# 10/26 Quantum Algorithms Part 1

In 1985, Deutsch found a toy problem that quantum computers could solve more efficiently than classical computers, in terms of queries (not time!)
↳ extended by Josza to a "real" problem

Problem:

Given classical function f w/ n input bits and 1 output bit
↳ usable via black box aka <u>oracle</u>

f is either constant (same output for all inputs) or balanced (0 for half of inputs, 1 for other half).

Determine if a given f is constant or balanced.

Classically, you just have to check tons of inputs until you've tested more than half of the possible queries worst case (if you keep getting the same output) to show it's constant.

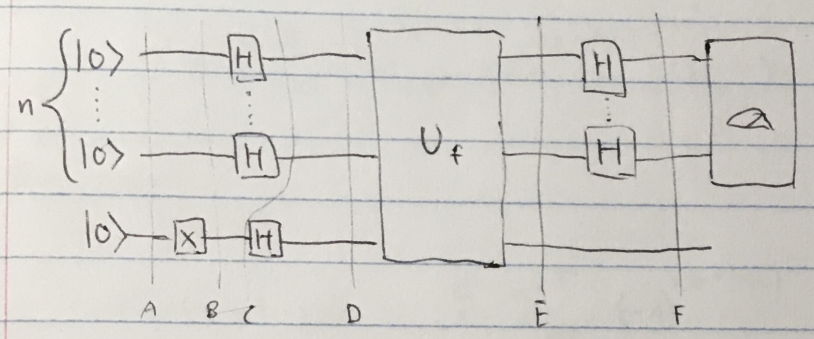
Quantum algorithm needs only 1 query instead of $2^{n-1}+1$

First, need unitary to call f: $U_f|x,y\rangle = |x, f(x) \oplus y\rangle$

eg. $|x,0\rangle \rightarrow |x, f(x)\rangle$

10/26 Deutsch-Josza Cont.

Quantum Circuit:




Note: $0^n$ means n 0s

A $|0^n, 0\rangle$

B $|0^n, 1\rangle$

C $|+^n, 1\rangle = \left(\frac{1}{\sqrt{2}^n}\sum_{x=0^n}^{1^n}|x\rangle\right)\otimes|1\rangle$, ie. equal superposition of all possible states

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

E $U_f|+^n,-\rangle$

$$U_f|x,-\rangle = \frac{1}{\sqrt{2}}\left(|x, f(x)\rangle - |x, 1\oplus f(x)\rangle\right)$$

if $f(x)=0$: $= |x,-\rangle$

if $f(x)=1$: $= -|x,-\rangle$

Thus, $U_f|x,-\rangle = (-1)^{f(x)}|x,-\rangle$

$$\therefore U_f|+^n,-\rangle = \frac{1}{\sqrt{2}^n}\sum_{x=0^n}^{1^n}(-1)^{f(x)}|x,-\rangle$$

F We show $H^{\otimes n}|z\rangle = \frac{1}{\sqrt{2}^n}\sum_{x=0^n}^{1^n}(-1)^{z\cdot x}|x\rangle$

Now:

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

10/26 Deutsch-Josza Cont.

F Cont. Applying and rearranging gives:

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

Measurement:

Coefficient of $|z\rangle$ is $\frac{1}{2^n}\sum_{x=0^n}^{1^n}(-1)^{xz+f(x)}$

For $z=0$, this is $\frac{1}{2^n}\sum_{x=0^n}^{1^n}(-1)^{f(x)}$

If $f=0$, the coefficient is $\frac{2^n}{2^n}=1$

If $f=1$, we get $\frac{-2^n}{2^n}=-1$

If $f$ is balanced, we get equal $+1$ and $-1$ which gives us $0$.

Thus, if $f$ is constant we measure $|0\rangle$ w/ probability $|\pm 1|^2=1$, and if $f$ is balanced we measure $|0\rangle$ w/ probability $|0|^2=0$

So if you measure $|z\rangle$ as $|0\rangle$, you know its constant and vice versa for $|1\rangle$ and balanced.

Limitations:

1. Bound on queries, not running time
2. "Toy" problem - no practical uses
3. Assumes "black box"
4. Compares to P not BPP (which allows occasional wrong answers)

10/26 Shor's Algorithm

Shor discovered efficient quantum algorithm for factoring, a very important problem, in 1994

↳ $O(n^3)$ vs classical $O(2^{n/3})$

Shor actually solves the "order finding" problem (which gives you factoring):

Given a number $y<N$, find smallest $r$ such that $y^r = 1 \pmod N$

Factoring by Order-Finding

Given $N$, find a divisor other than 1 or $N$

↳ suffient to find some $x$ s.t. $x$ and $N$ have a common factor $>1$, b/c Euclid's algorithm can find that factor

↳ suffient to find solution to $x^2 = 1 \pmod N$, which is equivalent to $x^2 - 1 = (x+1)(x-1) = 0 \pmod N$, so one of $(x+1)$ or $(x-1)$ has a common prime factor with $N$

↳ This is just order-finding

Find some $r$ for random $y$ s.t. $y^r = 1 \pmod N$

If $r$ is even, $(y^{r/2})^2 = 1 \pmod N$, so $y^{r/2}$ is our $x$

Since $y$ is random, $r$ is even w/ probability $\frac{1}{2}$.

If we try 2 different $y$, we have only $\frac{1}{4}$ chance of not getting an even number, which is good enough for BQP.

Note: order-finding can be generalized to <u>period-finding</u>:

Given function $f$, find smallest $x$ s.t. $f(r+x) = f(r)$ for all $r$

Quantum algorithm works for general problem