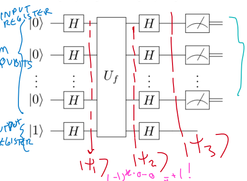
Generalizing Deutch's algorithm, Bernstein-Vazirani

Generalizing Deutch to f:{0,1}^{n}→{0,1}

- Now there are 2^n inputs "x". Suppose you know a priori that f(x) is either constant or balanced. How many queries to certify that it's constant?



IF YOU GET 0 ⇒ f CONSTANT!

IF YOU DO NOT GET 0 ⇒ f IS BALANCED!

$H^{\otimes n}|x\rangle_n = \frac{1}{\sqrt{2^n}}\sum_{z\in\{0,1\}^n}(-1)^{x\cdot z}|z\rangle$

n TIMES, $x = x_{n-1}x_{n-2}\cdots x_0$, ↑ # OF BITS IN x!

$x\cdot z = (x_{n-1}\cdot z_{n-1})\oplus(x_{n-2}\,z_{n-2})\oplus\cdots\oplus(x_0 z_0)$

EXAMPLE:

$H\otimes H|10\rangle = (H|1\rangle)(H|0\rangle) = \left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)\left(\frac{|0\rangle+|1\rangle}{\sqrt{2}}\right)$

$= \frac{1}{\sqrt{2^2}}\left\{(-1)^{10\cdot 00}|00\rangle + (-1)^{10\cdot 01}|01\rangle + (-1)^{10\cdot 10}|10\rangle + (-1)^{10\cdot 11}|11\rangle\right\}$

(=+1, =+1, =−1, =−1)

- IF f IS BALANCED: ONLY A FEW QUERIES TO FIND OUT!
- IF f IS CONSTANT, IT WILL TAKE YOU $\frac{2^n}{2}+1 = 2^{n-1}+1$ QUERIES TO CERTIFY!

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

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

"PHASE KICK-BACK TRICK"

$= |x\rangle\frac{|0\rangle-|1\rangle}{\sqrt{2}}$ IF $f(x)=0$

$= |x\rangle\frac{|1\rangle-|0\rangle}{\sqrt{2}}$ IF $f(x)=1$

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

$H^{\otimes n}$, H

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

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

$f(x)+z\cdot x = 0$ FOR $z=0$!

$\text{prob.}\left(|z\rangle=|0\cdots0\rangle\right) = \left|\frac{1}{2^n}\sum_{x\in\{0,1\}^n}(-1)^{f(x)}\right|^2 = \begin{cases}1 & \text{IF } f(x) \text{ IS CONSTANT}\\ 0 & \text{IF } f(x) \text{ IS BALANCED}\end{cases}$

Bernstein-Vazirani problem

- Suppose f:{0,1}^{n}→{0,1} is f(x)=a· x, where "a" is an unknown constant n-bit string:

$$f(x) = a\cdot x = (a_{n-1}x_{n-1})\oplus(a_{n-2}x_{n-2})\oplus\cdots\oplus(a_0x_0)$$

- How many "classical" queries to determine value of "a"?
- Ans.: Need to do n queries. To determine a_0, query x=00…01, to determine a_1, query x=00…010, etc.
- The generalized Deutch's algorithm finds the answer with one query!

$|\psi_3\rangle = \sum_{z\in\{0,1\}^n}\left(\frac{1}{2^n}\sum_{x\in\{0,1\}^n}(-1)^{f(x)+z\cdot x}\right)|z\rangle|1\rangle$ ($f(x)+z\cdot x = a\cdot x$)

$= \sum_{z\in\{0,1\}^n}\left(\frac{1}{2^n}\sum_{x\in\{0,1\}^n}(-1)^{(a\oplus z)\cdot x}\right)|z\rangle|1\rangle$

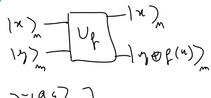
$= \sum_{z_0\in\{0,1\}}\sum_{z_1\in\{0,1\}}\cdots\sum_{z_{n-1}\in\{0,1\}}(-1)^{(a_{n-1}\oplus z_{n-1})x_{n-1}}(-1)^{(a_{n-2}\oplus z_{n-2})x_{n-2}}\cdots(-1)^{(a_0\oplus z_0)x_0}$

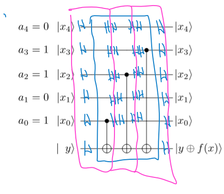
$(\cdot) = \prod_{j=0}^{n-1}\left[(-1)^{a_j\oplus z_j}+1\right] = \begin{cases}0 & \text{IF any } a_j\neq z_j\\ 2^n & \text{IF } a=z\end{cases}$

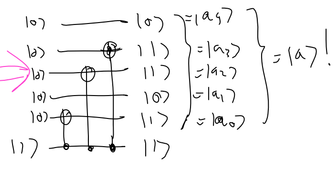
$\Rightarrow |\psi_3\rangle = |a\rangle|1\rangle$

Circuit derivation of Bernstein-Vazirani

- It's actually quite easy to implement the f(x)=a · x Oracle:







- What happens when we apply Hadamards to the left and right of all qubits ?

Summary

- Deutch's algorithm inputs a superposition state containing ALL computational basis states into the Oracle:

$$H\otimes H\otimes\cdots\otimes H\,|00\cdots0\rangle = \frac{1}{\sqrt{2^n}}\sum_{x\in\{0,1\}^n}|x\rangle$$

- Due to linearity, when the Oracle acts on this state, it evaluates f(x) for all x simultaneously! This is called "*quantum paralelism*".

- The resulting state can now be "engineered" with additional unitaries in order to yield useful information when read out. For example, applying H⊗H⊗..⊗H again converts the state into 00…0 if and only if f(x)=constant. This enables the certification that f(x)=const. with a speed exponentially faster than classical! In this case *quantum advantage increases exponentially with the size of the problem*.
- If f(x) is instead equal to a.x, where "a" is a n-bit string, the end state of the input register becomes equal to the constant "a". For this case the quantum speed up increases linearly with the size of the problem.