Deutsch Algorithm

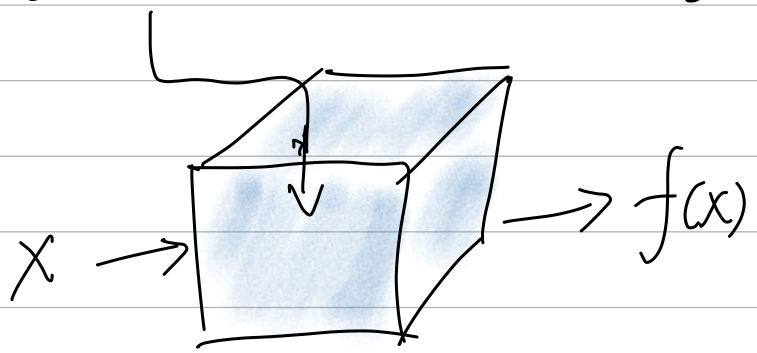
Problem Statement:

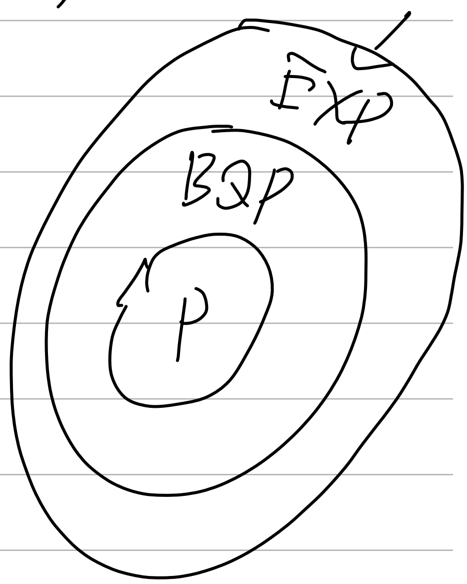
Given oracle access to a boolean function $f: \{0,1\} \to \{0,1\}$. Determine whether:

f

X	Y
0	
1	

$$\begin{cases} f(0) = f(1) \ ; \text{ or} \\ f(0) \neq f(1) \end{cases}$$





Quantum Supremacy:

- Classically, it requires 2 queries to solve it.

- Deutsch Algorith solves it using 1 Quantum query.

Quantum Oracle Access to classical functions

1: $O_f \; |x\rangle|y\rangle \mapsto |x\rangle|y \oplus f(x)\rangle$

2: Phase Oracle: $PhO_f \; |x\rangle \mapsto (-1)^{f(x)} |x\rangle$

(single-bit output functions

Algorith:

1. $H \cdot |0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$

2. Query phase oracle using ↙:

$$PhO_f \cdot \left[\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)\right] = \frac{1}{\sqrt{2}}\left[PhO_f|0\rangle + PhO_f|1\rangle\right]$$

$$= \frac{1}{\sqrt{2}}\left((-1)^{f(0)}|0\rangle + (-1)^{f(1)}|1\rangle\right)$$

$$= \begin{cases} (-1)^{f(0)}|+\rangle & \text{if } f(0) = f(1) \\ (-1)^{f(0)}|-\rangle & \text{if } f(0) \neq f(1) \end{cases}$$

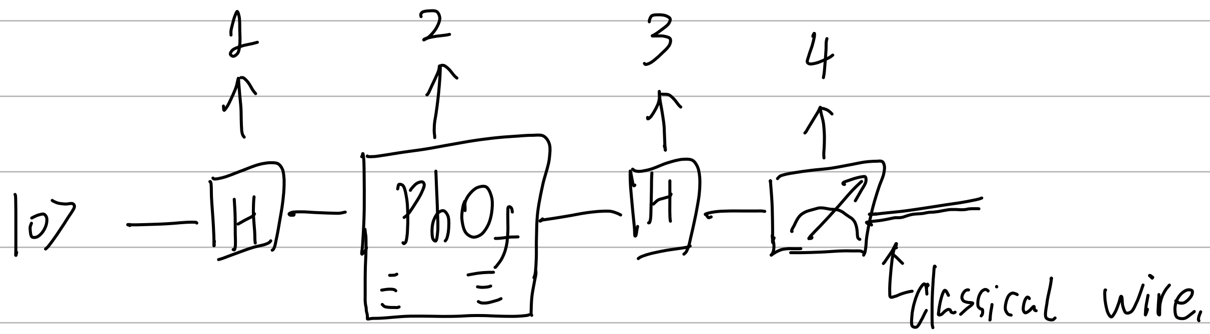
3: Apply H to the above state:

$H = H^{\dagger} = H^{-1}$ $\quad$ $H|0\rangle = |+\rangle$, $H|1\rangle = |-\rangle$ $\Rightarrow$

$$\begin{cases} (-1)^{f(0)}\,|0\rangle & \text{if } f(0)=f(1) \\ (-1)^{f(0)}\,|1\rangle & \text{if } f(0)\neq f(1) \end{cases}$$

4. Measure under comp./stand. basis: $\{|0\rangle, |1\rangle\}$

Quantum Circuit Diagram



Deutsch-Josza Algorithm:

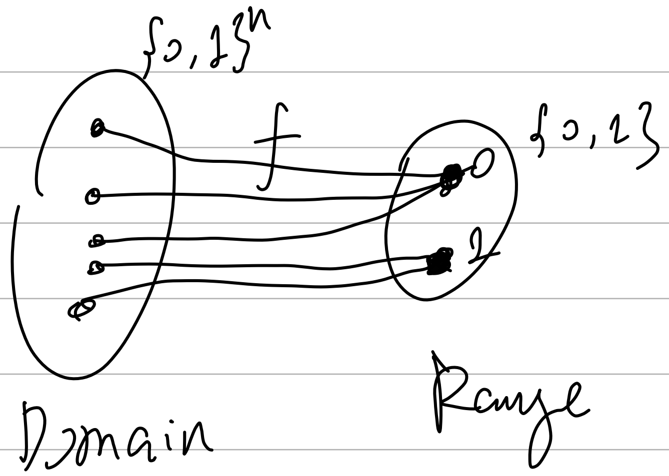
Problem Statement:

Given Oracle access to a boolean function:

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

Promise Problem. $\begin{cases} f \text{ is a constant function.} \\ f \text{ is a balanced function.} \end{cases}$

$f(x_1) = f(x_2) = \cdots = f(x_{2^n})$



2^n x in total.

$$f \begin{cases} \frac{2^n}{2} - 1 \text{ of } x\text{'s} \longrightarrow 0 \\ \frac{2^n}{2} + 1 \text{ of } x\text{'s} \longrightarrow 1 \end{cases}$$

no such bad f

Quantum Supremacy.

- Classical Algo needs to make $\frac{2^n}{2}+1$ queries to solve it perfectly.

∃ Randomized classical algorithm ~~that~~ makes constant # of queries, to solve ~~this~~ problem with small error ε.

- Deutsch-Josa solves it using 1 quantum query and solving it ~~perfectly~~.

Thm (Tensor of Hardamard Gates)

$\forall x \in \{0,1\}$, $H|x\rangle = \begin{cases} |+\rangle & x=0 \\ |-\rangle & x=1 \end{cases} = \dfrac{|0\rangle + (-1)^x |1\rangle}{\sqrt{2}}$

$\forall x \in \{0,1\}^n$,

$$\underbrace{H \otimes \cdots \otimes H}_{n \text{ times}} |x\rangle =: H^{\otimes n} |x\rangle = H^{\otimes n} |x_1 x_2 \cdots x_n\rangle$$

$$= H|x_1\rangle \otimes H|x_2\rangle \otimes \cdots \otimes H|x_n\rangle$$

$$= \left(\frac{|0\rangle + (-1)^{x_1}|1\rangle}{\sqrt{2}}\right) \otimes \cdots \otimes \left(\frac{|0\rangle + (-1)^{x_n}|1\rangle}{\sqrt{2}}\right)$$

Binary Quantum Fourier Transform

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

where $\begin{cases} y = y_1 y_2 \cdots y_n, \text{ and} \\ \langle x, y\rangle = \sum_{i=1}^{n} x_i \cdot y_i \pmod 2. \end{cases}$

$$\underbrace{(H\otimes H\otimes H \cdots \otimes H)\overbrace{(|0\rangle\otimes|1\rangle\otimes|0\rangle\otimes|0\rangle\otimes|0\rangle\otimes|0\rangle\otimes|1\rangle \cdots)}^{|0 1 0 0 0 1 \cdots\rangle}} = H|0\rangle\otimes H|1\rangle \cdots$$

$$\underline{(A\otimes B)(C\otimes D) = (A\cdot C)\otimes(B\cdot D)}$$

$\forall\ x \in \{0,1\}^n$, define:

$$|\tilde{x}\rangle := \frac{1}{\sqrt{2^n}} \sum_{y\in\{0,1\}^n} (-1)^{\langle x,y\rangle} |y\rangle$$

Claim: $\underbrace{\{ |\widetilde{00000}\rangle, |\widetilde{00\cdots 1}\rangle, \cdots |\widetilde{11\cdots 1}\rangle \}}_{\text{runs over all possible } x\in\{0,1\}^n}$

form a set of orthonormal basis of $\mathbb{C}^{2^n}$.

$00\cdots 0 = e_1$ $\qquad \{|e_1\rangle \cdots |e_{2^n}\rangle\}$ for $\mathbb{C}^{2^n}$

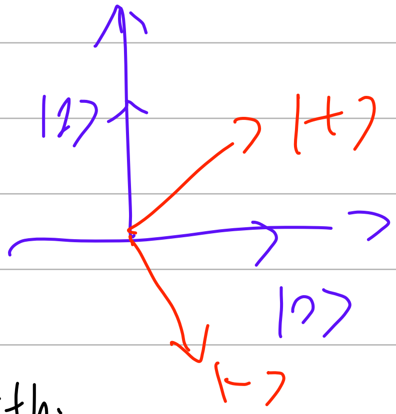
$00\cdots 1 = e_2$

$\vdots$ $\qquad \{|\tilde{e}_1\rangle \cdots |\tilde{e}_{2^n}\rangle\}$ for $\mathbb{C}^{2^n}$

$1\cdots 1 = e_{2^n}$

$$\forall j \in \{1 \cdots 2^n\}$$

$$H^{\otimes n}|e_j\rangle = |\tilde{e}_j\rangle$$



Algorith:

1. Prepare a uniform superposition:

$$H^{\otimes n}|0\cdots 0\rangle = \frac{1}{\sqrt{2^n}} \cdot \sum_{y\in\{0,1\}^n} (-1)^{\langle 0\cdots 0,\, y\rangle} |y\rangle$$

$$= \frac{1}{\sqrt{2^n}} \sum_{y\in\{0,1\}^n} |y\rangle$$

renaming

$y \to x$

$$= \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle$$

2. Query ~~the~~ phase Oracle:

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

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

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

$$= \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} (-1)^{f(x)} \underbrace{\left[H^{\otimes n} \cdot |x\rangle \right]}_{}$$

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

Focusing on $|y\rangle = |00\cdots 0\rangle$.

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

$$= \begin{cases} \pm |\vec{0}\rangle & f \text{ is constant.} \\ 0 \cdot |\vec{0}\rangle = 0 & f \text{ is balanced.} \end{cases}$$

$$\mathbb{C}^{2^n} = \underline{\operatorname{span}\{|\vec{0}\rangle\}} \cup \operatorname{span}\{|1\rangle \cdots |2^n - 1\rangle\}$$

$$A \qquad\qquad A^{\perp}$$

<u>POVM</u>:

$$\{ E_0 = |0\rangle\langle 0| , \ E_1 = \mathbb{1}_{\mathbb{C}^{2^n}} - |0\rangle\langle 0| \}$$

$$= |1\rangle\langle 1| + |2\rangle\langle 2| + \cdots |2^n - 1\rangle\langle 2^n - 1|$$

1: $H^{\otimes n}$