Understanding Quantumness And Testability.

David Ponarovsky

April 4, 2023

Contents

List of Figures

Chapter 1

Introduction

Chapter 2

Codes

2.1 Introduction

Chapter 3

Introduction

3.0.1 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.*

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 computation only with a tiny probability.

Definition 4. *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*

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

3.0.2 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. In the next section, we will review the Singleton bound, which states that for any code (not necessarily good), there must be a zero-sum game between the relative distance and the rate. Now, we are ready to formulate our contribution.

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.

3.0.3 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 $n - d + 1$ points. By define the code to be the subspace contains all the polynomials at degree at most d encoded by n numbers associated with their values. Formally we define:

Definition 5. *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:*

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

Lemma 1. *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 vectors $e_1, e_2..e_d = 1, x..x^d$ 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 $n - d$. □

Notice that encoding naively the aleph-bet of $\mathbb{F}_p$ in binary strings require to pay a factor $\log n$ bits, So the asymptotic rate of the code attends to zero. **[COMMENT]** Add a statement about the vanishing rate of the binary encoded version. And add a paragraph about Tanner code in which each edge correspond to a non binary alpha-bet.

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.)

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.

3.0.4 Tanner Code

The constructions require two main ingredients: a graph Γ, and for simplicity, we will restrict ourselves to a Δ regular graph. Secondly, 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 E 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 6. *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.*

It's also worth mentioning that the first construction of good classical codes, due to Sipser and Shpilman, are Tanner codes over expanders graphs [SS96].

Theorem. *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 □

Setting C_0 To Be The Polynomial Code.

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

3.0.5 Expander Codes

We saw how a graph could give us arbitrarily long codes with a positive rate. We will show, Sipser's result that if the graph is also an expander, we can guarantee a positive relative distance. We notice that the name expander codes is coined for a more general version than the one we will present.

Definition 7. *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:*

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

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 □

Chapter 4

Locally Testable Codes.

Chapter 5

Quantum Error Correction Codes.

Chapter 6

Good qLDPC and LTC.

Chapter 7

Local Majority $\neq$ Local Testability.

Claim 1. *Suppose that G is an expander graph with a second eigenvalue λ, then For any layer U there exist a layer U' such that:*

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

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 2. *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 2, 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 3. *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 4. *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 5. *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 1 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 3 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 □

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.

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.

[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.

[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.

[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.