Quantum Algorithms

Quantum algorithms are designed to help speed up computational problems by exploiting quantum features.

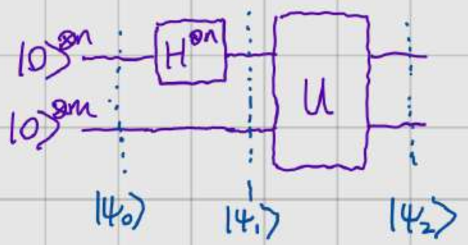
For example, we can use quantum parallelism.

Suppose we have a function $f : \{0,1\}^n \to \{0,1\}^m$. We can design an $n+m$ qubit gate U that acts like

$$U|\underline{x}\rangle|\underline{0}\rangle = |\underline{x}\rangle|f(\underline{x})\rangle$$

(for any input $|x\rangle|y\rangle$ output $|x\rangle|y \oplus f(x)\rangle$ then $U = \sum_{xy} |x\rangle\langle x| \otimes |y \oplus f(x)\rangle\langle y|$ is a valid unitary).

Then consider the circuit



$$|\psi_0\rangle = |0\rangle^{\otimes n} \otimes |0\rangle^{\otimes m}$$

$$|\psi_1\rangle = 2^{-n/2} \sum_{\underline{x} \in \{0,1\}^n} |\underline{x}\rangle \otimes |0\rangle^{\otimes m}$$

$$|\psi_2\rangle = 2^{-n/2} \sum_{\underline{x} \in \{0,1\}^n} |\underline{x}\rangle \otimes |f(\underline{x})\rangle$$

Equal superposition of all possible inputs and their corresponding outputs in one call to U.

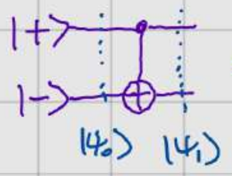
- This is not the same as just computing every possible input!

Why?

- We need to combine this with other quantum features to obtain something useful.

One other feature often used in conjunction with the above is phase Kickback

We've seen before the CNOT gate  the value of the control qubit can affect the target qubit. However, surprisingly the influence can also go the other way



$= \ ?$

$$|\psi_0\rangle = |+\rangle|-\rangle = \tfrac{1}{2}(|0\rangle+|1\rangle)(|0\rangle-|1\rangle) = \tfrac{1}{2}(|00\rangle - |01\rangle + |10\rangle - |11\rangle)$$

$$|\psi_1\rangle = \mathrm{CNOT}\cdot\tfrac{1}{2}(|00\rangle - |01\rangle + |10\rangle - |11\rangle) = \tfrac{1}{2}(|00\rangle - |01\rangle - |10\rangle + |11\rangle) = |-\rangle|-\rangle$$

So CNOT $|+\rangle|-\rangle = |-\rangle|-\rangle$, despite $|+\rangle$ being the control qubit.

We can combine phase-kickback and parallelism to create interesting interference which will enable us to 'select' the 'correct' answer to our computation.

Let's see our first example.

The Deutsch-Josza Algorithm

Suppose you are given a Boolean function $f: \{0,1\}^n \longrightarrow \{0,1\}$.
You are promised f is either

1) Constant $f(x) = 0 \quad \forall\ x \in \{0,1\}^n$
2) Balanced $f(x) = 0$ for half of the inputs

How do you decide if f is constant or balanced?

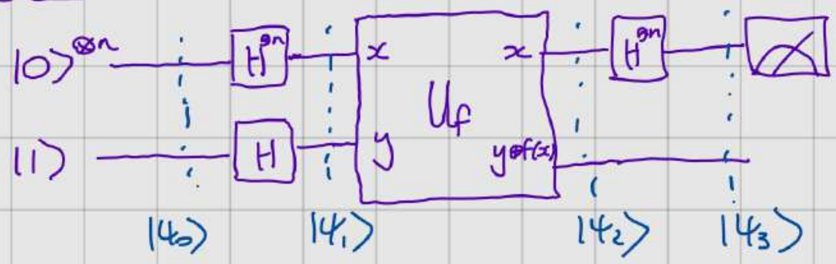
Classical analysis

- Best case - function is balanced and we observe in 2 queries.
- Worst case - We need $2^{n-1}+1$ queries to be certain.

Quantum Analysis

We will see that we only need 1 quantum query!

The circuit



Time steps

$$|\psi_0\rangle = |0\rangle^{\otimes n}|1\rangle$$

$$|\psi_1\rangle = 2^{-n/2} \sum_{x \in \{0,1\}^n} |x\rangle \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right)$$

$$|\psi_2\rangle = 2^{-n/2} \sum_{x \in \{0,1\}^n} |x\rangle \left(\frac{|f(x)\rangle - |f(x)\oplus 1\rangle}{\sqrt{2}} \right)$$

$f(x) = 0$ or 1

$$= 2^{-\frac{n}{2}} \sum_{x \in \{0,1\}^n} (-1)^{f(x)} |x\rangle \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right)$$

$$|\psi_3\rangle = 2^{-n/2} \sum_{x \in \{0,1\}^n} (-1)^{f(x)} \left(H^{\otimes n} |x\rangle \right) |-\rangle$$

$x \cdot y = x_0 y_0 \oplus x_1 y_1 \oplus \ldots$

$$= 2^{-n} \sum_{x \in \{0,1\}} (-1)^{f(x)} \left(\sum_{y \in \{0,1\}^n} (-1)^{x \cdot y} |y\rangle \right) |-\rangle$$

$$= 2^{-n} \sum_{y} \left(\sum_{x} (-1)^{f(x) + x \cdot y} \right) |y\rangle |-\rangle$$

Now measure the first n qubits.
Whats the probability we obtain the all 0 string?

$$\mathbb{P}(0^n) = 2^{-2n} \left| \sum_{x} (-1)^{f(x)} \right|^2$$ $\leftarrow$ 2^n terms in sum.

If f is constant then $\mathbb{P}(0^n) = 1$

If f is balanced then $\mathbb{P}(0^n) = 0$ $\swarrow$ equal # -ve and +ve contributions which cancel!!

The reasoning

If f is <u>constant</u> then U_f does nothing (apart from maybe a global phase). So we return to $|0\rangle^{\otimes n}$.

If f is <u>balanced</u> then U_f kicks back a phase onto exactly half of the $|x\rangle$ vectors in the superposition

$$\begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \\ 1 \end{pmatrix} \longmapsto \begin{pmatrix} 1 \\ 1 \\ -1 \\ -1 \\ 1 \\ \vdots \end{pmatrix}$$

This means $|\psi_1\rangle$ and $|\psi_2\rangle$ are <u>orthogonal</u>. So when we invert $H^{\otimes n}$ we arrive at some vector that is orthogonal to $|0^n\rangle$ and so we have $P(0^n) = 0$.

Quantum error correction

errorcorrectionzoo.org

We know from information theory that error correction is an important concept

* lossy communication
* lossy computation
* Damaged storage

data

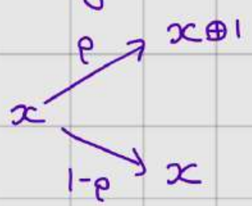
$\underline{x} \xrightarrow{\text{errors occur}} \underline{y}$

ECCs provide a way to recover x from y when certain errors occur!

Simplest example (Repetition code)

Single bit $x \in \{0,1\}$

Noisy channel



With probability p we lose our information about x.

Idea: add redundancy (encoding)

logical bit

codewords (physical bits)

$0 \longmapsto 000$

$1 \longmapsto 111$

What happens if we send the physical bits through the channel?

If $x=0$

Output	Prob
000	$(1-p)^3$
001	$p(1-p)^2$
010	$p(1-p)^2$
100	$p(1-p)^2$
...	$O(p^2)$

Output	Prob
111	$(1-p)^3$
110	$p(1-p)^2$
101	⋮
011	
...	negl

Decoding:

$$\left.\begin{matrix}000\\001\\010\\100\end{matrix}\right\} \mapsto 0 \qquad \left.\begin{matrix}111\\110\\101\\011\end{matrix}\right\} \mapsto 1$$

Probability we make a mistake is $p^3 + 3p^2(1-p)$

$$= 3p^2 - 2p^3 < p$$

whenever $\underline{p < \frac{1}{2}}$ we get an advantage.

By adding redundancy to our message we could protect it from errors.

A first attempt at QEC

In quantum systems we also need error correction (they're very noisy!)

A naive attempt would be to replicate the rep-code

$$|\psi\rangle \mapsto |\psi\rangle|\psi\rangle|\psi\rangle$$

But there are issues here (What can you think of?)

1) No cloning: if $|\psi\rangle$ is unknown then there's no way we can reliably copy it.
2) Detecting errors requires observing the string classically. Measurements disturb states (Problem?)
3) There are a lot more possible errors for quantum (a continuum) e.g. $R_\theta = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{pmatrix}$ can occur.

Our first scheme (Dealing with bitflip errors) Recall $X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ bitflip in $\{|0\rangle, |1\rangle\}$ basis.

Suppose for the moment we only care about X errors. So we have a qubit channel

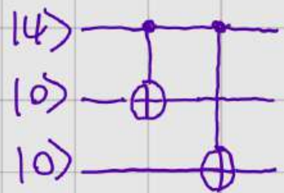
$$|\psi\rangle \begin{matrix} \xrightarrow{p} X|\psi\rangle \\ \xrightarrow{1-p} |\psi\rangle \end{matrix}$$

We make the following encoding

$$|0\rangle \longmapsto |000\rangle$$
$$|1\rangle \longmapsto |111\rangle$$

Why does this not violate no-cloning?

This can be done with a circuit



If $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ then circuit outputs

$$\alpha|000\rangle + \beta|111\rangle$$

entangled state if $\alpha \neq 0 \neq \beta$

Now we pass each qubit through the noisy channel:

Assuming errors act independently:

Prob	State
$(1-p)^3$	$\alpha|000\rangle + \beta|111\rangle$
$p(1-p)^2$	$\alpha|100\rangle + \beta|011\rangle$
$p(1-p)^2$	$\alpha|010\rangle + \beta|101\rangle$
$p(1-p)^2$	$\alpha|001\rangle + \beta|110\rangle$
negl	———

What can we do now?

States all live in orthogonal subspaces and so can be reliably distinguished!

$$P_0 = |000\rangle\langle 000| + |111\rangle\langle 111| \rightarrow \text{No error}$$
$$P_1 = |100\rangle\langle 100| + |011\rangle\langle 011| \rightarrow \text{Qubit 1 error}$$
$$P_2 = |010\rangle\langle 010| + |101\rangle\langle 101| \rightarrow \text{Qubit 2 error}$$
$$P_3 = |001\rangle\langle 001| + |110\rangle\langle 110| \rightarrow \text{Qubit 3 error}$$

1) These measurements will perfectly distinguish the different errors

E.g. $\langle\psi|P_i|\psi\rangle = \delta_{i0}$ $\quad\langle\psi|(X\otimes 1\otimes 1)P_i(X\otimes 1\otimes 1)|\psi\rangle = \delta_{i1}$

2) The measurement does not disturb the underlying state (Why?)

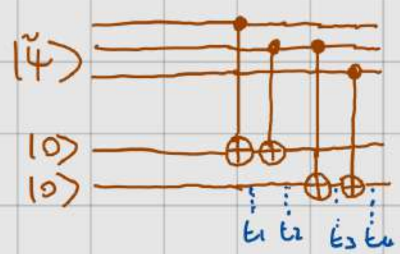
$\Rightarrow$ The measurement detects which (if any) X error occured and when we get outcome i the state after measurement is

$$X_i|\psi\rangle$$

where
$$X_0 = \mathbb{1}\otimes\mathbb{1}\otimes\mathbb{1}$$
$$X_1 = X\otimes\mathbb{1}\otimes\mathbb{1}$$
$$X_2 = \mathbb{1}\otimes X\otimes\mathbb{1}$$
$$X_3 = \mathbb{1}\otimes\mathbb{1}\otimes X$$

So we can correct the error we detected!

A circuit viewpoint on error detection



Recall CNOT $|0\rangle\langle 0|\otimes\mathbb{1} + |1\rangle\langle 1|\otimes X$

$|\tilde{\psi}\rangle \in \{|\psi\rangle, X_1|\psi\rangle, X_2|\psi\rangle, X_3|\psi\rangle\}$

$|\tilde{\psi}\rangle = \alpha|abc\rangle + \beta|\bar{a}\bar{b}\bar{c}\rangle$

At t_1: $\quad\alpha|abc\rangle|a\rangle|0\rangle + \beta|\bar{a}\bar{b}\bar{c}\rangle|\bar{a}\rangle|0\rangle$

t_2: $\quad\alpha|abc\rangle|a\oplus b\rangle|0\rangle + \beta|\bar{a}\bar{b}\bar{c}\rangle|\bar{a}\oplus\bar{b}\rangle|0\rangle$

$\vdots$

t_4 $\quad\alpha|abc\rangle|a\oplus b\rangle|b\oplus c\rangle + \beta|\bar{a}\bar{b}\bar{c}\rangle|\bar{a}\oplus\bar{b}\rangle|\bar{b}\oplus\bar{c}\rangle$

$\quad = (\alpha|abc\rangle + \beta|\bar{a}\bar{b}\bar{c}\rangle)|a\oplus b\rangle|b\oplus c\rangle$

The two extra states encode the parity of pairs of qubits.

1) Measure Qubit 4 in computational basis $\begin{cases} \xrightarrow{0} \text{No error on 1st 2 qubits} \\ \xrightarrow{1} \text{Qubit 1 or Qubit 2 has error} \end{cases}$

2) Measure Qubit 5 in computational basis $\begin{cases} \xrightarrow{0} \text{No error on 2nd 2 qubits} \\ \xrightarrow{1} \text{Qubit 2 or qubit 3 has error} \end{cases}$

Outcome	Error
(0,0)	$\mathbb{1}$
(0,1)	X_3
(1,0)	X_1
(1,1)	X_2

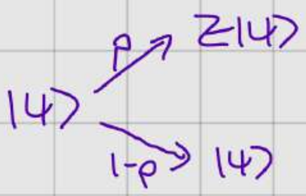
↑ Error Syndrome

2 bits of information sufficient to detect and correct the errors.

This gives hope that QEC is possible but this can only detect X errors, what if a Z error occurs?

↑ For this code we will always get syndrome (0,0) and not detect an error.

The phase flip code



← New noisy channel but with a phase flip instead of a bitflip.

Ideas?

Trick is to change viewpoint:

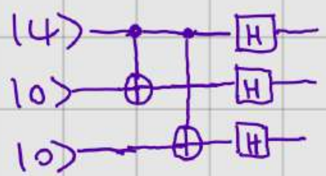
Move to Hadamard basis $|+\rangle/|-\rangle$

Z - phase flip in $|0\rangle/|1\rangle$
X - bit flip in $|0\rangle/|1\rangle$

Z - bitflip in $|+\rangle/|-\rangle$
X - phaseflip in $|+\rangle/|-\rangle$

Then use encoding $|0\rangle \longmapsto |{+}{+}{+}\rangle$
$|1\rangle \longmapsto |{-}{-}{-}\rangle$

Encoding map?

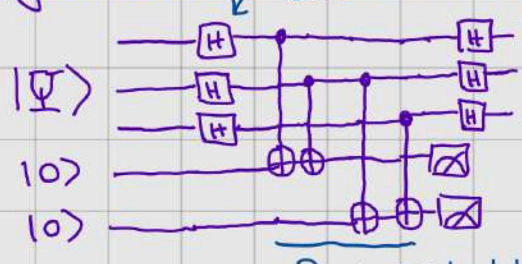


$\alpha|0\rangle + \beta|1\rangle \longmapsto \alpha|{+}{+}{+}\rangle + \beta|{-}{-}{-}\rangle$

$HZH = X$ So exactly like X error occured on X encoding

How to get syndrome?

Reverts corrupted state back to computational basis



converts back to $|+\rangle/|-\rangle$ basis

Does error check like in X code

The 9 qubit code (Shor)

Concatenating . phase and bitflip code

Encoding: Step 1) $|0\rangle \longmapsto |{+}{+}{+}\rangle \quad |1\rangle \longmapsto |{-}{-}{-}\rangle$

Step 2) Each qubit $|0\rangle \longmapsto |000\rangle \quad |1\rangle \longmapsto |111\rangle$

$|+\rangle \longmapsto \frac{|000\rangle + |111\rangle}{\sqrt{2}}$

Overall

$$|0\rangle \longmapsto \frac{1}{2\sqrt{2}}\left(|000\rangle + |111\rangle\right)^{\otimes 3}$$

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

$$\alpha|0\rangle + \beta|1\rangle \longmapsto \frac{\alpha\left(|000\rangle+|111\rangle\right)^{\otimes 3} + \beta\left(|000\rangle - |111\rangle\right)^{\otimes 3}}{2\sqrt{2}}$$

Question: What circuit implements the 9 qubit encoding?

Syndrome detection

X errors

Same as above: check parities of

$\underbrace{(1,2),(2,3)\ (4,5)(5,6)\ (7,8)(8,9)}$

6 bits of information (6 qubits) needed to detect bitflips on the state.

Can we detect multiple errors here?

Z errors

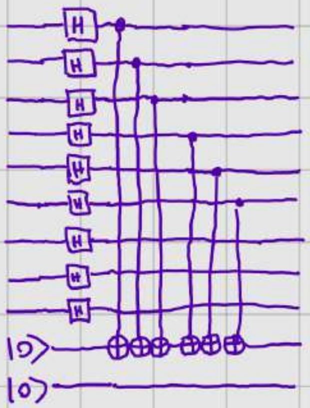
Known as a degenerate code

Different correctable errors lead to same corrupted state.

Suppose we have a Z error on qubit $^{1,2\text{ or }3}$ then the state becomes

$$\alpha\left(|000\rangle - |111\rangle\right)\left(|000\rangle + |111\rangle\right)^{\otimes 2} + \beta\left(|000\rangle + |111\rangle\right)\left(|000\rangle - |111\rangle\right)^{\otimes 2}$$

Need to compare the phases between the different partitions



$|\psi_0\rangle = \frac{1}{\sqrt{2}}\left(|000\rangle + |111\rangle\right) \quad |\psi_1\rangle = \frac{1}{\sqrt{2}}\left(|000\rangle - |111\rangle\right)$

$H^{\otimes 3}|\psi_0\rangle = |000\rangle + |011\rangle + |101\rangle + |110\rangle$ ← even parity

$H^{\otimes 3}|\psi_1\rangle = |001\rangle + |010\rangle + |100\rangle + |111\rangle$ ← odd parity

Then CNOT triple gives $+1$ if $|\psi_1\rangle$
0 if $|\psi_0\rangle$

The 6 CNOTs check that the parity of input state are the same ie. 0 if $|\psi_0\rangle|\psi_0\rangle$ or $|\psi_1\rangle|\psi_1\rangle$
1 if $|\psi_0\rangle|\psi_1\rangle$ or $|\psi_1\rangle|\psi_0\rangle$

Therefore can check phase difference between two triples!
Can correct it!

Remainder of circuit is done by checking triples 2 & 3 so can determine if a phase error occured. Can you detect multiple phase errors?
And then applying Hadamard gates recovers the original state which can be corrected depending on the syndrome observed!

<u>Remark</u>: The two detection steps are completely independent, neither affects the encoded state. Therefore we can detect and correct both an X and a Z error even if they occur on the same qubit!

We now know how to correct X and Z errors but there are an awful lot more errors to consider!

What about arbitrary errors?

Let's just give it a go...

<u>Ex</u>:
$$U_\theta = \cos(\theta/2)\mathbb{1} - i\sin(\theta/2)X$$

Suppose this error occurs on the 1st qubit in our bit flip code $|\Psi\rangle$ 3 qubit state

Then after error we have
$$|\Psi_E\rangle = \cos(\theta/2)|\Psi\rangle - i\sin(\tfrac{\theta}{2})X_1|\Psi\rangle$$

Let's put this through the syndrome detection circuit

2 qubit states
$$|\Psi_E\rangle \longmapsto \cos(\tfrac{\theta}{2})|\Psi\rangle|\text{no } X\text{ error}\rangle - i\sin(\tfrac{\theta}{2})X_1|\Psi\rangle|X_1\text{ error}\rangle$$

physical qubits are now entangled with the X error detection qubits

What happens when we measure the X error register?

Prob	Outcome	Post Measurement state
$\cos^2(\theta/2)$	No error	$\lvert\Psi\rangle \lvert \text{no X error}\rangle$
$\sin^2(\theta/2)$	X_1 error	$X_1 \lvert\Psi\rangle \lvert X_1 \text{ error}\rangle$

Magic! By measuring the syndrome we force the state to choose whether the X error occurs or not :)

Consistent with θ small being a small rotation error / coincides with small probability of X error occuring

<u>How does this help with arbitrary errors?</u>

Any error E can be expressed in Pauli basis

$Y = iXZ$

$$E = e_0 \mathbb{1} + e_1 X + e_2 Z + e_3 XZ$$

Extending the above example we see it can detect and correct all such errors! Shor code can correct all single qubit errors!

<u>Remark</u> Any QECC that can correct errors E and F can correct any linear combination $aE + bF$!