CS 294-6, Quantum Computing (Umesh Vazirani)
Fall 2004, University of California, Berkeley
Lecture # (Quantum Error Correcting Codes)

Notes by Rohit Khandekar

1 Quantum error correction

Throughout the discussion of quantum computation so far, we assumed that the quantum system we were interested in was error-free. In reality, however, this is far from being true. The system continuously interacts with its environment. This interaction may corrupt the state of the system and the need for some error correcting mechanism becomes inevitable.

We first discuss two kinds of errors the environment may introduce in the system:

- bit flips
- bit measurements

Later we shall introduce a formal model of errors and discuss how to tackle them.

1.1 Bit flips

Suppose for simplicity that our system consists of a single qubit. We start with errors we are familiar with from the classical setting – bit-flips. Such an error converts the original state, say $\alpha|0\rangle + \beta|1\rangle$, into $\alpha|1\rangle + \beta|0\rangle$. We can correct these kind of errors using classical error correcting codes.

For example, we may use a repetition code, i.e., encoding $|0\rangle$ as $|000\rangle$ and $|1\rangle$ as $|111\rangle$. Thus, we encode $\alpha|0\rangle + \beta|1\rangle$ as $\alpha|000\rangle + \beta|111\rangle$. Now suppose the second bit gets flipped. Then the new state becomes $\alpha|010\rangle + \beta|101\rangle$. We assume that when any bit gets flipped, it is flipped in all superpositions. In order to locate the error, we attach an "error locating register" initially set to zero. This register is supposed to store the location of the bit in error. The overall state prior to any computation is thus

$$(\alpha|010\rangle + \beta|101\rangle) \otimes |0\rangle.$$

We then perform the required computations to determine the location of bits in error as we do in the classical setting, resulting in the state

$$\alpha|010\rangle \otimes |2\rangle + \beta|101\rangle \otimes |2\rangle.$$

Notice that the register which stores the error location is in tensor with the data qubits, i.e., the current state is

$$(\alpha|010\rangle + \beta|101\rangle) \otimes |2\rangle.$$

We can now recover the original qubits by flipping the bit at the location given by the error register to get

$$\alpha|000\rangle \otimes |2\rangle + \beta|111\rangle \otimes |2\rangle,$$

which is

$$(\alpha|000\rangle + \beta|111\rangle) \otimes |2\rangle.$$

From this state, the original state $\alpha|0\rangle + \beta|1\rangle$ can be recovered.

This scheme works even when the bit-flip errors occur in superposition. Suppose, for example, the first bit gets flipped with amplitude $1/\sqrt{2}$ and the second bit gets flipped with amplitude $1/\sqrt{2}$. In this case, the state after the error is

$$\frac{\alpha}{\sqrt{2}}|100\rangle + \frac{\beta}{\sqrt{2}}|011\rangle + \frac{\alpha}{\sqrt{2}}|010\rangle + \frac{\beta}{\sqrt{2}}|101\rangle.$$

After adding the error register and locating the bit in error we get

$$\begin{aligned} &\frac{\alpha}{\sqrt{2}}|100\rangle \otimes |1\rangle + \frac{\beta}{\sqrt{2}}|011\rangle \otimes |1\rangle + \frac{\alpha}{\sqrt{2}}|010\rangle \otimes |2\rangle + \frac{\beta}{\sqrt{2}}|101\rangle \otimes |2\rangle \\ &= \frac{1}{\sqrt{2}}(\alpha|100\rangle + \beta|011\rangle) \otimes |1\rangle + \frac{1}{\sqrt{2}}(\alpha|010\rangle + \beta|101\rangle) \otimes |2\rangle \end{aligned}$$

Now we measure the error register so that the system collapses to either $(\alpha|100\rangle+\beta|011\rangle)\otimes|1\rangle$ or $(\alpha|010\rangle + \beta|101\rangle) \otimes |2\rangle$. We then, as before, flip the bit in the location given by the error register to recover the original state.

1.2 Bit measurements

Another kind of error which is unique to the quantum setting is an unwanted measurement by the environment resulting in collapse of the current state. For example, if the current state is $\frac{1}{\sqrt{3}}(|001\rangle + |010\rangle + |101\rangle)$ and the second qubit gets measured, the state collapses to either $\frac{1}{\sqrt{2}}(|001\rangle + |101\rangle)$ or $|010\rangle$. During the measurement, some of the states in superposition get lost. As a consequence of the linearity of quantum mechanics, each basis state being unaware of the others, cannot tell that some of the others have disappeared. Therefore at first look, recovering from such errors seems very hard or even impossible.

One may be tempted to use the repetition code again to tackle these errors. For instance, we may try encoding the state $|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle$ as $|\Psi\rangle \otimes |\Psi\rangle \otimes |\Psi\rangle$. In such an encoding, if one of the three registers gets measured, we may hope to recover $|\Psi\rangle$ from the other two. However, such an encoding needs copying a qubit into three qubits and hence is impossible by No-cloning theorem.

We defer the discussion of how to take care of measurements or any general error.

1.3 Model of quantum errors

Now we introduce the model of general quantum errors as follows. Abstractly, our system is coupled with the environment. An error is modeled as a unitary transformation U that is applied to some qubits in our system and some other in the environment. Finally, some qubits get measured. The resulting state of our system depends on the result of the measurement. This overall transformation is linear. It is enough to consider only bit-flips, phase-flips, and their combinations since the space of linear operators is spanned by the operators representing the different possible bit-flips, phase flips, combinations of bit and phase-flips, and the identity operator. We illustrate the above fact for one bit errors. Suppose that the initial state is $\alpha|0\rangle + \beta|1\rangle$. The space of one bit linear operators is spanned by the following four operators.

$$\begin{aligned}
\text{no errors:} \quad & I\begin{pmatrix}\alpha\\ \beta\end{pmatrix} = \begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}\begin{pmatrix}\alpha\\ \beta\end{pmatrix} = \begin{pmatrix}\alpha\\ \beta\end{pmatrix}\\
\text{bit-flip:} \quad & X\begin{pmatrix}\alpha\\ \beta\end{pmatrix} = \begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix}\begin{pmatrix}\alpha\\ \beta\end{pmatrix} = \begin{pmatrix}\beta\\ \alpha\end{pmatrix}\\
\text{phase-flip:} \quad & Z\begin{pmatrix}\alpha\\ \beta\end{pmatrix} = \begin{pmatrix}1 & 0\\ 0 & -1\end{pmatrix}\begin{pmatrix}\alpha\\ \beta\end{pmatrix} = \begin{pmatrix}\alpha\\ -\beta\end{pmatrix}\\
\text{bit+phase-flip:} \quad & XZ\begin{pmatrix}\alpha\\ \beta\end{pmatrix} = \begin{pmatrix}0 & 1\\ -1 & 0\end{pmatrix}\begin{pmatrix}\alpha\\ \beta\end{pmatrix} = \begin{pmatrix}\beta\\ -\alpha\end{pmatrix}
\end{aligned}$$

This suggests that in order to correct general errors it is enough to be able to correct bit-flips and phase-flips. In Section 1.1, we saw how to correct bit-flip errors using repetition code. Now we see how to correct phase-flip errors by transforming them to bit-flips.

1.4 Phase flips

Recall that the Hadamard operator

$$H = \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}$$

transforms $|0\rangle$ into $|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ and $|1\rangle$ into $|-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$. It also transforms X into Z and vice versa, in the sense that $H^\dagger X H = Z$ and $H^\dagger Z H = X$. Thus in the Hadamard basis $\{|+\rangle, |-\rangle\}$, the phase-flip errors get transformed into "bit-flip" errors, i.e., $Z|+\rangle = |-\rangle$ and $Z|-\rangle = |+\rangle$.

With these observations, to correct the phase-flip errors, it is natural to encode $|0\rangle$ as $|+++\rangle$ and $|1\rangle$ as $|---\rangle$. We illustrate how to correct one qubit phase-flip error as follows. We encode the state $\alpha|0\rangle + \beta|1\rangle$ as $\alpha|+++\rangle + \beta|---\rangle$. A phase-flip errors, say in the second qubit, results in the state $\alpha|+-+\rangle + \beta|-+-\rangle$. To correct this, we first transform this into standard basis by applying tensored Hadamard operator $H^{\otimes 3}$:

$$\alpha|+-+\rangle + \beta|-+-\rangle \xmapsto{H^{\otimes 3}} \alpha|010\rangle + \beta|101\rangle.$$

Now using the bit error correction, we restore the state to $\alpha|000\rangle + \beta|111\rangle$. Again applying $H^{\otimes 3}$, we get $\alpha|+++\rangle + \beta|---\rangle$ we desired.

1.5 Shor's 9 qubit code

Combining the bit and phase-flip corrections, Shor designed 9 qubit code to correct both one-qubit bit-flip and phase-flip errors. In this code we encode one qubit into nine qubits as follows. To protect against phase-flip errors, we first encode the qubit $\alpha|0\rangle + \beta|1\rangle$ as

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

Then to protect against bit-flip errors, we encode each $|0\rangle$ as $|000\rangle$ and each $|1\rangle$ as $|111\rangle$ and get the following nine bit encoding:

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

The correction algorithm first corrects the bit-flip errors (if any) in each of the three groups of three qubits each. It then undoes the bit error encoding and recovers from the phase-flip errors if any.

Since the bit operations are completely impervious to the phases, the phase errors are left unchanged when we correct the bit-flip errors. The bit information is similarly left unchanged when we recover from the phase-flip errors.

2 Classical linear error correcting code over $\mathbb{F}_2$

We consider an error correcting code (ECC) $C \subseteq \mathbb{F}_2^n$ with $|C| = 2^k$. Such a code has two important matrices, the $k \times n$ generator matrix G and the $(n-k) \times n$ parity matrix P.

2.1 The generator matrix

The rows of the generator matrix G span the codewords of C so that a k-bit message $m \in \mathbb{F}_2^k$ can be encoded to an n-bit codeword $c \in \mathbb{F}_2^n$ via

$$c = mG.$$

In general, such a code is classified as an $[n, k, d]$ code where n and k are as above and d is the minimum Hamming distance between any two codewords in C. We note that since C is a linear code (i.e., $c_1, c_2 \in C \Rightarrow c_1 + c_2 \in C$), d may be equivalently defined as the minimum Hamming weight among non-zero codewords. A general $[n, k, d]$ code allows for the detection of up to $d - 1$ bit errors (these errors would necessarily produce an invalid codeword), and allows for the correction of up to $(d - 1)/2$ errors (by choosing the codeword of minimum Hamming distance from the received message).

2.2 The parity matrix

The parity matrix P has the property that its columns generate $C^\perp$ and thus

$$cP = 0 \iff c \in C.$$

For a general received message of the form $c' \equiv c + e$ where $c \in C$ and $e \in \mathbb{F}_2^n$ represents errors in the received message, the syndrome of c' is defined to be

$$c'P = (c + e)P = eP.$$

For $0 \leq |e| \leq d - 1$, the syndrome eP will always be non-zero and hence permit error detection. Moreover, if $|e| \leq (d - 1)/2$, the error e can be determined *uniquely* and hence corrected. Finally, we note that there exist classical $[n, k, d]$ ECCs with $n = O(k)$ and $d = \Theta(n)$.

3 CSS codes

As an example of quantum error correcting codes (QECC) we consider CSS codes, due to Calderbank, Shor, and Stein.

Recall that a general quantum error in an n-qubit state can be represented by a linear transformation. We first consider the case of a single qubit and examine the possible errors and corresponding linear transformations. Recall that the four matrices X, Z, XZ, I represent a basis for all such errors. These matrices correspond respectively to bit-flip errors, phase-flip errors, bit and phase errors, and no errors. For n qubits, the basis matrices are the set of all n-fold tensor products of the single-qubit matrices.

We consider quantum states in a group G (for concreteness, consider $G = \mathbb{Z}_2^n$). In particular, consider a state $|H\rangle$ which is a uniform superposition over some subgroup $H \leq G$. That is,

$$|H\rangle = \frac{1}{\sqrt{|H|}} \sum_{h \in H} |h\rangle.$$

Then a bit-flip error will map $|H\rangle$ to a uniform superposition over a coset of H, namely the state

$$|H + g\rangle = \frac{1}{\sqrt{|H|}} \sum_{h \in H} |h + g\rangle$$

where $g \in G$ indicates which bits have been flipped. For example, with $G = \mathbb{Z}_2^n$, the nonzero bits of g correspond to the flipped bits of $|H\rangle$.

Similarly a phase-flip error will map $|H\rangle$ to a superposition over H with phases determined by some character function $\chi_{g'} : G \to \mathbb{C}$. For the case $G = \mathbb{Z}_2^n$, the phase error induced by $\chi_{g'}$ on $|H\rangle$ yields the state

$$\chi_{g'}|H\rangle = \frac{1}{\sqrt{|H|}} \sum_{h \in H} (-1)^{g' \cdot h} |h\rangle.$$

Thus an initial state $|H\rangle$ will, after arbitrary errors, be a superposition of states of the form

$$\chi_{g'}|H + g\rangle = \frac{1}{\sqrt{|H|}} \sum_{h \in H} \chi_{g'}(h + g)|h + g\rangle.$$

We recall that under the Fourier Transform, bit-flip and phase-flip errors of the type defined above are interchanged (see figure), so that the Fourier Transform of the preceding state is given by

$$\mathrm{FT}_G : \chi_{g'}|H + g\rangle \mapsto \chi_g|H^\perp + g'\rangle.$$

Here $H^\perp \leq G^*$ is defined as $\{k \in G^* | \ k(h) = 1, \text{ for all } h \in H\}$ where G^* is the dual group of G.

3.1 CSS implementation

We begin with two classical ECCs, C_1 and C_2, described by $[n, k_1]$ and $[n, k_2]$ respectively. We additionally require that $C_1 \subseteq C_2$ and that $C_1^\perp$ and C_2 are each capable of correcting t errors (thus the minimum distance between codewords in each is at least $d = 2t + 1$). We

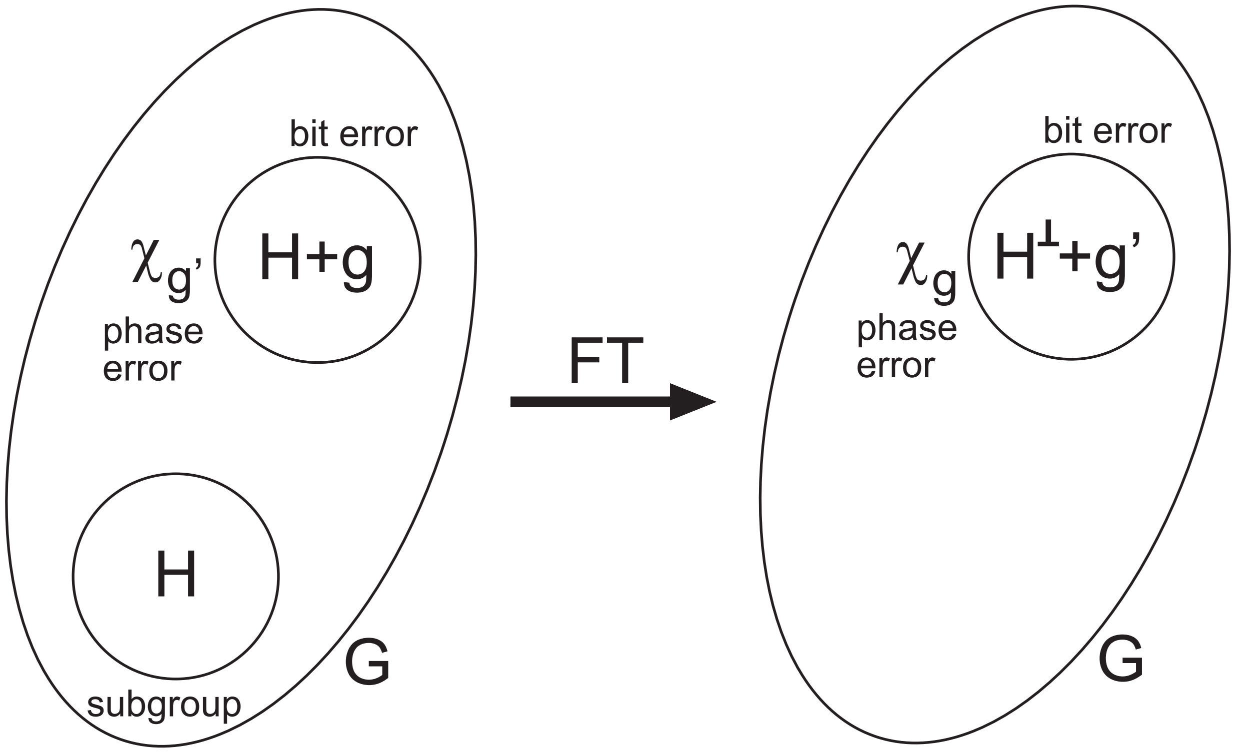

Figure 1: Quantum errors under the Fourier transform

now define an $[n, k_2 - k_1, d]$ code as follows. The codewords will be superpositions over cosets of C_1 in C_2. Thus there are $|C_2|/|C_1|$ codewords each of the form

$$|x + C_1\rangle \equiv \frac{1}{\sqrt{|C_1|}} \sum_{y \in C_1} |x + y\rangle$$

where x is a representative element in C_2.

For concreteness we now consider the case $G = \mathbb{Z}_2^n$. Then as described in the previous section, arbitrary bit-flip and phase-flip errors can be characterized by two group elements $e_b, e_p \in \mathbb{Z}_2^n$ respectively. The erroneous state is then

$$\frac{1}{\sqrt{|C_1|}} \sum_{y \in C_1} (-1)^{(x+y+e_b)\cdot e_p} |x + y + e_b\rangle \equiv \chi_{e_p} |C_1 + x + e_b\rangle.$$

The error correction algorithm proceeds in two stages. To correct the bit-flip errors, we first apply the parity matrix P_2 for the code C_2 and store the result in a set of ancillary qubits:

$$\chi_{e_p}|C_1 + x + e_b\rangle \otimes |0^n\rangle \xmapsto{P_2} \chi_{e_p}|C_1 + x + e_b\rangle \otimes |e_b P_2\rangle.$$

We now measure the second register and determine e_b as in the classical case. By applying the bit-flips indicated by e_b and discarding the ancillary qubits we arrive at the bit-flip corrected state

$$\chi_{e_p}|C_1 + x + e_b\rangle|e_b P_2\rangle \xmapsto{\text{correct}} \chi_{e_p}|C_1 + x\rangle.$$

To correct for phase errors, we apply a Fourier transform which, as described above, interchanges bit-flip and phase errors:

$$\chi_{e_p}|C_1 + x\rangle \xmapsto{\text{FT}} \chi_x|C_1^\perp + e_p\rangle.$$

From here we simply repeat the bit-error correction scheme above by applying the parity matrix $P_1^\perp$ for the code $C_1^\perp$ and correcting the measured error:

$$\chi_x|C_1^\perp + e_p\rangle|0^n\rangle \xmapsto{P_{1^\perp}} \chi_x|C_1^\perp + e_p\rangle|e_p P_{1^\perp}\rangle \xmapsto{\text{correct}} \chi_x|C_1^\perp\rangle.$$

By applying another Fourier transform we regain the original state:

$$\chi_x|C_1^\perp\rangle \xmapsto{\text{FT}} |C_1 + x\rangle.$$