High Speed Computing with Clustered and Quantum Computers

M. A. Robson[1]

LASACS Department
Liberal Arts and Science Academy

Period 1
Independent Study Computer Science

Fall Final, December 2025

What Is Quantum Computing?

Quantum computing covers many concepts, but it is important to first start with classical computing.

Classical Information

In a classical computer it stores information as a 1 or a 0 in what is called a bit.

Some Common Gates

NAND, NOR, AND, OR, XOR $\rightarrow$ Complex structures such as ALUs and data storage.

$$\mathrm{A} \longrightarrow \boxed{NOT} \longrightarrow \bar{A}$$

Circuit Complexity

In circuits it is important to cover the complexity required to achieve certain goals. Complexity is classified in Big O Notation.

Additionally, we can classify these further into P, NP, NP-complete, etc.

Time Complexity Example

A loop of N elements has a time complexity of $\mathcal{O}(N)$.

Theorem 1.0

Any NP complete problem can be solve in polynomial time on a classical computer.

Turing Completeness

Simply, a Turing-complete computer is one that is able to compute any and Turing-computable function.

Definition 1.0

Some computational device is taken as Turing-complete or Turing-equivalent if it is probable that it can compute the values for a function for every function of its argument. ie. A common place computer.

While Turing-complete computers are powerful, they begin to break down at the meta level in cases such as the infamous Halting Problem. (see the LASACS lunch time lecture)

A Single Qubit

A single qubit is given by $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$.

α is given to be a number in the reals and β is given to be a number in the complex space. Because of this we use a global phase to ensure α stays real.

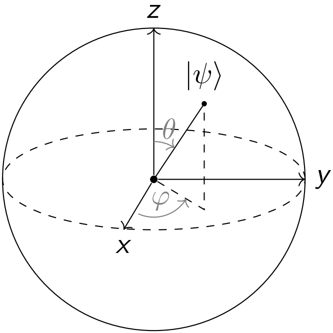
Definition 2.0

A qubit is best described by $|\psi\rangle = e^{i\gamma}(\alpha|0\rangle + \beta|1\rangle)|\alpha\epsilon\mathbb{R}, \beta\epsilon\mathbb{C}$

This is important to understand in the context of Euler's equation: $e^{ix} = \cos x + i \sin x$

Quantum Representation

$$|\psi\rangle = e^{i\gamma}(\alpha\,|0\rangle + \beta\,|1\rangle) | \alpha \epsilon \mathbb{R}, \beta \epsilon \mathbb{C}$$



$$\alpha = \cos\frac{\theta}{2}$$
$$\beta = e^{i\varphi}\sin\frac{\theta}{2}$$

Linear Algebra Crash Course

In terms of our qubit we can define some vectors to represent our previous kets for 0 and 1:

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

Plugging this into our previous formulas we can see:

$$|\psi\rangle = \alpha \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \beta \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}$$

We rewrite all of this work we have done in terms of vectors because it allows for our later use of matrices to act as the gates and for massively parallelized work.

Now with the use of these vectors we can perform some algebraic operations such as the tensor product allowing for us to combine gates and qubits.

$$\begin{pmatrix} \alpha & \beta \\ \gamma & \delta \end{pmatrix} \otimes \begin{pmatrix} A & B \\ \Gamma & \Delta \end{pmatrix} = \begin{pmatrix} \alpha \begin{pmatrix} A & B \\ \Gamma & \Delta \end{pmatrix} & \beta \begin{pmatrix} A & B \\ \Gamma & \Delta \end{pmatrix} \\ \gamma \begin{pmatrix} A & B \\ \Gamma & \Delta \end{pmatrix} & \delta \begin{pmatrix} A & B \\ \Gamma & \Delta \end{pmatrix} \end{pmatrix} =$$

$$\begin{pmatrix} \alpha A & \alpha B & \beta A & \beta B \\ \alpha\Gamma & \alpha\Delta & \beta\Gamma & \beta\Delta \\ \gamma A & \gamma B & \delta A & \delta B \\ \gamma\Gamma & \gamma\Delta & \delta\Gamma & \delta\Delta \end{pmatrix}$$

Quantum Gates

What are Quantum Gates?

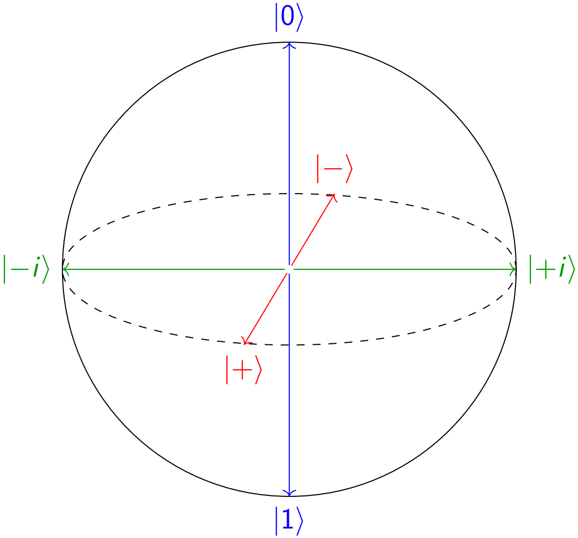
A quantum gate like a classical gate changes the state of a piece of information. Unlike many classical gates, a quantum gate is always reversible.

Pauli-X Gate

$$|0\rangle \; — \boxed{X} — \; |1\rangle$$

The Pauli-X gate acts in a similar fashion to that of a classical not gate.

Quantum Measurement



Change of Basis

It can be seen that the $|+i\rangle, |-i\rangle, |-\rangle, |+\rangle$ all fall along the equator of the bloch sphere. We can represent these points on the sphere using the following formulas:

Basis

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

It is trivial to see how this concept can be extended to further to any arbitrary basis.

More Quantum Gates

As we saw before, we can turn a $|0\rangle$ into a $|1\rangle$ using the Pauli X gate.

$$|0\rangle \;—\boxed{X}—\; |1\rangle$$

Following this idea we should be able to perform the same action on both other axis that we saw in the last slide:

Inversion Gates

$$|0\rangle \;—\boxed{X}—\; |1\rangle$$

$$|+i\rangle \;—\boxed{Y}—\; |-i\rangle$$

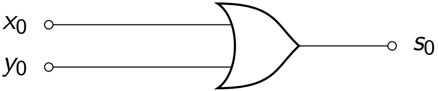
$$|+\rangle \;—\boxed{Z}—\; |-\rangle$$

Issue With Multi-Qubit Quantum Gates

In order to preform actual logic it is important to have qubits that care about multiple qubits in input. The issue is the fact that we must follow the reversibility law.

Fundamental Law of Quantum Gates

Any gate in a quantum computer must be reversible with some other gate. ie. Given some output and what the gate does you must always be able to perfectly predict what the input is.



Multi-Qubit Gates Solved

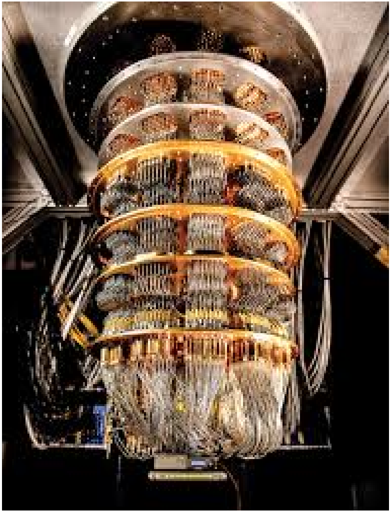
In order to have a reversible gate we could construct a gate that follows this type of format.

$$\begin{pmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{pmatrix} \rightarrow \begin{pmatrix} c_0 \\ c_1 \\ c_3 \\ c_2 \end{pmatrix}$$

We can simply see that this can be constructed as a matrix below.

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{pmatrix} = \begin{pmatrix} c_0 \\ 0 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ c_1 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ c_2 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ c_3 \\ 0 \end{pmatrix}$$

The Importance of Quantum Computers



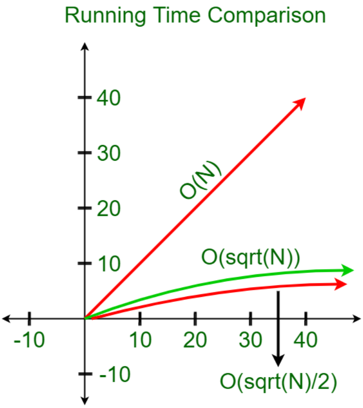
Impacts over $5 trillion in Industry!

- Encryption Cracking Speed-Up
- Molecular Chemical Interaction Simulations
- Materials Logistics

Grover's Algorithm

Remember that NP problems are those that can't be solved easily (if you can please tell me) but can be checked to be true or false very quickly.

Grover's Algorithm claims that it can solve one of these problems in $\mathcal{O}\sqrt{N}$ time using a quantum computer which is much faster than the $\mathcal{O}(N)$ time using a classical computer.

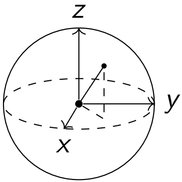


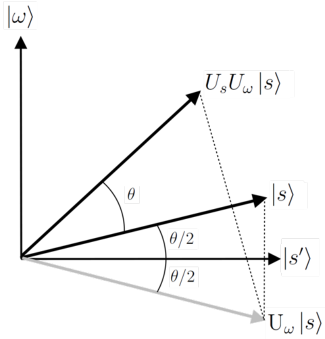
Grover's Solution

$$|s\rangle = |+\rangle^{\otimes n} = \frac{1}{\sqrt{N}} \sum_{x \epsilon \{0,1\}^n} |x\rangle$$

$$|s\rangle = \frac{1}{\sqrt{N}} |w\rangle + \frac{1}{\sqrt{N}} \sum_{i \neq w} |i\rangle$$

$$|s\rangle = \frac{1}{\sqrt{N}} |w\rangle + \sqrt{\frac{N-1}{N}} |r\rangle$$



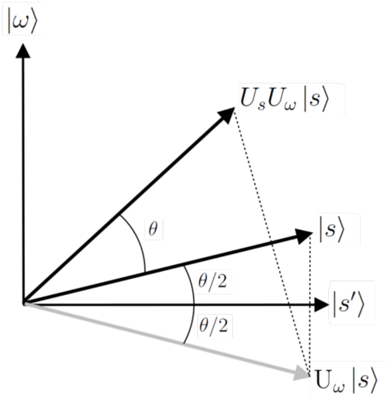


Approximating the Solution

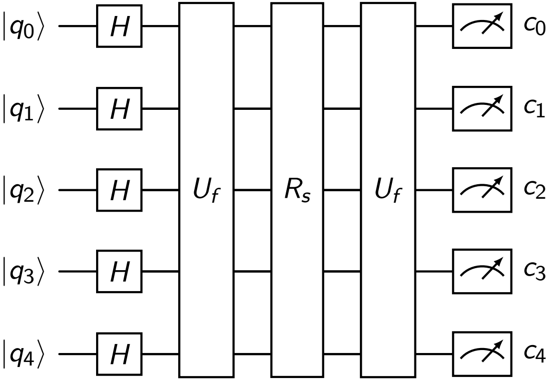
$$(-1)^1 \sin\theta\,|w\rangle + (-1)^0 \cos\theta\,|r\rangle$$

$$= -\sin\theta\,|w\rangle + \cos\theta\,|r\rangle$$

Repeatedly applying this transformation followed by a flip across the vector $|s\rangle$ will follow the process shown in diagram to the right where we are able to flip θ rads each process.



A Circuit Diagram



Expanding The Diagram