Quantum Computing The Next Frontier
Project Fibonacci STEAM Leadership Conference

Roopesh Mathur

July 28 2026

Quantum Computing - Beginnings

Quantum Computers – first proposed by Prof. Richard Feynman in 1981 as a new way of computing inspired by nature

Simulating Physics with Computers

Richard P. Feynman

Department of Physics, California Institute of Technology, Pasadena, California 91107

Received May 7, 1981

1. INTRODUCTION

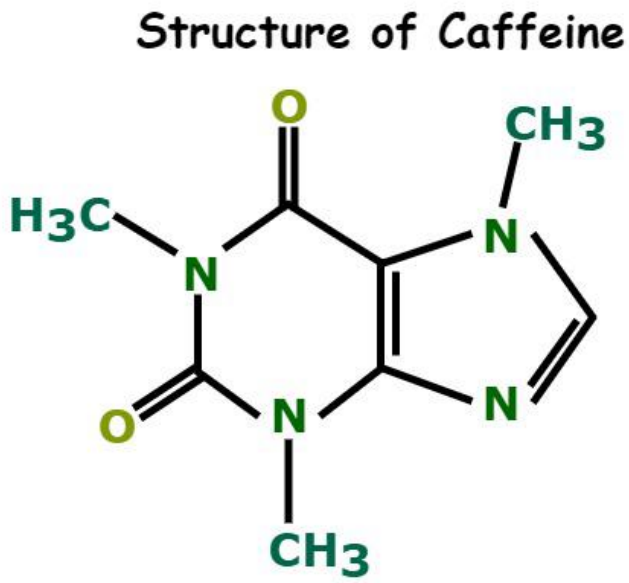
On the program it says this is a keynote speech—and I don't know what a keynote speech is. I do not intend in any way to suggest what should be in this meeting as a keynote of the subjects or anything like that. I have my own things to say and to talk about and there's no implication that anybody needs to talk about the same thing or anything like it. So what I want to talk about is what Mike Dertouzos suggested that nobody would talk about. I want to talk about the problem of simulating physics with computers and I mean that in a specific way which I am going to explain. The reason for doing this is something that I learned about from Ed Fredkin, and my entire interest in the subject has been inspired by him. It

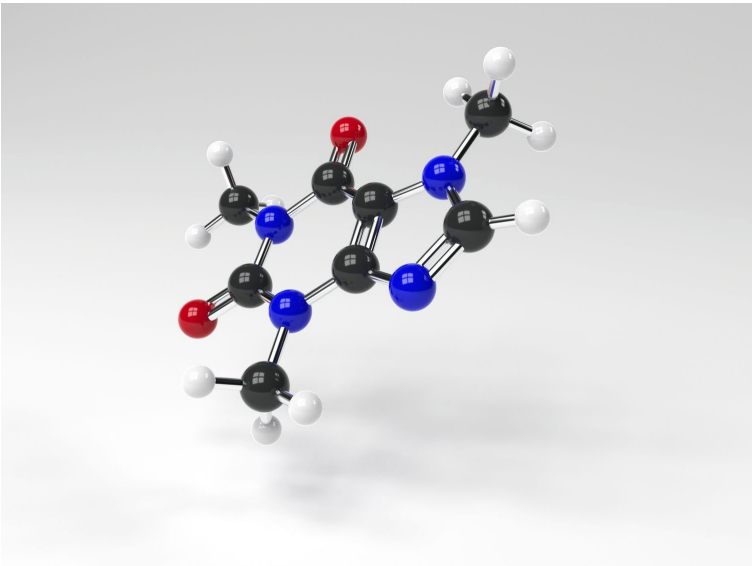
4. QUANTUM COMPUTERS—UNIVERSAL QUANTUM SIMULATORS

The first branch, one you might call a side-remark, is, Can you do it with a new kind of computer—a quantum computer? (I'll come back to the other branch in a moment.) Now it turns out, as far as I can tell, that you can simulate this with a quantum system, with quantum computer elements. It's not a Turing machine, but a machine of a different kind. If we disregard the continuity of space and make it discrete, and so on, as an approximation (the same way as we allowed ourselves in the classical case), it does seem to

Quantum Computing - Beginnings

Nature routinely computes on an extremely large scale using quantum mechanical principles – why can't we?

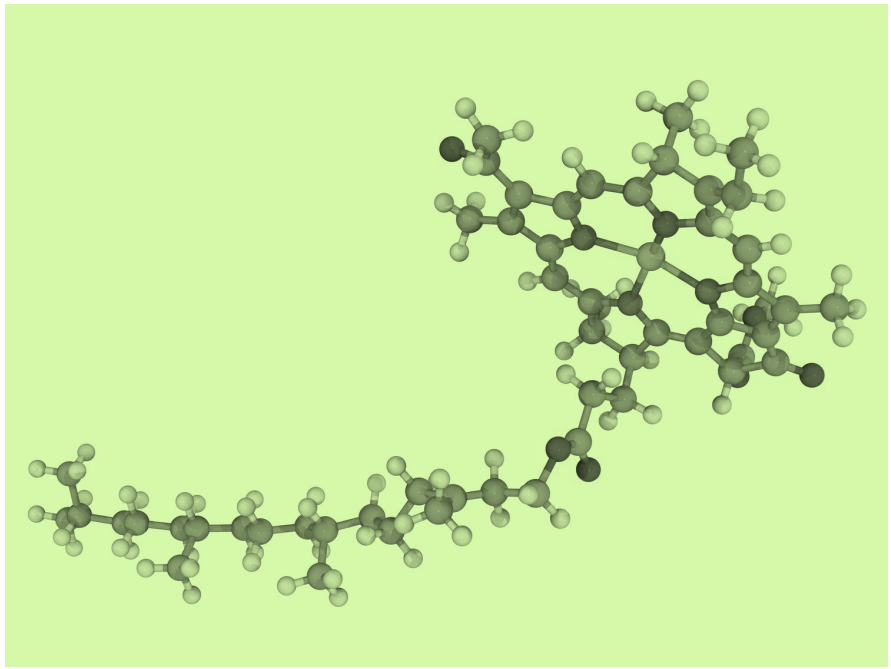




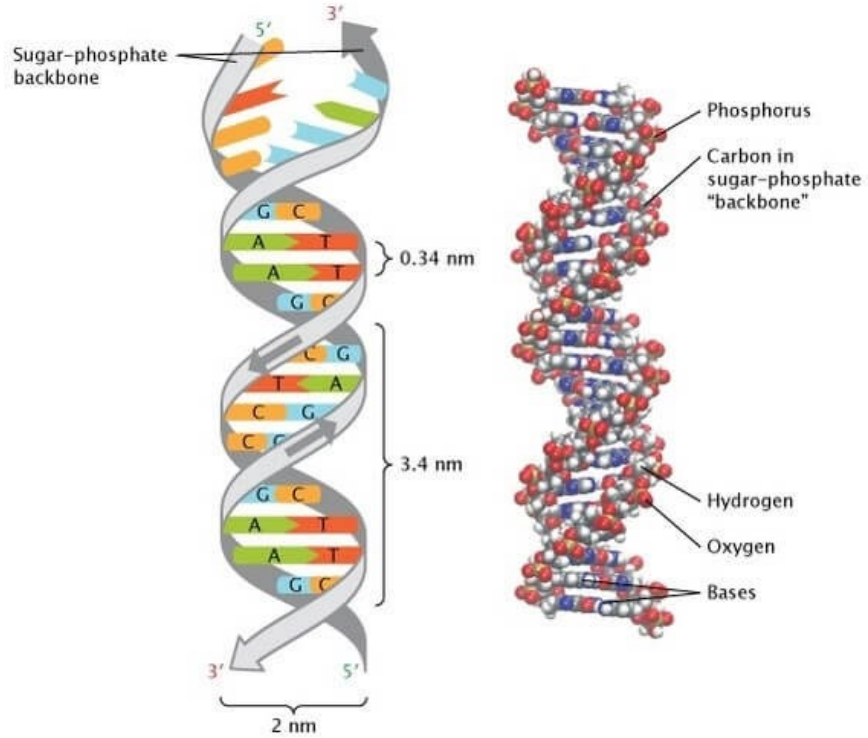
Number of possible configurations of caffeine molecule: $2^{242} \sim 10^{73}$
Number of bits on conventional computers to simulate caffeine: 10^{48}
Number of stars in the universe: $\sim 10^{24}$
Number of grains on sand on all the beaches on Earth: $7.5 \times 10^{18 - 21}$

Quantum Computing - Beginnings

Nature routinely computes on a much large scale using quantum mechanical principles – why can't we?



Chlorophyll molecule



DNA

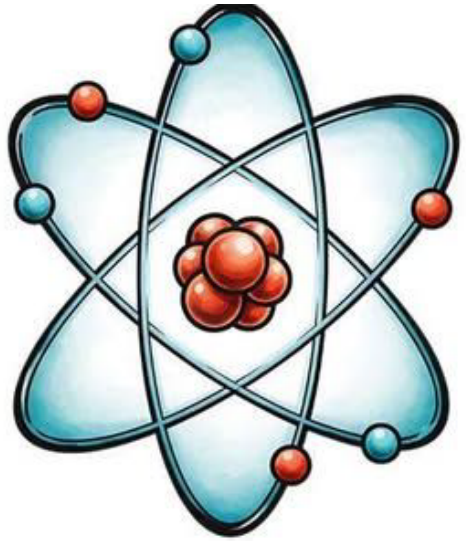
Quantum Computing - Beginnings

Key quantum principles – can we use them in computation?

1. Everything is Energy
At the deepest level, all matter is made of vibrating energy.
Solid objects are not truly solid — they're fields of movement.

2. Wave-Particle Duality
Particles behave as both waves and particles.
Reality exists as possibility until it's observed.

3. The Observer Effect
The act of observation changes the outcome.
Conscious attention influences physical reality.

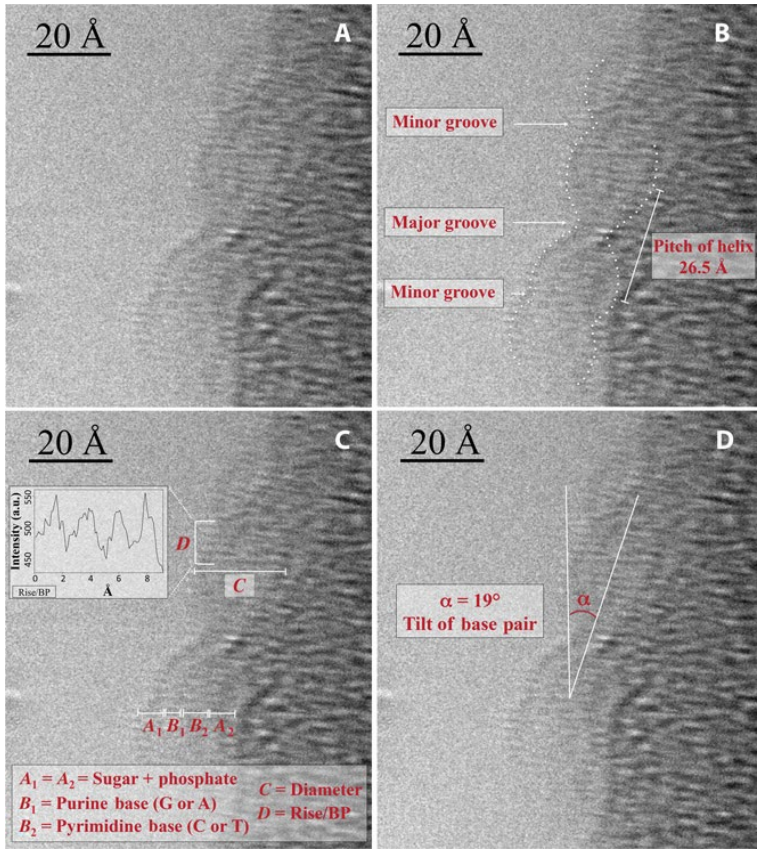


4. Superposition
A particle can exist in multiple states at once.
Reality holds many potential versions simultaneously.

5. Quantum Entanglement
Particles can remain connected across vast distances.
A change in one instantly affects the other.

6. Uncertainty Principle
You can't know everything about a system at the same time.
The universe has built-in limits to certainty and control.

7. Probability, Not Certainty
Events don't happen deterministically — only probabilistically.
Reality is a field of likelihoods, not fixed outcomes.



Scanning Electron Microscope (SEM) of DNA

Quantum Computing – Deutsch & Josza

Prof. David Elieser Deutsch and Prof. Richard Jozsa formulated the first concrete principles underlying modern quantum computers and created the very first quantum algorithm

Quantum theory, the Church-Turing principle and the universal quantum computer

DAVID DEUTSCH*

Appeared in *Proceedings of the Royal Society of London A* **400**, pp. 97-117 (1985)†

(*Communicated by R. Penrose, F.R.S. — Received 13 July 1984*)

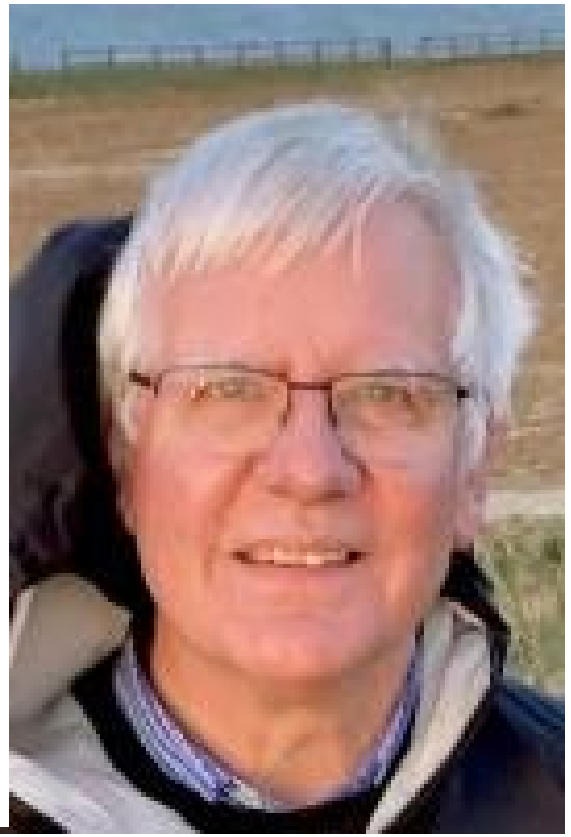
Abstract

It is argued that underlying the Church-Turing hypothesis there is an implicit physical assertion. Here, this assertion is presented explicitly as a physical principle: 'every finitely realizable physical system can be perfectly simulated by a universal model computing machine operating by finite means'. Classical physics and the universal Turing machine, because the former is continuous and the latter discrete, do not obey the principle, at least in the strong form above. A class of model computing machines that is the quantum generalization of the class of Turing machines is described, and it is shown that quantum theory and the 'universal quantum computer' are compatible with the principle. Computing machines resembling the universal quantum computer could, in principle, be built and would have many remarkable properties not reproducible by any Turing machine. These do not include the computation of non-recursive functions, but they do include 'quantum parallelism', a method by which certain probabilistic tasks can be performed faster by a universal quantum computer than by any classical restriction of it. The intuitive explanation of these properties places an intolerable strain on all interpretations of quantum theory other than Everett's. Some of the numerous connections between the quantum theory of computation and the rest of physics are explored. Quantum complexity theory allows a physically more reasonable definition of the 'complexity' or 'knowledge' in a physical system than does classical complexity theory.

*Current address: Centre for Quantum Computation, Clarendon Laboratory, Department of Physics, Parks Road, OX1 3PU Oxford, United Kingdom. Email: david.deutsch@qubit.org

†This version (Summer 1999) was edited and converted to LaTeX by Wim van Dam at the Centre for Quantum Computation. Email: wimvdam@qubit.org

Quantum theory of computation and construction of the universal quantum computer (1984)



PROCEEDINGS OF THE ROYAL SOCIETY A MATHEMATICAL, PHYSICAL & ENGINEERING SCIENCES

Rapid Solution of Problems by Quantum Computation

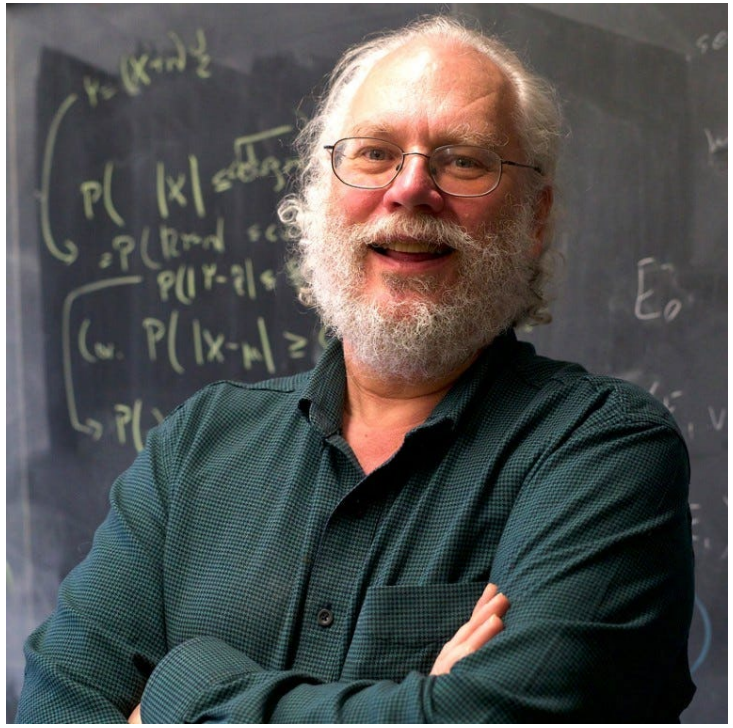
David Deutsch and Richard Jozsa

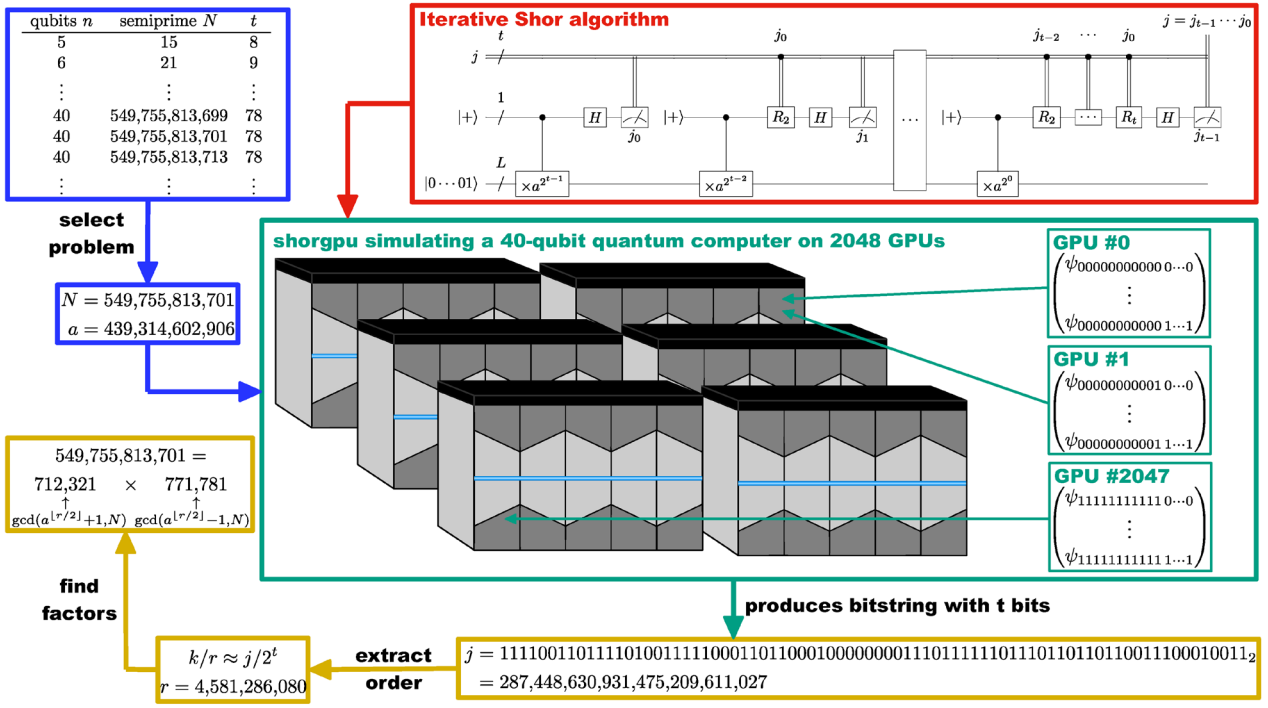
Proc. R. Soc. Lond. A 1992 **439**, 553-558
doi: 10.1098/rspa.1992.0167

Deutsch-Josza Quantum Algorithm (1992)

Quantum Computing – Shor's Algorithm

Prof. Peter Shor invented "Shor's Algorithm" a very fast quantum computing algorithm in 1994 to factorize any number into primes with potential to break modern encryption methods (RSA -128/256 bit encryption)

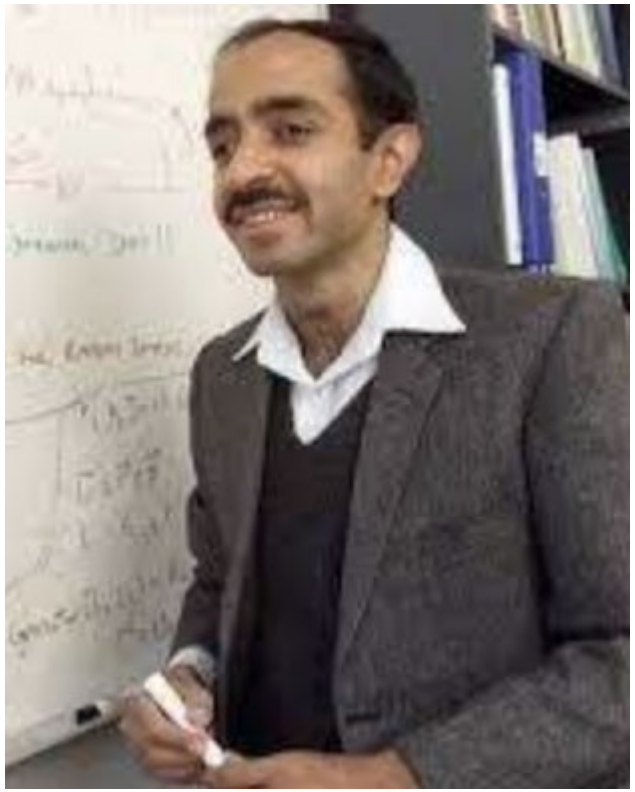


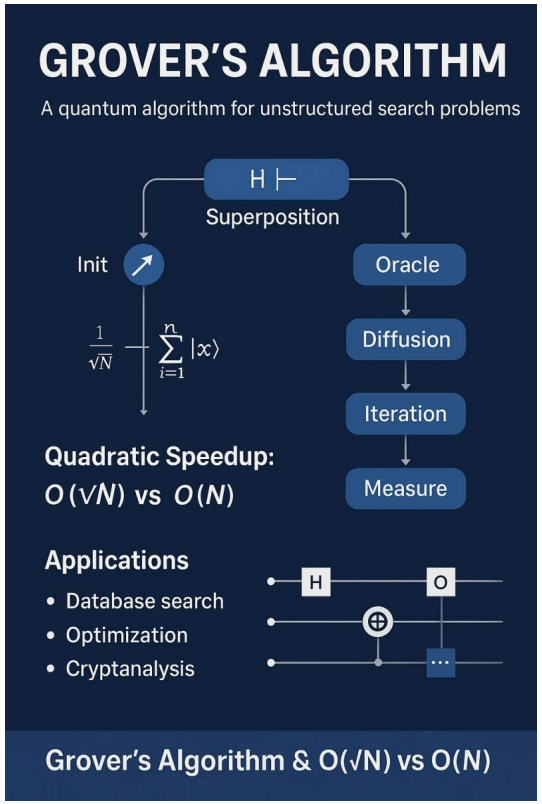


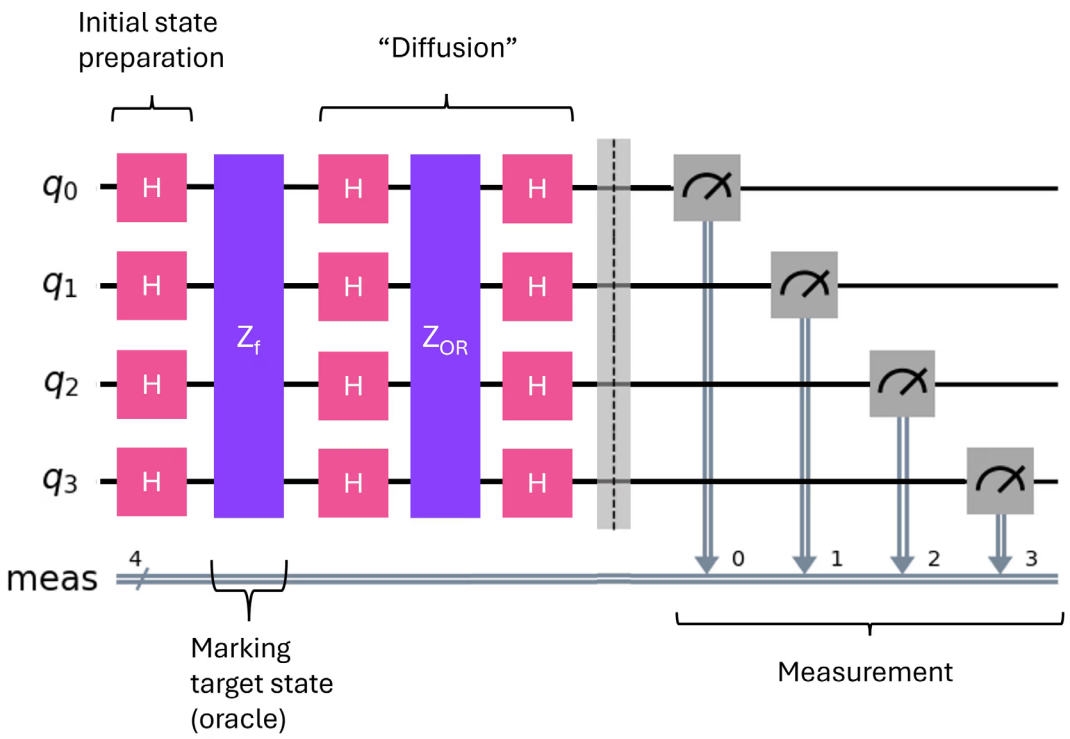
Largest number factorized 39-bit number: 549,755,813,701
(Simulating Shor's algorithm running on a quantum computer on a supercomputer)

Quantum Computing – Grover's Algorithm

Dr. Lov Kumar Grover proposed Grover's Algorithm, which can search a database faster than a conventional search (1996), also useful in breaking encryption methods







Grover's Algorithm

Quantum Computing: First Quantum Computers

1998: First experimental quantum computer: 2-qubit nuclear magnetic resonance (NMR) quantum computer running Deutsch-Josza algorithm

2011: First commercial quantum computer: D-Wave One (sold to Lockheed Martin)

Implementation of a Quantum Search Algorithm on a Nuclear Magnetic Resonance Quantum Computer

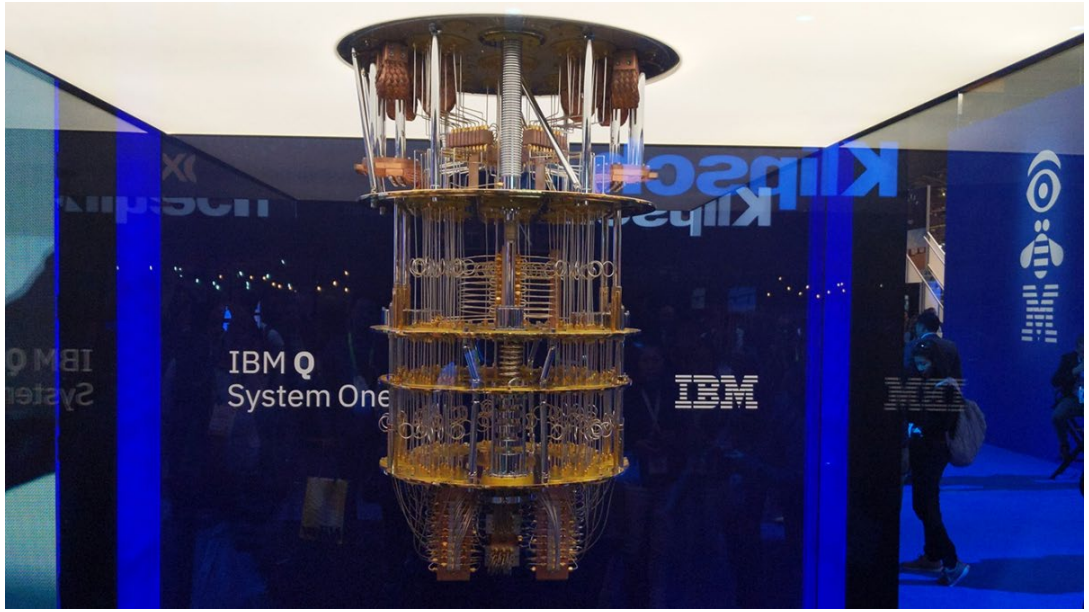
Jonathan A. Jones*†, Michele Mosca†‡, Rasmus H. Hansen†
* Oxford Centre for Molecular Sciences, New Chemistry Laboratory, South Parks Road, Oxford, OX1 3QT, UK,
† Centre for Quantum Computation, Clarendon Laboratory, Parks Road, Oxford OX1 3PU, UK
‡ Mathematical Institute, 24–29 St Giles', Oxford, OX1 3LB, UK

The simulation of quantum mechanical systems with classical computers appears to be a computationally intractable problem. In 1982 Feynman[1] reversed this observation, suggesting that quantum mechanical systems have an information processing capability much greater than that of corresponding classical systems, and thus could be used to implement a new type of powerful computer. In 1985 Deutsch[2] described a quantum mechanical Turing machine, showing that quantum computers could indeed be constructed. Since then there has been extensive research in this field, but while the theory is fairly well understood actually building a quantum computer has proved extremely difficult, and only two methods have been used to demonstrate quantum logic gates: ion traps,[3,4] and nuclear magnetic resonance (NMR).[5,6] NMR quantum computers have previously been used to demonstrate quantum algorithms to solve the two bit Deutsch problem.[7,8] Here we show how such a computer can be used to implement a fast quantum search algorithm initially developed by Grover.[9,10]

Among other applications Grover's algorithms enable an extremely rapid search over the domain of a binary function to find elements for which this function is satisfied (that is, the function has the value 1). This approach is simpler if the number of satisfying values is known beforehand, and is particularly simple when precisely one quarter of the elements in the domain satisfy the function.[11] The algorithm can be demonstrated using a computer with two quantum bits (qubits) to search a two bit domain in which one of the four elements satisfies the function. A classical search of this domain would require between 1 and 3

IBM System One - 2019

In 2019, IBM released IBM System One, the world's first integrated, circuit-based commercial quantum computer designed for use outside of research labs
In parallel IBM released “Qiskit” (*Quantum Information Science Kit*), a Python based SDK to develop applications
Made available IBM System Ones to any developers on the IBM Quantum Platform



The Quantum Computing Explosion

INTERNATIONAL YEAR OF Quantum Science and Technology

The quantum ecosystem expanded rapidly from 2020 onwards

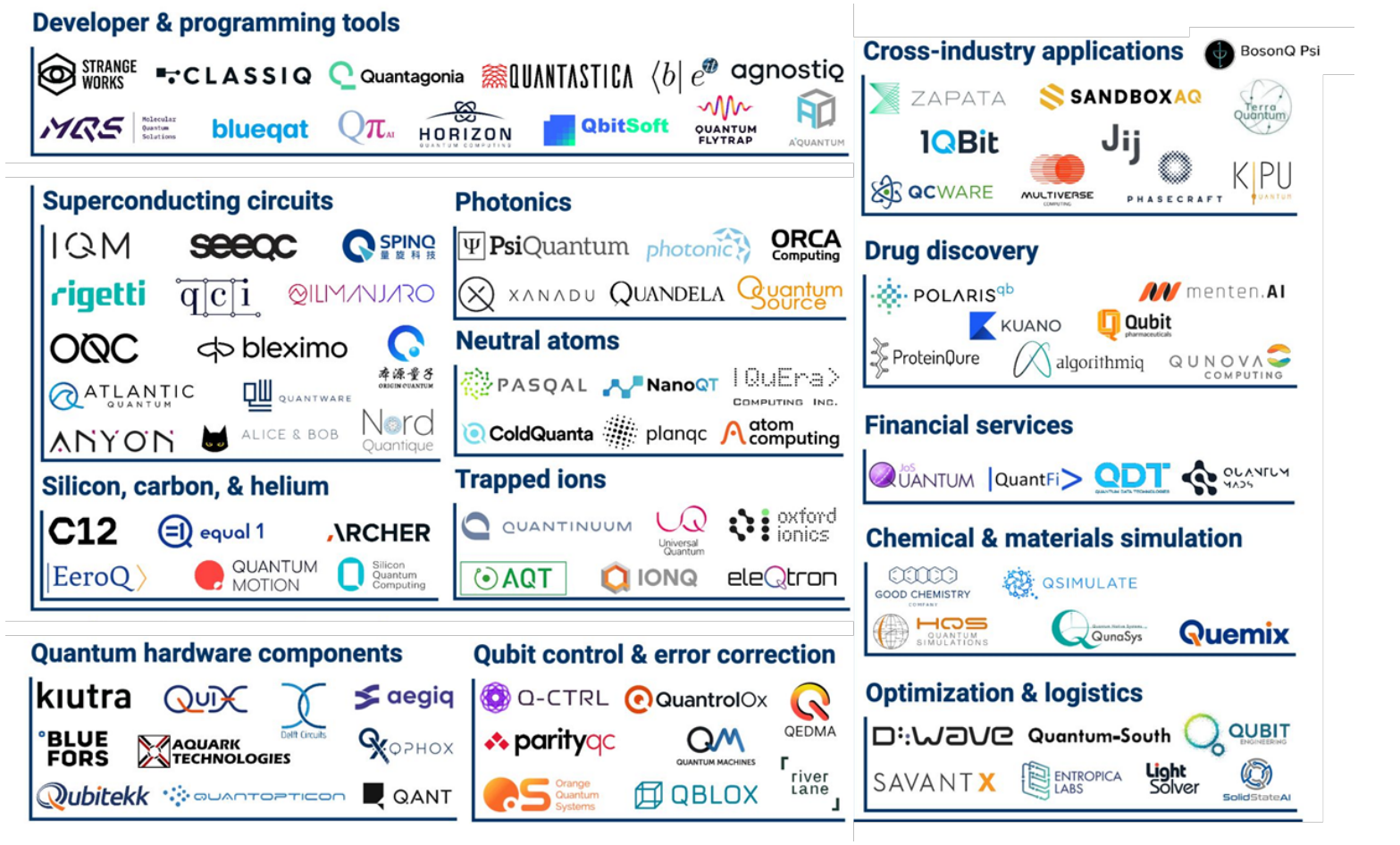
https://quantum2025.org

User-level

Applications/Algorithms

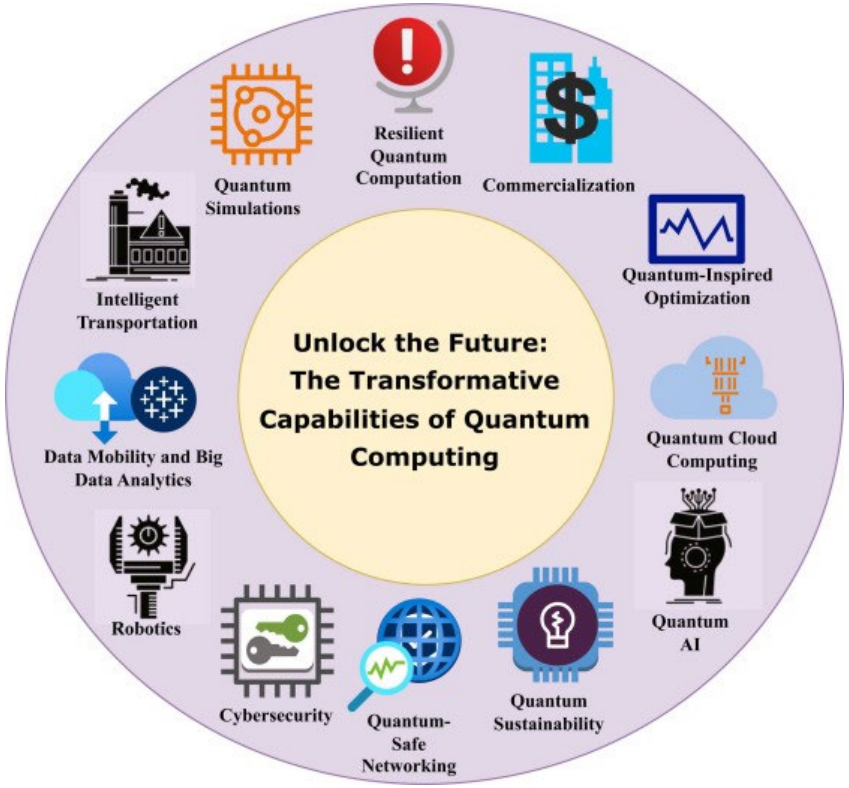
Middleware

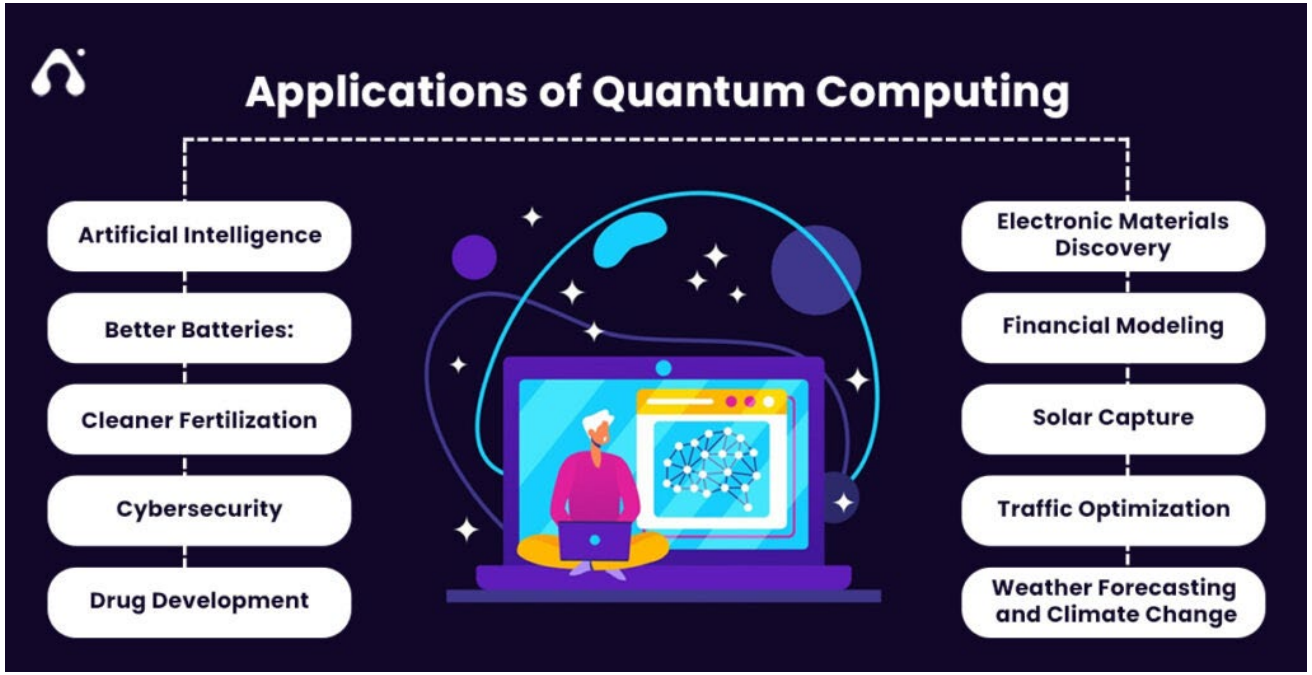
Hardware



Quantum System Applications

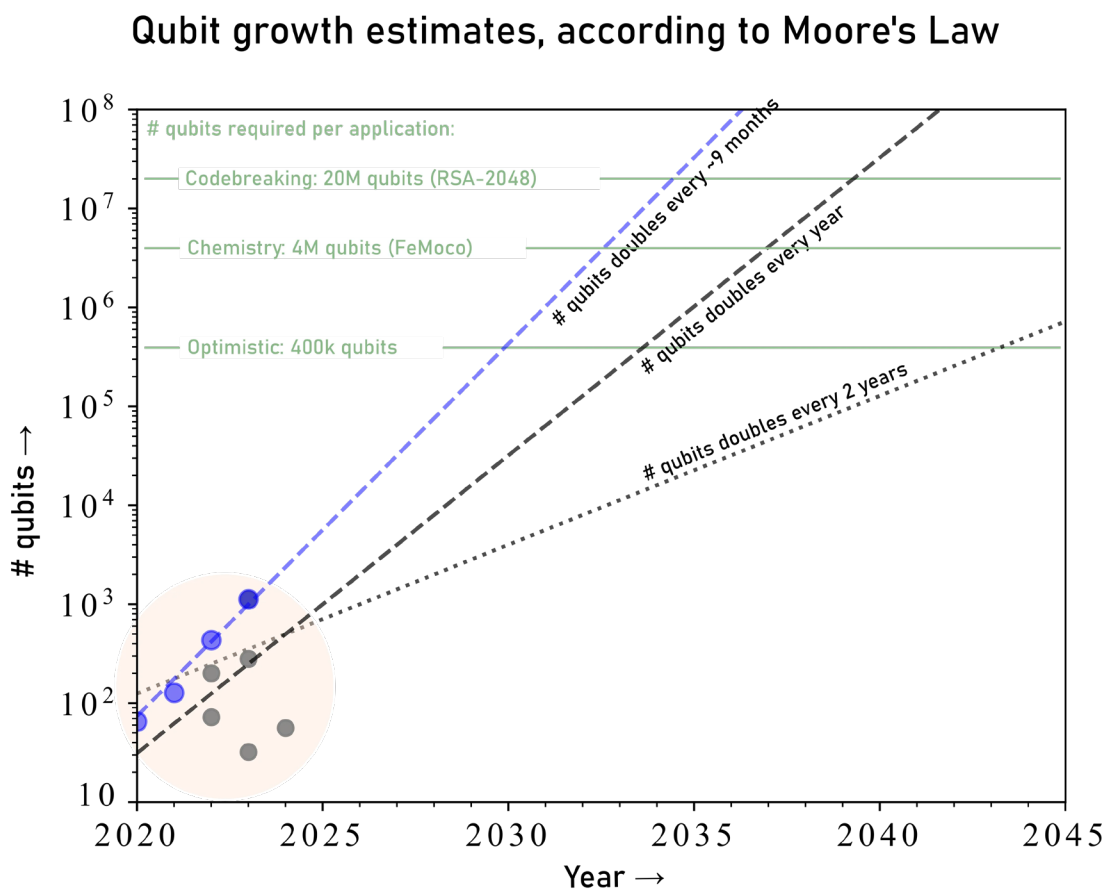
Applications of quantum computing (Native and inspired) are manifold and already in progress
Transportation and scheduling: airline scheduling, cargo loading, ports
Cyber security: encryption, de-encryption, communication

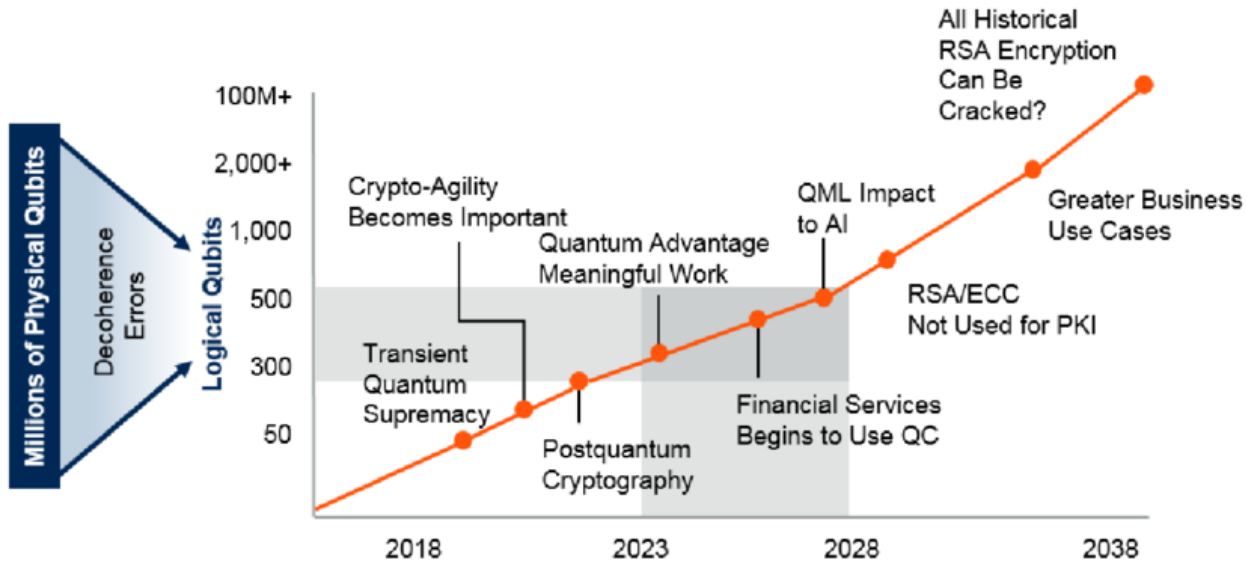




Quantum Computing Future Projection

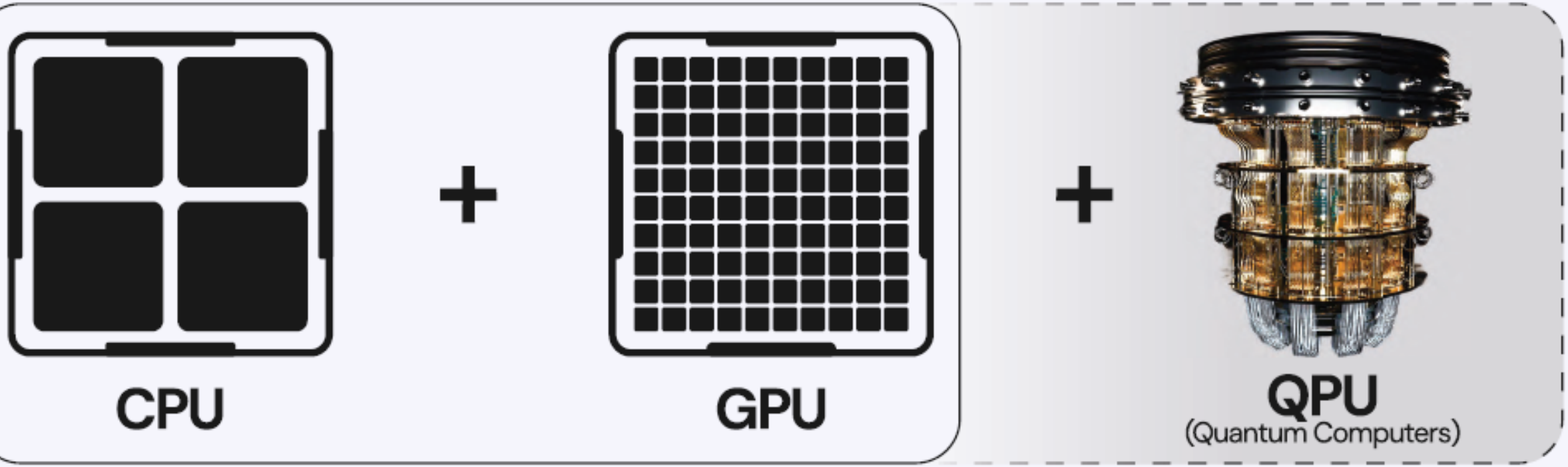
Quantum computers are on the path to mature and become more powerful than any existing computer to unlock new applications





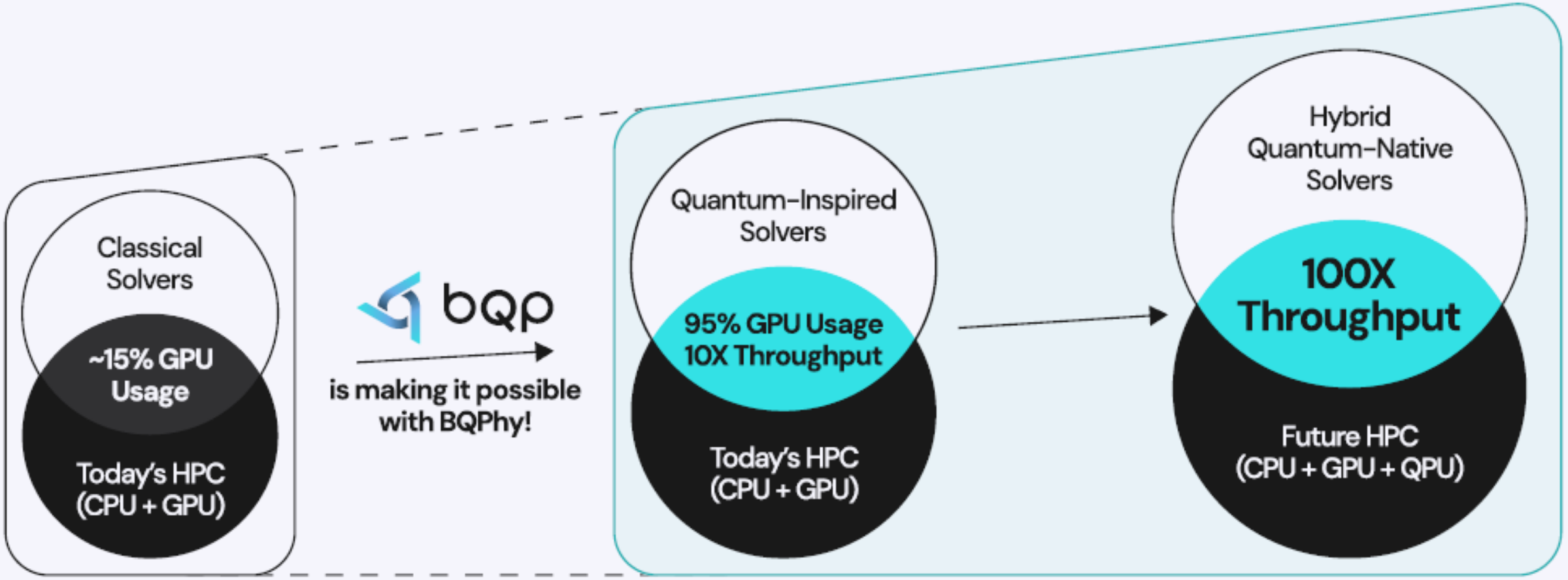
BQP (Boson Quantum Psi)

We're still using legacy CPU math to run on GPUs. **Today, 85% of GPU compute is going to waste.**

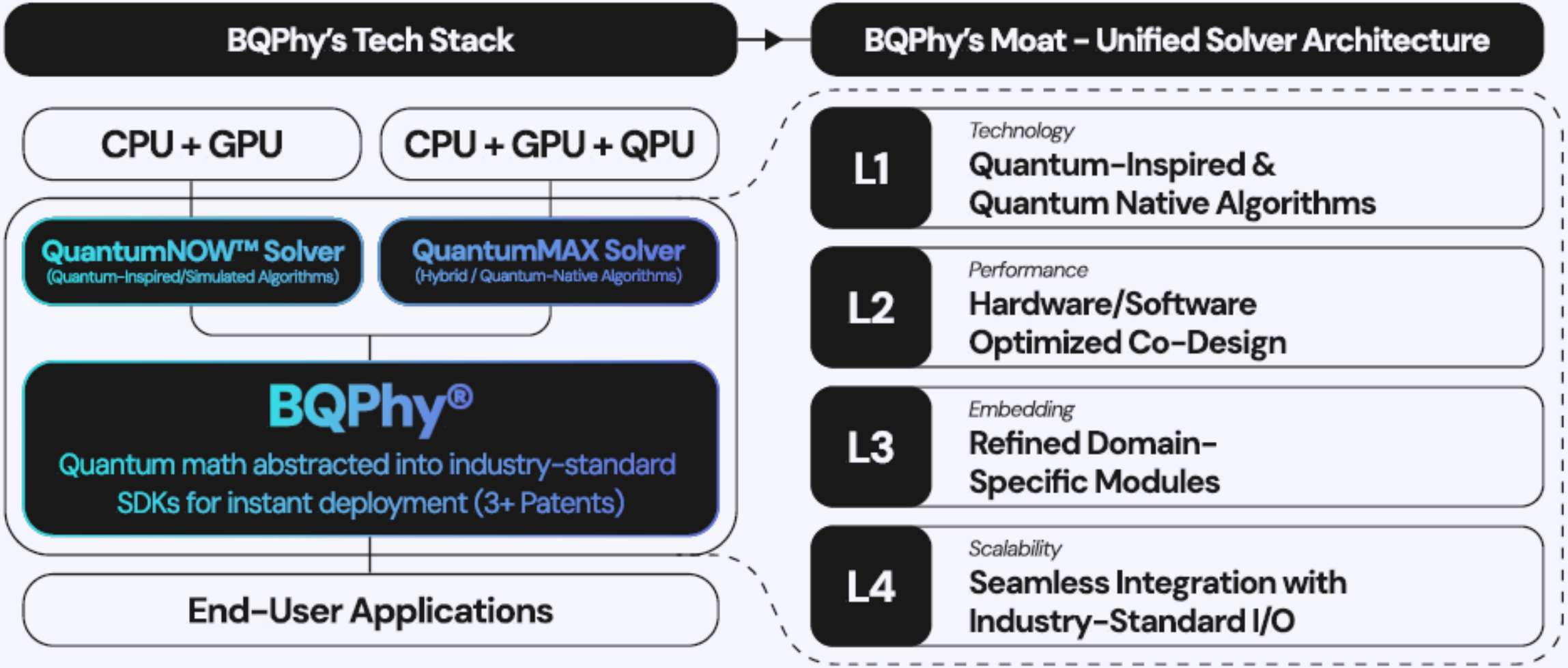


Today's HPC Data Centers → No Bridge to Quantum → Future HPC Data Centers

Refactoring Legacy Math with Quantum-Inspired Solvers: **10X ROI Today. Quantum-Native Utility Tomorrow.**



BQPhy: The Unified Platform for Digital Engineering & Physics AI



BQPhy Validated

(QI Solver for Physics AI)

Challenge

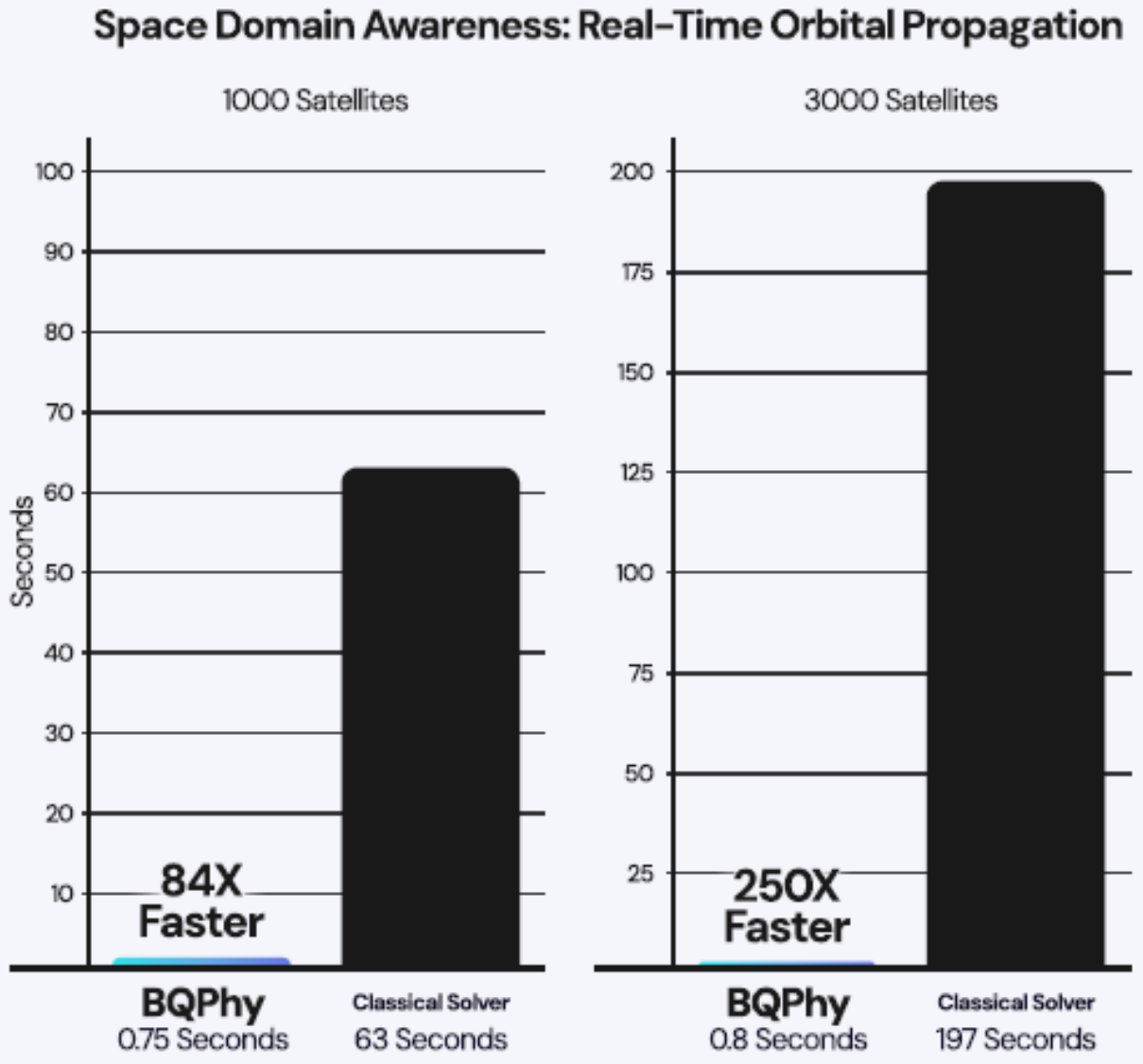
- Astro Engineers have to choose between fast but inaccurate (SGP4) and accurate but slow (Orekit) legacy solvers.

Outcome

- BQPhy achieved a **250x speedup** over legacy solvers, processing the entire 3000+ LEO satellite catalog in **<1 sec** on 1 GPU.
- Quantum-Inspired algorithm compressed the Physics AI model from 14M parameters to 2K – leading to deployment on NVIDIA Jetson Nano.

Operational Deployment

- Deployed in 3 commercial companies and a government lab for test & validation of edge computing for autonomous mission.



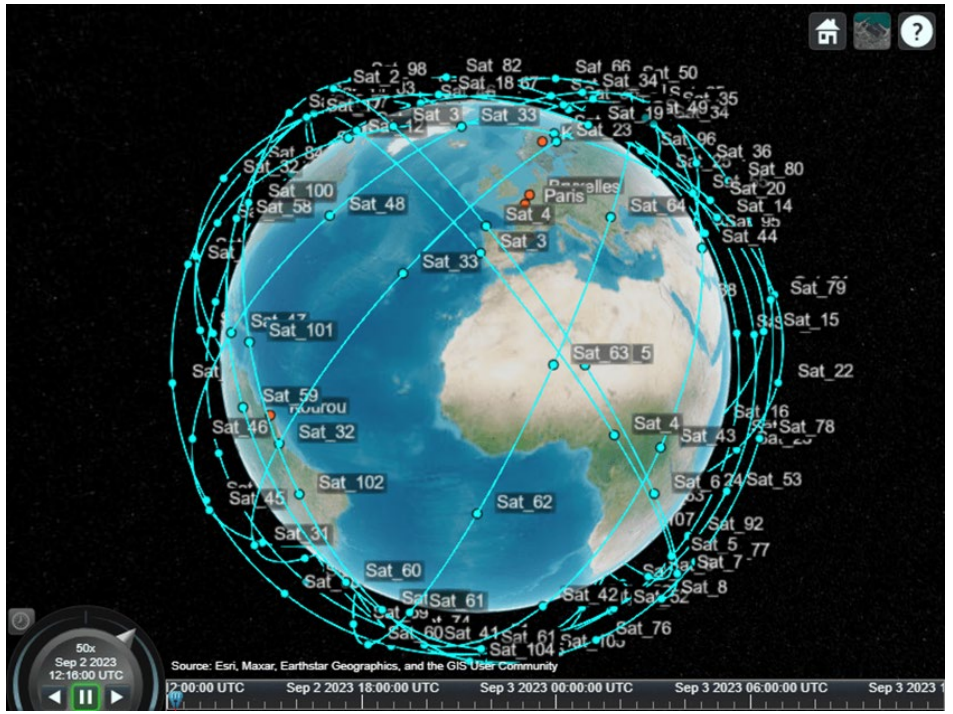
Application: Satellite Constellation Design

Poor Constellation, Mission Goals Compromised

- Poor or unsolvable satellite constellation design leads to satellite servicing gaps,, inefficient resource use, and mission failure — impacting communications, Earth observation, and global navigation systems
- **Quantum-inspired Optimization** enables constellation designs that **delivers faster response times with fewer satellites** → cutting deployment costs while enhancing performance efficiency.

Design Parameters

- Objective – **Minimize Total Satellites**
- Design Variable – **Orbit Altitude, Planes, Satellite/Plane, Inclination**
- Constraints – **Minimum Observability**



Smarter Design, Global Coverage Achieved

- **BQPhy requires 10% FEWER satellites than the leading solution** while maintaining high observability → minimizing costs and ensuring mission success

Application – Material Design Optimization

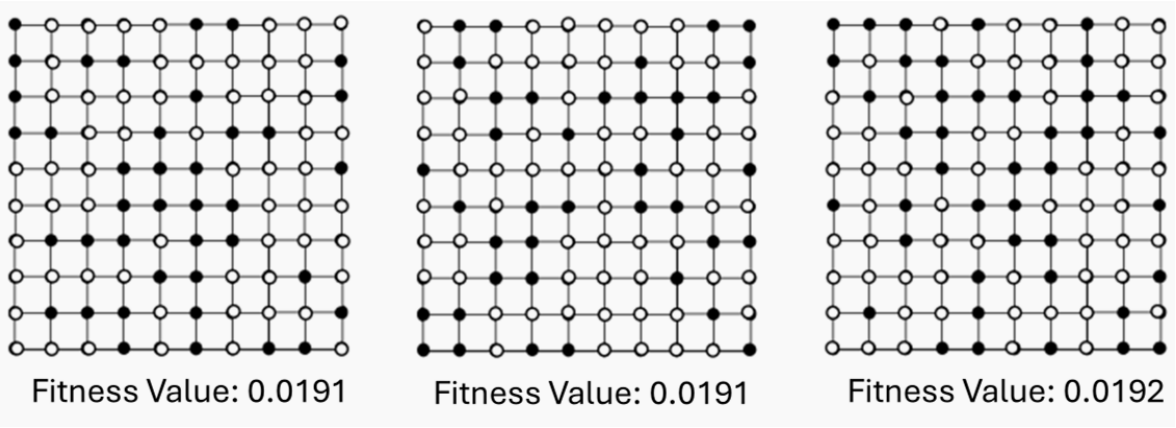
Complex Molecular Designs. Suboptimal formations.

- Complex molecular structures in magnetic materials often result in suboptimal configurations
- Inefficient designs lead to higher material costs, as well as longer production cycles

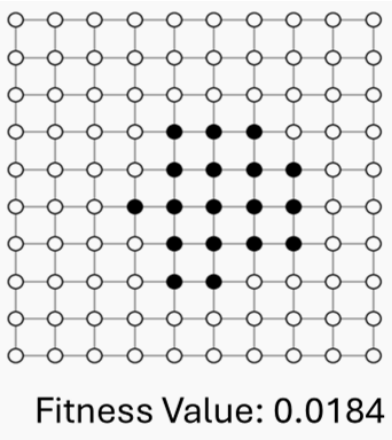
Optimized Materials. Accelerate Innovation.

- BQPhy delivered superior material configurations **12X faster** than conventional methods
- **4% Reduction** in free energy → increased stability for improved performance & reliability
- Reduction of material consumption and extended lifecycle → lower operation costs

Classical Optimization



Quantum-Inspired Optimization



Next step:
Scale to 100x100 1000 x 1000 atom configurations (cannot be solved with GA)

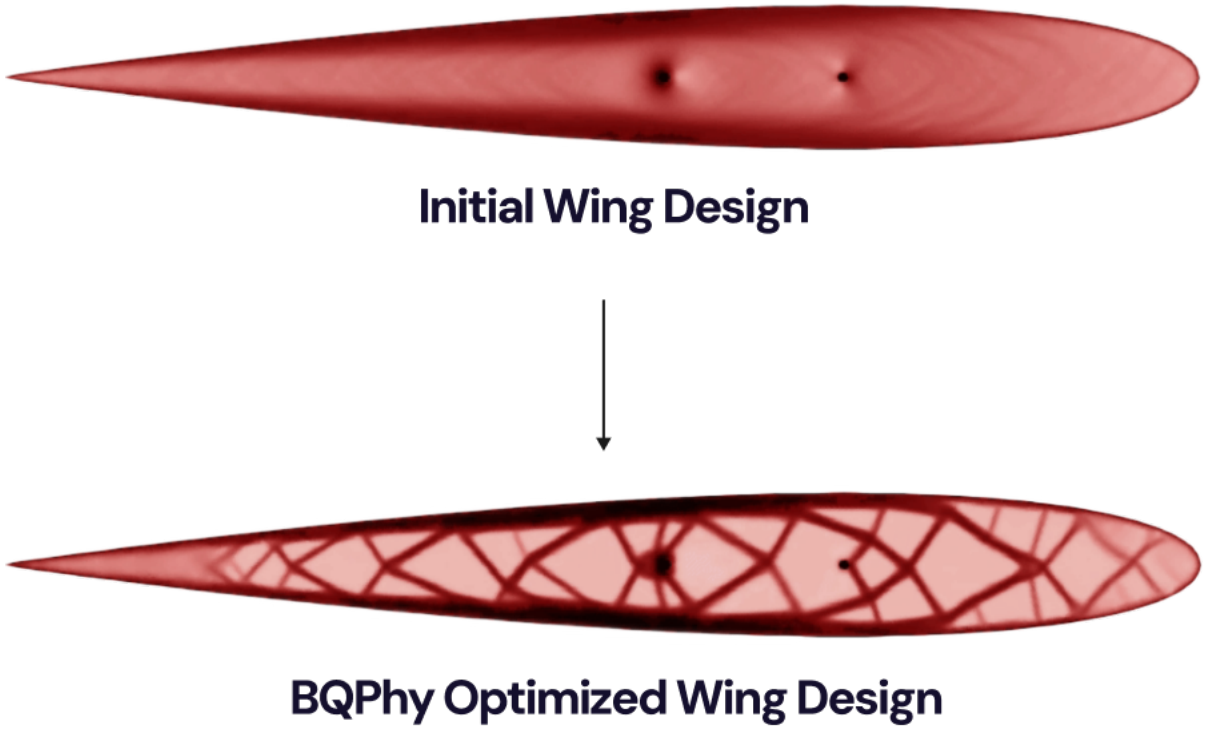
Application – Design Optimization

Wasteful Product Cycles. Slow Delivery.

- Airbus A380 took 2X time and 5X the original budget
- Companies are only capable of testing 20% of the total design envelope

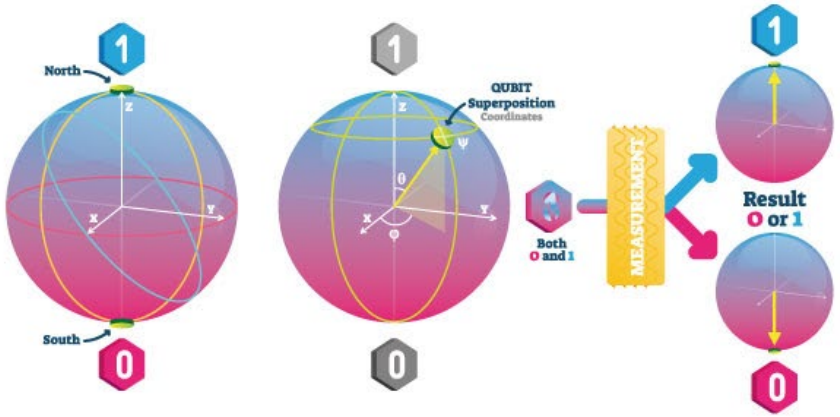
Cut Years. Save Billions.

- Compared to existing solutions, BQPhy achieved 6% lighter design in **1/10 th** of the time
- **$40M fuel cost saving** for 13K trans-Atlantic flight per year
- Reduced wing development time by **1-2 years** (from 6-7) & **$1.5-2B in cost** (from $8-10B)



Quantum Computing Fundamentals (Some Mathematics Required)

Qubits

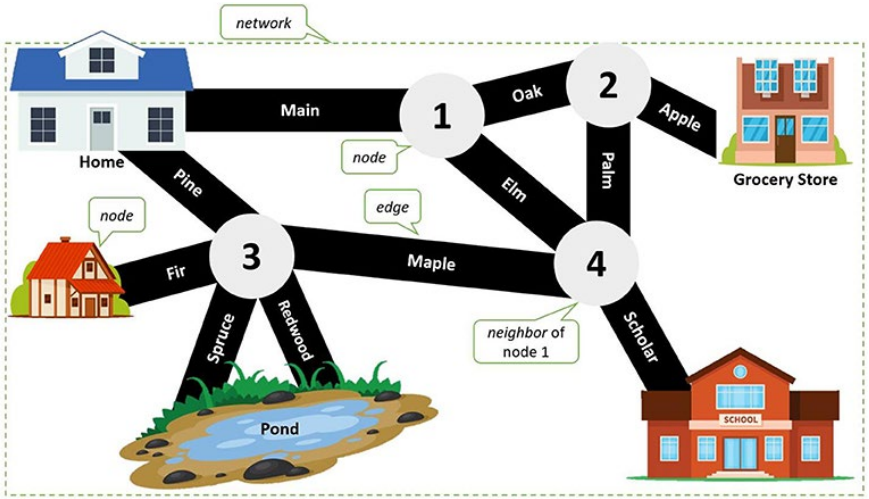


Quantum Superposition

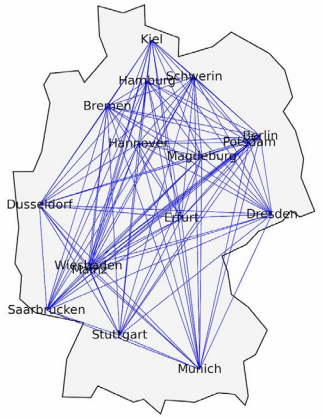
Qubits exist in multiple states → can store more information

Quantum Entanglement

Qubits can be interconnected even if physically distant → information grows exponentially when scaling in a quantum system:
N bits in classical can be represented by 2^N qubits in quantum

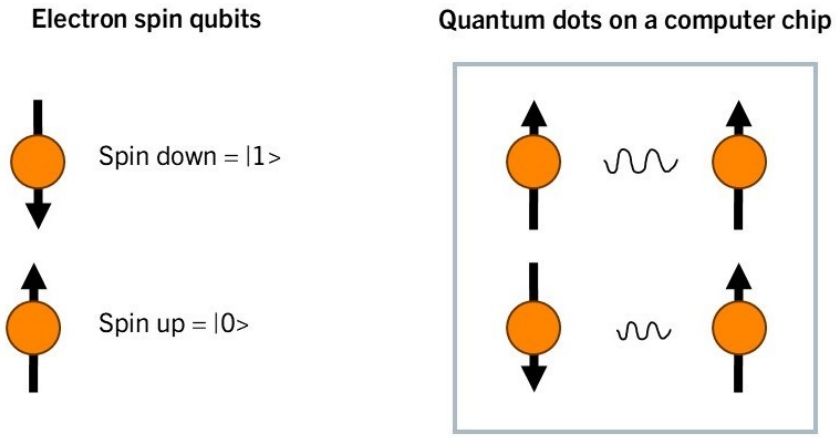


All possible paths between any two state capitals in Germany



Bra, Ket and Tensors

"Bra" and "Ket" Representation of a qubit



Bra: $\langle 0| = [1 \quad 0], \langle 1| = [0 \quad 1]$

Ket: $|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$

Unknown spin: $|\psi\rangle = \alpha\,|0\rangle + \beta|1\rangle = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}$

Combining qubits using tensor operations

$$|10\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\ 1 \begin{bmatrix} 1 \\ 0 \end{bmatrix} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

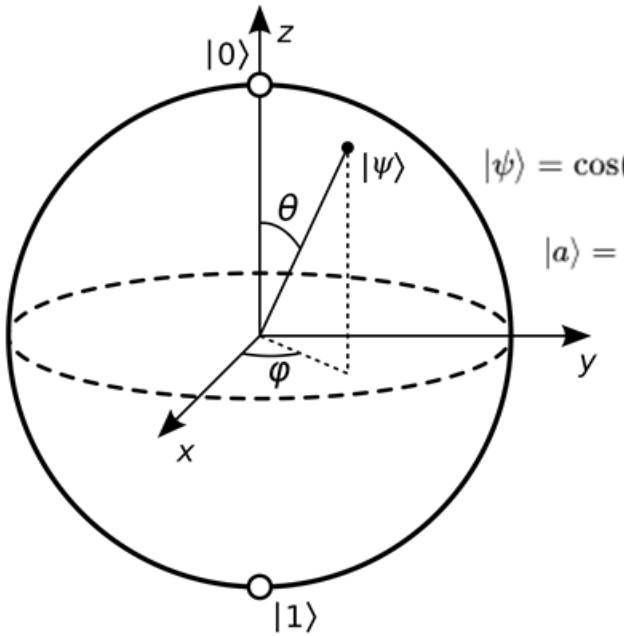
$$|01\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \otimes \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\ 0 \begin{bmatrix} 0 \\ 1 \end{bmatrix} \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

$$|11\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\ 1 \begin{bmatrix} 0 \\ 1 \end{bmatrix} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

$$|00\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\ 0 \begin{bmatrix} 1 \\ 0 \end{bmatrix} \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

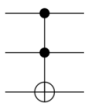
Quantum Gates

Quantum Gates operate on one or more Qubits to change their state
Qubits represented on a “Bloch sphere”, or as vectors (“kets”)
Represented by matrix operations, vector and tensor products

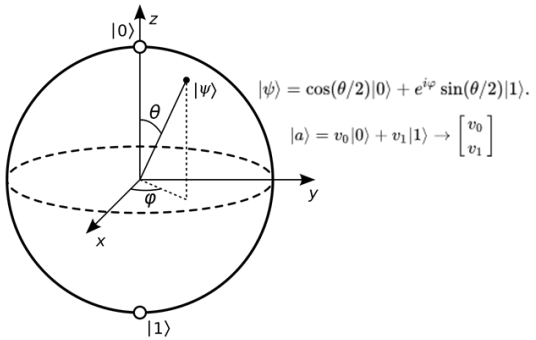


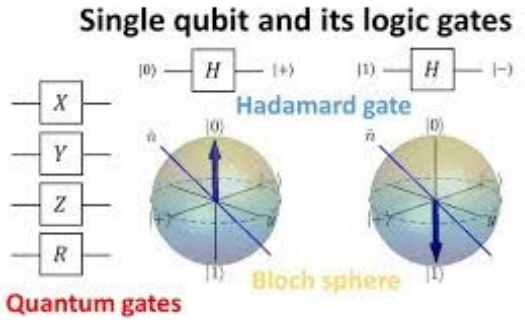
$$|\psi\rangle = \cos(\theta/2)|0\rangle + e^{i\varphi}\sin(\theta/2)|1\rangle.$$

$$|a\rangle = v_0|0\rangle + v_1|1\rangle \rightarrow \begin{bmatrix} v_0 \\ v_1 \end{bmatrix}$$

Operator	Gate(s)		Matrix
Pauli-X (X)			$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$
Pauli-Y (Y)			$\begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$
Pauli-Z (Z)			$\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$
Hadamard (H)			$\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$
Phase (S, P)			$\begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix}$
$\pi/8$ (T)			$\begin{bmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{bmatrix}$
Controlled Not (CNOT, CX)			$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$
Controlled Z (CZ)			$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}$
SWAP			$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
Toffoli (CCNOT, CCX, TOFF)			$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$

Single and Two Qubit Gates





$$Pauli_x: X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}\begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} v_2 \\ v_1 \end{bmatrix}$$

$$Pauli_y: Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}\begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} -iv_2 \\ +iv_1 \end{bmatrix}$$

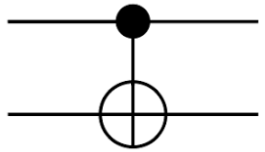
$$Pauli_z: Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}\begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} v_1 \\ -v_2 \end{bmatrix}$$

$$Hadamard: H|a\rangle$$

$$\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}\begin{bmatrix} v_1 \\ v_2 \end{bmatrix} == \begin{bmatrix} \frac{1}{\sqrt{2}}(v_1 + v_2) \\ \frac{1}{\sqrt{2}}(v_1 - v_2) \end{bmatrix}$$

Controlled NOT or $CNOT\ gate$

The second bit flips if the first one is 1



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

$$\text{CNOT } |10\rangle = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = |11\rangle$$

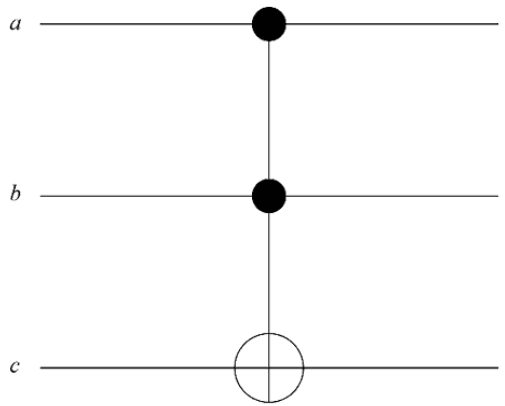
$$\text{CNOT } |01\rangle = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} = |01\rangle$$

CNOT $|11\rangle = |10\rangle$

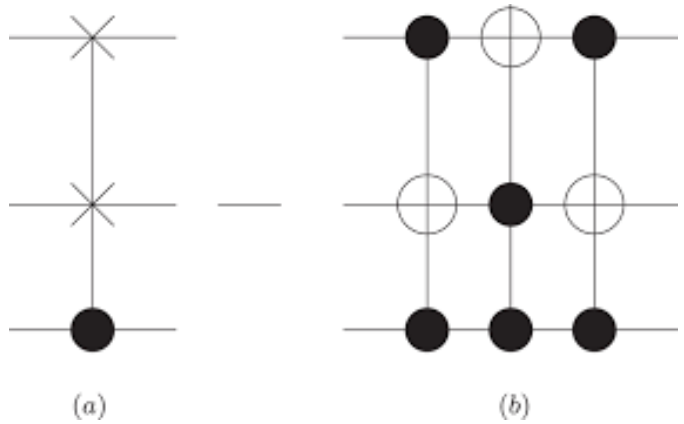
CNOT $|00\rangle = |00\rangle$

Three Qubit Gates

Toffoli Gate



Friedkin Gate



The first two qubits are the control qubits and remain unchanged, If both of them are identical, then the second qubit is modified with a NOT operation

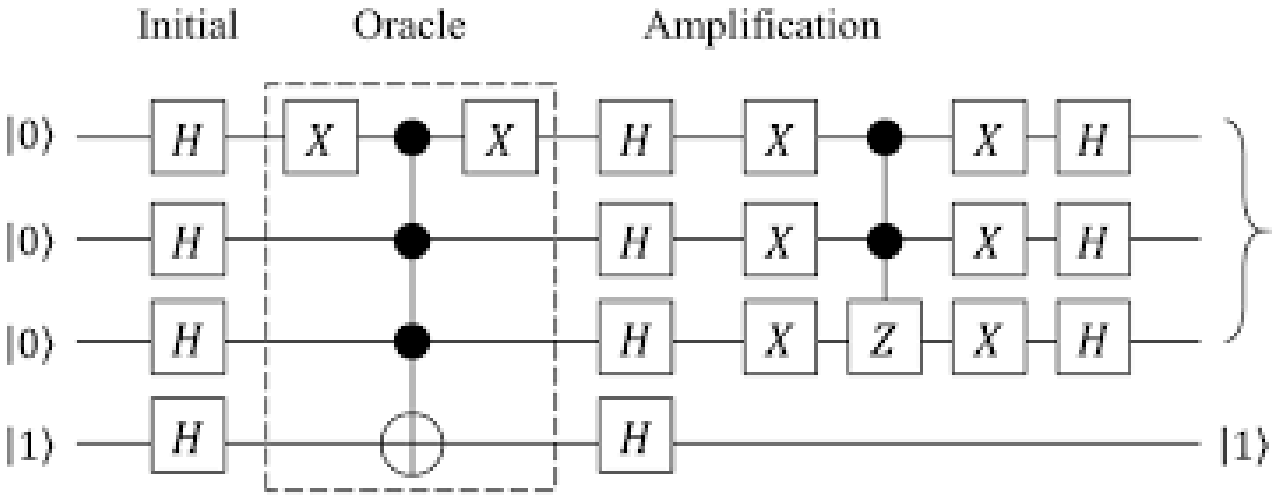
T $|110\rangle = |111\rangle$: both control qubits are |1> hence third qubit flips, T $|101\rangle = |101\rangle$: control qubits are |1> and |0>, hence the third qubit remains the same

The first qubit is the control qubit
If the first qubit is |1> then the second and third qubits are swapped

F $|110\rangle = |101\rangle$: control qubit is|1> hence second and third qubits swap
F $|001\rangle = |101\rangle$: control qubits is |0>, hence they don't swap

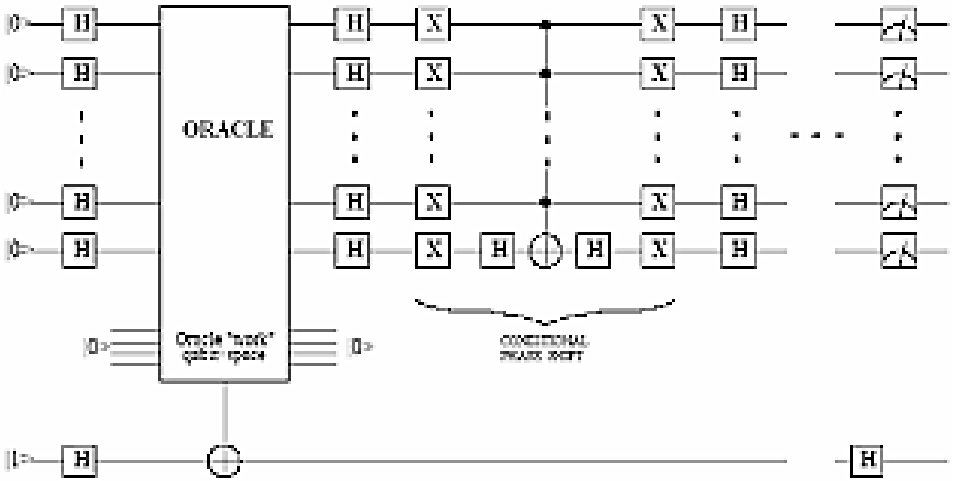
Quantum Circuits

- Sequence of gates, entanglements on multiple qubits, which encode any kind of information and measure at the end
- Designed to accomplish a specific task, hence they are also algorithms
- A new toolkit for computation and information processing
- Can be faster than or accelerate conventional algorithms for the same task



Grover's Search Algorithm: Circuit for four qubits

Fast search algorithm ($O\left(\sqrt{N}\right)$ vs

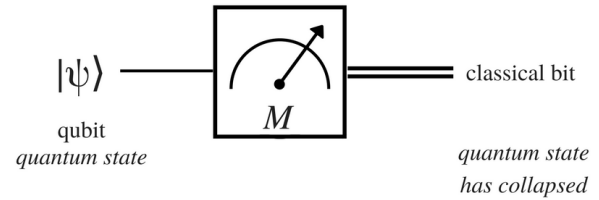


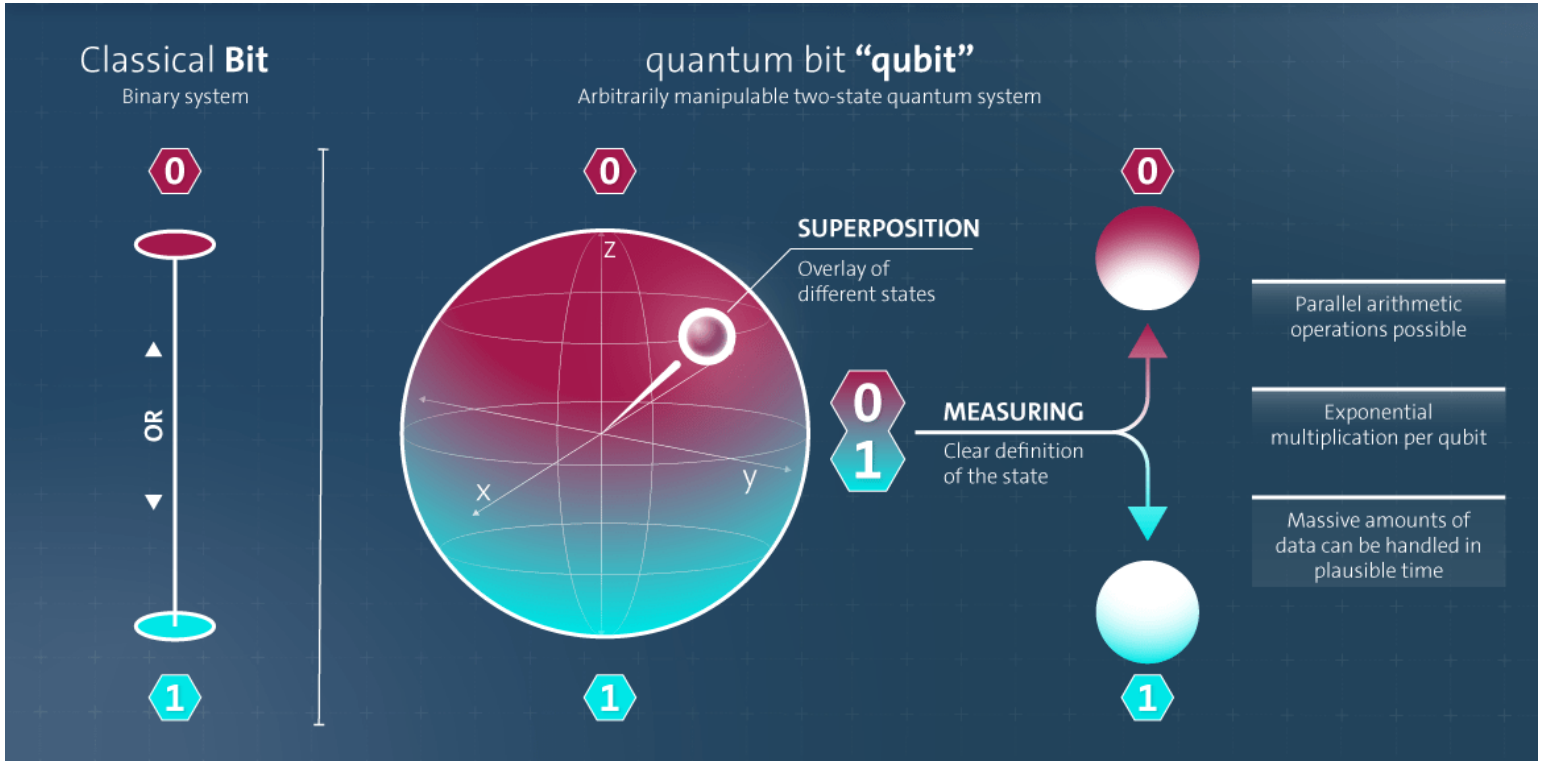
Grover's Search Algorithm for Any number of qubits

Potential to break 128-bit/256-bit encryption

Qubit Measurements

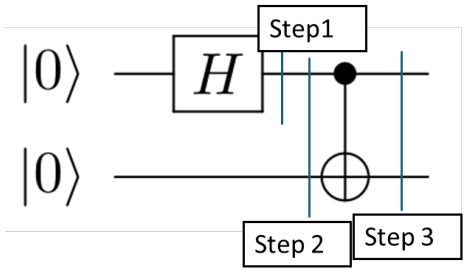
- Qubits are in superposition taking all possible values
- The measurement step makes qubits collapse to a final definite value
- Typically final step in a quantum circuit





Quantum Entanglement

Quantum Entanglement: Two qubits are connected together, and a change in one changes the other, irrespective of distance (“spooky action at a distance”)



Step 1: *Apply Hadamard Gate to first (primary) qubit*

$$H|\psi\rangle = \frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix}\begin{bmatrix}\alpha\\ \beta\end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix}\alpha+\beta\\ \alpha-\beta\end{bmatrix}$$

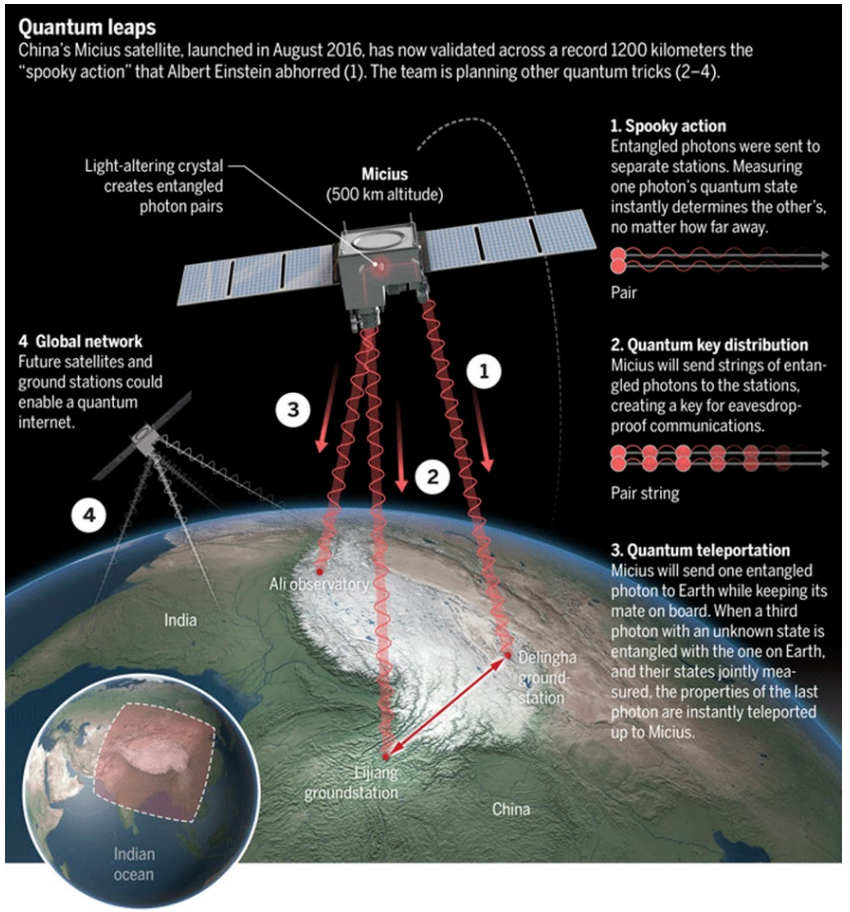
$$H|0\rangle = \frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix}\begin{bmatrix}1\\ 0\end{bmatrix} = \begin{bmatrix}\frac{1}{\sqrt{2}}\\ \frac{1}{\sqrt{2}}\end{bmatrix}$$

Step 2: *Take Tensor Product with second (control) qubit*

$$|H|0\rangle\,0\rangle = \begin{bmatrix}\frac{1}{\sqrt{2}}\\ \frac{1}{\sqrt{2}}\end{bmatrix}\otimes\begin{bmatrix}1\\ 0\end{bmatrix} = \begin{bmatrix}\frac{1}{\sqrt{2}}\begin{bmatrix}1\\ 0\end{bmatrix}\\ \frac{1}{\sqrt{2}}\begin{bmatrix}1\\ 0\end{bmatrix}\end{bmatrix} = \begin{bmatrix}\frac{1}{\sqrt{2}}\\ 0\\ \frac{1}{\sqrt{2}}\\ 0\end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\ 0\\ 0\\ 0\end{bmatrix} + \frac{1}{\sqrt{2}}\begin{bmatrix}0\\ 0\\ 1\\ 0\end{bmatrix} = \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|10\rangle$$

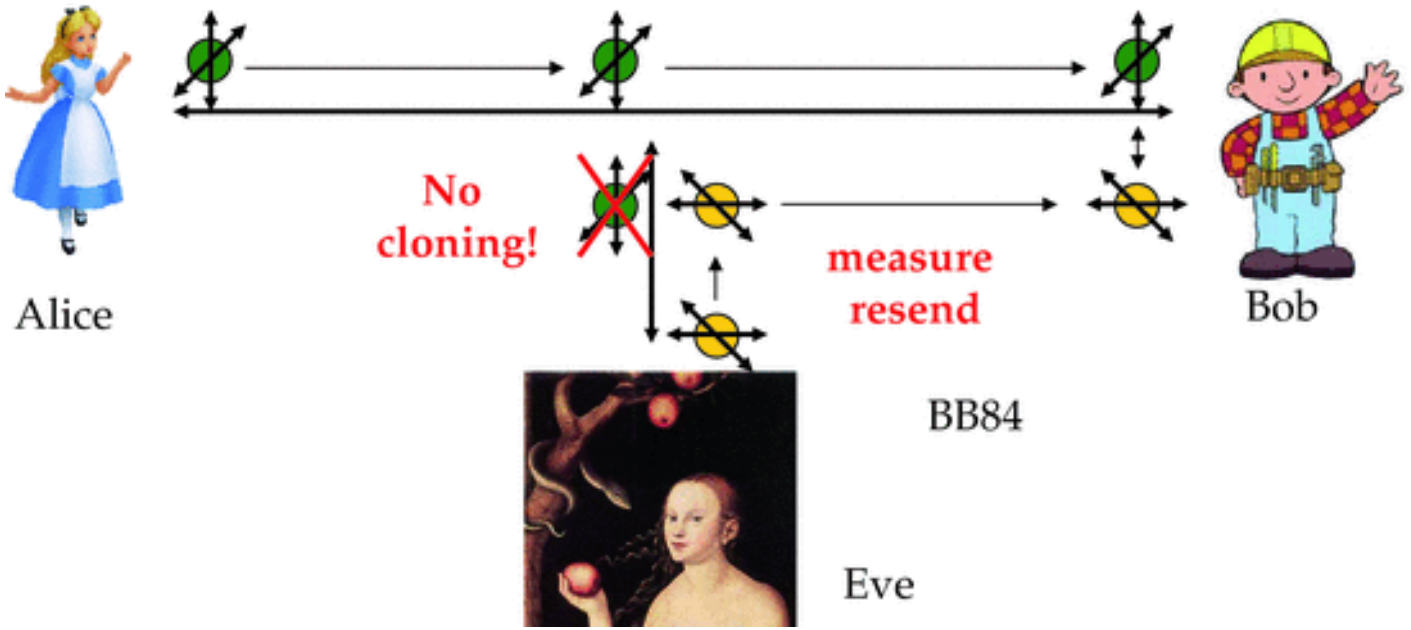
Step 3: *Apply CNOT gate to tensor product*

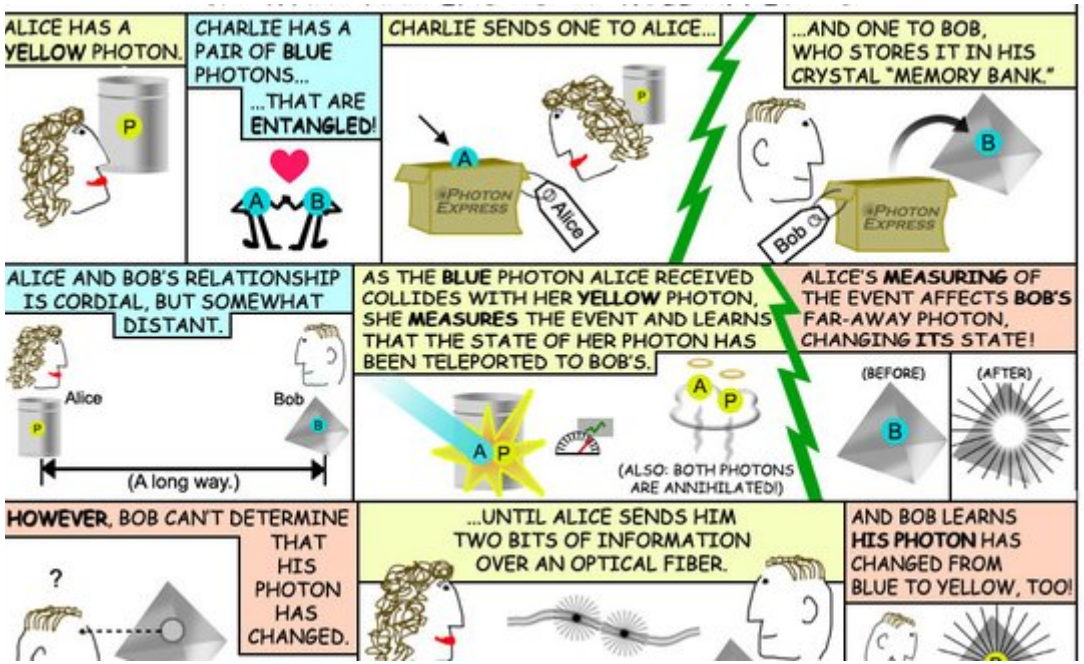
$$\text{CNOT}\,|H|0\rangle\,0\rangle = \begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 1\\ 0 & 0 & 1 & 0\end{bmatrix}\begin{bmatrix}\frac{1}{\sqrt{2}}\\ 0\\ \frac{1}{\sqrt{2}}\\ 0\end{bmatrix} = \begin{bmatrix}\frac{1}{\sqrt{2}}\\ 0\\ 0\\ \frac{1}{\sqrt{2}}\end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\ 0\\ 0\\ 0\end{bmatrix} + \frac{1}{\sqrt{2}}\begin{bmatrix}0\\ 0\\ 0\\ 1\end{bmatrix} = \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle$$



No Cloning

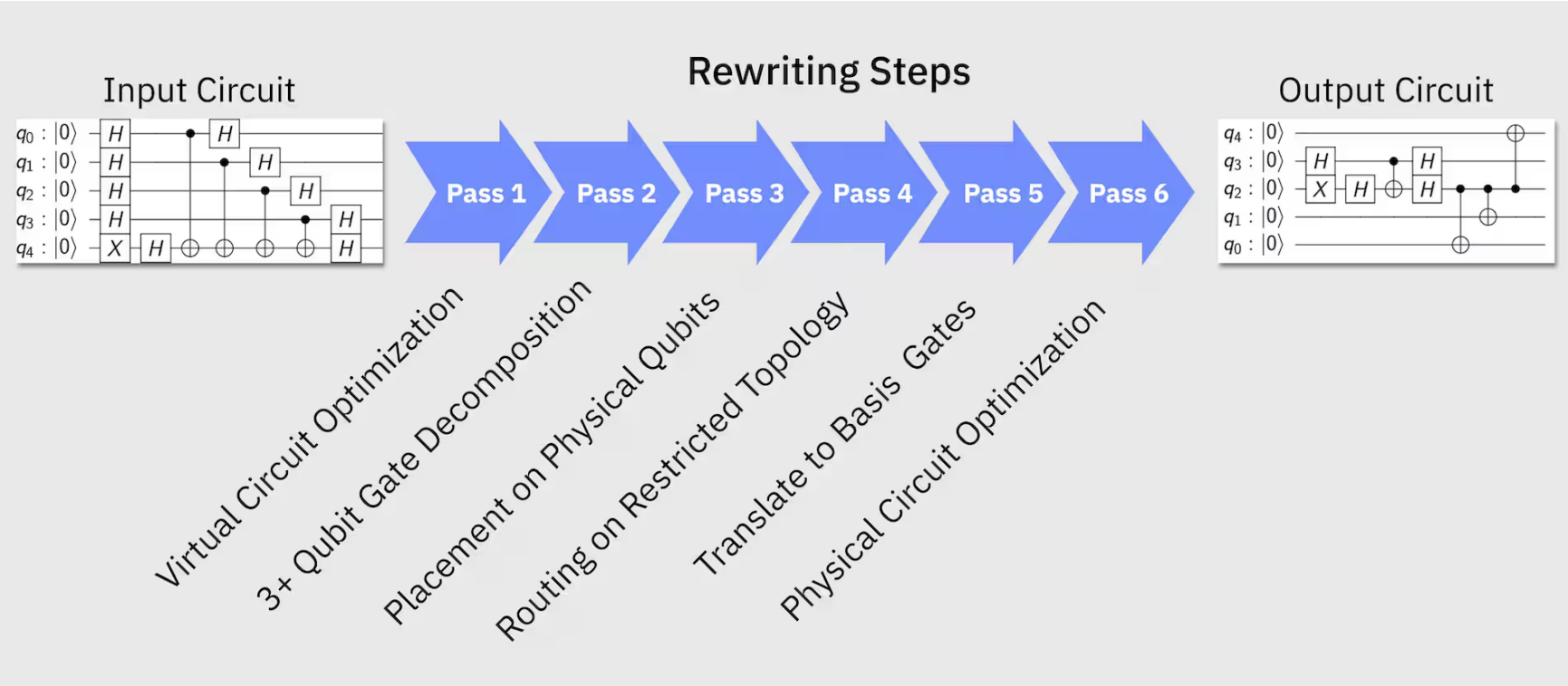
Qubits cannot be intercepted or cloned - creates a new tool for secure communication
Quantum teleportation possible!





Transpilation

Transpilation: conversion of quantum circuits to hardware level circuit execution on physical qubits and gates
Gate depth: Optimize both given quantum circuit and the physical output circuit to reduce the number of gates
Shots: On the hardware, the physical circuit is run multiple times to average out probabilistic behavior

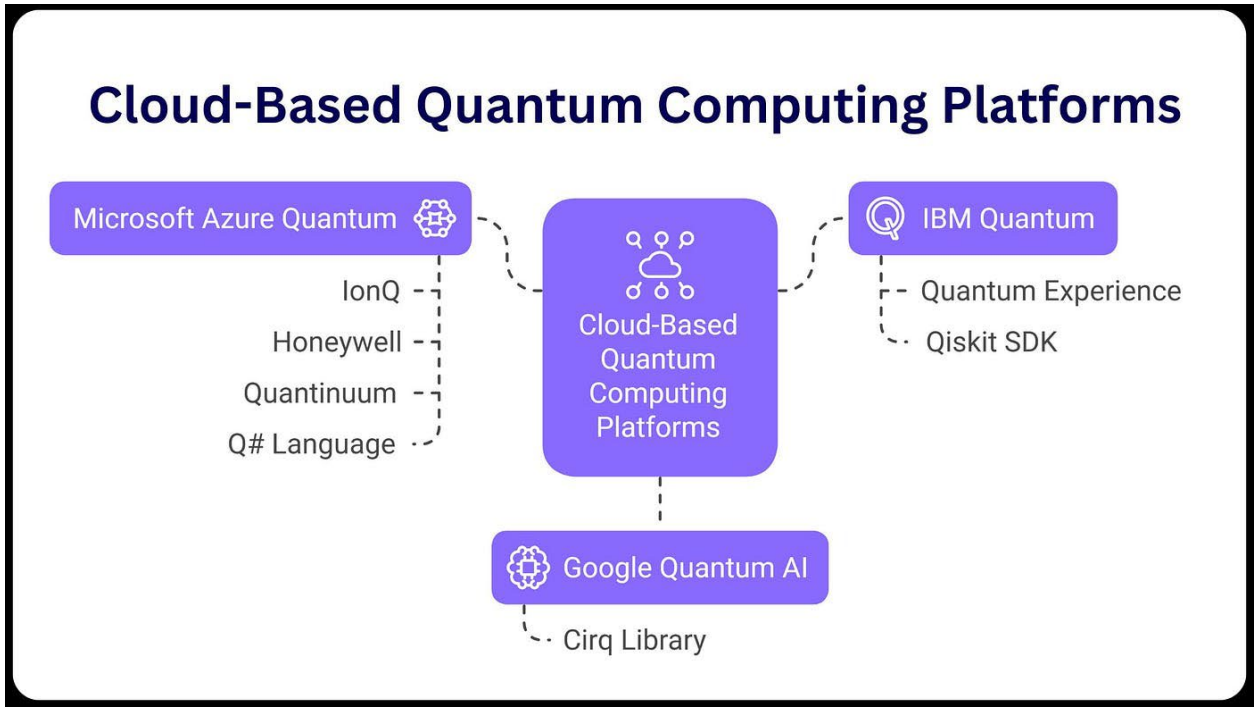


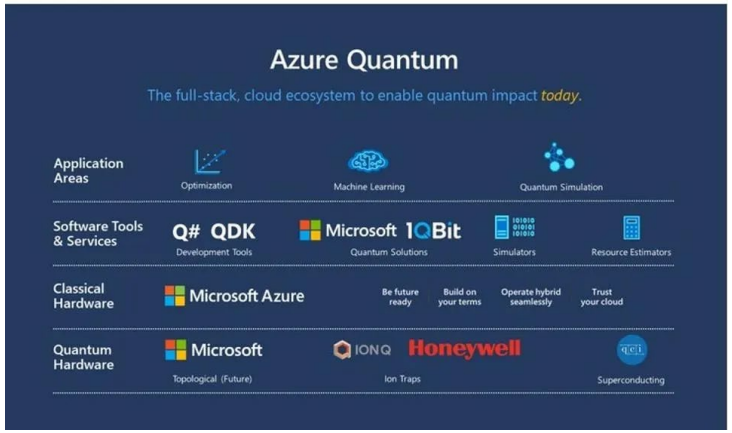
Quantum System Development Tools

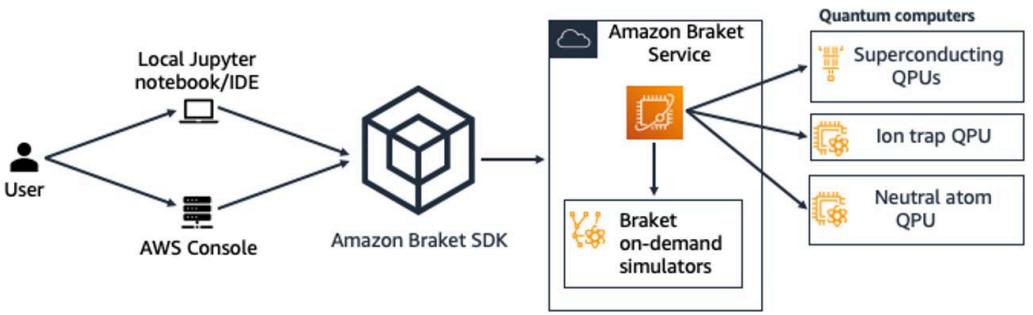
Many frameworks and libraries available to develop quantum algorithms
Describe quantum algorithms at high level in the framework using libraries
Simulate the operations using emulators to ensure correct results
Transpile to quantum operations for specified target hardware (e.g. Qiskit – IBM, Cirq – Google)

Quantum Computing Cloud Platforms

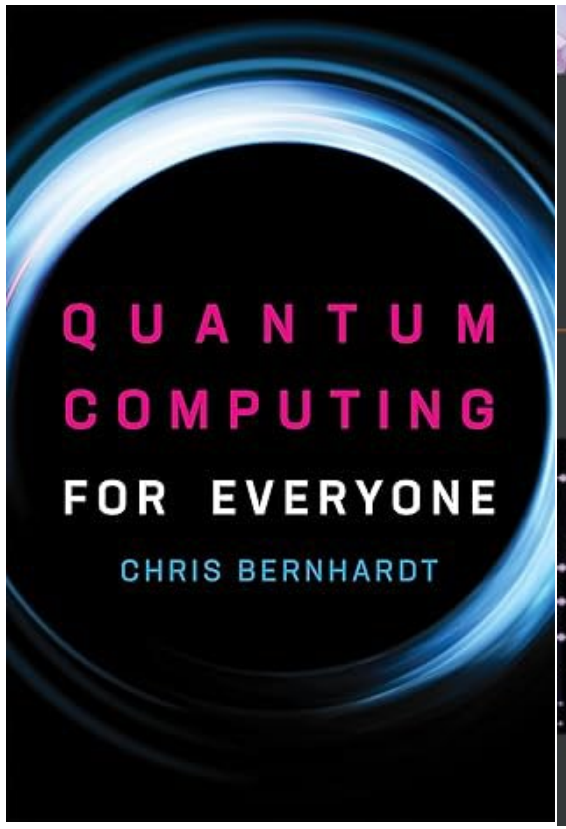
Access quantum computers without installing and paying for a quantum computer
Accompanied by SDK, simulators, solution examples and development resources
Range of hardwares available on each platform, e.g. AWS Braket and Microsoft Azure Quantum
Can be expensive to run jobs with many qubits and complex circuits

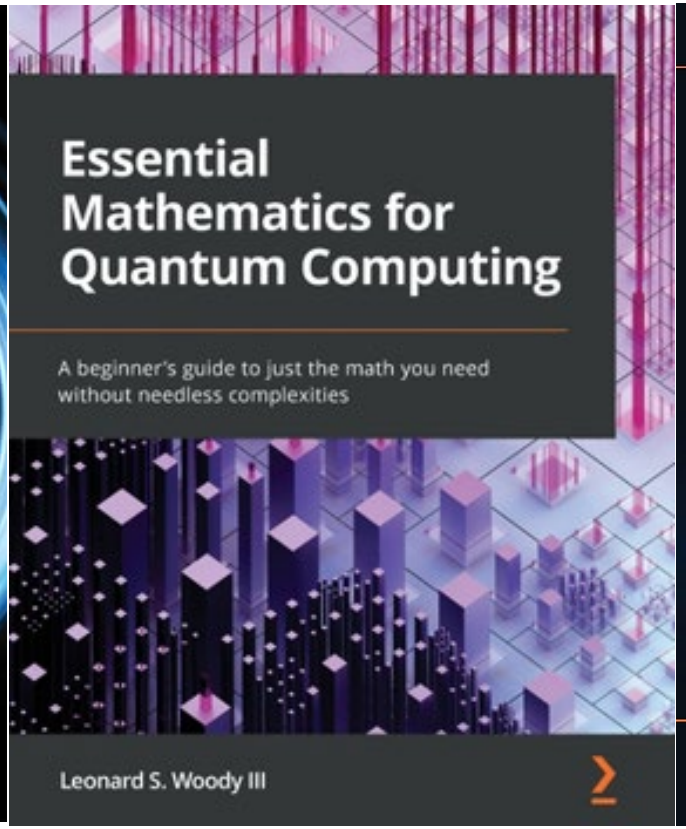


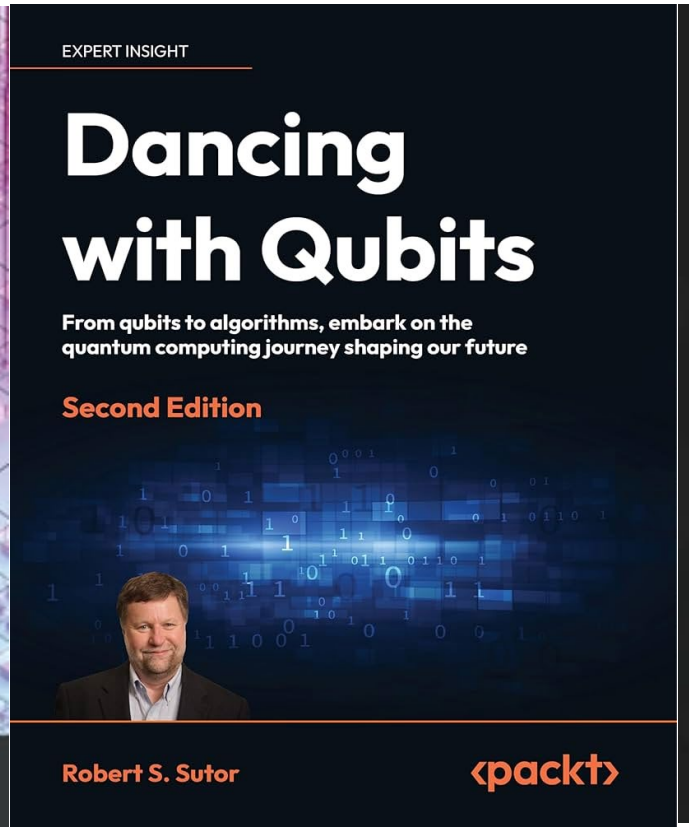


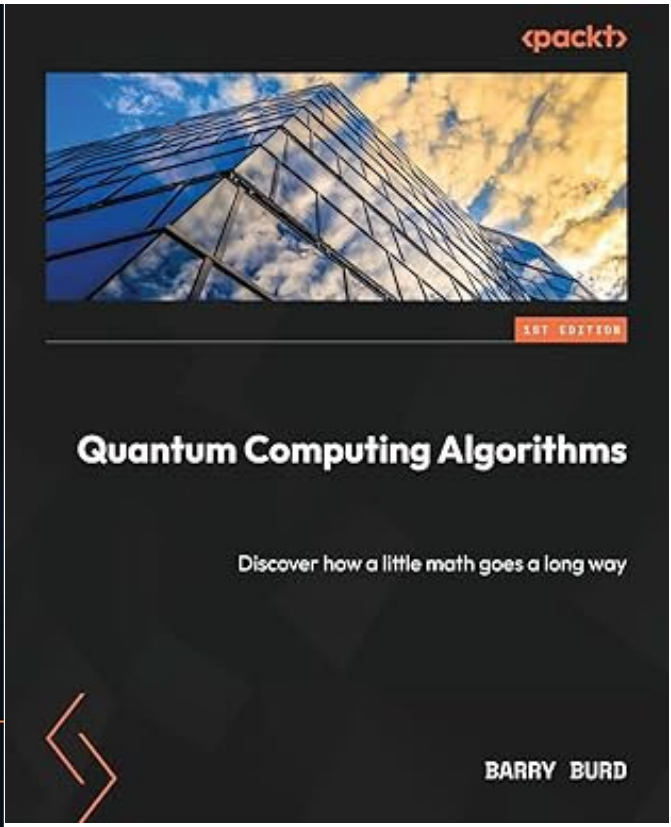


References


QUANTUM
COMPUTING
FOR EVERYONE
CHRIS BERNHARDT


Essential
Mathematics for
Quantum Computing
A beginner's guide to just the math you need
without needless complexities
Leonard S. Woody III


EXPERT INSIGHT
Dancing
with Qubits
From qubits to algorithms, embark on the
quantum computing journey shaping our future
Second Edition
Robert S. Sutor
packt


packt
1ST EDITION
Quantum Computing Algorithms
Discover how a little math goes a long way
BARRY BURD

Scan to access Google folder

DEMO