Topological Quantum Error Correcting Codes Part II

Isaac Smith

July 25, 2019

1 Recall from last time

Theorem 1.5. ***(Quantum error-correcting conditions)*** *Let $\mathcal{C}$ be a quantum code and let $P_\mathcal{C}$ be the proejctor onto $\mathcal{C}$. Suppose $\mathcal{E}$ is a quantum operation with elements $\{E_i\}$. There exists an error-correction procedure $\mathcal{R}$ that satisfies*

$$\mathcal{R}(\mathcal{E}(|\psi\rangle)) \propto |\psi\rangle$$

if and only if

$$P_\mathcal{C} E_j^\dagger E_i P_\mathcal{C} = \alpha_{ij} P_\mathcal{C}$$

where $\alpha = (\alpha_{ij})$ is a Hermitian matrix over $\mathbb{C}$.

2 Stabilizer Formalism and Stabilizer Codes

Definition 2.0.1. The **general Pauli group** on n qubits, G_n, is the group with elements that are n-fold tensor products of the following

$$\{\pm I, \pm iI, \pm X, \pm iX, \pm Y, \pm iY, \pm Z, \pm iZ\}$$

Definition 2.0.2. Let $S \subset G_n$ be a subgroup. Define V_S to be the subspace of the n-qubit state space such that for all $v \in V_S$ and $g \in S$, $gv = v$. We say that S is the **stabilizer** for V_S.

Typically, we take as part of the definition of a stabilizer group that $-I \notin S$ and that all elements of S commute, otheriwse V_S is trivial.P We tend to write the stabilizer group S in terms of its generators, that is, $S = \langle g_1, ..., g_l\rangle$ and we often insist that these generators be independent.

Proposition 2.1. *Let $S = \langle g_1, ..., g_{n-k}\rangle$ be generated by $n-k$ independent and commuting elements of G_n such that $-I \notin S$. Then V_S is a 2^k-dimensional subspace.*

Proof. Let $S = \langle g_1, ..., g_{n-k} \rangle$ be generated by $n-k$ independent and commuting elements of G_n such that $-I \notin S$ and let $x = (x_1, ..., x_{n-k}) \in \mathbb{Z}_2^{n-k}$. Define the projectors

$$P_S^x = \frac{\prod_{j=1}^{n-k}(I + (-1)^{x_j} g_j)}{2^{n-k}}$$

For each j, the projector $(I + g_j)/2$ projects onto the $+1$-eigenstate of g_j, so we can write the projector onto V_S as $P_S^{(0,0,...,0)}$. Now consider the following lemma.

Lemma. ***(Proposition** 10.4 **in Ref [1])*** *Let S be as above. Fix i in the range 1 to $n-k$. Then there exists $g \in G_n$ such that $g g_i g^\dagger = -g_i$ and $g g_j g^\dagger = g_j$ for all $j \neq i$.*

Proof. Sketch: The g_i are all independent which means that for each i, there exists a subset of position in the tensor product of g_i that contains non-identity elements of the Pauli group, such that no other generator contains the same elements in the same positions. Defining $g \in G_n$ to be the element that is the tensor product of I in all positions except for the marked positions of g_i, in which positions g contains a tensor product of elements of the Pauli group that anticommutes with the tensor product of elements in the same positions of g_i.

For the full proof, written in terms of check matrices, refer to Ref [1]. □

In light of the above lemma, let us define, for any $x' \in \mathbb{Z}_2^{n-k}$,

$$h_{x'} = h_1^{x'_1} ... h_{n-k}^{x'_{n-k}}$$

where h_i is such that

$$h_i g_i h_i^\dagger = -g_i \text{ and}$$
$$h_i g_j h_i^\dagger = g_j \text{ for all } j \neq i$$

and the notation $h_i^{x'_i}$ merely denotes that if $x'_i = 0$, then $h_i^{x'_i}$ is the trivial element so doesn't feature in the product $h_{x'}$. Then we see that

$$h_{x'} P_S^{(0,0,...,0)} h_{x'}^\dagger = P_S^{x'}$$

It follows that for each x', $P_S^{x'}$ has the same dimension as V_S. For distinct x, the P_S^x are orthogonal since the intersection of $+1$- and -1eigenstates of any of the projectors is trivial. We also note that

$$\sum_x P_S^x = I$$

and since I is a projector onto a 2^n-dimensional space, and the sum is over 2^{n-k} dimensional orthogonal projectors all of the same dimensions, it follows that each projector, and in particular V_S, has dimension 2^k. □

Remark. Notice the similarities between the projectors P_S^x defined above and the projectors P_k defined in the proof of Theorem 1.5 from last week. This suggests that the operation elements for the error-correcting procedure will be related to $g_x^\dagger P_S^x$.

Proposition 2.2. *Suppose $S = \langle g_1, ..., g_l \rangle$ stabilizes V_S and the g_i all commute and $-I \notin S$. Then, for any unitary operation U, the space UV_S is stabilized by $USU^\dagger$ and moreover, $USU^\dagger$ is generated by $Ug_1U^\dagger, ..., Ug_lU^\dagger$.*

Proof. Let $|\psi\rangle \in V_S$. Let $g \in S$ be arbitrary and let U be a unitary operation. Then we get that

$$\begin{aligned} U|\psi\rangle &= Ug|\psi\rangle \\ &= UgU^\dagger U|\psi\rangle \end{aligned}$$

Hence, $UgU^\dagger$ stabilizes $U|\psi\rangle$ for all $g \in S$, so UV_S is stabilized by $USU^\dagger$. Since any element of S can be written as the product $g_{i_1} g_{i_2} ... g_{i_m}$ it follows that any element of $USU^\dagger$ can be written

$$Ug_{i_1} g_{i_2} ... g_{i_m} U^\dagger = Ug_{i_1}U^\dagger Ug_{i_2}U^\dagger ... Ug_{i_m}U^\dagger$$

so $Ug_1U^\dagger, ..., Ug_lU^\dagger$ generate $USU^\dagger$. □

Remark. The above proposition work for any unitary U, but only 'plays nicely' when U is in the normaliser of G_n, that is $UG_nU^\dagger = G_n$. For other unitaries, the resulting stabilizer group is often messy to work with. As it turns out, all encoding, decoding, error detection and recovery in quantum stabilizer codes can be enacted with unitaries from the normaliser of G_n.

We arrive at the definition of a stabilizer code.

Definition 2.0.3. An $[n, k]$ **stabilizer code** $C(S)$ is a vector space V_S stabilized by a subgroup $S \subset G_n$ such that $-I \notin S$ and S has $n - k$ independent and commuting elements.

We know by a previous proposition that this means that V_S is 2^k-dimensional, and we can choose any orthonormal basis of V_S to act as our computational basis (this can be done in a systematic way).

Theorem 2.3. ***(Error-correction conditions for stabilizer codes)*** *Let S be the stabilizer for a stabilizer code $C(S)$. Suppose $\{E_j\}$ is a set of operators in G_n such that $E_j^\dagger E_k \notin N(S) - S$ for all j, k, where $N(S)$ denotes the normalizer of S. Then $\{E_j\}$ is a correctable set of errors for the code $C(S)$.*

Proof. Let P be the projector onto $C(S)$ and let $\{E_i\}$ be a set of operators in G_n such that $E_j^\dagger E_k \notin N(S) - S$ for all j, k. This means that either $E_j^\dagger E_k \in S$ or $E_j^\dagger E_k \in G_n - N(S)$. First suppose that $E_j^\dagger E_k \in S$. Then $PE_j^\dagger E_k P = gP$ for some $g \in S$. But P is invariant under multiplication of elements of the stabilizer of the space that P projects onto. Thus $PE_j^\dagger E_k P = P$, so by Theorem 1.5, the set $\{E_i\}$ is correctable.

Let us note that the normaliser of S equals the centraliser of S in this case, since we are implicitly assuming $-I \notin S$. The centraliser of S is always contained in the normaliser of S, denoted $N(S)$, so we need only shown the other inclusion, which we do by contradiction. Suppose $E \in N(S)$ is not in the centraliser for S. In particular that means

$$EIE^\dagger = g \neq I$$

However, $EIE^\dagger = EE^\dagger = \pm I$ for all $E \in G_n$, which forms a contradiction.

Now suppose that, for all j, k with $j \neq k$, $E_j^\dagger E_k \in G_n - N(S)$. Thus there exists a $g \in S$ such that $E_j^\dagger E_k$ anticommutes with g (since the normaliser and centraliser of S are the same). Let $g_1, ..., g_{n-k}$ be the generators for S, so that we can define P as

$$P = \frac{\prod_{i=1}^{n-k}(I + g_i)}{2^{n-k}}$$

Suppose without loss of generality that $E_j^\dagger E_k$ anticommutes with g_1. Then we have

$$\begin{aligned} PE_j^\dagger E_k P &= P\Big((I - g_1)E_j^\dagger E_k \frac{\prod_{i=2}^{n-k}(I + g_i)}{2^{n-k}}\Big) \\ &= \frac{\prod_{i=2}^{n-k}(I + g_i)}{2^{n-k}}(I + g_1)\Big((I - g_1)E_j^\dagger E_k \frac{\prod_{i=2}^{n-k}(I + g_i)}{2^{n-k}}\Big) \\ &= 0 \end{aligned}$$

since $(I + g_1)(I - g_1) = 0$, that is, the intersection of the $+1$- and -1-eigenspaces of g_1 is trivial. Thus,

$$PE_j^\dagger E_k P = 0$$

This completes the proof that the set $\{E_i\}$ is correctable when $E_j^\dagger E_i \notin N(S) - S$. □

The above theorem is a useful result but does not explain how to go about error-detection and correction. Suppose $C(S)$ is an $[n, k]$ stabilizer code with $S = \langle g_1, ..., g_{n-k}\rangle$ and $\{E_j\}$ is a set of correctable errors for the code. Error-detection is performed by measuring the generators $g_1, ..., g_{n-k}$ in turn to obtain the **error syndrome**, that is, the measurement results $\beta_1, ..., \beta_{n-k}$. If an error E_j occurs, then $E_j g_l E_j^\dagger = \beta_l g_l$. If E_j is the unique error that produces this measurement outcome β_l then applying $E_j^\dagger$ corrects the error. Furthermore, if E_j and $E_{j'}$ are distinct errors that produce the same error syndrome β_l then $E_j^\dagger E_{j'} \in S$ so applying either $E_j^\dagger$ or $E_{j'}^\dagger$ corrects the error in this case. This notion of correcting any error corresponding to the given syndrome is a big feature of the toric code and other topological codes. The previous theorem provides us with an opportunity to write down some analogous definitions to certain features of classical codes in the quantum setting.

Definition 2.0.4. The **weight** of an error $E \in G_n$ is the number of terms in the tensor product that aren't equal to the identity. The **distance** of a stabilizer code $C(S)$ is the minimum weight of an element of $N(S) - S$. If $C(S)$ has distance d, we say that it is an $[n, k, d]$ stabilizer code. If $d \geq 2t + 1$ then $C(S)$ can correct arbitrary errors on t qubits.

Example 2.0.1. (**Three qubit bit- and phase-flip codes**) We can write the three qubit bit-flip code as a stabilizer code with $S = \langle Z_1 Z_2, Z_2 Z_3 \rangle$ where it is implictly assumed that an identity acts on the other qubit. It is easy to see that V_S is the subspace spanned by $|000\rangle$ and $|111\rangle$ just as we had before. This time instead of measuring the four projectors, we measure the two syndrome measurements. No error corresponds to measuring $Z_1 Z_2$ and $Z_2 Z_3$ both in their $+1$-eigenstates, while an X_1 error corresponds to $-1, +1$, an X_2 error corresponds to $-1, -1$ and an X_3 error corresponds to $+1, -1$.

Similarly, the three qubit phase-flip code is stabilized by $\langle X_1 X_2, X_2 X_3 \rangle$ and error detection proceeds by measuring these generators analogously to the bit-flip case: $+1, +1$ corresponds to no error, $-1, +1$ corresponds to a Z_1 error, and so on.

These simple examples do not portray the power of the stabilizer formalism, but it will become much more apparent when we turn our attention to the Toric code and other topological codes. But first, let us quickly return to the Shor code in the stabilizer formalism.

Example 2.0.2. (**Shor Code**) The generators of the stabilizer of the Shor code are

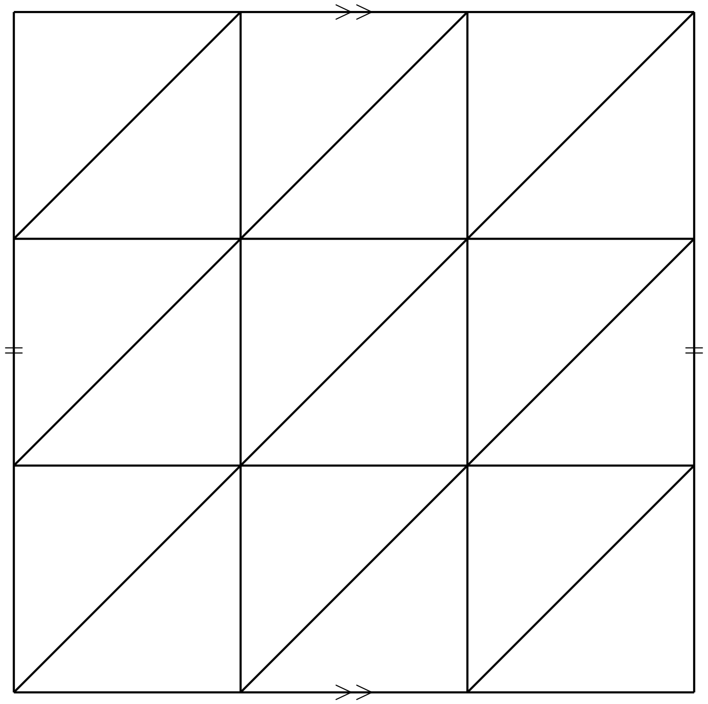
$$\begin{aligned} g_1 &= ZZIIIIIII \\ g_2 &= IZZIIIIII \\ g_3 &= IIIZZIIII \\ g_4 &= IIIIZZIII \\ g_5 &= IIIIIIZZI \\ g_6 &= IIIIIIIZZ \\ g_7 &= XXXXXXIII \\ g_8 &= IIIXXXXXX \end{aligned}$$

It can be shown that these generators stabilize $|0\rangle_L$ and $|1\rangle_L$ defined earlier, as well as that every product of two single qubit errors is either in S or anticommutes with at least one element of S (and hence not in $N(S)$) which proves that the Shor code can correct for an arbitrary single qubit error.

2.1 The Toric Code

Recall from Definition 1.1.1 from last week that a quantum error correcting code is a subspace of the state space of the quantum system being considered. So what quantum system is being considered here? Let us introduce the Toric code by first constructing the state space of the system, then analysing the subspace of the system that corresponds to the code. The Toric code is a stabilizer code, so we will detail the set of stabilizers that define the codespace as per the general theory in the previous section.

Consider a square lattice on the torus. Note that we saw a couple of weeks ago a natural way of finding this square lattice, by considering the following triangulation



and then "forgetting" all the diagonal lines of the lattice (see Figure 1.)

Once we have the square lattice, attach a qubit to every edge. Let V denote the set of vertices, E the set of edges and P the set of plaquettes (not to be confused with errors E or projectors P - this should be clear from context). So the state space for the lattice is $2^{|E|}$-dimensional. Now let us write down the stabilizers for the Toric code. There are two types of operators

$$A_v = \bigotimes_{j \ni v} \sigma_j^X$$

$$B_p = \bigotimes_{j \in \partial P} \sigma_j^Z$$

where the products are over the edges that contain the vertex v for the first operator, and over the edges that bound a plaquette p in the second operator. The tensor product with the identity on all other qubits of the lattice is implied. These operators are defined for all vertices and plaquettes and are all clearly elements of G_n, and none are equal to $-I$. Furthermore, all these operators commute since they either share 0 or 2 edges, so even though X and Z Pauli operators anticommute, there is never an odd number of edges for which X and Z are both applied. Thus $S = \langle A_v, B_p | \forall v \in V, p \in P \rangle$ is a valid stabilizer group. Now these stabilizers are not all independent; they satisfy the following relations

$$\prod_v A_v = 1 \text{ and } \prod_p B_p = 1$$

This means that there are $|V| + |P| - 2$ independent stabilizers. Now we know from Theorem 2.1 that the dimension of V_S is given by $n - m$ where n is the number of physical qubits used in the code, and m is the

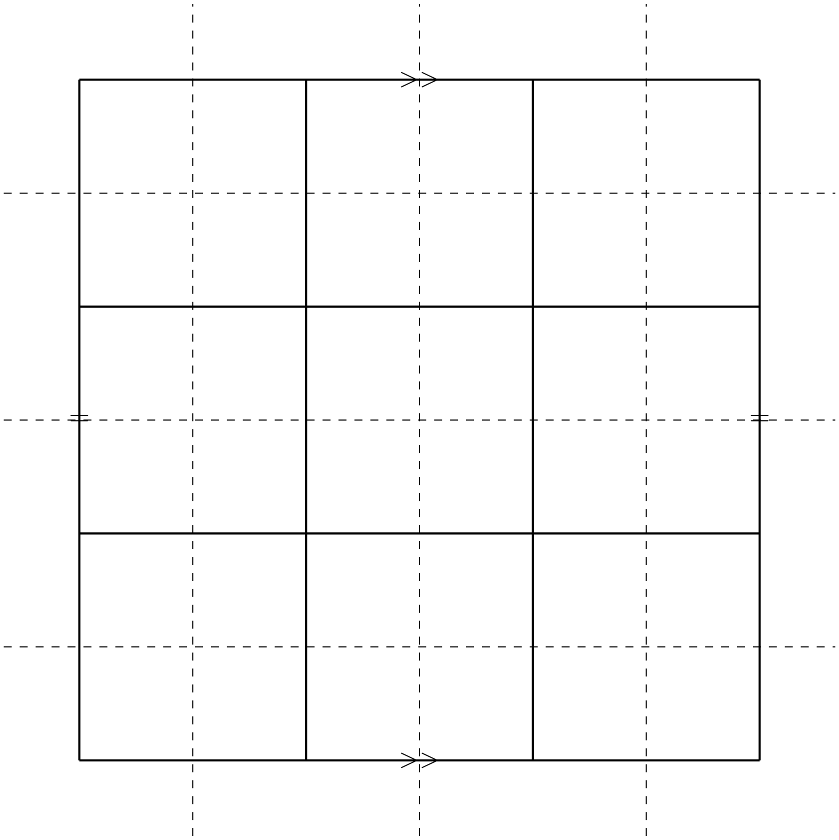


Figure 1: Primal graph (solid lines) and dual graph (dashed lines).

number of independent stabilizers. So, without even considering a specific lattice, we can write down the dimension of the codespace:

$$\begin{aligned}\dim(V_S) &= 2^{|E|-(|V|+|P|-2)} \\ &= 2^{2-\chi} \\ &= 2^{2g+b}\end{aligned}$$

where χ is the Euler characteristic, g is the genus and b is the number of boundary components ($b = 0$ in this case). The relation of dimension of the codespace to the Euler characteristic has been written this way to indicate that this generalises to other surface than the torus, and also to indicate it's invariance, that is, *any graph arising from any cellular decomposition of the surface will produce a codespace of this dimension.* So, for the torus, we obviously have $\dim(V_S) = 4$.

Let $\mathcal{H}$ denote the Hilbert space that is the state space of the lattice. Then we can write $V_S \subseteq \mathcal{H}$ as

$$V_S = \{|\psi\rangle \in \mathcal{H} | A_v|\psi\rangle = |\psi\rangle, B_p|\psi\rangle = |\psi\rangle, \text{ for all } v, p\}$$

The aim now is to relate the codespace to the (co)homology of the torus, as this will inform the error-correction procedure. We will do this using the terminology of homology on the primal graph and homology on the dual graph. Let us set up the notation as follows. Let C_0, C_1 and C_2 denote the free $\mathbb{Z}_2$-vector space over vertices (0-cells), edges (1-cells) and plaquettes (2-cells) in the primal graph respectively. Similarly, denote be C'_0, C'_1 and C'_2 the corresponding vector spaces for vertices, edges and plaquettes in the dual graph. We have the following chain complexes

$$... \to 0 \to C_2 \xrightarrow{\partial_2} C_1 \xrightarrow{\partial_1} C_0 \to 0$$
$$... \to 0 \to C'_2 \xrightarrow{\partial'_2} C'_1 \xrightarrow{\partial'_1} C'_0 \to 0$$

and let us denote the homology groups for the primal graph and dual graph by $H_*(T;\mathbb{Z}_2)$ and $H'_*(T;\mathbb{Z}_2)$ respectively.

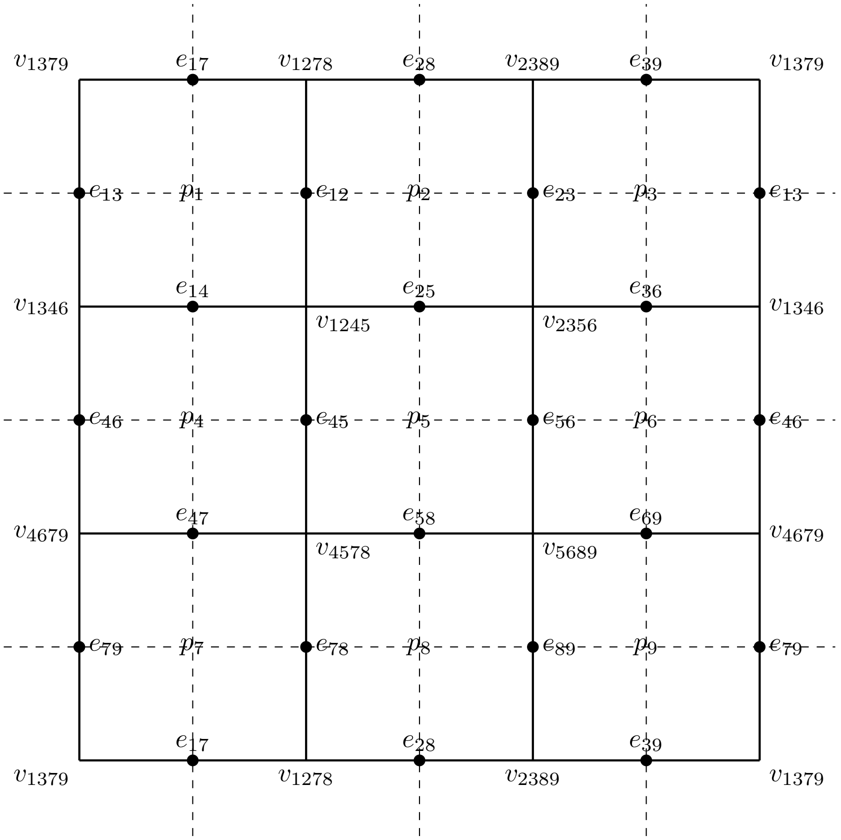
Let us choose a basis for $\mathcal{H}$ by assigning a label $t_j = 0, 1$ to every edge j of the primal graph (which also assigns the same label to the corresponding edge in the dual graph). Denote this basis by $\mathfrak{B}$. An element $|\psi\rangle \in \mathfrak{B}$ then corresponds to some element in C_1 (and C'_1). Moreover, enforcing the constraint that an element $|\psi\rangle \in \mathfrak{B}$ satisfy $B_p|\psi\rangle = |\psi\rangle$ for all p corresponds to the corresponding element to $|\psi\rangle$ in C'_1 lying in the kernel of ∂'_1. We can equivalently think about this as requiring the sum of edges around any plaquette equalling $0 \mod 2$.

Then, we can take basis vectors for V_S related to the homology classes in the dual graph setting as follows

$$|\psi_{00}\rangle := \sum_\alpha a_\alpha|\alpha\rangle$$
$$|\psi_{01}\rangle := \sum_\beta b_\beta|\beta\rangle$$
$$|\psi_{10}\rangle := \sum_\gamma c_\gamma|\gamma\rangle$$
$$|\psi_{11}\rangle := \sum_\epsilon d_\epsilon|\epsilon\rangle$$

where the α are the elements of C'_1 that correspond to the homology class $(0, 0)$, the β correspond to the class $(0, 1)$ and so on. The constraints arising from the A_v ensure that all the a_α are equal, b_β are equal, c_γ are equal and d_ϵ are equal. Let us see this more closely in an example.

Example 2.1.1. (3×3 **Toric code**) Consider the following 3×3 lattice with boundary edges identified as per the usual identification of a square to give the torus.



First, let us write out somewhat explicitly the set of basis vectors of V_S as per the theory above. Let A denote the group generated by $\langle A_v \rangle$. Then

$$\begin{aligned}
|00\rangle &= \frac{1}{\sqrt{|A|}} \sum_{g \in A} g|0....0\rangle \\
|01\rangle &= \frac{1}{\sqrt{|A|}} \sum_{g \in A} g|e_{46}, e_{45}, e_{56} = 1, 0 \text{ else}\rangle \\
|10\rangle &= \frac{1}{\sqrt{|A|}} \sum_{g \in A} g|e_{17}, e_{14}, e_{47} = 1, 0 \text{ else}\rangle \\
|11\rangle &= \frac{1}{\sqrt{|A|}} \sum_{g \in A} g|e_{46}, e_{45}, e_{56}, e_{17}, e_{14}, e_{47} = 1, 0 \text{ else}\rangle
\end{aligned}$$

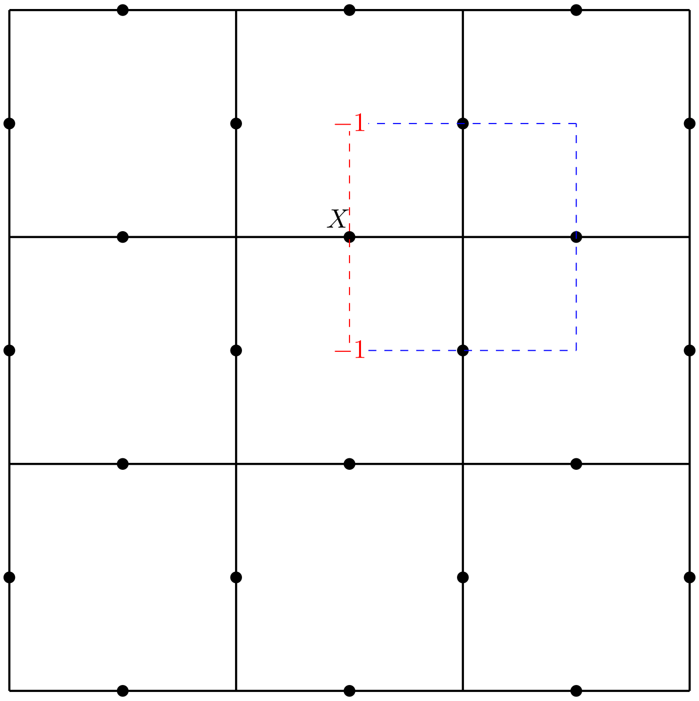
Now let us look at the effects of single errors. Suppose the code is initially in a state $|\psi\rangle$ prior to an X error occurring on the edge e_{25} (with identity operations occurring on all other edges implied), which we denote by X_{25}, producing the state $|\psi'\rangle = X_{25}|\psi\rangle$. There are four stabilizers that interact with the edge e_{25}, namely $B_2, B_5, A_{1245}, A_{2356}$. The interactions of these stabilizers with $|\psi'\rangle$ allows us to diagnose the error (the measurement result of the stabilizers is called the **error syndrome**). The vertex stabilizers commute with X errors:

$$\begin{aligned} A_v|\psi'\rangle &= A_v X_{25}|\psi\rangle \\ &= X_{25}A_v|\psi\rangle \\ &= |\psi'\rangle \end{aligned}$$

so $|\psi'\rangle$ is in the +1-eigenspace of all the A_v. However, the X error anticommutes with B_2 and B_5:

$$\begin{aligned} B_2|\psi'\rangle &= B_2 X_{25}|\psi\rangle \\ &= -X_{25}B_2|\psi\rangle \\ &= -|\psi'\rangle \end{aligned}$$

and similarly for B_5. Thus, $|\psi'\rangle$ is in the -1-eigenspace of the B_2 and B_5 stabilizers (and only these two). Thus, measuring all the stabilizers determines the location and type of error. Note that -1 measurement results always occur in pairs.



Obviously, applying an X to e_{25} will correct the error, but we don't actually know the error occurred at e_{25}. We only can detect the "endpoints" of the cochain. Consider the situation where errors X_{56}, X_{36}, X_{23} all occur. Well then B_5 and B_6 would give -1 measurements due to X_{56}. But B_6 also would give a -1 measurement from the X_{36}. These -1 measurements cancel, or equivalently, we can understand this by noting that two (or four) X errors on edges surrounding a plaquette conserve the $0 \mod 2$ sum requirement of edges provided that the edges satisfied it originally. The same cancellation occurs with B_3, and so again we only get see the endpoints of the cochain, however this time the chain consists of three edges and is represented in blue above. The point here though, is that it doesn't matter how we correct for the error with syndrome given by B_2, B_5 giving -1, so long as the correction is (co)homologically equivalent to the actual error. This is understood by noting that if the X_{56}, X_{36}, X_{23} errors occurred and we apply X to e_{25} to correct them, then

$$X_{25} \otimes X_{56} \otimes X_{36} \otimes X_{23} = A_{2356}$$

and thus stabilizes the original state.

The point about any correction that is homologically equivalent to the error is crucial. In the above picture, if X errors occured on e_{58} and e_{28}, then we see the same error syndrome, but if we correct by applying X to e_{25}, then we have produced a non-trivial cycle on the torus, and thus have produced a logical error on our inital state. This argument proves that the above 3 Toric code can correct with certainty at most errors on 1 qubit.

Let us briefly also consider the effect of a Z error. Suppose a Z error has occurred on e_{25}. This error commutes with B_2, B_5 but anticommutes with A_{1245} and A_{2356}. Thus error correction can occur via an chain in the lattice that is homologically equivalent to the error, i.e. applying Z to any chain of edges that, along with Z_{25}, corresponds to a product of B_p's. Again, we have to be careful not to apply a chain of Z operators that form a non-trivial cycle around the torus.

The points raised in the above example regarding non-trivial cycles of Z and X operators on the lattice are important for actually performing computation with the code. There are two non-trivial cycles of X operators and two non-trivial cycles of Z operators and these correspond to the logical X and Z operations that we can perform on the two logical qubits encoded in the Toric code.

Desirable Properties of the Toric Code:

Some properties of the Toric code that are useful include

- Each stabilizer operator involves at most 4 local qubits
- Each qubit is involved in at most 4 local stabilizer operators
- Provided an arbitrary large lattice can be constructed, any number of errors can be corrected
- The codespace can be written as the ground state of a (fairly) realistic Hamiltonian

Let us expand on the last point. Consider the Hamiltonian

$$H = -\sum_v A_v - \sum_p B_p$$

This is a nice Hamiltonian for a number of reasons: it is easily diagonalisable since all the A_v and B_p commute, all it's excited states are separated by an energy gap since the difference between eigenvalues for the stabilizers is 2, and the Hamiltonian involved local interactions so it may be feasible to physically realise. Futhermore, it is stable to (small enough) local perturbations (for more see Ref [2]).

References

[1] M. A. Nielsen and I. Chuang, “Quantum computation and quantum information,” 2002.

[2] A. Y. Kitaev, “Fault-tolerant quantum computation by anyons,” *Annals of Physics*, vol. 303, no. 1, pp. 2–30, 2003.