CS 593/MA 595 - Intro to Quantum Computing
Spring 2024
Tuesday, April 18 - Lecture 14.2

Today's scribe: Shahbaz [Note: not proofread by Eric]

Agenda:

1. Distance and Rate for Pauli Stabilizer Code
2. Toric Code

1 Recall

A Pauli stabilizer code is any (non-trivial) subspace $C_S \subseteq (\mathbb{C}^2)^{\otimes n}$ defined by a set of Pauli operators $S = \{g_1, \ldots, g_r \mid g_i \in G_n\}$.

We might as well take $C_S = C_{\langle S \rangle}$.

Theorem: Pauli stabilizer codes on n-qubits are in bijection with isotropic subspaces of $(\mathbb{F}_2^n \oplus \mathbb{F}_2^n, \omega)$.

2 Rate and Distance for Pauli Stabilizer Code

2.1 A Crucial Lemma

A Pauli logical operator on C_S is any $E \in G_n$ such that $E(C_S) = C_S$.

Lemma: $E \in G_n$ is a logical operator iff E commutes with every element of $\langle S \rangle$.

Thus, the set of all Pauli errors on C_S is exactly

$$E_S := Z_{G_n}(S) = \{E \in G_n \mid Eg = gE \forall g \in S\}$$

In particular, if $\pi : G_n \longrightarrow \mathbb{F}_2^n \oplus \mathbb{F}_2^n$ is the same as last class, then we have

$$\pi(Z_{G_n}(S)) = \pi(\langle S \rangle)^{\perp}$$

In fact, $Z_{G_n}(S) = \pi^{-1}(\pi(\langle S \rangle)^{\perp})$

2.2 Computing Rate

Note that if E_1, E_2 are two Pauli logical operators on C_S, then E_1, E_2 implement the same operation on C_S iff $E_1 E_2^{-1}$ is in $\langle S \rangle$ (up to phase.)

Thus, the group of non-trivial Pauli logical operators (ignoring phases $\pm 1, \pm i$) is ${}^{\pi(\langle S \rangle)^{\perp}}/_{\pi(\langle S \rangle)}$.

In other words, if G_S is the group of all non-trivial Pauli operators of C_S,

$$\{\pm 1, \pm i\} \longrightarrow G_S \longrightarrow {}^{\pi(\langle S \rangle)^{\perp}}/_{\pi(\langle S \rangle)}$$

The picture to keep in mind is:

$$\begin{array}{ccccc}
\{\pm 1, \pm i\} & \longrightarrow & G_n & \xrightarrow{\pi} & \mathbb{F}_2^n \oplus \mathbb{F}_2^n \\
\parallel & & \cup\!| & & \cup\!| \\
\{\pm 1, \pm i\} & \longrightarrow & E_S & \xrightarrow{\pi} & \pi(\langle S \rangle)^{\perp} \\
\cup\!| & & \cup\!| & & \cup\!| \\
\{\pm 1\} & \longrightarrow & \langle S \rangle & \longrightarrow & \pi(\langle S \rangle)
\end{array}$$

Fun Fact: If $W \subseteq \mathbb{F}_2^n \oplus \mathbb{F}_2^n$ is isotropic, then ${}^{W^{\perp}}/_{W}$ has a symplectic product inherited from ω.

Now, $G_S = {}^{E_S}/_{\langle S \rangle}$ and we have

$$\{\pm 1, \pm i\} \longrightarrow G_S \longrightarrow {}^{\pi(\langle S \rangle)^{\perp}}/_{\pi(\langle S \rangle)}$$

so the number of logical qubits is $\frac{1}{2}\dim({}^{\pi(\langle S \rangle)^{\perp}}/_{\pi(\langle S \rangle)})$

2.3 Computing Distance

We need to know what the smallest element of E_S acting non-trivially (ignoring phase) on C_S is.

Given a vector $(\vec{x}, \vec{z}) \in \mathbb{F}_2^n \oplus \mathbb{F}_2^n$, define the symplectic weight, denoted $wt(\vec{x}, \vec{z})$, as follows:

Let $\vec{x} = (x_1, \ldots, x_n)$ and $\vec{z} = (z_1, \ldots, z_n)$. Then, the symplectic weight is the number of columns in

$$\begin{pmatrix} x_1 & \ldots & x_n \\ z_1 & \ldots & z_n \end{pmatrix}$$

that do not contain both zeroes.

The distance of C_S is then

$$\min_{(\vec{x}, \vec{z}) \in \pi(E_S) - \pi(\langle S \rangle)} wt(\vec{x}, \vec{z})$$

This is hard to compute, but it is at least a combinatorial quantity.

2.4 Summary

A Pauli stabilizer code is determined by an isotropic subspace $W \subseteq \mathbb{F}_2^n \oplus \mathbb{F}_2^n$.

The number of logical qubits is $\frac{1}{2}\dim({}^{\pi(\langle S \rangle)^{\perp}}/_{\pi(\langle S \rangle)})$.

The distance is $\min_{(\vec{x}, \vec{z}) \in \pi(E_S) - \pi(\langle S \rangle)} wt(\vec{x}, \vec{z})$

This characterizer Pauli stabilizer codes completely classically. We still need to understand encoding and decoding circuits for these codes.

We won't do this, but the summary is: They exist. They are efficient.

Given a set of stabilizers S, $\exists$ a polynomial time classical algorithm to construct encoding and decoding circuits.

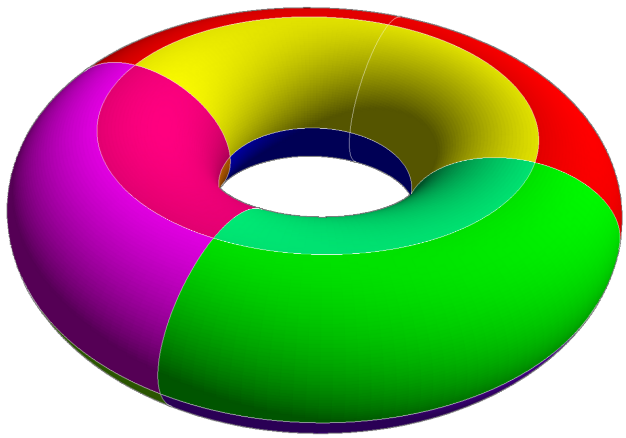
Unfortunately, in the world, these circuits are noisy.

Today, there is effort to optimize overhead in encoding/decoding circuits.

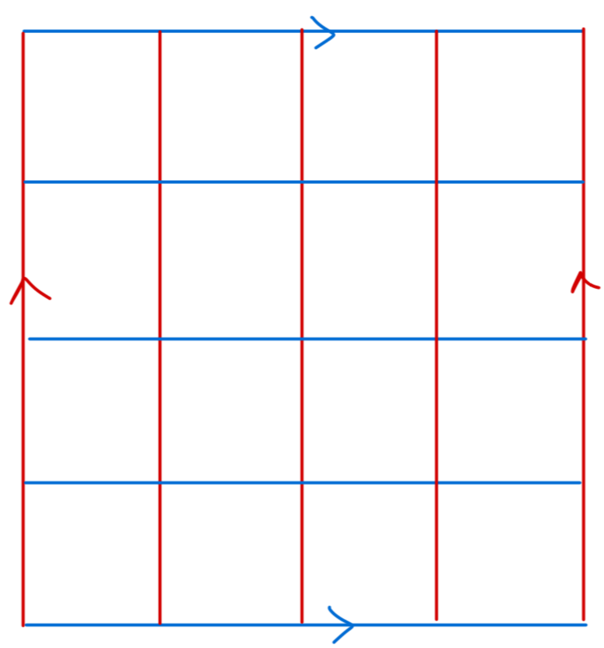
In principle, the threshold theorem says that as long as errors in gates are small enough, we can win.

3 Toric Code

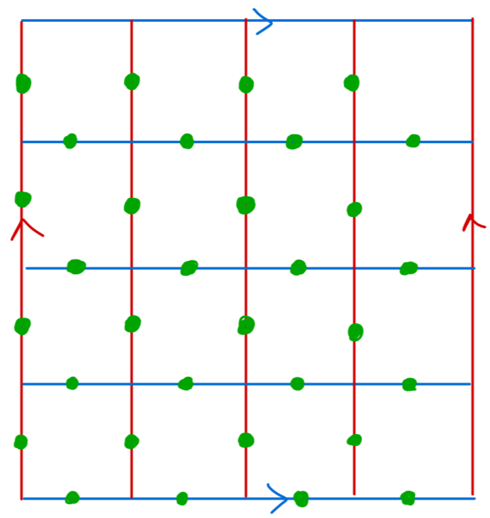
Start with a square grid on a torus.



Rather than work with this figure, we cut it open to a square with periodic boundary condition.



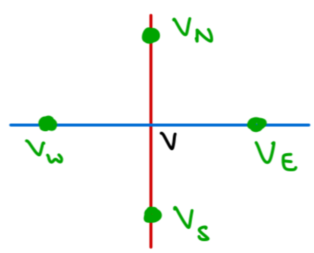
Now, put a qubit on each edge.



So, for a *nxn* grid, we have $2n^2$ qubits.

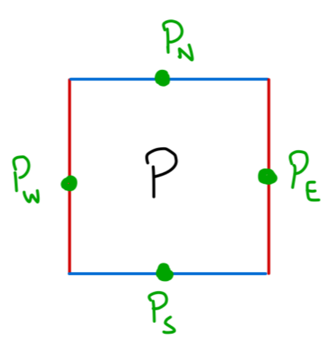
For each vertex v, and plaquette P in this grid, we define some Pauli stabilizers.

First, for a vertex v, label the 4 qubits around it as follows:



We define the vertex stabilizer, $A_v := X_{v_N} X_{v_E} X_{v_S} X_{v_W}$.

Second, for a plaquette P, label the 4 qubits around it similarly:



We define the plaquette stabilizer, $B_P := Z_{P_N} Z_{P_E} Z_{P_S} Z_{P_W}$.

The toric code on $2n^2$ qubits (indexed by a square grid) is the Pauli stabilizer code generated by all the A_v and B_P.

Theorem: Toric code is a $[[2n^2, 2, n]]$ QEC.

The moral is that this is decent code. We can get 2 logical qubits with arbitrary distance d using ”local” stabilizers.

The key to proving this is to understand logical Pauli operators ”geometrically.”