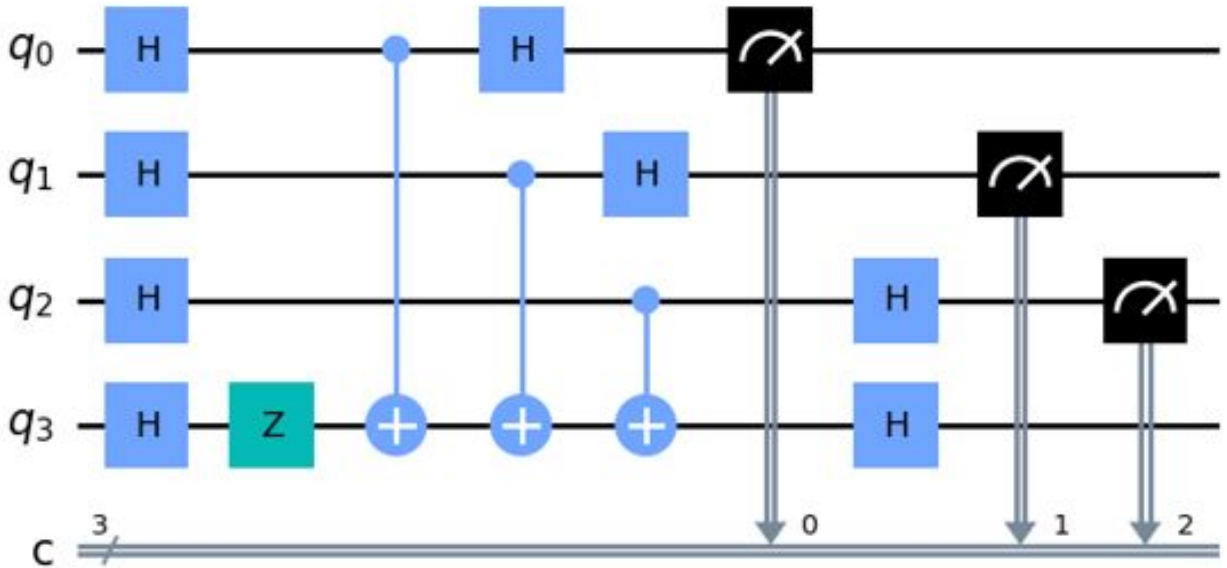
Intro to Quantum Algorithms

Michael Silver, ECE2T6, University of Toronto Quantum Computing Club

Previously...

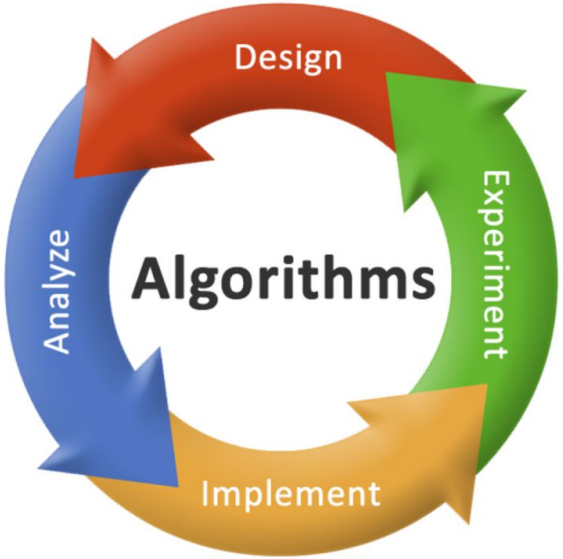


Quantum Circuit Model

Algorithm Design and Computer Science

Computing with Purpose

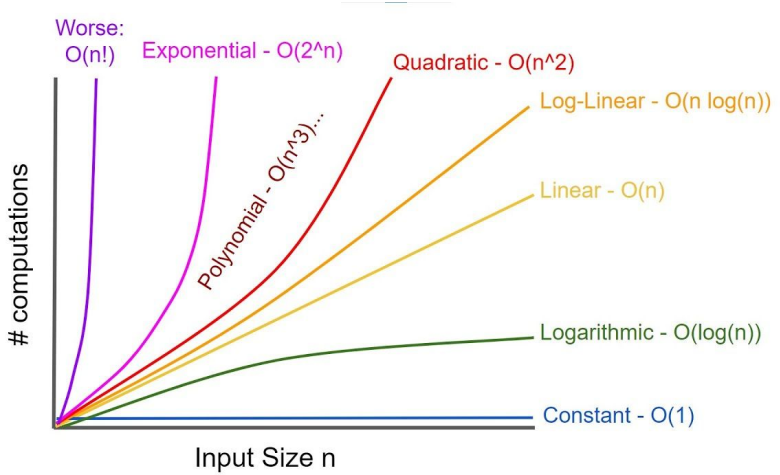
- What?
- Why?



Advantage

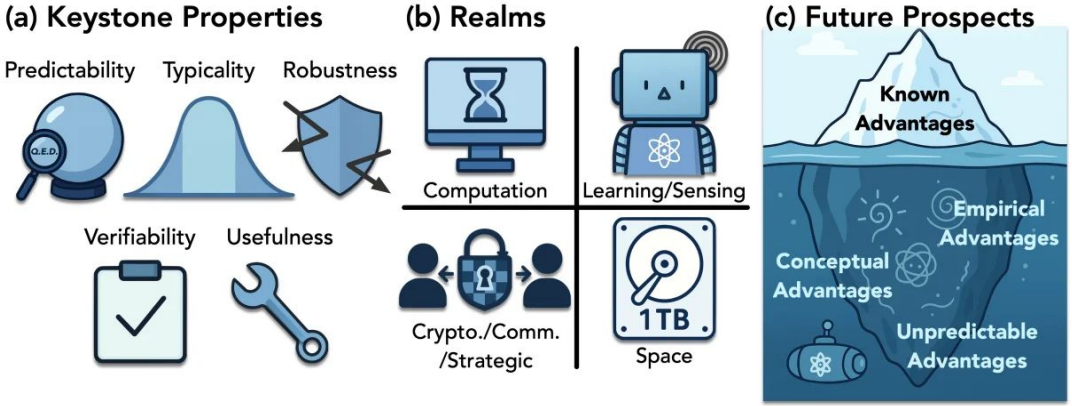
- How good?

-> Computational Complexity



Quantum Advantage

- NOT one size fits all
- No broad category where quantum beats classical
  - Factorization (Shor's algorithm)
  - Chemical and Physics simulations
  - Quantum Machine Learning
  - Optimization
  - Oracle <- We will talk about today



How Algorithms Ask Questions

- The **ORACLE** model -> For an queryable unknown function f(x), we want to get as much information out of it as possible using an oracle algorithm
- Example: We have a binary password we want to unlock; f(x) is the ‘lock function’ of each digit
- How would we find the password? What is the complexity of your solution?

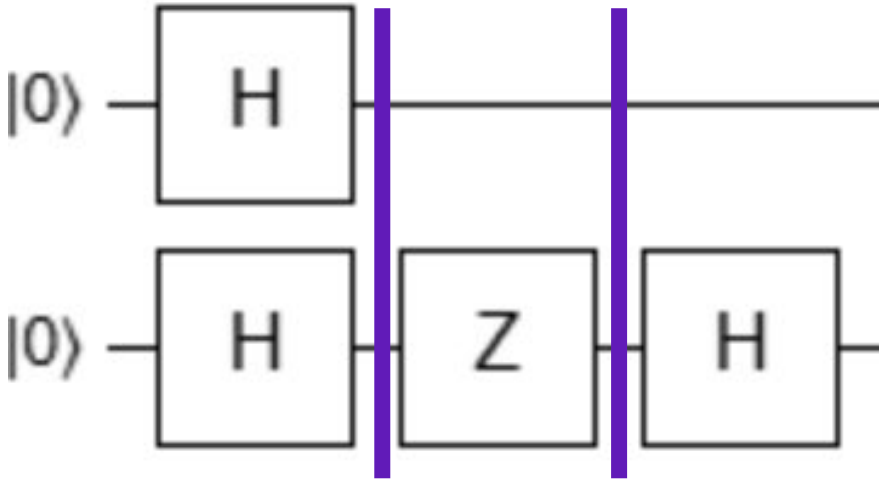
$$f(\mathbf{x}) = \begin{cases} 1 & \mathbf{x} = \mathbf{s}; \\ 0 & \text{otherwise.} \end{cases}$$

(1 means correct, 0 means incorrect, s represents the correct input)

The Quantum Oracle

- We must make our oracle a quantum (unitary) operator (we call this U_f)
- Becomes our ‘mystery’ gate that applies f(x) $|x, y\rangle \rightarrow |x, y \oplus f(x)\rangle$
- Oracle is just a gate that ‘stores’ f(x) on the second qubit
  - Keeps function reversible, can you think why that is?
- Where the advantage comes in: using superposition
- If we put x in a superposition, we can learn many things about f(x) from just querying it once! We let **quantum interference** reveal a pattern
- **Interference:** playing with qubit amplitudes to get desired outcome
  - Adding up -> constructive interference (state becomes more likely)
  - Cancel out -> destructive interference (state becomes less likely)

Interference Example



$$\rightarrow \frac{1}{\sqrt{2}}(|01\rangle + |11\rangle)$$

$$|00\rangle \rightarrow \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle) \rightarrow \frac{1}{2}(|00\rangle - |01\rangle + |10\rangle - |11\rangle)$$

The Deutsch-Jozsa Algorithm

Problem: We are given a black-box function f(x) that takes n bits and outputs 0 or 1

We are promised that f is either:

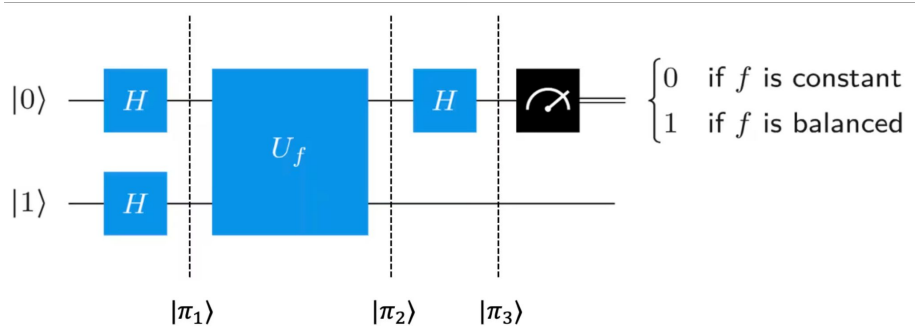
- Constant: same answer for every x, OR
- Balanced: outputs 0 for exactly half of all inputs and 1 for other half

Goal: decide if f is constant or balanced

Classically: O(2^n) queries | **Quantum:** Needs 1 query

How it works

1) Put both qubits into superposition (x-> + state and y-> - state)
2) Query the oracle once
   a) Because the y is in the - state, flipping it when f(x)=1 multiplies the state by (-1)^f(x)
   b) Answer gets hidden in the phase of the first qubit
3) Applying Hadamard to first qubit; two cases:
   a) If f is constant
      i) f(0)=f(1)
   b) If f is balanced
      i) f(0) /= f(1)



$$|\pi_1\rangle = \frac{|0\rangle + |1\rangle}{\sqrt{2}} \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}}$$

$$|\pi_2\rangle = \frac{(-1)^{f(0)}|0\rangle + (-1)^{f(1)}|1\rangle}{\sqrt{2}} + \frac{|0\rangle + |1\rangle}{\sqrt{2}}$$

$|\pi_3\rangle$:

If f constant, first qubit: $\frac{|0\rangle + |1\rangle}{\sqrt{2}}$, when measured: $|0\rangle$

If f balanced, first qubit: $\frac{|0\rangle - |1\rangle}{\sqrt{2}}$, when measured: $|1\rangle$

Deutsch-Jozsa in summary

- We figured out our black-box function f(x) with only one oracle query, as opposed to a classical algorithm which may need up to 2^(n-1) + 1 queries
  - Superposition queries all inputs at once
  - Oracle writes answer as **phase flip**
  - Hadamard turns phase information into measurable probability
  - Where is the interference in this case? What case corresponds to the types of interference?
- This is our first example of **quantum advantage**, where a quantum algorithm beats a classical one
- Note that Deutsch-Jozsa is not a terribly useful algorithm, and is more meant to demonstrate the idea of the quantum oracle