

Basic idea of quantum computing and the operation of a specific quantum computing algorithm

Quantum Engineering II

3rd of June 2021

Quantum Parallelism

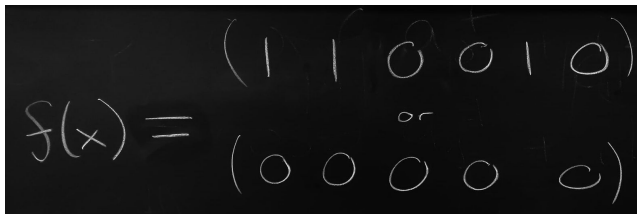
Quantum Parallelism: Parallel computation on superposition states

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

Interference can make calculation that depends on all results.

Deutsch-Jozsa algorithm

Determine if the function of 2^n entries even or odd?



A handwritten image on a black background showing a function $f(x)$ defined by two possible output vectors. The first vector is $(1, 1, 0, 0, 1, 0)$ and the second vector is $(0, 0, 0, 0, 0, 0)$. The word "or" is written between the two vectors, indicating that the function is either constant (all 1s) or balanced (all 0s).

$$f(x) = \begin{pmatrix} 1 & 1 & 0 & 0 & 1 & 0 \end{pmatrix} \text{ or } \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Classical: Worst case $\frac{2^n}{2} + 1$ evaluations

Deutsch-Jozsa algorithm

Begin with a query and answer registers

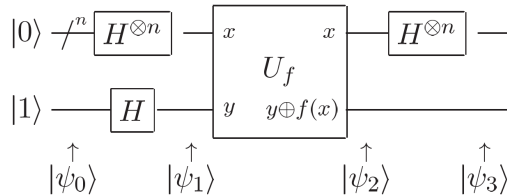
$$|\psi_0\rangle = |0\rangle^{\otimes n}|1\rangle$$

Put into superposition with Hadamard

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

Calculate function $|-\rangle \rightarrow (-1)^{f(x)}|-\rangle$

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



Deutsch-Jozsa algorithm

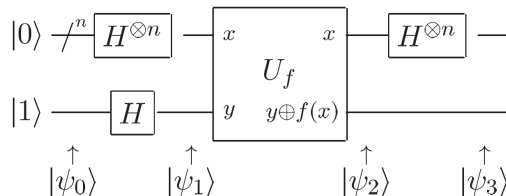
Use Hadamard on all again

$$H^{\otimes n}|x\rangle_n = \sum_z (-1)^{x \cdot z} |z\rangle_n / \sqrt{2^n}$$

$$|\psi_3\rangle = \sum_z \left(\sum_x \frac{(-1)^{x \cdot z + f(x)}}{2^n} \right) |z\rangle_n |-\rangle$$

$$c_z = \sum_x \frac{(-1)^{x \cdot z + f(x)}}{2^n}$$

$$c_0 = \sum_x \frac{(-1)^{f(x)}}{2^n} \begin{cases} \pm 1 & \text{(constant)} \\ 0 & \text{(balanced)} \end{cases}$$



If projection onto $|0\rangle_n$ then (constant) if not then (balanced)

Pros and Cons

Pros

Only one iteration is needed compared $\frac{2^n}{2} + 1$

Cons

Not important problem

Classically you can be lucky after 2 iterations