Topics in Quantum Information **U. Waterloo CO 781, Fall 2006**
Debbie Leung **Sevag Gharibian**
Quantum Error Correction **September 25, 2006**

Lecture Outline for QEC (Quantum Error Correction) Criterion

- Necessary and sufficient conditions
- Reversal map
- Discretized errors

1 Necessary and Sufficient Conditions for QEC

Let P be the projector on to the code space C_L, and let $\mathbb{E} = \{E_i\}$ be a set of errors.

The following statements are then equivalent:

1. $PE_i^\dagger E_i P = C_{ij}P$, $\forall i, j$ and $C \geq 0$, where:
   - The notation $C \geq 0$ means C is positive semi-definite, i.e. its spectral decomposition has non-negative eigenvalues.
   - The E_i's are usually chosen to be Pauli operators (i.e. I, X, Y, Z).
2. Any completely positive (CP) map $\varepsilon[\cdot] = \sum_k A_k \cdot A_k^\dagger$, $A_k \in span(\mathbb{E})$, can be reversed on C, where:
   - The $\cdot$ is a placeholder for the argument to the map.
   - This means $\exists$ a trace-preserving completely positive (TCP) map R s.t. $R \circ \varepsilon[\rho] = [tr(\varepsilon(\rho))] \cdot \rho$, $\forall \rho$ s.t. $P\rho P = \rho$ (i.e. R is a reversal map).

Proof : Statement $1 \Rightarrow$ Statement 2

By our assumption of Statement 1, we have some $C \geq 0$. Therefore, let us diagonalize C s.t.

$$\exists U \text{ s.t. } U^\dagger C U = D, \tag{1}$$

where D is diagonal and has non-negative eigenvalues.

Now, take

$$F_k = \sum_j u_{jk} E_j \tag{2}$$

.

Therefore,

$$\begin{aligned}
PF_l^\dagger F_k P &= \sum_{ij} (u_{il}^*)(u_{jk}) P E_i^\dagger E_j P && \text{(substitute (2))} \qquad (3)\\
&= \sum_{ij} (u_{il}^*)(u_{jk})(c_{ij} P) && \text{(by Statement 1)} \qquad (4)\\
&= \sum_{ij} (U^\dagger)_{li}(u_{jk})(c_{ij} P) && \qquad (5)\\
&= \left(\sum_{ij} U_{li}^\dagger c_{ij} u_{jk} \right) P && \qquad (6)\\
&= [U^\dagger C U]_{lk} P && \qquad (7)\\
&= d_{kk} \delta_{kl} P && (U^\dagger C U \text{ is diagonal by (1))} \qquad (8)
\end{aligned}$$

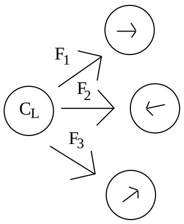
Intuitively, this is because the errors corresponding to k and l should be distinguishable. We can now write:

$$\begin{aligned}
F_k P &= U_k \sqrt{P F_k^\dagger F_k P} && \text{(Polar Decomposition)} \qquad (9)\\
&= U_k \sqrt{d_{kk} P} && \text{(substitute (8))} \qquad (10)\\
&= U_k \sqrt{d_{kk}} P, && \qquad (11)
\end{aligned}$$

since the operator under the square root should be positive semidefinite, so we can apply the square root to its eigenvalues. Thus F_k acts unitarily (like U_k) on the code space. This is sometimes called the nondeforming condition.

The idea now is to first measure our errors, and based on the result of that measurement, apply the appropriate reversal operator. So, in specifying what our reversal map R should look like, we follow two steps:

1. First "detect" which "k" we're dealing with by distinguishing between P_k's, i.e. depending on which error F_k has taken place, our state will be mapped from the code space to a corresponding error spaces.



2. Once we know "k", we can define our reversal map R by applying the corresponding $U_k^\dagger$. Here, note that $P_k = U_k P U_k^\dagger$, the P_k's being orthogonal. We therefore have:

$$\begin{aligned}
R[\cdot] &= \sum_k U_k^\dagger (P_k) \cdot (P_k) U_k && \qquad (12)\\
&= \sum_k U_k^\dagger (U_k P U_k^\dagger) \cdot (U_k P U_k^\dagger) U_k && \qquad (13)\\
&= \sum_k P U_k^\dagger \cdot U_k P && (U_k \text{ unitary}) \qquad (14)
\end{aligned}$$

The idea in the expression above is to first make a measurement with our P_k measurement operators, and then apply the necessary $U_k^\dagger$ operator.

Let us now doublecheck that our reversal map works as expected. To begin, we express $A_l P$ as:

$$A_l P = \sum_m b'_{lm} E_m P = \sum_m b_{lm} F_m P = \sum_m b_{lm}(\sqrt{d_{mm}} U_m P), \tag{15}$$

since we can write the E operators in terms of F operators, and using the definition of $F_k P$ from before.

Now apply R to our state, ρ, after it has undergone the quantum operation $\varepsilon(\cdot)$:

$$\begin{align}
R \circ \varepsilon(\rho) &= \sum_{kl} (PU_k^\dagger)(A_l P)\rho(PA_l^\dagger)(U_k P) \tag{16}\\
&= \sum_{kl} (PU_k^\dagger)(\sum_m b_{lm}\sqrt{d_{mm}} U_m P)\rho(\sum_{m'} b^*_{lm'}\sqrt{d_{m'm'}} PU_{m'}^\dagger)(U_k P) \qquad \text{(substitute (15))} \tag{17}\\
&= \sum_{klmm'} (PU_k^\dagger)(b_{lm}\sqrt{d_{mm}} U_m P)\rho(b^*_{lm'}\sqrt{d_{m'm'}} PU_{m'}^\dagger)(U_k P) \tag{18}\\
&= \sum_{klmm'} (b_{lm}\sqrt{d_{mm}} P(U_k^\dagger U_m)P)\rho(b^*_{lm'}\sqrt{d_{m'm'}} P(U_{m'}^\dagger U_k)P) \tag{19}\\
&= \sum_{klmm'} (b_{lm}\sqrt{d_{mm}}\delta_{km} P)\rho(b^*_{lm'}\sqrt{d_{m'm'}}\delta_{km'} P) \qquad \text{(U_k's unitary and orthogonal)} \tag{20}\\
&= \sum_{kl} (b_{lk}\sqrt{d_{kk}} P)\rho(b^*_{lk}\sqrt{d_{kk}} P) \tag{21}\\
&= \sum_{kl} (b_{lk} b^*_{lk} d_{kk}) P\rho P \tag{22}\\
&= \left(\sum_{kl} (b_{lk} b^*_{lk} d_{kk})\right)\rho \qquad \text{(by defn in Statement 2)} \tag{23}\\
&= [tr(\varepsilon(\rho))]\rho \tag{24}
\end{align}$$

The last line is derived from the fact that:

$$\begin{align}
tr(\varepsilon(\rho)) &= tr\left(\sum_l A_l P\rho P A_l^\dagger\right) \tag{25}\\
&= \sum_l tr[(PA_l^\dagger A_l P)\rho] \qquad \text{(trace is linear)} \tag{26}\\
&= \sum_l tr[(\sum_{m'} b^*_{lm'}\sqrt{d_{m'm'}} PU_{m'}^\dagger)(\sum_m b_{lm}\sqrt{d_{mm}} U_m P)\rho] \qquad \text{(substitute (15))} \tag{27}\\
&= \sum_{lmm'} b^*_{lm'} b_{lm}\sqrt{d_{m'm'}}\sqrt{d_{mm}}\delta_{mm'} tr(PP\rho) \qquad \text{(U_m's unitary and orthogonal)} \tag{28}\\
&= \sum_{lm} b^*_{lm} b_{lm} d_{mm} tr(P\rho P) \qquad \text{(trace is cyclic)} \tag{29}\\
&= tr(\rho)\sum_{lm} b^*_{lm} b_{lm} d_{mm} \qquad \text{(by defn in Statement 2)} \tag{30}\\
&= \sum_{lm} b^*_{lm} b_{lm} d_{mm}, \qquad \text{(tr(density op) is always 1)} \tag{31}
\end{align}$$

as required, thus proving $1 \Rightarrow 2$.

Now let us prove Statement $2 \Rightarrow$ Statement 1. As in assumptions for 2, we start with a CP (though not necessarily TCP) map:

$$\varepsilon[\cdot] = \sum_k E_k \cdot E_k^\dagger \tag{32}$$

By Statement 2, it is promised that we can find a reversal map R. Let us write it as follows:

$$R[\cdot] = \sum_l B_l \cdot B_l^\dagger \tag{33}$$

Therefore, we have, $\forall \rho$:

$$\begin{aligned} R \circ \varepsilon \circ P(\rho) &= \sum_{lk} B_l E_k P \rho P E_k^\dagger B_l^\dagger & (34) \\ &= tr[\varepsilon \circ P(\rho)] \cdot P(\rho) \qquad \text{(defn of } R \text{ in Statement 2)} & (35) \end{aligned}$$

Now, the RHS's of the above 2 equations are two different Kraus representations corresponding to the same quantum operation. Hence, by unitary freedom in operator-sum representation (Theorem 8.2, p372 from Nielsen and Chuang), we can relate the operation elements of the 2 RHS's as:

$$\forall_{l,k}\ B_l E_k P = f_{lk} P, \tag{36}$$

for some f_{lk}. We then have that

$$(P E_{k'}^\dagger B_l^\dagger)(B_l E_k P) = f_{lk'}^* f_{lk} P \tag{37}$$

Then summing over l gives us

$$\begin{aligned} P E_{k'}^\dagger E_k P &= \left(\sum_l f_{lk'}^* f_{lk} \right) P & (38) \\ &= c_{k'k} P \qquad \text{(C from Statement 1)} & (39) \end{aligned}$$

Finally, we need to prove that C is positive semi-definite. To see this, note that $c_{k'k}$ above is the $k'k$ entry of $F^\dagger F$, where F is the matrix having lk entry f_{lk}. But F is positive semi-definite, and hence so is C. □

1.1 Degeneracy

Using the same definitions as before, for our error set $\mathbb{E} = \{E_i\}$, if C is full rank, we say that the code space C_L is *non-degenerate*. Otherwise, we call it *degenerate*. This latter case essentially means that multiple errors affect our code in the same way (i.e. we don't have a unique mapping). In this degenerate case, we therefore have that $\{E_i P\}$ is linearly dependent.

1.2 Quantum Hamming Bound

For a *non-degenerate* code s.t. C_L encodes k qubits in n qubits, the Quantum Hamming Bound states that

$$\begin{aligned} 2^n &\geq 2^k \cdot |\mathbb{E}|, \text{or by taking the log of both sides,} & (40) \\ n - k &\geq log|\mathbb{E}| & (41) \end{aligned}$$

Note that here 2^k is the dimension of the code space, and $|\mathbb{E}|$ is the number of errors. If C is full rank, then $|\mathbb{E}|$ is also the number of distinguishable errors. We can think of this statement as saying that, in the non-degenerate case, each error gives us a 2^k-dimension subspace of the 2^n-dimension total space.

1.2.1 Example: 7-qubit code, t=1, n=7

Note that t is the number of errors our code corrects. Here we have $|\mathbb{E}| = 1 + \binom{7}{1} \cdot 3 = 22$, where the 1 handles the case of no error or the identity being applied,$\binom{7}{1}$ tells us which of the 7 qubits the error affected, and 3 indicates which error occured. We thus have

$$\begin{aligned} 2^7 &\geq 2^k \cdot 22 && (42)\\ 7-k &\geq log(22) && \text{(apply log to both sides)} \quad (43)\\ k &\leq 7 - log(22) && (44)\\ k &\leq 2.54 && (45) \end{aligned}$$

Therefore our code can encode up to 2 qubits worth of data.

1.2.2 Example: 5-qubit code, t=1, n=5

We have $|\mathbb{E}| = 1 + \binom{5}{1} \cdot 3 = 16$ this time. Thus,

$$\begin{aligned} 2^5 &\geq 2^k \cdot 16 && (46)\\ 5-k &\geq log(16) && \text{(apply log to both sides)} \quad (47)\\ k &\leq 5 - log(16) && (48)\\ k &\leq 1 && (49) \end{aligned}$$

In this case, we can encode 1 qubit of data. Note that we can achieve equality in the bound this time (i.e. no decimal portion in the right hand side), meaning the 5-qubit code saturates the Hamming Bound. In such a case, we call the code *perfect*.

1.3 Quantum Singleton Bound for [n,k,d] Code

Suppose we have an n-qubit code. If we could correct $\geq \frac{n}{2}$ erasure errors with it, then we could clone states, violating the No-Cloning Theorem.

To see this, imagine that we have an initial encoded state of length n, $|\psi\rangle$. Take the first $\frac{n}{2}$ qubits. Since we can correct for $\geq \frac{n}{2}$ erasure errors, we can recover the last $\frac{n}{2}$ qubits that have "been erased", and hence we can recover the entire state. We can then repeat the procedure with the last $\frac{n}{2}$ qubits of our initial state, $|\psi\rangle$, obtaining a second copy of $|\psi\rangle$. Thus, we now have two copies of $|\psi\rangle$ in our possession, contradicting the No-Cloning Theorem.

Along similar directions, and following a more careful proof, (see Gottesman's lecture notes in CO639, year 2004) we can derive the Quantum Singleton Bound, which states that

$$n - k \geq 2(d-1) \tag{50}$$

1.4 Stabilizer Codes

1.4.1 Definitions and Properties

Let P_n denote the n-qubit Pauli group. This is defined as the n-fold tensor product of all the Pauli matrices (I, X, Y, Z). Note that in our analysis, we often omit the phase factors ± 1 and $\pm i$.

Definition 1 *Let G be a subset of P_n s.t. $G = \{S_i\}_{i=1}^{n-k}$, $S_i \in P_n$, and $S =< G >$ is the subgroup generated (multiplicatively) by G, with the following two conditions:*

1. *$[S_i, S_j] = 0$ if $i \neq j$ (elements of S commute).*

2. *$-I$ is not an element of S*

*Then $S =< G >$ is called the **stabilizer** of a non-trivial vector space, V_S. This vector space V_S is precisely the intersection of the vector spaces stabilized by each of the elements of G, or equivalently, the intersection of the $+1$ eigenspaces of each of the elements of S (remember that the $+1$ eigenspace of element S_i is the set of vectors $|\psi_i\rangle$ s.t. $S_i|\psi_i\rangle = |\psi_i\rangle$, which is exactly what we want from a stabilizer).*

Properties:

1. *Representation:* Multiplying one $S_i \in G$ by another $S_j \in G$ does not change S This follows from the fact that G generates S.

2. *Unitary Transformation:* $\forall U \in \mathbb{U}(2^n), \forall|\psi\rangle \in C_L, \forall i,$

$$U|\psi\rangle = US_i|\psi\rangle = (US_iU^\dagger)U|\psi\rangle, \tag{51}$$

   since $U^\dagger U = I$. Therefore $U|\psi\rangle$ is stabilized by $US_iU^\dagger$. Thus, the new stabilizer is $USU^\dagger$.

3. *Measurements of some $P \in P_n$:*

   - Case 1: $P \in S$. In this case, we have already been promised that any state $|\psi\rangle \in C_L$ is in the $+1$ eigenstate of each element in S. Thus, we will always get the outcome corresponding to the +1 eigenvalue (otherwise $|\psi\rangle$ is not in the code), the state of the system will remain unaltered, and so our stabilizer also remains unaltered.
   - Case 2: $P \notin S$. We will deal with this case later.

4. **Definition 2** *Assume we have a subgroup S of G_n, $-I \notin S$, and $S =< g_1, ..., g_{n-k} >$, where $g_1, ..., g_{n-k}$ are independent and commuting generators. Then the vector space V_S stabilized by S is called an **[n,k] stabilizer code**, and is denoted C(S).*

   Now, let $\Delta \in P_n$ be an error corrupting our stabilizer code $C(S)$. We then have three possibilities regarding Δ:

   - $\Delta \in S$: In this case, Δ actually stabilizes our encoded state, hence leaving it unaltered. This means there's no "error" to correct.
   - Δ anti-commutes with an element of S: Δ then maps C(S) to an orthogonal subspace, so we can apply the appropriate projective measurement to detect the error.
   - Δ commutes with all elements of S, but $\Delta \notin S$: This implies that $\Delta \in N(S) - S$, where $N(S)$ is what is known as the *normalizer* of the stabilizer S, s.t. $N(S) = \{M \in P_n : MSM^\dagger \subseteq S\}$,
     where we can replace $\subseteq$ by $=$ because conjugation is reversible. Here $N(S)$ contains $2k$ generators. Δ's like this correspond to non-trivial logical operators, which will prove important later for fault tolerance.

Theorem 3 *:*

1. *Let $\mathbb{E} = \{E_i\}$ be our set of errors, with $E_i \in P_n$, P_n the Pauli group on n qubits. If $\forall i, j$ $E_i^\dagger E_j \in S \cup \overline{N(S)}$, then QECC condition 1 is satisfied. Here, $\overline{N(S)}$ is the complement of the normalizer of S.*

2. *The code is degenerate $\Leftrightarrow \exists i \neq j$ s.t. $E_i^\dagger E_j \in S$*

Proof :

Aside:

$$PS_i = S_iP = P \tag{52}$$

where P is a projector onto our code space, and S_i is an element in our stabilizer. This follows from the fact that S_i stabilizes the vector space P projects onto.

Let us begin by assuming statement 1 in the theorem above, i.e. we have $E_i^\dagger E_j \in S \cup \overline{N(S)}$.

Case 1: If $E_i^\dagger E_j \in S$, then $PE_i^\dagger E_j P = PP = P$ (by (52)), so $c_{ij} = 1$.

Case 2: If we're not in Case 1, we must be in Case 2 by our assumption, i.e. $E_i^\dagger E_j \notin N(S)$. By Property 4 of stabilizers earlier, $E_i^\dagger E_j$ must therefore anti-commute with an element of S. So $\exists S_k$ s.t. $\{S_k, E_i^\dagger E_j\} = 0$. We can therefore write

$$\begin{aligned}
PE_i^\dagger E_j P &= PE_i^\dagger E_j S_k P && \text{(by (52))} && (53)\\
&= -PS_k E_i^\dagger E_j P && (S_k \text{ and } E_i^\dagger E_j \text{ anticommute}) && (54)\\
&= -PE_i^\dagger E_j P && \text{(by (52))} && (55)\\
&= 0 && (x = -x \Rightarrow x = 0) && (56)\\
&= 0 \cdot P && && (57)\\
\Rightarrow c_{ij} &= 0 && && (58)
\end{aligned}$$

So to summarize, we have:

$$\begin{aligned}
c_{ij} &= 1 \text{ if } E_i^\dagger E_j \in S\\
c_{ij} &= 0 \text{ if } E_i^\dagger E_j \in \overline{N(S)}
\end{aligned}$$

It remains to show $C \geq 0$. For this, we have that if $E_i^\dagger E_j \in S$ and $E_j^\dagger E_k \in S$, then $E_i^\dagger E_k \in S$. This follows from the fact that, for some state $|\psi\rangle \in C_L$,

$$\begin{aligned}
|\psi\rangle &= E_i^\dagger E_j |\psi\rangle && (E_i^\dagger E_j \in S) && (59)\\
&= E_i^\dagger E_j (E_j^\dagger E_k)|\psi\rangle && (E_j^\dagger E_k \in S) && (60)\\
&= E_i^\dagger E_k |\psi\rangle && (E_j \in P_n \text{ unitary}) && (61)\\
& && && (62)
\end{aligned}$$

In this case, since $c_{ij} = 1$ and $c_{jk} = 1$, we have $c_{ik} = 1$. Thus, $C \geq 0$ as claimed. □

1.4.2 Examples of Stabilizer Codes

The 5-Qubit Code

The 5-qubit stabilizer code is the smallest code that can detect and correct an error on a single qubit. Its stabilizer generators are:

$$S_1 = XZZXI \tag{63}$$

$$S_2 = IXZZX \tag{64}$$

$$S_3 = XIXZZ \tag{65}$$

$$S_4 = ZXIXZ \tag{66}$$

where $XZZXI$ is shorthand for $X \otimes Z \otimes Z \otimes X \otimes I$. Thus, any element in the group generated by these operators stabilizes any state in our code space. Notice that if we take any 5-qubit operator of weight at most 2 and composed of the Pauli matrices, it will anti-commute with at least one of the generators above (remembering that weight is defined as the number of non-identity terms in the 5-fold tensor product that makes up the operator). For example, take $IIIXI$, which has weight 1. This anti-commutes with S_2 since

$$(IXZZX)(IIIXI) + (IIIXI)(IXZZX) = (iIXZYX)(-iIXZYX) = 0 \tag{67}$$

An example of weight 2 is $IIIYZ$, which anti-commutes with S_1. Therefore if we measure our error state using our generators as observables, we will be able to detect the error (as stated earlier under properties of the stabilizer).

Note that the logical X and Z, denoted $\overline{X} = XXXXX$ and $\overline{Z} = ZZZZZ$, respectively, commute with every generator above, but are *not* in the stabilizer itself. They are hence in the normalizer of S.

The 9-Qubit Shor Code
The 9-bit stabilizer code has generators:

$$S_1 = \texttt{XXX XXX III} \tag{68}$$
$$S_2 = \texttt{III XXX XXX} \tag{69}$$
$$S_3 = \texttt{ZZI III III} \tag{70}$$
$$S_4 = \texttt{IZZ III III} \tag{71}$$
$$S_5 = \texttt{III ZZI III} \tag{72}$$
$$S_6 = \texttt{III IZZ III} \tag{73}$$
$$S_7 = \texttt{III III ZZI} \tag{74}$$
$$S_8 = \texttt{III III IZZ} \tag{75}$$

This code is capable of correcting any arbitraty single qubit error (similar argument as above for the 5-qubit code). It is, however, de-generate. To see this, note that for distinct errors $ZIIIIIIII$ and $IZIIIIIII$, we will get the same error syndrome when measuring with the generator $S_3 = ZZIIIIIII$. This implies that both errors can be corrected with the same recovery operation.

2 Degenerate Codes

Graeme Smith, September 29, 2006

From last time, for a set of errors ε, the Quantum Hamming Bound states that

$$k \leq n - log|\varepsilon| \tag{76}$$

for any $[n, k]$ non-degenerate code, k denotes the number of encoded qubits, and n denotes the block length.

So what's a non-degenerate code? Simply a code in which each error $E \in \varepsilon$ gets its own error syndrome. A degenerate code is then the opposite, where many $E \in \varepsilon$ can have the same error syndrome. In the case where there is "a lot" of such duplicate mapping, the code is sometimes called "grossly degenerate".

Idea: Try to beat the Hamming Bound using degenerate codes (i.e. try to encode more qubits than the bound allows for non-degenerate codes).

2.1 The Depolarizing Channel

The depolarizing channel acts on a given input state ρ as follows:

$$\mathcal{N}_p(\rho) = (1-p)\rho + \frac{p}{3}X\rho X + \frac{p}{3}Y\rho Y + \frac{p}{3}Z\rho Z \tag{77}$$

i.e. with probability $1-p$, the state remains unaltered, and with probability $\frac{p}{3}$ we apply X, Y, or Z. Extending this to an n-qubit state ρ_n gives:

$$\mathcal{N}_p^{\otimes n}(\rho_n) = \sum_{\vec{u},\vec{v}} p_{\vec{u}\vec{v}} X^{\vec{u}} Z^{\vec{v}} \rho_n Z^{\vec{v}} X^{\vec{u}} \tag{78}$$

$$\begin{aligned} \text{w.h.p., } \#I &\sim (1-p)n && (79)\\ \#X &\sim \frac{p}{3}n && (80)\\ \#Y &\sim \frac{p}{3}n && (81)\\ \#Z &\sim \frac{p}{3}n && (82)\\ & && (83) \end{aligned}$$

We are therefore going to try and correct errors of the form:

$$\varepsilon \approx \{X^{\vec{u}}Z^{\vec{v}}| \text{ s.t. } n_I, n_X, n_Y, n_Z \sim ((1-p), \frac{p}{3}, \frac{p}{3}, \frac{p}{3})\} \tag{84}$$

where n_I, n_X, n_Y, and n_Z are the number of I, X, Y, and Z errors, respectively. We then have

$$|\varepsilon| \approx \binom{n}{pn} 3^{pn} = 2^{n(H(p)+p\log_2 3)}, \tag{85}$$

where the $\binom{n}{pn}$ accounts for the number of different places to put the error, the 3^{pn} denotes which error occurred, and the second inequality comes from Sterling's Approximation, which states that $\binom{n}{np} \approx 2^{nH(p)}$. The Hamming Bound tells us that $R \leq 1 - H(p) - p\log_2 3$.

2.1.1 CSS-Hashing

We are going to go off topic briefly to discuss a trick in information theory for picking a random code called *CSS-Hashing* which we will later need.

<u>Idea</u>: First correct all the amplitude errors (X), then correct the phase errors (Z).

We start with $\{Z^{\vec{a}_l}\}_{l=1}^{n-k}$, which are the syndromes we've measured, with $\vec{a}_l \in_R \{0,1\}^n$ being random strings. Note that the syndromes will be $\omega(X^{\vec{u}}, Z^{\vec{a}_l}) = \vec{u} \cdot \vec{a}_l$. The typical number of amplitude errors (i.e. the number of appearances of X in the error string) will be about $\frac{2p}{3}$. So, after doing our parity check, what's the probability of incorrectly identifying our error?

$$Pr[\text{amp. error}] = \sum_{\vec{u}=\varepsilon_{typical}} p_{\vec{u}} Pr[\exists \vec{u}' \in \varepsilon_{typ}, \vec{u}' \neq \vec{u}, \vec{a}_l \cdot \vec{u}' = \vec{a}_l \cdot \vec{u} \text{ for } l = 1...n-k] \tag{86}$$

$$\text{(approximately)} \leq \frac{|\varepsilon_{typ}|}{2^{nH(\frac{2p}{3})}} Pr[\exists \vec{u}' \in \varepsilon_{typ}, \vec{u}' \neq \vec{u}, \vec{a}_l \cdot \vec{u}' = \vec{a}_l \cdot \vec{u} \text{ for } l = 1...n-k] \tag{87}$$

$$= \frac{|\varepsilon_{typ}|}{2^{nH(\frac{2p}{3})}} Pr\left[\bigcup_{\vec{u}' \in (\varepsilon|\vec{u})} \{\vec{a}_l \cdot \vec{u}' = \vec{a}_l \cdot \vec{u}\}\right] \tag{88}$$

$$\leq \frac{|\varepsilon_{typ}|^2}{2^{nH(\frac{2p}{3})}} Pr[\vec{a}_l \cdot (\vec{u} + \vec{u}') = 0] \tag{89}$$

Here we have for $\vec{a} \in_R \{0,1\}^n$ and $\vec{x} \neq \vec{0}$, $Pr[\vec{a} \cdot \vec{x} = 0] = \frac{1}{2}$. Therefore, use this in the $Pr[]$ expression above to get:

$$\leq |\varepsilon_{typ}| \cdot \frac{1}{2^{n-k}} \qquad\qquad (|\varepsilon_{typ}| = 2^{nH(\frac{2p}{3})}) \tag{90}$$

$$\leq \frac{2^{nH(\frac{2p}{3})}}{2^{n-k}} \tag{91}$$

Now let $k \leq n(1 - H(\frac{2p}{3} - \delta))$. Therefore

$$Pr[\text{amp. error}] \leq \frac{1}{2^{\delta \cdot n}} \tag{92}$$

Now, given $\vec{u}$, $|\varepsilon_{typ}^{\vec{u}}| \sim 2^{nH(u|v)}$, where $\varepsilon_{typ}^{\vec{u}}$ are the errors conditional on $\vec{u}$, and $H(u|v)$ is the conditional entropy. So choose $\vec{x}^{\vec{b}_l}$, $l = 1...k - k_2$, and choose enough to distinguish between all elements of the typical errors conditioned on $\vec{u}$. This forces you to choose

$$k_2 = k - kH(z|x) \tag{93}$$

$$= (n-k)(1 - H(z|x)) \tag{94}$$

$$= n(1 - H(p) - plog3) \tag{95}$$

How does this mapping from typical phase errors to phase errors on the logical space go?

$$\vec{v} \to \vec{v}_L \tag{96}$$

$$\text{Say } \vec{v_1}, \vec{v_2} \to \vec{v}_L \tag{97}$$

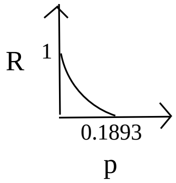
For now, let's just worry about these vectors. Now, $|L(\varepsilon_{typ}^{\vec{u}})| \to \leq 2^{nH(z|x)}$,
and $L : \{0,1\}^n \to \{0,1\}^k \approx \{0,1\}^n \backslash \{A\}$, where the domain of L is n-bit phase errors,
and $A = span(\vec{a_l})$. We then have

$$L\vec{v_1} = L\vec{v_2} \Leftrightarrow Z^{\vec{v_1} \oplus \vec{v_2}} \in < Z^{\vec{a_l}} >, \tag{98}$$

where $Z^{\vec{a_l}}$ are the parity checks, and $< Z^{\vec{a_l}} >$ denotes the stabilizer (i.e. the group generated by the parity checks). Note that we also have $\vec{v_1} + \vec{v_2} \in span(\vec{a_l})$. So, we're trying to figure out the typical weight of $\vec{v_1} \oplus \vec{v_2}$ at $p \approx 0, 1893$. Hence

$$\frac{2pn}{3} - wt = \frac{2p}{3}(1 - \frac{2p}{3})n \sim \frac{1}{8}n \tag{99}$$

This gives us the following sketch of a curve, with bound 0.1893. We are interested in improving this bound.



So far it has been shown, for example, that it can be improved to 0.1903 using degenerate codes.

Ok, now turning back from our slight digression... So let's start by looking at a "normal" "non-random" code. Let us look at a repetition $[5, 1]$ code, with stabilizer generators $Z_1Z_2, Z_1Z_3, Z_1Z_4, Z_1Z_5$. Encoded states are of the form $\alpha|0_L\rangle^{\otimes 5} + \beta|1_L\rangle^{\otimes 5}$, where $|0_L\rangle$ and $|1_L\rangle$ are the logical 0 and 1 states, respectively. We also have that the logical zero $\overline{X} = XXXXX$, and $\overline{Z} = ZIIII$ or $IZIII$. This code is good at correcting amplification error, but bad with phase errors. Finally, for this code, we also have that

$$\vec{S} = \begin{bmatrix} u_1 \oplus u_3 \\ u_1 \oplus u_3 \\ u_1 \oplus u_4 \\ u_1 \oplus u_5 \end{bmatrix} \tag{100}$$

$$X^{u_1} Z^{(\bigoplus_{l=1}^{m} v_l)} \tag{101}$$

Idea: Now let's concatenate this code with a random code to get a 5n-qubit code to see what we get.

Again, we're going to have a repetition code with blocks of 5. This time let's also choose a random amp. code.

$$\overline{Z}^{\vec{a_l}}, \vec{a_l} \in_R \{0, 1\}^n \tag{102}$$

So what's the typical weight of $\vec{a_l}$ when chosen at random? The answer is $\frac{n}{2}$ (odds are about half our digits will be 1's), and so $|\vec{a_l}| = \frac{n}{2}$. But $wt(\overline{Z}^{\vec{a_l}}) = \frac{n}{2}$ as well, where $\overline{Z}^{\vec{a_l}}$ is now a 5n-bit operation. Therefore

$$wt(\overline{Z}^{\vec{a_l}}) \sim \frac{n}{2} \Rightarrow \frac{1}{10} \sim 1's, \frac{9}{10}0's, \tag{103}$$

i.e. $wt(\overline{Z}^{\vec{a_l}})$ is close to the typical value for $\vec{v}_1 \oplus \vec{v}_2$.