SC-QLDPC Codes.

① Cycle Optimization

Rigid Cycles.

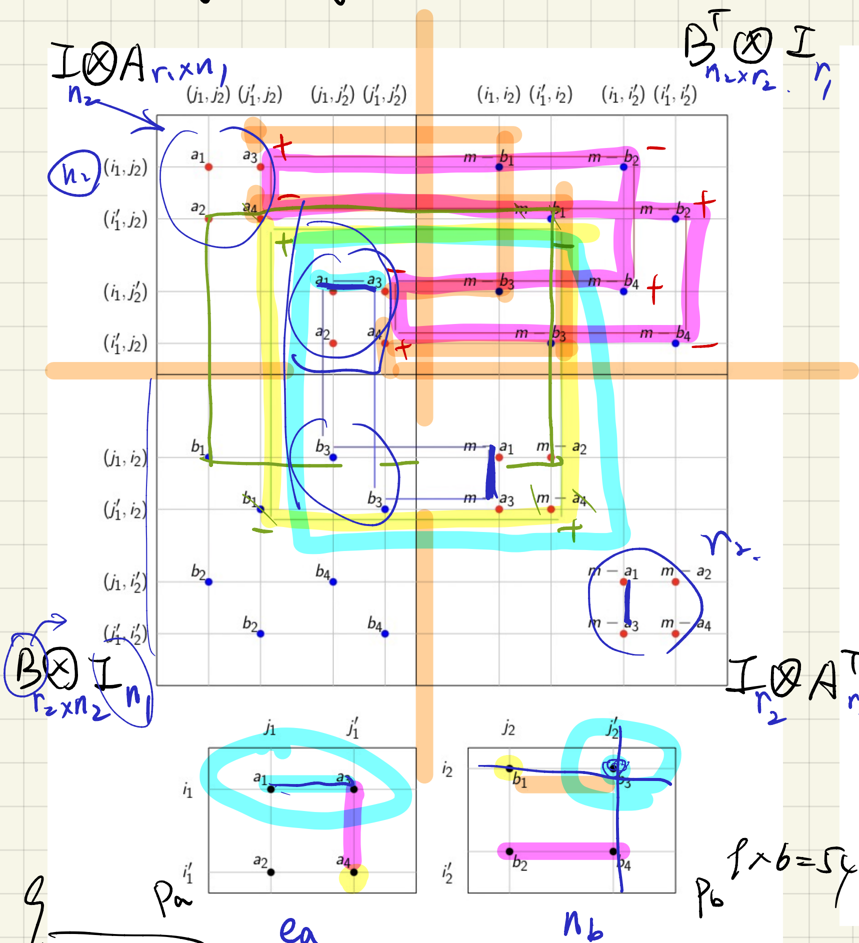


Figure: Rigid cycles.

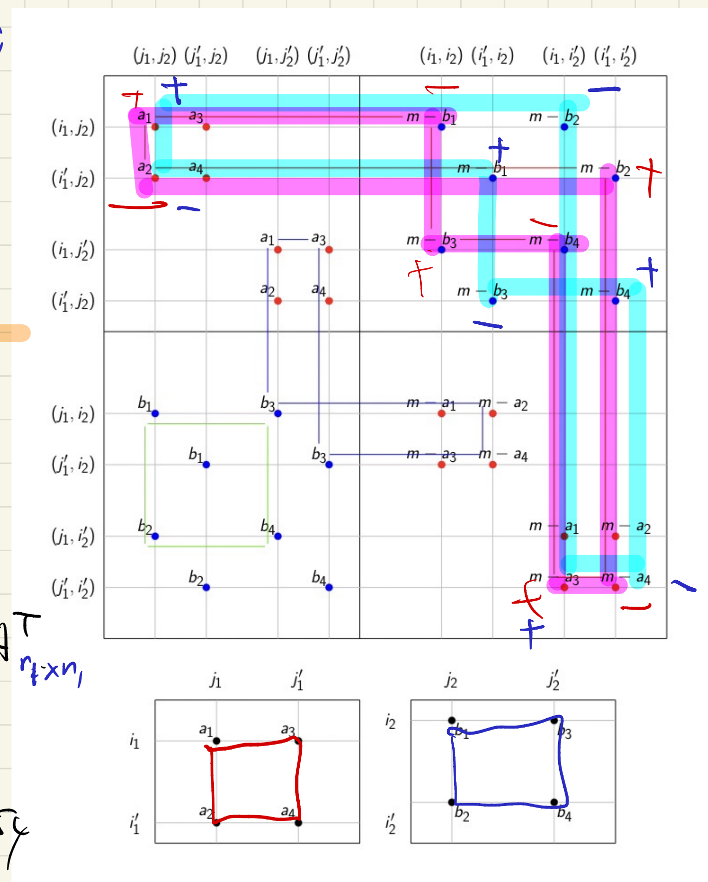
$\cancel{a_3} - (\cancel{m}-\cancel{b_2}) + (\cancel{m}-\cancel{b_4}) - \cancel{a_3} + \cancel{a_4} - (\cancel{m}-\cancel{b_4}) + (\cancel{m}-\cancel{b_2}) - \cancel{a_4} = 0$

$\cancel{a_1} - \cancel{a_3} + \cancel{b_3} - (\cancel{m}-\cancel{a_3}) + (\cancel{m}-\cancel{a_1}) - \cancel{b_3} = 0$

$\cancel{a_4} - (\cancel{m}-\cancel{b_1}) + (\cancel{m}-\cancel{a_4}) - \cancel{b_1} = 0$

Independent of a_i, b_i.

Flexible Cycles.



$a_1 - (\cancel{m}-b_1) + (\cancel{m}-b_3) - (\cancel{m}-b_4) + (\cancel{m}-a_3) - (\cancel{m}-a_4) + (\cancel{m}-b_2) - a_2$

$= (a_1 - a_2 + a_4 - a_3) + (b_1 - b_2 + b_4 - b_3)$

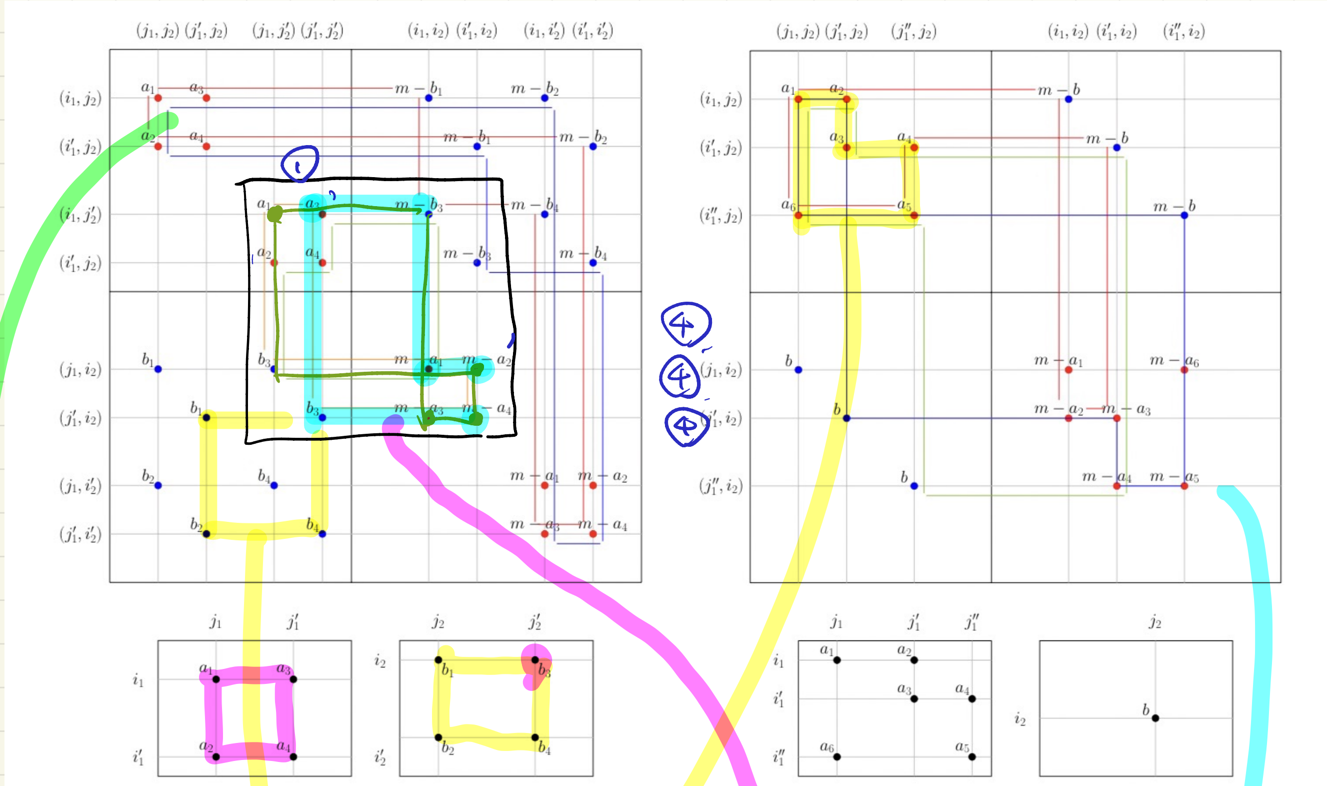
$a_1 - (m-b_2) + (m-a_3) - (m-a_4) + (m-b_4) - (m-b_3) + (m-b_1) - a_2$

$= (a_1 - a_2 + a_4 - a_3) - (b_1 - b_2 + b_4 - b_3)$

If $S = |a_1 - a_2 + a_4 - a_3| = |b_1 - b_2 + b_4 - b_3|$

$S = 0 \Rightarrow$ 2 cycles-8

$S \neq 0 \Rightarrow$ 1 cycle-8.



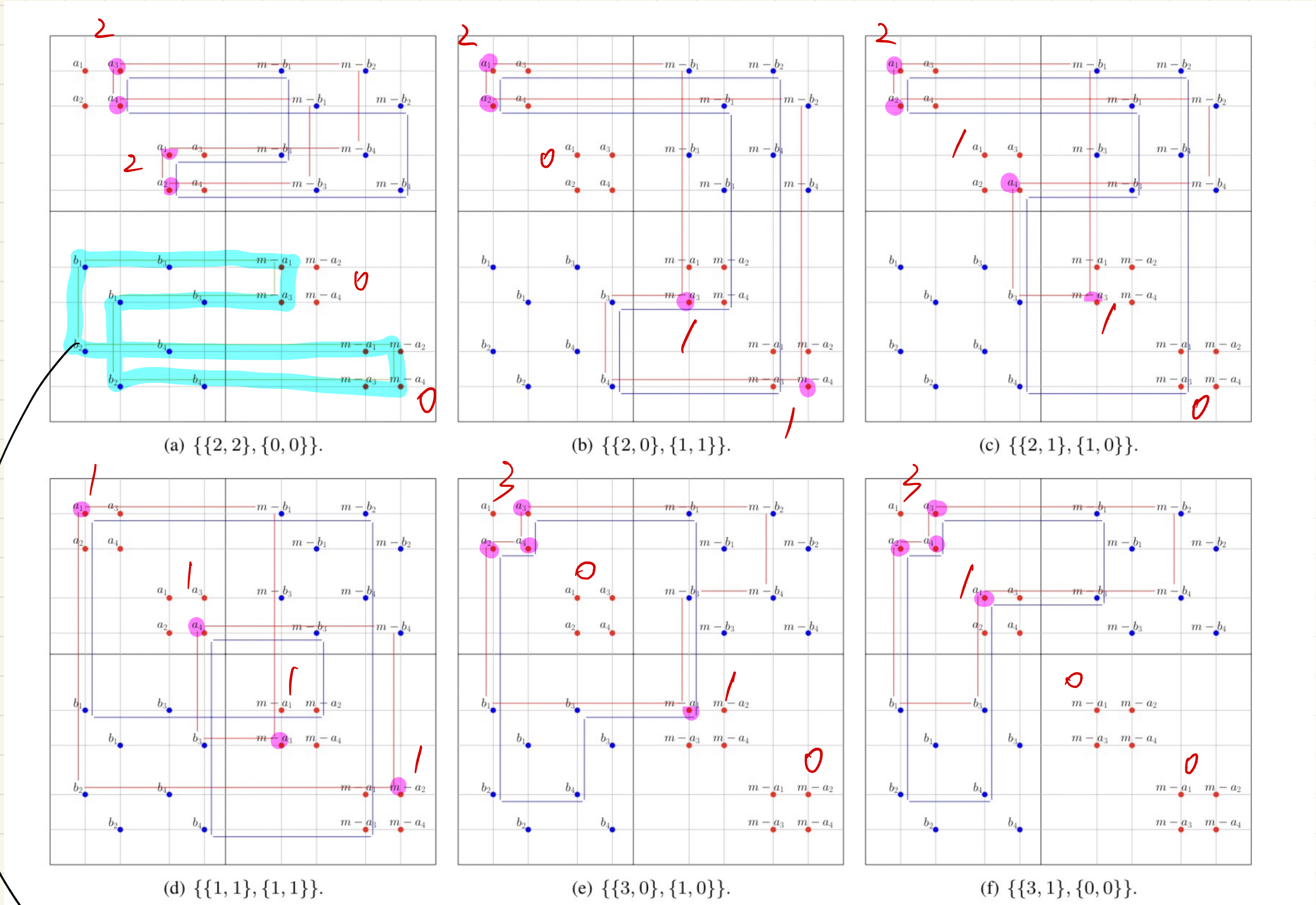
(a) Cycles-4 induced flexible cycle candidates. (b) Cycles-6 induced flexible cycle candidates.

$$
\begin{aligned}
N_4 =& n^A_{4,0,0}(n_2+r_2) + n^B_{4,0,0}(n_1+r_1),\\
N_6 =& n^A_{6,0,0}(n_2+r_2) + n^B_{6,0,0}(n_1+r_1) + 12(n^A_{4,0,0}e_2 + n^B_{4,0,0}e_1),\\
N_8 =& n^A_{8,0,0}(n_2+r_2) + n^B_{8,0,0}(n_1+r_1) + 30(n^A_{6,0,0}e_2 + n^B_{6,0,0}e_1)\\
&+ 124\Big(2n^A_{4,0,0}n^B_{4,0,0} + \sum_{(i,j)\neq(0,0)} (n^A_{4,i,j} + n^A_{4,-i,-j})(n^B_{4,i,j} + n^B_{4,-i,-j})\Big).
\end{aligned}
$$
[4]

$n^A_{2g,i,j}$ ($n^B_{2g,i,j}$): number of closed paths of length $2g$ with alternating sum i, j in 1st/2nd dimension.

$$f^A(S,T) = \sum_{i=0}^{m_1}\sum_{j=0}^{m_2} P^A_{i,j} S^i T^j, \quad f^B(S,T) = \sum_{i=0}^{m_1}\sum_{j=0}^{m_2} P^B_{i,j} S^i T^j$$

$$
\begin{aligned}
\mathbb{E}_{p^A,p^B}[N_6] =& n^A_6(n_2+r_2)\Big[(f^A(S,T)f^A(S^{-1},T^{-1}))^3\Big]_{0,0} + n^B_6(n_1+r_1)\Big[(f^B(S,T)f^B(S^{-1},T^{-1}))^3\Big]_{0,0},\\
\mathbb{E}_{p^A,p^B}[N_8] =& n^A_8(n_2+r_2)\Big[(f^A(S,T)f^A(S^{-1},T^{-1}))^4\Big]_{0,0} + n^B_8(n_1+r_1)\Big[(f^B(S,T)f^B(S^{-1},T^{-}))^4\Big]_{0,0}\\
&+ 30\left(n^A_6 e_B\Big[(f^A(S,T)f^A(S^{-1},T^{-1}))^3\Big]_{0,0} + n^B_6 e_A\Big[(f^B(S,T)f^B(S^{-1},T^{-1}))^3\Big]_{0,0}\right)\\
&+ 248 n^A_4 n^B_4\Big[(f^A(S,T)f^A(S^{-1},T^{-1}))^2(f^B(S,T)f^B(S^{-1},T^{-1}))^2\Big]_{0,0}.
\end{aligned}
$$



(a) {{2, 2}, {0, 0}}.
(b) {{2, 0}, {1, 1}}.
(c) {{2, 1}, {1, 0}}.
(d) {{1, 1}, {1, 1}}.
(e) {{3, 0}, {1, 0}}.
(f) {{3, 1}, {0, 0}}.

$\cancel{b_1}-(m-a_1)+(m-a_3)-\cancel{b_1}+\cancel{b_2}-(m-a_4)+(m-a_2)-\cancel{b_2}$

$= a_1 - a_2 + a_4 - a_3 \longrightarrow$ induced by a single cycle-4 in P_a or P_b. We didn't count this because cycles-4 will be removed anyways.

Optimization

1. Start with a random pair of permutation matrix
2. Modify each value to the value that locally minimizes the number of cycles (or weighted sum of N_6 & N_8).
   → first priority: remove all cycles-4.

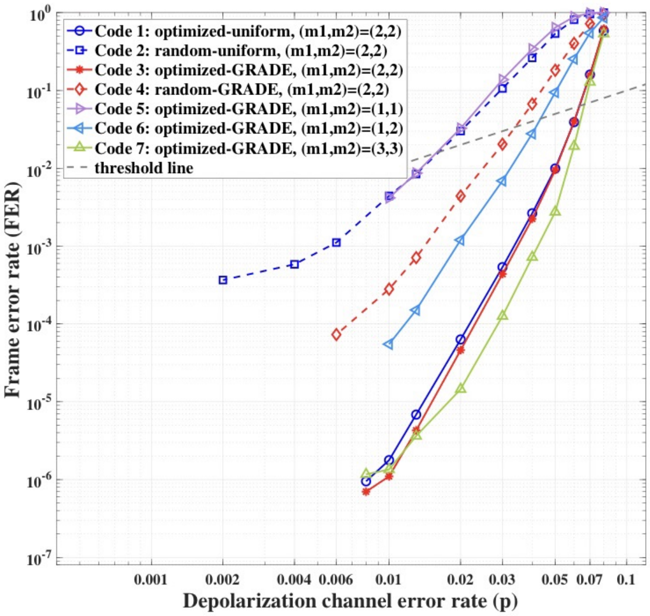


Table: Number of Flexible Cycles in [[7300, 2500]] codes

Common parameters	$(r_1, n_1, r_2, n_2, L_1, L_2) = (3, 8, 3, 8, 10, 10)$						
Codes	Code 1	Code 2	Code 3	Code 4	Code 5	Code 6	Code 7
Distributions	Uniform		GRADE				
Optimized by GRADE	✓	×	✓	×	✓		
(m_1, m_2)	(2, 2)				(1, 1)	(1, 2)	(3, 3)
Number of cycles 4	0	110	0	66	0	0	0
Number of cycles 6	11	264	11	143	583	198	0
Number of cycles 8	5113	>48142	5131	>23120	64585	23266	1144

has cycles-4

- Parameters:
  - $r = r_1 n_2 + r_2 n_1 = 48$
  - $n = n_1 n_2 + r_2 r_1 = 73$
  - $k = n - r = 25$
  - $N = n L_1 L_2 = 7300$
  - $K = k L_1 L_2 = 2500$

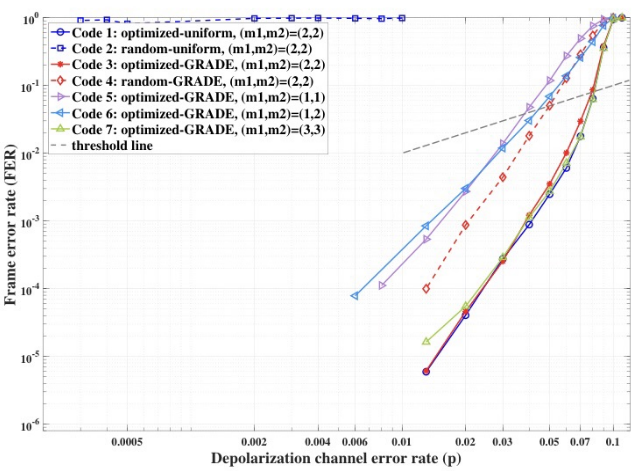


Table: Number of Flexible Cycles in [[5800, 1600]] codes

Common parameters	$(r_1, n_1, r_2, n_2, L_1, L_2) = (3, 7, 3, 7, 10, 10)$						
Codes	Code 1	Code 2	Code 3	Code 4	Code 5	Code 6	Code 7
Distributions	Uniform		GRADE				
Optimized by GRADE	✓	×	✓	×	✓		
(m_1, m_2)	(2, 2)				(1, 1)	(1, 2)	(3, 3)
Number of cycles 4	0	70	0	40	0	0	0
Number of cycles 6	0	70	0	140	320	70	0
Number of cycles 8	2316	>17708	2466	>18710	30424	8594	380

- Parameters:
  - $r = r_1 n_2 + r_2 n_1 = 42$
  - $n = n_1 n_2 + r_2 r_1 = 58$
  - $k = n - r = 16$
  - $N = n L_1 L_2 = 5800$
  - $K = k L_1 L_2 = 1600$

Decoder

① Binary (only for CSS)

- Split X, Z errors: e_x, e_z.
- Decode e_x, e_z by H_x, H_z using binary BP decoder, respectively.

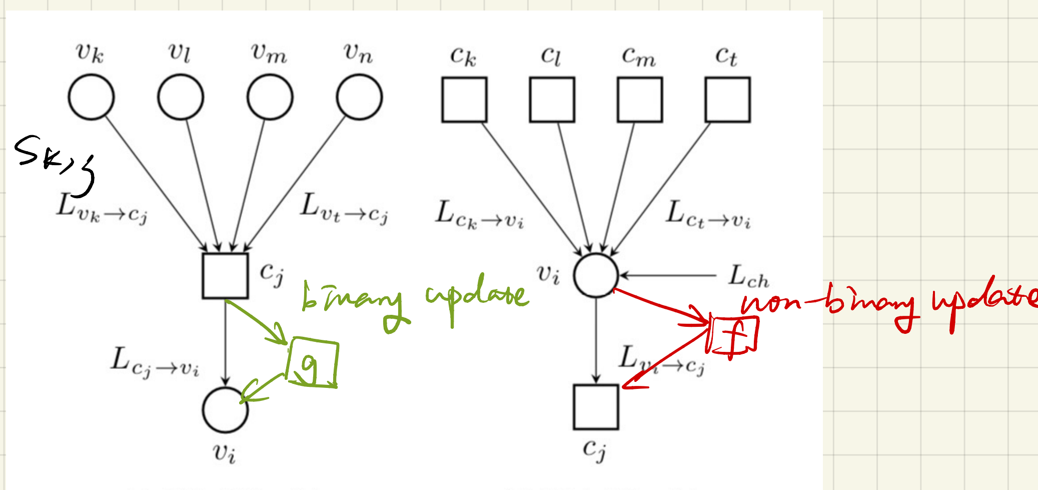
② Non-binary (also work for non-CSS codes)

- $\{I, X, Y, Z\} \longrightarrow \{0, 1, 2, 3\}$.

  $e \in \{0,1,2,3\}^N$, $H \in \{0,1,2,3\}^{(N-K)\times N}$
- syndrome $s \in \{0,1\}^N$.

$\log \dfrac{P[\langle S_{k,j}, V_k\rangle_S = 0]}{P[\langle S_{k,j}, V_k\rangle_S = 1]} \rightarrow$

$\left(\log\frac{P_X}{P_I}, \log\frac{P_Y}{P_I}, \log\frac{P_Z}{P_I}\right) \rightarrow$



(a) CN-to-VN update. (b) VN-to-CN update.

$$g(l; s) = \begin{cases} (0, -l, -l), & s = X \\ (-l, 0, -l), & s = Y \\ (-l, -l, 0), & s = Z \end{cases}$$

$$f((l_x, l_y, l_z); s) = \begin{cases} \log(1+e^{l_x}) - \log(e^{l_y}+e^{l_z}), & s = X \\ \log(1+e^{l_y}) - \log(e^{l_x}+e^{l_z}), & s = Y \\ \log(1+e^{l_z}) - \log(e^{l_x}+e^{l_y}), & s = Z. \end{cases}$$

③ At each iteration, have an estimation

$$\hat{V}_i = \begin{cases} I(0), & \text{if } l_x, l_y, l_z < 0 \\ X(1), & \text{if } l_x > 0,\ l_x > l_y, l_z. \\ Y(2), & \text{if } l_y > 0,\ l_y > l_x, l_z \\ Z(3), & \text{if } l_z > 0,\ l_z > l_x, l_y. \end{cases}$$

④ Check Syndrome values

→ If all syndromes are 0. → stop.

⑤ Check If decoder succeed or not.

→ If syndromes are all zero — success.

→ If Syndromes are not all zero.

→ Check if $\hat{V}_i$ belongs to the stabilizer group

↑ (For CSS codes:

① Have a systematic form of H_x, H_z at the beginning ($\tilde{H}_x$, $\tilde{H}_z$)

② split $\hat{V}$ into X, Z errors V_x, V_z.

Use $\tilde{H}_x$, $\tilde{H}_z$ to evaluate if $V_x \in \text{Span}(\tilde{H}_x)$

$V_z \in \text{Span}(\tilde{H}_z)$

↑

rowspace.

→ If yes, convergent error.

→ If no, non-convergent error

Degenerate Errors in QLDPC Codes.

Recall Definition of Stabilizer Codes.

$$S = \langle S_1, \dots, S_r \rangle,\ S_i \in \{I, X, Y, Z\}^{\otimes n}.$$

$$\mathcal{C} = \{ |\psi\rangle \mid S_i|\psi\rangle = |\psi\rangle,\ \forall i \in [r],\ |\psi\rangle \in \mathbb{C}^{2^n} \}.$$

If error $E \in S$. $E|\psi\rangle = |\psi\rangle \Rightarrow E \in S$ does not need to be corrected

↳ "Degenerate" Errors

BP Decoder are likely to perform worse in Quantum Codes due to degenerate errors

Exist Schemes

① Ordered Statistics Decoder (OSD)
② Stabilizer Inactivation (SI)

} both are post-processing algorithms after performing BP.

Ordered Statistics Decoder

errors after BP <

- non-convergent (large percent) — cannot be categorized but can likely be solved. (recall Maxwell Decoder in BEC)
- convergent — logical errors

$$w_i = |l_i| = \left|\log \frac{P(x_i=0)}{P(x_i=1)}\right| \longrightarrow \text{reliability of estimation } \hat{e}_i = \begin{cases} 0 & l_i > 0 \\ 1 & l_i < 0. \end{cases}$$

Order w_i of all VNs:

$$w_{\sigma(1)} \le w_{\sigma(2)} \le \cdots \le w_{\sigma(n)}.$$

$I = [n] / J$ information set

① $r = rk(H)$, find the first r indices $J = \{j_1, j_2, \cdots, j_r\}$, s.t., $rk(H_J) = r$.

② Change $\hat{e}_J$ into values such $H\hat{e}' = s$. (convergent). (*)

OSD-0 algorithm

That is not necessarily sufficient for finding $\hat{e}'$.

Modification: relax (*) by allowing modification over the least reliable w VNs in $\hat{e}_I$.

to OSD-w.

<u>OSD-w:</u>

find the $x \in \mathbb{F}_2^{w}$ s.t. if replacing $(\hat{e}_I)_{[w]}$ by x, the resulting $\hat{e}'(x)$ obtained from ② is of the smallest weight (among all choices of x).

OSD in Quantum Codes

$$p_i \leftarrow \max\{p_{i,X}, p_{i,Y}, p_{i,Z}, p_{i,I}\}.$$

Order the Qubits by

$$p_{\sigma(1)} \le p_{\sigma(2)} \le \cdots \le p_{\sigma(n)}.$$

Use the binary representation to perform OSD-w algorithm.

$$H' = (H_X \mid H_Z), \quad S' = (S_X \mid S_Z).$$

Obtain $\hat{e} = (\hat{e}_X \mid \hat{e}_Z) \Rightarrow \tilde{e} = \hat{e}_X X + \hat{e}_Z Z.$

Stabilizer Inactivation

Stabilizer splitting errors:

X error e_X, row r_X in H_X.

$$\mathrm{wt}|\underbrace{e_X + r_X}_{=e_X'}| = \mathrm{wt}|e_X^{*}| \longrightarrow$$

~~hard to~~ . BP can go into either choice (splits the probability of the coset into smaller probabilities of different representatives).

$$\delta(e_X, e_X') = \sum_{i \in \mathrm{supp}(e_X)} |\tilde{p}_i| - \sum_{i \in \mathrm{supp}(e_X')} |\tilde{p}_i| = \sum_{i \in \mathrm{supp}(r_X)} (-1)^{i \in e_X} |\tilde{p}_i|$$

$$\le \sum_{i \in \mathrm{supp}(r_X)} |\tilde{p}_i| = f(r_X).$$

Small $f(r_x)$ is more likely to lead to small $\delta(e_x, e_x')$'s, and thus is more likely to be unreliable.

Inactivate least reliable stabilizer r_x:

$$H_Z' = \left[\begin{array}{c|c} H_{Z|r_x} & A \\ \hline & H_{Z|\bar{r}_x} \end{array}\right].$$

Use $H_{Z|\bar{r}_x}$ to redo BP to obtain $\hat{e}_{x|\bar{r}_x}$.

Scan through $r_{x,\sigma(i)}$ with $f(r_{x,\sigma(1)}) \le f(r_{x,\sigma(2)}) \le \dots \le f(r_{x,\sigma(n)})$.

Until $H_{Z|\bar{r}_x} \hat{e}_{Z|\bar{r}_x} = S_{x|\bar{r}_x}$, and $H_{Z|r_x} e_{x|r_x} = S_{x|r_x} + A \hat{e}_{x|\bar{r}_x}$ has a solution.