Lecture 21. Encoding Q-convolutional codes

21.1 Given a polynomial stabilizer matrix

$$S(D) = (\ X(D) \mid Z(D)\ )$$

transform it into the form

$$S_0(D) = [\ 0 \mid I\ 0\ ]$$

Then: encoding of the code with stabilizer $S_0(D)$ is trivial, insert $n-k$ qubits (qudits) in the state $|0\rangle$ after every block of k inputs.

21.2 Local transformations:

apply the local Clifford gate on qubit i in every block of size n

$$H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \Rightarrow \hat{H} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \in \mathbb{F}_2^{2\times 2}$$

$$P = \begin{pmatrix} 1 & \\ & i \end{pmatrix} \Rightarrow \hat{P} = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} \in \mathbb{F}_2^{2\times 2}$$

These gates preserve the weight of the stabilizers.

21.3 Two-qubit gates

$CNOT^{(i,j)}$: action on a pair of positions i, j

- add X from control to target
- add Z from target to control

$$\widetilde{CNOT}^{(i,j)} = \underbrace{\left[\begin{array}{cc|cc} 1 & 1 & & \\ & 1 & & \\ \hline & & 1 & \\ & & 1 & 1 \end{array}\right]}_{X \;|\; Z} \begin{array}{l} \\ \\ i \\ j \end{array}$$

(columns labelled i, j) within one block

CNOT spanning more than one block:

$i, \quad j + l \cdot n \qquad i \not\equiv j \mod n$

$\Rightarrow$ $i \equiv j \mod n$ results in an infinite cascade of CNOTs

$$\widetilde{CNOT}^{(i,j+ln)} = \left[\begin{array}{cc|cc} 1 & D^{l} & & \\ & 1 & & \\ \hline & & 1 & \\ & & D^{-l} & 1 \end{array}\right]$$

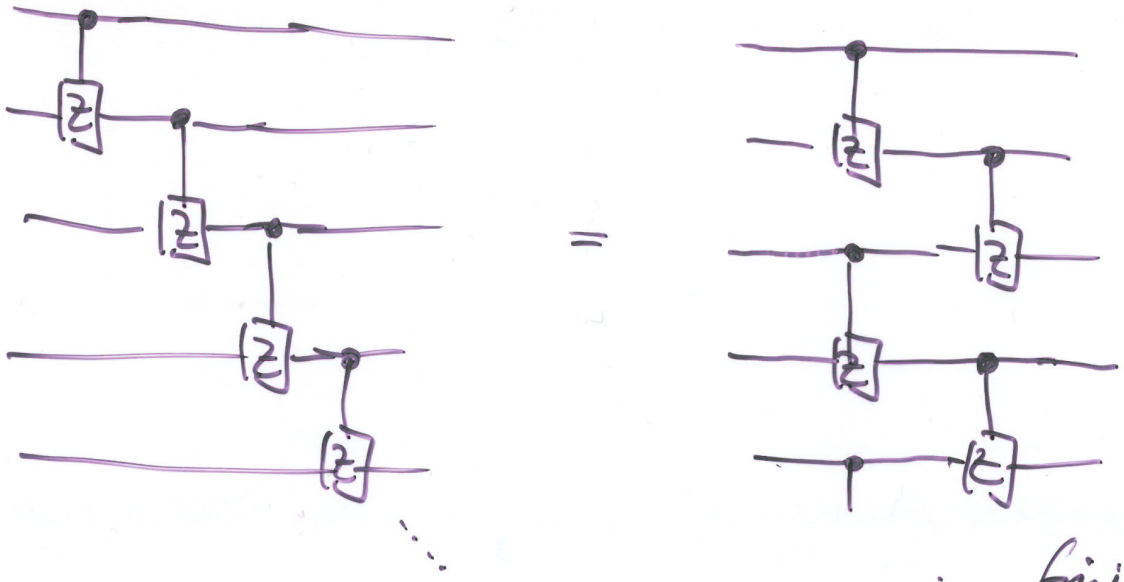
$1 \le i, j \le n$

$\Rightarrow$ We may negative powers of D in the transformed stabilizer (Laurent polynomials)

Multiply by some power of D at the end (shift the stabilizers)

21.4 Controlled phase gates

there is no no. 3



: finite depth circuit!

$$\widetilde{CSIGN}(i,j\neq i\bmod n) = \left[\begin{array}{cc|cc} 1 & & 0 & D^{\ell} \\ & 1 & D^{-\ell} & 0 \\ \hline & & 1 & \\ & & & 1 \end{array}\right]$$

for $i \neq j \bmod n$

different qubits in the blocks

$$\widetilde{CSIGN}(i,i+\ell n) = \left[\begin{array}{c:c} 1 & D^{-\ell}+D^{\ell} \\ \hdashline 0 & 1 \end{array}\right] = \tilde{P}_{\ell}$$

X Z

$\ell \neq n$ operating on the same qubit in different blocks

$$\tilde{P} = \tilde{P}_0 = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$$

21.5 Trapezoid-Lemma

Use local gates H, P as well as the pseudo-local gate P_e to transform a stabilizer of the form

$$(f(D) \mid g(D)) \doteq s$$

acting on qubit i in several blocks into

$$(h(D) \mid 0)$$

where $h(D) \in \mathbb{F}_2[D, D^{-1}]$ (Laurent polyn.)

the stabilizer s is a tensor product of some Pauli matrices commuting with all its shifted versions

$$s = \ldots I B Y Z \ldots A I \ldots$$

$$s' = \ldots I B Y Z \ldots A I \ldots$$

As s and s' commute, A and B must be the same Pauli matrix, w.l.o.g $A = B = Z$.

Now $f(D)$ and $g(D)$ are of the following form

$$f(D) \qquad D^{\nu_0} + \dots + D^{\nu_0}$$

$$g(D) = \underline{D^{\mu_0} + \dots} + g_{\nu_0} D^{\nu_0} + \dots + g_{\nu_1} D^{\nu_1} \underline{+ \dots + D^{\mu_1}}$$

$\widetilde{P}_\ell$ with $\ell = \min(\underline{\nu_0 - \mu_0}, \underline{\mu_1 - \nu_0})$

$$g'(D) = (D^{-\ell} + D^{\ell}) \cdot f(D) + g(D)$$

$\Rightarrow$ The degree of $g'(D)$, i.e the difference $\mu_1' - \mu_0'$, is strictly smaller than that of $g(D)$ given by $\mu_1 - \mu_0$.

(Swap f and g, as $\widetilde{P}_\ell$ changes the $\mathbb{Z}$-part)
if degree(g) < degree(f)

Repeat this step, until the new polynomial is zero, possibly using $\widetilde{H}$ to swap f and g

$\Rightarrow$ At the end, we have an X- only stabilizer, but not necessarily of weight one!

Sketch of the general algorithm:

(i) Use CNOT gates between different pairs to reduce the degrees in the X-part of the first row, actually: we can achieve a form

$$\left(f_1(D) \;\; 0 \;\ldots\; 0 \;\middle|\; g_1(D) \;\; g_2(D) \ldots g_n(D) \right)$$

trapezoid lemma

$$\left(f_1'(D) \;\; 0 \;\ldots\; 0 \;\middle|\; 0 \;\; g_2(D) \ldots g_n(D) \right)$$

(ii) Use Hadamard to obtain

$$\left(f_1'(D) \;\; g_2(D) \ldots g_n(D) \;\middle|\; \cancel{g_1(D)}\,0 \;\; 0 \ldots 0 \right)$$

(iii) repeat step (i) to obtain

$$\left(f_1''(D) \;\; 0 \ldots 0 \;\middle|\; \cancel{g_1'(D)}\,0 \;\; 0 \ldots 0 \right)$$

(iv) use the trapezoid lemma to obtain

$$\left(f_1'(D) \;\; 0 \ldots 0 \;\middle|\; 0 \ldots 0 \right)$$