Understanding Quantumness And Testability.

David Ponarovsky

July 31, 2023

Contents

1 Introduction **7**

2 Codes **9**
2.1 Introduction 9
2.2 Notations, Definitions, And Our Contribution 9
2.3 Singleton Bound 10
2.4 Tanner Code 11
2.5 Expander Codes 13
2.6 Randomized Constructions. 14
2.7 Locally Testable Codes. 15
2.7.1 Polynomial Code. 19

3 Quantum Error Correction Codes. **21**
3.1 Introduction. 21
3.2 Quantum Noise. 22
3.3 CSS Codes. 23
3.4 qLDPC Codes. 26
3.5 Quantum Expander Codes. 27

4 Good qLDPC and LTC. **31**
4.1 Decoding and Testing 33

5 First Attempt For Good qLTC. **37**
5.1 The Polynomial-Code Is Not w-Robust. 37

6 Local Majority $\neq$ Local Testability. **39**
6.0.1 Overcoming The Vanishing Rate. 41

List of Figures

2.1 Peterson Graph. 12

2.2 The plot $x \mapsto (x-1)(x-2)$ and $x \mapsto (x-1)(x-4)$ presents the extension of the polynomials 20

3.1 Toric Graph. 27

Chapter 1

Introduction

The PCP theorem states that there exists a class of computational problems that can be probabilistically verified with high accuracy by reading only a constant number of bits from a proof that is polynomial in the input size of the problem. This means that given a proof for a problem, a verifier can check the proof for correctness by reading only a few bits of the proof, making the verification process efficient.

Chapter 2

Codes

2.1 Introduction

Coding theory has emerged due to the need to transfer information in noisy communication channels. By embedding a message in a higher-dimensional space, one can guarantee robustness against possible faults. The ratio of the original content length to the transmitted message *length* is the *rate* of the code, and it measures how consuming our communication protocol is. Additionally, the *distance* of the code quantifies how many faults the scheme can absorb such that the receiver can recover the original message. We can consider the code as a collection of all strings that satisfy specified restrictions.

Non-formally, a code is good if its distance and rate scale linearly with the encoded message length. In practice, one is also interested in implementing these checks efficiently. We say that a code is an LDPC if any bit is involved in a constant number of restrictions, each of which is a linear equation, and if any restriction contains a fixed number of variables.

Moreover, another characteristic of the code is its testability, which is the complexity of the number of random checks one must do to verify that a given candidate is in the code. Besides being considered efficient in terms of robustness and overhead, good codes are also vital components in establishing secure multiparty computation [BGW19] and have a deep connection to probabilistic proofs.

In Section 2, we state the notations, definitions, and formal theorem. Then, in Sections 3 and 4, we review past results and provide their proofs to make this paper self-contained. Readers familiar with the basic concepts of LDPC, Tanner, and Expanders codes construction may consider skipping directly to Section 5, in which we provide our proof. Readers familiar with the basic concepts of LDPC, Tanner, and Expanders codes construction may skip Sections 2, 3, and 4 and proceed directly to Section 5, where we provide our proof.

2.2 Notations, Definitions, And Our Contribution

Here we focus only on linear binary codes, which one could think about as linear subspaces of $\mathbb{F}_2^n$. A common way to measure resilience is to ask how many bits an evil entity needs to flip such that the corrupted vector will be closer to another vector in that space than the original one. Those ideas were formulated by Hamming [Ham50], who presented the following definitions.

Definition 1. *Let $n \in \mathbb{N}$ and $\rho, \delta \in (0,1)$. We say that C is a **binary linear code** with parameters $[n, \rho n, \delta n]$. If C is a subspace of $\mathbb{F}_2^n$, and the dimension of C is at least ρn. In* *addition, we call the vectors belong to C codewords and define the distance of C to be the minimal number of different bits between any codewords pair of C.*

From now on, we will use the term code to refer to linear binary codes, as we don't deal with any other types of codes. Also, even though it is customary to use the above parameters to analyze codes, we will use their percent forms called the relative distance and the rate of code, matching δ and ρ correspondingly.

Definition 2. *A* ***family of codes*** *is an infinite series of codes. Additionally, suppose the rates and relative distances converge into constant values ρ, δ. In that case, we abuse the notation and call that family of codes a code with $[n, \rho n, \delta n]$ for fixed $\rho, \delta \in [0, 1)$, and infinite integers $n \in \mathbb{N}$.*

Notice that the above definition contains codes with parameters attending to zero. From a practical view, it means that either we send too many bits, more than a constant amount, on each bit in the original message. Or that for big enough n, adversarial, limited to changing only a constant fraction of the bits, could disrupt the transmission. That distinction raises the definition of good codes.

Definition 3. *We will say that a family of codes is a* ***good code*** *if its parameters converge into positive values.*

2.3 Singleton Bound

To get a feeling of the behavior of the distance-rate trade-of, Let us consider the following two codes; each demonstrates a different extreme case. First, define the repetition code $C_r \subset \mathbb{F}_2^{n \cdot r}$, In which, for a fixed integer r, any bit of the original string is duplicated r times. Second, consider the parity check code $C_p \subset \mathbb{F}_2^{n+1}$, in which its codewords are only the vectors with even parity. Let us analyze the repetition code. Clearly, any two n-bits different messages must have at least a single different bit. Therefore their corresponding encoded codewords have to differ in at least r bits. Hence, by scaling r, one could achieve a higher distance as he wishes. Sadly the rate of the code decays as $n/nr = 1/r$. In contrast, the parity check code adds only a single extra bit for the original message. Therefore scaling n gives a family which has a rate attends to $\rho \to 1$. However, flipping any two different bits of a valid codeword is conversing the parity and, as a result, leads to another valid codeword.

To summarize the above, we have that, using a simple construction, one could construct the codes $[r, 1, r]$, $[r, r-1, 2]$. Each has a single perfect parameter, while the other decays to the worst.

Besides being the first bound, Singleton bound demonstrates how one could get results by using relatively simple elementary arguments. It is also engaging to ask why the proof yields a bound that, empirically, seems far from being tight.

Theorem (Singleton Bound.). *For any linear code with parameter $[n, k, d]$, the following inequality holds:*

$$k + d \leq n + 1$$

Proof. Since any two codewords of C differ by at least d coordinates, we know that by ignoring the first $d-1$ coordinate of any vector, we obtain a new code with one-to-one corresponding to the original code. In other words, we have found a new code with the same dimension embedded in $\mathbb{F}_2^{n-d+1}$. Combine the fact that dimension is, at most, the dimension of the container space, we get that:

$$\dim C = 2^k \leq 2^{n-d+1} \Rightarrow k + d \leq n + 1$$

□

It is also well known that the only binary codes that reach the bound are: $[n, 1, n]$, $[n, n-1, 2]$,$[n, n, 1]$ [AF22]. In particular, there are no good binary codes that obtain equality(And no binary code which get close to the equality exits). Let's review the polynomial code family [RS60], which is a code over none binary field that achieve the Singleton Bound.

Next, we will review Tanner's construction, that in addition to being a critical element to our proof, also serves as an example of how one can construct a code with arbitrary length and positive rate.

2.4 Tanner Code

The constructions require two main ingredients: a graph Γ, and for simplicity, we will restrict ourselves to a Δ regular graph, Yet notice that the following could be generalize straightforwardly for graphs with degree at most Δ. The second ingredients is a ;small' code C_0 at length equals the graph's regularity, namely $C_0 = [\Delta, \rho\Delta, \delta\Delta]$. We can think about any bit string at length Δ as an assignment over the edges of the graph. Furthermore, for every vertex $v \in \Gamma$, we will call the bit string, which is set on its edges, the local view of v. Then we can define, [Tan81]:

Definition 4. *Let $C = \mathcal{T}(\Gamma, C_0)$ be all the codewords which, for any vertex $v \in \Gamma$, the local view of v is a codeword of C_0. We say that C is a **Tanner code** of Γ, C_0. Notice that if C_0 is a binary linear code, So C is.*

Example 1. *Consider the Petersen graph Γ, which is a regular graph with degree 3. Let C_0 be the set of all words with even parity. It follows that C_0 contains all even-length binary strings of length 3: 000, 110, 101, and 011. However, the size of $\mathcal{T}(\Gamma, C_0)$ is significantly larger, as shown in Figure fig. 2.1. Specifically, any rotation of the inner and outer cycles simultaneously gives rise to another valid codeword, so any assignments that are not invariant under these rotations would produce five additional valid codewords.*

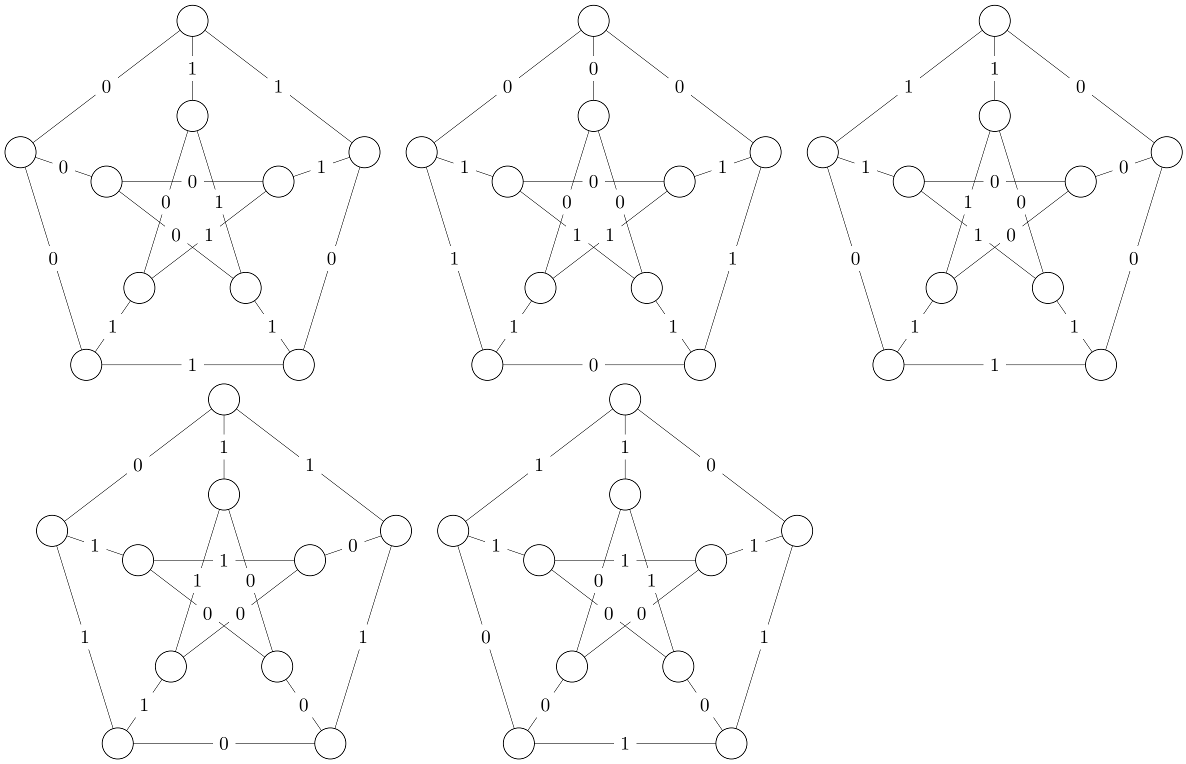

Figure 2.1: Peterson Graph.

Lemma 1. *Tanner codes have a rate of at least* $2\rho - 1$.

Proof. The dimension of the subspace is bounded by the dimension of the container minus the number of restrictions. So assuming non-degeneration of the small code restrictions, we have that any vertex count exactly $(1-\rho)\,\Delta$ restrictions. Hence,

$$\dim C \geq \frac{1}{2}n\Delta - (1-\rho)\,\Delta n = \frac{1}{2}n\Delta\,(2\rho - 1)$$

Clearly, any small code with rate $> \frac{1}{2}$ will yield a code with an asymptotically positive rate $\square$

Based on Lemma 1, we can obtain a recipe for constructing codes with a almost non-vanishing rate for arbitrarily large lengths and dimensions. This recipe involves concatenating a series of Tanner codes over complete graphs. To be more precise, we can define a family of codes as follows:

$$\begin{aligned} C_{i+1} &= \mathcal{T}\left(K_{n(C_i)+1}, C_i\right) \\ C_0 &= \text{ Some simple } \Delta[1, \rho_0, \delta_0] \text{ code.} \end{aligned}$$

Where $n(C_i)$ represents the code length of the ith code. Repeating the process described above $\log^*_\Delta(n)$ times allows us to extend the initial code $\Delta[1, \rho_0]$ to $n[1, \sim 2\rho^{\log^*_\Delta(n)}]$. Interestingly, any family of finite groups generated by a constant-size generator set can define a family of codes by utilizing their Cayley graphs as a basis for Tanner codes.

Once we have seen that Tanner codes enable us to achieve rates, the next natural question to ask is about the distance of the codes. Achieving a linear distance requires a little bit more from the graphs, but to understand this idea better, let us return to the repetition code. For instance, the repetition code can be presented as a Tanner code over the cycle graph.

Example 2. *In this representation, each vertex checks if the bits on its edges are equal. A valid codeword is an assignment in which all the bits are equal, since otherwise, there would be an edge with no supporting vertex. An illustration of a legal assignment is provided in* **??**.

Recall that the distance of a linear code is the minimal weight of the non-zero codewords. Consider a codeword $c \in C$ and group the vertices by four sets V_i such that V_i is the set of vertices that see $i \in \{0,1\}^2$. Since $c \in C$, we have that $|V_{10}| = |V_{01}| = 0$. Additionally, any vertex in V_{00} is not connected to V_{11}, which gives us two possible cases: either all the vertices in V_{11} are isolated, or the graph is not connected. Hence, the distance of the code is equal to $\frac{1}{2}\sum |V_i| \cdot |i| = \frac{1}{2} 2 \cdot n = n$.

Analyzing the repetition code gives a clue of how, in certain cases, one might prove a lower bound on the code distance. We would like to say that, if the weight of the code word is below the distance, then it is must be that is at least one vertex see non trivial local view which is not a codeword in C_0. Put it differently, we can't spread a small weight codeword over $\{V_i\}$, defined above, without to expanse into subsets corresponds to low $|i|$. Next we are going to present the Expander codes, which are Tanner codes constructed by graph with good algebraic expansion.

2.5 Expander Codes

We saw how a graph could give us arbitrarily long codes with a positive rate. We will show, Sipser's result [SS96] that if the graph is also an expander, we can guarantee a positive relative distance.

Definition 5. *Denote by λ the second eigenvalue of the adjacency matrix of the Δ-regular graph. For our uses, it will be satisfied to define expander as a graph $G = (V, E)$ such that for any two subsets of vertices $T, S \subset V$, the number of edges between S and T is at most:*

$$\mid E\left(S,T\right) - \frac{\Delta}{n}|S||T| \mid \leq \lambda\sqrt{|S||T|}$$

This bound is known as the Expander Mixining Lemma. We refer the reader to [HLW06] for more deatilied survery.

Theorem. *Theorem, let C be the Tanner Code defined by the small code $C_0 = [\Delta, \delta\Delta, \rho\Delta]$ such that $\rho \geq \frac{1}{2}$ and the expander graph G such that $\delta\Delta \geq \lambda$. C is a good LDPC code.*

Proof. We have already shown that the graph has a positive rate due to the Tunner construction. So it's left to show also the code has a linear distance. Fix a codeword $x \in C$ and denote By S the support of x over the edges. Namely, a vertex $v \in V$ belongs to S if it connects to nonzero edges regarding the assignment by x, Assume towards contradiction that $|x| = o(n)$. And notice that $|S|$ is at most $2|x|$, Then by The Expander Mixining Lemma we have that:

$$\begin{aligned}\frac{E(S,S)}{|S|} &\leq \frac{\Delta}{n}|S| + \lambda \\ &\leq_{n\to\infty} o(1) + \lambda\end{aligned}$$

Namely, for any such sublinear weight string, x, the average of nontrivial edges for the vertex is less than λ. So there must be at least one vertex $v \in S$ that, on his local view, sets a string at a weight less than λ. By the definition of S, this string cannot be trivial. Combining the fact that any nontrivial codeword of the C_0 is at weight at least $\delta\Delta$, we get a contradiction to the assumption that v is satisfied, videlicet, x can't be a codeword □

Setting C_0 To Be The Polynomial Code.

$$\begin{aligned}\log\Delta \dim C &\geq \frac{1}{2}n\Delta - (1-\rho)\Delta n = \frac{1}{2}n\Delta(2\rho - 1) \\ \Rightarrow \dim C &\geq \frac{1}{\log\Delta}\frac{1}{2}n\Delta(2\rho-1)\end{aligned}$$

2.6 Randomized Constructions.

Claim 1. *Let A be a random matrix in $M(\mathbb{F}_2^{k\times n})$. For any non-zero $x \in \mathbb{F}$, we have that Ax is distributed uniformly.*

Proof. By the fact that $x \neq 0$, there exists at least one coordinate $i \in [k]$ such that $x_i \neq 0$. Thus, we have

$$\begin{aligned}(Ax)_j &= \sum_k A_{jk}x_k = \sum_{i\neq k} A_{jk}x_k + A_{ji}x_i \\ &= \sum_{i\neq k} A_{jk}x_k + A_{ji}\end{aligned}$$

Notice that due to the fact that $\mathbb{F}_2$ is a field, there is exactly one assignment that satisfies the equation conditioned on all the values A_{jk} where $j \neq k$.

$$\begin{aligned}\mathbf{Pr}\left[(Ax)_j = 1\right] &= \sum_{A_{jk};k\neq i} \mathbf{Pr}\left[(Ax)_j = 1 \mid A_{jk}; k\neq i\right]\mathbf{Pr}\left[A_{jk}; k\neq i\right] \\ &= \frac{1}{2}\end{aligned}$$

Therefore, any coordinate of Ax is distributed uniformly $\Rightarrow$ Ax is distributed uniformly. □

By the uniformity of Ax, we obtain that the expected Hamming weight of Ax is:

$$\mathbf{E}\left[|Ax|\right] = \mathbf{E}\left[\sum_i^n (Ax)_i\right] = \frac{1}{2}n$$

As the coordinates of A_x are independent (each row of A is sampled separately), we can use the Hoff's bound to conclude that:

$$\mathbf{Pr}\left[\left||Ax| - \mathbf{E}\left[|Ax|\right]\right| \geq \left(\frac{1}{2} - \delta\right)n\right] \leq e^{-n\left(\frac{1}{2}-\delta\right)^2}$$

Now, we will use the union bound to show that any $x \in \mathbb{F}_2^k$, Ax is of weight at least δ.

$$\mathbf{Pr}\left[|Ax| \geq \delta : \forall x \in \mathbb{F}_2^k\right] \geq 1 - |\mathbb{F}_2^k| \cdot e^{-n\left(\frac{1}{2}-\delta\right)^2}$$

Denote $k = \rho n$ and notice that the above holds when $\rho \geq \left(\frac{1}{2} - \delta\right)^2$.

Note On Quantum Polynomial Code.

Let's define the code C such that any state in C is a coset of the polynomials at degree at most d shifted by $x \in \mathbb{F}_p$. In other words the codeword associated with x is the state $|\underline{c}\rangle = \sum_{\substack{f \in \mathbb{F}_d[x] \\ f(0)=0}} |c + f\rangle$. The inner product between any d-degree polynomial with zero free coefficient is:

$$\langle f | x^j \rangle = \sum_{i \leq d} \langle a_i x^i | x^j \rangle = \sum_{i \leq d} a_i \mathbf{E}\left[x^i x^j\right] = \sum_{i \leq d} a_i \mathbf{1}_{i+j=_n 0}$$

[COMMENT] Say some words about the classily testability of the polynomial code, and why for quantum it doesn't work. (The dual space of polynomials of low degree is the subspace of all the polynomials with heigh degree.)

2.7 Locally Testable Codes.

Apart from distance and rate here, we interest also that the checking process will be robust. In particular, we wish that against significant errors, forgetting to perform a single check will sabotage the entire computation only with a tiny probability.

Definition 6. *Consider a code C a string x, and denote by $\xi(x)$ the fraction of the checks in which x fails. C will be called a **local-testability** $f(n)$ If there exists $\kappa > 0$ such that*

$$\frac{d(x, C)}{n} \leq \kappa \cdot \xi(x) f(n)$$

We are going to introduce the polynomial code in an opposite order as in the usual literature. Instead of starting by presenting polynomials and the code itself we will begin by defining an abstract decoder, and define the code to be all the strings on which the decoder is apathetic. Consider the alphabet Σ, and let D be a decoder such that on every $d+1$ coordinate $x_1, x_2, ..x_{d+1}$ and candidate for a code word c, read $c_1, c_2..c_d$ and returns a charter in Σ. We will think of D as tester also. On given candidate $c \in \Sigma^n$, D accept if for any $x_1, x_2..x_{d+1}$ it holds that $C_{d+1} = D(c; x_1, x_2..x_{d+1})$. We will associciate a single check with spesific $d+1$ coordinates.

The setting avbove is kind general, and we will need to rquire form the code more to be a locally testable. Let us define the following property which could be thought as induction for cheks degenration:

Definition 7. *We will say that code C has the degeneration property if for any chack h there is a set of checks $H(h)$ such that:*

1. *if H satisfied then also h satisifed.*
2. *it easy to chack H.*

Claim 2. *The code defined by all the strings on which D accept with probability 1, when the randomness is over the sampled coordinates $x_1, x_2..x_{d+1}$ is a locally testable code. In particular, if the c is in the code, then a random check successes with probability 1 and for every c that is ε-far from the code the probability to, accept is at most $\binom{d}{2}|\Sigma|\varepsilon$.*

Proof. Notice that by definition if the tester accept with probability 1 then $c \in C$. Suppose that c pass the test with probability at least $1-\varepsilon$. First, let us define the codeword l such that $l_y = Majoritiy_{x_1..x_d} D\left(c, x_1..x_d, y\right)$.

Claim 3. *The distance between l and c is at most $1 - \left(1 - \frac{1}{\Sigma}\right)^{-1}\varepsilon$.*

Proof. Denote by α the probability that for given y, D accept on more then $\frac{1}{|\Sigma|}$ fraction to choose $x_1, x_2..x_d$. Then we have that:

$$\begin{aligned}
1-\varepsilon &\leq \mathbf{Pr}_{x_1..x_{d+1}}\left[D\left(x_1..x_{d+1}\right)\right] \\
&= \mathbf{Pr}_{D(x_1..x_{d+1})}\left[x_1..x_{d+1} \middle| x_{d+1} \in A\right] \cdot \mathbf{Pr}\left[x_{d+1} \in A\right] + \\
&\quad \mathbf{Pr}_{D(x_1..x_{d+1})}\left[x_1..x_{d+1} \middle| x_{d+1} \notin A\right] \cdot \mathbf{Pr}\left[x_{d+1} \notin A\right] \\
&\leq 1 \cdot \alpha + \frac{1}{|\Sigma|}(1-\alpha) = \left(1 - \frac{1}{|\Sigma|}\right)\alpha + \frac{1}{|\Sigma|} \\
&\Rightarrow \alpha \geq 1 - \left(1 - \frac{1}{\Sigma}\right)^{-1}\varepsilon
\end{aligned}$$

□

So we showed that l and c are close. It's left to show that l is a code word, namely that any check by D pass. Fix a check, which equivalence as fix $d+1$ coordinates $x_1..x_{d+1}$. Now notice that by the degenration property if there exists y such that for every $i \in [d+1]$ $D\left(l; y_1..y_d, x_i\right) = l_{x_i}$ than it follows that also $D\left(l; x_1..x_{d+1}\right)$ pass.

$$\{D(G(c); x)\} \subset \bigcap_{x_i} \{D\left(x_i; y_1..y_d\right)\}$$

□

Denote by $V(x)$ the vandermonde matrix defined by x namely $V(x)_{ij} = x_i^j$. and by w the d root of 1.

Claim 4. *Now $\sum_t V\left(x + w^t y\right) = t \cdot \left(V(x) + V(y)\right)$.*

$$\left(\sum_t V(x + w^t y)\right)_j = \sum \sum w^{l \cdot t} \binom{j}{l} x_i^l y_i^{j-l} = t\left(\cdot x^j + y^j\right) + \sum_{l \neq 0, j} \frac{w^{l \cdot d} - 1}{w^l - 1} \cdot \binom{j}{l} x_i^l y_i^{j-l}$$

Claim 5. *Consider the finte field $\mathbb{F}_q$ for prime q, denote by w the d root of 1, namely $w^d = 1$ then for any polynomial f with degree at most d it holds:*

$$f(x) = \frac{1}{d} \sum_t^d f(x + w^t y) \quad \forall x, y \in \mathbb{F}_q \tag{2.1}$$

Proof. Denote by a_j the jth coffecicient of f. And recall that for any fiexd $l \in \mathbb{F}_q$ the summtiom $\sum_t^d w^{t \cdot l}$ equals:

$$\sum_t^d w^{t \cdot l} = \begin{cases} \frac{w^{l \cdot d}-1}{w^l-1} = 0 & l \neq 0 \\ d & l = 0 \end{cases}$$

Thus:

$$\begin{aligned} &\sum_t f\left(x + w^t y\right) = \sum_{t,j} a_j \left(x + w^t y\right)^j = \sum_{t,j} a_j \left(\sum_l x^{j-l} w^{t \cdot l} y^l \binom{j}{l}\right) \\ &= \sum_j a_j \sum_l x^{j-l} y^l \binom{j}{l} \sum_t w^{t \cdot l} \\ &= \sum_j a_j x^j \cdot d = f(x) \cdot d \end{aligned}$$

□

For example consider a linear function over $\mathbb{F}_q$, then it $w^2 = 1 \Rightarrow w = -1$ and it clears that $f(x) = \frac{1}{2}\left(f(x-y) + f(x+y)\right)$.

Claim 6. *Suppose that $f : \mathbb{F}_q \to \mathbb{F}_q$ satisfies eq. (2.1) than, ethier f is polyonomial at degree at most d or that f is the zero function.*

Proof. Assume by contradiction that f has more then $d+1$ zeros (w.l.g). Denote them by $x_1, x_2..x_{d+1}$. We will show that for any $x \in \mathbb{F}_q$ one can find $y \in \mathbb{F}_q$ such that $x + w^t y = x_t$. Those equations compose a linear diophantine system which can reduced to the following system:

$$\begin{bmatrix} w^0 & 0 & \dots & 0 & 0 \\ w^1 & 1 & \dots & 0 & 0 \\ \vdots & \ddots & \vdots & & \\ w^{d-1} & 0 & \dots & 1 & 0 \\ w^d & 0 & \dots & 0 & 1 \end{bmatrix} \begin{bmatrix} y \\ k_2 \\ k_3 \\ \vdots \\ k_d \end{bmatrix} = \begin{bmatrix} x_1 - x \\ x_2 - x \\ \vdots \\ x_d - x \\ x_{d+1} - x \end{bmatrix}$$

The determinant of the above matrix is $w \neq 0$ and therefore there exists a y satisfies the equations. Hence for any x we have $f(x) = \frac{1}{d}\left(f(x_1) + f(x_2) + .. + f(x_{d+1})\right) = 0$. □

Consider the following decoding process. Given a candidate $f : \mathbb{F}_q \to \mathbb{F}_q$, we will set for any point $x \in \mathbb{F}_q$ the plurality of eq. (2.1) over the $y \in \mathbb{F}_q$. Denote the output by g, and we have that:

$$g(x) = \arg_{z \in \mathbb{F}_q} \max \mathbf{Pr}_y \left[\sum_i^d f\left(x + w^i y\right) = z\right]$$

Claim 7. *If f pass the test with probability $\geq 1 - \varepsilon$ then the relative distance between f and g is at most 2ε.*

Proof. Notice that for any x for which there exists z such $\mathbf{Pr}_y\left[\sum_i^d f\left(x + w^i y\right) = z\right] \geq \frac{1}{2}$ it must hold that $g(x) = f(x)$. Denote by A that set and by α the probability to hit such x

(namely $|A|/|\mathbb{F}_q|$. The probability to pass the test can be bounded by:

$$\begin{aligned}1-\varepsilon \leq \mathbf{Pr}\left[f \text{ pass}\right] &= \mathbf{Pr}\left[f \text{ pass} \cap x \in A\right] + \mathbf{Pr}\left[f \text{ pass} \cap x \notin A\right]\\ &\leq 1\cdot\alpha + \frac{1}{2}\cdot(1-\alpha)\\ \Rightarrow &\alpha \geq 1-2\varepsilon\end{aligned}$$

Thus, at most 2ε of the points do not agree with g. □

Claim 8. *Let f be a function that pass the test with probability $1-\varepsilon$ and fix $x \in \mathbb{F}$ then:*

$$\mathbf{Pr}_{u,v}\left[\sum_i^d f\left(x+w^i u\right) = \sum_i^d f\left(x+w^i v\right)\right] \geq 1-2\varepsilon\cdot d$$

Proof. Recall that for any i the random variabile $x+w^i u$ is distributed uniformaly random regardless x. Therefore we have that for any i the following equation holds with probability at least $1-\varepsilon$:

$$f\left(x+w^i u\right) = \sum_j^d f\left(x+w^i u+w^j v\right)$$

Thus, by the union bound, those equastions satisfied simulatnusly for all the i with probability $1-d\cdot\varepsilon$. By reappting on the same arguments but swpping the u and v we have that

$$f\left(x+w^i v\right) = \sum_j^d f\left(x+w^i v+w^j u\right)$$

also happens with probability at least $1-d\cdot\varepsilon$ and therefore, again by the union bound, both of the equations sets hold with probability at least $1-2d\cdot\varepsilon$. But that is exactly the event in which:

$$\sum_i f\left(x+w^i u\right) = \sum_i^d\sum_j^d f\left(x+w^i u+w^j v\right) = \sum_j f\left(x+w^j v\right)$$

□

Claim 9. *For any x the probability over v that $g(x) = \sum_i f(x+w^i v)$ is grater than $1-\Theta(d\cdot\varepsilon)$.*

Proof. Fix arbitrary x and denote by α the probability to hit v such that $g(x) = \sum_i^d f\left(x+w^i v\right)$ and by A the set of v's satisfy the relation. Notice that the probabiliy to pick a pair v, u such that $\sum_i^d f\left(x+w^i v\right) = \sum_i^d f\left(x+w^i u\right)$ is lower than the prbability that either v and u are both belong to A or they both belong to his complentarty. Namely: $\alpha^2+(1-\alpha)^2 \geq 1-2\varepsilon\cdot d$. Hence we get $\alpha \geq \frac{1}{2}\left(1+\sqrt{1-4\varepsilon d}\right) = 1-\Theta(\varepsilon\cdot d)$. □

Claim 10. *For $\varepsilon = \Theta(1/d^2)$ g is a polynomial at degree at most d.*

Proof. Fix $x, y \in \mathbb{F}_q$. From claim 9 with probability at least $1 - \Theta(d \cdot \varepsilon)$ it holds that $g(x + w^j y) = \sum_j^d f(x + w^i v + w^j y)$. Moreover, as $v + w^j y$ distributed uniformly for any j, it follows that one can fix j and has with probability at least $1 - \Theta(\varepsilon d)$ that $g(x) = \sum_i^d (f(x + w^i(v + w^j y)))$. So, the raltion holds for all the j's with probability $1 - \Theta(\varepsilon d^2)$ combining all togeter obtains:

$$g(x) = \sum_{i,j}^{d} f\left(x + w^i\left(v + w^j y\right)\right) = \sum_{j}^{d}\sum_{i}^{d} f\left(x + w^i v + w^j y\right) = \sum_{j}^{d} g\left(x + w^j y\right)$$

With probability $1 - \Theta(\varepsilon \cdot d^2)$. However, the probability is about sampling v such those transimitions are vaild. But as the value of g is independed on the choice of v it's enough to have just a single v to garuntee $g(x) = \sum_j^d g(x + w^j y)$ namely that the probability is positive. So if $\varepsilon = \Theta(1/d^2)$ we have that for any x, y there exists v such section 2.7 is satisfied and therfore g is satisfies eq. (2.1) for any x, y. $\square$

2.7.1 Polynomial Code.

Consider the field $\mathbb{F}_m$ for an arbitrary prime power $m = q^l$ greater than n. The polynomial codes relay on the fact that any two different polynomials in the ring $\mathbb{F}_m[x]$ at degree at most d different by at least $m - d + 1$ points. For example consider a polynomials pair at degree 1, namely two linear straight lines. If they are not identical than they have at most single intersection point, and the disagree on each of the $n - 1$ remaining points.

So by define the code to be the subspace contains all the polynomials at degree at most d, in such way that any codeword is an image of such polynomail encoded by n numbers, one can garntee a lower bound on the code's distance. Formally we define:

Definition 8 (Polynomial Code. [RS60])**.** *Fix $m > n$ to be a prime power and let $a_0, a_1, a_2, \ldots a_n$ distinct points of the field $\mathbb{F}_m = R$ and define the code $C \subset R$ as follows:*

$$C = \{p(a_0), p(a_1), p(a_2), \cdots p(a_n) : p \text{ is polynomial at degree at most } d\}$$

Observe that C is a linear code at length n over the aleph-bet $\mathbb{F}_m$. The following Lemma states the realtion between the maximal degree of the polynomials and the properites of the code.

Lemma 2. *Fix the degree of the polynomial code to be at most d. Then the parameters of the code are $[n, d+1, n-d]$.*

Proof. The dimension of the code equals to the dimension of the polynomials space at degree at most d which is spanned by the monomial base $e_0, e_1, e_2 \ldots e_d = 1, x \ldots x^d$ and therefore is $d + 1$. In addition suppose that f, g are different polynomials i.e $f \neq g$.

Hence $h = f - g$ is a non-0 polynomial at degree at most d and therefore has at most d roots. Namely at most d points in which f equals g and at least $n - d$ in which they disagree. Put in another way the distance between any two different codewords of the code is at least $n - d$. $\square$

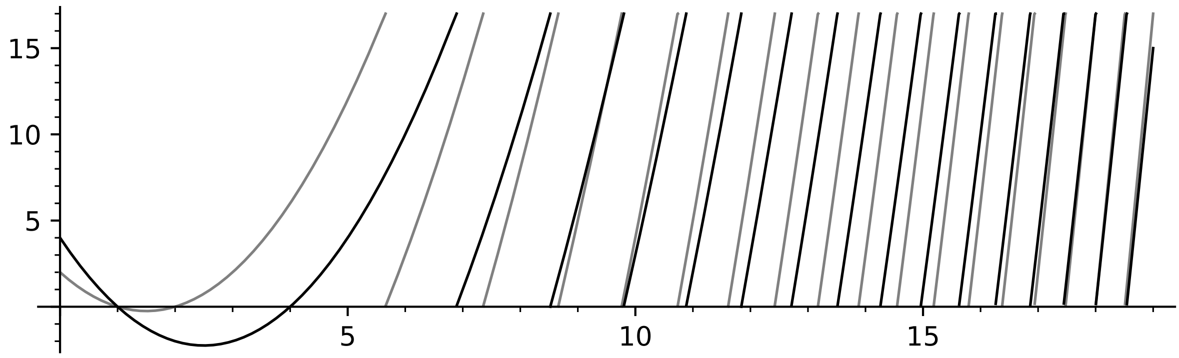

Figure 2.2: The plot $x \mapsto (x-1)(x-2)$ and $x \mapsto (x-1)(x-4)$ presents the extension of the polynomials

Fact 1. *Given $d+1$ points, there is a unique polynomial at degree at most d that pass through all those points. Nevertheless, there is an algorithm G that takes those points as input and outputs the corresponding polynomial.*

Lemma 3 (Testability of Polynomail Code.)**.** *The polynomial code is $(d \cdot \log(n), \varepsilon)$ testable.*

Proof. Denote by $f \in \mathbb{F}_m^n$ a codeword candidate and consider the following test. First, choose uniformly at random $d+1$ points $x_1, x_2, x_3, \ldots x_{d+1}$ and use G to output a polynomial that agree on that points, denote it by $G(f)$. Then uniformly choose an additional point, return **True** if $G(f)(x_{d+2}) = f(x_{d+2})$ and **False** otherwise.

Now, observe that if $f \in C$ then by fact 1 we have that $G(f) = f$. In particular $G(f)(x_{d+2}) = f(x_{d+2})$, Namely the test validate a codeword with probability 1.

Now consider the case that $f \neq C$. And observes that $1 - \frac{1}{n} d(f, C)$ is the maximal fraction η such that there exists a polynomial agree with f on at least ηn points. Then we have:

$$\begin{aligned} \eta &\leq \mathbf{Pr}_{x_1, x_2 \ldots x_d \sim U, x_{d+1} \sim \mathbb{F}} [(Gf)(x) = f(x)] \\ &\leq \mathbf{Pr}[(Gf)(x) = f(x)] \leq \varepsilon \\ 1 - \eta &\geq 1 - \varepsilon \end{aligned}$$

$\square$

Chapter 3

Quantum Error Correction Codes.

3.1 Introduction.

It's widely believed that quantum machines have a significant advantage over the classical in the range of computational tasks[Gro96], [AK99]. Simple algorithms could be interpreted as the quantum version of scanning all the options, cutting the running time by the square root of the classical magnitude.

Nevertheless, Shore has shown a polynomial depth quantum circuit that solves the hidden abelian subgroup [Sho97], which is considered a breakthrough, as it made the computer science community believe that a quantum computer might offer an exponential advantage.

Yet, even though there is a consensus about the superiority of ideal quantum computation model, it is still unclear whether implementing such a machine in the presence of noise is feasible. Still, just pointing on the existence of noise is not powerful enough to cancel the feasibility of computation. Evidence of this is that classical computers also suffer from a certain rate of faults. Thus, to fully understand the hardness, let us compare two main reasons that made it realize a hard task. First is the magnitude of the error rate, classical computers also have errors, and sometimes we witness system failures (blue screen, for example). The error rate of modern computers is so low that the probability for error to propagate stays negligible even if the length of the computation is polynomial in the scale of what is considered reasonable input size. It's worth mentioning that in exascale computing when supercomputers perform around 10^{18} operations per second, It is hard to miss the faults. In quantum, we become aware of their existence much earlier.

The second difference, which is a tricky point, is that quantum states are sensitive to additional types of error. Along with the chance for bit-flip error, a quantum state might also change its phase. For example, consider the initial state $|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$, and suppose that due to noise the state transformed into $\frac{1}{\sqrt{4}}\left(\sqrt{3}\,|0\rangle + |1\rangle\right)$. While classical circuits are blind to such faults. Namely, their run would stay identical as no error occurs. Quantum circuits usually would affect and might fail. Furthermore, when planning a decoder for quantum error correction codes, If one is willing to use a classical code to defend against phase flips, he has to ensure that the decoding doesn't cause bit-flip errors.

3.2 Quantum Noise.

Definition 9 (Bit and pahse flip.). *Consider a quantum state $|\psi\rangle$ encoded in the computation base. We will say that a bit-flip occurs in a scenario the operator Pauli X is applied on* *one of our state's qubits. The bit-flip event could be considered as exactly as the standard bit-flip error in the classical regime. Similarly, phase-flip occurs when the Pauli Z is applied on one of the qubits.*

However, even though quantum noise is so violent, It was proven that any ideal circuit at polynomial depth could be transformed to a robust circuit at poly-logarithmic cost [AB99]. Or in other words, There is a threshold, If the physicists would provide qubits and a finite gate set that suffers from a rate of noise below that threshold, then BQP, the class of polynomial time ideal quantum computation is feasible and could be computed on a realistic machine.

The basic ingredient in [AB99] was to show the existence of quantum error correction code, such that one can perform all the logic operations in a way that restricts present errors from propagating on. That allows them to separate any operation of the computation into stages; one of them is the operation itself, another one is an error correction stage. That process comes with an additional cost, in both space and time, yet it might decrease the probability that the final state at the end would be faulted. The trade-off between the resource needed to pay and the decreasing rate defines the threshold. And if the balance is positive, then one can repeat in a recursion manner, and after log-log iterations, the failure probability decay to zero. At the same time, the circuit would scale at most poly-logarithmic wide and depth factors.

Let's return to the repetition code presented in Chapter 2. We would like to have an analog; a first and natural attempt might consider duplicating copies of the state. Unfortunately, copying a general state is not a linear operation and therefore can not be done in the circuit model (and any other believed to be feasible). In particular there is no circuit U which duplicate simultaneity the states $|0\rangle, |1\rangle, |+\rangle, |-\rangle$.

To overcome the issue, Shor came up with the nine-quibt code [Sho95], which at first glance might seem a naive straightforward implantation of "duplication", but instead uses a clever insight about quantumness in general. Any operation can be seen as a linear (and even unitary) operation over a subspace embedded in large enough dimensions. The encoding is given as follow:

$$|\overline{0}\rangle = \frac{1}{2\sqrt{2}}\left(|000\rangle + |111\rangle\right)^{\otimes 3}$$
$$|\overline{1}\rangle = \frac{1}{2\sqrt{2}}\left(|000\rangle - |111\rangle\right)^{\otimes 3} \ .$$

For convenient let us use the notation, $|\mathbf{GHZ}^{\pm}\rangle = |0^m\rangle \pm |1^m\rangle$. One can also consider the Shor code over m^2 qubits which defined as above beside that any logical state contain m product over m qubits, So the state $|\overline{0}\rangle$ over m^2 qubits can be written as $|\mathbf{GHZ}^{+}\rangle^m$. We are now ready to prove a statement regards to the robustness.

Lemma 4. *The Shor code over* 9 *qubits enable to correct a single either bit or phase flip.*

It is evident that a single bit-flip error can be handled in the same way as in the conventional case. The decoder will check if any of the triples have the same value, and if not, it will correct it by majority. To create a decoder that can also correct a phase-flip error, we need the following statement. In this chapter, we denote the Hadamard gate over m qubits as H^m.

Claim 11. $H^m |\mathbf{GHZ}^{\pm}\rangle = \sum_{x\cdot \mathbf{1} =_2 \pm} |x\rangle$

Proof.

$$H^m |\mathbf{GHZ}^{\pm}\rangle = H^m |0^m\rangle \pm H^m |1^m\rangle = \sum_{x\in\mathbb{F}_2^m} |x\rangle \pm \sum_{x\in\mathbb{F}_2^m} (-1)^{x\cdot\mathbf{1}} |x\rangle$$
$$= \sum_{x\in\mathbb{F}_2^m} \left(1 \pm (-1)^{x\cdot\mathbf{1}}\right) |x\rangle = \sum_{x\cdot\mathbf{1}=_2\pm} |x\rangle$$

□

Now it is clear how to correct a phase flip. One can apply the Hadamard transform and compute the parity of each triple. By the assumption that only a single phase flip may occur, either all the triples have the same parity or the faulted one has an opposite parity and needs to be corrected. Thus, we obtain an $[[9, 1, 3]]$ quantum error correction code. Asymptotically, this is an $\left[\left[m^2, 1, m\right]\right]$ code.

3.3 CSS Codes.

The Shor code is a specific case of the more general CSS (Calderbank-Shor-Steane) code [CS96]. A family composed by two binary codes C_X, C_Z such that $C_Z^{\perp} \subset C_X$.

Definition 10 (CSS Code)**.** *Let C_X, C_Z classical linear codes such that $C_Z^{\perp} \subset C_X$ define the $Q(C_X, C_Z)$ to be all the code words with following structure:*

$$|\mathbf{x}\rangle := |x + C_Z^{\perp}\rangle = \frac{1}{\sqrt{C_Z^{\perp}}} \sum_{z\in C_Z^{\perp}} |x + z\rangle$$

Clearly, the codewords are all the codewords in C_X which don't belong to $C_Z^{\perp}$ and therefore the dimension of the quantum code is $\dim Q(C_X, C_Z) = \dim C_X - \dim C_{Z^{\perp}} = \dim C_X + \dim C_Z - n$. Yet, it's not stems immediately how one can correct faults. Next, we are going to repeat the decoding process of the Shor code in the general setting of CSS codes.

Lemma 5. *Let C_X, C_Z classical codes such $Q(C_X, C_Z)$ is a CSS code. Let d_X be the minimal weigh of codeword in C_X which is not in $C_Z^{\perp}$, and define by the same way d_Z to be the minimal weight of codeword in C_Z which doesn't belong to $C_X^{\perp}$. Then the distance of $Q(C_X, C_Z)$ equals to $\min d_X, d_Z$. Moreover there is a decoder which correct any fault with weight at most $d/2$.*

Proof. First let us prove the following claim:

Claim 12. *Denote by $H^{\otimes n}$ the Hadamard gate over n qubits. Then for any code C it holds that: $H^n |C^{\perp}\rangle = |C\rangle$*

Proof.

$$\begin{aligned} H^n |C^{\perp}\rangle &= \frac{1}{\sqrt{|C_Z^{\perp}|}} \sum_{z\in C_Z^{\perp}} H^n |z\rangle = \frac{1}{\sqrt{2^n}} \frac{1}{\sqrt{|C_Z^{\perp}|}} \sum_{z\in C_Z^{\perp}} \sum_{y\in\mathbb{F}_2^n} (-1)^{\langle z,y\rangle} |y\rangle \\ &= \frac{1}{\sqrt{2^n}} \frac{1}{\sqrt{|C_Z^{\perp}|}} \sum_{z\in C_Z^{\perp}} \left(\sum_{y\in C_Z} |y\rangle + \text{ other terms} \right) \end{aligned} \tag{3.1}$$

Since the columns of matrix H_Z form a basis for the complementary space $C_Z^{\perp}$, and due to the dimensional theorem and the equivalence between the row rank and column rank of a matrix, we can deduce that $\dim \operatorname{rank} H_Z^{\top} + \dim \ker H_Z = n$, which implies that $|C_Z^{\perp}||C_Z| = 2^n$. Thus the norm of

$$\frac{1}{\sqrt{2^n}} \frac{1}{\sqrt{|C_Z^{\perp}|}} \sum_{z \in C_Z^{\perp}} \sum_{y \in C_Z} |y\rangle = \sqrt{\frac{|C_Z^{\perp}|}{2^n}} \sum_{y \in C_Z} |y\rangle \tag{3.2}$$

equals 1 and the summation over the vectors $y \notin C_Z$ in the inner closure in equation 3.1 must cancel. So we left only with a uniform superposition over the codewords in C_Z, Or in other words $H^n |C^{\perp}\rangle = |C\rangle$. □

Claim 12 states that, when considering CSS codes, pauli X operators can be seen as pauli Z operators in the rotated frame. That it,

$$H^n X^f |C_Z^{\perp}\rangle = \overbrace{H^n X^f H^n}^{Z^f} H^n |C_Z^{\perp}\rangle = Z^f |C\rangle$$

That insight hints a description to decoder for the quantum code. If one knows how to correct errors for each of the classical code C_X, C_Z than he can start by correct the bit flips in using the decoder of C_X, rotate the state by applying the Hadamard transform and then correct, what was before the transformation phase flips and now a bit flips by using the decoder of C_Z. Then in the end applying the Hadamard transform again for backing to the initial computation space. Indeed that decoder correct $\min\{d(C_X)/2, d(C_Z)/2\}$ errors. Yet more work is needed to show that this decoder also correct $d(Q(C_X, C_Z))/2$ errors.

Let us assume the existences of decoders of the classical codes C_X and C_Z, Denoted $D_X : \mathbb{F}_2^n \to C_x$ and D_Z. In particular for any $\xi \in \mathbb{F}_2^n$, $D_X(\xi) = \arg\min_{x \in C_X} |x + \xi|$.

Claim 13. *For any $x_0 \in C_X$ and $z_1 \neq z_2 \in C_z^{\perp}$, D_X correct $|x + z_1 + f\rangle$, $|x + z_2 + f\rangle$ into two different words in C_X.*

Proof. Suppose not, namely there exists $y \in C_X$ such that D_X correct $|x + z_1 + f\rangle$, $|x + z_2 + f\rangle$ into $|y\rangle$. Then we have that for both $i \in \{1, 2\}$ it holds that $d(x + z_i + f, y) \leq d\left(C_Z^{\perp}/2\right)$ and therefore $d(x + z_1 + f, x + z_2 + f) \leq d\left(C_Z^{\perp}\right)$. But

$$\begin{aligned} & d(x + z_1 + f, x + z_2 + f) = |x + z_1 + f + x + z_2 + f| \\ = & |z_1 + z_2| = d(z_1, z_2) \end{aligned}$$

contradiction for the assumption that $z_1, z_2 \in C_Z^{\perp}$. □

We are ready to show step by step the decoding process. Let $P = X^f Z^e$ be an error such that $e, f < d/2$ act on the state $|\mathbf{x}\rangle$. Denote by H_X, H_Z the parity check matrices of C_X, C_Z.Using the commute relation $[X^f, Z^e] = (-1)^{\langle e,f \rangle}$ we have that:

$$|\mathbf{x}\rangle \mapsto^P \frac{1}{\sqrt{|C_Z^{\perp}|}} \sum_{z \in C_Z^{\perp}} X^f Z^e |x + z\rangle = \frac{1}{\sqrt{|C_Z^{\perp}|}} \sum_{z \in C_Z^{\perp}} (-1)^{\langle e,f \rangle} Z^e |x + z + f\rangle$$

Now the decoder computes and stores the syndrome relative to the bits code using the parity check matrix H_X. And apply the inverse gate.

$$\begin{aligned}
&\mapsto^{H_X} \frac{1}{\sqrt{|C_Z^\perp|}} \sum_{z\in C_Z^\perp} (-1)^{\langle e,f\rangle} Z^e \left|x+z+f\right\rangle \left|H_X\left(x+z+f\right)\right\rangle \\
&= \quad \frac{1}{\sqrt{|C_Z^\perp|}} \sum_{z\in C_Z^\perp} (-1)^{\langle e,f\rangle} Z^e \left|x+z+f\right\rangle \left|H_X f\right\rangle \\
&\mapsto^{X^f} \frac{1}{\sqrt{|C_Z^\perp|}} \sum_{z\in C_Z^\perp} Z^e \left|x+z\right\rangle \left|H_X f\right\rangle
\end{aligned}$$

Then rotating into the phases base:

$$\begin{aligned}
&\mapsto^{H^{\otimes n}} \frac{1}{\sqrt{|C_Z^\perp|}} \sum_{z\in C_Z^\perp} X^e H^{\otimes n} \left|x+z\right\rangle \\
&= \frac{1}{\sqrt{|C_Z^\perp|}} \sum_{z\in C_Z^\perp} X^e H^{\otimes n} X^x \left|z\right\rangle = \frac{1}{\sqrt{|C_Z^\perp|}} \sum_{z\in C_Z^\perp} X^e Z^x H^{\otimes n} \left|z\right\rangle \\
&= \frac{1}{\sqrt{|C_Z^\perp|}} X^e Z^x H^{\otimes n} \sum_{z\in C_Z^\perp} \left|z\right\rangle = \frac{1}{\sqrt{|C_Z|}} X^e Z^x \sum_{z\in C_Z} \left|z\right\rangle \\
&= \frac{1}{\sqrt{|C_Z|}} (-1)^{\langle x,e\rangle} Z^x \sum_{z\in C_Z} X^e \left|z\right\rangle = \frac{1}{\sqrt{|C_Z|}} (-1)^{\langle x,e\rangle} Z^x \sum_{z\in C_Z} \left|z+e\right\rangle
\end{aligned}$$

So now we have back into the begging. Only now the phase flips are playing the role of bit flips relative to the code C_Z.

$$\begin{aligned}
&= \frac{1}{\sqrt{|C_Z|}} (-1)^{\langle x,e\rangle} Z^x \sum_{z\in C_Z} \left|z+e\right\rangle \left|H_Z e\right\rangle \\
&\mapsto \frac{1}{\sqrt{|C_Z|}} (-1)^{\langle x,e\rangle} Z^x \sum_{z\in C_Z} \left|z\right\rangle \left|H_Z e\right\rangle \mapsto^{H^n} \frac{1}{\sqrt{|C_Z^\perp|}} X^x \sum_{z\in C_Z^\perp} \left|z\right\rangle \\
&= \frac{1}{\sqrt{|C_Z^\perp|}} \sum_{z\in C_Z^\perp} \left|z+x\right\rangle = \left|\mathbf{x}\right\rangle
\end{aligned}$$

□

There is still one big difference between the classic repetition code and the Shor code. While each parity check of the Shor code examines a square root number of qubits, any check of the repetition code touches no more than a constant number of qubits; that is, any check just tests if any two adjacent bits are equal. That brings us to ask whether the Shors code is really the quantum analogy for the repetition code?

For getting an hint before formally presenting a quantum LDPC code, let's take another look on the general structure of the CSS codes. The decoding procedure the proof above teach us an additional point about CSS code, the task of finding a good code quantum code, could be reduce for finding a two classic binary linear codes which their parity check matrices ortogonal to each other. Furthermore, if one is willing to has an qLDPC code, then H_X and H_Z can't be parity check matrices of good classical code as any column of $H_z^\top$ is a codeword of C_X.

$$C_Z^\perp \subset C_X \Rightarrow H_X H_Z^\top = 0$$

And by being an LDPC code, the rows wights of H_Z is bounded by constant. Therefore there is a codeword $\in C_X$ which is also a row of H_Z that has a constant weight.

3.4 qLDPC Codes.

As exactly as in the classic case, qLDPC codes are codes in which any check act non trivially on at most a constant number of qubits, It was proved that using a good Quantum LDPC code one can achieve a fault tolerance threshold theorem at the cost of only constant overhead[1] [Got14]. We are now about to embark on a detailed review of the first quantum LDPC code [Den+02].

Recall that one way to present a code is by define the parity check matrix, Consider the $l \times l$ Tours, namely the Cayley graph of the group product $\mathbb{Z}_l \times \mathbb{Z}_l$. Associate any coordinate (bit/qubit) with an edge on the Tours. And consider the following two restrictions:

1. Each vertex requires form it's local view, the bits lay on his supported edges, To has an even party.

2. Each face, requires the almost the same from it's supported edges but that the face compute the parity in different (specific) base. That it, the face is first rotate the qubits by apply the Hadamard transform on them, and than computes their xor. Finally, rotate back the qubits to the computation base.

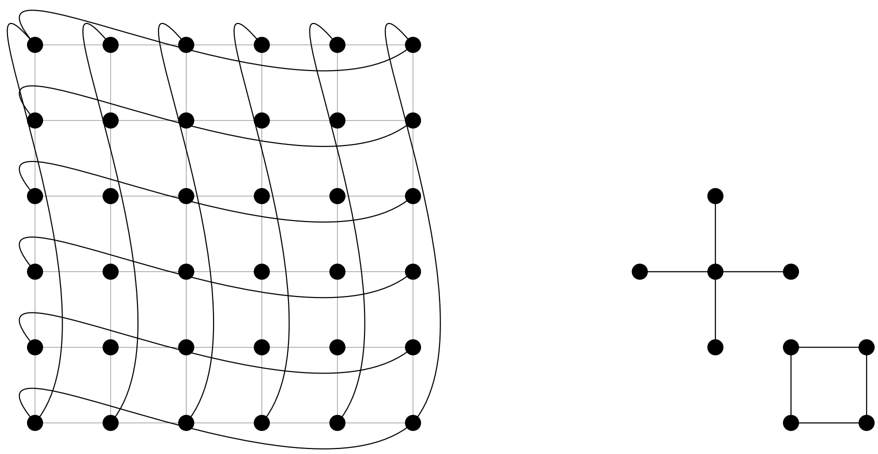

Figure 3.1: Toric Graph.

[1]under the assumption of holding an efficient decoder.

For example consider some vertex v on the Torus, and let $|\psi\rangle = \sum_x |\cdots x_{e_0} x_{e_1} x_{e_2} x_{e_3} \cdots\rangle$ when e_0, e_1, e_2, e_3 are the edges compose the local view of v. Then in any ket can be in the support of $|\psi\rangle$ only if the parity of e_0, e_1, e_2, e_3 is even.

3.5 Quantum Expander Codes.

As similar to the classical case, the next natural question to ask is whether there are codes with positive rates. The quantum expanders were the first quantum LDPC codes to achieve a square-root distance and positive rate [TZ14; LTZ15]. The leading insight was the idea that the Toric code could be represented as a variant product of the repetition code. For example, consider the cross restriction in Figure fig. 3.1; that restriction can be obtained by gluing two vertices of two different cycle graphs.

Definition 11. *For any two matrices A, B, with the same number of rows, denote by $[A, B]$ the matrix obtained by attach B next to A from right. Let $H_1, H_2 \in \mathbb{F}_2^{n\times r}$ be the parity check matrices. Define the bit and the phase parity checks matrices to be:*

$$H_X = \begin{bmatrix} H_1 \otimes I_r & I_n \otimes H_2^\top \end{bmatrix}$$
$$H_Z = \begin{bmatrix} I_r \otimes H_2 & H_1^\top \otimes I_n \end{bmatrix}$$

The matrices are orthogonal to each other as $H_X H_Z^\top = H_1 \otimes H_2^\top + H_1 \otimes H_2^\top = 0$ and therefore the pair define a valid CSS code. We will call to that code the Hyperproduct and denote it by $Q(H_1 \times H_2)$.

Obliviously, if $H_1, H_2, H_1^\top, H_2^\top$ are parity checks matrices of an LDPC codes, so are H_X, H_Z as their maximal row weight is at most two times larger.

Example 3. *The Toric code could be thought as the Hyperproduct of the repetition code with himself. The parity check matrices of the codes are given follow. The left 3×3 matrix corresponds to the repetition code while the right 18×9 corresponds to the vertices check of the Toric code.*

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

Claim 14. *Let $A, B \in \mathbb{M}_{n\times r_1}, \mathbb{M}_{n\times r_2}$ then $\dim\ker[A, B]$ is $\dim\ker A + n$.*

Proof. The proof is omitted. $\square$

Claim 15. *Let k_1, k_2 be the demission of the codes with the full rank parity check matrices H_1, H_2. Then the dimension of the Hyperproduct code is $\geq k_1 k_2$.*

Proof. We will find the dimensions of each of the classical codes defined by $\ker H_X$ and $\ker H_Z$. Notice that the length of the $H \otimes I_n$ equals $n \times n = n^2$, And assuming the fullness of the ranks, the length of $I_{r_1} \otimes H_2^\top$ is $r_1 \cdot r_2$. Thus the length of $\ker H_X$ is $n^2 + r_1 r_2$ Now, recall that for any matrix A it holds that $\dim \ker (A \otimes I_l) = l \cdot \dim \ker A$. Therefore using claim 14 we obtain that the dimension of $\ker H_X$ is $k_1 n + r_1 r_2$

By the same arguments we have that $\dim C_Z = k_2 n + r_1 r_2$ Thus the dimension of the quantum code is:

$$\begin{aligned}\dim Q\left(C_X, C_Z\right) &= \dim C_X + \dim C_Z - \left(n^2 + (n-k_1)(n-k_2)\right) \\ &= \left(k_1 + k_2\right) n + 2(n-k_1)(n-k_2) - \left(n^2 + (n-k_1) \cdot (n-k_2)\right) \\ &= k_1 k_2\end{aligned}$$

□

Remark 1. *Let H_1' be a parity check matrix obtained by puncturing columns from H_1, denote by k_1' the dimension of that code. Then the Hyperproduct $Q(H_1' \times H_2)$ is a CSS code with dimension $k_1' k_2$. Moreover if the number of columns left after the puncturing is less the distance of $\ker H_1$ then it must to holds that $k_1' = 0 \Rightarrow \dim Q(H_1' \times H_2) = 0$. Otherwise, one can take a non trivial codeword of $\ker H_1'$ and extending it to a valid codeword of $\ker H_1$ by set any punctured coordinate of it to zero. The yielded codeword has weight less than d which is contradiction.*

Claim 16. *Denote by d the minimal distance of $\ker H_1$. Any codeword x of $C_X = \ker H_X$ with edge at most d belongs to $C_Z^\perp$.*

Proof. Define by H_1' the matrix which obtained by puncturing from H_1 the columns associated with the coordinates e_i such that subspace corresponding to $e_i \otimes I_n$ doesn't support x. Denote the by S the set of the reaming coordinates. For example if $H_1 \otimes I_n = I_2 \otimes I_2$ and $x = [1, 0, 0, 0]$ then H_1' is the unit matrix $[1, 0]^\top$ obtained by puncturing the second column of $H_1 = I_2$, and $S = \{e_1\}$.

As $|x| < d$ we have that $|S| < d$, Namely H_1' supported on less than d coordinates and therefore $\dim Q\left(H_1' \times H_2\right) = k_1' k_2 = 0$. Thus, by the fact that for any CSS code $\dim C = \dim C_X - \dim C_Z^\perp$ it follows that $\dim \ker H_X' = \dim \ker H_Z'^\perp \Rightarrow \ker H_X' = \ker H_Z'^\perp$. Denote by x' the restriction of x to the columns of H_X' and clearly, by the definition of the construction, x' is a codeword of $\ker H_X'$. Thus x' is also codeword of $\ker H_Z'^\perp$ and by the same argument, x is also a codeword of $\ker H_Z'^\perp$. □

Immediately from claim 16 we obtain that existences of quantum LDPC codes with positive rate and $\Theta\left(\sqrt{n}\right)$ distance by taking the Hyperproduct of two classical expender codes.

Theorem 1. *There exists an infinity family of QLDPC codes with positive rate and $\Theta(\sqrt{n})$ distance.*

Chapter 4

Good qLDPC and LTC.

[LZ22], [PK21], [Din+22] We are now ready to present the Good qLDPC codes.

Definition 12. *Let $C = \mathcal{T}(G, C_0)$. We say that $x \in C_\oplus$ is **reducible** if there exists a vertex v and a small codeword c_v, for which, adding the assignment of c_v over the v's edges to x decreases the weight. Namely, $|x + c_v| < |x|$. If $x \in C_\oplus$ is not a reducible codeword then we say that x is **irreducible** .*

Definition 13 (w-Robustness)**.** *Let C_0 be code of length Δ with minimum distance $\delta_0\Delta$. $C = C_0 \otimes \mathbb{F} + \mathbb{F} \otimes C_0^\perp$ will be said w-robust if any codeword $c \in C$ at weight less than w it follows that c is supported on at most $2 \cdot w/\delta_0\Delta$ rows and cols.*

Definition 14 (p-Resistance to Puncturing.)**.** *Let p, w be integers. We will say that the dual tensor code $C_A \otimes \mathbb{F} + \mathbb{F} \otimes C_B$ is w-robust with p-resistance to puncturing, if the code obtained by removing (puncturing) a subset of at most p rows and columns is w-robust.*

Theorem 2. *For $\varepsilon \in \left(0, \frac{1}{2}\right)$, $\gamma \in \left(\frac{1}{2} + \varepsilon, 1\right)$, $\delta_0 > 0$, large enough Δ, and small codes C_A, C_B with distance at least $\delta_0\Delta$ if the dual tensor code of C_A, C_B is w-robust and Δ^γ resistance to puncturing, then there exists an infinite family of square complexes for which the Tanner code defined by the complexes and the dual tensor code such that any codeword with weigh less than $\frac{\delta_0 n}{4\Delta^{3/2+\varepsilon}}$ is reducible definition 12.*

Claim 17. *Consider a codeword c such there exists a negative vertex v that adjoins only for normal vertices. Then c is reducible, and there exits $y \in C_A^\perp \otimes C_B^\perp$ such that y supported only on v and $|c_v + y| < |c_v|$.*

Proof. Any row in the local view of v can be thought as a codeword of C_B plus faults made by adding $C_A \otimes \mathbb{F}_2$ to the corresponding sibling of v regarding the row. As the dual tensor is $\Delta^{3/2+\varepsilon}$ robust. Then, [KP22] □

Claim 18. *The distance of the dual tensor code is at least $\delta\Delta$.*

Proof. By the robustness any codeword with weight less than $\delta_0\Delta$ supported on at most one row. Now as the multiplication with any code word of $C_A^\perp \otimes C_B^\perp$ can be decomposed to a pairwise multiplication over the rows We obtain that only non trivial row must to be in $\left(C_B^\perp\right)^\perp$ and there fore at weight at least $\delta_0\Delta$ and that is contradiction. □

Claim 19. *for any $\varepsilon \in (0, 1)$ and large enough Δ it holds that $|S| \leq \Delta^\varepsilon |T|$*

Proof. Suppose not, namely that $|S| > \Delta^{\varepsilon}|T|$, then $|x|/|T| > \Delta^{\varepsilon}|x|/|S| > \Delta^{\varepsilon} \cdot \delta_0 \Delta$ But:

$$\frac{|x|}{|T|} = \frac{\Theta\left(E(T,T)\right)}{|T|} \leq \Theta(\Delta^2)\frac{|T|}{n} + \Theta(\Delta) \to_{n\to\infty} \Theta(\Delta)$$

□

Claim 20. *There is a negative vertex which adjoins to only normal vertices.*

Proof. Denote by S_e, S_n the exceptional and the normal vertices. Namely the weight of the local view for any vertex in S_e is grater than $\Delta^{3/2+\varepsilon}$, and the normal vertices is the complementary vertices set. Suppose through contradiction that any negative vertex $v_- \in V_-$ has at least one sibling in S_e. Therefore $|T| \leq \Delta|S_e|$ combining with claim 19 it follows that $|S| \leq \Delta^{1+\varepsilon}|S_e|$:

$$\begin{aligned}\Delta^{3/2+\varepsilon} \leq \frac{E(S,S_e)}{|S_e|} &= \Theta\left(\Delta^2\right)\frac{|S|}{n} + \Theta\left(\Delta\right)\sqrt{\frac{|S|}{|S_e|}} \\ &\leq \Theta(\Delta^2)\frac{|S|}{n} + \Theta(\Delta)\Theta\left(\Delta^{\frac{1+\varepsilon}{2}}\right)\end{aligned}$$

□

$$\begin{aligned}\alpha|S_e| \leq \quad & \theta^2\tfrac{|S_e||S|}{|V|} - \lambda^2\sqrt{|S_e||S|} \\ \beta|T| \leq \quad & 2\theta\tfrac{|T||S|}{2|V|} + 2\lambda\sqrt{|T||S|}\end{aligned} \Rightarrow \begin{aligned}|S_e| \leq \quad & \left(\alpha - \theta^2\tfrac{|S|}{|V|}\right)^{-2}\lambda^4|S| \\ |T| \leq \quad & \left(\beta - 2\theta\tfrac{|S|}{2|V|}\right)^{-2}4\lambda^2|S|\end{aligned}$$

$$\begin{aligned}&\Delta^{1\frac{1}{2}+\varepsilon}|T| \leq |x| \leq \Delta^{1\frac{1}{2}}|S| \\ &|T|\Delta^{\varepsilon} \leq |S|\end{aligned}$$

Definition 15 (The Disagreement Code)**.** *Given a Tanner code $C = \mathcal{T}(G, C_0)$, define the code $C_\oplus$ to contain all the words equal to the formal summation $\sum_{v\in V(G)} c_v$ when c_v is an assignment of a codeword $c_v \in C_0$ on the edges of the vertex $v \in V(G)$. We call to such code the* ***disagreement code*** *of C, as edges are set to 1 only if their connected vertices contribute to the summation codewords that are different on the corresponding bit to that edge. In addition, we will call to any contribute c_v, the* ***suggestion*** *of v. And notice that by linearity, each vertex suggests, at most, a single suggestion.*

Finally, given a bits assessment $x \in \mathbb{F}_2^E$ over the edges of G, we will denote by $x^\oplus \in C_\oplus$ the codeword which obtained by summing up suggestions set such each vertex suggests the closet codeword to his local view. Namely, for each $v \in V$ define:

$$\begin{aligned}c_v &\leftarrow \arg_{\tilde{c}\in C_0}\min d(x|_v, \tilde{c}) \quad \forall v \in V \\ x^\oplus &\leftarrow \sum_{v\in V} c_v\end{aligned}$$

We will think about $x^\oplus$ as the disagreement between the vertices over x.

Lemma 6 (Linearity Of The Disagreement)**.** *Consider the code $C = \mathcal{T}(G, C_0)$. Let $x \in \mathbb{F}_2^E$ then for any $y \in C$ it holds that:*

$$(x+y)^\oplus = (x)^\oplus$$

Proof. Having that $y \in C$ followes $y|_v \in C_0$ and therefore $\arg_{\tilde{c}\in C_0} \min d(z,\tilde{c}) = y|_v + \arg_{\tilde{c}\in C_0} \min d(z,\tilde{c}+y|_v)$, Hence the suggestion made by vertrx v is:

$$\begin{aligned} c_v \leftarrow & \arg_{\tilde{c}\in C_0} \min d((x+y)|_v,\tilde{c}) \\ \leftarrow & y|_v + \arg_{\tilde{c}\in C_0} \min d((x+y)|_v,\tilde{c}+y|_v) \\ \leftarrow & y|_v + \arg_{\tilde{c}\in C_0} \min d(x|_v,\tilde{c}) \end{aligned}$$

It follows that:

$$\begin{aligned} (x+y)^{\oplus} = & \sum_{v\in V} c_v = \sum_{v\in V} y|_v + \sum_{v\in V} \arg_{\tilde{c}\in C_0} \min d(x|_v,\tilde{c}) \\ = & y^{\oplus} + x^{\oplus} = x^{\oplus} \end{aligned}$$

When the last transition follows immediately by the fact that $y \in C$ and therefore any pair of connected vertices contribute the same value for their associated edge □

4.1 Decoding and Testing

For completeness, we show exactly how Theorem 1 implies testability. The following section repeats Leiverar's and Zemor's proof [LZ22]. Consider a binary string x that is not a codeword. The main idea is the observation that the number of bits filliped by (any) decoder, while decoding x, bounds the distance $d(x,C)$ from above. In addition, the number of positive checks in the first iteration is exactly the number of violated restrictions.

Definition 16. *Let $L = \{L_i\}_0^{2|E|}$ be a series of $2|E|$. Such that for each vertex $v \in V$ $\sum_{e=\{u,v\}} L_{e_v} \in C_0$. We will call L a Potential list and refer to the e_v'the element of L as a suggestion made by the vertex $v \in V$ for the edge $e \in E$. Sometimes we will use the notation L_v to denote all the L's coordinates of the form $L_{e_v} \forall e \in$ Support(v). Define the Force of L to be the following sum $F(L) = \sum_{e=\{v,u\}\in E} (L_{e_v} + L_{e_u})$ and notice that $F(L) \in C_{\oplus}$. And define the state $S(L) \subset \mathbb{F}_2^{|E|}$ of L as the vector obtained by choosing an arbitrary value from $\{L_{e_v}, L_{e_u}\}$ for each edge $e \in E$.*

Claim 21. *Let L be the Potential list. If $F(L) = 0$ then $S(L) \in C$.*

Proof. Denote by $\phi(e) \subset \{L_{e_v}, L_{e_u}\}$ the value which was chosen to $e = \{v,u\} \in E$. By $F(L) = 0$, it follows that $L_{e_v} + L_{e_u} = 0 \Rightarrow L_{e_v} = L_{e_u} = \phi(e)$ for any $e \in E$. Hence for every $v \in V$ we have that $S(L)|_v = \sum_{u\sim v} \phi(\{v,u\}) = \sum_{u\sim v} L_{e_v} \in C_0 \Rightarrow S(L) \in C$ □

The decoding goes as follows. First, each vertex suggests the closet C_0's codeword to his local view. Those suggestions define a Potential list, denote it by L, then if $F(L) < \tau$, by Theorem 1, one could find a suggestion of vertex v and a codeword c_v such that updating the value of $L_v \leftarrow L_v + c_v$ yields a Potential list with lower force. Therefore repeating the process till the force vanishes, obtain a Potential list in which its state is a codeword.

Definition 17. *Let $\tau > 0, f : \mathbb{N} \to \mathbb{R}^+$, and consider a Tanner Code $C = \mathcal{T}(G, C_0)$. Let us Define the following decoder and denote it by $\mathcal{D}$.*

Data: $x \in \mathbb{F}_2^n$
Result: $\arg\min \{y \in C : |y + x|\}$ if $d(y, C) < \tau$ and False otherwise.

1 $L \leftarrow \text{Array}\{\}$
2 **for** $v \in V$ **do**
3     $c'_v \leftarrow \arg\min \{y \in C_0 : |y + x|_v|\}$
4     $L_v \leftarrow c'_v$
5 **end**
6 $z \leftarrow \sum_{v \in V} c'_v$
7 **if** $|z| < \tau \frac{n}{f(n)}$ **then**
8   **while** $|z| > 0$ **do**
9   find v and $c \in C_0$ such that $|z + c_v| < |z|$
10         $z \leftarrow z + c_v$
11         $L_v \leftarrow L_v + c_v$
12   **end**
13 **else**
14   reject.
15 **end**
16 **return** $S(L)$

Algorithm 1: Decoding

Theorem 3. *Consider a Tanner Code $C = [n, n\rho, n\delta]$ and the disagreement code $C_\oplus$ defined by it. Suppose that for every codeword $z \in C_\oplus$ in $C_\oplus$ such that $|z| < \tau' n / f(n)$, there exists another codeword $y \in C_\oplus$ such that $|y| < |z|$, set $\tau \leftarrow \frac{\tau'}{6\Delta}\delta$ then,*

1. *$\mathcal{D}$ corrects any error at a weight less than $\tau n / f(n)$.*
2. *C is $f(n)$ testable code.*

Proof. So it is clear from the claim claim 21 above that if the condition at line (6) is satisfied, then $\mathcal{D}$ will converge into some codeword in C. Hence, to complete the first section, it left to show that $\mathcal{D}$ returns the closest codeword. Denote by e the error, and by simple counting arguments; we have that $\mathcal{D}$ flips at most:

$$d_{\mathcal{D}}(x, C) \leq 2|e|\Delta + \tau \frac{n}{f(n)} \Delta$$

bits. Hence, by the assumption,

$$d_{\mathcal{D}}(x, C) \leq 3\Delta\tau \frac{n}{f(n)} \leq 3\Delta\tau\delta n < \frac{1}{2}\delta n$$

Therefore the code word returned by $\mathcal{D}$ must be the closet. Otherwise, it contradicts the fact that the relative distance of the code is δ. To obtain the correctness of the second section, we will separate when the conditional at the line (5) holds and not. And prove that the testability inequality holds in both cases. Let $x \in \mathbb{F}_2^n$ and consider the running of $\mathcal{D}$ over x. Assume the first case, in which the conditional at line (5) is satisfied. In that case, $\mathcal{D}$ decodes x into its closest codeword in C. Therefore:

$$\begin{aligned} d(x, C) &\leq d_D(x, C) \leq m\xi(x)\Delta + |z|\Delta \\ &\leq m\xi(x)\Delta + m\xi(x)\Delta^2 \\ \frac{d(x, C)}{n} &\leq \kappa_1 \xi(x) \end{aligned}$$

Now, consider the other case in which: $|z| \geq \tau \frac{n}{f(n)}$.

$$\begin{aligned}\frac{d\left(x,C\right)}{n} &\leq 1 \leq \frac{|z|}{\tau n} f\left(n\right) \leq \frac{m}{n}\frac{1}{\tau}\Delta\xi\left(x\right)f\left(n\right)\\ &\leq \kappa_2 \xi\left(x\right)f\left(n\right)\end{aligned}$$

Picking $\kappa \leftarrow \max\{\kappa_1, \kappa_2\}$ proves $f\left(n\right)$-testability $\square$

Chapter 5

First Attempt For Good qLTC.

5.1 The Polynomial-Code Is Not w-Robust.

One idea for constructing is to use the polynomial code instead of C_0. This follows from the fact that if one picks a degree strictly greater than $\Delta/2$, then $C_0^{\perp} \subset C_0$ and therefore one could choose C_z to be the same code defined on the negative vertices of the graph.

Here we prove that the dual-tensor code, in that case, is not w-robust, meaning that any such construction should be considered another way of proving the Reduction Lemma.

Claim 22. *Let C_0 be the $[\Delta, d, \Delta - d]$ polynomial code. Then any code word in $\left(C_0^{\perp} \otimes C_0^{\perp}\right)^{\perp}$ is a polynomial in $F[x, y]$ at degree at most $\Delta + d$*

Proof. Consider base element $C_0 \otimes \mathbb{F}$, denote it by $c = g_i \otimes e_j$. And notice that c has representation in $F[x, y]$ of $\prod_{y' \neq j} (y - y') g_i(x)$. By the fact that $g_i(x) \in C_0$ we have that degree of c is at most $\Delta + \delta$. Hence any element in the subspace of $C_0 \otimes \mathbb{F}$ is a polynomial at degree at most $\Delta + d$. □

Claim 23. *The dual-tensor polynomial code is not w-robust.*

Proof. Consider the following polynomial

$$P(x, y) = \prod_{i \neq \Delta - 1} (x + iy) = \prod_{i \neq 1} (x - iy)$$

The degree of any monomial is at most $\Delta - 1$, Thus it clear that $P \in \left(C_0^{\perp} \otimes C_0^{\perp}\right)^{\perp}$. And by the fact that for any $x \neq y$ there exists $i \neq 1$ such that $x = iy$ we have that $P(x, y) = 0$. Hence the weight of P is at most $|\{(x, y) : x = y\}| = \Delta$. Yet, for any $x = y$ it follows:

$$P(x, x) = \prod_{i \neq \Delta - 1} (x + ix) = x^{\Delta - 1} \prod_{i \neq \Delta - 1} (1 + i) = (\Delta - 1)! =_{\Delta} -1 \neq_{\Delta} 0$$

Put it differently, the diagonal of the matrix has only non-zero values, therefore the P is supported on the entire collection of rows and columns. Namely, we found a codeword in the dual tensor code at weight less than $\Delta^{1\frac{1}{2}}$ which is not restricted to at most $\Delta^{1\frac{1}{2}}/\delta_0 \Delta$. So, the dual-tensor polynomial code is not a w-robust code, for $w = \Delta^{1\frac{1}{2}}$. □

Chapter 6

Local Majority $\neq$ Local Testability.

Claim 24. *Suppose that G is an expander graph with a second eigenvalue λ, then For any layer U there exist a layer U' such that:*

$$\begin{aligned}&(1)\quad |U'| \geq |U|\\&(2)\quad w_{E/E'}\left(x|_{U'}\right) \geq \Delta|U'|\left(\delta_0 - \frac{2}{3} - \frac{2\lambda}{\Delta}\right)\end{aligned}$$

Proof. Consider layer U and denote by U_{-1} and U_{+1} the preceding and the following layers to U in T. It follows from the expander mixing lemma that:

$$\begin{aligned}w_{E/E'}\left(x|_U\right) \geq \delta_0\Delta|U| - w\left(E(U_{-1}\bigcup U_{+1}, U)\right) \geq\\ \delta_0\Delta|U| - E(U_{-1}\bigcup U_{+1}, U)\\ \delta_0\Delta|U| - \Delta\frac{|U||U_{-1}|}{n} - \Delta\frac{|U||U_{+1}|}{n}\\ - \lambda\sqrt{|U||U_{-1}|} - \lambda\sqrt{|U||U_{+1}|}\end{aligned}$$

Claim 25. *We can assume that $|U| \geq |U_{-1}|, |U_{+1}|$.*

Proof. Suppose that $|U_{+1}| > |U|$, so we could choose U to be U_{+1}. Continuing stepping deeper till we have that $|U| > |U_{+1}|, |U_{-1}|$. Simiraly, if $|U| > |U_{+1}|$ but $|U_{-1}| > |U|$, the we could take steps upward by replacing U_{-1} with U. At the end of the process, we will be left with U at a size greater than the initial layer and $|U| > |U_{+1}|, |U_{-1}|$ □

Using claim 25, we have that $\left(|U_{+1}| + |U_{-1}|\right)/n < \frac{2}{3}$ and therefore:

$$w_{E/E'}\left(x|_U\right) \geq \left(\delta_0 - \frac{2}{3} - \frac{2\lambda}{\Delta}\right)\Delta|U|$$

□

That immediately yields the following: let $U_{\max} = \arg\max_{U \text{ layer in } T} |U|$ then:

$$|x| \geq w_{E/E'}\left(x|_{U_{\max}}\right) \geq \left(\delta_0 - \frac{2}{3} - \frac{2\lambda}{\Delta}\right)\Delta|U_{\max}|$$

Claim 26. *Consider again the maximal layer $U_{\max}$ then:*

$$w_{E/E'}\left(x\right) \geq \left(\delta_0 - \frac{|U_{\max}|}{n} - \frac{\lambda}{\Delta}\right)\Delta|T|$$

Proof. Similarly to above, now we will bound the flux that all the nodes in T induce over E/E'. Denote by $U_0, U_1..U_m$ the layers of T ordered corresponded to their height, thus we obtain:

$$\begin{aligned}
w_{E/E'}(x) \geq \delta_0\Delta|T| - \sum_{i\in[m]} w\left(E\left(U_i, U_{i+1}\right)\right) \\
\geq & \delta_0\Delta|T| - \sum_{i\in[m]} E\left(U_i, U_{i+1}\right) \\
\geq & \delta_0\Delta|T| - \sum_{i\in[m]} \frac{\Delta}{n}|U_i||U_{i+1}| + \lambda\sqrt{|U_i||U_{i+1}|} \\
\geq & \delta_0\Delta|T| - \sum_{i\in[m]} \frac{\Delta}{n}|U_i||U_{i+1}| + \lambda\frac{|U_i| + |U_{i+1}|}{2} \\
\geq & \delta_0\Delta|T| - \frac{\Delta}{n}|T||U_{\max}| - \lambda|T| \\
\geq & \left(\delta_0 - \frac{|U_{\max}|}{n} - \frac{\lambda}{\Delta}\right)\Delta|T|
\end{aligned}$$

□

Claim 27. *Consider $G = (V, E)$ a Δ-ramunjan graph and let U be a subset of V such that $|U| \geq \frac{1}{9}n$ then, there is must to be at least one vertex in U such the number of closed loops pass through it, is less than $\sqrt{\Delta} \cdot n$.*

Claim 28. *Alternate proof of fulx inequality, which dosn't assume that there is no interference inside the layers.* $w\left(E\left(U, U\right)\right) > 0$.

Proof. Separeate into the following cases, First assume that $|U_{\max}|/n > \frac{1}{3}$ then we have that the total interference with $U_{\max}$ layers is at most:

$$\begin{aligned}
&\frac{\Delta|U_{\max}|\left(n - |U_{\max}|\right)}{n} + \lambda\sqrt{|U_{\max}|n} \leq \left(1 - \frac{|U_{\max}|}{n} + \sqrt{3}\frac{\lambda}{\Delta}\right)\Delta|U_{\max}| \\
&\leq \left(\frac{2}{3} + \sqrt{3}\frac{\lambda}{\Delta}\right)\Delta|U_{\max}|
\end{aligned}$$

And therefore we have that the flux induced by $U_{\max}$ is at least:

$$\left(\delta_0\Delta - \frac{2}{3} + \sqrt{3}\frac{\lambda}{\Delta}\right)\Delta|U_{\max}|$$

So it lefts to consider the case in which for every layer it holds that $|U_{\max}| \leq \frac{1}{3}n$. At that case we count the fulx induced by the whole three T which is what exactly we have prove in **??** minus the inner interference at the tree, That it we need only to subtract $\sum \frac{\Delta|U_i|^2}{n} + \lambda|U_i| \leq \left(\frac{|U_{\max}|}{n} + \lambda/\Delta\right)|T|$ So we obtained that in that case:

$$w_{E/E'}(x) \geq \left(\delta_0 - 2\frac{|U_{\max}|}{n} - 2\lambda/\Delta\right)\Delta|T| \geq \left(\delta_0 - \frac{2}{3} - 2\frac{\lambda}{\Delta}\right)\Delta|T|$$

□

Proof of Theorem 1. Consider the size of the maxiaml layer $|U_{\max}|$ and sepearte to the following two cases. First, consider the case that $|U_{\max}| \geq \alpha n$ in that case it follows immedily by claim 24 that if $\delta_0 > \frac{2}{3} - \frac{2\lambda}{\Delta}$ there exists $\alpha' > 0$ such that:

$$|x| \geq \left(\delta_0 - \frac{2}{3} - \frac{2}{\lambda}\Delta\right)\Delta|U_{\max}| \geq \alpha' n$$

So, it is lefts to consider the second case in which $|U_{\max}| < \alpha n$ in that case, we have from claim 26 inequality that:

$$\begin{aligned}|x| \geq w_{E/E'}(x) &\geq \left(\delta_0 - \frac{|U_{\max}|}{n} - \frac{\lambda}{\Delta}\right)\Delta|T| \\ &\geq \left(\delta_0 - \alpha - \frac{\lambda}{\Delta}\right)\Delta|T|\end{aligned}$$

Setting $\alpha \geq \frac{2}{3}$ we complete the proof $\square$

Unfortunately, Singelton bound doesn't allow both $\delta_0 > \frac{2}{3}$ and $\rho_0 \geq \frac{1}{2}$, so in total, we prove the existence of code LDPC code which is good in terms of testability and distance yet has a zero rate. In the following subsection, we will prove that one can overcome this problem, by considering a variton of Tanner code, in which every vertex cheks only a $\frac{2}{3}$ fraction of the edges in his support.

6.0.1 Overcoming The Vanishing Rate.

Consider the following code; instead of associating each edge with pair of checks, let's define the vertices to be the checks of small codes over $q \in [0, 1]$ fraction of their edges. That is, now each vertex defines only $(1 - \rho_0)\, q\Delta$ restrictions. Hence, the rate of the code is at least:

$$\begin{aligned}\rho\frac{1}{2}\Delta n &\geq \frac{1}{2}\Delta n - (1 - \rho_0)\, q\Delta n \\ \Rightarrow \rho &\geq \left(2\rho_0 + \left(\frac{1}{q} - 2\right)\right) q \\ \rho_0 &\geq 1 - \frac{1}{2q}\end{aligned}$$

for example, if $q = 2/3$, then for having constant rate, it is enough to ensure that $\rho_0 \geq 1 - \frac{3}{4} = \frac{1}{4}$.

Intuition For Testability. Before expand the construction let's us justify why one should even expects that removing constraints preserves testability. Assume that is guaranteed that the lower bound of the flux on the trivial vertices remains up to multiplication by the fraction factor q, or put it differently, one could just stick q in every inequality without lose correctness, Then:

$$\begin{aligned}w_{E/E'}(x|_U) &\geq \delta_0 q\Delta|U| - qw\left(E(U_{-1}\bigcup U_{+1}, U)\right) \\ \Rightarrow |x| &\geq \left(\delta_0 - \frac{2}{3} - \frac{2\lambda}{\Delta}\right) q\Delta|U_{\max}|\end{aligned}$$

As you can see, ireducable [12] words of the disagreement have a linear weight, despite that the original code has non-vanish rate.

Yet, We still require more to prove a linear distance. By repeating on the *Singleton Bound 2.3* proof it follows that the small code $\tilde{C_0}$ obtained by ignoring arbitrary $\left(q - \frac{1}{2}\right)\Delta$ coordinates yield a code with distance:

$$\left(\delta_0 q - \left(q - \frac{1}{2}\right)\right)\Delta$$

So assume that we could engineer an expander family such that the graphs obtained by removing $\frac{1}{2}$ of the edges connected for each vertex result also expanders, and in addition, regarding $\tilde{C_0}$ each edge is checked by both vertices on its support. Namely, a good Tanners Code could be defined on the restricted graphs; Then, any string that satisfies the original checks also has a linear weight. To achieve this property, we will restrict ourselves to a particular family of Cayley Graphs.

Theorem (Theorem 1+)**.** *There exist a constant $\alpha > 0$ and infinite family of codes which satisfies Theorem 1 and also good.*

Definition 18 (Testability Tanner Code)**.** *Let $q > \frac{1}{2}$ and let J be a generator set for group Γ such that $|J| = \Delta$, $q|\Delta$, J closed for inverse, and there exist subset of J, denote it by, J' such that J' is a generator set of Γ and $|J'| = \frac{1}{2}\Delta$. Let C_0 be a code with parameters $C_0 = q\Delta\left[1, \rho_0, \delta_0\right]$. For any vertex associate a subset $\bar{J}_v \subset J/J'$ at size:*

$$|\bar{J}_v| = \left(q - \frac{1}{2}\right)\Delta \Rightarrow |\bar{J}_v \cup J'| = q\Delta$$

Define the code $\mathcal{T}(J, q, C_0)$ to be the subspace such that any vertex's local view over the edges defined by $\bar{J}_v \cup J'$ is a codeword of C_0. In addition, let's associate a code $\tilde{C}_v$ obtained for any vertex by ignoring the bits supported on the $\bar{J}_v$ coordinates. Notice that code defined by requiring that the local view of any vertex v of $\mathrm{Cayley}(\Gamma, J')$ *is a codeword of $\tilde{C}_v$ is a TannerCode. Denote it by $\tilde{\mathcal{T}}(J, q, C_0)$.*

Claim 29. *Let J be defined as above such that both* $\mathrm{Cayley}(\Gamma, J)$, $\mathrm{Cayley}(\Gamma, J')$ *are expanders with algebraic expansion greater then λ and C_0 with the parameters $\rho_0 > 1 - \frac{1}{2q}$ and $\delta_0 q - \left(q - \frac{1}{2}\right) > 2\lambda/\Delta$. Then the code $\mathcal{T}(J, q, C_0)$ is a good code.*

Proof. We already proved that the code has a positive rate, So it left to show a constant relative distance.

Consider a codeword x and denote by x' the restriction of x to *Cayley* (Γ, J') which is a codeword of $\tilde{C} = \tilde{\mathcal{T}}(J, q, C_0)$. But $\tilde{C}$ is a Tanner Code such that any vertex sees at least $\tilde{\delta_0}\Delta := \left(\delta_0 - \left(q - \frac{1}{2}\right)\right)\Delta$ nontrivial bits. Denote by S the vertices subset supports x', and by $E(S, S)$ the edges from S to itself, and by using the fact that *Cayley*(Γ, J') is an expander with second eigenvalue at most δ we have that:

$$\frac{|x'|}{|S|} \geq \tilde{\delta_0}\Delta \Rightarrow |S| \geq \left(\tilde{\delta} - \frac{2\lambda}{\Delta}\right)\Delta n$$

By the assumption that $\tilde{\delta} > 2\lambda/\Delta$ we have that S must has liner size, and therefore $|x'|$ also must to be linear in n. Finally as $x' \subset x$ we obtain the correctness of the claim. □

Claim 30 (Existence of such *Cayley's*)**.** *Let S be a generator set such that* $\mathrm{Cayley}(\Gamma, S)$ *has a second largest eigenvalue greater then λ, And consider an arbitrary group element $g \in \Gamma$ and denote by S_g the set gSg^{-1}. Then the second eigenvalue of the graph obtained by $(\Gamma, S) \cup (\Gamma, S)$ is at most 2λ.*

Proof. Denote by G, G' the *Cayley* graphs corresponding to S, S_g, for convenient we will use the notation of $\sum_{v \sim_G u}$ to denote a summation over all the neighbors of v in the graph G. Let $A_{G'}$ be the adjacency matrix of G'. Recall that G' is a Δ regular graph, and therefore the uniform distribution $\mathbf{1}$ is the eigenstate with the maximal eigenvalue, and the second eigenvalue is given by the min-max principle:

$$\begin{aligned}\max_{f \perp \mathbf{1}} \frac{f^\top A_{G'} f}{f^\top f} &= \max_{f \perp \mathbf{1}} \sum_{v} \sum_{u \sim_{G'} v} \frac{f(u) f(v)}{f^\top f} \\ &= \max_{f \perp \mathbf{1}} \sum_{v} \sum_{\tau \in S} \frac{f\left(g\tau g^{-1} v\right) f(v)}{f^\top f} \\ &= \max_{f \perp \mathbf{1}} \sum_{gv} \sum_{\tau \in S} \frac{f\left(g\tau g^{-1} gv\right) f(gv)}{f^\top f} \\ &= \max_{f \perp \mathbf{1}} \sum_{gv} \sum_{\tau \in S} \frac{f(g\tau v) f(gv)}{f^\top f} \\ &= \max_{f \perp \mathbf{1}} \sum_{gv} \sum_{u \sim_G v} \frac{f(gu) f(gv)}{f^\top f}\end{aligned}$$

As for any function $f : V \to \mathbb{R}$ one could define a function $f' : E \to \mathbb{R}$ such that $f'(v) = f(v)$ and f' preserves the norm:

$$f'^\top f' = \sum_{v \in V} f'(v) f'(v) = \sum_{v \in V} f^\top(vg) f(vg) = f^\top f$$

$$\Rightarrow \max_{f \perp \mathbf{1}} \frac{f^\top A_{G'} f}{f^\top f} = \max_{f \perp \mathbf{1}} \sum_{gv} \sum_{u \sim_G v} \frac{f(gu) f(vg)}{f^\top f}$$

By the Interlacing Theorem, [Hae95] the second eigenvalue of any subgraph of G' is less than the λ', In particular, the eigenvalue of the graph obtained by taking the edges that are associated with elements of the S_g/S.

Denote that subgraph by $G'_{/S}$. Because $S_g/S \cap S = \emptyset$, we have that the edges sets of G, G' are disjointness sets. Hence the adjacency matrix of the graphs union equals the sum of their adjacency matrices. So in total, we obtain that:

$$\begin{aligned}\lambda' &= \max_{f \perp \mathbf{1}} \frac{f^\top \left(A_G + A_{G'_{/S}}\right) f}{f^\top f} \\ &\le \max_{f \perp \mathbf{1}} \frac{f^\top A_G f}{f^\top f} + \max_{f \perp \mathbf{1}} \frac{f^\top A_{G'_{/S}} f}{f^\top f} \\ &\le \lambda + \lambda = 2\lambda\end{aligned}$$

□

Claim 31. *If Δ is a constant greater than two, and G is a λ-algebraic expander with girth at length $\Omega(\log n)$, then there exists a $g \in \Gamma$ such that $S_g \cap S = \emptyset$.*

Proof. As $\Delta > 2$ there must be at least two different elements $s_1, s_2 \in S$ such that $s_1 \neq s_2, s_2^{-1}$. Pick $g = s_1 s_2$. Now assume through contradiction that there exists a pair $s, r \in S$ such that $gsg^{-1} = r \Rightarrow gs = rg$ and notice that the fact that $s_1 \neq s_2^{-1}$ guarantees that both terms are a product of 3 element group. Therefore either that there is a 6-length cycle in the graph, Or that there is element-wise equivalence, namely $s_1 = r, s_2 = s_1, s = s_2$. The first case contradict the lower bound on the expander girth, which is at least $\Omega(\log_\Delta(n))$, while the other stand in contradiction to the fact that $s_1 \neq s_2$ □

Remark. Regarding Quantum Codes. Notice that any complex designed to hold CSS qLDPC codes must have constant length cycles. Otherwise, the distance of C_x will not be constant, and therefore the condition $H_x H_z^\top = 0$ could be satisfied only if H_z is not a constant row-weight matrix, Put differently C_z is not an LDPC code. Consequently, any trial to generalize the construction for obtaining quantum codes must not rely on claim 31.

Remark. Note On Random Construction. One might wondering if using *Cayley* is necessary. We conjecture that there is a constant $c > 0$ such that sampling pair of $(1+c)\frac{1}{2}\Delta$ regular random graphs, and than take the anti-symmetry union of them might also obtain a good expander such that each of the residue part also has good expansion with heigh probability.

Claim 32. *Consider the graph G and the code C as defined in [18] and let S, T be a pair of disjointness vertices subsets. And let x_S and x_T codewords of the $C_\oplus$ such that x_S suggested only by vertices in S, and in similar manner x_T suggested only by T's vertices. Then the flux of S over T is at most:*

$$w_T(x_S) \leq E_{G'}(S,T) \leq \frac{1}{2}\Delta\frac{|S||T|}{n} + \lambda\sqrt{|S||T|}$$

Proof. The only edges that can interfere are the edge defined by J', Namely the edges which belong to *Cayley*(Γ, J'). Therefore it's enough to use the mixing expander lemme on the $\frac{1}{2}\Delta$-regular graph. □

Proof of Theorem 1+. Notice that $\frac{1}{2} < \frac{2}{3} = q$, Thus repeating exactly over proof above obtains that:

$$\begin{aligned}
w_{E/E'}(x|_U) &\geq \delta_0 q\Delta|U| - w\left(E(U_{-1}\bigcup U_{+1}, U)\right) \\
&\geq \delta_0 q\Delta|U| - \frac{\Delta}{2}\frac{|U|\left(|U_{-1}| + |U_{+1}|\right)}{n} - \lambda\sqrt{|U|\left(|U_{-1}| + |U_{+1}|\right)} \\
&\geq \left(\delta_0 - \frac{2}{3} - \frac{1}{q}\frac{\lambda}{\Delta}\right) q\Delta|U|
\end{aligned}$$

When the last inequality follows form the fact that the proof of **??** doesn't relay on graph structure arguments. The same arguments leads also to analog inequality for **??**.

Choosing J such that *Cayley*(Γ, J) is ramnujan provide that $\frac{2\lambda}{\Delta q}$ scale as $\Theta\left(\frac{1}{\sqrt{\Delta}}\right)$. That close the case in which there is a linear size layer of nontrivial suggestions. In other case, in which any such layer is at size less than $\alpha' n$ ($\alpha' = \left(\delta_0 - \left(q - \frac{1}{2}\right)\right)$?) then we obtain the testability for free □

Bibliography

[Ham50] R. W. Hamming. "Error detecting and error correcting codes". In: *The Bell System Technical Journal* 29.2 (1950), pp. 147–160. DOI: 10.1002/j.1538-7305.1950.tb00463.x.

[RS60] Irving S. Reed and Gustave Solomon. "Polynomial Codes Over Certain Finite Fields". In: *Journal of The Society for Industrial and Applied Mathematics* 8 (1960), pp. 300–304.

[Tan81] R. Tanner. "A recursive approach to low complexity codes". In: *IEEE Transactions on Information Theory* 27.5 (1981), pp. 533–547. DOI: 10.1109/TIT.1981.1056404.

[Hae95] Willem H. Haemers. "Interlacing eigenvalues and graphs". In: *Linear Algebra and its Applications* 226-228 (1995). Honoring J.J.Seidel, pp. 593–616. ISSN: 0024-3795. DOI: https://doi.org/10.1016/0024-3795(95)00199-2. URL: https://www.sciencedirect.com/science/article/pii/0024379595001992.

[Sho95] Peter W. Shor. "Scheme for reducing decoherence in quantum computer memory". In: *Phys. Rev. A* 52 (4 Oct. 1995), R2493–R2496. DOI: 10.1103/PhysRevA.52.R2493. URL: https://link.aps.org/doi/10.1103/PhysRevA.52.R2493.

[CS96] A. R. Calderbank and Peter W. Shor. "Good quantum error-correcting codes exist". In: *Physical Review A* 54.2 (Aug. 1996), pp. 1098–1105. DOI: 10.1103/physreva.54.1098. URL: https://doi.org/10.1103%2Fphysreva.54.1098.

[Gro96] Lov K. Grover. *A fast quantum mechanical algorithm for database search*. 1996. arXiv: quant-ph/9605043 [quant-ph].

[SS96] M. Sipser and D.A. Spielman. "Expander codes". In: *IEEE Transactions on Information Theory* 42.6 (1996), pp. 1710–1722. DOI: 10.1109/18.556667.

[Sho97] Peter W. Shor. "Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer". In: *SIAM Journal on Computing* 26.5 (Oct. 1997), pp. 1484–1509. DOI: 10.1137/s0097539795293172. URL: https://doi.org/10.1137%2Fs0097539795293172.

[AB99] Dorit Aharonov and Michael Ben-Or. *Fault-Tolerant Quantum Computation With Constant Error Rate*. 1999. arXiv: quant-ph/9906129 [quant-ph].

[AK99] Ashish Ahuja and Sanjiv Kapoor. *A Quantum Algorithm for finding the Maximum*. 1999. arXiv: quant-ph/9911082 [quant-ph].

[Den+02] Eric Dennis et al. "Topological quantum memory". In: *Journal of Mathematical Physics* 43.9 (Sept. 2002), pp. 4452–4505. DOI: 10.1063/1.1499754. URL: https://doi.org/10.1063%2F1.1499754.

[HLW06] Shlomo Hoory, Nathan Linial, and Avi Wigderson. “Expander graphs and their applications”. In: *Bulletin of the American Mathematical Society* 43.4 (2006), pp. 439–561.

[Got14] Daniel Gottesman. *Fault-Tolerant Quantum Computation with Constant Overhead*. 2014. arXiv: 1310.2984 [quant-ph].

[TZ14] Jean-Pierre Tillich and Gilles Zemor. “Quantum LDPC Codes With Positive Rate and Minimum Distance Proportional to the Square Root of the Blocklength”. In: *IEEE Transactions on Information Theory* 60.2 (Feb. 2014), pp. 1193–1202. DOI: 10.1109/tit.2013.2292061. URL: https://doi.org/10.1109%2Ftit.2013.2292061.

[LTZ15] Anthony Leverrier, Jean-Pierre Tillich, and Gilles Zemor. “Quantum Expander Codes”. In: *2015 IEEE 56th Annual Symposium on Foundations of Computer Science*. IEEE, Oct. 2015. DOI: 10.1109/focs.2015.55. URL: https://doi.org/10.1109%2Ffocs.2015.55.

[BGW19] Michael Ben-Or, Shafi Goldwasser, and Avi Wigderson. “Completeness Theorems for Non-Cryptographic Fault-Tolerant Distributed Computation”. In: *Providing Sound Foundations for Cryptography: On the Work of Shafi Goldwasser and Silvio Micali*. New York, NY, USA: Association for Computing Machinery, 2019, pp. 351–371. ISBN: 9781450372664. URL: https://doi.org/10.1145/3335741.3335756.

[PK21] Pavel Panteleev and Gleb Kalachev. *Asymptotically Good Quantum and Locally Testable Classical LDPC Codes*. 2021. DOI: 10.48550/ARXIV.2111.03654. URL: https://arxiv.org/abs/2111.03654.

[AF22] “Maximum distance separable (MDS) code”. In: *The Error Correction Zoo*. Ed. by Victor V. Albert and Philippe Faist. 2022. URL: https://errorcorrectionzoo.org/c/mds.

[Din+22] Irit Dinur et al. *Good Locally Testable Codes*. 2022. DOI: 10.48550/ARXIV.2207.11929. URL: https://arxiv.org/abs/2207.11929.

[KP22] Gleb Kalachev and Pavel Panteleev. *Two-sided Robustly Testable Codes*. 2022. arXiv: 2206.09973 [cs.IT].

[LZ22] Anthony Leverrier and Gilles Zémor. *Quantum Tanner codes*. 2022. arXiv: 2202.13641 [quant-ph].