Meeting 8.2: Quantum error correction

I. Overview

II. Discretization of errors

I. Overview

Should "fully programmable" quantum computers actually be built, it is generally expected that BQP will be correct abstraction of "quantum polynomial time."

But realistically, two practical issues to grapple with when engineering a quantum computer:

1. Storing quantum states in a stable way.
2. Implementing correct quantum gates.

What's the problem? **NOISE.**

1. Quantum states very delicate. ("Accidentally measuring" changes the state)
2. Unitary group $U(n)$ is not discrete ("Continuous errors" can compound.)

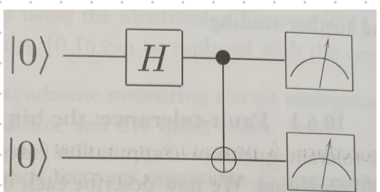
In theory, these issues should be solvable, by two techniques:

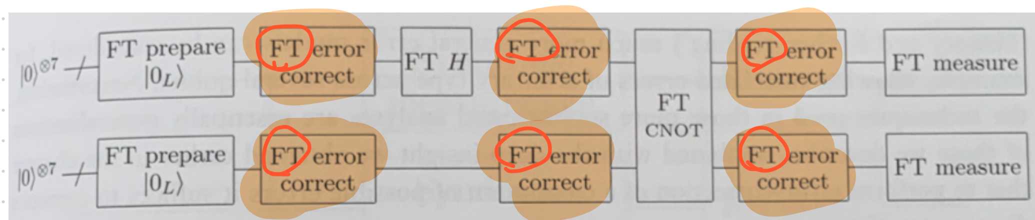
1. Quantum error correcting codes
2. Fault tolerant quantum computation.

We will focus on the first, but let me first address the second.

Basic idea of fault-tolerance:

in addition to using codes to store states, use encoded quantum gates.





"Concatenating" two codes (encoding one code inside another) costs polynomial overhead, but can lead to an exponential improvement in error rate. Iterating yields:

Threshold theorem for quantum computation: A quantum circuit containing $p(n)$ gates may be simulated with probability of error at most ϵ using

$$O(\text{poly}(\log p(n)/\epsilon)p(n)) \tag{10.116}$$

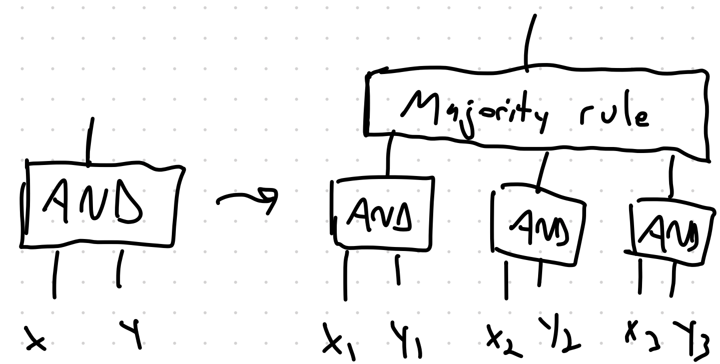
gates on hardware whose components fail with probability at most p, provided p is below some constant *threshold*, $p < p_{\text{th}}$, and given reasonable assumptions about the noise in the underlying hardware.

$p \overset{?}{\sim} 10^{-6}$

Note: Fault tolerant classical computing much easier to achieve. If there's a constant error rate $0 < \varepsilon < 1/3$ at every step of a classical Boolean circuit causing independent bit flip errors, then repetition codes, e.g.

$$0 \mapsto 000$$

$$1 \mapsto 111$$



allow us to make $\varepsilon > 0$ as small as we'd like.

Quantum analog of this code is not very good...

What's a quantum code?

An **n-qubit quantum error-correcting code of dimension d** is a Hilbert subspace

$$\mathcal{H} \leq \underbrace{\mathbb{C}^2 \otimes \mathbb{C}^2 \otimes \cdots \otimes \mathbb{C}^2}_{n \text{ "physical" qubits}} = (\mathbb{C}^2)^{\otimes n} \cong \mathbb{C}^{2^n}$$

(n called "length" of code)

If $d = 2^k$, then we say $\mathcal{H}$ encodes k logical qubits.

$\mathcal{H}$ is sometimes called the code space.

Not all subspaces are same! How they sit in $(\mathbb{C}^2)^{\otimes n}$ w.r.t. tensor decomposition matters.

Compare: Inside $(\mathbb{C}^2)^{\otimes n}$

$$\mathcal{H}_1 = \mathrm{span}\{|000\cdots 0\rangle, |111\cdots 1\rangle\} \quad (\text{"(quantum) repetition code"})$$

$$\mathcal{H}_2 = \mathrm{span}\{|000\cdots 0\rangle, |100\cdots 0\rangle\} \quad (\text{"trivial code"})$$

Both 2-dimensional, so they both encode a single qubit

$$\mathcal{H}_1 \cong \mathbb{C}^2 \cong \mathcal{H}_2$$

but $\mathcal{H}_1$ appears "spread out" more.

How to make precise?

Local bit flip error supported on single qubit can exchange $|00\cdots0\rangle$ and $|10\cdots0\rangle$.

Not true for $|00\cdots0\rangle$ and $|11\cdots1\rangle$.

More importantly, $\mathcal{H}_1$ is an entire subspace, not just the two basis states. Since quantum computers want to exploit superposition and entanglement, we want to detect and correct errors on arbitrary states in the codespace.

The repetition code will be able to detect up to $n-1$ bit flip errors and correct up to $\lfloor n/2 \rfloor$

Recall

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \qquad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

Given $|b_1 b_2 \cdots b_n\rangle \in (\mathbb{C}^2)^{\otimes n}$, define

$$X_i |b_1 b_2 \cdots b_n\rangle = Id_{(\mathbb{C}^2)^{\otimes i-1}} \otimes X \otimes Id_{(\mathbb{C}^2)^{\otimes n-i}} |b_1 b_2 \cdots b_n\rangle$$

$$= |b_1 \cdots b_i\rangle \otimes X |b_i\rangle \otimes |b_{i+1} \cdots b_n\rangle$$

So X_i is a bit flip error at i^{th} qubit

We define Z_i similarly, as a (relative) phase flip at the i^{th} qubit.

E.g. $n=5$. $\mathcal{H} = \text{span}\{|00000\rangle, |11111\rangle\}$

$$X_1 X_4 \left(\sqrt{\tfrac{2}{5}}|00000\rangle + \sqrt{\tfrac{3}{5}}|11111\rangle\right)$$

$$= \sqrt{\tfrac{2}{5}}|10010\rangle + \sqrt{\tfrac{3}{5}}|01101\rangle$$

"Majority rule" corrects this $X_1 X_4$ error CORRECTLY.

$$X_2 X_3 X_5 \left(\sqrt{\tfrac{2}{5}}|00000\rangle + \sqrt{\tfrac{3}{5}}|11111\rangle\right)$$

$$= \sqrt{\tfrac{2}{5}}|01101\rangle + \sqrt{\tfrac{3}{5}}|10010\rangle$$

We could measure to see that errors occurred, but majority rule will think $X_1 X_4$ error occured, and will recover incorrect state.

I've been uncareful. How do we see errors occured without spoiling the states?

Do measurement w/ operators

$$P_0 = |00000\rangle\langle 00000| + |11111\rangle\langle 11111| \quad \text{(no error presumed)}$$

$$P_1 = |10000\rangle\langle 10000| + |01111\rangle\langle 01111| \quad \text{(bit flip on 1 presumed)}$$

$$P_2 = |01000\rangle\langle 01000| + |10111\rangle\langle 10111| \quad \text{(bit flip on 2 presumed)}$$

$$\vdots$$

$$P_k = |11000\rangle\langle 11000| + |00111\rangle\langle 00111| \quad \text{(bit flips on 1+2 presumed)}$$

$$P_{k+1} = |10100\rangle\langle 10100| + |01011\rangle\langle 01011| \quad \text{(bit flips on 1+3 presumed)}$$

$$\vdots$$

$$P_N = |00011\rangle\langle 00011| + |11100\rangle\langle 11100| \quad \text{(bit flips on 4+5 presumed)}$$

$$I - P_1 - P_2 - \cdots - P_N$$

So repetition code good at detecting bit flip errors

However, $\mathcal{H}_1$ is still a bad QUANTUM code.
A single local Z error on $\mathcal{H}_1$ can swap orthogonal states:

$$Z_1 \left(\frac{|00\cdots0\rangle + |11\cdots1\rangle}{\sqrt{2}} \right) = \frac{|00\cdots0\rangle - |11\cdots1\rangle}{\sqrt{2}}$$

$\mathcal{H}_1$ can not detect any Z errors.

So we're left high and dry for now.
Two key issues:

1. Do good quantum error correcting codes exist?

2. What about errors that aren't X or Z?

II. Discretization of errors

Fortunately, if two errors are correctable, so is any linear combination of them.

Need to make some things precise first:

1. Error and noise.
2. Detectable error and code distance
3. Correctable error

If $\mathcal{H} \subseteq (\mathbb{C}^2)^{\otimes n}$, a noise (or error) space is any subspace $\mathcal{E} \subseteq \mathcal{B}((\mathbb{C}^2)^{\otimes n})$ ← all linear transformations $(= \mathrm{Mat}((\mathbb{C}^2)^{\otimes n}))$

An error is any $E \in \mathcal{E}$.

We say $\mathcal{H}$ detects E if exists $\lambda_E \in \mathbb{C}$ such that

$$\langle \varphi | E | \psi \rangle = \lambda_E \langle \varphi | \psi \rangle$$

for all $|\varphi\rangle, |\psi\rangle \in \mathcal{H}$.

If P is orthogonal projection onto $\mathcal{H}$, equivalent to saying

$$PE|\psi\rangle = \lambda_E |\psi\rangle$$

for all $|\psi\rangle \in \mathcal{H}$.

Intuition: when we measure w/ P, $I-P$ operators to decide whether $|\psi\rangle$ is still a codeword after E acts, if answer is "Yes" (happens w/ prob. $|\lambda_E|^2$), we still have $|\psi\rangle$.

The distance of $\mathcal{H}$ is smallest $d \in \mathbb{N}$ such that there exists an error supported on d qubits that $\mathcal{H}$ can not detect.

(Trivial and repetition codes both have distance 1.)

$\mathcal{H}$ corrects errors from $\mathcal{E}$ if for all $X, Y \in \mathcal{E}$ $\mathcal{H}$ detects $X^{+}Y$.

<u>Theorem</u> This is correct definition of "correcting errors from $\mathcal{E}$." I.e., it's equivalent to requiring there exist an "error correcting procedure."

Equivalently:

Theorem 10.1: **(Quantum error-correction conditions)** Let C be a quantum code, and let P be the projector onto C. Suppose $\mathcal{E}$ is a quantum operation with operation elements $\{E_i\}$. A necessary and sufficient condition for the existence of an error-correction operation $\mathcal{R}$ correcting $\mathcal{E}$ on C is that

$$PE_i^\dagger E_j P = \alpha_{ij} P, \tag{10.16}$$

for some Hermitian matrix α of complex numbers.

Take-aways

1. If $\mathcal{H}$ corrects/detects X and Y, then it corrects/detects $aX + bY$.
2. dist $\mathcal{H} > 2k$ if and only if $\mathcal{H}$ corrects all errors on k qubits.
3. Because products of X's and Z's and I's generate $\mathcal{B}((\mathbb{C}^2)^{\otimes k})$, suffices to correct them on all k-qubit subsets in order to correct ALL k-qubit errors.

<u>Theorem</u> $\mathcal{H}$ corrects all errors on k qubits if and only if it detects all errors that are products of at most $2k$ X_i's and Z_j's.