

# Magic States Distillation Using Quantum LDPC Codes.

David Ponarovsky

March 6, 2024

## 1 Good Codes With Large $\Lambda$ .

**Definition 1.1.** Let  $M \in \mathbb{F}_2^{k \times n}$  upper triangular matrix such that  $k < n$ . We say that  $M$  has the 1-stairs property if  $M_{ij} = 1$  any  $j < i$ .

**Claim 1.1.** Any  $M \in \mathbb{F}_2^{k \times n}$  upper triangular matrix can be turn into upper triangular matrix that has the 1-stairs property by elementary operation.

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & 1 & 1 & 1 & 1 & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & 1 & 1 & 1 & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & 0 & 1 & 1 & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & 0 & 0 & 1 & \cdot & \cdot & \cdot & \cdot & \cdot \end{bmatrix}$$

*Proof.* Consider the following algorithm: Let  $M$  be our initial matrix. We iterate over the rows from left to right. In the  $i$ th iteration, we check for any row  $j < i$  if  $M_{ji} = 1$ . If not, we set  $M$  to be the matrix obtained by adding the  $i$ th row to the  $j$ th row. Since  $M$  is an upper triangular matrix, adding the  $i$ th row does not change any entry  $M_{js}$  for  $s < i$ . Therefore, the obtained matrix is still an upper triangular matrix and the entries at  $M_{js}$  for  $j, s < i$  remain the same, namely 1 if and only if  $j \leq s$ .

Continuing with the process eventually yields, after  $k$  iterations, a matrix with the 1-stair property.  $\square$

**Claim 1.2.** Let  $C$  be a  $[n, k, d]$  binary linear code, and let  $\Lambda$  be subcode  $\Lambda \subset C$  at dimension  $k'$  and distance  $d'$ . Then there exists a code  $C' = [\leq 2n, \geq k - k'/2, d]$  and a subcode of it  $\Lambda'$  in it at dimension  $\geq k'/2$  and distance  $d'$ , such:

1. For every  $x \in \Lambda'$  and  $y \in C'$   $x \cdot y = 0$
2. For every  $x \in \Lambda'$  and  $y, z \in C'$   $x \cdot y \cdot z = 0$

*Proof.* First, we can assume that the generator matrix of  $C$  is an upper triangular matrix, such that the first  $k'$  rows span  $\Lambda$ . Notice that after applying the algorithm from claim 1.1 starting from the first row and stopping at the  $k'$ th row, the first  $k'$  rows are kept in  $\Lambda$ . So let's assume that is the form of the generator matrix.

Now, let's consider the following process: going uphill, from right to left, starting at the  $k'$  row. Initially, set  $j \leftarrow k'$  and in each iteration, advance it to be the index of the next row, namely  $j \leftarrow j - 1$ . In each iteration, ask how many rows  $G_m$ , such that  $m \leq j$ , satisfy  $G_m G_j = 0$  and how many pairs of rows  $G_m, G_{m'}$  such that  $m, m' \leq j$  satisfy  $G_m \cdot G_{m'} \cdot G_j = 0$ . Denote by  $p$  the probability to fall on unsatisfied equation from the above.

- If  $p \geq \frac{1}{2}$  then we move on to the next iteration.
- Otherwise, we encode the  $j$ th coordinate by  $C_0$ , which maps  $1 \rightarrow w$  such that  $w \cdot w = 0$ . This flips the value of  $G_m G_j$  for any pair and  $G_m G_{m'} G_j$  for any triple such that  $m, m' \leq j$ , so we get that the majority of the equations are satisfied. Also notice that the concatenation doesn't change the value of any multiplication at the form  $G_m G_{j'}$  for  $j' > j$ . Therefore, for any  $j < j' \leq k'$  the number of the satisfied equations relative to  $j'$  is not changed, meaning it is still the majority.

Set  $G$  to be the new matrix after the concatenation by  $C_0$ .

In the end of the process  $G$  is going to be the generator matrix of  $C'$ . It's left to construct  $\Lambda'$ , we are going to do so by taking from the  $k'$  rows a subset that satisfies the desired property in Claim 1.2.

Let  $S$  be the set of rows among the first  $k'$  rows for which there is at least one unsatisfied equation. We will now prove that if  $k'$  is large enough, specifically linear in  $k$ , then  $|S|$  is small enough to obtain  $\Lambda'$  by removing the rows in  $S$ .

Observe that the number of satisfied equations is at least:

$$\begin{aligned} & \frac{1}{2} (k' - 1 + (k' - 1)^2) + \frac{1}{2} (k' - 1 + (k' - 1)^2) + \frac{1}{2} (k' - 2 + (k' - 2)^2) + \dots + \frac{1}{2} (1 + (1)^2) \\ &= \frac{1}{2} \left( \binom{k' + 1}{2} + \frac{k'(k' + 1)(2k' + 1)}{6} \right) \end{aligned}$$

So

$$\begin{aligned} |S| \cdot k + |S| \cdot k^2 &\leq k' (k + k^2) - \frac{1}{2} \left( \binom{k' + 1}{2} + \frac{k'(k' + 1)(2k' + 1)}{6} \right) \\ \Rightarrow |S| &< k' - \frac{1}{2} \left( \frac{1}{k^2 + k} \binom{k' + 1}{2} + \frac{1}{k^2 + k} \frac{k'(k' + 1)(2k' + 1)}{6} \right) \\ \Rightarrow |S| &< k' - \frac{k'^3}{24k^2} \end{aligned}$$

Therefore, if  $k' \geq \alpha k$  we have that  $|S| < (\alpha - \frac{\alpha^3}{24})k$  implies that  $\dim \Lambda' \geq \frac{\alpha^3}{24}k$ . □

**Claim 1.3** (Not Formal). *It is easy to see that by using concatenation again, one can obtain the code  $\dim \Lambda' \leftarrow \frac{1}{2} \dim \Lambda'$ . For any  $x \in \Lambda'$ ,  $|x|_4 = 1$ , and for any  $x \in C'/\Lambda'$ , we have  $|x|_4 = 0$ .*

**[COMMENT]** The argument above that the distance  $d'$  remain the same is not correct. Yet, if we are defining the distance of any codeword in  $C/(C/\Lambda)$  to be greater than  $d'$  then we win. (The problem was that gauss elimination might change the weight of rows associate with  $\Lambda$  generators.

## 2 Distillate $|\Lambda + C_Z^\perp\rangle$ Into Magic.

Let  $|f\rangle$  be a codeword in  $C_{\hat{X}}$ , and let  $\hat{X}_g$  be the indicator that equals 1 if  $f$  has support on  $\hat{X}_g$ , and 0 otherwise. Observe that applying  $T^\otimes$  on  $|f\rangle$  yields the state:

$$\begin{aligned} T^{\otimes n} |f\rangle &= T^{\otimes n} \left| \sum_g \hat{X}_g g \right\rangle = \exp \left( i\pi/4 \sum_g \hat{X}_g |g| - 2 \cdot i\pi/4 \sum_{g,h} \hat{X}_g \hat{X}_h |g \cdot h| \right. \\ &\quad \left. + 4 \cdot i\pi/4 \sum_{g,h} \hat{X}_g \hat{X}_h \hat{X}_l |g \cdot h \cdot l| - 8 \cdot i\pi/4 \cdot \text{integers} \right) |f\rangle \\ &= \exp \left( i\pi/4 \sum_g \hat{X}_g |g| - 2 \cdot \pi/4 \sum_{g,h} \hat{X}_g \hat{X}_h |g \cdot h| + 4 \cdot i\pi/4 \sum_{g,h} \hat{X}_g \hat{X}_h \hat{X}_l |g \cdot h \cdot l| \right) |f\rangle \end{aligned}$$

So in our case:

$$\begin{aligned} T^{\otimes n} |f\rangle &= \\ &= \exp \left( i\pi/4 \sum_{g \in \Lambda} \hat{X}_g \right. \\ &\quad - 2 \cdot \pi/4 \sum_{g \in \Lambda, h} 2\hat{X}_g \hat{X}_h \\ &\quad - 2 \cdot \pi/4 \sum_{g,h \in C_Z^\perp} \hat{X}_g \hat{X}_h |g \cdot h| \\ &\quad \left. + 4 \cdot i\pi/4 \sum_{g,h \in C_Z^\perp} \hat{X}_g \hat{X}_h \hat{X}_l |g \cdot h \cdot l| \right) |f\rangle \end{aligned}$$

So eventually, we have a product of gates when non-Clifford gates are applied on only one generator of  $C_Z^\perp$ .

$$T^n |f\rangle = \prod_{g \in \Lambda} T_g \prod_{g \in \Lambda, h} \{CZ_{g,h}|I\rangle\} \prod_{g,h \in C_Z^\perp} \{CS_{g,h}|CZ_{g,h}|I\rangle\} \prod_{g,h,l \in C_Z^\perp} \{CCZ_{g,h,l}|I\rangle\} |f\rangle$$

Decompose  $f = f_1 + f_2$ , where  $f_1$  is supported only on  $C_X/C_Z^\perp$  and  $f_2$  is supported only on  $C_Z^\perp$ . By using commuting relations, the above can be turned into.

$$T^n |f\rangle = \prod_{g \in \Lambda, h} \{CZ_{g,h}|I\rangle\} \prod_{g \in \Lambda} T_g X_{f_1} \prod_{g,h \in C_Z^\perp} \{CS_{g,h}|CZ_{g,h}|I\rangle\} \prod_{g,h,l \in C_Z^\perp} \{CCZ_{g,h,l}|I\rangle\} |f_2\rangle$$

Denote by  $M_1, M_2$  the gates:

$$M_1 = \prod_{g \in \Lambda, h} \{CZ_{g,h}|I\rangle\}$$

$$M_2 = \prod_{g,h \in C_Z^\perp} \{CS_{g,h}|CZ_{g,h}|I\rangle\} \prod_{g,h,l \in C_Z^\perp} \{CCZ_{g,h,l}|I\rangle\}$$

And then we get that

$$\prod_{g \in \Lambda} T_g |f\rangle = M_1^\dagger T^n M_2^\dagger |f\rangle$$

$$\prod_{g \in \Lambda} T_g |f\rangle = M_1^\dagger T^n E L[M_2^\dagger] |L[f]\rangle$$

$$\prod_{g \in \Lambda} (I + T_g X_g) |C_Z^\perp\rangle = M_1^\dagger T^n \prod_{g \in \Lambda} E L[M_2^\dagger] \prod_{g \in C_Z^\perp \cup \Lambda} (I + X_{L[g]}) |0\rangle$$

**Claim 2.1.** *The state  $|C_Z^\perp + \text{span } \Lambda\rangle |0\rangle$  can be compute, such that the light cone depth of any non-clifford gate is bounded by constant.*

*Proof.*

$$\begin{aligned} (H_X \otimes I) C X_{n \rightarrow n} (E \otimes E) I \otimes L[M_2^\dagger] \prod_{J \in \{\Lambda, C_Z^\perp\}} \prod_{g \in J} (I + X_{L[g]}) & |0\rangle |0\rangle \\ = (H_X \otimes I) C X_{n \rightarrow n} \sum_{\substack{z \in C_Z^\perp \\ x \in \text{span } \Lambda}} e^{\varphi(z)} & |x\rangle |z\rangle \\ = \sum_{\substack{z \in C_Z^\perp \\ x \in \text{span } \Lambda}} e^{\varphi(z)} & |x+z\rangle |0\rangle \\ = \sum_{\substack{z \in C_Z^\perp \\ x \in \text{span } \Lambda}} (M_2^\dagger \otimes I) & |x+z\rangle |0\rangle \\ = (M_2^\dagger \otimes I) & |C_Z^\perp + \text{span } \Lambda\rangle |0\rangle \end{aligned}$$

□

**Claim 2.2.** *The logical operator  $CX_g$  relative the code  $C_Z^\perp$  can be implement such it acts on constant number of qubits. **Notice,** implementation of the gate  $CX_g$  relative to  $C_Z^\perp$  might incorrect for computing  $CX_g$  relative to  $C_X$ .*

