Random classical LDPC code can be good: $k, d \sim n$

↳ ~~quantum?~~ Can we try to replicate our success?

NO! need to enforce $H_X H_Z^T = 0$

It's hard to enforce all of these constraints using a random construction.

Still we would like to engineer quantum codes w/ better k,d scaling. To understand how to do that, need to return to (co)homology...

Classical code → homology:

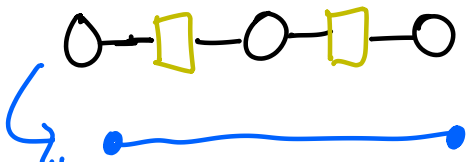
pad chain complex w/ trival spaces if desired.

$$0 \xrightarrow{0} \underbrace{\mathbb{F}_2^n}_{\text{(bits)}} \xrightarrow{H} \underbrace{\mathbb{F}_2^m}_{\text{(checks)}} \xrightarrow{0} 0$$

⇒ homology group: $H_1 = \frac{\ker(H)}{\mathrm{im}(0^T)} = \ker(H)$ ⇒ codespace

example: 1d repetition code:

$$H = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix} \longrightarrow$$



"discretization" of 1d line interval.

Now often we can build higher dimensional objects out of lower dimensional objects in math by taking "products" of spaces...

2d region $0 \le x, y \le 1$ = $0 \le x \le 1$ × $0 \le y \le 1$

Importantly, this product plays very nicely with (co)homology...

$$\begin{pmatrix}\text{vertex}\\ \text{y-edge}\end{pmatrix} \times (\text{vertex, x-edge}) = \begin{pmatrix}\text{vertex} & \text{x-edge}\\ \text{y-edge} & \text{face}\end{pmatrix}$$

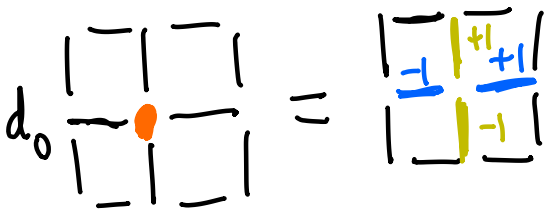
The 2d space has a nice cohomology...:

$$\text{face} \xleftarrow{d_1} (\text{x-edge})\oplus(\text{y-edge}) \xleftarrow{d_0} \text{vertex}$$

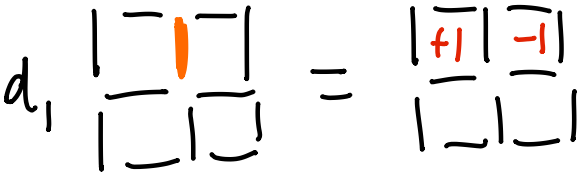
Key point: given $d_0^{(y)}$ & $d_0^{(x)}$ from starting cohomology, construct d_0 & d_1 for larger cochain complex.

Easy to visualize what needs to happen.

gradient d_0:



curl d_1:



$$\hookrightarrow d_0\left[|x\rangle\otimes|y\rangle\right] = d_0^{(x)}|x\rangle\otimes|y\rangle + |x\rangle\otimes d_0^{(y)}|y\rangle$$

$$\Rightarrow d_0 = \begin{pmatrix} d_0^{(x)}\otimes I \\ I\otimes d_0^{(y)}\end{pmatrix} \begin{matrix}(\text{x-edges})\\ (\text{y-edges})\end{matrix}$$

$$d_1\left[|e_x\rangle\otimes|y\rangle + |x'\rangle\otimes|e_y'\rangle\right] = |e_x\rangle\otimes d_0^{(y)}|y\rangle \ominus d_0^{(x)}|x'\rangle\otimes|e_y'\rangle$$

important! comes from $dx\wedge dy = -dy\wedge dx$

$$d_1 = \begin{pmatrix} I\otimes d_0^{(y)} & -d_0^{(x)}\otimes I\end{pmatrix}$$

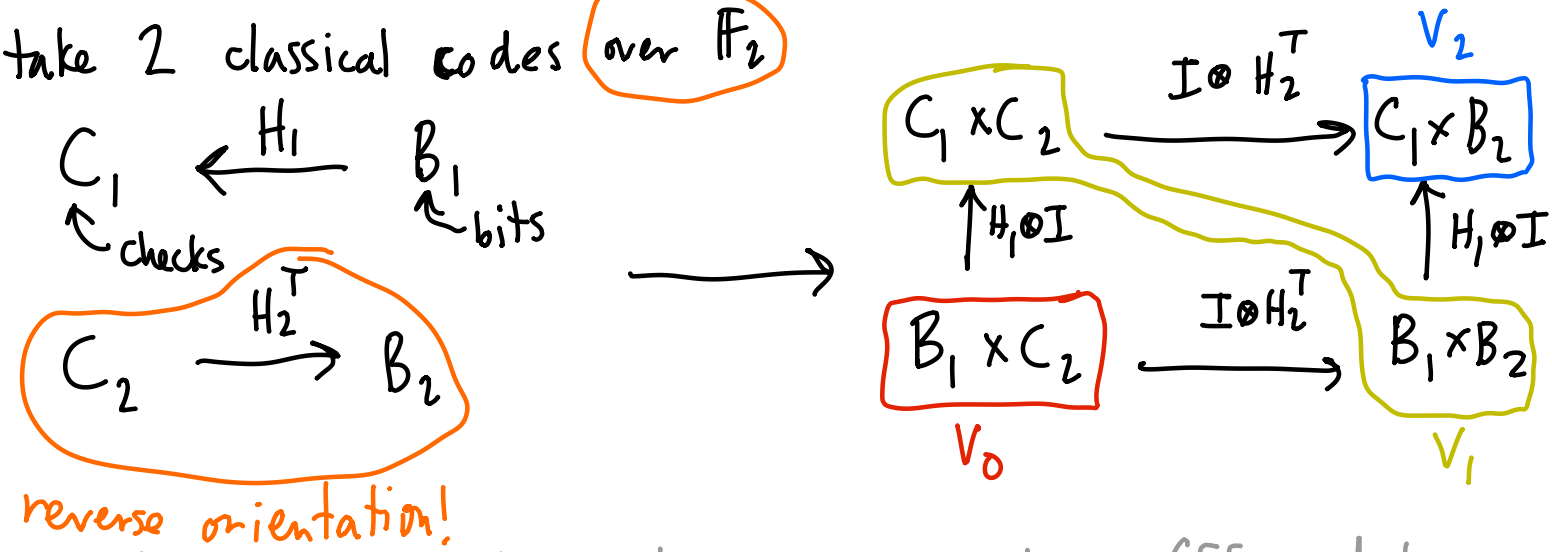
$$\Rightarrow d_1 d_0 = d_0^{(x)}\otimes d_0^{(y)} - d_0^{(x)}\otimes d_0^{(y)} = 0$$

Therefore the cohomology of higher chain complex works.

So far we've had a geometric picture here but we can abstract it away...

For reasons that become clear soon we'll make the following conventions...

take 2 classical codes over $\mathbb{F}_2$ ($-1=1$)



Now interpret this chain complex as a quantum CSS code!

Def: Hypergraph product code:

$$V_0 \xrightarrow{H_X^T} V_1 \xrightarrow{H_Z} V_2$$

(V_0: X-checks; V_1: qubits; V_2: Z-checks)

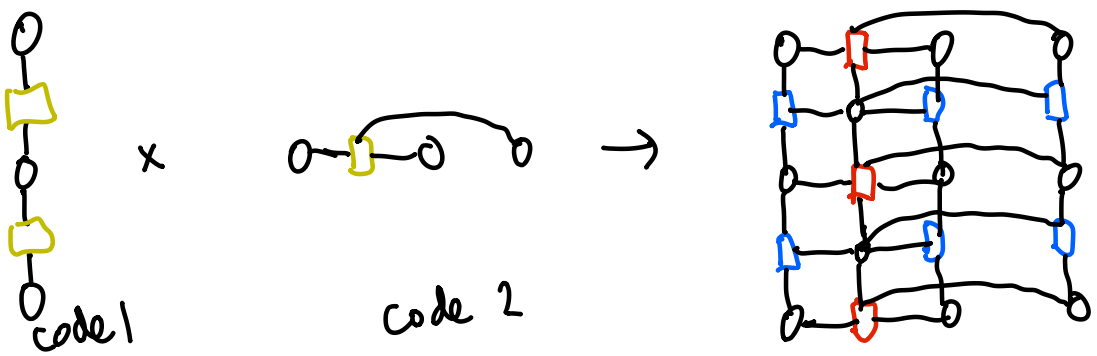
where $H_X = \left(I \otimes H_2 \mid H_1^T \otimes I \right)$ $\quad H_Z = \left(H_1 \otimes I \mid I \otimes H_2^T \right)$

$I \otimes H_2$: $B_1 \times B_2$ "bit-bit"; $H_1^T \otimes I$: $C_1 \times C_2$ "check-check"

CSS condition: $H_X H_Z^T = (I \otimes H_2)(H_1^T \otimes I) + (H_1^T \otimes I)(I \otimes H_2)$

$= 0$ so this is a valid quantum code.

It's useful to think about this with Tanner graphs...



Notice that this looks a bit like 2d code w/ an extra "line" stuck onto it (like 3 planes meeting). Not an accident... HGP of 1d rep code → 2d surface code!

Now time to ask, what are properties of HGP codes?

Prop: HGP code remains LDPC: if classical codes had (Δ_B^c, Δ_C^c) quantum code has $(\Delta_B^q \le \max(\delta_B^c, \Delta_C^c),\ \Delta_C^q \le \Delta_B^c + \Delta_C^c)$

↑ for X & Z checks individually

Proof follows immediately from row/column weights in H_X / H_Z.

Thm: Suppose for simplicity that $\ker(H_1^T) = \ker(H_2^T) = 0$ (no redundant checks)

Then given classical $[n_1, k_1, d_1]$ and $[n_2, k_2, d_2]$ codes, HGP code has $[[n_1 n_2 + (n_1 - k_1)(n_2 - k_2),\ k_1 k_2,\ \min(d_1, d_2)]]$.

Proof: let's figure out X-logicals/cohomology.

$$\ker(H_Z) = \ker\left(\left(H_1 \otimes I \mid I \otimes H_2^T\right)\right)$$

$$= \left\{ \begin{pmatrix} b \\ c \end{pmatrix} \text{ (bit-bit)} \ :\ (H_1 \otimes I) b = (I \otimes H_2^T) c \right\}$$

↑ check-check qubit

$\ker(H_1^T) = 0 \Rightarrow$ all vectors in $\mathbb{F}_2^{V_2} \cong \mathbb{F}_2^{C_1 \times B_2}$ take form $(H_1 \otimes I)b$ for some b. Expand $c = \sum c_1^a \otimes c_2^a \neq 0$.

$$\sum_{a,\alpha} c_1^a \otimes H_2^T c_2^\alpha = \sum_{a} (H_1 b_1^a) \otimes b_2^\alpha \quad (b_2^\alpha = H_2^T c_2^\alpha)$$

So for any c, there exist b such that

$$\begin{pmatrix} b \\ c \end{pmatrix} = \begin{pmatrix} I \otimes H_2^T \\ H_1 \otimes I \end{pmatrix} \sum_{a,\alpha} b_1^a \otimes c_2^\alpha \in \mathrm{im}(H_X^T).$$

What if $b \neq 0$ but $c = 0$? states $\sum x^a \otimes H_2^T c_2$ are in $\mathrm{im}(H_X^T)$.
($x \in \ker(H_1)$)

Take $x \otimes y \notin \mathrm{im}(H_2^T)$. ($x \in \ker(H_1)$) Then $x \otimes y \in \ker(H_1 \otimes I)$ is a non-trivial logical!

Lemma: If $k_1, k_2 > 0$, $\exists$ weight 1 vector $\notin \mathrm{im}(H_2^T)$.

Proof: If false, $\mathrm{im}(H_2^T) = \mathrm{span}(\text{weight-1 vectors in } \mathbb{F}_2^{n_2}) = \mathbb{F}_2^{n_2}$
but $\mathrm{im}(H_2^T) \cong \mathbb{F}_2^{n_2 - k_2}$ which is a contradiction ▨

$\Rightarrow d_X \leq \min |x \otimes y| = d_1 \times 1 = d_1$ → Lemma: $d_X \geq d_1$ (proof elided)
$\Rightarrow d_X = d_1$

$k = \#$ of logical generators $= k_1 \cdot k_2$

k_1: # of independent x

k_2: # of independent y up to $\mathrm{im}(H_2^T)$

Repeating same arguments reading chain complex other way $\Rightarrow d_Z = d_2$ ▨

Most interesting application of HGP codes is to take random products of good classical codes! (Δ_B odd $\Rightarrow$ no redundant check)

Cor: HGP code out of 2 random classical good LDPC

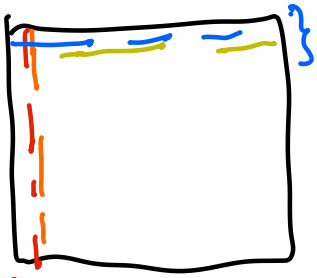
$\Rightarrow$ quantum HGP code w/ $[[n_c^2, \Theta(n_c^2), \Theta(n_c)]]$

equivalently, $[[n, \Theta(n), \Theta(\sqrt{n})]]$

constant rate codes! growing but sublinear distance.

For most practical purposes this could be good enough as LDPC condition along w/ $d \sim \sqrt{n}$ ensures individual logicals have FT memory.

What do logicals look like in a constant rate HGP code?



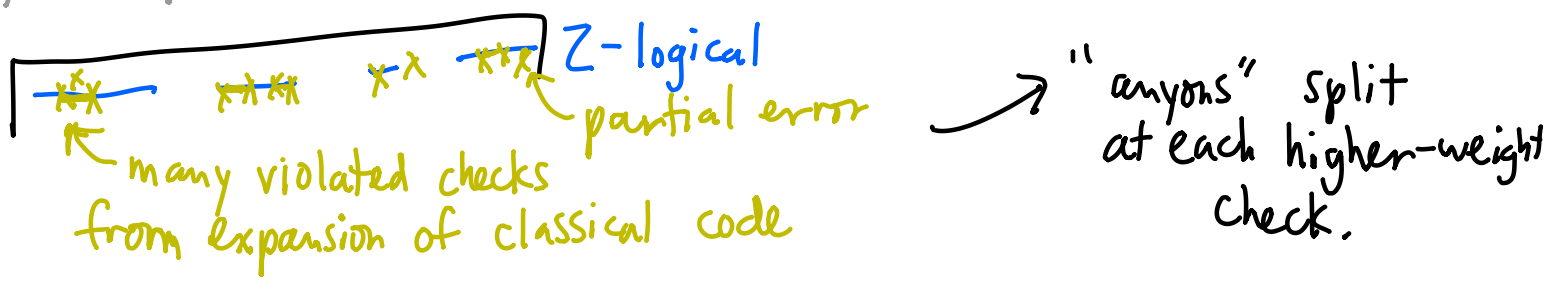
You might worry that "string like" logical ⇒ bad stability at finite T? But actually this is not true...

Thm: for sufficiently large Δ_B, HGP code has <u>linear confinement</u>:

$$|H_X e_Z| \geq \eta |e_Z|_{red} \quad \text{if } |e_Z|_{red} < \alpha d \sim \sqrt{n}$$

(η: $O(1)$)

Proof is tedious due to reduced weight, and is omitted.
Physical picture is as follows:



From a physics perspective HGP codes are much like random classical counterparts...

Prop: no redundant checks in HGP code

Proof: $n = n_1 n_2 + (n_1 - k_1)(n_2 - k_2) = \underbrace{k_1 k_2}_{k} + \underbrace{(n_1 - k_1) n_2}_{\text{rank}(H_2)} + \underbrace{n_1 (n_2 - k_2)}_{\text{rank}(H_X)}$

Cor: Partition function is trivial: $Z(\beta) = 2^k \left[1 + e^{-\beta}\right]^{n-k}$

And analogously to classical codes we also have:

Thm: Gibbs sampler protects logical information for time $\exp[O(d)]$ at sufficiently large β
as a consequence of LDPC + linear confinement.