An Introduction to the Toric Code as Topological Quantum Computing

by Milo Moses

February 28, 2023

Abstract

One of the most promising forms of quantum computation proposed today is Topological Quantum Computation. In this manuscript, we describe the simplest non-trivial example of Topological Quantum Computation: The toric code. We give explanations in terms of elementary mathematics and physics.

1 Introduction

Of the many approaches at quantum computation, Topological Quantum Computation (TQC) has the distinction of being both one of the most mathematically complicated and one of the most potentially powerful methods. In this manuscript, we describe the simplest non-trivial example of TQC: The toric code. While not being useful in itself (toric code TQC is far from being universal), a thorough understanding of the toric code undoubtedly elucidates the general TQC methodology.

To describe a theory of Quantum Computation, one describe the following:

1. How quantum information is stored (i.e, what physical model of qubits one is using)
2. How quantum information is acted on (i.e, what physical actions one can perform on the qubits)
3. How quantum information is measured (i.e, what observables can be physically measured in the system)

A *universal* model of quantum computation is one which can simulate all others. Generally, this will mean that the space of possible physical actions specified by the quantum computation model is dense in the space of all possible transformations on the space of qubits (i.e, can be used to approximate every transformation arbitrarily well).

While there are many proposed methods of quantum computation (superconducting quantum computers [Wen17], trapped ion quantum computers [DLF+16],

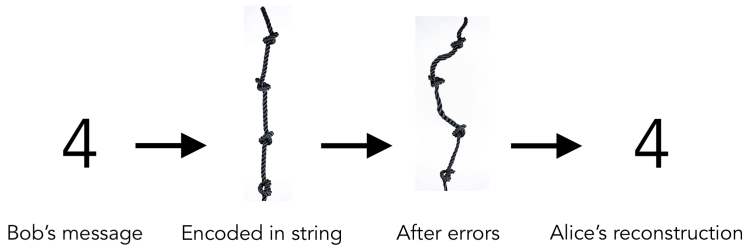

Figure 1.1: A model of a (non-quantum) topological message

semiconductor based quantum computers [Kan98], etc...), it is expected that every such reasonable model will be essentially equivalent, in the sense that they can all *effectively* simulate each other: This is the content of the Freedman-Church-Turing thesis [FKLW03].

In a a sequence of seminal works by Freedman, Kitaev, Larsen, and Wang, it was shown that universal quantum computers can effectively model any TQC, and conversely that there are models of TQC that can effectively simulate a universal quantum computer [FLW02] [FKW02].

If all forms of quantum computation are roughly equivalent it is reasonable to ask why one would consider TQC to be more promising than other models. This is an especially relevant question seeing as Google's superconducting quantum computer can harness 53 qubits and has demonstrated quantum superiority, but Microsoft's TQC has not been able to reliably harness a single qubit [AAB$^+$19].

The intuition is as follows. The #1 challenge in quantum computation is error correction. While fault tolerant quantum computers can provably exist (i.e, quantum computers that fix errors faster than they happen), the error rate must be un-physically low [Got98]. *Topology* is the mathematical study of those properties of geometric objects that are invariant under small perturbations. The key insight of TQC is that instead of storing quantum information in the states of individual particles, the information is stored in topological invariants of geometric objects. As such, even when physical errors happen (i.e, the geometric object is perturbed) the information stored in the qubits remains the same. More than being error correcting, TQC is naturally error resistant [Kit03]!

As a thought experiment to reinforce this idea, suppose that Alice and Bob are placed across the country, and are given only a string to communicate. By "string" we really mean string: This is some physical piece of twine or rope. They can ship this string via train, and through this process there will undoubtedly be errors (i.e, the string gets pushed around during the voyage). How can Alice and Bob effectively communicate, while being relatively confident there are no errors?

Figure 1.2: An Incan Quipu

The answer is simple: Store the information in knots! By tying a certain number of simple knots in the string Alice can specify an integer, which Bob can simply read off by counting (as in Figure 1.1)! The beauty is in the fact that while the string may have moved around during the sending process, it would take a very specialized and unlikely error to untie the rope or to accidently re-tie an extra knot. This knotting number is a topological invariant (small perturbations don't change how many knots were tied), and so we can see intuitively here that topological invariants are naturally error resistant.

Additionally, the above situation is more than just a thought experiment: This is exactly the scheme that the ancient South American Incas used over 4000 years ago! The Incas stored all sorts of information in *Quipus*, intricately knotted collections of fibered strings [AA81], as seen in Figure 1.2. Storing information in knot invariants was also common practice in ancient Chinese, Tibetan, and Polynesian cultures [Day21]. In some sense, these are the earliest examples of topological computation.

In TQC, information is still stored in knots. The main difference is that the strings being knotted are no longer physical pieces of twine, but *trajectories of quasiparticles through spacetime*. For instance, suppose X and Z are two quasiparticles (we will elaborate more on this in later). Moving through space from time t_0 to t_1, the trajectories can look something like Figure 1.3. Quantum interactions cause the knotting to yield real differences in physical states, and hence these knots can be used to store quantum information.

Notice that to make a 3 dimensional spacetime, we are modeling space as being 2 dimensional. While one might initially think this is a quirk of our human incapacity of visualizing 4 dimensional space, there is a deeper mathematical truth here: There are no knots in 4 dimensional space. The extra dimension always gives the strands space to evade and move past each other without collision. In particular, for TQC to work we must have space be two dimensional. While this task seems initially impossible, (essentially) two dimensional quantum phases of matter have been experimentally constructed. Things that behave like particles in these 2 dimensional phases of matter are known as quasiparticles, and form non-trivial knots when braided.

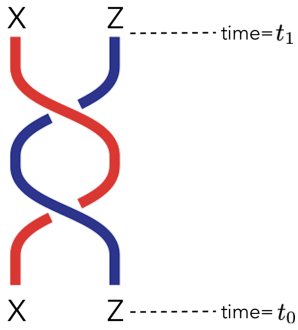

Figure 1.3: Braiding of quasiparticles in spacetime

A rough description of how these 2 dimensional electron gasses are constructed is described as follows. One begins by preparing a series of layers of graphene with a small gap in the middle. Upon subjecting the system to extremely cold temperatures and an extremely high magnetic field, the electrons in the graphene begin to move around. To balance the electric charge on both sides, all of the electrons move to the exact center of the setup. This resulting thin layer of electrons is the two dimensional electron gas. In such extreme conditions, all of the electrons will become highly entangled with each other, forming a quantum phase of matter [YZB$^+$21].

A key insight of Kitaev [Kit03], and one of the motivating pushes towards quantum computation, was that the topological properties of the 2 dimensional phase of matter will determine how the electrons entangled with each other. In other words, 2 dimensional sheets of electrons will quantize differently depending on their shape.

To understand this better, suppose that you have a 2 dimensional sphere of electrons. They will want to quantize, and align their spins together in the same direction. This amounts to choosing a unit tangent vector at each point on the sphere. However, there is no coherent way to do this. Every choice of tangent vectors will necessarily have some discontinuity or singularity: This is the content of the "Hairy Ball Theorem".

If your sheet of electrons was on a donut, however, the situation is much different. There are several ways of assigning unit tangent vectors coherently to each point, and thus there are several ways for all of the electrons to quantize their spins together, as seen in Figure 1.4. This is aptly known as a *spin liquid*. In mathematical language, a "donut" is called a torus, hence the name *toric code*. The spin liquid associated with this procedure on a torus is called the "$\mathbb{Z}_2$ spin liquid", and it is the physical realization of the toric code. We will spend the body of this manuscript describing the mathematics of the toric code in more detail. Note that generally when making such $\mathbb{Z}_2$ spin liquids in labs one does not make an actual torus; instead, one artificially simulates the boundary

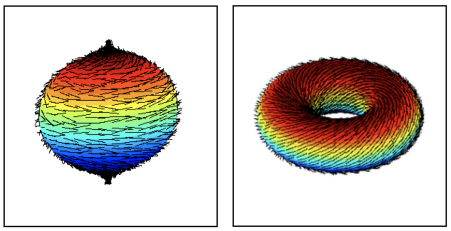

Figure 1.4: Assigning vector fields (spins) to a sphere, verses to a torus

conditions of a torus, for technical reasons.

We find it illustrative here to make an analogy with classical computing. Namely, there is the following puzzle: Classical bits are stored in the magnetization of small regions on a hard disk. The magnetization of each atom is highly sensitive to thermal fluctuations, and so one might ask why it is that classical computers seem so resistant to errors. The answer is that since all of the atoms are magnetized in the same direction, any one atom flipping will automatically be corrected back by the normalizing influence of the other atoms: magnets are naturally error resistant. It is exactly the same with these spin liquids that quantize together: Any one electron's spin decohering will immediately be corrected by the normalizing influence of all of its neighbors.

Before moving on to an in-depth treatment of the toric code, we offer a general description of the TQC process, in the style of the three points listed in the beginning of the introduction:

1. Information is stored in the ground states of topological quantum materials.

2. Ground states are acted on by braiding of quasiparticles, that is, by generating pairs of quasiparticles and making them knot in spacetime.

3. Measurement is performed by observing the topological properties of the resulting ground state.

Here, *ground state* refers to a state in the system with lowest possible energy. Note that in the previous description of spin liquids, all electrons having the same spin is a result of them being in the lowest energy state. An "excited" electron with deviant spin will raise the energy of the system. In this way, quasiparticles can be interpreted as excitations of the topological quantum material. Note additionally this motivates the fact that (topological) quantum computers must be exceptionally cold to function: Any extra energy will correspond to extra excitations, causing the computer to malfunction, since by definition the states information is stored in are ground states.

The possibilities for topological quantum materials and TQC are extremely exciting, and we are eager to see where the field will go in the coming years.

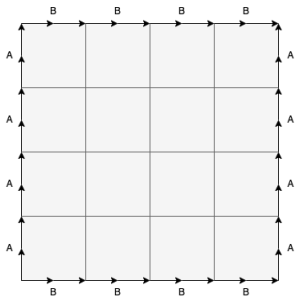

Figure 2.1: Celluation of the torus, obtained by gluing opposite sides together.

2 The Toric Code

Consider a torus. We will be imagining the torus as a whole as being a quantum system, corresponding physically to the quantum system one would observe when the torus is in the $\mathbb{Z}_2$ spin liquid topological quantum phase of matter. The *code space* of the torus is the space of states on which we will be building our quantum computer, i.e, those states we will be using to store quantum information. In general Topological Quantum Computing (TQC) fashion, the code space of the toric code will be its ground states.

Our mathematical priorities are thus as follows: To define the quantum system, and to define a Hamiltonian operator on it. A Hamiltonian is an operator corresponding to the total energy of a quantum system. Namely, the eigenvalue of an eigenstate of the Hamiltonian corresponds to the total energy of that state. The code space will thus be the lowest eigenspace of the Hamiltonian.

Working with a continuous torus and the corresponding infinite dimensional vector spaces is cumbersome and unnecessary. Instead, we celluate the torus into an n by n square lattice with opposing sides identified, as in Figure 2.1. We will work with the understanding that the real physical system is the limit as $n \to \infty$. We define the quantum system associated with the n by n celluated torus to be the Hilbert space

$$\mathcal{N} = \bigotimes_{\substack{\text{edges of} \\ \text{torus}}} \mathbb{C}^2,$$

obtained by "putting a qubit on every edge". Here and throughout, *vertices*, *edges*, and *faces*, when used as indexing sets, will refer to the set of *vertices*, *edges*, and *faces* of our celluated torus. We will choose a canonical basis $\{|0\rangle, |1\rangle\}$ for $\mathbb{C}^2$, reflecting our information theoretic intentions. To more forward with defining the Hamiltonian, we introduce the Pauli matrices

$$\sigma_X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \ \sigma_Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \ \sigma_Z = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}.$$

The Hamiltonian is defined by

$$H = -\sum_{\text{verticies } v} A_v - \sum_{\text{faces } p} B_p,$$

where

$$A_v = \bigotimes_{\substack{\text{edges} \\ \text{touching } v}} \sigma_Z, \; B_p = \bigotimes_{\substack{\text{edges} \\ \text{touching } p}} \sigma_X.$$

All of the power of the toric code comes from this highly non-obvious choice of Hamiltonian. The physical interpretation for this choice of Hamiltonian comes from gauge theory. Namely, the $U(1)$ Lattice Gauge Theory has two fields: The Compact Gauge Field and the Electric Field. Exponentiating the Compact Gauge Field we get the σ_X operators, and exponentiating the Electric Field we get the σ_Z operators. Thus, the A_v contribute a "Gauss' Law" term, and the B_p contribute a "Magnetic Field" term to the Hamiltonian [OKMH22]. While potentially physically illuminating, this discussion of gauge theory will have no influence on the rest of the mathematics presented in this manuscript, and does not need to be understood to appreciate the toric code.

Letting I denote the identity matrix, the key facts about the A_vs and B_ps are summarized in the following proposition:

Proposition 2.1. *We have that*

(i) $A_v^2 = B_p^2 = I$ *for all* v, p

(ii) All A_v*s and* B_p*s have half eigenvalues* $+1$ *and half eigenvalues* -1

(iii) All A_v*s and* B_p*s commute*

(iv) $\prod_{vertices\ v} A_v = I$ *and* $\prod_{faces\ p} B_p = I$

Proof. (i). Multiplying tensor product matrices corresponds to simply multiplying componentwise. Hence, this part follows immediately from the relations $\sigma_X^2 = \sigma_Z^2 = I$.

(ii). We define an isomorphism between the $+1$ eigenspace and -1 eigenspace of A_v. Namely, apply σ_X to an edge e touching v. Since $\sigma_X \sigma_Z = -\sigma_Z \sigma_X$, the computation

$$A_v \left(\bigotimes_{\text{edge } e} \sigma_X \right) |\psi\rangle = -A_v \left(\bigotimes_{\text{edge } e} \sigma_X \right) A_v |\psi\rangle$$

show that a $+1$ eigenstate will be transformed into a -1 eigenstate, and a -1 eigenstate will be turned into $+1$ eigenstate. Thus, this defines an isomorphism between the desired eigenspaces. Applying σ_Z instead of σ_X, we can define a similar isomorphism for B_p.

this will have the effect of turning a $+1$ eigenstate into a -1 eigenstate and vice-versa,

(iii). All the A_vs commute with each other since σ_Z commutes with itself, and all the B_ps commute with each other since σ_X commutes with itself. What's left to check is that $A_v B_p = B_p A_v$. Notice that if v is not touching the face p, none of the σ_Zs in the tensor product of A_v will be in the same spots as any of the σ_Xs as the tensor product for B_p. Hence, A_v and B_p commute in this case. If v is touching p, then exactly two of the σ_Zs in the tensor product of A_v will be in the same spots as σ_Xs in the tensor product of B_p. Hence, pulling B_v through A_v corresponds to switching σ_X and σ_Z. Since $\sigma_X \sigma_Z = -\sigma_Z \sigma_X$, this introduces an overall phase shift of $(-1)^2 = 1$. Hence, $A_v B_p = B_p A_v$ as desired!

(iv). Applying $\prod_{\text{vertices } v} A_v$, is the same as applying σ_Z to each vertex 2 times, since each edge touches exactly 2 vertices. Hence,

$$\prod_{\text{vertices } v} A_v = \bigotimes_{\text{edges}} \sigma_Z^2 = \bigotimes_{\text{edges}} I = I.$$

Similarly, since every edge touches exactly 2 faces, the fact that $\prod_{\text{faces } p} B_p = I$ follows from $\sigma_X^2 = 1$. □

Using the above facts about the A_vs and B_ps, we can describe the eigenspaces of H well enough to compute their dimension:

Proposition 2.2. *All eigenvalues of H are of the form $-2n^2+4q$, for an integer $q \leq n^2/2$. The $-2n^2+4q$ can be described as the space of states $|\psi\rangle$ such that*

$$|\{\, v, p |\, A_v \,|\psi\rangle = -1,\ B_p \,|\psi\rangle = -1\}| = 2q,$$

that is, the space of states with $2q$ excitations. There will always be an even number of v such that $A_v \,|\psi\rangle = -1$, as well as an even number of p such that $B_p \,|\psi\rangle = -1$. The dimension of of this eigenspace is

$$4 \sum_{k=0}^{q} \binom{n^2}{2k} \binom{n^2}{2(q-k)}.$$

In particular, the code space of the toric code is 4 dimensional, and consists of those vectors $|\psi\rangle$ such that $A_v \,|\psi\rangle = B_p \,|\psi\rangle = |\psi\rangle$ for all v, p.

Proof. To begin, we observe the following general fact from linear algebra. If M and N are commuting matrices and $|\psi\rangle$ is an eigenvector for N with eigenvalue λ, then

$$N(M \,|\psi\rangle) = M(N \,|\psi\rangle) = \lambda(M \,|\psi\rangle).$$

Hence, M respect the eigenspaces of N, and vice versa. This implies that the eigenspaces for H will be simultaneous eigenspaces for all of the A_vs and B_ps, since all of the A_vs and B_ps commute by Proposition 2 (iii).

Suppose that $|\psi\rangle$ is an eigenstate with

$$|\{\, v, p|\, A_v\,|\psi\rangle = -1,\ B_p\,|\psi\rangle = -1\}| = q.$$

Then, we find that

$$\begin{aligned} H\,|\psi\rangle &= (-\sum_v A_v - \sum_p B_p)\,|\psi\rangle \\ &= \left(\sum_{\substack{v,p \\ -1 \text{ eigenvalue}}} 1 - \sum_{\substack{v,p \\ 1 \text{ eigenvalue}}} 1 \right) |\psi\rangle \\ &= (q - (n^2 - q)) \\ &= -n^2 + 2q. \end{aligned}$$

Thus, to complete the initial description of the eigenstates, we must show that the number of v such that $A_v\,|\psi\rangle = -1$ and the number of p such that $B_p\,|\psi\rangle = -1$ is even. This follows from the computation that is this number where odd, then we would have by Proposition 2 (iv) that

$$|\psi\rangle = \left(\prod_v A_v \right) |\psi\rangle = -\,|\psi\rangle\,,$$

which is a contradiction since we are supposing that $|\psi\rangle \neq 0$. The exact same argument applies to the B_p. We now compute the dimensions of the eigenspaces. Let D denote the dimension of the group space. We show that given any even sized sets $\mathbf{v}, \mathbf{p}$ of vertices and faces respectively, the space

$$\mathcal{N}_{\mathbf{v},\mathbf{p}} = \{\, |\psi\rangle |\, (A_v\,|\psi\rangle = -1 \iff v \in \mathbf{v})\,,\ (B_p\,|\psi\rangle = -1 \iff p \in \mathbf{p})\}$$

is D dimensional. We proceed by induction on $|\mathbf{v} + \mathbf{p}|$. If $|\mathbf{v} + \mathbf{p}| = 0$, then this is the definition of D. Without loss of generality, suppose $|\mathbf{v}| \geq 2$. If $|\mathbf{p}| \geq 2$, we apply the same arguement with verticies replaced by faces. Choose two verticies $v_0, v_1 \in \mathbf{v}$. Choose a path γ along the edges of the torus that connect v_0 and v_1. We show that $\bigotimes_{\text{edges in } \gamma} \sigma_X$ gives an isomorphism between $\mathcal{N}_{\mathbf{v},\mathbf{p}}$ and $\mathcal{N}_{\mathbf{v}-\{v_0,v_1\},\{p\}}$. Namely it is clear from $\sigma_X^2 = \sigma_X$, so this map is its own inverse, so it is sufficient to show that the image is in the desired space. To prove this, we observe that $\bigotimes_{\text{edges in } \gamma} \sigma_X$ commutes with all the B_ps, and commutes with all of the A_vs at verticies that γ passes through an even number of times. The only verticies that γ passes through an odd number of times are its endpoints (by definition), and hence A_v has exactly the effect of flipping the eigenvalues at A_{v_0} and A_{v_1}. Thus, the image of a point in $\mathcal{N}_{\mathbf{v},\mathbf{p}}$ is in $\mathcal{N}_{\mathbf{v}-\{v_0,v_1\},\mathbf{p}}$, as desired.

Combining, we find that the $-2n^2 + 2q'$ eigenstate can be decomposed as direct sums of $\mathcal{N}_{\mathbf{v},\mathbf{p}}$, where $\mathbf{v}$ and $\mathbf{p}$ range over even sized sets with $|\mathbf{v}+\mathbf{p}| = q'$.

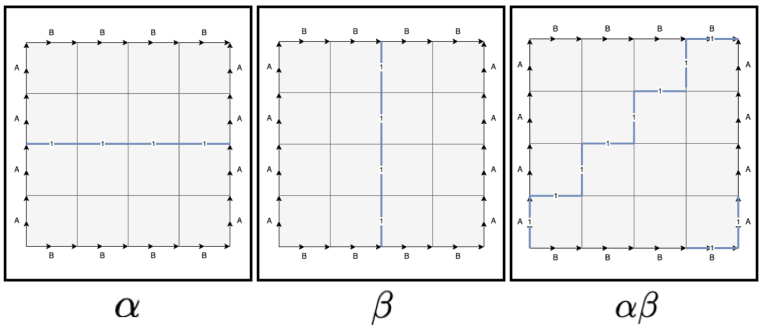

Figure 2.2: The three non-trivial homology classes of a torus

In particular, $q' = 2q$ must be even. The dimension of this space is equal to D times the number of way of choosing the sets $\mathbf{v}$ and $\mathbf{q}$, i.e,

$$D \sum_{k=0}^{q} \binom{n^2}{2k} \binom{n^2}{2(q-k)}.$$

The total dimension of $\mathcal{N}$ can be computed as

$$\begin{aligned} D \sum_{q=0}^{2n^2} \sum_{k=0}^{q} \binom{n^2}{2k} \binom{n^2}{2(q-k)} &= D \left(\sum_{q=0}^{2n^2} \binom{n^2}{2k_0} \right) \\ &= D \cdot \left(2^{n^2-1}\right)^2 = D \cdot 2^{2n^2-2}. \end{aligned}$$

However, it is clear from the definition of $\mathcal{N} = \bigotimes_{\text{edges}} \mathbb{C}^2$ that the space is 2^{2n^2} dimensional. Hence, we conclude that $D = 4$, and our proof is complete. □

The fact that the code space is four dimensional can be motivated as follows. By Proposition (ii), being in the $+1$ eigenspace for each A_v and B_p will impose a condition that decreases the dimension of your space by $1/2$. Since $\mathcal{N}$ is 2^{2n^2} dimensional, imposing all n^2 of these conditions decreases the code space to 1 dimension. However, the fact that $\prod_{\text{vertices } v} A_v = I$ and $\prod_{\text{faces } p} B_p = 1$ from Proposition (iv) shows that two of these conditions imposed were redundant, brining the code space dimension back up to $2^2 = 4$ dimensions.

To describe the generators of the codespace explicitly we will need to use the basics of homology theory with $\mathbb{Z}_2$ coefficients, where $\mathbb{Z}_2 = \{0, 1\}$ is the additive group modulo 2. For those unfamiliar, a brief introduction is included in Appendix A.

A pure state on $\mathcal{N}$ is specified by a pure state on each qubit, namely, a choice of $|0\rangle$ and $|1\rangle$ for each edge. This is exactly the data to specify a $\mathbb{Z}_2$-chain. Given

a $\mathbb{Z}_2$-chain γ, we write $|\gamma\rangle$ for the associated pure state. Given any γ, γ', we write $\gamma \sim \gamma'$ to mean that γ and γ' are homologous. The following elucidates the meaning of the codespace of the toric code:

Proposition 2.3. *Let $\mathbf{0}, \alpha, \beta$, and $\alpha\beta$ be the four $\mathbb{Z}_2$ homology classes on the torus, as in Figure 2.2. Choose $\mathbf{0}_0$, α_0, β_0, and $(\alpha\beta)_0$ respectively to be representatives. Then, letting γ run over all $\mathbb{Z}_2$-cycles, we have that*

$$|\mathbf{0}\rangle = \frac{1}{\sqrt{2^{n^2-1}}} \sum_{\gamma \sim \mathbf{0}_0} |\gamma\rangle\,, \;\; |\alpha\rangle = \frac{1}{\sqrt{2^{n^2-1}}} \sum_{\gamma \sim \alpha_0} |\gamma\rangle\,,$$
$$|\beta\rangle = \frac{1}{\sqrt{2^{n^2-1}}} \sum_{\gamma \sim \beta_0} |\gamma\rangle\,, \;\; |\alpha\beta\rangle = \frac{1}{\sqrt{2^{n^2-1}}} \sum_{\gamma \sim (\alpha\beta)_0} |\gamma\rangle\,,$$

are all normalized eigenstates of that Hamiltonian H, and serve as a canonical orthonormal basis of the codespace.

Proof. Choose $\omega \in H_1(T; \mathbb{Z}_2)$. To show that $|\omega\rangle$ is in the codespace, we observe that it is in the $+1$ eigenspace of every A_v and B_p. Since σ_Z sends $|0\rangle$ to $|0\rangle$ and $|1\rangle$ to $-|1\rangle$, A_v has the effect of sending a pure state $|\gamma\rangle$ to $\pm|\gamma\rangle$, depending on whether γ has an odd or even count of edges touching the vertex v. In particular, because γ is running over cycles, we have that each $|\gamma\rangle$ is in the $+1$ eigenspace of all the A_v, and hence the same applies to $|\omega\rangle$.

For B_ps, we observe that applying B_p to a pure state $|\gamma\rangle$ has the effect of flipping all of the qubits around the face p. By definition of being $\mathbb{Z}_2$ homologous, B_p maps the space of all cycles homologous to $|\omega\rangle$ back into the space of all cycles homologous to $|\omega\rangle$. In particular, $|\omega\rangle$ is in the $+1$ eigenspace of B_p for every p.

To show that $|\omega\rangle$ is normalized, we observe that there are exactly that there are exactly 2^{n^2-1} cycles homologous to ω. This is proved as follows. Starting with a fixed representative ω_0 of ω, cylces homologous to ω correspond to flipping qubits around the edges, i.e, applying B_ps at faces. Since there are n^2 faces, this gives 2^{n^2} cycles. This overcounts the space of cycles homologous to ω by a factor of 2, since $\prod_{\text{faces } p} B_p = I$ by Proposition 2 (iv). The fact that the codespace is 4 dimensional says that this is the *only* relation between the B_ps. Hence, there are 2^{n^2-1} cycles.

To show that these states are mutually orthogonal, we observe simply that no cycle can be homologous to two of the $\{\mathbf{0}, \alpha, \beta, \alpha\beta\}$, hence $\{|\mathbf{0}\rangle, |\alpha\rangle, |\beta\rangle, |\alpha\beta\rangle\}$ have disjoint support, hence they are orthogonal. □

Letting T denote the torus, the above shows that we can view the codespace of the toric code as a physical realization of the vector space $H_1(T; \mathbb{Z}_2) \otimes_{\mathbb{Z}} \mathbb{C} = H_1(T, \mathbb{C})$. Quantum physics gives the physical analogue of the abstract mathematical notation of equivalence class, namely, an equivilance class is realized as the superposition over all possible representatives. This can be compared with the path integral formulation of quantum mechanics, where one integrates over all possible paths between two points.

We now give *quasiparticle* interpretation of the toric code. A quasiparticle is an excitation in the toric code. Namely, given an eigenstate $|\psi\rangle$, we say that there is a quasiparticle at a vertex v if $A_v |\psi\rangle = -|\psi\rangle$, and get a face p we say that there is a quasiparticle at p if $B_p |\psi\rangle = -|\psi\rangle$.

Let $|\psi\rangle$ be an eigenstate, and let v_0, v_1 be adjacent verticies connected by an edge e. Suppose that there is a quasiparticle at v_0, and that there is not a quasiparticle at v_1. Let $|\psi'\rangle$ be the state obtained by applying σ_X to the edge e. We observe that

$$A_{v_0} |\psi'\rangle = A_{v_0} \left(\bigotimes_{\text{edge } e} \sigma_X \right) |\psi\rangle = -A_{v_0} |\psi\rangle = |\psi\rangle ,$$

$$A_{v_1} |\psi'\rangle = A_{v_1} \left(\bigotimes_{\text{edge } e} \sigma_X \right) |\psi\rangle = -A_{v_1} |\psi\rangle = -|\psi\rangle ,$$

where we keely used the fact that $\sigma_X \sigma_Z = -\sigma_Z \sigma_X$. Additionally, $A_v |\psi'\rangle = A_v |\psi\rangle$ for $v \neq v_0, v_1$, since applying σ_X to e only affects the verticies v_0 and v_1. We can interpret this computation as saying the following: Applying σ_X has the effect of *moving the quasiparticle along* e, from v_0 to v_1. Applying longer chains of σ_Xs, we see in general that applying σ_X corresponds to moving quasiparticles at verticies along the edges. If neither v_0 nor v_1 had quasiparticles, then again tensoring with σ_X at e would have the effect of flipping the eigenvalues at v_0 and v_1, i.e, the effect of *creating quasiparticles at the endpoints of* e, at v_0 and v_1. If both v_1 and v_1 has quasiparticles, then tensoring with σ_X at e would have the effect of *anhilating quasiparticles at the endpoints of* e.

In summary, the quasiparticles at edges are their own antiparticle. Creating particle/antiparticle pairs, moving the quasiparticles, and annihilating particle/antiparticle pairs all are mathematically realized by the simple operation of tensoring edges with σ_X.

Similarly, we can describe the quasiparticles sitting at faces, corresponding to faces with an excitation $B_p |\psi\rangle = -|\psi\rangle$. Tensoring with σ_X has no effect on these quasiparticles, since tensoring with σ_X at any edge commutes with all the B_p: σ_X commutes with itself. However, it is now tensoring with σ_Z that causes the motion of particles. Given faces p_0, p_1 with common edge e, tensoring with σ_Z at e has the effect of moving a quasiparticle from p_0 to p_1 if exactly one of the faces had a quasiparticle, has the effect of creating a particle/antiparticle pair if neither of the faces has quasiparticles, and has the effect of annihilating a particle/antiparticle pair if both faces have quasiparticles.

Summarizing, we find that the toric code naturally have two types of quasiparticles, an X-type that lives on vertices which moves by tensoring by σ_X, and a Z-type that lives in faces and moves by tensoring with σ_Z. This allows us to mathematically implement a topological quantum computer. Namely:

1. Quantum information is stored in the ground state of the toric code, i.e, the lowest eigenvalue eigenspace of the Hamiltonian.

2. Ground states are acted on by generating and manipulating quasiparticles, moving them around the torus, and annihilating them. Mathematically, this is realized by repeatedly tensoring with σ_X and σ_Z along edges, until one returns to a ground state.

3. Quantum information is measured by observing the ground state with respect to the canonical orthonormal basis of the codespace, given in Proposition 2.3

As an example, we implement the "NOT_α" gate, which flips the input state depending on whether it has an α component, namely

$$|\mathbf{0}\rangle \mapsto |\alpha\rangle\,,\ |\beta\rangle \mapsto |\alpha\beta\rangle$$
$$|\alpha\rangle \mapsto |\mathbf{0}\rangle\,,\ |\alpha\beta\rangle \mapsto |\beta\rangle\,.$$

Proposition 2.4. *The following computation has the effect of performing the NOT_α gate. First, generate a particle/antiparticle pair of X-type particles. Then, move one of the particles around the torus via a path homologous to α. Finally, fuse your two adjacent X-type particles together.*

Proof. Let v_0 and v_1 be adjacent vertices. Let α_0 be a cycle homologous to α going from v_1 to itself. The process described in the statement of the proposition can be reworded as saying that the following. First, we create a particle pair at v_0 and v_1, i.e, we tensor with σ_X at the edge connecting v_0 and v_1. Then we move v_1 along α_0, i.e, we tensor with σ_X along the edges in α_0. Then, we fuse the X-type quasiparticles at v_0 and v_1 back together, i.e, we tensor along the edge connecting v_0 and v_1 again. Since $\sigma_X^2 = 1$, this whole process can be described mathematically as

$$\bigotimes_{\text{edges in } \alpha_0} \sigma_X.$$

Seeing as σ_Xs corresponds to flipping $|0\rangle$s to $|1\rangle$s in pure states, this process has the effect of flipping all of the qubits along α_0. On the level of cycles, this means that we take $|\gamma\rangle$ to $|\gamma + \alpha_0\rangle$, where addition is in the group of cycles. Seeing as adding a cycle homologous to α to a cycle homologous to ω results in a cycle homologous to $\omega + \alpha$ we find thus that this process has the effect of sending $|\omega\rangle$ to $|\omega + \alpha\rangle$. Seeing as $\alpha + \alpha = 0$ in $H_1(T; \mathbb{Z}_2)$, this process is exactly the NOT_α gate. □

Similarly, we can implement the "$(-1)_\alpha$" gate, which flips the input state depending on whether it has an α component, namely

$$|\mathbf{0}\rangle \mapsto |\mathbf{0}\rangle\,,\ |\beta\rangle \mapsto |\beta\rangle$$
$$|\alpha\rangle \mapsto -|\alpha\rangle\,,\ |\alpha\beta\rangle \mapsto -|\alpha\beta\rangle\,.$$

Proposition 2.5. *The following computation has the effect of performing the (-1)$_\alpha$ gate. First, generate a particle/antiparticle pair of Z-type particles. Then, move one of the particles around the torus via a path homologous to β. Finally, fuse your two adjacent Z-type particles together.*

Proof. Let p_0 and p_1 be adjacent faces. Let β_0 be a cycle homologous to β going from p_1 to itself. Note that since Z-type particles live on faces, β_0 does not consist of a series of edges. Instead, it is a path going through the centers of faces. We take $\widehat{\beta}_0$ to be the set of edges that β_0 passes through. Tensoring with σ_Z along $\widehat{\beta}_0$ corresponds to motion of a particle from p_1 along β_0 back to itself.

These cycles that go through faces of the torus are called *dual cycles*, and are standard practice in the theory of homology. Namely, they are cycles in the dual complex. Whereas the edges associated with a normal cycle satisfy the property 'every vertex touches an even number of 1s', the edges associated with a cycle in the dual complex satisfy the dual condition 'every face touches an even number of 1s'.

As before, we find that the whole process can be described mathematically as

$$\bigotimes_{\text{edges in } \widehat{\beta}_0} \sigma_Z.$$

The matrix σ_Z acts on pure states by sending $|0\rangle$ to itself, and $|1\rangle$ to $-|1\rangle$. Thus, this process has the effect of introducing a -1 global phase shift for every 1 in states along $\widehat{\beta}_0$. Thus, when acting on a pure state γ, the definition of $\widehat{\beta}_0$ shows that this process has the effect of introducing a phase shift of -1 to the power of the number of intersection between γ and $\widehat{\beta}_0$.

It is a well known fact that this signed intersection number (-1 to the power of the number of intersections) is an invariant in $\mathbb{Z}_2$ homology. To see this, observe that changing representatives of a homology class correspond to flipping qubits around a face. By the 'dual cycle' condition, this flipped face will touch an even number of elements in the dual cycle. Hence, the intersection number will change by an even amount, leaving -1 to the power of that number invariant.

In particular, $\mathbf{0}$ doesn't intersect β, β doesn't intersect β (representatives can be chosen to be parallel), α intersects β (horizonal loops and vertical loops meet at exactly one point), and $\alpha\beta$ intersects β. Thus, this process has the effect of adding a -1 phase shift to those states which include an 'α', as desired. □

Sadly for the toric codes, these are essentially the only gates that can be implemented. No matter how one moves around particles, there is not enough complexity in the system to generate interesting gates. We formalize this by writing out the *group of gates* of the toric codes. We can think of quantum gates on a system as forming a group, where the group law is given by the composition of gates, and every element has an inverse since unitary matrices are invertible. For the toric codes, we have the following:

Proposition 2.6. *There are exactly 8 possible computations in the toric codes. The group of gates is (non canonically) isomorphic to the Pauli group, i.e, the group whose objects are*

$$\{\pm I, \pm iI, \pm\sigma_X, \pm i\sigma_X, \pm\sigma_Y, \pm i\sigma_Y, \pm\sigma_Z, \pm i\sigma_Z\}$$

and whose group operation is given by matrix multiplication. A minimal generating set is given by $\{NOT_\alpha, NOT_\beta, (-1)_\alpha\}$.

Proof. To begin, we define $\mathrm{NOT}_\beta, \mathrm{NOT}_{\alpha\beta}, (-1)_\beta$, and $(-1)_{\alpha\beta}$ in complete analogy to how we define NOT_α and $(-1)_\alpha$. Namely, NOT_β flips whether or not a state has a 'β' in it, and $\mathrm{NOT}_{\alpha\beta}$ flips whether or not a state has an '$\alpha\beta$' in it, i.e,

$$|\mathbf{0}\rangle \mapsto |\alpha\beta\rangle\,,\ |\beta\rangle \mapsto |\alpha\rangle$$
$$|\alpha\rangle \mapsto |\beta\rangle\,,\ |\alpha\beta\rangle \mapsto |\mathbf{0}\rangle\,.$$

The relation $\sigma_X\sigma_Z = -\sigma_Z\sigma_X$ implies that we can switch the order of operations between first applying all our σ_Xs and then applying all our σ_Zs, up to an operator-wise phase shift -1. Any process of creating and anhilating X-type quasiparticles can be modeled in sequence as repeatedly creating particles, moving them around a loop, then annihilating them. Following the proof of Proposition 2.5, this is the same as repeatedly applying NOT_ω gates, for homology classes ω. Similarly, the processes on Z-type particles will be compositions of $(-1)_\omega$ gates.

Hence, we now have a full set of generators for our gate group: $\{\pm I, \mathrm{NOT}_\omega, (-1)_\omega\}$, where ω runs over homology classes. The relations $\mathrm{NOT}_\alpha\mathrm{NOT}_\beta = \mathrm{NOT}_{\alpha\beta}$ and $(-1)_\alpha(-1)_\beta = (-1)_{\alpha\beta}$ allow one to reduce the generating set further. The relations

$$\mathrm{NOT}_\alpha(-1)_\alpha\mathrm{NOT}_\alpha = (-1)_\beta$$

and

$$(-1)_\alpha\mathrm{NOT}_\alpha(-1)_\alpha = -I$$

reduce the generating set to $\{\mathrm{NOT}_\alpha, \mathrm{NOT}_\beta, (-1)_\alpha\}$. Verifying simple relations gates, it is simple to see that the gate group is isomorphic to the Pauli group, as desired.

□

Before wrapping up we make a few final remarks about the behavior of quasiparticles on the toric codes. Namely, we observe the following. Consider the full Hilbert space $\mathcal{N}$, and adjacent two adjacent X-type and Z-type quasiparticles. Consider the simple braiding of these particles around each other, as in Figure 1.3. A line going under another corresponds to the particle having passed through that space first, before the other particle. This braiding can be obtained

by first performing a twist halfway around the circle by X, then a twist all the way around the circle by Z, then finally moving the second half of the circle by X. This process corresponds to a transformation $\mathcal{N} \to \mathcal{N}$ (i.e, tensoring with the appropriate σ_Xs and σ_Zs). The observation is that this transformation is *not* the identity on the codespace. Namely, since $\sigma_Z\sigma_X = -\sigma_X\sigma_Z$, this operation corresponds to a global phase shift of -1 on the system.

Thus, the braiding of X type and Z type particles corresponds to a phase shift of -1. This is in contrast to braiding two identical X type or Z type particles, which corresponds to the identity since σ_X and σ_Z commute with themselves. All particles in the standard model of physics are *fermions*, which give a phase shift of -1 when you braid them with themselves, or *bosons*, which act by the identity when you braid them with themselves. Seeing as X type and Z type particles in the toric code braid by the identity with themselves, one would expect them to be bosons. However, bosons always braid by the identity with each other and hence the -1 phase shift from braiding X and Z type particles should be impossible. The conclusion is that these really are *quasi*particles, which behave much differently than particles in the standard model. Quasiparticles with simultaneously non-bosonic and non-fermionic braiding rules are known as *anyons*. All interesting particles in topological quantum phases of matter will be anyons. Finding new and better forms of topological quantum computation corresponds to finding anyons with more and more interesting braiding rules.

A $\mathbb{Z}_2$ Homology Theory

In this appendix we introduce the basic notations of homology theory with $\mathbb{Z}_2$ coefficients, namely, chains, cycles, and homological equivalence. The settings for homology are *simplicial complexes*, which can be loosely thought of as collections of vertices, edges, and faces, with some edges and vertices identified, just as was done for the torus in this text. A $\mathbb{Z}_2$-chain on a space is an assignment of an element of $\mathbb{Z}_2$ to every edge, where $\mathbb{Z}_2 = \{0, 1\}$ is the additive group modulo 2. The set of $\mathbb{Z}_2$-chains forms a group under edge-wise addition. A $\mathbb{Z}_2$-cycle is a $\mathbb{Z}_2$-chain which can be obtained by starting at a vertex and walking along edges, flipping 1s to 0s and vice versa as you go along, and returning back where you started at the end. Equivalently, a $\mathbb{Z}_2$-cycle is a $\mathbb{Z}_2$-chain which has an even number of 1s touching each vertex. The $\mathbb{Z}_2$-cycles form a subgroup of the group of $\mathbb{Z}_2$-chains. Seeing as all chains and cycles discussed in these notes take coefficients in $\mathbb{Z}_2$, we ease notation by simply saying "chain" and "cycle".

The goal of homology theory is to describe cycles on a geometric object, up to deformations. If one cycle can be continuously deformed into another, then they should be considered equivalent. On the sphere, for example, all loops can be contracted away into nothing. On the torus there are four distinct cycles, namely, the zero cycle, the cycle that goes around the torus horizontally, the cycle that goes around the torus vertically, and the cycle that twists around the torus, as in Figure 2.2 These non-trivial cycles correspond exactly to the

continuous vector fields described in the introduction [Fra57].

Loosely, we will call two cycles homologically equivalent if they can be continuously deformed one to another. Given any face, the cycle consisting of 1s along the edges touching that face should be 'homologically trivial", i.e, homologically equivalent to the 0 cycle, since it can be contracted away into nothingness. In a strong sense, this is the only condition one needs to impose. Letting X be our simplicial complex, we let $C_1(X;\mathbb{Z}_2)$ be the group of chains. We let $Z_1(X;\mathbb{Z}_2)$ be the subgroup generated by the cycles consisting of 1s the boundaries of faces. This is the subgroup of homologically trivial cycles. This lets us define the quotient

$$H_1(X;\mathbb{Z}_2) = C_1(X;\mathbb{Z}_2)/Z_1(X;\mathbb{Z}_2),$$

called the (1st) homology group of X. Two elements are called homologically equivalent if they are in the same coset of $H_1(X;\mathbb{Z}_2)$. Alternatively, two elements are homologically equivalent if one can be obtained from the other by repeatedly flipping 1s and 0s along the boundaries of squares.

It is a well known fact that the first homology group of the torus has four elements, corresponding to the zero class, the horizontal cycle around the torus, the vertical cycle around the torus, and the diagonal cycle.

The observant reader might find the above discussion frustrating. In particular, we seem to be using the following intuitions interchangeably:

1. Cycles being continuously deformed to each other
2. Cycles that can be obtained from one another by flipping edges along the boundary of faces.

The worry regarding the distinction between these two notions is justified. In general, the group obtained by imposing the equivalence relation of continuous deformations will not be equal to the homology group $H_1(X)$. The group resulting by imposing the continuous deformation restriction is called the *fundamental group* of X, and is denoted $\pi_1(X)$. In general $\pi_1(X)$ can be quite a bit larger than $H_1(X)$, i.e, the equivalence relation can be weaker. The groups are always related by the fact that $H_1(X)$ is canonically isomorphic to the abelianization of $\pi_1(X)$, i.e, the maximal abelian quotient of $H_1(X)$. In the case that $\pi_1(X)$ is abelian (for example, when X is a torus), this means that there is no distinction between these spaces, and one should not make any worry about the discrepancies in intuition.

The canonical reference for this subject (known as *Algebraic Topology*) is Alan Hatcher's textbook [Hat05].

References

[AA81] Marcia Ascher and Robert Ascher. Code of the quipu. *Ann Arbor: University of Michigan Press*, pages 56–74, 1981.

[AAB+19] Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C Bardin, Rami Barends, Rupak Biswas, Sergio Boixo, Fernando GSL Brandao, David A Buell, et al. Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779):505–510, 2019.

[Day21] Cyrus Lawence Day. *Quipus and Witches' Knots: The Role of the Knot in Primitive and Ancient Culture, with a Translation and Analysis of" Oribasius de Laqueis"*. University Press of Kansas, 2021.

[DLF+16] Shantanu Debnath, Norbert M Linke, Caroline Figgatt, Kevin A Landsman, Kevin Wright, and Christopher Monroe. Demonstration of a small programmable quantum computer with atomic qubits. *Nature*, 536(7614):63–66, 2016.

[FKLW03] Michael Freedman, Alexei Kitaev, Michael Larsen, and Zhenghan Wang. Topological quantum computation. *Bulletin of the American Mathematical Society*, 40(1):31–38, 2003.

[FKW02] Michael H Freedman, Alexei Kitaev, and Zhenghan Wang. Simulation of topological field theories by quantum computers. *Communications in Mathematical Physics*, 227:587–603, 2002.

[FLW02] Michael H Freedman, Michael Larsen, and Zhenghan Wang. A modular functor which is universal for quantum computation. *Communications in Mathematical Physics*, 227:605–622, 2002.

[Fra57] Theodore Frankel. Homology and flows on manifolds. *Annals of Mathematics*, pages 331–339, 1957.

[Got98] Daniel Gottesman. Theory of fault-tolerant quantum computation. *Physical Review A*, 57(1):127, 1998.

[Hat05] Allen Hatcher. *Algebraic topology*. Tsinghua University Press, 2005.

[Kan98] Bruce E Kane. A silicon-based nuclear spin quantum computer. *nature*, 393(6681):133–137, 1998.

[Kit03] A Yu Kitaev. Fault-tolerant quantum computation by anyons. *Annals of physics*, 303(1):2–30, 2003.

[OKMH22] Yun-Tak Oh, Jintae Kim, Eun-Gook Moon, and Jung Hoon Han. Rank-2 toric code in two dimensions. *Physical Review B*, 105(4):045128, 2022.

[Wen17] Göran Wendin. Quantum information processing with superconducting circuits: a review. *Reports on Progress in Physics*, 80(10):106001, 2017.

[YZB+21] Fangyuan Yang, Alexander A Zibrov, Ruiheng Bai, Takashi Taniguchi, Kenji Watanabe, Michael P Zaletel, and Andrea F Young. Experimental determination of the energy per particle in partially filled landau levels. *Physical review letters*, 126(15):156802, 2021.