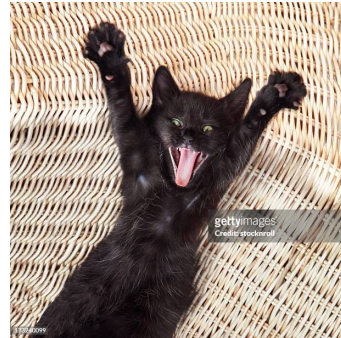


A Brief Introduction to Quantum Computing

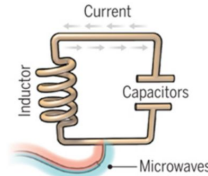
“I think I can safely say that nobody understands quantum mechanics” - Feynman



Representation of Data

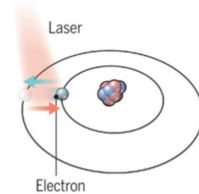
Quantum bit → Qubit

- Can be in state 0, in state 1, or in any other state that is a linear combination of the two states → quantum systems evolve according to linear equations (Schrödinger's equation: $d|\psi\rangle/dt = -i\hat{H}(t)|\psi\rangle/\hbar$,)
- Can be partially measured with a given probability
- They are changed by measurement
- Cannot be copied or erased



Superconducting loops

A resistance-free current oscillates back and forth around a circuit loop. An injected microwave signal excites the current into superposition states.



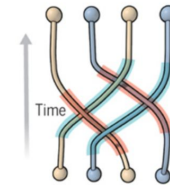
Trapped ions

Electrically charged atoms, or ions, have quantum energies that depend on the location of electrons. Tuned lasers cool and trap the ions, and put them in superposition states.



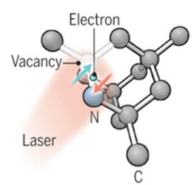
Silicon quantum dots

These "artificial atoms" are made by adding an electron to a small piece of pure silicon. Microwaves control the electron's quantum state.



Topological qubits

Quasiparticles can be seen in the behavior of electrons channeled through semiconductor structures. Their braided paths can encode quantum information.



Diamond vacancies

A nitrogen atom and a vacancy add an electron to a diamond lattice. Its quantum spin state, along with those of nearby carbon nuclei, can be controlled with light.

How do we initialize the qubits?

Superconducting qubits:

- Kept at extremely low temperatures.
- They are designed such that their lowest energy state corresponds to the $|0\rangle$ state.
- To ensure that the qubit is in the $|0\rangle$ state, microwave pulses can be applied to flip qubits in the $|1\rangle$ state to the $|0\rangle$ state.



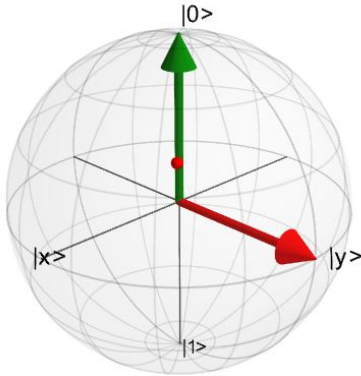
Superposition

Definition: Creation of a quantum state where a qubit exists in multiple bits simultaneously.

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

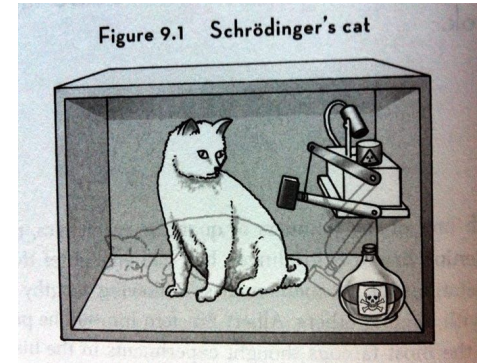
$$|\alpha|^2 + |\beta|^2 = 1$$

Complex numbers that
specify the probabilities of
measuring the qubit in the
states $|0\rangle$ and $|1\rangle$



Bloch sphere

North pole represents 0,
south pole represents 1.
Any point on the sphere
represents a
superposition state.





Entanglement

Definition: Phenomenon where the states of two or more qubits become linked, regardless of physical separation. So, a state of one qubit directly depends on the state of the other.

- If the state of one changes, the other is instantly adjusted as well.
- If a measurement is made on one, the other will automatically collapse

$$|\psi\rangle = \frac{1}{\sqrt{2}} (|00\rangle + |11\rangle) \longrightarrow$$

Two qubit example. Here, if qubit A is measured to be $|0\rangle$, qubit B will also be $|0\rangle$

Interference

Definition: Quantum states can combine in ways that amplify or cancel the probabilities of outcomes → constructive and destructive interference

Example: Applying a Hadamard gate (H) to a qubit twice:

- Has no classical equivalent

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \rightarrow$$

Input	Output
0	$\frac{1}{\sqrt{2}} (0\rangle + 1\rangle)$
1	$\frac{1}{\sqrt{2}} (0\rangle - 1\rangle)$

$$\rightarrow |0\rangle \rightarrow H \rightarrow \frac{(|0\rangle + |1\rangle)}{\sqrt{2}} \rightarrow H \rightarrow |0\rangle$$

Regarding the Necessity of S.E.I.

While all three phenomena are not necessary for every single computation, the algorithms and calculations that provide a quantum advantage do use all three simultaneously.

1. Superposition alone: useful in quantum rand. number generators; enables parallelism for simultaneous evaluation of functions
2. Entanglement alone: needed for quantum teleportation; used in quantum cryptography protocols
3. Interference alone: essential for quantum sensing
4. All three: Shor's and Grover's algorithms, most quantum machine learning algorithms

Data Retrieval and Manipulation

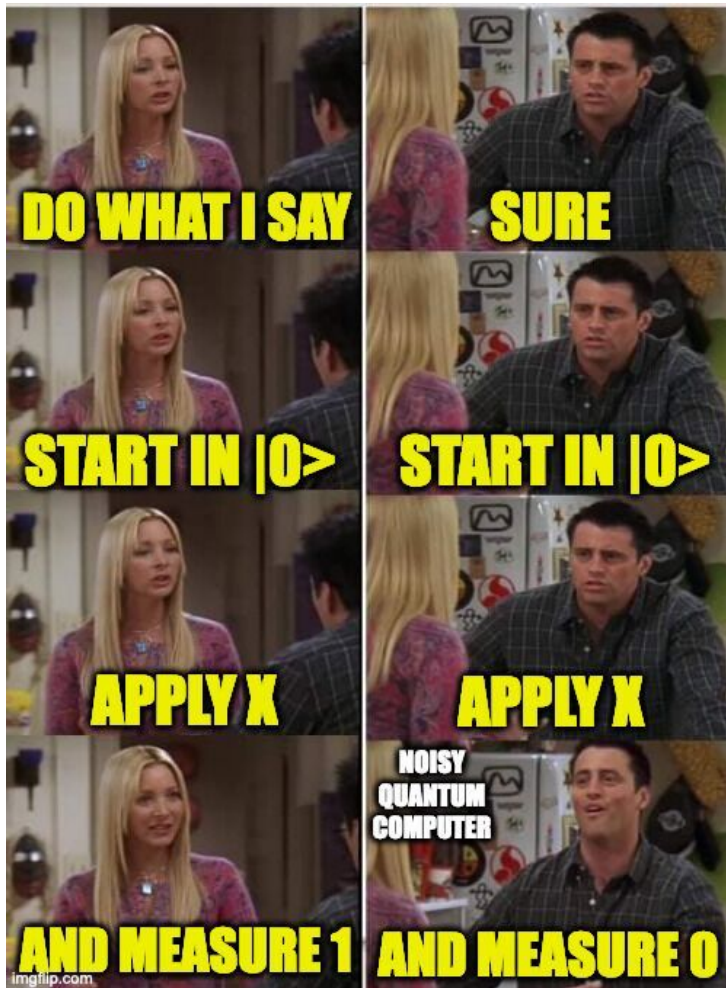
- In a quantum computer, a register (collection of qubits) can exist in a superposition of many different states. For example, an n -qubit register can represent 2^n possible combinations of 0s and 1s at the same time.
- However, when you measure data from a quantum register, the superposition collapses to a single classical state.

Why does this happen? Because of **Quantum Decoherence** → loss of superposition due to a spontaneous interaction between a quantum system and its environment.

E.G., Given the following state, The probability of reading the rightmost bit as 0 is

$$|\psi\rangle = 0.316|00\rangle + 0.447|01\rangle + 0.548|10\rangle + 0.632|11\rangle$$

$$|0.316|^2 + |0.548|^2 = 0.4$$



Quantum Gates and Circuits

- Like classical logic gates, quantum gates are basic q. circuits that operate on qubits.
- Every gate operation U has to be unitary, satisfying $UU^* = I$
 - Q. gates ensure that probabilities always sum to 1
- All q. gates are reversible \rightarrow input can be conducted from the output

CNOT Gate (Controlled NOT)

Acts on two qubits:

- If the control qubit is set to 0, target qubit is the same
- If the control qubit is set to 1, target qubit is flipped

$$\begin{aligned} |00\rangle &\rightarrow |00\rangle, & |01\rangle &\rightarrow |01\rangle \\ |10\rangle &\rightarrow |11\rangle, & |11\rangle &\rightarrow |10\rangle \end{aligned}$$

Matrix Representation

$$CNOT = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Bell State- using qubits to generate an entanglement state

1. Initialization: start with two qubits in the state $|0\rangle$. The joint state of the system will be $|00\rangle$.
2. Apply an H gate to the first qubit: the Hadamard gate transforms the state $|0\rangle$ into the superposition:

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

This will result in the combined state of:

$$\frac{1}{\sqrt{2}} (|0\rangle + |1\rangle) \otimes |0\rangle = \frac{1}{\sqrt{2}} (|00\rangle + |10\rangle)$$

3. Apply a CNOT gate: keep the first qubit as the control and the second qubit as the target

The CNOT gate flips the state of the target qubit if the control is $|1\rangle$

- The state $|00\rangle$ remains $|00\rangle$ and the state $|10\rangle$ becomes $|11\rangle$

Quantum Programming

Quantum Computation Language (QCL)

```
/* Remove "/" if starting interpreter with -n option */  
// extern operator H(quireg q);
```

C-like syntax

```
procedure FlipCoin() {
```

```
  quireg q[1]; int x;
```

```
  reset;
```

```
  H(q);
```

```
  measure q, x;
```

```
  if x == 1 { print "Heads"; }
```

```
  if x == 0 { print "Tails"; }
```

```
  reset;
```

```
}
```

*allows combining of
quantum and
classical code*

IBM QISKit (Quantum Information Science Kit)

```
from qiskit import QuantumRegister, ClassicalRegister, QuantumCircuit  
from qiskit.tools.visualization import circuit_drawer  
import numpy as np
```

```
qr = QuantumRegister(2)
```

```
cr = ClassicalRegister(2)
```

```
qp = QuantumCircuit(qr,cr)
```

```
qp.rx( np.pi/2,qr[0])
```

```
qp.cx(qr[0],qr[1])
```

```
qp.measure(qr,cr)
```

```
circuit_drawer(qp)
```

Quantum Programming, cont.

Q#

```
namespace QuantumExample {  
    open Microsoft.Quantum.Primitive;  
  
    operation HelloQuantum() : Unit {  
        Message("Hello, Quantum!");  
        return ();  
    }  
}
```

Cirq

```
import cirq  
  
# Create a quantum circuit  
circuit = cirq.Circuit()  
qubits = cirq.LineQubit.range(2)  
circuit.append([cirq.H(qubits[0]), cirq.CNOT(qubits[0], qubits[1])])  
  
# Simulate the circuit  
simulator = cirq.Simulator()  
result = simulator.run(circuit, repetitions=1000)  
print(result.histogram(key='0,1'))
```

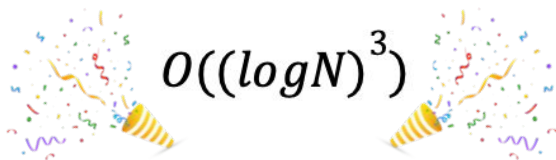
Shor's Algorithm

Problem Statement: given an integer N , find its prime factors p and q such that $N = p \times q$.

Classical algorithms that solve this problem take super-polynomial time: 🙅


- Trial Division → checks all integers up to the square root of N , runs in $O(\sqrt{N})$
- Pollard's Rho algorithm → uses a pseudorandom sequence to find factors, runs in $O(\sqrt[4]{N})$
- General Number Field Sieve → factorizes large integers using algebraic number fields, runs in $O(\exp((\frac{64}{9})^{1/3}(\log N)^{1/3}(\log \log N)^{2/3}))$ 😬

Shor's quantum algorithm runs in polynomial time:


$$O((\log N)^3)$$

Overview of Shor's Algorithm

Problem Statement: given an integer N , find its prime factors p and q such that $N = p \times q$.

1. Choose a random integer a such that $1 < a < N$
2. Compute the greatest common divisor (GCD) of a and N .
If $\text{GCD}(a, N) \neq 1$, a non-trivial factor of N is found.
-  3. Find the order r of a modulo N , which will be the smallest positive integer r such that
$$a^r = 1 \pmod{N}$$
4. If r is even and $a^{r/2} \not\equiv -1 \pmod{N}$:
 - Compute factor 1 = $\text{GCD}(a^{r/2} - 1, N)$
 - Compute factor 2 = $\text{GCD}(a^{r/2} + 1, N)$



If any of the conditions are not satisfied, repeat the algorithm by choosing a new random integer a .

Step 3- the quantum step

The quantum aspect of the algorithm comes in at step 3, where quantum period finding is used to find order r efficiently.

Order r is the smallest positive integer r such that $a^r = 1 \pmod{N}$

1. Construct a quantum state that represents a superposition of all possible exponents. Quantum computers use two registers:

- First register: initialized from 0 to $2^n - 1$ (n = number of qubits)
- Second register: initialized to 0

Normalizes the
superposition so
total probability = 1

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

Sums over all
possible values of x
that the first
register can take

Second register is
initialized to 0

The quantum step, cont.

2. Apply the modular exponentiation function $f(x) = a^x \pmod{N}$ to the second register:

$$\frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x\rangle |a^x \pmod{N}\rangle$$



• **Entanglement!** → this operation entangles the state of the first register with the results of the second register

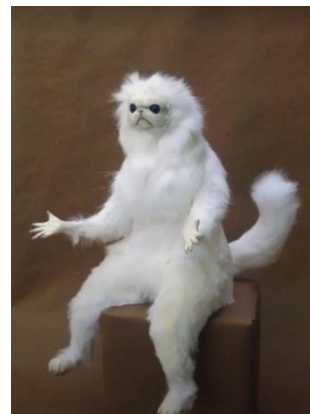
The quantum step, cont.

3. Measure the second register:

- Assuming the measurement of the second register gives us value k , the first register collapses to a superposition of all x such that $a^x \pmod N = k$
- We can denote this set of x values as $\{x_0, x_0 + r, x_0 + 2r, \dots\}$
- The state after is a superposition of the x values that satisfy $a^x \pmod N = k$

$$M \approx \frac{2^n}{r}$$

$$\frac{1}{\sqrt{M}} \sum_{j=0}^{M-1} |x_0 + jr\rangle |k\rangle$$



The quantum step, cont.

4. Apply the Quantum Fourier Transform (QFT) to the first register:

Broad Level: The QFT is the quantum analog of the Discrete Fourier Transform (DFT). It performs a linear transformation on qubits and is used for tasks such as period finding and phase estimation.

- The QFT maps the state $|x\rangle$ to a new state $|y\rangle$

Complex exponential term that encodes the phase information of the original state in the new state

$$QFT(|x\rangle) = \frac{1}{\sqrt{2^n}} \sum_{y=0}^{2^n-1} e^{2\pi i \cdot xy / 2^n} |y\rangle$$

The quantum step, cont.

- Applying the QFT to our superposition:

$$QFT \left(\frac{1}{\sqrt{M}} \sum_{j=0}^{M-1} |x_0 + jr\rangle \right) = \frac{1}{\sqrt{M}} \sum_{j=0}^{M-1} \left(\frac{1}{\sqrt{2^n}} \sum_{y=0}^{2^n-1} e^{2\pi i \cdot (x_0 + jr)y/2^n} |y\rangle \right)$$

- Simplifies to:

$$\frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} e^{2\pi i x_0 k/r} \left| k \cdot \frac{2^n}{r} \right\rangle$$



Periodic Superposition

So what does applying the QFT actually do?

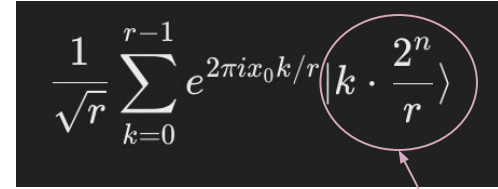
- The initial state before QFT is a superposition of states that periodically repeat with period r
- It maps the periodic superposition in the time domain to the frequency domain.



Interference! → this operation shifts the basis from the computational basis (each state represents an integer) to the frequency domain (each state represents a possible period)

After applying the QFT, the period manifests as frequency peaks at integer multiples of $\frac{2^n}{r}$

The quantum step, cont.

$$\frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} e^{2\pi i x_0 k / r} \left| k \cdot \frac{2^n}{r} \right\rangle$$


5. Post-QFT measurement of the first register

- Measurement collapses the superposition to one of the states: $|k \cdot \frac{2^n}{r}\rangle$
- The probability of measuring a particular value depends on the amplitudes of the states, which are evenly spread across the frequencies by the QFT
- Then, we are most likely to measure a value close to $m \frac{2^n}{r}$ for some integer m . Following this measurement of the first register, the state collapses to $|y\rangle$

The quantum step, cont.

6. Classical Post-Processing

- Supposed the measured value is y
- $y = m \cdot \frac{2^n}{r}$ for some integer m
- Calculate the ratio $\frac{y}{2^n} \approx \frac{m}{r}$
- Use the continued fractions algorithm to approximate $\frac{m}{r}$
- Identify the denominator r , which is in the order of $a \pmod{N}$

Error Correction

Quantum Decoherence: quantum states are HIGHLY susceptible to errors → error correction techniques are necessary to detect and correct errors without measuring the quantum state directly

Quantum Error Correction Codes (QECC)

- Shor Code: encodes one qubit into nine qubits to protect against single-qubit errors
- Steane Code: encodes one logical qubit into seven qubits to correct single-qubit errors
- Surface Codes: use a 2D grid of qubits

These encodings spread the information of the qubit across multiple physical qubits.

A series of parity checks are performed, after which the ancilla (additional) qubits are measured. These syndrome measurements then provide information about type/location of errors.

Shor's Algorithm
in theory



I will break
RSA encryption

imgflip.com

Shor's Algorithm
in real life



$15 = 3 \times 5$
(m 80% sure...)



Applications

Quantum factoring can be done exponentially faster than on classical computers

$15 = 5 \times 3 \rightarrow$ easy

$38647884621009378468487740631 = ? \times ? \rightarrow$ not easy

Best classical algorithm: 10^{24} steps	Shor's quantum algorithm: 10^{10} steps
On classical THz computer: 150,000 years	On quantum THz computer: <1 second

Big concern: public key cryptography (RSA) relies on the inability to factor such large numbers