Lecture 3

QLDPC codes : The future of QEC ?

The encoding rate $\frac{k}{n}$ of a code is a measure of the qubit overhead for QEC.

For surface codes : $\frac{k}{n} \approx \frac{O(1)}{n} \xrightarrow[n\to\infty]{} 0$

Can we hope to have a topological code (i.e. local check operators) with high rate and large distance ? No :

Bravyi - Poulin - Terhal '09

For topological (geometrically local) codes with euclidean metric in D dimensions

$$k d^{2/D-1} \leq cn$$

in 2D : $k d^2 \leq cn$ (saturated by surface code)

We need long-range interactions

We want to retain low density parity checks (LDPC):

every qubit (check) only interacts with a constant number of check (qubits)

$\Rightarrow H_X, H_Z$ are sparse

Why is this property important?
Otherwise the syndrome measurement circuit would be macroscopic (and errors can spread to macroscopic number of qubits).

It's easy to show that good quantum codes exist: $k \sim n$, $d \sim n$

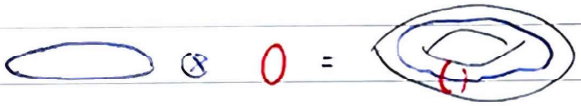
But only last year or two we've learned that good QLDPC codes exist!

But today, we'll be satisfied with a "pretty good" QLDPC code

$k \sim n$, $d \sim \sqrt{n}$

Hypergraph / tensor product codes

insight: torus = product of two circles



can we write the toric code as a product of 1D classical codes?

: parity check (Z-check) / bit (qubit)

Which code is this?

Toric code =
The product of two repetition codes!

~~Toric for~~ $[n,1,n] \otimes [n,1,n] = [[2n^2, 2, n]]$

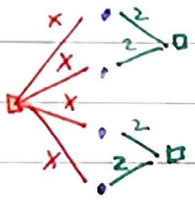
Can this be generalized to any classical code? yes!

preliminaries: Tanner graph representation of a classical code

→ classical repetition code (i.e. only bit flips) [5,1,5]

Tanner graph of a quantum code

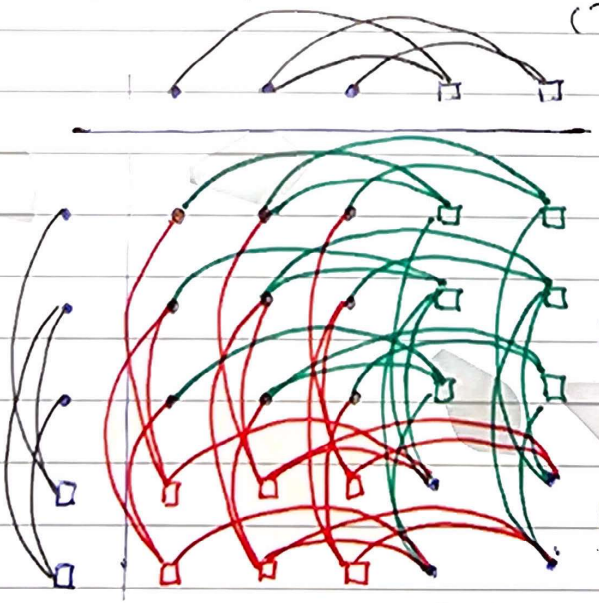
CSS construction: combining two classical codes into a quantum code. X and Z checks need to overlap on <u>even</u> nr. of qubits.



$$H = \begin{bmatrix} X & X & X & X \\ I & I & Z & Z \\ Z & Z & I & I \end{bmatrix}$$

Hypergraph product construction

(Tillich-Zémor)



Z-checks

<u>HW</u>

check that this is a surface code

X-checks

extra qubits to ensure commutation

Important: The 2 classical codes don't have to be identical. In fact, they can be any two codes (commutation is automatically satisfied)

How many qubits are there?

$$n' = n^2 + r^2 = n^2 + (n-k)^2$$

How many stabilizer generators?

$$r' = 2nr = 2n(n-k)$$

How many logical qubits?

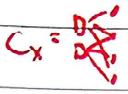
$$k' = n' - r' = k^2$$

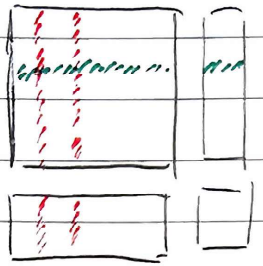
$$\frac{k'}{n'} = \frac{k^2}{n^2+(n-k)^2} \approx \frac{k^2}{n^2}$$

the encoding rate of the quantum code inherits that of the classical codes! in particular, if the classical code is good $\left(\frac{k}{n}\right) = c$, so is the quantum code: $\left(\frac{k'}{n'}\right) \approx c^2$

* by inspection, if the classical code is LDPC, so is the quantum code.

What about the distance of the new code?

C_x =  & C_z = 



applying a logical bit flip of code C_1 on a column or of C_2 on a row will result in a logical operation on the quantum code

$\Rightarrow d' = \min(d_1, d_2)$

Conclusion: if we take the product of

$[n, k, d]$ & $[n, k, d]$

we get a quantum code

$\longrightarrow [[\sim n^2, \sim k^2, d]]$

In particular, if we take a good LDPC code, we get a QLDPC code with

$[[n', \sim n', \sim\sqrt{n'}]]$

Good classical LDPC codes have been known to exist for decades

Unleash ingenuity

Sketch for constructing good classical LDPC code

To get a good encoding rate, we can choose a code with $r = 3/4\, n$ checks.
$\Rightarrow k = n - r = 0.25\, n$.

Now let's take care of the distance. Suppose the two closest codewords are z_a and z_b, $d = |z_a - z_b|$. But $z_a - z_b$ is also a codeword z' and so the distance is the smallest weight of any codeword $d = \min_{z \in C} |z|$

→ We want a code with $\min_{z \in C} |z| \propto n$

Let's use a Δ-regular graph (each bit sees Δ checks) that is expanding. This works for a random graph

Expanding graph: Any small enough set of bits, e.g. $|z| = 0.01\, n$ has close to $\Delta |z|$ neighboring checks.

for example, with $d = 64$, small enough z has $\geq 0.8 \times 64 \times 121$ neighboring checks.

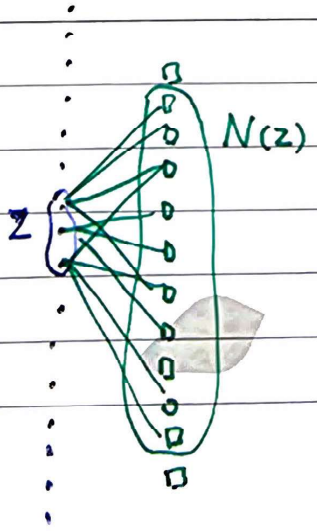
This expander code spreads out so much that there must be at least one neighboring check that sees only 1 edge

→ the parity check is violated

→ z is not a code word

→ $d = \min_{z \in C} |z| \geq 0.001$

So a random expander graph has $k \propto n$ and $d \propto n$.



question: can we not just stick two expanders together using CSS construction? No!

$H_x H_z^T \neq 0$