Note: 2nd qubit need not be measured (and contains no information!)

Two core ideas:

- Use input $\sum |x\rangle$ to evaluate f on all inputs simultaneously
- Need way to read out relevant info!

b) The Deutsch-Jozsa algorithm

Consider $f: \{0,1\}^n \to \{0,1\}$ w/ promise that

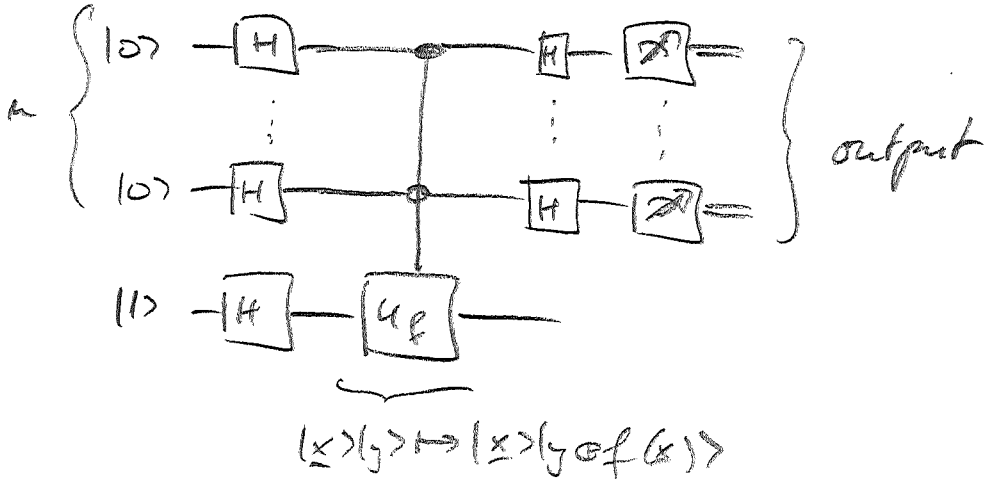
either $f(x) = c \ \forall x$ "constant"

or $\#\{x \mid f(x)=0\} = \#\{x \mid f(x)=1\}$ "balanced"

Want to know: Is f constant or balanced?

→ How many queries to f do we need?

Use same idea: Input $\sum |x\rangle$ and $\frac{|0\rangle - |1\rangle}{\sqrt{2}}$:



What is action of $H^{\otimes n}$?

$$H: |x\rangle \longmapsto \frac{1}{\sqrt{2}} \sum (-1)^{x \cdot y} |y\rangle$$

$$H^{\otimes n}: |x_1, \dots, x_n\rangle \longmapsto \frac{1}{\sqrt{2^n}} \sum (-1)^{x_1 y_1} (-1)^{x_2 y_2} \cdots |y_1, \dots, y_n\rangle$$

$$|\underline{x}\rangle \longmapsto \frac{1}{\sqrt{2^n}} \sum (-1)^{\underline{x} \cdot \underline{y}} |\underline{y}\rangle$$

$$(\underline{x} \cdot \underline{y} := x_1 y_1 \oplus x_2 y_2 \oplus \dots \oplus x_n y_n)$$

$$\Rightarrow |\underline{0}\rangle|1\rangle \xmapsto{H^{\otimes n} \otimes H} \text{(omit prefactors!)} \left(\sum |\underline{x}\rangle\right)(|0\rangle - |1\rangle)$$

$$\xmapsto{U_f} \left(\sum_{\underline{x}} (-1)^{f(\underline{x})} |\underline{x}\rangle\right)(|0\rangle - |1\rangle)$$

$$\xmapsto{H^{\otimes n} \otimes I} \left(\sum_{\underline{y}} \underbrace{\sum_{\underline{x}} (-1)^{f(\underline{x}) + \underline{x} \cdot \underline{y}}}_{\circledast} |\underline{y}\rangle\right)(|0\rangle - |1\rangle)$$

f constant: $\circledast = (-1)^{f(\underline{x})} \cdot \underbrace{\sum_{\underline{x}} (-1)^{\underline{x} \cdot \underline{y}}}_{\delta_{\underline{y},\underline{0}}} = (-1)^{f(\underline{x})} \delta_{\underline{y},\underline{0}}$ (const.)

f balanced: For $\underline{y} = \underline{0}$, $\circledast = \sum_{\underline{x}} (-1)^{f(\underline{x}) + \underline{x} \cdot \underline{0}} = \sum_{\underline{x}} (-1)^{f(\underline{x})} = \underline{0}$

$\Rightarrow$ output is $y = 0 \Longrightarrow f$ constant

— " — $y \neq 0 \Longrightarrow f$ balanced.

$\Rightarrow$ Deterministic discrimination w/ one eval. of f!

What is speed-up?

- Quantum: 1 use of f.
- Class: Worst case, we need to test $2^{n/2} - 1$ values of f to be sure $\Rightarrow$ const. vs. exp.!
- Bad: If we are happy w/ correct answer w/ high prob. ($\leftrightarrow$ q. comp. will also have errors), e.g. $p = 1 - \varepsilon$, then for k tests

$$P_{\text{error}} \approx \underbrace{2 \cdot \left(\frac{1}{2}\right)^k}_{\substack{\text{prob. of } k \text{ eq. outcomes} \\ \text{if } f \text{ is balanced}}} \stackrel{!}{=} \varepsilon$$

$\Rightarrow k \sim \log 1/\varepsilon.$

$\Rightarrow$ Much smaller speed-up vs. probabilistic class. algorithm!

c) Simon's algorithm

$$f: \{0,1\}^n \to \{0,1\}^n$$

Promise: $\exists a$ s.th. $f(x) = f(y)$ iff $x \oplus a = y$.

("hidden periodicity")

Problem: Find a.

Classical: Need to query $f(x_i)$ until $f(x_i) = f(x_j)$ is found.

For k queries: $\sim k^2$ pairs; $P(f(x_i) = f(x_j)) = 2^{-n}$.

$\Rightarrow$ $P_{\text{succes}} \leq k^2 2^{-n}$

$\Rightarrow$ need $k \sim \exp(cn)$ queries!

Quantum:

Start with $\frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x\rangle = H^{\otimes n} |0 \dots 0\rangle$

$$U_f : \left(\frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x\rangle_A\right) |0\rangle_B \longmapsto \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x\rangle_A |f(x)\rangle_B .$$

Now meas. B $\Rightarrow$ collapse onto <u>random</u> $f(x_0)$.

A is collapsed to

$$\frac{1}{N} \sum_{x : f(x) = f(x_0)} |x\rangle = \frac{1}{\sqrt{2}} \left(|x_0\rangle + |x_0 \oplus a\rangle\right)$$

How can we extract a? (Meas. in comp. basis gives x_0 or $x_0 \oplus a$!)

Apply $H^{\otimes n}$ again:

$$H^{\otimes n}\left(\frac{1}{\sqrt{2}}(|x_0\rangle + |x_0 \oplus a\rangle)\right) = \frac{1}{\sqrt{2^{n+1}}} \sum_y \underbrace{\left[(-1)^{x_0 \cdot y} + (-1)^{(x_0+a)\cdot y}\right]}_{= 2\cdot(-1)^{x_0\cdot y} \text{ if } a\cdot y = 0,\ = 0 \text{ if } a\cdot y = 1} |y\rangle$$

$$= \frac{1}{\sqrt{2^{n-1}}} \sum_{y : a\cdot y = 0} (-1)^{x_0 \cdot y} |y\rangle$$

Measure $|y\rangle \Rightarrow$ find random vector s.th. $ay = 0$.

$(n-1)$ lin. indep. y allow to compute a.

Need $O(n)$ random y to have $(n-1)$ lin. indep. ones.

$\Rightarrow$ a is found in $O(n)$ steps!

$\Rightarrow$ Exponential speed-up!

Notes: • Need not measure B register! (Outcome indep. of B meas.!)

• $H^{\otimes n}$ can be seen as Fourier transform over $(\mathbb{Z}_2)^{\otimes n}$

$\Rightarrow$ period finding via Fourier trafo (cf. Lect. 1)

3. Grover's algorithm

Common hard computational problem:

We know how to check solution efficiently, but we want to find a solution:

Many problems: Graph coloring, factoring, 3-SAT, Hamiltonian path, ...

Class of "NP problems".

Re-formulation:

We know how to compute $f(x) \in \{0,1\}$ ("verifier" for solution x, $1 \equiv$ "good solution"), and want to find x_0 s.th. $f(x_0) = 1$.

(Can be seen as "database search": Want to find "marked element" x_0 in an unstructured database.)

Assume for now that $x_0 : f(x_0) = 1$ is unique.
(Generalization: Later / homework)

Classically: Will need $O(N)$ queries to f for unstructured search (i.e. don't use any properties of f).

Will see: Quantumly, $O(\sqrt{N})$ queries enough.

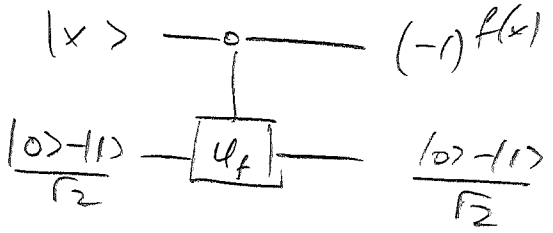
(Note: Quadr. speedup – worse than Simon – but very relevant problem!)

Consider $f: \{0,\dots,N-1\} \to \{0,1\}$

Ingredient 1:

Oracle $O_f : |x\rangle \longmapsto (-1)^{f(x)}|x\rangle = (-1)^{\delta_{x,x_0}}|x\rangle$

Build via



(Note: $x = 0,\dots,N-1$; $|x\rangle \in \mathbb{C}^N$)

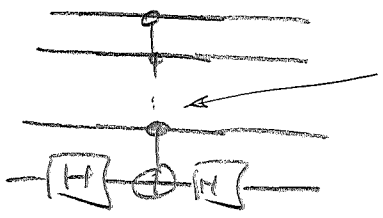
Note that $O_f = I - 2\cdot|x_0\rangle\langle x_0|$:

Flips amplitude of "marked" element.

Ingredient 2:

Unitary $O_0 : |x\rangle \longmapsto (-1)^{\delta_{x,0}}|x\rangle$

Corresponds to $C^{n-1}Z =$



We have $O_0 = I - 2|0\rangle\langle 0|$

$$\Rightarrow O_\omega := H^{\otimes n} O_0 H^{\otimes n} = I - 2|\omega\rangle\langle\omega|; \quad |\omega\rangle = \frac{1}{\sqrt{N}}\sum_{x=0}^{N-1}|x\rangle.$$

Algorithm:

Start from $|\psi_0\rangle = |\omega\rangle$ and apply

Grover iteration $G = -H^{\otimes n} O_0 H^{\otimes n} O_f = (-O_\omega) \cdot O_f$:

$$|\psi_k\rangle \longmapsto |\psi_{k+1}\rangle = G|\psi_k\rangle = -O_\omega \cdot O_f |\psi_k\rangle.$$

Note: Only 2 "special" vectors in O_f & O_ω: $|x_0\rangle$ and $|\omega\rangle$

$\Rightarrow$ can analyze everything in 2-dim space spanned by $|x_0\rangle$ & $|\omega\rangle$!

Define vectors

$$|\alpha\rangle := |x_0\rangle$$

$$|\beta\rangle := \frac{1}{\sqrt{N-1}} \sum_{x \neq x_0} |x\rangle \qquad |\alpha\rangle \perp |\beta\rangle.$$

Since $|\omega\rangle = \frac{1}{\sqrt{N}}|\alpha\rangle + \sqrt{\frac{N-1}{N}}|\beta\rangle$, we can always rewrite

$$a|\alpha\rangle + b|\beta\rangle = x|\omega\rangle + y|\omega^\perp\rangle$$

with $|\omega^\perp\rangle \perp |\omega\rangle$.

What is effect of O_f & $(-O_\omega)$?

$$O_f\left(a|\alpha\rangle + b|\beta\rangle\right) \underset{A}{=} -a|\alpha\rangle + b|\beta\rangle$$

$$O_f = I - 2|\alpha\rangle\langle\alpha|$$

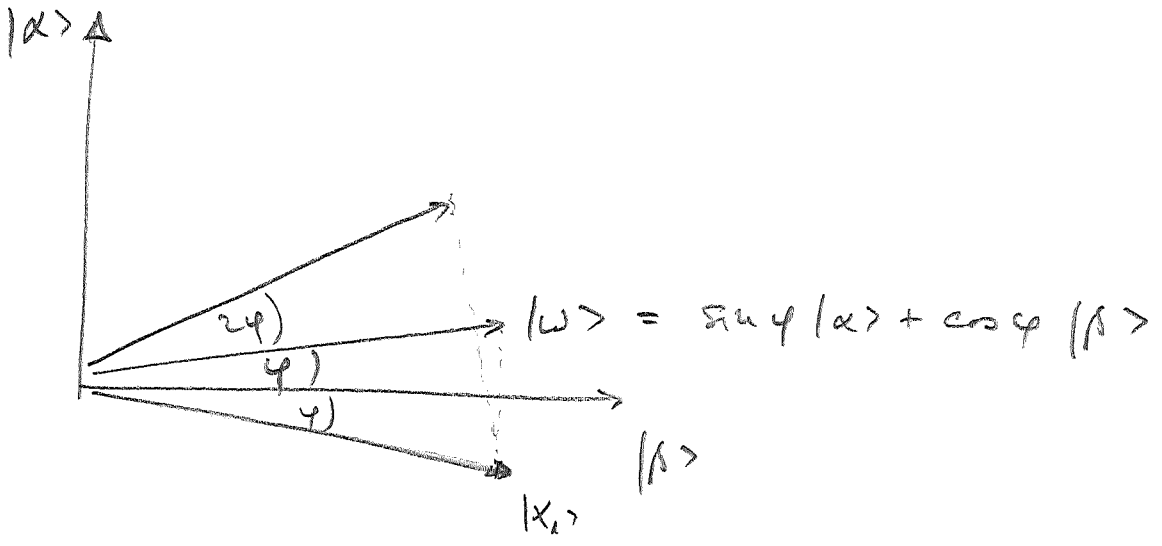
$\Rightarrow$ Reflection about $|\beta\rangle$!

$$(-O_\omega)\left(x|\omega\rangle + y|\omega^\perp\rangle\right) = x|\omega\rangle - y|\omega^\perp\rangle$$

$\Rightarrow$ Reflection about $|\omega\rangle$!

So... what happens in a Grover iteration, starting with $|\psi_0\rangle = |\omega\rangle$?

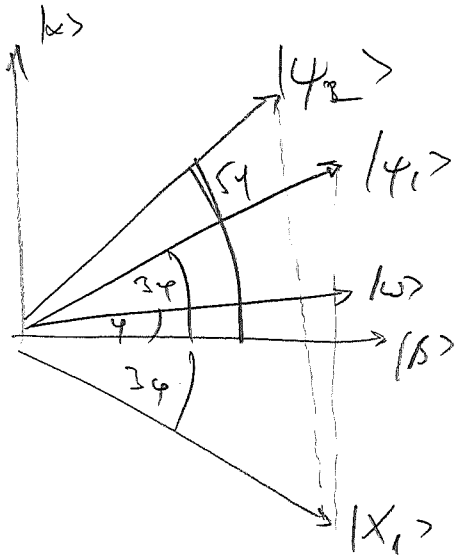
$$|\psi_1\rangle = -O_\omega \underbrace{O_f|\omega\rangle}_{|x_1\rangle} \quad ; \quad |\omega\rangle = \sin\varphi\,|\alpha\rangle + \cos\varphi\,|\beta\rangle.$$



$$\Rightarrow \quad |\psi_1\rangle = \sin(3\varphi)\,|\alpha\rangle + \cos(3\varphi)\,|\beta\rangle.$$

Next iteration:

$$|\psi_2\rangle = -O_\omega \cdot \frac{O_f |\psi_1\rangle}{|\chi_2\rangle}$$



$$\Rightarrow |\psi_2\rangle = \sin(5\varphi)|\alpha\rangle + \cos(5\varphi)|\beta\rangle.$$

$$\Rightarrow |\psi_k\rangle = \sin((2k+1)\varphi)|\alpha\rangle + \cos((2k+1)\varphi)|\beta\rangle$$

Want that $\varphi_k = (2k+1)\varphi \approx \frac{\pi}{2}$

$\Rightarrow$ measurement will with high prob. yield $|\alpha\rangle = |\omega\rangle$!

We have: $$\frac{\sin\varphi}{\cos\varphi} = \frac{\frac{1}{\sqrt{N}}}{\sqrt{\frac{N-1}{N}}} = \frac{1}{\sqrt{N-1}}$$

$$\Rightarrow \varphi \approx \frac{1}{\sqrt{N}} \text{ for large } N!$$

$$\Rightarrow k \approx \frac{\pi}{4}\sqrt{N}$$

$\Rightarrow O(\sqrt{N})$ steps sufficient!

$\Rightarrow$ Quadratic speed-up for general search problems!

Note: • K solutions. Same method works with $O\left(\sqrt{\frac{N}{K}}\right)$ steps.
($\rightarrow$ HW).

• Can also be adapted to the case when K is unknown.