Let's begin by quickly reviewing last time.

Consider group algebra $R = \mathbb{F}_2[G]$ (elements are $\sum c_g g$; interpret as $|G| \times |G|$ matrix)

Left action: $L_h \cdot g = hg$ Right action: $R_h \cdot g = gh^{-1}$

↳ commute: $L_h R_{h'} = R_{h'} L_h$

2BGA codes from group algebra perspective:

$$H_X = (L_a \quad R_b) \qquad H_Z = (R_b^\dagger \quad L_a^\dagger)$$

$b = \sum b_g g$ $\qquad R_b^\dagger := \sum b_g R_{g^{-1}} := R_{b^*}$

Here is some additional motivation for why we wanted to study codes based on group algebras...

Recall HGP: $[n_{cl}, \alpha \cdot n_{cl}, \beta \cdot n_{cl}] \times [n_{cl}, \alpha n_{cl}, \beta n_{cl}] \to [[\gamma n_{cl}^2, (\alpha n_{cl})^2, \beta n_{cl}]]$

$\frac{k}{n}$ stays $O(1)$ but $\frac{d}{n} \to \frac{1}{n_{cl}}$

Reason: qubits indexed by (v_1, v_2); logicals exist at fixed v_2.

bit/check from classical code

checks connect $(v_1, v_2) \leftrightarrow (v_1', v_2)$ and $(v_1, v_2) \leftrightarrow (v_1, v_2')$

Now suppose v_1 indexed by group G:

$$L_h(g_1, g_2) = (hg_1, g_2) \qquad R_h(g_1, g_2) = (g_1, g_2 h^{-1})$$

So L/R actions can be reinterpreted as acting on L/R copies...

Group algebra is a <u>quotient</u>: $G \times G \to G$

$$(g_1, g_2) \sim (g_1 h, h^{-1} g_2) \sim (1, g_1 g_2)$$

Idea: could quotient make $n, k, d \sim n_{cl}$?

Unfortunately not that simple. A simple example of what can go wrong:

Recall: 2d TC is HGP of 1d repetition code.

represent as classical group algebra code on $\mathbb{F}_2[\mathbb{Z}_L]$: $H = 1 + x$ ($\{1, x, \dots, x^{L-1}\}$)

literal HGP: group $G = \mathbb{Z}_L \times \mathbb{Z}_L$

$$H_X = \begin{pmatrix} I \otimes (1+x) & (1+x^{-1}) \otimes I \end{pmatrix}$$
$$H_Z = \begin{pmatrix} (1+x) \otimes I & I \otimes (1+x^{-1}) \end{pmatrix}$$

} 2d TC

quotiented HGP: group $G = \mathbb{Z}_L$

$$H_X = H_Z = \begin{pmatrix} 1+x & 1+x^{-1} \end{pmatrix}$$

$\Rightarrow d \leq 2$ as showed previously

So the quotient takes $[[2L^2, 2, L]] \longrightarrow [[2L, 2, 2]]$

Can we add some structure to prevent this from happening? Idea is to replace 2BGA code w/ a larger matrix of $\mathbb{F}_2[G]$-valued entries.

Def: Lifted product (LP) code: Choose

$$A \in R^{m_1 \times l_1} \qquad B \in R^{m_2 \times l_2}$$

[2BGA: $m_{1,2} = l_{1,2} = 1$]

Code has $n = |G| \cdot (m_1 m_2 + l_1 l_2)$ qubits

$V \in R^{m_1 \times m_2}$ $\quad$ $U \in R^{l_1 \times l_2}$

Using stabilizer formalism we abstractly define:

$$H_X(U_Z, V_Z) := AU_Z + V_Z B \in R^{m_1 \times l_2}$$
$$H_Z(U_X, V_X) := A^\dagger V_X + U_X B^\dagger \in R^{l_1 \times m_2}$$

If we expand out R as $|G| \times |G|$ $\mathbb{F}_2$ matrices, these are ordinary CSS codes.

Calculate: $\langle S_2, H_2(U_x, V_x)\rangle = \langle AS_2, V_x\rangle + \langle S_2B, U_x\rangle$

usual inner product: $\langle A,B\rangle := \sum_{g\in G}\sum_{ij}(A_{ij})_g(B_{ij})_g =$

$\Rightarrow (H_X H_2^T)(S_2) = H_X(S_2B, AS_2) = 2AS_2B = 0$

So the LP code is indeed a CSS code!

Prop: LP code has $k \geq |G|(\ell_1-m_1)(\ell_2-m_2)$

Proof: $k \geq n - \text{rank}(H_X) - \text{rank}(H_Z)$ [bound rank by # of X & Z checks]

$\geq |G|\left[(\ell_1\ell_2+m_1m_2) - m_1\ell_2 - \ell_1m_2\right]$ ▨

$\frac{k}{n} \geq \frac{(\ell_1-m_1)(\ell_2-m_2)}{\ell_1\ell_2+m_1m_2} \Rightarrow$ constant rate code if $\ell_1 > [1+O(1)]m_1$, $\ell_2 > [1+O(1)]m_2$.

LP codes can vastly improve over 2BGA or BB codes in near-term ranges of interest.

Recent LP codes: $\ell_{1,2}=2$ $m_{1,2}=1$, $k/n \geq 1/5$

$[[240, 48, 12]]$ using non-Abelian G

Only basic examples of transversal gates etc. Codes usually similar in terms of pseudo-threshold to BB codes. Active research area.

Thm: ∃ **LDPC** LP codes w/ $[[n, \Theta(n), \Theta(n)]]$

Proof of this result is extremely non-trivial but we can give a sense for why it is true.

Key idea: ① take $\ell_{1,2}, m_{1,2}$ finite
② take family of growing G w/ useful properties.

We'll focus on ② as this is actually somewhat more approachable...

Choose $G = PSL(2, \mathbb{F}_p)$ → replace 2 w/ large prime p

Def: projective special linear group:

$$SL(2, \mathbb{F}_p) = \left\{ \begin{pmatrix} a & b \\ c & d \end{pmatrix} : a, b, c, d \in \mathbb{F}_p, \quad ad - bc = 1 \right\}$$

$$PSL(2, \mathbb{F}_p) = SL(2, \mathbb{F}_p) / \{I, -I\}$$

To argue why this is a good choice of G, let's establish some facts about G:

$|G| = \frac{1}{2}p(p^2-1) \Rightarrow$ code's $n \to \infty$ as we take $p \to \infty$.

Crucially, this group looks locally free:

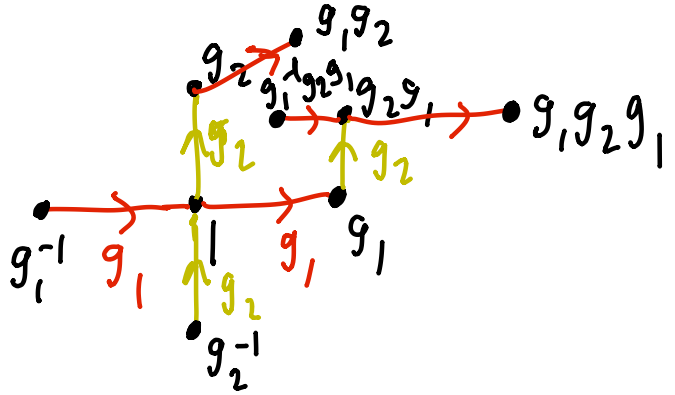
Lemma: large $p \Rightarrow$ we can find generating sets $\langle \underbrace{g_1^{\pm 1}, \dots, g_q^{\pm 1}}_{\text{set } M} \rangle = G$ such that if $s \le O(\log p)$ and

$$g_{i_1} g_{i_2} \cdots g_{i_s} = 1 \Rightarrow \text{word is trivial:}$$

$$g_2 g_1 g_1^{-1} g_2^{-1} g_2^{-1} g_2$$

Examples: $g_1 = \begin{pmatrix} 1 & 2 \\ 0 & 1 \end{pmatrix}$ $g_2 = \begin{pmatrix} 1 & 0 \\ 2 & 1 \end{pmatrix}$ → need $(1+2)^s \gtrsim p$ for non-trivial identity

Resulting Cayley graph is locally tree-like and expanding:



If we were only looking for a classical code, we could indeed build interesting codes out of such groups already!

Now consider a LP code where matrices

$$A = \begin{pmatrix} H_1 \\ H_1 \tilde{A} \end{pmatrix} \qquad B = \begin{pmatrix} H_2 \\ H_2 \tilde{B} \end{pmatrix}$$

where H_1, H_2 are check matrices of a finite classical code (details elided)

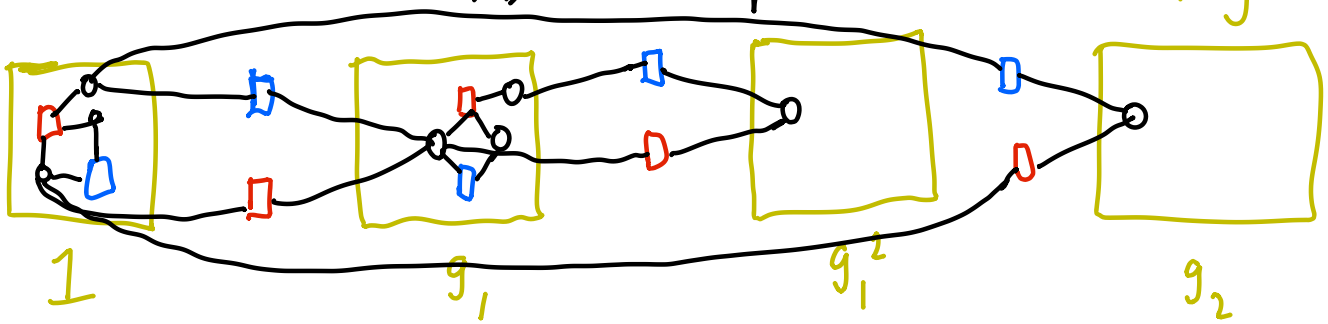
↳ interpret in R as $H_1 = H_1 \cdot 1$ (H_1: $\mathbb{F}_2$ coeff. matrix; 1: group identity)

$i = 1,2$

$H_i \in \mathbb{F}_2^{r_i \times w_i} \Rightarrow w_i = \ell_i, \; 2r_i = m_i$

and $\tilde{A} = \begin{pmatrix} g_1 & & \\ & g_2 & \\ & & \ddots & \\ & & & g_{\ell_1} \end{pmatrix} \qquad \tilde{B} = \begin{pmatrix} g_1' & & \\ & \ddots & \\ & & g_{\ell_2}' \end{pmatrix}$

H_1 expands locally... $\tilde{A}/\tilde{B}$ cause expansion btwn different g



Tanner graph not tree-like: small loops from CSS condition.

Key lemma: every reduced weight error w/ $H_X e_Z = 0$ has support on $O(1)$ fraction of g } requires special H_1, H_2

$|G| \sim n \Rightarrow d_Z \sim n$ (ditto for X)

Choose $\ell_{1,2} > m_{1,2}$ finite so $H_{1,2}$ exist $\Rightarrow$ asymptotically good quantum LDPC code.

These good quantum codes are mostly a theoretical curiosity so far — no clear approach for implementing in expt. But LP codes used to make them are useful for "near-term" high-rate codes!

Families of good quantum codes have also been used to affirm (or disprove) a number of conjectures:

① 3rd law of thermodynamics: $H_0 + \varepsilon V$ has, for small enough ε, finite-rate code

$$\lim_{T\to 0} \lim_{n\to\infty} \frac{S(T)}{n} > 0$$

② NLTS: no low-energy trivial states:

if $\langle\psi|H_0|\psi\rangle \leq \varepsilon \cdot n$,

H_0: good code w/ linear confinement

$|\psi\rangle$ not preparable w/ finite depth circuit