. The *Pauli operators* act on single physical qubits and are given by

$$\sigma_x = \begin{bmatrix}0 & 1\\ 1 & 0\end{bmatrix}, \sigma_y = \begin{bmatrix}0 & -i\\ i & 0\end{bmatrix}, \sigma_z = \begin{bmatrix}1 & 0\\ 0 & -1\end{bmatrix}, I = \begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}. \tag{1.3}$$

By inspection, for any Pauli operator σ and its conjugate transpose σ^*:

(1) σ either commutes or anticommutes with other Pauli operators,
(2) σ has eigenvalues of 1 and -1,
(3) σ is *self-inverse* ($\sigma^2 = I$) and *Hermitian* ($\sigma = \sigma^*$). Therefore, σ is *unitary* ($\sigma\sigma^* = I$).

The Pauli operators form a basis for the real vector space of $\{M \in \mathbb{Z}_{2x2} \| \ M = M^*\}$, the set of all possible single-qubit operators. Additionally, the set

$$\{\pm\sigma_x, \pm\sigma_y, \pm\sigma_z, \pm I, \pm i\sigma_x, \pm i\sigma_y, \pm i\sigma_z, \pm iI\}$$

forms a group under the group operation of matrix multiplication [1]. This can be shown using the identity

$$\sigma_y = i\sigma_x\sigma_z. \tag{1.4}$$

Definition 1.5. The tensor product of n Pauli operators, interpreted as an attachment of a single Pauli operator to each of n qubits (further discussed in section 3.3), is called an *n-qubit Pauli operator*.

[1] A more detailed look into Dirac notation can be found in [7].
[2] As in Definition 1.1, we denote physical qubits by "ket" vectors without subscripts.

Based on the properties of single Pauli operators, it can be shown that the n-qubit Pauli operators have properties analogous to (1)-(3). Additionally, if given prefactors of $\pm i$ and ± 1, they also form a group under the group operation of matrix multiplication [1].

2. The stabilizer formalism

The stabilizer formalism defines two sets of operators that act on the quantum state by matrix multiplication. The first is a set of stabilizer operators that correspond physically to taking measurements of the quantum state [1].

Definition 2.1. Let $\mathcal{N}$ be the Hilbert space of all possible physical states of a quantum code. Suppose $\mathcal{N}$ has basis states $\{|\psi_j\rangle \parallel j \in J\}$. An n-qubit Pauli operator P_k is a *stabilizer operator* if for all $j \in J$, $P_k |\psi_j\rangle = |\psi_j\rangle$.

Proposition 2.2. *The set of all stabilizer operators P is an abelian group under the group operation of matrix multiplication, called the* stabilizer group.

Proof. As previously shown, the set of all n-qubit Pauli operators (with prefactors $\pm i$ and ± 1) is a group under the group operation of matrix multiplication. Note that P is a subset of this set.

For any $P_1, P_2 \in P$ and $j \in J$, $P_1 P_2 |\psi_j\rangle = P_1 |\psi_j\rangle = |\psi_j\rangle$. Then by Definition 2.1, $P_1 P_2 \in P$, so P is closed under matrix multiplication. Assume for the sake of contradiction that P_1 and P_2 do not commute. Then they must anticommute, so $P_1 P_2 = -P_2 P_1$. Since $P_2 P_1 \in P$:

$$\forall j \in J, P_1 P_2 |\psi_j\rangle = -P_2 P_1 |\psi_j\rangle = -|\psi_j\rangle .$$

However, this is a contradiction. Hence, P is an abelian group. □

Definition 2.3. A set of *stabilizer generators* is defined as any $G \subseteq P$ such that $P = \langle G \rangle$. Because they are n-qubit Pauli operators, stabilizer generators are self-inverse, so for any $g \in P$ and set of stabilizer generators $G = \{g_j \parallel 1 \leq j \leq m\}$:

$$g = \prod_{j=1}^{m} g_j^{a_j}, a_j \in \{0, 1\}. \tag{2.4}$$

Theorem 2.5. *If a quantum code includes k logically-encoded qubits, n physical qubits, and m independent stabilizer generators, then*

$$m = n - k. \tag{2.6}$$

Proof. Since each qubit encodes a value of either 0 or 1, there are 2^n possible physical states and 2^k possible logical states. By (2.4), the order of the stabilizer group is 2^m.

Since the code state can be described by any stabilizer operator combined with any logical state, $2^m 2^k = 2^n$. Then $m = n - k$. □

Definition 2.7. Let $|\psi\rangle \in \mathcal{N}$ be the state of a quantum code. The *syndrome* is the set of eigenvalues for $S|\psi\rangle$, $\forall S \in P$. We call the state *error-free* if the syndrome is $\{1\}$. If the syndrome contains -1, then an error has been detected [1].

We then define *encoded logical operators*, n-qubit Pauli operators that act on single logical qubits analogously to how Pauli operators act on single physical qubits [1].

Definition 2.8. The *encoded logical X operator* $\overline{X}$ is given by

$$\overline{X}\,|\psi\rangle_L := \sigma_x \begin{bmatrix}\alpha\\ \beta\end{bmatrix}_L = \begin{cases}|1\rangle_L & |\psi\rangle_L = |0\rangle_L\\ |0\rangle_L & |\psi\rangle_L = |1\rangle_L\end{cases}. \tag{2.9}$$

The *encoded logical Z operator* $\overline{Z}$ is given by

$$\overline{Z}\,|\psi\rangle_L := \sigma_z \begin{bmatrix}\alpha\\ \beta\end{bmatrix}_L = \begin{cases}-\,|\psi\rangle_L & |\psi\rangle_L = \pm\,|1\rangle_L\\ |0\rangle_L & |\psi\rangle_L = |0\rangle_L\end{cases}. \tag{2.10}$$

By (1.4), the set of all encoded logical operators is defined completely by $\overline{X}$ and $\overline{Z}$. Encoded logical operators are not unique, since we can add up to 2^m distinct stabilizer operators that yield the same result [1].

Proposition 2.11. *Encoded logical operators commute with every element of the stabilizer group.*

Proof. Any encoded logical operator $\overline{L}$ and stabilizer operator S are both n-qubit Pauli operators, which either commute or anticommute with each other. Let $|\psi\rangle$ be the quantum code state. Suppose S and $\overline{L}$ anticommute. Then,

$$S\overline{L}\,|\psi\rangle = -\overline{L}S\,|\psi\rangle = -\overline{L}\,|\psi\rangle\,.$$

Hence, S has the eigenvector $\overline{L}\,|\psi\rangle$ with corresponding eigenvalue -1. By Definition 2.7, the state contains an error—a contradiction. Therefore, S and $\overline{L}$ must commute. □

Definition 2.12. The *centralizer* is the set of all n-qubit Pauli operators that commute with every element of P [1]. Hence, the centralizer includes all stabilizer operators and all encoded logical operators.

Let C be an operator in the centralizer. Then for any $S \in P$, given the quantum code state $|\psi\rangle \in \mathcal{N}$,

$$SC\,|\psi\rangle = CS\,|\psi\rangle = C\,|\psi\rangle\,.$$

The only eigenvalue of S is 1, so centralizer operators are not detectable by syndrome measurement.

Definition 2.13. The *weight* of any n-qubit Pauli operator is the number of qubits it acts non-trivially on (i.e. the number of non-identity factors). The *code distance* is the minimum weight of any non-identity encoded logical operator [1].

Once errors are detected, error correction operators can be applied to the code state. Because of *code degeneracy*–the property of different errors yielding the same syndrome–a special algorithm called a *decoder* is needed to select the best error correction operator [1].

3. The toric code

Consider the $k \times k$ 2-dimensional lattice formed by the *cellulation*, or division of a surface into polygonal cells meeting edge-to-edge and corner-to-corner, of squares on the torus. We call this lattice the *primal lattice*, and form a *dual lattice* by shifting the primal lattice by half a unit-length vertically and horizontally [1]. The

"hybrid lattice" formed by overlaying the primal and dual lattices, as shown in Figure 1, forms the foundation for the toric code TOR(k) [6].[3]

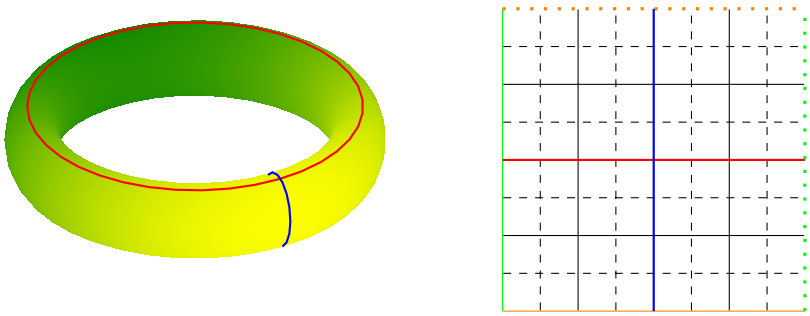

FIGURE 1. Formation of a 4×4 lattice through cellulation of squares on the torus. The dotted lines denote where the top edge meets the bottom edge (orange) and the rightmost edge meets the leftmost edge (green). The dashed lines denote the dual lattice, while the solid lines denote the primal lattice.

Definition 3.1. In either the primal or dual lattice: a *vertex* is a point at which a horizontal and vertical line in the lattice meet. A *plaquette* is a unit square in the lattice. An *edge* is a unit length on a horizontal or vertical line in the lattice [1].

3.1. **Homological foundations.** *Homology* is the study of boundaries. *Cellular homology* refers to the homology of lattices resulting from cellulations, through which we describe TOR(k). In particular, we use $\mathbb{Z}_2$ *homology*, which maps each part of the square lattice to a member of the ring $\mathbb{Z}_2$ [1].

Definition 3.2. An n-*cell* is an n-dimensional object on the lattice (either primal or dual). On a square lattice, 0-*cells* are vertices, 1-*cells* are edges, and 2-*cells* are plaquettes [1]. Henceforth, we denote the set of all n-cells on the primal lattice by $S_n := \{c_j \| \ j \in J_n\}$, where J_n is an indexing set.

Definition 3.3. An n-*chain* c is a finite subset of S_n with the characteristic function $\chi_c : S_n \to \mathbb{Z}_2$:

$$\chi_c(j) := \begin{cases} \overline{1} & c_j \in c \\ \overline{0} & c_j \notin c \end{cases}.$$

Hence, c can equivalently be written as a "coloring" of an element of $\mathbb{Z}_2$ to each $j \in J_n$:

$$c = \sum_{j \in J_n} \chi_c(j) c_j \ [4]. \tag{3.4}$$

The *null* n-*chain* 0_n is defined as $\emptyset \subset S_n$, or

[3]We will build up gradually to the precise definition, found in the beginning of subsection 3.3.

$$0_n := \sum_{j \in J_n} \overline{0} c_j \text{ [1]}.$$

Hence, for $n \geq 0$, the set of n-chains C_n is the $\mathbb{Z}_2$ vector space over the basis S_n [4]. By convention, any -1-*chain*, including the null -1-chain 0_{-1}, is defined as the "coloring" of 0 to all 0-cells, 1-cells, and 2-cells [1].

Definition 3.5. Let $c_j \in S_n$ and d be the $(n-1)$-chain that is its boundary.[4] The *n-boundary map* $\partial_n : C_n \to C_{n-1}$ is the group homomorphism given by $\partial_n c := d$. For any 0-cell c, the 0-boundary map ∂_0 is given by $\partial_0 c = 0_{-1}$ [1].

Definition 3.6. An n-chain c is an *n-cycle* if $\partial_n c = 0_{n-1}$ [1].

Proposition 3.7. *For any $n \in \mathbb{N}$, the set of all n-cycles Z_n forms a group under the group operation of addition (i.e. $Z_n := \ker \partial_n$).*

Proof. $Z_n \subset C_n$, so it suffices to show that Z_n is closed under addition. Let $c_1, c_2 \in Z_n$. Then $\partial_n(c_1 + c_2) = \partial_n c_1 + \partial_n c_2 = 0_{n-1} + 0_{n-1} = 0_{n-1}$, so $c_1 + c_2 \in Z_n$. □

Definition 3.8. An n-chain b is an *n-boundary* if it is the boundary of an $(n+1)$-chain [1].

Proposition 3.9. *For any $n \in \mathbb{N}$, the set of all n-boundaries B_n forms a group under the group operation of addition (i.e. $B_n := \operatorname{Im} \partial_{n+1}$).*

Proof. $B_n \subset C_n$, so it suffices to show that B_n is closed under addition. Let $b_1, b_2 \in B_n$. By Definition 3.8, $\exists c_1, c_2 \in C_{n+1}$ such that $b_1 = \partial_{n+1} c_1$ and $b_2 = \partial_{n+1} c_2$. Then $b_1 + b_2 = \partial_{n+1} c_1 + \partial_{n+1} c_2 = \partial_{n+1}(c_1 + c_2)$. Since $c_1 + c_2 \in C_{n+1}$, $b_1 + b_2 \in B_n$. □

Lemma 3.10. *For any n-chain c with $n \geq 1$:*

$$\partial_{n-1} \partial_n c = 0_{n-2}.$$

Proof. Let $n \geq 1$. For some $j \in J_n$, let $\{d_k \| \ k \in K_j\}$ be the set of $(n-1)$-cells such that $\partial_n c_j = \sum_{k \in K_j} \overline{1} d_k$. Let $\{e_l \| \ l \in L_j\}$ be the set of $(n-2)$-cells such that $e_l \in \partial_{n-1} d_k$ for any $k \in K_j$. For each $l \in L_j$, there are exactly two $k_{l_1}, k_{l_2} \in K_j$ such that $e_l \in \partial_{n-1} d_{k_{l_1}} \cap \partial_{n-1} d_{k_{l_2}}$. Then:

$$\partial_{n-1} \partial_n c_j = \partial_{n-1} \left(\sum_{k \in K_j} \overline{1} d_k \right) = \sum_{l \in L_j} \left(\overline{1} + \overline{1} \right) e_l = 0_{n-2}.$$

Hence, for any n-chain c given by (3.4):

$$\partial_{n-1} \partial_n c = \partial_{n-1} \partial_n \left(\sum_{j \in J_n} \chi_c(j) c_j \right) = \sum_{j \in J_n} \partial_{n-1} \partial_n \chi_c(j) c_j = \sum_{j \in J_n} 0_{n-2} = 0_{n-2}.$$

□

Theorem 3.11. *Every n-boundary is an n-cycle.*

[4]The boundary of a set is defined as the closure minus the interior. Hence, the boundary of a 2-cell is the surrounding 1-cells, the boundary of a 1-cell is the adjacent 0-cells, and the boundary of a 0-cell is the empty set.

Proof. Let $b \in B_n$. By Definition 3.8, $\exists c \in C_{n+1}$ such that $b = \partial_{n+1}c$. By Lemma 3.10, $\partial_n b = \partial_n \partial_{n+1} c = 0_{n-1}$, so $b \in Z_n$. □

Definition 3.12. Two n-chains c and d are *homologically equivalent* if $c = d + b$, where $b \in B_n$ [1].

Definition 3.13. The *nth homology group* H_n is the quotient group

$$H_n := Z_n / B_n.$$

Thus, members of the nth homology group define various homological equivalence classes on the primal lattice [1].

3.2. **Cohomology.** *Cohomology* uses dual vector spaces to generalize the previous subsection to the dual lattice. Similarly to Definition 3.3, an *n-cochain* is a (not necessarily finite) subset of S_n with an associated characteristic function and equivalent "coloring" definition as in (3.4), so the set of n-cochains C^n is a $\mathbb{Z}_2$ vector space over S_n [4]. Then the *null n-cochain* 0^n is equivalent to 0_n, and by convention, any 3-*cochain*, including the null 3-cochain 0^3, is equivalent to the -1-chain.

Definition 3.14. Let $c = \sum_{j \in J_n} \chi_c(j) c_j$ be an n-chain and $p = \sum_{j \in J_n} \chi_p(j) c_j$ be an n-cochain. The *Kronecker pairing* $<,>: C^n \times C_n \to \mathbb{Z}_2$ between c and p is the bilinear map given by

$$\langle p, c \rangle := \sum_{c_j \in c} \chi_p(j) = \sum_{c_j \in p} \chi_c(j) = \sum_{j \in J_n} \chi_p(j)\chi_c(j) \text{ [4].}$$

C^n and C_n are related in the following way:

Proposition 3.15. *For $p \in C^n$, the map $p \to \langle p, \rangle$ is an isomorphism between C^n and* $\hom(C_n, \mathbb{Z}_2)$.

Proof. Let $p \in C^n$. For $j \in J_n$, $\langle p, c_j \rangle = \overline{1}$ if and only if $c_j \in p$, and $\langle p, c_j \rangle = \overline{0}$ otherwise. Hence, $p \to \langle p, \rangle$ is injective.

Let $h \in \hom(C_n, \mathbb{Z}_2)$. Define $a \in C^n$ by

$$a := \{c_j \in S_n \| \ h(c_j) = \overline{1}\}.$$

For any $j \in J_n$, $\langle a, c_j \rangle = h(c_j)$, so $h = \langle a, \rangle$. Then $p \to \langle p, \rangle$ is surjective, so it is a bijection. Hence, it is also an isomorphism between C^n and $\hom(C_n, \mathbb{Z}_2)$. □

Definition 3.16. Let p be an n-cochain. The *n-coboundary map* $\tilde{\partial}^n : C^n \to C^{n+1}$ is the group homomorphism given by

$$\langle \tilde{\partial}^n p, c \rangle = \langle p, \partial_{n+1} c \rangle, \tag{3.17}$$

where c is any $(n+1)$-chain [1].

For analogous reasoning to Lemma 3.10, by Proposition 3.15, given any n-cochain p with $n \leq 1$, $\tilde{\partial}^{n+1}\tilde{\partial}^n p = 0^{n+2}$ [4].

As with Definitions 3.6 and 3.8, an n-cochain p is an *n-cocycle* if $\tilde{\partial}^n p = 0^{n+1}$ and an *n-coboundary* if it is the coboundary of some $(n-1)$-cochain [1]. For analogous reasoning as Propositions 3.7 and 3.9, the set of all n-cocycles $Z^n := \ker \tilde{\partial}^n$ and the set of all n-coboundaries $B^n := \operatorname{Im} \tilde{\partial}^{n-1}$ are groups under addition.

A useful visualization for the dual relationship between homology and cohomology is shown in Figure 2: for any n-cell on the primal lattice, there is a corresponding $(2-n)$-cell on the dual lattice. Then any n-chain c on the primal lattice has a corresponding $(2-n)$-chain on the dual lattice—equivalent to the corresponding n-cochain $\tilde{c}$ under Proposition 3.15 [1].[5]

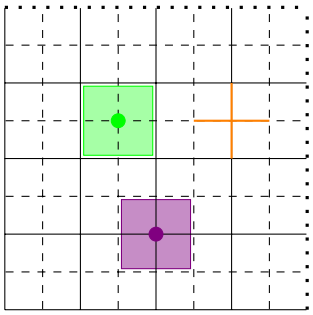

FIGURE 2. n-cells on the primal lattice and their corresponding $(2-n)$-cells on the dual lattice. Green: primal 2-cell and corresponding dual 0-cell. Orange: primal 1-cell and corresponding dual 1-cell. Purple: primal 0-cell and corresponding dual 2-cell.

Definition 3.18. Analogous to Definition 3.13, the *nth cohomology group* is the quotient group

$$H^n := Z^n/B^n.$$

Thus, like the nth homology group on the primal lattice, members of the nth cohomology group define homological equivalence classes on the dual lattice [1].

H_1 and H^1 are of particular importance to the toric code. Representative members of their equivalence classes are shown in Figure 3.

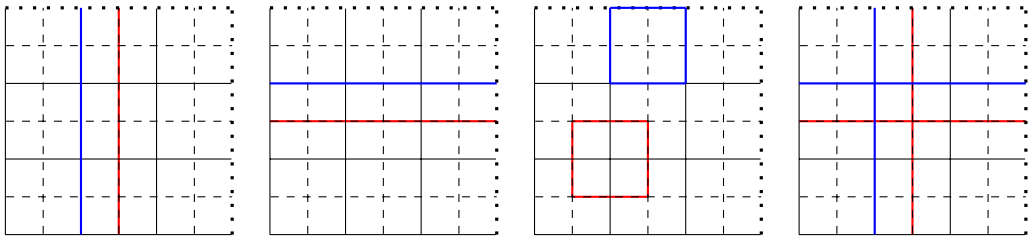

FIGURE 3. Representatives of the four equivalence classes for H_1 (blue) and H^1 (red). The far left shows c_{z_2} and c_{x_1}, and the center left shows c_{z_1} and c_{x_2}. The center right shows contractible loops. The far right shows $c_{z_1} + c_{z_2}$ and $c_{x_1} + c_{x_2}$.

[5]This more antiquated way of conceptualizing cohomology is used most often in discussing the toric code and is depicted in all of the figures.

3.3. **Topological operators.** Now we can characterize the toric code more precisely. We denote the Hilbert space of TOR(k)'s all possible quantum states by $\mathcal{N}$ [6]. Each element of S_1 is a physical qubit, so there are $2k^2$ physical qubits in total. Any $2k^2$-qubit Pauli operator is given by the assignment of some Pauli operator σ to each $j \in J_1$, denoted by the superscript σ^j.[6] *TOR(k) is the non-abelian group formed by these $2k^2$-qubit Pauli operators.*

Definition 3.19. Let s be a 0-cell on the primal lattice and $C_s = \{c_j, j \in J_1 \| \ s \in \partial_1 c_j\}$. The corresponding *vertex operator* A_s is the $2k^2$-qubit Pauli operator

$$A_s := \bigotimes_{c_j \in C_s} \sigma_x^j \bigotimes_{c_j \in S_1 \setminus C_s} I^j. \tag{3.20}$$

Let p be a 2-cell on the primal lattice and $C_p = \{c_j, j \in J_1 \| \ c_j \in \partial_2 p\}$. The corresponding *plaquette operator* B_p is the $2k^2$-qubit Pauli operator

$$B_p := \bigotimes_{c_j \in C_p} \sigma_z^j \bigotimes_{c_j \in S_1 \setminus C_p} I^j \text{ [6]}. \tag{3.21}$$

The constructions of the sets C_s and C_p are shown in Figure 4.

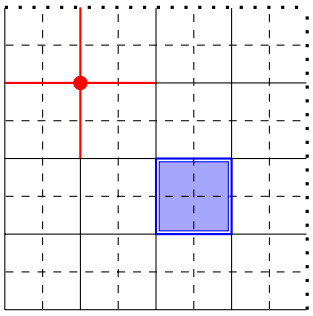

FIGURE 4. Primal vertex s and 1-chain C_s as in (3.20) for defining A_s (red). Primal plaquette p and 1-chain C_p as in (3.21) for defining B_p (blue).

Definition 3.22. The *protected subspace* $\mathcal{L}$ of TOR(k) is defined as

$$\mathcal{L} := \{|\xi\rangle \in \mathcal{N} \| \ \forall s \in S_0, \forall p \in S_2 : A_s |\xi\rangle = |\xi\rangle, B_p |\xi\rangle = |\xi\rangle\}.$$

Thus, A_s and B_p are stabilizer operators of TOR(k). Then for $|\xi\rangle \in \mathcal{N}$, the *syndrome measurement* of a vertex s or plaquette p is given by the eigenvalues of $A_s |\xi\rangle$ or $B_p |\xi\rangle$, respectively. We denote the stabilizer group of TOR(k) by $\mathcal{F}$ [6].

Proposition 3.23. *TOR(k) has $2k^2 - 2$ independent stabilizer generators.*

Proof. The lattice of TOR(k) has k^2 vertices and k^2 plaquettes, so there are $2k^2$ distinct stabilizer operators that generate $\mathcal{F}$. However:

[6]This is the tensor product described in Definition 1.5.

$$\prod_{s\in S_0} A_s = \bigotimes_{j\in J_1} \left(\sigma_x^2\right)^j = \bigotimes_{j\in J_1} I^j, \prod_{p\in S_2} B_p = \bigotimes_{j\in J_1} \left(\sigma_z^2\right)^j = \bigotimes_{j\in J_1} I^j.$$

Then to maintain independence, we must choose all but one $A_s, s \in S_0$ and $B_p, p \in S_2$. Then the total number of independent generators is $2k^2 - 2$. □

For a more complete picture of the protected subspace, we must consider the algebra $\mathbf{L}(\mathcal{L})$ of all linear operators on $\mathcal{L}$ [6].

Lemma 3.24. *Let $\mathcal{G} \subseteq \mathbf{L}(\mathcal{L})$ be the centralizer of TOR(k) and $\mathcal{E} \subseteq \mathbf{L}(\mathcal{L})$ be the group generated by $A_s - \mathbf{1}$ and $B_p - \mathbf{1}$, where $\mathbf{1} = \bigotimes_{j\in J_1} I^j$. Then $\mathcal{E} \subset \mathcal{G}$ and*

$$\mathbf{L}(\mathcal{L}) \cong \mathcal{G}/\mathcal{E}.$$

Proof. For any $s \in S_0, p \in S_2$, and $F \in \mathcal{F}$:

$$(A_s - \mathbf{1})(B_p - \mathbf{1}) = A_sB_p - B_p - A_s + \mathbf{1} = B_pA_s - A_s - B_p + \mathbf{1} = (B_p - \mathbf{1})(A_s - \mathbf{1}),$$

$$(A_s - \mathbf{1})F = A_sF - F = FA_s - F = F(A_s - \mathbf{1}),$$

$$(B_p - \mathbf{1})F = B_pF - F = FB_p - F = F(B_p - \mathbf{1}).$$

Then any operator generated by $A_s - \mathbf{1}$ and $B_p - \mathbf{1}$ commutes with $\mathcal{F}$, so $\mathcal{E} \subset \mathcal{G}$.

By the proof of Proposition 2.11, $\mathbf{L}(\mathcal{L}) \subseteq \mathcal{G}$. Let $\alpha : \mathcal{G} \to \mathcal{N}$ be the homomorphism given by $\alpha(G) = G|\psi\rangle$ for $|\psi\rangle \in \mathcal{N}$. Then $\mathcal{E} = \ker\alpha$. Each $L \in \mathbf{L}(\mathcal{L})$ corresponds to a unique $L|\psi\rangle$ for $|\psi\rangle \in \mathcal{N}$. Then under the identity homomorphism, $\mathbf{L}(\mathcal{L}) \cong \mathcal{G}/\mathcal{E}$. □

Now, we consider the construction of $\mathbf{L}(\mathcal{L})$ using the four equivalence classes in H_1 and H^1.

Definition 3.25. A *string* is a 1-chain or 1-cochain that is not a closed path. For $t \in C_1$ or $t' \in C^1$, *string operators* are defined as the following $2k^2$-qubit Pauli operators:

$$S^z(t) := \bigotimes_{c_j\in t} \sigma_z^j \bigotimes_{c_j\in S_1\setminus t} I^j, S^x(t') := \bigotimes_{c_j\in t'} \sigma_x^j \bigotimes_{c_j\in S_1\setminus t'} I^j. \tag{3.26}$$

Definition 3.27. For non-contractible loops $c_{z_1}, c_{z_2} \in H_1$ and $c_{x_1}, c_{x_2} \in H^1$ (such as the examples shown in Figure 3), define the following $2k^2$-qubit Pauli operators:

$$Z_1 := S^z(c_{z_1}), Z_2 := S^z(c_{z_2}), \tag{3.28}$$

$$X_1 := S^x(c_{x_1}), X_2 := S^x(c_{x_2}). \tag{3.29}$$

Theorem 3.30. *$\{Z_1, Z_2, X_1, X_2\}$ is a generating set for $\mathbf{L}(\mathcal{L})$.*

Proof. Let C_s and C_p be defined as in Definition 3.19. Any $L \in \mathbf{L}(\mathcal{L})$ can be written as the product of string operators. However, for any vertex s such that for some $c \in C_1$, $s \in \partial_1 c$:

$$S^z(c)A_s = \bigotimes_{c_j\in c\setminus C_s} \sigma_z^j \bigotimes_{c_j\in C_s\setminus c} \sigma_x^j \bigotimes_{c_j\in c\cap C_s} (\sigma_z\sigma_x)^j \bigotimes_{c_j\in S_1\setminus c\cup C_s} I^j \tag{3.31}$$

$$= - \bigotimes_{c_j \in c \setminus C_s} \sigma_z^j \bigotimes_{c_j \in C_s \setminus c} \sigma_x^j \bigotimes_{c_j \in c \cap C_s} (\sigma_x \sigma_z)^j \bigotimes_{c_j \in S_1 \setminus c \cup C_s} I^j = -A_s S^z(c).$$

Similarly, for any plaquette p such that for some $c' \in C^1$, $p \in \partial^1 c'$:

$$S^x(c')B_p = \bigotimes_{c_j \in c' \setminus C_p} \sigma_x^j \bigotimes_{c_j \in C_p \setminus c'} \sigma_z^j \bigotimes_{c_j \in c' \cap C_p} (\sigma_x \sigma_z)^j \bigotimes_{c_j \in S_1 \setminus c' \cup C_p} I^j \tag{3.32}$$

$$= - \bigotimes_{c_j \in c' \setminus C_p} \sigma_x^j \bigotimes_{c_j \in C_p \setminus c'} \sigma_z^j \bigotimes_{c_j \in c' \cap C_p} (\sigma_z \sigma_x)^j \bigotimes_{c_j \in S_1 \setminus c' \cup C_p} I^j = -B_p S^x(c').$$

Then $S^z(c)$ and $S^x(c')$ can only generate $\mathbf{L}(\mathcal{L})$ if $\partial_1 c = 0_0$ and $\partial^1 c' = 0^2$. Hence, we consider only $c \in H_1$ and $c' \in H^1$.

If c and c' are contractible[7] (i.e. in the third homological equivalence class shown in Figure 3), $S^z(c) = B_{p_1} B_{p_2}$ for some $p_1, p_2 \in S_2$, and $S^x(c') = A_{s_1} A_{s_2}$ for some $s_1, s_2 \in S_0$ [6].

Instead, if c and c' are non-contractible, then they must be given by $c = c_{z_1}$ or $c = c_{z_2}$ and $c' = c_{x_1}$ or $c' = c_{x_2}$, as in Definition 3.27. By (3.28) and (3.29), this defines Z_1, Z_2, X_1, and X_2.

Let $c \in H_1$ and $c' \in H^1$ be in the fourth homological equivalence class shown in Figure 3. Then there are some $c_{z_1}, c_{z_2} \in H_1, c_{x_1}, c_{x_2} \in H^1$ as in Definition 3.27 such that $S^z(c) = Z_1 Z_2 = Z_2 Z_1$ and $S^x(c') = X_1 X_2 = X_2 X_1$. Hence, $\{Z_1, Z_2, X_1, X_2\}$ is a generating set for $\mathbf{L}(\mathcal{L})$. □

Corollary 3.33. *dim*$\mathcal{L} = 4$ *over* $\{Z_1, Z_2, X_1, X_2\}$.

Proof. Most directly, by Theorem 3.30, since there are four elements in the generating set of $\mathbf{L}(\mathcal{L})$, dim$\mathcal{L} = 4$.

Another proof follows from the stabilizer formalism. There are $2k^2$ physical qubits and $2k^2 - 2$ independent stabilizer generators by Proposition 3.23. Then by Theorem 2.5, there are $2k^2 - \left(2k^2 - 2\right) = 2$ logical qubits. Since each logical qubit has two possible states, dim$\mathcal{L} = 2^2 = 4$. □

We can interpret the second proof of Corollary 3.33 as a k^2-qubit repetition code, with the first qubit $|\psi\rangle_{L1}$ constructed from horizontal edges on the lattice and the second qubit $|\psi\rangle_{L2}$ constructed from vertical edges on the lattice.

For $1 \leq i \leq k$, let $c_{z_1}^i$ denote the ith c_{z_1}-equivalent 1-chain, $c_{z_2}^i$ denote the ith c_{z_2}-equivalent 1-chain, $c_{x_1}^i$ denote the ith c_{x_1}-equivalent 1-cochain, and $c_{x_2}^i$ denote the ith c_{x_2}-equivalent 1-cochain. Let $Z_{1i} := S^z(c_{z_1}^i)$, $Z_{2i} := S^z(c_{z_2}^i)$, $X_{1i} := S^x(c_{x_1}^i)$, and $X_{2i} := S^x(c_{x_2}^i)$. Define the following $2k^2$-qubit Pauli operators:

$$\overline{Z_1} = \prod_{1 \leq i \leq k} Z_{1i}, \overline{Z_2} = \prod_{1 \leq i \leq k} Z_{2i}, \tag{3.34}$$

$$\overline{X_1} = \prod_{1 \leq i \leq k} X_{1i}, \overline{X_2} = \prod_{1 \leq i \leq k} X_{2i}. \tag{3.35}$$

Then:

[7] Meaning they can be continuously deformed to a single point.

$$\overline{Z}\left|\psi\right\rangle_{L1}=\begin{cases}-\left|\psi\right\rangle_{L1} & \left|\psi\right\rangle_{L1}=\pm\left|1\right\rangle_{L1}\\ \left|0\right\rangle_{L1} & \left|\psi\right\rangle_{L1}=\left|0\right\rangle_{L1}\end{cases},\overline{X}\left|\psi\right\rangle_{L1}=\begin{cases}\left|1\right\rangle_{L1} & \left|\psi\right\rangle_{L1}=\left|0\right\rangle_{L1}\\ \left|0\right\rangle_{L1} & \left|\psi\right\rangle_{L1}=\left|1\right\rangle_{L1}\end{cases},\text{ and}$$

$$\overline{Z}\left|\psi\right\rangle_{L2}=\begin{cases}-\left|\psi\right\rangle_{L2} & \left|\psi\right\rangle_{L2}=\pm\left|1\right\rangle_{L2}\\ \left|0\right\rangle_{L2} & \left|\psi\right\rangle_{L2}=\left|0\right\rangle_{L2}\end{cases},\overline{X}\left|\psi\right\rangle_{L2}=\begin{cases}\left|1\right\rangle_{L2} & \left|\psi\right\rangle_{L2}=\left|0\right\rangle_{L2}\\ \left|0\right\rangle_{L2} & \left|\psi\right\rangle_{L2}=\left|1\right\rangle_{L2}\end{cases}.$$

Hence, by Definition 2.8, $\overline{Z_1}$, $\overline{Z_2}$, $\overline{X_1}$, and $\overline{X_2}$ are encoded logical operators for TOR(k), so the topological construction of TOR(k) leads naturally to the stabilizer formalism!

To determine TOR(k)'s error detection and correction capabilities, consider the generic n-qubit error represented by the following $2k^2$-qubit Pauli operator:

$$E=\sigma(\alpha_1,...,\alpha_n;\beta_1,...,\beta_n)=\bigotimes_{i=1}^{n}\left(\left(\sigma_x\right)^{\alpha_i}\left(\sigma_z\right)^{\beta_i}\right)^{j_i}\bigotimes_{j\in J_1\setminus J'}I^j, \tag{3.36}$$

$$1\leq i\leq n:\alpha_i,\beta_i\in\{0,1\},J':=\{j_i\|\ 1\leq i\leq n\}\subset J_1.$$

Theorem 3.37. *The code distance of TOR(k) is k.*

Proof. Let $E\in\mathcal{G}\setminus\mathcal{F}$. Then E is non-trivial but undetectable by syndrome measurements. By the proof of Theorem 3.30, E must be the product of some $S^z(c)$ and $S^x(c')$, where c or c' are non-contractible. Hence, if E is given by (3.36),

$$\text{Supp}(E):=n\left(\{1\leq i\leq n\|\ \alpha_i=1\text{ or }\beta_i=1\}\right)\geq k.$$

Then the code distance, which is the minimum weight of an undetectable error, is k [1]. □

Thus, TOR(k) will be more error-tolerant for larger lattice sizes [1]. TOR(k) can detect $k-1$ errors and correct $\lfloor\frac{k-1}{2}\rfloor$ errors [6].[8] For any error E, the decoder must find a corresponding error correction operator $C_E\in\mathbf{L}(\mathcal{L})$ such that $C_EE\in\mathcal{F}$ [1].

By the proof of Theorem 3.37, since $Z_1,Z_2,Z_1Z_2\in\mathcal{G}\setminus\mathcal{F}$, if E is given by (3.36), then E, Z_1E, Z_2E, and $Z_1Z_2E=Z_2Z_1E$ are "homologically inequivalent" operators that yield the same syndrome measurements, as shown in Figure 5. This is a manifestation of code degeneracy.

3.4. **Abelian anyons.** To perform error correction, we use quasiparticles corresponding to real-world phenomena in solid-state systems [6].

Definition 3.38. Let $\left|\xi\right\rangle\in\mathcal{N}$. An *elementary excitation* or *particle* occurs at a vertex s or plaquette p if either $A_s\left|\xi\right\rangle\neq\left|\xi\right\rangle$ or $B_p\left|\xi\right\rangle\neq\left|\xi\right\rangle$ [6]. We conceive of such s or p as quasiparticles because they raise the physical energy levels of the ground states of associated qubits [5].

Theorem 3.39. *For $t\in C_1$, $S^z(t)$ commutes with all B_p and all A_s except for $s\in\partial_1t$. Similarly, for $t'\in C^1$, $S^x(t')$ commutes with all A_s and all B_p except for $p\in\tilde{\partial}^1t'$.*

[8] This is a result of the Hamming distance, an information theory concept beyond the scope of this paper.

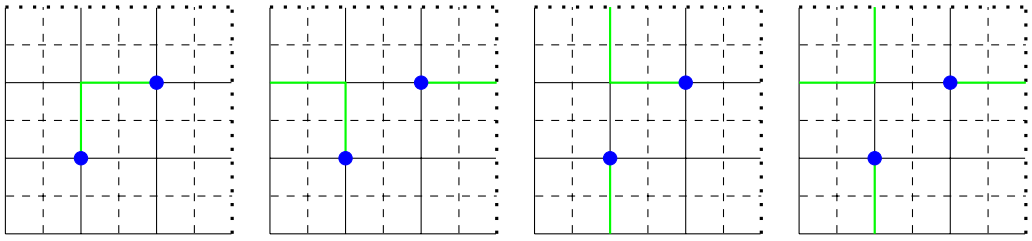

FIGURE 5. Examples of homologically-inequivalent errors with identical syndrome measurements: 1-cells shaded black correspond to I, 1-cells shaded green correspond to σ_z, and 0-cells shaded blue denote the presence of -1 eigenvalues. Far left: E. Center left: Z_1E. Center right: Z_2E. Far right: $Z_1Z_2E = Z_2Z_1E$.

Proof. This follows from the proof of Theorem 3.30. Let $t \in C_1$ and $t' \in C^1$. By (3.31), $S^z(t)$ and A_s anticommute if and only if $s \in \partial_1 t$. By (3.32), $S^x(t)$ and B_p anticommute if and only if $p \in \tilde{\partial}^1 t'$. □

For $t \in C_1$, the primal 0-cells in $\partial_1 t$ are called *z-type particles* or *electric charges*. For $t' \in C^1$, the dual 0-cells[9] in $\tilde{\partial}^1 t'$ are called *x-type particles* or *magnetic vortices* [6]. String operators can be interpreted physically as moving a particle from one endpoint of the chain or cochain to the other [5].

Proposition 3.40. *Let $|\xi\rangle \in \mathcal{N}$. For $t \in C_1$, let $E = S^z(t)$ be an error operator. Then if and only if $t' \in C_1$ such that $\partial_1 t' = \partial_1 t$ and $t \cap t' = \emptyset$:*

$$S^z(t')\, E\, |\xi\rangle = |\xi\rangle .$$

Similarly, for $t \in C^1$ and error operator $E = S^x(t)$, then if and only if $t' \in C^1$ such that $\tilde{\partial}^1 t' = \tilde{\partial}^1 t$ and $t \cap t' = \emptyset$:

$$S^x(t')\, E\, |\xi\rangle = |\xi\rangle .$$

Proof. Let $t, t' \in C_1$ such that $\partial_1 t' = \partial_1 t$ and $t \cap t' = \emptyset$. If $E = S^z(t)$:

$$S^z(t')E = \bigotimes_{c_j \in t+t'} \sigma_z^j \bigotimes_{c_j \in S_1 \setminus t+t'} I^j .$$

Since $\partial_1(t + t') = 0_0$, $S^z(t')E \in \mathcal{F}$ by the proof of Theorem 3.30. Then for $|\xi\rangle \in \mathcal{N}$, $S^z(t')E\,|\xi\rangle = |\xi\rangle$.

The proof for the second part of Proposition 3.40 follows analogously. □

As a result of code degeneracy, the decoder must find distinct error correction operators for errors in the four homological equivalence classes, as shown in Figure 5, to avoid overlapping with the 1-chain or 1-cochain associated with the error [1].

For any $t_1, t_2 \in C_1$, $S^z(t_1)$ and $S^z(t_2)$ commute, and for any $t_1, t_2 \in C^1$, $S^x(t_1)$ and $S^x(t_2)$ commute [5]. On the other hand, consider when both S^z and S^x operators are applied, such as when an electric charge is "moved around" the magnetic vortex or vice versa, as shown in Figure 6.

[9]Equivalent to primal 2-cells, as previously discussed.

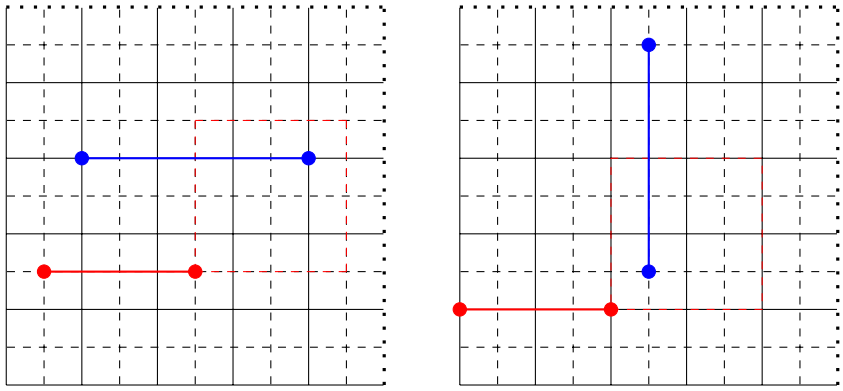

FIGURE 6. Left: x-type particle moving through $q \in C^1$ (red) including contractible loop $c \in H^1$ (dashed red) around z-type particle moving through $t \in C_1$ (blue). Right: z-type particle moving through $q \in C_1$ (red) including contractible loop $c \in H_1$ (dashed red) around x-type particle moving through $t \in C^1$ (blue).

Theorem 3.41. *As shown in Figure 6, let $t \in C_1$ and $q \in C^1$ (or $t \in C^1$ and $q \in C_1$) such that q forms a contractible loop c around a particle on t. If the initial state of the code is $|\psi_i\rangle$, then the final state $|\psi_f\rangle$ after moving one particle in a loop around another differently-typed particle is $|\psi_f\rangle = -|\psi_i\rangle$.*

Proof. Let $t \in C_1$, $q \in C^1$ with q forming the loop $c \in Z^1$ as described. $S^x(c)$ and $S^z(t)$ anticommute because they share exactly one $j^* \in J_1$ such that $c_{j^*} \in t \cap c$:

$$S^x(c)S^z(t) = \left(\bigotimes_{c_j \in t \setminus c_{j^*}} \sigma_z^j \bigotimes_{c_j \in c \setminus c_{j^*}} \sigma_x^j \bigotimes_{c_j \in S_1 \setminus t \cup c} I^j \right) \otimes (\sigma_x \sigma_z)^{j^*}$$

$$= - \left(\bigotimes_{c_j \in t \setminus c_{j^*}} \sigma_z^j \bigotimes_{c_j \in c \setminus c_{j^*}} \sigma_x^j \bigotimes_{c_j \in S_1 \setminus t \cup c} I^j \right) \otimes (\sigma_z \sigma_x)^{j^*} = -S^z(t)S^x(c).$$

Then if $|\psi\rangle$ is the originally-unaltered quantum state:

$$|\psi_i\rangle = S^z(t)S^x(q-c)\,|\psi\rangle\,,$$

$$|\psi_f\rangle = S^x(c)S^z(t)S^x(q-c)\,|\psi\rangle = -S^z(t)S^x(c)S^x(q-c)\,|\psi\rangle$$

$$= -S^z(t)S^x(q-c)\,|\psi\rangle = -|\psi_i\rangle\,.$$

Let $q \in C_1$, $t \in C^1$ such that q forms a loop $c \in Z_1$ as described. Then for analogous reasoning, $|\psi_f\rangle = -|\psi_i\rangle$. □

In physics, electric charges and magnetic vortices are called *anyons*, which have this unique sign-flipping property when moving one particle around a differently-typed particle [6]. An anyon has 0-*charge* if its associated syndrome measurement is 1 and +1-*charge* if its associated syndrome measurement is −1. Thus, errors create +1-charge particles, and error correction operators annihilate them [1].

Acknowledgments

It is a pleasure to thank my mentor, Lauren Tsai, for assisting me with the writing process, helping me understand the mathematical background included in this paper, and providing vital references such as [3], [4], and [5]. I would also like to thank Professor Peter May for organizing the REU program, including remote mentoring, which enabled my studies. Additionally, I would like to thank my peers and fellow REU participants Ryan O'Farrell, Vincent Tran, and Isabel Vargas-Hurlston for their insightful discussions and providing lecture notes.

References

[1] Browne, D. (2014). *Lectures on Topological Codes and Quantum Computation.*
[2] Dummit, D. S. and Foote, R. M. (2004). *Abstract Algebra.* John Wiley and Sons, Inc.
[3] Hatcher, A. (2001). *Algebraic Topology.*
[4] Hausmann, J.-C. (2014). *Mod Two Homology and Cohomology.* Springer International Publishing.
[5] Herringer, P. (2020). *The Toric Code.*
[6] Kitaev, A. Y. (2003). "Fault-tolerant quantum computation by anyons". *Annals of Physics*, 303:2–30.
[7] Likharev, K. K. (2022). *Essential Graduate Physics - Quantum Mechanics*, chapter 4. LibreTexts.
[8] Munkres, J. (2000). *Topology.* Pearson Education, Inc.
[9] Nielsen, M. and Chuang, I. (2010). *Quantum Computation and Quantum Information.* Cambridge University Press.