Using Quantum Mechanics *for* Computing *and* Cryptography

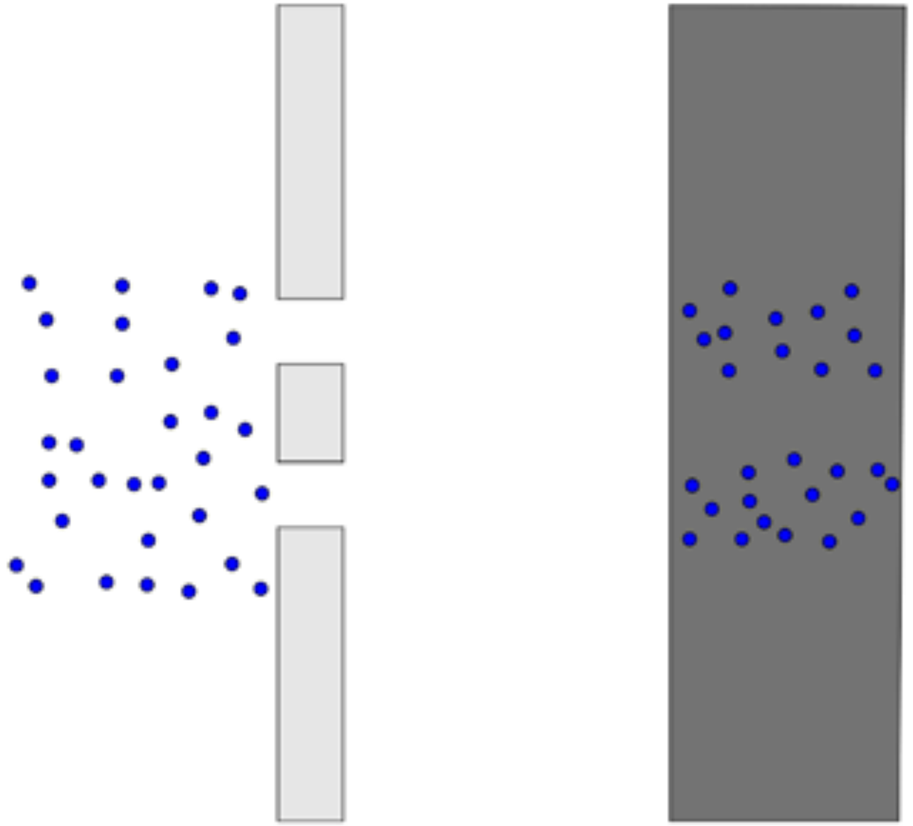
Raghav Govind Jha
Jefferson National Lab, Virginia, USA

September 04, 2024

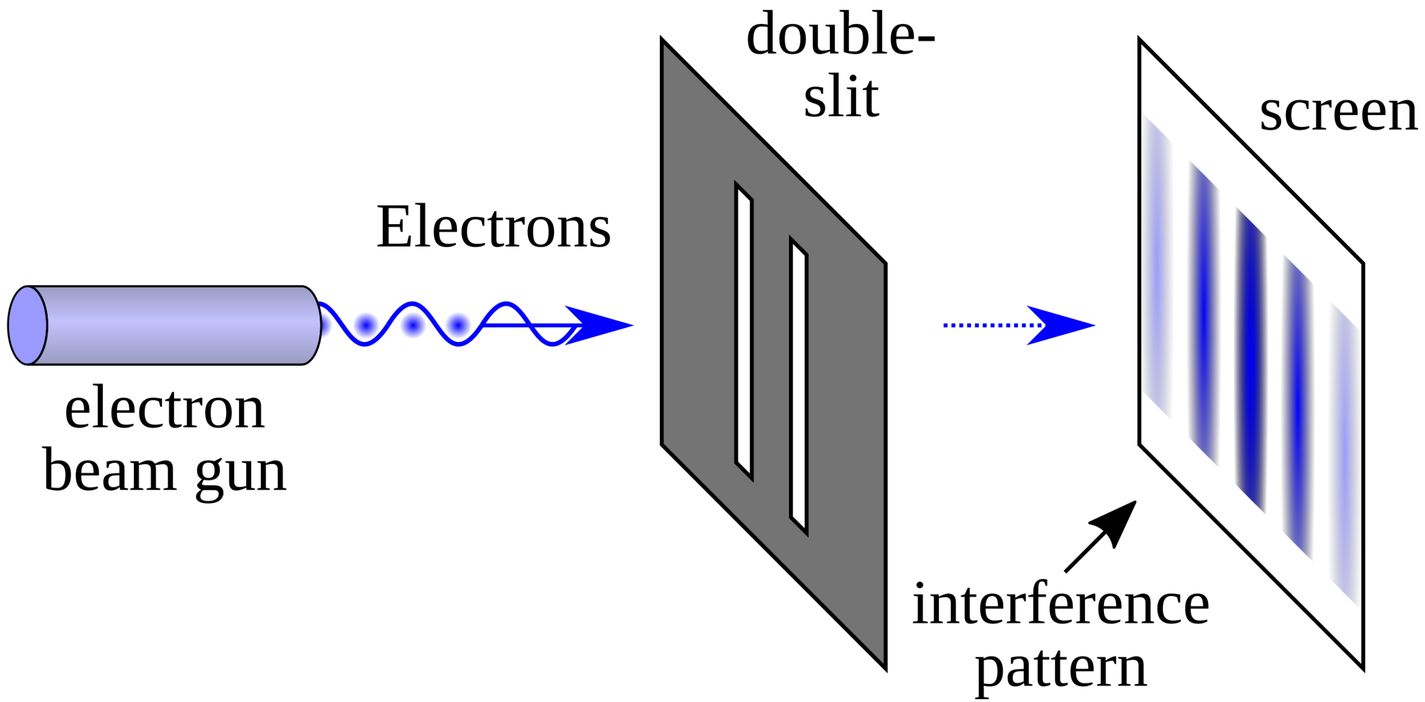
Outline

- Our understanding of the world: classical to quantum
- Hardness of problems: Why do we need help from quantum? Can it solve all NP problems?
- Basic fundamentals and quantum rules
- Classical and quantum cryptography. From RSA to BB84 quantum-key distribution protocol.
- Summary

Intuition: Classical Mechanics



Strange: Quantum Mechanics

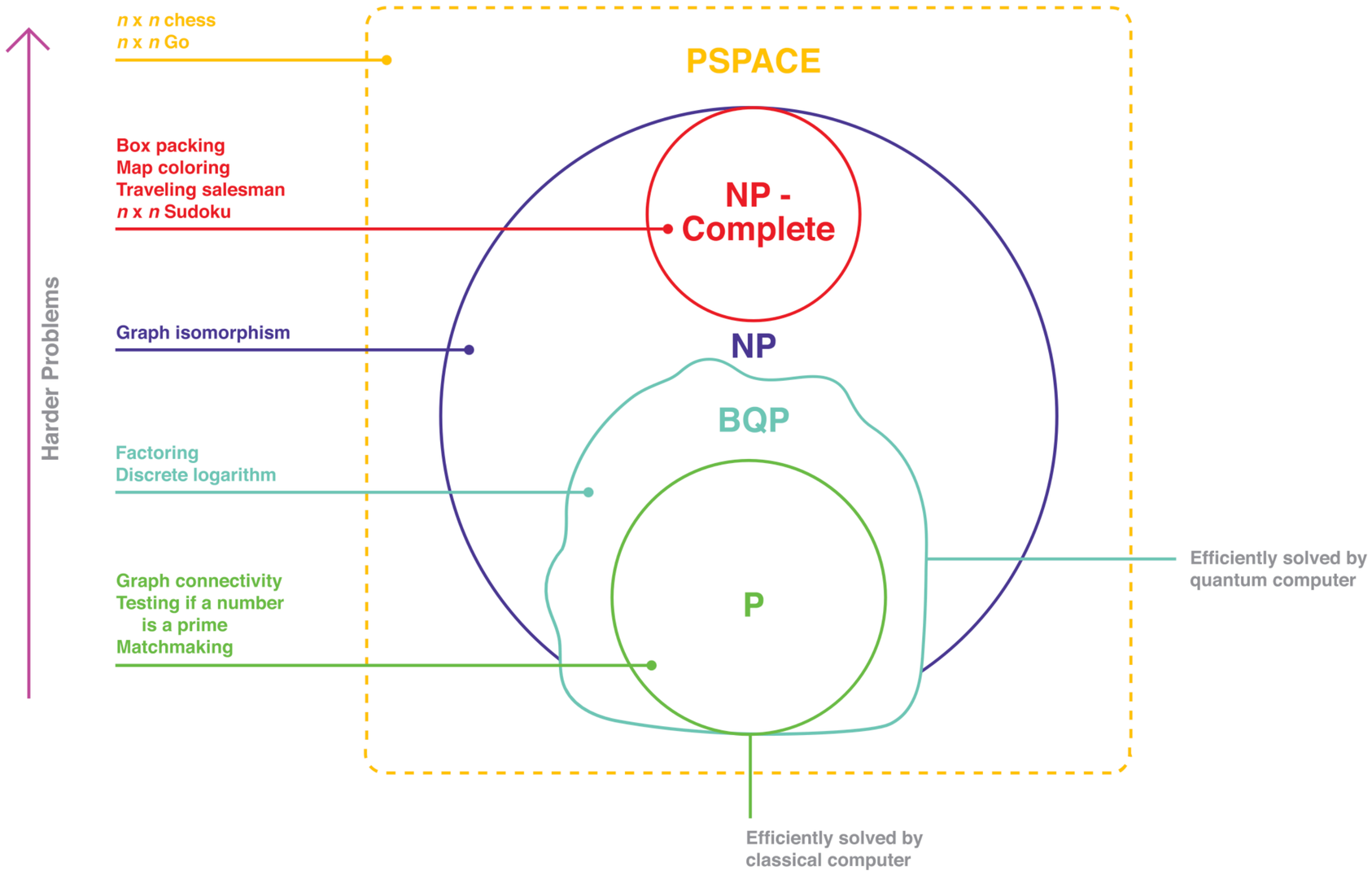


NOT JUST FOR PHOTONS (or LIGHT)

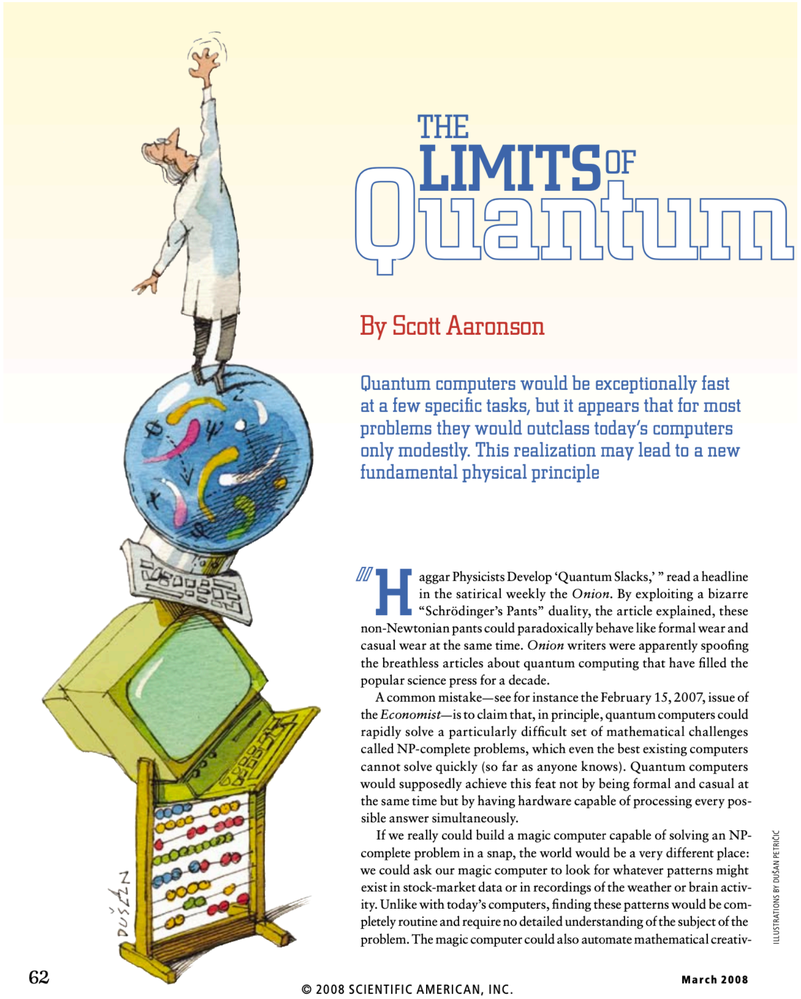
Why is it strange?

- I first studied quantum mechanics when I was less than 4 miles from this place in 2008. It is strange to me even after 16 years. It has been strange to the smartest of physicists now for hundred years! Feynman once said - “I think it is safe to say that nobody understands quantum mechanics.” But why?

Computation



Solve all problems impossible with classical computers?*



THE LIMITS OF Quantum

By Scott Aaronson

Quantum computers would be exceptionally fast at a few specific tasks, but it appears that for most problems they would outclass today's computers only modestly. This realization may lead to a new fundamental physical principle

"Haggar Physicists Develop 'Quantum Slacks,' " read a headline in the satirical weekly the *Onion*. By exploiting a bizarre "Schrödinger's Pants" duality, the article explained, these non-Newtonian pants could paradoxically behave like formal wear and casual wear at the same time. *Onion* writers were apparently spoofing the breathless articles about quantum computing that have filled the popular science press for a decade.

A common mistake—see for instance the February 15, 2007, issue of the *Economist*—is to claim that, in principle, quantum computers could rapidly solve a particularly difficult set of mathematical challenges called NP-complete problems, which even the best existing computers cannot solve quickly (so far as anyone knows). Quantum computers would supposedly achieve this feat not by being formal and casual at the same time but by having hardware capable of processing every possible answer simultaneously.

If we really could build a magic computer capable of solving an NP-complete problem in a snap, the world would be a very different place: we could ask our magic computer to look for whatever patterns might exist in stock-market data or in recordings of the weather or brain activity. Unlike with today's computers, finding these patterns would be completely routine and require no detailed understanding of the subject of the problem. The magic computer could also automate mathematical creativ-

ILLUSTRATIONS BY DUŠAN PETRIČIĆ

* No

Fate of impossible problems

- Many problems won’t have any advantage with fault-tolerant quantum computers.
- Some problems like factoring large semi-primes and searching database might see potential applications.
- There are only handful of problems where we know a “reliable” advantage due to QCs.
- Computation is not complete! There are things which you cannot do even with quantum computers. Fate of those problems can (most likely) never be known.

A new type of computer

Quantum Mechanical Computers

By Richard P. Feynman

Introduction

This work is a part of an effort to analyze the physical limitations of computers due to the laws of physics. For example, Bennett[1] has made a careful study of the free energy dissipation that must accompany computation. He found it to be virtually zero. He suggested to me the question of the limitations due to quantum mechanics and the uncertainty principle. I have found that, aside from the obvious limitation to size if the working parts are to be made of atoms, there is no fundamental limit from these sources either.

We are here considering ideal machines; the effects of small imperfections will be considered later. This study is one of principle; our aim is to exhibit some Hamiltonian for a system which could serve as a computer. We are not concerned with whether we have the most efficient system, nor how we could best implement it.

Since the laws of quantum physics are reversible in time, we shall have to consider computing engines which obey such reversible laws. This problem already occurred to Bennett[1], and to Fredkin and Toffoli[2], and a great deal of thought has been given to it. Since it may not be familiar to you here, I shall review this, and in doing so, take the opportunity to review, very briefly, the conclusions of Bennett[2], for we shall confirm them all when we analyze our quantum system.

It is a result of computer science that a universal computer can be made by a suitably complex network of interconnected primitive elements. Following the usual classical analysis we can imagine the interconnections to be ideal wires carrying one of two standard voltages representing the local 1 and 0. We can take the primitive elements to be just two, NOT and AND (actually just the one element NAND = NOT AND suffices, for if one input is set at 1 the output is the NOT of the other input). They are symbolized in Fig. 1, with the logical values resulting on the outgoing wires, resulting from different combinations of input wires.

From a logical point of view, we must consider the wires in detail, for in other systems, and our quantum system in particular, we may not have wires as such. We see we really have two more logical primitives, FAN OUT when two wires are connected to one, and EXCHANGE, when wires are crossed. In the usual computer the NOT and NAND primitives are implemented by transistors, possibly as in Fig. 2.

What is the minimum free energy that must be expended to operate an ideal computer made of such primitives? Since, for example, when the AND operates the output line, c' is being determined to be one of two values no matter what it was before the entropy change is $ln(2)$ units. This represents a heat generation of $kT\,ln(2)$ at temperature T. For many years it was thought that this represented an absolute minimum to the quantity of heat per primitive step that had to be dissipated in making a calculation.

The question is academic at this time. In actual machines we are quite concerned with the heat dissipation question, but the transistor system used actually dissipates about $10^{10}kT$! As Bennett[3] has pointed out, this arises because to change a wire's voltage we dump it to ground through a resistance; and to build it up again we feed charge, again through a resistance, to the wire. It could be greatly reduced if energy could be stored in an inductance, or other reactive element.

Richard P. Feynman is a professor of theoretical physics at California Institute of Technology. This article is based on his plenary talk presented at the CLEO/IQEC Meeting in 1984.

However, it is apparently very difficult to make inductive elements on silicon wafers with present techniques. Even Nature, in her DNA copying machine, dissipates about 100 kT per bit copied. Being, at present, so very far from this $kT\,ln(2)$ figure, it seems ridiculous to argue that even this is too high and the minimum is really essentially zero. But, we are going to be even more ridiculous later and consider bits written on one atom instead of the present 10^{11} atoms. Such nonsense is very entertaining to professors like me. I hope you will find it interesting and entertaining also.

What Bennett pointed out was that this former limit was wrong because it is not necessary to use irreversible primitives. Calculations can be done with reversible machines containing only reversible primitives. If this is done the minimum free energy required is independent of the complexity or number of logical steps in the calculation. If anything, it is kT per bit of the output answer.

But even this, which might be considered the free energy needed to clear the computer for further use, might also be considered as part of what you are going to do with the answer—the information in the result if you transmit it to another point. This is a limit only achieved ideally if you compute with a reversible computer at infinitesimal speed.

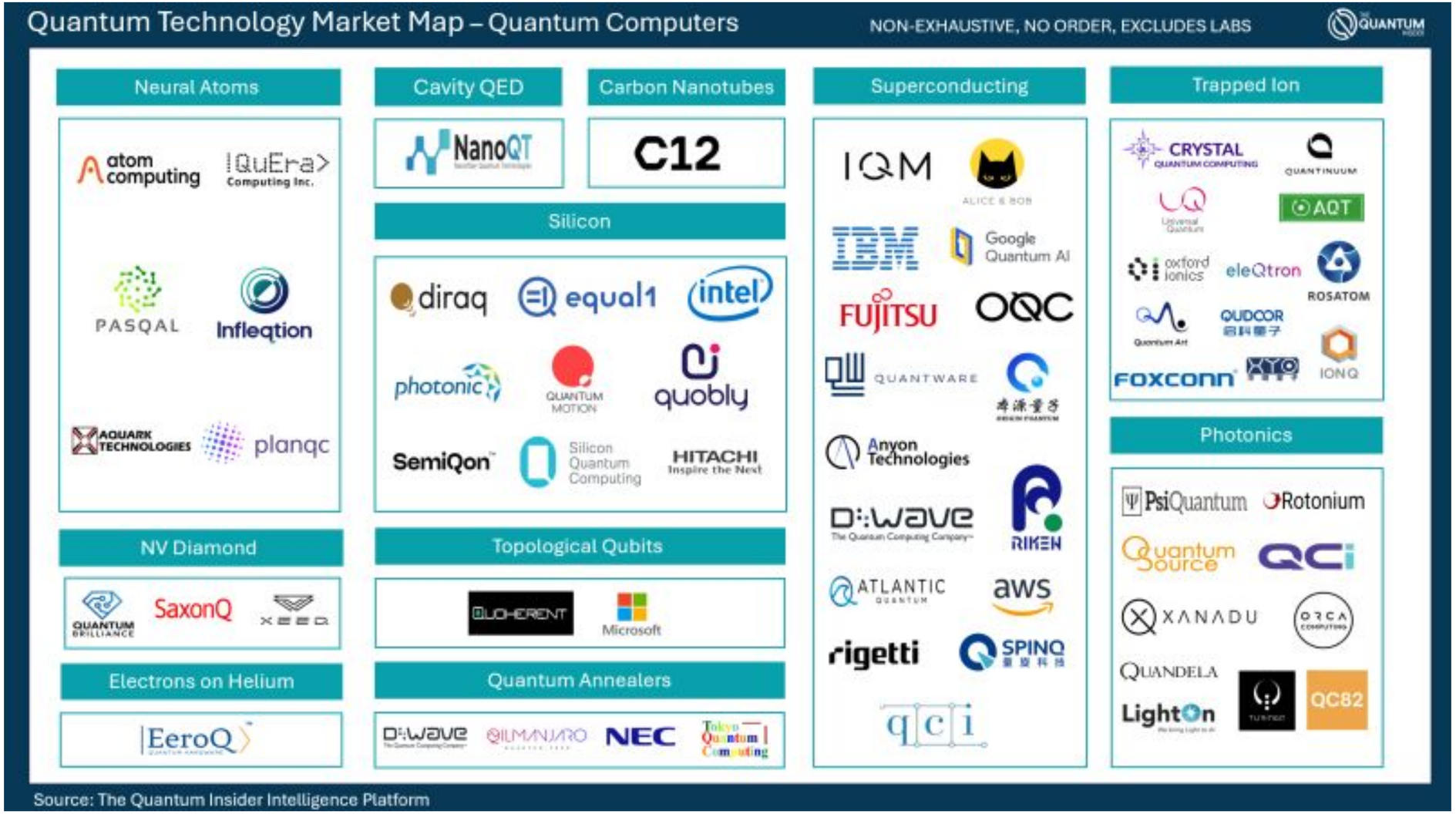
Computation with a reversible machine

We will now describe three reversible primitives that could be used to make a universal machine (Toffoli[4]). The first is the NOT which evidently loses no information, and is reversible, being reversed by acting again with NOT. Because the conventional symbol is not symmetrical we shall use an X on the wire instead (see Fig. 3a).

Next is what we shall call the CONTROLLED NOT (see Fig. 3b). There are two entering lines, a and b and two exiting lines, a' and b'. The a' is always the same as a, which is the control line. If the control is activated $a = 1$ then the out b' is the NOT of b. Otherwise b is unchanged, $b' = b$. The table of values

Based on a talk in 1984

Quantum industry boom



Comparison

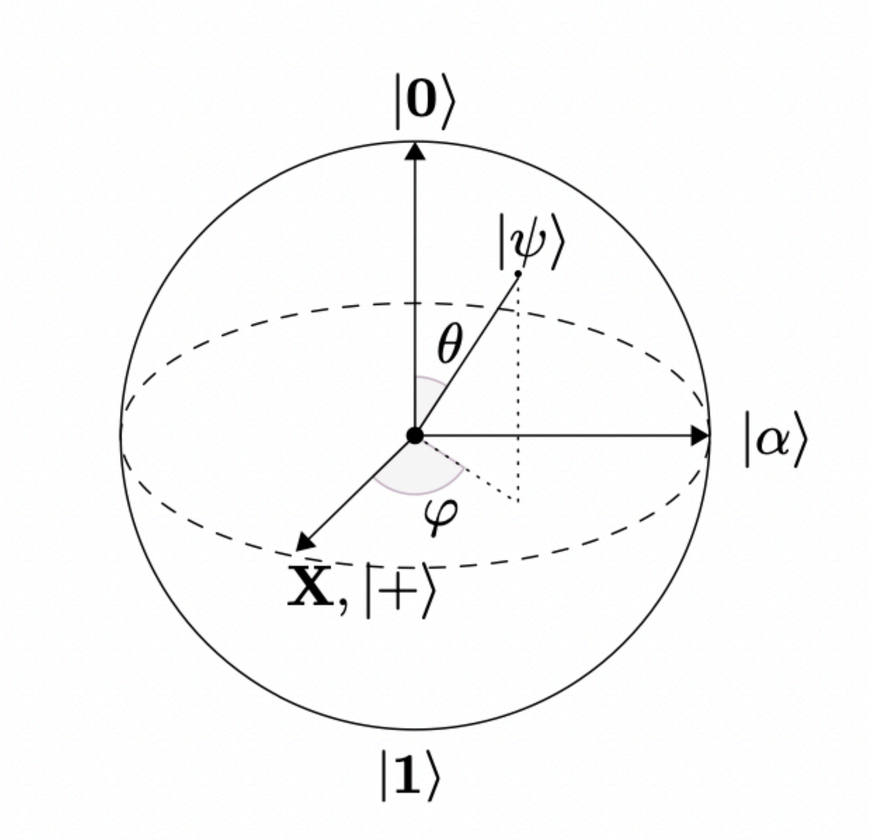
Elements	Classical	Quantum
Algebra	Boolean	Linear
Basic Unit	Bit	Qubit
Gates	Logic gates	Unitary gates
Reversibility of gates	Sometimes	Always
Example of universal gate set	{NAND}	{H, T, CNOT}
Correction of errors	Repetition	Shor-like code

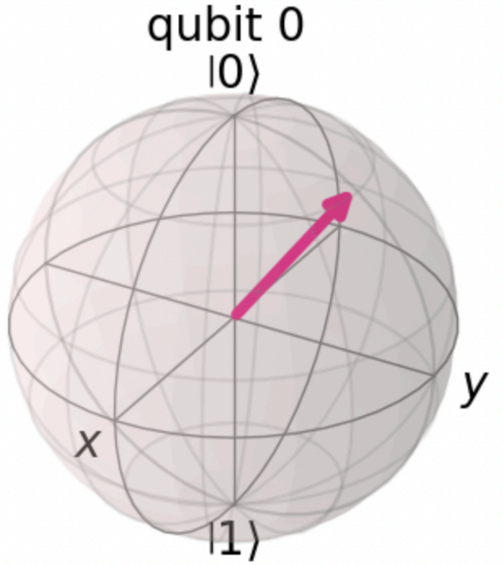
What is a qubit?

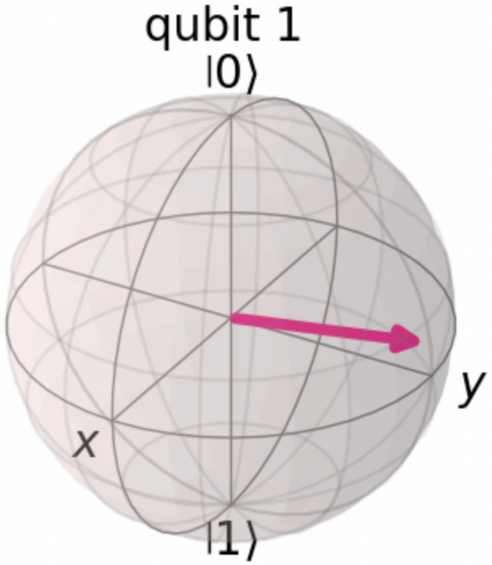
- Qubit or quantum bit is the quantum analog of classical bit (0 and 1). In quantum mechanics, there are several properties which have no classical analog. For example, the spin of the electron and two-state quantum systems.

- Electron is a spin-1/2 particle and has two states since $2s+1 = 2$. In addition, there are other ways to build qubits. One way which we will encounter later in the talk is based on two polarisation states of a photon.

Qubit representation







Classical and Quantum logic

A	B	AND $(A \cdot B)$	OR $(A + B)$	XOR$(A \oplus B)$
0	0	0	0	0
0	1	0	1	1
1	0	0	1	1
1	1	1	1	0

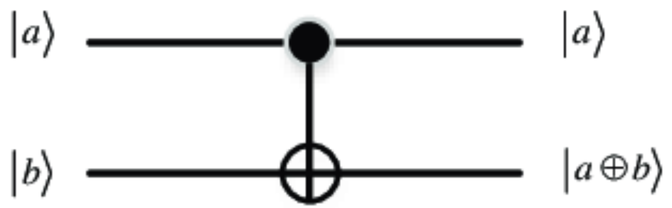
Classical: Boolean Algebra

$\|A\rangle$	$\|B\rangle$	$\|A\rangle$	$\|A \oplus B\rangle$
$\|0\rangle$	$\|0\rangle$	$\|0\rangle$	$\|0\rangle$
$\|0\rangle$	$\|1\rangle$	$\|0\rangle$	$\|1\rangle$
$\|1\rangle$	$\|0\rangle$	$\|1\rangle$	$\|1\rangle$
$\|1\rangle$	$\|1\rangle$	$\|1\rangle$	$\|0\rangle$

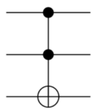
XOR operation: Classical and Quantum

A	B	Q
0	0	0
0	1	1
1	0	1
1	1	0

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



Quantum logic gates

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)		$\begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix}$
π/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)		$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$

Basic principles of the quantum world

- Uncertainty principle: We cannot measure certain observables simultaneously. There is an element of uncertainty in the microscopic world. Note that this is fundamental principle and not due to the lack of experimental apparatus or precision.

- Superposition principle: The electron can be in a superposition between up and down (0 or 1) states. In the sense, it has both spin-up and spin-down until it is measured.

- Entanglement: The spins of two electrons (or two qubits) can be entangled such that they cannot be separated or dealt independently.

- If we go and measure the spin of the electron or the polarization of the photon, we will destroy the superposition and electron will be forced to either choose “up” or “down”. In cryptography, this is important since it clearly implies that eavesdropping can be easily detected!

Classical Cryptography

- RSA-250 is the current state of the art with 250 decimal digits and 829 bits. Useful conversion is ~10/3 times number of digits for large integers. This semi-prime was factored in 2020. An example of semi-prime is 15.
- The fundamental idea is: It is *almost impossible* to factor large semi-primes using any classical computer and algorithm!
- One of the best known quantum algorithms, Shor's algorithm, has an exponential advantage over best-known classical method for factoring integers. While the classical complexity is $\exp(cd^{1/3})$ where c is a constant and d is the number of digits, the Shor's algorithm is $\sim d^3$.

Classical Cryptography

The famous RSA algorithm introduced in 1977 was given by

1. Generate two large random primes, p and q of approximately equal size

2. Compute $n = p \cdot q$ and $\phi = (p-1) \cdot (q-1)$

3. Choose integer e, $1 < e < \phi$ s.t. $\gcd(e, \phi) = 1$ & compute the secret exponent d, $1 < d < \phi$ s.t $e \cdot d = 1 \pmod{\phi}$

4. The public key is (n, e) and the private key (d, p, q) or (n, d).

We refer to e as the public exponent (encryption exponent) and d as the secret exponent (decryption)

Classical Cryptography: Example

1. Generate two large random primes, p and q of approximately equal size –> 11 and 3

2. Compute $n = p \cdot q$ and $\phi = (p-1) \cdot (q-1)$ –> n = 33, $\phi = 20$

3. Choose integer e, $1 < e < \phi$ s.t. $\gcd(e, \phi) = 1$ & compute the secret exponent d, $1 < d < \phi$ s.t $e \cdot d = 1 \pmod{\phi}$ –> e = 3, d = 7

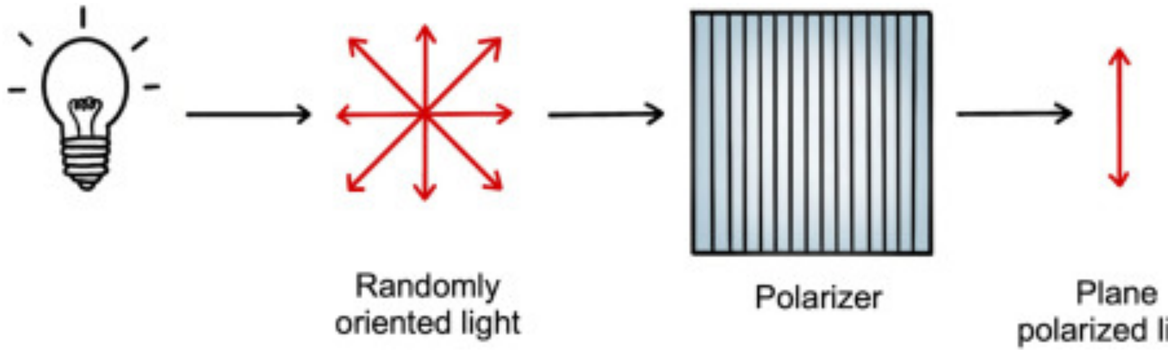
4. The public key is (n, e) = (33,3) and the private key (d, p, q) or (n, d) = (33,7)

We refer to e as the public exponent (encryption exponent) and d as the secret exponent (decryption). Note that if p, q are known

Quantum Cryptography

- Can quantum mechanics help with transfer of secure information? Are there any advantages?
- The first quantum key distribution protocol was proposed by Bennett and Brassard in 1984, now known as BB84. There are several others but for our purposes, this is sufficient.

Quantum Cryptography



$|V\rangle$

$|H\rangle$

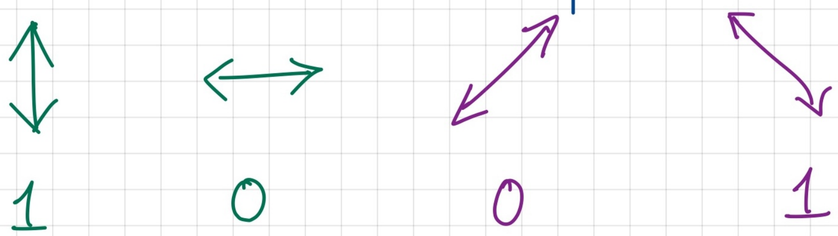
$|D\rangle$

$|A\rangle$

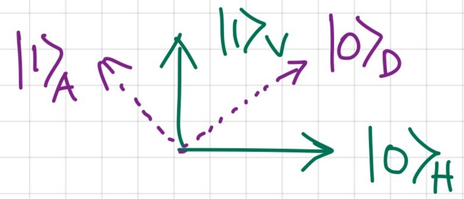
- Can also use rectilinear and circular polarisation states. We will stick to diagonal and rectilinear bases.

Quantum Cryptography

Alice Random Bit-String	0	1	1	0	0
Alice's Basis	+	+	X	+	X
Alice's Polarization	H	V	A	H	D
Bob's Random Basis	+	X	X	+	+
Bob's Polarization Obtained	H	D	A	H	V
Bob's Bit-String	0	0	1	0	1
Discussion (Public)	Yes	No	Yes	Yes	No
Sifted Key	0		1	0	



1 0 0 1



+ to X : rotate by 45° ↷

X to + : rotate by 45° ↶

Quantum bit error rate (QBER) ~ 25-40% considered secure key transfer!!

Quantum Cryptography

- An advantage of quantum key distribution is that eavesdropping can be found out.
- QBER: Suppose Alice sends 1000 qubits to Bob. After basis reconciliation, they find that 500 qubits were measured in matching bases. To estimate the QBER, they randomly select 100 of these 500 qubits and compare their values. If they find 5 discrepancies, the QBER is 5%. If it exceeds 25%, eavesdropping is likely.

Summary

- Quantum mechanics is a revolutionary way forward for computing and cryptography. It is not clear which problems would benefit the use of quantum. Open area of research.
- In coming decades, post-quantum (classical) cryptography and quantum cryptography will be an important research direction and might become a way to secure the internet or communication after several decades.

Thank you