Quantum Coding Theory **(UC Berkeley CS294, Spring 2024)**

Lecture 1: Quantum Tanner Codes IV

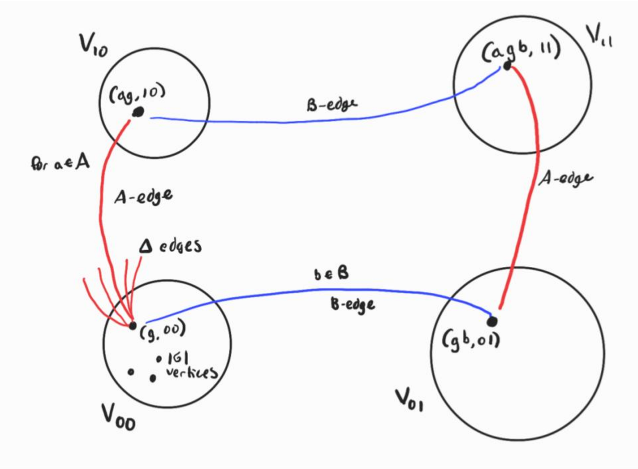
January 24, 2024

Lecturer: John Wright *Scribe: Tobias Scott*

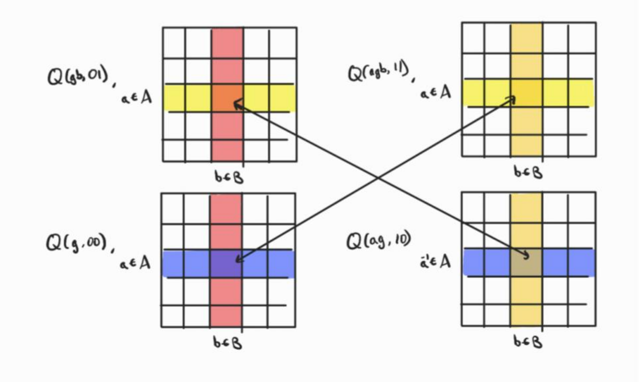
1 Recap

1.1 Left-Right Cayley Complex

To build a Quantum Tanner Code, we start with a group of checks G that we want to pass; consider sets of generators A, B of this group, such that $|A| = |B| = \Delta$. We can build a left-right Cayley Complex X, as a graph consisting of $V_{00} \cup V_{01} \cup V_{10} \cup V_{11}$, where $V_{ij} = G \times \{ij\}$, and each (g, ij) is connected to each $(ag, (i+1)j)$ and $(gb, i(j+1))$ by edges identified by $a \in A, b \in B$ (in our case, each $a = a^{-1}$ so the graph is undirected).



A square is identified by $\{(g, 00), (ag, 10), (gb, 01), (agb, 11)\}$; we can identify the set of all squares by Q and for any vertex v, can identify $Q(v)$ as the squares incident to v. Each square incident to a vertex is identified by all $a \in A$ and $b \in B$ that can define the edges to the square at that vertex, forming a $\Delta \times \Delta$ matrix. We can note that $(g, 00)$ and $(ag, 10)$ share an edge, so share a set of squares that contain this edge, defined by a row in $Q(g, 00)$ and $Q(ag, 10)$. Likewise, $Q(g, 00)$ and $Q(gb, 01)$ share a column corresponding to the squares containing the edge shared by these vertices; $Q(g, 00)$ and $Q(agb, 11)$ share a single entry corresponding to the square defined by a, b.



1.2 Defining a CSS Code

First, we establish C_A and C_B, linear error correcting codes on A and B respectively (that is, spaces of strings orthogonal to the checks in A and B). For our CSS code, we instantiate physical bits as each element in the set of squares Q, of which there are Δ^2 squares at each of $|G|$ vertices in some V_{ij}. Each string in the code is a sum of strings defined by a vertex in some V_{ij}, so we can zoom into a particular v, where the string is zero everywhere except on the indices corresponding to the squares defining $Q(v)$, where its values can be taken to form a matrix.

We can define an X-code Code_0 as the space of strings such that for each $v \in V_{00} \cup V_{11}$, the strings on $Q(v)$ pass the space of checks in $C_A \otimes C_B$. Elements of $(C_A \otimes C_B)^\perp$ have zero dot product with $C_A \otimes C_B$, so can be decomposed into a component where the columns of $Q(v)$ pass checks in C_A and a component where the rows of $Q(v)$ pass checks in C_B, Hence, the local code across each $v \in V_{00} \cup V_{11}$ requires that each $Q(v) = c + r$ for $c \in C_A^\perp \otimes \mathbb{F}_2^B$ and $r \in \mathbb{F}_2^A \otimes C_B^\perp$, which we write as $Q(v) \in C_A^\perp \otimes \mathbb{F}_2^B + \mathbb{F}_2^A \otimes C_B^\perp$. We can then define the Z-code Code_1 as the space of strings such that for each $v \in V_{01} \cup V_{10}$, the strings on $Q(v)$ pass the checks in $C_A^\perp \otimes C_B^\perp$, so each matrix $Q(v) \in C_A \otimes \mathbb{F}_2^B + \mathbb{F}_2^A \otimes C_B$.

To show that this is a CSS code, we consider without loss of generality a $u \in V_{00}$ and $v \in V_{01}$, and an arbitrary $h_X \in C_A \otimes C_B$, $h_Z \in C_A^\perp \otimes C_B^\perp$ on these respective vertices. There are 2 cases: these vertices are not connected by any $b \in B$, so $Q(u) \cap Q(v) = \emptyset$. This means that the checks are nonzero on different substrings defined by $Q(u)$ and $Q(v)$, so $h_X \cdot h_Z = 0$. If alternatively u, v are connected by an edge $b \in B$, $Q(u) \cap Q(v)$ is a set of all edges $a \in A$ that each create a square with this shared edge b. This defines a column in each matrix $Q(u)$ and $Q(v)$ where they agree; a product of parity checks on each vertex is therefore a check on a column in $Q(u)$ times a check on a column in $Q(v)$ (plus 0 corresponding to the rest of the space, where they are not both nonzero). Because $Q(u) \in \text{Code}_0, Q(v) \in \text{Code}_1$, $h_X \cdot h_Z = (h_x \in C_A) \cdot (h_z \in C_A^\perp) + 0 = 0$. The orthogonality of these checks ensures that $\text{Code}_0^\perp \subseteq (\text{Code}_1^\perp)^\perp = \text{Code}_1$ and $\text{Code}_1^\perp \subseteq (\text{Code}_0^\perp)^\perp = \text{Code}_0$.

1.3 Parameters of CSS Code

We began with linear error correcting codes C_A and C_B; if we strategically choose "good" codes, we can create a "good" CSS code. Each code acts on the generating edges in A, B, so has Δ bits; C_A encodes a number of logical bits linear in this number, an arbitrary $\rho\Delta$, and C_B is strategically chosen to have a number of logical bits equal to the number of independent checks constraining C_A. Hence $C_A = [\Delta, \rho\Delta], C_B = [\Delta, (1-\rho)\Delta]$ (each has some distance that will be discussed later).

For each vertex in $V_{00} \cup V_{11}$, the X-code is constrained by checks on $Q(v)$ corresponding to the $(\rho\Delta)((1-\rho)\Delta)$ parity checks in $C_A \otimes C_B$. Hence the dimension of X-constraints is $2|G|\rho(1-\rho)\Delta^2$. Similarly, for each vertex in $V_{01} \cup V_{10}$, the Z-code is constrained by checks on $Q(v)$ corresponding to the $((1-\rho)\Delta)(\rho\Delta)$ parity checks in $C_A^\perp \otimes C_B^\perp$.

For each of $|G|$ vertices in a V_{ij}, there are Δ^2 combinations of a, b edges defining all possible squares in the space. This space of bits is constrained by both of these sets of X-checks and Z-checks, so the dimension of the CSS code is $|G|\Delta^2 - 4|G|\rho(1-\rho)\Delta^2 = n(1-2\rho)^2$, so linear in the number of logical bits.

Importantly, each parity check is applied to $\leq \Delta^2$ bits, and each bit, identified by a square, is involved in $\leq 4\rho(1-\rho)\Delta^2$ checks: hence, the code is low-density.

2 Quantum Tanner Code Distance

2.1 Distance from Tanner Codes

Having established that C_A, C_B are "good" codes, we recognize that their code distance is linear in their number of bits Δ, as are their orthogonal codes $C_A^\perp, C_B^\perp$. We can define the minimum distance of these code spaces to be $\delta\Delta$. A challenge remains to ensure that the graphs instantiated by this code are not so expansionary that their distances are hard to constrain from constraints on the terms in their products. This is a familiar problem from our discussion of Tanner codes.

In fact, we can recognize that our X-code and Z-code are each Tanner codes, which are applied to a graph and a base code, and require that the sub-strings corresponding to the sets of edges at each vertex are in the base code. In particular, we can define $G_0^\square$ as the graph with vertices in $V_{00} \cup V_{11}$ and edges corresponding to each square connecting such vertices. Now, the X-code is defined on this graph, and requires that the substrings at each vertex are in the code space $C_A^\perp \otimes \mathbb{F}_2^B + \mathbb{F}_2^A \otimes C_B^\perp$, so is $\mathrm{Tan}(G_0^\square, C_A^\perp \otimes \mathbb{F}_2^B + \mathbb{F}_2^A \otimes C_B^\perp)$. Likewise, the Z-code is the Tanner code $\mathrm{Tan}(G_1^\square, C_A \otimes \mathbb{F}_2^B + \mathbb{F}_2^A \otimes C_B)$.

We note that the number of distinct edges at $G_0^\square$ correspond to the a edges that can be taken from V_{00} to V_{10} times the b edges that can be taken from V_{10} to V_{11}, each in composition defining a different edge of $G_0^\square$. This means that our constraint on the individual Cayley graphs connecting V_{ij} to its neighbor will multiply.

We must consider the constraints we can impose on our Cayley graphs. For a graph $G = (V, E)$, its adjacency matrix has rows and columns consisting of vertices u, v in the graph, and entries of 1 if u and v are connected by an edge. This real, symmetric matrix has real eigenvalues in an orthonormal basis, which we order as $\lambda_1 \geq ... \geq \lambda_n$. We can define $\lambda = \max\{\lambda_2, |\lambda_n|\}$, and recognize that for an r-regular graph, where each vertex is connected to r other vertices, λ_1 is guaranteed to be around r, but a smaller λ provides a tighter constraint on how the graph expands. A graph is Ramanujan if $\lambda \leq 2\sqrt{r-1}$; we choose strategically to have Cayley graphs that are Ramanujan: specifically, that $Cay_L(G, A)$ and $Cay_R(G, B)$ are such that $\lambda \leq 2\sqrt{\Delta}$; now, $G_0^\square$ and $G_1^\square$ have $\lambda \leq (2\sqrt{\Delta})(2\sqrt{\Delta}) = 4\Delta$.

Finally, we examine a condition on the base codes $C_A^\perp \otimes \mathbb{F}_2^B + \mathbb{F}_2^A \otimes C_B^\perp$ and $C_A \otimes \mathbb{F}_2^B + \mathbb{F}_2^A \otimes C_B$ (which is fulfilled by all of our attempted example codes, so finding a counterexample may yet prove challenging): the codes must be κ-product expanding. Focusing on $x \in C_A \otimes \mathbb{F}_2^B + \mathbb{F}_2^A \otimes C_B$, we recognize that it decomposes as $c + r$ where c has columns in C_A and r has rows in C_B, $|x| = |c + r| \geq \kappa\Delta(||c|| + ||r||)$, where $|z|$ is the Hamming weight and $||c||$ $(||r||)$ consists of the number of nonzero columns (rows). This constrains the expansion of columns in C_A and rows in C_B, so limits the Hamming weight of $c_v + r_v$ in terms of the number of nonzero rows and columns in c_v and r_v.

2.2 Minimal Representations

A string $x \in \text{Code}_1 \backslash \text{Code}_0^\perp$ at a particular vertex $v \in V_{01} \cup V_{10}$ is some $x_v \in C_A \otimes \mathbb{F}_2^B + \mathbb{F}_2^A \otimes C_B$. We call the minimal representation of its restriction to a vertex v to be its decomposition $x_v = c_v + r_v$ minimizing $||c_v|| + ||r_v||$. These c_v, r_v across all vertices in a particular V_{ij} define the value of x on every set of edges going through every vertex in V_{ij}, so serve as a basis by which we can decompose an entire $x \in Code_1 \setminus Code_0^\perp$. Choosing this decomposition across the vertices of V_{01}, we get $x = C_0 + R_1$ where $C_0 = \sum_{v \in V_{01}} c_v$ and $R_1 = \sum_{v \in V_{01}} r_v$. We can also get an equivalent decomposition across vertices in V_{10}, $x = C_0 + R_1$ where $C_0 = \sum_{v \in V_{10}} c_v$ and $R_1 = \sum_{v \in V_{10}} r_v$.

We note that strings c_v, r_v are weight $\leq \Delta^2$, and that for $v_1, v_2 \in V_{10}$, no squares contains both vertices, so their strings $c_{v_1}, r_{v_1}, c_{v_2}, r_{v_2}$ are non-overlapping. However, for $v_1 \in V_{10}$ and $v_2 \in V_{01}$, there are some squares that share both vertices, so their strings $c_{v_1}, r_{v_1}, c_{v_2}, r_{v_2}$ could overlap.

A representation of x is minimal if $||C_0|| + ||C_1|| + ||R_0|| + ||R_1||$ is minimized. We define the norm $||x||$ to be $||C_0|| + ||C_1|| + ||R_0|| + ||R_1||$ for the minimal representation (C_0, C_1, R_0, R_1).

This is a crucial piece of analysis: we must establish a relationship between the constraints on V_{10} and V_{01}, and yet the constraints on the former create decompositions of the form $C_0 + R_1$ and the constraints on the latter create decompositions of the form $C_1 + R_0$. To relate them, we create new strings $C_0 + R_0$ in a new "Frankenstein" code space $\text{Tan}(G_0^{\square}, C_A \otimes \mathbb{F}_2^B + \mathbb{F}_2^A \otimes C_B)$ consisting of the vertices from $Code_0$ and the connections from $Code_1$.

First, we consider decompositions $x = C_0 + R_1 = C_1 + R_0$, and recognize that because we work in addition modulo 2, now $C_0 + R_0 = C_1 + R_1$. We call this string x_0. Our goal is to create (C_0, C_1, R_0, R_1) that will be minimal as a representation for x_0.

Suppose a decomposition is not minimal at some $v \in V_{00}$. Now $(x_0)_v = (C_0)_v + (R_0)_v = c_v + r_v$. Replace these with $c'_v + r'_v$; because these must still add to x_0, whatever term t_v displaces c'_v from c_v also displaces r'_v from r_v (again, note that $t_v = -t_v$ in addition modulo 2). For c'_v and c_v to both be $\in C_A \otimes \mathbb{F}_2^B$, their difference t_v must have columns in C_A; likewise for r'_v and r_v to both be $\in \mathbb{F}_2^A \otimes C_B$, their difference t_v must have rows in C_B. Hence, $t_v \in C_A \otimes C_B = (C_A^{\perp} \otimes \mathbb{F}_2^B + \mathbb{F}_2^A \otimes C_B^{\perp})^{\perp} = Code_0^{\perp}$. So, a new choice of representation of x_0 corresponds to a shift of C_0 to $C_0 + t_v$ and R_0 to $R_0 + t_v$. So, a new representation for $x_0 = (C_0 + t_v) + (R_0 + t_v)$ creates a new $x + t_v = C_0 + R_1 + t_v = C_1 + R_0 + t_v$, which differs from the old one by an element of $\text{Code}_0^{\perp}$.

This proves a Lemma: if x has the minimum norm in $x + Code_0^{\perp}$, then the minimum representation of x corresponds to the decomposition of x_0 with minimal norm.

This will prove important: we can use the minimum-norm $x \in x + \text{Code}_0^{\perp}$, and define expansionary properties on x in terms of $S_{ij} = \{v \in V_{ij} : (C_i + R_j)|_{Q(v)} \neq 0\}$, which makes use of $C_0 + R_0$ and $C_1 + R_1$ to describe expansion through V_{00} and V_{11}.

2.3 The Objective

We want to prove that the Z-distance is linear in the number of bits (and apply symmetrical arguments that we do not examine for the X-distance). This condition is met if $||x|| \geq \frac{\delta^2 \kappa n}{512\Delta^2}$. If this is true, we can prove the following theorem:

$$d_Z^{+} = \min_{x \in \text{Code}_1 \setminus \text{Code}_0^{\perp}} |x| \geq \frac{\delta^{22} n}{1024\delta^2}$$

To prove this, let $x \in \text{Code}_1 \setminus \text{Code}_0^{\perp}$ and (C_0, C_1, R_0, R_1) be the minimial representation.

$$x = C_0 + R_1 = \sum_{v \in V_{01}} c_v + r_v$$

$$|x| = |C_0 + R_1| = |\sum_{v\in V_{01}} c_v + r_v|$$

Because the vertices in V_{01} are non-overlapping in their squares, $|\sum_{v\in V_{01}} c_v + r_v| = \sum_{v\in V_{01}} |c_v + r_v|$. Furthermore, each $|c_v + r_v| \geq \kappa\Delta(||c_v|| + ||r_v||)$; recall that a κ-product expanding graph constrains expansion of columns in C_A and rows in C_B, so limits the Hamming weight of $c_v + r_v$ in terms of the number of nonzero rows and columns in c_v and r_v.

$$|x| = \sum_{v\in V_{01}} |c_v + r_v| \geq \sum_{v\in V_{01}} \kappa\Delta(||c_v|| + ||r_v||) = \kappa\Delta(||C_0|| + ||R_1||)$$

Similarly:

$$|x| \geq \kappa\Delta(||C_1|| + ||R_0||)$$

Combining these:

$$|x| \geq \kappa\Delta\frac{1}{2}\big(||C_0|| + ||C_1|| + ||R_0|| + ||R_1||\big) = \kappa\Delta\frac{1}{2}\big(||x||\big) = \kappa\Delta\frac{1}{2}\big(\frac{\delta^2\kappa n}{512\Delta^2}\big)$$

The rest of this lecture attempts to prove the goal that $||x|| \geq \frac{\delta^2\kappa n}{512\Delta^2}$. This is true for any x if it is true for the minimum-norm $x \in x + \text{Code}_0^\perp$.

3 Proving the Code Distance

3.1 Exceptional and Ordinary Vertices

Assume $||x|| < \frac{\delta^2\kappa n}{512\Delta^2}$. Now define $S_{ij} = \{v \in V_{ij} : (C_i + R_j)|_{Q(v)} \neq 0\}$, the set of vertices on which the relevant matrix is nonzero.

We assume $C_i + R_j$ to be the minimal representation (of either x or x_0), which can only be true because we have assumed that x has the minimum norm in $x + \text{Code}_0^\perp$). Because of this, each vertex on which $C_i + R_j$ is nonzero contributes either a row or a column to C_i or R_j:

$$|S_{ij}| \leq ||C_i|| + ||R_j||$$

Adding whichever terms are missing:

$$|S_{ij}| \leq ||C_0|| + ||C_1|| + ||R_0|| + ||R_1|| = ||x|| < \frac{\delta^2\kappa n}{512\Delta^2}$$

Now, we examine each $v \in S_{ij}$, and define it as either exceptional or ordinary. v is exceptional if $||c_v|| + ||r_v|| \geq \alpha\Delta$ for $\alpha = \frac{\delta^2}{256}$, and v is ordinary if $||c_v|| + ||r_v|| > \alpha\Delta$. Define S_{ij}^e to be the set of exceptional vertices and S_{ij}^o to be the set of ordinary vertices.

3.2 Weight of Ordinary Vertices

Assume v is ordinary. Any nonzero column in c_v comes from C_A, so fulfills that code's distance and so has $\geq \delta\Delta$ 1s. Each 1 in this column that gets cancelled by an entry in a row corresponds to a nonzero row in r_v, so because v is ordinary this can only be the case for $< \alpha\Delta$ rows. Hence c_v has $\geq \delta\Delta - \alpha\Delta$ 1s (in either x_v or $(x_0)_v$, whichever is relevant for this S_{ij}). $\alpha \ll \delta$, so c_v has $\geq \frac{\delta}{2}\Delta$ 1's.

We will create a bound on the number of exceptional vertices, allowing us to describe our behavior mostly in terms of these ordinary vertices.

3.3 Number of Exceptional Vertices

Lemma: $|S_{00}^e|, |S_{11}^e| \leq \frac{64}{\alpha^2\kappa^2\Delta^2}|S_{00} \cup S_{11}|$, $|S_{01}^e|, |S_{10}^e| \leq \frac{64}{\alpha^2\kappa^2\Delta^2}|S_{01} \cup S_{10}|$.

We note that Δ, the number of generators used to define the original graph, so is a parameter under our control, which can allow us to scale $|S_{ij}^e|$ to being arbitrarily small. This is the condition that we use to create a contradiction.

The proof of this Lemma is an exercise on Homework 4. However, we present an intuitive sketch here.

Consider $v \in S_{00}^e$. $Q(v)$ appears in x_0; because v is exceptional, x_0 restricted to $Q(v)$ has many nonzero rows and nonzero columns; these rows and columns appear in the minimal representation of x_0 and provide a bound on Hamming weight because of the κ-product expansion. So there are many squares that have vertex v and appear in x_0.

Considering $u \in S_{11} \subseteq V_{11}$, so $x_1|_{Q(u)}$ contains at least one 1.

Examining the squares at v defined by adjacent edges in x_0, and connecting to an edge adjacent to some $u \in V_{11}$, we find an adjacency relationship defined by $G_0^{\square}$, as seen previously. This is the graph with all vertices in $V_{00} \cup V_{11}$ edges defining squares. This is an expander graph: because we have defined it to be Ramanujan, $\leq 4\Delta$, which means that each $1 \in x_0|_{Q(v)}$ ends up connecting to many different $u \in V_{11}$; and we recall that there are many such 1s in x_0. Qualitatively:

$$|S_{00}^e| \cdot \gamma \leq |S_{11}| \leq |S_{00} \cup S_{11}|$$

for a large γ. This means that $|S_{00}^e|$, the number of exceptional vertices, is constrained; too many of them would expand into a result too big in S_{11}. (The formal proof, an exercise, uses the Expander Mixing Lemma for this step).

Assume $|S_{01} \cup S_{10}| \geq |S_{00} \cup S_{11}|$ and $|S_{01}| \geq |S_{10}|$: most nonzero neighborhoods are next to vertices in V_{01}. Now,

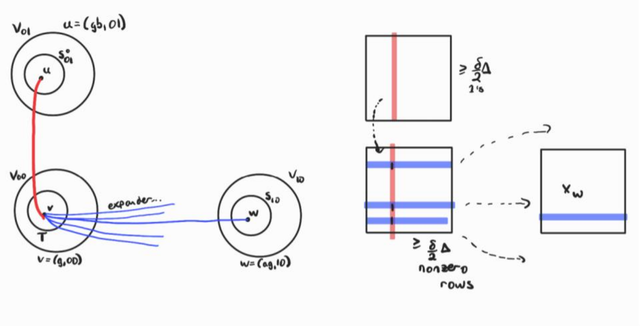
$$|S_{01}^e| \leq \frac{64}{\alpha^2\kappa^2\Delta^2}|S_{01} \cup S_{10}| \leq \frac{64}{\alpha^2\kappa^2\Delta^2}2|S_{01}|$$

So by controlling Δ, we can make $|S_{01}^e| \leq$ any small constant $\cdot|S_{01}|$; the proportion of ordinary vertices can be made to be $\geq a$ for any $a < 1$.

3.4 Expansion into S_{10}

Now, consider $u \in S_{01}^0$; now $x_u = c_u + r_u$, and there is at least one nonzero column in c_u or row in r_u. We can assume without loss of generality that at least half of ordinary vertices produce a nonzero column (if not, it would be true for rows): hence the number of nonzero columns is $\geq \frac{1}{2}|S_{01}^0| \geq \frac{a}{2}|S_{01}|$. We call these "ordinary columns", and know that their Hamming weight is $\geq \frac{\delta}{2}\Delta$.

We can define $T = \{v \in S_{00}$: a nonzero column of c_v is shared with an ordinary $u \in S_{01}^o\}$. Now, we show a Lemma: $|T| \leq \frac{64}{\delta^2\Delta}|S_10|$, again replacing a proof using the Expander Mixing Lemma (an exercise) with an intuitive picture.



For a $v = (g, 00) \in T$, $x|_{Q(v)} = x_v = c_v + r_v$ shares a column with some $u \in S_{01}^o$ (we recall that $Q(u = (gb, 01))$ shares some column with $Q(v = (g, 00))$). Because this is an ordinary column, with $\geq \frac{\delta}{2}\Delta$ ones, this column in c_v also has $\geq \frac{\delta}{2}\Delta$ ones. This means that it must have at least $\frac{\delta}{2}\Delta$ nonzero rows.

Now, consider $w = (ag, 10) \in V_{10}$ (recall that $Q(w = (ag, 10))$ shares some row with $Q(v = (g, 00))$). Now, x_w has a nonzero row, so is in S_{10}; the adjacency relation between these vertices, given that they share a row, is defined by $\mathrm{Cay}_L(G, A)$, which is a Ramanujan expander graph, ensuring that this expansion hits enough elements in S_{10}.

There are $|T|$ vertices; each contributes at least $\frac{\delta}{2}\Delta$ nonzero rows. In the expander graph, these rows correspond to different edges to different vertices in S_{10}, up to a factor proportional to δ; hence, the number of vertices in $|S_{10}|$ is greater than this number, and we arrive at $|T|\frac{\delta}{2}\Delta \leq \frac{\eta}{\delta}|S_{10}|$.

3.5 Averaging Across T

Consider the average of $||c_v||+||r_v||$ across all $v \in T = \{v \in S_{00}$: a nonzero column of c_v is shared with an o

$$\text{avg}_{v\in T}(||c_v|| + ||r_v||) \geq \frac{1}{|T|}\sum_{v\in T}||c_v||$$

We are summing over every v that shares a nonzero column with an ordinary vertex $\in S_{01}^o$; hence this is lower bounded by the number of "ordinary columns," which we assumed without loss of generality to be $\geq \frac{a}{2}|S_{01}|$. Using the lower bound on $|T|$:

$$\text{avg}_{v\in T}(||c_v|| + ||r_v||) \geq \frac{\delta^2\Delta}{64|S_{10}|}\left(\frac{a}{2}|S_{01}|\right)$$

Noting that there are at least as many vertices in S_{01} as in S_{10}, we note:

$$\text{avg}_{v\in T}(||c_v|| + ||r_v||) \geq \frac{\delta^2\Delta a}{128}$$

Now, recall that $v \in S_{00}^e$ if $||c_v|| + ||r_v|| \geq \frac{\delta^2\Delta}{256}$, so the average norm of a vertex in T is almost twice the lower bound on what is needed for a vertex to be exceptional; hence, there must be many exceptional vertices, which we will use to arrive at our long-awaited contradiction.

Where p is the fraction of exceptional vertices in T, the average norm across vertices in T is upper bounded by the fraction that are exceptional (which have norm at most Δ, an entire column) plus the fraction that are ordinary (which have norm at most $\frac{\delta^2\Delta}{256}$). So:

$$\frac{\delta^2\Delta a}{128} \leq \text{avg}_{v\in T}(||c_v|| + ||r_v||) \leq p\Delta + (1-p)\frac{\delta^2\Delta}{256}$$

$$2a\frac{\delta^2\Delta}{256} \leq \text{avg}_{v\in T}(||c_v|| + ||r_v||) \leq p\Delta + \frac{\delta^2\Delta}{256}$$

$$(2a-1)\frac{\delta^2\Delta}{256} \leq p\Delta$$

So:

$$p \geq (2a-1)\frac{\delta^2}{256}$$

which is a constant fraction, independent of Δ.

3.6 Finding a Contradiction

A constant fraction of T are exceptional; if T is large enough, we get a number of exceptional vertices that can break our previous bounds.

At any $v \in V_{01}$, there are at least $\frac{a}{2}|S_{10}|$ ordinary columns. T consists of all the vertices sharing an ordinary column; the smallest it could be would be defined by a small number of vertices, all of whose columns are ordinary columns. Because each vertex has Δ columns, we can take at most Δ ordinary columns that could be shared by a single vertex in T. Hence, the number of vertices in T is at least $\frac{a}{2}|S_{10}|\frac{1}{\Delta}$, which is $\geq \frac{a}{2\Delta}|S_{10} \cup S_{01}|\frac{1}{2}$.

The set of exceptional vertices is at least the size of T times the fraction p of T that is exceptional (which we just lower bounded):

$$|S_{00}^e| \geq |T| \cdot p \geq (2a-1)\frac{\delta^2}{256}\big(\frac{a}{4\Delta}|S_{10} \cup S_{01}|\big) = \frac{C_1}{\Delta}|S_{00} \cup S_{11}|$$

for some constant C_1. But we have already established that $|S_{00}^e| \leq \frac{64}{\alpha^2\kappa^2\Delta^2}|S_{00} \cup S_{11}| = \frac{C_2}{\Delta^2}|S_{00} \cup S_{11}|$; for any $C_1, C_2, |S_{00} \cup S_{11}|$, we can choose some Δ large enough that it cannot be true that:

$$\frac{C_1}{\Delta}|S_{00} \cup S_{11}| \leq |S_{00}^e| \leq \frac{C_2}{\Delta^2}|S_{00} \cup S_{11}|$$

So, it cannot have been true that $||x|| < \frac{\delta^2\kappa}{512\Delta^2}n$. As shown, meeting this goal establishes that our minimum-weight error in $\text{Code}_1 \setminus \text{Code}_0^\perp$ is linear in n; this establishes the Quantum Tanner Code as our first-ever "good" quantum code.

References

[1] John Wright (Apr. 2024) *Quantum Tanner Codes IV*, CompSci 294: Quantum Coding Theory.

[2] Shouzhen Gu, Christopher A. Pattison and Eugene Tang (Jun. 2023) *An efficient decoder for a linear distance quantum LDPC code*, Proceedings of the 55th Annual ACM Symposium on Theory of Computing.

[3] Anthony Leverrier and Gilles Zemor (Oct. 2022) *Quantum Tanner codes*, 2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS).

[4] Anthony Leverrier and Gilles Zémor (Aug. 2023) *Decoding Quantum Tanner codes*, IEEE Transactions on Information Theory.

[5] A. Lubotzky, R. Phillips and P. Sarnak (Sep. 1988) *Ramanujan graphs*, Combinatorica.

[6] Daniel Gottesman (Jan. 2018) *Quantum Error Correction and Fault Tolerance*, QIC 890.

[7] Wikipedia (Jan 2024) *Expander Mixing Lemma*, Wikipedia Foundation.