Ph219/CS219 Problem Set 5

Solutions by Hui Khoon Ng

February 7, 2007

5.1 Correcting a shift

(a) Consider for any $|j\rangle$:

$$\begin{aligned}[M_X, M_Z]\,|j\rangle &= (X^{nr_1}Z^{nr_2} - Z^{nr_2}X^{nr_1})|j\rangle \\ &= X^{nr_1}(\omega^j)^{nr_2}|j\rangle - Z^{nr_2}|j + nr_1(\text{mod } d)\rangle \\ &= \left[\omega^{jnr_2} - (\omega^{j+nr_1})^{nr_2}\right]|j + nr_1(\text{mod } d)\rangle \\ &= \omega^{jnr_2}\left[1 - (\omega^{nr_1r_2})^n\right]|j + nr_1(\text{mod } d)\rangle,\end{aligned}$$

but $d = nr_1r_2$, so $\omega^{nr_1r_2} = \omega^d = \exp(2\pi i) = 1$. This gives

$$[M_X, M_Z]\,|j\rangle = 0 \qquad \forall j \qquad \Rightarrow \qquad [M_X, M_Z] = 0.$$

(b) We are given $ZX = \omega XZ$, which we can use to work out commutation relations of powers of X and Z. Consider Z^kX^l:

$$Z^kX^l = \underbrace{ZZ\ldots Z}_{k}\,\underbrace{XX\ldots X}_{l} = \omega^k X\,\underbrace{ZZ\ldots Z}_{k}\,\underbrace{XX\ldots X}_{l-1} = \omega^{kl}X^lZ^k.$$

Using this, we have

$$\begin{aligned} M_X(X^aZ^b) &= X^aX^{nr_1}Z^b = \bar{\omega}^{bnr_1}(X^aZ^b)M_X, \\ \text{and} \qquad M_Z(X^aZ^b) &= Z^{nr_2}X^aZ^b = \omega^{anr_2}(X^aZ^b)M_Z.\end{aligned}$$

(c) We want the generators of the normalizer group $S^\perp$ which consists of all Pauli operators X^aZ^b that commute with M_X and M_Z. From our answers in part (b), we know that X^aZ^b commutes with both M_X and M_Z iff a and b satisfy:

$$\begin{aligned}\omega^{anr_2} = 1 &\Rightarrow \exp\left(2\pi i\frac{anr_2}{d}\right) = \exp\left(2\pi i\frac{a}{r_1}\right) = 1 \Rightarrow \underline{a = kr_1,\ k \in \mathbb{Z}}, \\ \bar{\omega}^{bnr_1} = 1 &\Rightarrow \exp\left(-2\pi i\frac{bnr_1}{d}\right) = \exp\left(-2\pi i\frac{b}{r_2}\right) = 1 \Rightarrow \underline{b = lr_2,\ l \in \mathbb{Z}}.\end{aligned}$$

Therefore, $S^\perp = \{X^{kr_1}Z^{lr_2} : k, l \in \mathbb{Z}\}$, and the two generators are $\tilde{X} := X^{r_1}$ and $\tilde{Z} := Z^{r_2}$. They satisfy

$$\tilde{Z}\tilde{X} = Z^{r_2}X^{r_1} = \omega^{r_1 r_2}X^{r_1}Z^{r_2} = \tilde{\omega}\tilde{X}\tilde{Z}$$

with $\tilde{\omega} := \omega^{r_1 r_2} = \exp(2\pi i/n)$. Notice that this is exactly the commutation relation eq. (3), with $d \to n$, $X \to \tilde{X}$, $Z \to \tilde{Z}$ and $\omega \to \tilde{\omega}$, and since the normalizer group acts as the logical operations on the codespace, we suspect that there is one encoded qu$n$it. Let us confirm this by explicitly constructing the codespace.

Recall that the codespace is the $+1$ simultaneous eigenspace of the stabilizer generators. Now, $M_X|j\rangle = |j + nr_1 (\mathrm{mod}\ d)\rangle$, i.e. M_X shifts j by nr_1, so the $+1$ eigenstates of M_X must be uniform superpositions of states from the cosets

$$C_j := \{|j + mnr_1 (\mathrm{mod}\ d)\rangle : m = 0, 1, \ldots, r_2 - 1\}$$

with $j = 0, 1, \ldots, nr_1 - 1$. So, the $+1$ eigenspace of M_X is spanned by the orthonormal states $\{|\psi_j\rangle\}_{j=0}^{nr_1-1}$ where

$$|\psi_j\rangle = \frac{1}{\sqrt{|C_j|}}\sum_{m=0}^{r_2-1}|j + mnr_1\rangle = \frac{1}{\sqrt{r_2}}\sum_{m=0}^{r_2-1}|j + mnr_1\rangle.$$

Within this $+1$ eigenspace of M_X, we want the $+1$ eigenstates of M_Z. Now,

$$M_Z|\psi_j\rangle = \frac{1}{\sqrt{r_2}}\sum_{m=0}^{r_2-1}\left(\omega^{j+mnr_1}\right)^{nr_2}|j + mnr_1\rangle = \omega^{jnr_2}|\psi_j\rangle,$$

so $|\psi_j\rangle$ is a $+1$ eigenstate of M_Z if $\omega^{jnr_2} = \exp(2\pi i j/r_1) = 1$, i.e. $j = kr_1, k \in \mathbb{Z}$. Since $j = 0, 1, \ldots, nr_1 - 1$, there are exactly n values of j that fulfill this condition, each of which gives an orthonormal state $|\psi_j\rangle$. This confirms that the codespace is n-dimensional, i.e. one encoded qu$n$it.

(d) The distance d_X of the code for X errors is given by the minimum weight of the X-type operators in $S^\perp \backslash S$, so $d_X = \mathrm{wt}(\tilde{X}) = r_1$. Therefore, this code can correct an amplitude shift (X errors) of up to $|a| = \frac{d_X - 1}{2} = \frac{r_1 - 1}{2}$. Similarly, $d_Z = \mathrm{wt}(\tilde{Z}) = r_2$, so the largest phase shift (Z errors) this code can correct is $|b| = \frac{d_Z - 1}{2} = \frac{r_2 - 1}{2}$.

5.2 Polynomial CSS codes

(a) To prove that C_1 is a vector space, you can prove all the vector space axioms directly on C_1. Alternatively, you can note that there is an equivalence between vectors of C_1 and the vectors of the $m + 1$ coefficients of polynomials of degree m. More formally, given $\{x_0, x_1, \ldots, x_{n-1}\}$, there is an isomorphism between C_1 and a more familiar set $\mathbb{F}_p^{m+1}$ defined as

$$\mathbb{F}_p^{m+1} := \{\vec{a} = (a_0, a_1, \ldots, a_m) : a_i \in \mathbb{F}_p\}$$

with addition mod p, and scalar multiplication over the field $\mathbb{F}_p$. $\mathbb{F}_p^{m+1}$ is closed under addition mod p, has identity $\vec{0}$ and has inverses $(\vec{a})^{-1} = (a_0^{-1}, a_1^{-1}, \ldots, a_m^{-1})$ since $\mathbb{F}_p$ is a field (and thus a_i^{-1} exists). Other vector space axioms are also clearly satisfied, so $\mathbb{F}_p^{m+1}$ is a vector space over $\mathbb{F}_p$. The isomorphism $\varphi : \mathbb{F}_p^{m+1} \longrightarrow C_1$ acts as

$$\varphi(\vec{a}) = (f_{\vec{a}}(x_{n-1}), f_{\vec{a}}(x_{n-2}), \ldots, f_{\vec{a}}(x_0))$$

where $f_{\vec{a}}(x)$ is the polynomial $a_0 + a_1 x + \ldots + a_m x^m$. φ clearly respects vector addition and scalar multiplication. Since C_1 is isomorphic to a vector space $\mathbb{F}_p^{m+1}$, it is also a vector space.

(b) A nonzero polynomial f of degree m has at most m zeros over $\mathbb{F}_p$, i.e. for any $f(x)$ defining a vector in C_1, there exist at most m distinct x_i's for which $f(x_i) = 0$. Therefore, since $x_0, x_1, \ldots, x_{n-1}$ are all distinct, each vector (length $= n$) $(f(x_{n-1}), \ldots, f(x_0)) \neq 0$ in C_1 has at most m zero entries and thus have weight at least $n - m$. This implies $d_1 \geq n - m$.

(c) C_2 is isomorphic to $\mathbb{F}_p^m$ in the similar way as discussed in part (a). $\mathbb{F}_p^m$ is a vector space, by the argument in (a) (with $m \to m-1$), so C_2 is a vector space. It is clearly a subspace of C_1 since it consists of those (and only those) vectors in C_1 with f such that the mth degree coefficient is zero.

(d) We want a degree $m-1$ polynomial f such that $f(z_i) = y_i \forall i$, i.e. we want to fit an $(m-1)$-degree polynomial through m distinct points. This is just Lagrange interpolation - through any 2 points, there is a unique line, through any 3 points, there is a unique quadratic, ..., through any m points, there is a unique degree $m-1$ polynomial. f can be constructed explicitly as

$$\begin{aligned} f(z) = &\frac{(z-z_2)(z-z_3)\ldots(z-z_m)}{(z_1-z_2)(z_1-z_3)\ldots(z_1-z_m)} y_1 + \frac{(z-z_1)(z-z_3)\ldots(z-z_m)}{(z_2-z_1)(z_2-z_3)\ldots(z_2-z_m)} y_2 \\ &+ \ldots + \frac{(z-z_1)(z-z_2)\ldots(z-z_{m-1})}{(z_m-z_1)(z_m-z_2)\ldots(z_m-z_{m-1})} y_m \\ = &\sum_{i=1}^{m} y_i \left(\prod_{j\neq i} \frac{z-z_j}{z_i-z_j} \right). \end{aligned}$$

Notice that only the ith term in the sum is nonzero (equals 1) for $f(z_i)$, and so $f(z_i) = y_i \forall i$ as desired. Furthermore, the denominators are well-defined since the z_i's are all distinct, and f is of degree $m-1$ since each term in the sum consists of a product of exactly $m-1$ factors of $(z - z_j)$ in the numerator.

(e) The dual code of C_2 is defined as

$$C_2^{\perp} = \{\vec{u} := (u_{n-1}, \dots, u_1, u_0) : u_i \in \mathbb{F}_p,\ \vec{u} \cdot \vec{v} = 0 \quad \forall \vec{v} \in C_2\}.$$

Pick m different indices from 0 to $n-1$ and write this set as $\mathcal{I} := \{i_\alpha\}_{\alpha=1}^m$, $i_1 < i_2 < \dots < i_m$. Consider, $\forall \vec{v} \in C_2$, the projection $\vec{v} = (f(x_{n-1}), \dots, f(x_0)) \mapsto \vec{v}\big|_{\mathcal{I}} := (f(x_{i_m}), \dots, f(x_{i_\alpha}), \dots, f(x_{i_1})) \in \mathbb{F}_p^m$. Let $z_\alpha := x_{i_\alpha} \forall \alpha = 1, \dots, m$. Recall that all the x_i's are distinct. Now, pick any vector $\vec{y} = (y_m, y_{m-1}, \dots, y_1)$ in $\mathbb{F}_p^m$. From part (d), we know that there exists a polynomial of degree $m-1$, call it f_y such that

$$f_y(z_\alpha) = y_\alpha, \quad \forall \alpha = 1, 2, \dots, m,$$

so $\vec{y}$ is exactly the projection of the vector $(f_y(x_{n-1}), \dots, f_y(x_{i_m}), \dots, f_y(x_{i_1}), \dots, f_y(x_0)) \in C_2$ into $\mathbb{F}_p^m$. Since $\vec{y}$ was an arbitrary vector in $\mathbb{F}_p^m$, the projection of C_2 according to $\mathcal{I}$ into $\mathbb{F}_p^m$ is the whole space $\mathbb{F}_p^m$. This is true for any choice of the projection indices $\mathcal{I}$.

Now, suppose a nonzero vector $\vec{u} \in C_2^{\perp}$ has at most m nonzero components. Then, $\vec{u}$ can be thought of as a vector $\vec{u}' \in \mathbb{F}_p^m$, by discarding $n-m$ zero components. Let us choose $\mathcal{I}$ such that it contains the indices of all the nonzero components of $\vec{u}$. From our argument above, there exists a nonzero $\vec{v} \in C_2$ such that $\vec{v}\big|_{\mathcal{I}} = \vec{u}'$, and so $\vec{v}\big|_{\mathcal{I}} \cdot \vec{u}' = \vec{v} \cdot \vec{u} \neq 0$. Hence, there cannot be such a vector $\vec{u}$ in $C_2^{\perp}$, i.e. every nonzero vector in $C_2^{\perp}$ must have at least $m+1$ nonzero components, which implies $d_2 \geq m+1$.

(f) Recall the definition of a coset: $\vec{u}, \vec{v} \in C_1$ belong to the same coset of C_2 iff $\vec{u} - \vec{v} \in C_2$. C_1 consists of vectors constructed from degree m polynomials, so $\vec{u}, \vec{v} \in C_1$ can differ by a vector in C_2, which is constructed from degree $m-1$ polynomials, iff $\vec{u}$ and $\vec{v}$ have the same coefficient for x^m. Since the coefficients are chosen from $\mathbb{F}_p$, there are exactly p distinct possibilities for the coefficient of x^m, and hence there are exactly p distinct cosets.

Alternatively, you could have recalled Lagrange's theorem: the number of distinct cosets of C_2 in C_1 is $|C_1|/|C_2| = |\mathbb{F}_p^{m+1}|/|\mathbb{F}_p^m| = p^{m+1}/p^m = p$.

Therefore, the number of encoded qu$p$its is $\log_p(\text{dim. of code space}) = \log_p(\text{number of distinct cosets}) = \log_p p = 1$.

(g) We want to correct $t = (d-1)/2$ errors, and for a CSS code, $d = \min(d_1, d_2)$, so we require $d_1 \geq 2t+1$ and $d_2 \geq 2t+1$. From parts (b) and (e), we know $d_1 \geq n-m$ and $d_2 \geq m+1$, hence it suffices to impose

$$\begin{aligned} d_1 \geq n - m \geq 2t+1 \quad &\Rightarrow \quad n \geq m + 2t + 1, \\ d_2 \geq m + 1 \geq 2t+1 \quad &\Rightarrow \quad m \geq 2t. \end{aligned}$$

Suppose we take $m = 2t$, then $n \geq 4t+1$, so we can also take $n = 4t+1$ as required by the question. Since $n \leq p$, such a code can only be constructed for $p \geq 4t+1$ and p prime.

5.3 Decoherence-free subspaces and noiseless subsystems

(b) We want to find three two-qubit Hermitian operators $\bar{X}, \bar{Y}, \bar{Z}$ satisfying $\bar{X}^2 = \bar{Y}^2 = \bar{Z}^2 = I \otimes I \equiv I$, $\bar{X}\bar{Y} = -\bar{X}\bar{Y} = i\bar{Z}$, and also commute with $X \otimes X$, the only non-trivial operator in $\mathcal{E}$.

Suppose we take $\bar{X} = X \otimes I$. This is clearly Hermitian, squares to I and commutes with $X \otimes X$. We want another operator that anticommutes with $\bar{X}$ but commutes with $X \otimes X$. One possibility is $Z \otimes Z$. It clearly squares to I and satisfies the required commutation relations. Let $\bar{Z} = Z \otimes Z$.

The third operator ($\bar{Y}$) must be chosen so that $\bar{X}\bar{Y} = i\bar{Z}$, i.e. $\bar{Z}\bar{X} = i\bar{Y}$ since $\bar{Z}^2 = I$ and we need $\bar{Y}^2 = I$. Multiplying our chosen $\bar{Z}$ and $\bar{X}$ together, we get

$$\bar{Z}\bar{X} = (Z \otimes Z)(X \otimes I) = (ZX) \otimes Z = i(Y \otimes Z).$$

Notice that the operator $Y \otimes Z$ on the RHS is Hermitian and squares to I. Furthermore, it anticommutes with $\bar{X}$ and $\bar{Z}$, but commutes with $X \otimes X$. Therefore, it can be taken as $\bar{Y}$.

Altogether then, we have $\bar{X} = X \otimes I$, $\bar{Y} = Y \otimes Z$ and $\bar{Z} = Z \otimes Z$, and these are the logical Pauli operators of the one-qubit NS for $\mathcal{E}$.

(c) Note that $\mathcal{E}$ is invariant under permutation of qubits, i.e. the noise does not differentiate amongst the qubits. This tells us that the logical operations $\bar{X}$, $\bar{Y}$ and $\bar{Z}$, written as three-qubit operators, must also be permutation invariant. This immediately suggests three candidates: $X \otimes X \otimes X$, $Y \otimes Y \otimes Y$ and $Z \otimes Z \otimes Z$. Clearly, these three operators are Hermitian and square to the identity. They also anticommute with one another, and commute with all the operators in $\mathcal{E}$. Furthermore,

$$(Y \otimes Y \otimes Y)(X \otimes X \otimes X) = (YX) \otimes (YX) \otimes (YX) = i(Z \otimes Z \otimes Z),$$

so we can take $\bar{X} = Y \otimes Y \otimes Y$, $\bar{Y} = X \otimes X \otimes X$, and $\bar{Z} = Z \otimes Z \otimes Z$.

5.4 Good CSS codes

(a) The derivation of the Gilbert-Varshamov (GV) bound for CSS codes follows closely the argument discussed in class for the general GV bound, except that we want to include the specific property that CSS codes correct X and Z errors separately. This requires us to deal with X and Z operators separately.

Let n be the block size of the code, and take n_X to be the number of X-type stabilizer generators, and n_Z to be the number of Z-type generators, with $n_X + n_Z < n$ for a nontrivial code space. We will see how to choose n_X and n_Z later. For fixed n_X and n_Z, imagine a list (call it $\mathscr{S}$) of all stabilizers S with n_X X-type generators and n_Z Z-type generators[1]. Let $S_X \subseteq S$ be the set of operators generated by the X-type generators only, and $S_Z \subseteq S$ be the set generated by the Z-type generators only. For each stabilizer $S \in \mathscr{S}$, list all its dangerous errors, i.e. Pauli operators that are in $S^\perp \backslash S$. Note that a general Pauli operator P can be written (up to an overall phase) as $P_X P_Z$ where P_X is X-type and P_Z is Z-type, and since a CSS code corrects X and Z errors separately, P is dangerous for S iff either P_X is dangerous for S or P_Z is dangerous for S (or both).

Now, an X-type error (call it E_X) is dangerous iff $E_X \notin S_X$ and E_X commutes with S_Z. Similarly, a Z-type error (E_Z) is dangerous for $S \in \mathscr{S}$ iff $E_Z \notin S_Z$ and E_Z commutes with S_X. Therefore, for each $S \in \mathscr{S}$,

$$\begin{aligned}\left.\begin{array}{c}\text{number of dangerous}\\ X\text{-type errors}\end{array}\right|_S &= \begin{array}{c}\text{number of } X \text{ operators}\\ \text{that commute with } S_Z\end{array} - |S_X| \\ &= 2^{n-n_Z} - 2^{n_X}.\end{aligned}$$

The numbers in the last row are worked out as follows: the number of X operators is given by 2^n (each of the n positions in the code block can either be X or I), but this is subjected to n_Z independent constraints from commutation with S_Z, thus giving 2^{n-n_Z} as the number of X operators that commute with S_Z. $|S_X| = 2^{n_X}$ because every operator in S_X is given by $M_{X_1}^{a_1} M_{X_2}^{a_2} \dots M_{X_{n_X}}^{a_{n_X}}$, with $\{M_{X_i}\}$ as the n_X X-type generators of S_X, and $a_i \in \{0, 1\}$. Note that the number of dangerous X-type errors is independent of the particular choice of S. Similarly, the number of dangerous Z errors is given by

$$\left.\begin{array}{c}\text{number of dangerous}\\ Z\text{-type errors}\end{array}\right|_S = 2^{n-n_X} - 2^{n_Z}.$$

Now, using the Clifford group symmetry from class, restricted to X-type operators only, each nontrivial (i.e. $\neq I$) X-type operator E_X is dangerous for the same number (say N_X) of stabilizers $S \in \mathscr{S}$. Using the following identity, also from class, but adapted for X-type operators only:

$$\begin{aligned}|\mathscr{S}| \times \begin{pmatrix}\text{no. of dangerous } X\text{-type}\\ \text{errors for each } S \in \mathscr{S}\end{pmatrix} &= \begin{pmatrix}\text{no. of nontrivial}\\ X\text{-type errors}\end{pmatrix} \\ &\quad \times \begin{pmatrix}\text{no. of times each } X\text{-type}\\ \text{error appears in } \mathscr{S}\end{pmatrix} \\ \Rightarrow \qquad |\mathscr{S}| \left(2^{n-n_Z} - 2^{n_X}\right) &= (2^n - 1) N_X \\ \Rightarrow \qquad \frac{N_X}{|\mathscr{S}|} &= \frac{2^{n-n_Z} - 2^{n_X}}{2^n - 1}.\end{aligned}$$

[1]Note that this is *not* the same as listing all stabilizers with $n_X + n_Z$ generators. The generators must be either X-type or Z-type for it to be a CSS code.

A similar argument for the Z-type errors gives

$$\frac{N_Z}{|\mathscr{S}|} = \frac{2^{n-n_X} - 2^{n_Z}}{2^n - 1}$$

where N_Z is the number of $S \in \mathscr{S}$ each Z-type error is dangerous for.

Suppose we want to correct X-type errors from the set $\mathcal{E}^X$ and Z-type errors from the set $\mathcal{E}^Z$. We define the sets

$$\begin{aligned} &\mathcal{E}^{X(2)} := \{E_a^\dagger E_b : E_a, E_b \in \mathcal{E}^X\}, \\ \text{and} \qquad &\mathcal{E}^{Z(2)} := \{E_a^\dagger E_b : E_a, E_b \in \mathcal{E}^Z\}. \end{aligned}$$

For each nontrivial operator E in $\mathcal{E}^{X(2)}$ and $\mathcal{E}^{Z(2)}$, delete from $\mathscr{S}$ all stabilizers for which E is dangerous. After going through the two sets, we would have deleted at most $(|\mathcal{E}^{X(2)}| - 1)N_X + (|\mathcal{E}^{Z(2)}| - 1)N_Z$ stabilizers. There will be some codes left in the list if

$$\begin{aligned} |\mathscr{S}| &> (|\mathcal{E}^{X(2)}| - 1)N_X + (|\mathcal{E}^{Z(2)}| - 1)N_Z, \\ \text{i.e.} \qquad 1 &> (|\mathcal{E}^{X(2)}| - 1)\frac{N_X}{|\mathscr{S}|} + (|\mathcal{E}^{Z(2)}| - 1)\frac{N_Z}{|\mathscr{S}|} \\ &= (|\mathcal{E}^{X(2)}| - 1)\frac{2^{n-n_Z} - 2^{n_X}}{2^n - 1} + (|\mathcal{E}^{Z(2)}| - 1)\frac{2^{n-n_X} - 2^{n_Z}}{2^n - 1}. \end{aligned}$$

We finally get

$$(|\mathcal{E}^{X(2)}| - 1)(2^{n-n_Z} - 2^{n_X}) + (|\mathcal{E}^{Z(2)}| - 1)(2^{n-n_X} - 2^{n_X}) < 2^n - 1$$

as the GV bound for CSS codes.

(b) Suppose we want a CSS code that corrects t_X X-type errors and t_Z Z-type errors. Then, the error sets are

$$\begin{aligned} \mathcal{E}^X &= \{X\text{-type error } e^X : \mathrm{wt}(e^X) \le t_X\} \Rightarrow \mathcal{E}^{X(2)} = \{E^X : \mathrm{wt}(E^X) \le 2t_X\}, \\ \mathcal{E}^Z &= \{Z\text{-type error } e^Z : \ \mathrm{wt}(e^Z) \le t_Z\} \Rightarrow \mathcal{E}^{Z(2)} = \{E^Z : \mathrm{wt}(E^Z) \le 2t_Z\}. \end{aligned}$$

Therefore,

$$\begin{aligned} |\mathcal{E}^{X(2)}| - 1 &= \sum_{j=1}^{2t_X} \binom{n}{j} \overset{n\to\infty}{\approx} \binom{n}{2t_X} \overset{\text{Stirling}}{\approx} 2^{nH_2(2t_X/n)}, \\ |\mathcal{E}^{Z(2)}| - 1 &= \sum_{j=1}^{2t_Z} \binom{n}{j} \overset{n\to\infty}{\approx} \binom{n}{2t_Z} \overset{\text{Stirling}}{\approx} 2^{nH_2(2t_Z/n)}, \end{aligned}$$

where $H_2(x) = -x\log_2 x - (1-x)\log_2(1-x)$ is the binary entropy function, which is bounded between 0 and 1.

Putting the above into the GV bound from part (a), we get $(n \to \infty)$

$$2^{nH_2(2t_X/n)}(2^{n-n_Z} - 2^{n_X}) + 2^{nH_2(2t_Z/n)}(2^{n-n_X} - 2^{n_Z}) \lesssim 2^n.$$

Dividing both sides of the inequality by 2^n, and noting that $2^{n_X}/2^n \to 0$ and $2^{n_Z}/2^n \to 0$ for $n \to \infty$, since $n_X, n_Z < n$, we have

$$2^{nH_2(2t_X/n)-n_Z} + 2^{nH_2(2t_Z/n)-n_X} \lesssim 1.$$

Suppose we take, for some $\epsilon > 0$,

$$n_X = (1+\epsilon)nH_2\left(\frac{2t_Z}{n}\right), \qquad n_Z = (1+\epsilon)nH_2\left(\frac{2t_X}{n}\right),$$

where we assume $2t_X/n, 2t_Z/n$ small enough so that $H_2(.) < 1/2$, and ϵ should be taken small enough so that $n_X + n_Z < n$. In fact, we can take $\epsilon \to 0$ as $n \to \infty$. Then,

$$2^{-n\epsilon H_2(2t_X/n)} + 2^{-n\epsilon H_2(2t_Z/n)} \lesssim 1.$$

This is clearly satisfied for n large enough. Therefore,

$$\begin{aligned} & k = n - (n_X + n_Z) \\ \Rightarrow \quad & \frac{k}{n} = 1 - \frac{n_X}{n} - \frac{n_Z}{n} \quad \xrightarrow{n\to\infty,\epsilon\to 0} \quad 1 - H_2\left(\frac{2t_Z}{n}\right) - H_2\left(\frac{2t_X}{n}\right). \end{aligned}$$