Quantum Computing

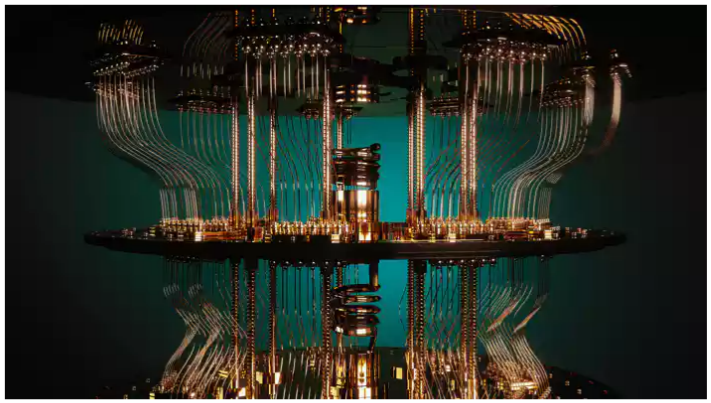
Introduction

Uwe Egly

Knowledge-Based Systems Group
Institute of Logic and Computation
Vienna University of Technology

Some slides have been inspired by this talk

Propaganda: Why is it important to learn about QC?



Bundesregierung: Tausende Fachkräfte für Quantentechnologien fehlen

Deutschland soll sich laut der Regierung etwa bei Quantencomputern "einen Platz an der Weltspitze" sichern. Doch der Fachkräftemangel schlägt auch hier zu.

18:31 Uhr 4

www.heise.de, Sept. 19, 2023

☞ Job offers in academia and industry in the future

The very early days of computing

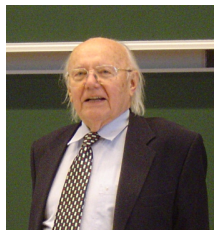
Mechanical machines of

- W. Schickard (1592–1635)
- B. Pascal (1623–1662),
- G. W. Leibniz (1646–1716)
- C. Babbage (1791–1871) Difference Engine No. 1 1821
  - First mechanical (specialized) computer
  - Not finished because of manufacture difficulties (high precision)
  - 1991 reconstructed for the London Science Museum
  - Babbage planed a programmable computer: Analytical Engine

The early days of computing

Technology and examples

- Relay: K. Zuse (1910–1995) Z3 from 1941, programmable
- Vacuum tube: Whirlwind I (late 40s, MIT)
- Transistor: H. Zemanek (1920–2014) Mailüfterl (May breeze)
- ICs
- Microprocessors



Since then: Make HW smaller and smaller and smaller and smaller ...

Moore's Law: *The number of transistors in a dense integrated circuit doubles about every two years.*

End of it at the horizon! What then?

What is quantum computing?

Quantum computing is computing using quantum-mechanical phenomena, such as superposition and entanglement.

From Wikipedia (quantum computing)

Quantum mechanics is a mathematical framework for the development of physical theories. On its own quantum mechanics doesn't tell you what laws a physical system must obey, but it does provide a mathematical and conceptual framework for the development of such laws.

Quantum Computation and Quantum Information, p. 80

Postulates of quantum mechanics

Obtained experimentally in a long process of trial and error

- "Closed" systems are described by unit state vectors in a state space of the system
- State space: a complex Hilbert space (dimension often finite)
- The temporal evolution is described by unitary transformations
- Measurement operators (acting on state space) describe the measurement result and the successor state probabilistically
- The state space of a composite system is given by the tensor product of the state spaces of its components

Model of computation for quantum computers

- A quantum computer has a state represented by a quantum register which is initializes in a predefined way (↪ start state)
- The state evolves by applying operations
  (specified in advance as an algorithm)
- At the end of the computation, some information on the state is obtained by measuring parts of the quantum register

Superposition

Bit vs quantum bit (qubit)

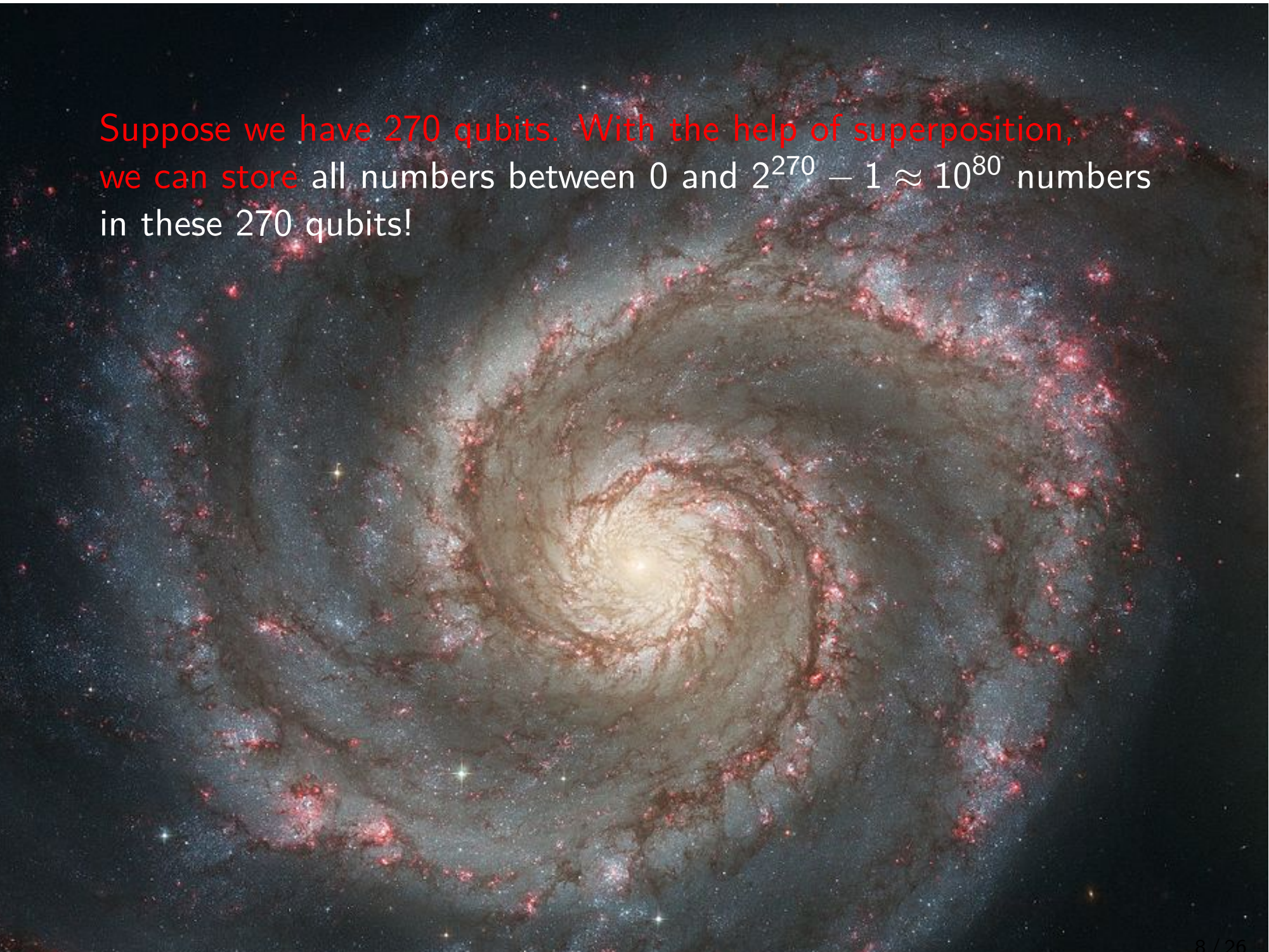
- Classical computers store information as (strings of) bits
- A bit is either false (in the $|0\rangle$ state) or true (in the $|1\rangle$ state)
- A qubit can be in state $|0\rangle$, $|1\rangle$, or in a "mixed" state
- A mixed state is $\alpha|0\rangle + \beta|1\rangle$ with α, β complex numbers and

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

- Such a mixed state is called a (quantum) superposition
- n qubits can be in a superposition of 2^n states, e.g., for $n = 2$,

$$\alpha_0|00\rangle + \alpha_1|01\rangle + \alpha_2|10\rangle + \alpha_3|11\rangle$$

- A quantum computer can deal with these 2^n states in parallel



Suppose we have 270 qubits. With the help of superposition, we can store all numbers between 0 and $2^{270} - 1 \approx 10^{80}$ numbers in these 270 qubits!

Wally and his quantum computer project

Tuesday April 17, 2012

Measurement

- Measurement of some qubits: outcome with probability equal to the norm of its corresponding coefficient squared
- The norm of a complex number $c = a + bi$ is

$$|c| = \sqrt{(a + bi)(a - bi)} = \sqrt{a^2 + b^2}$$

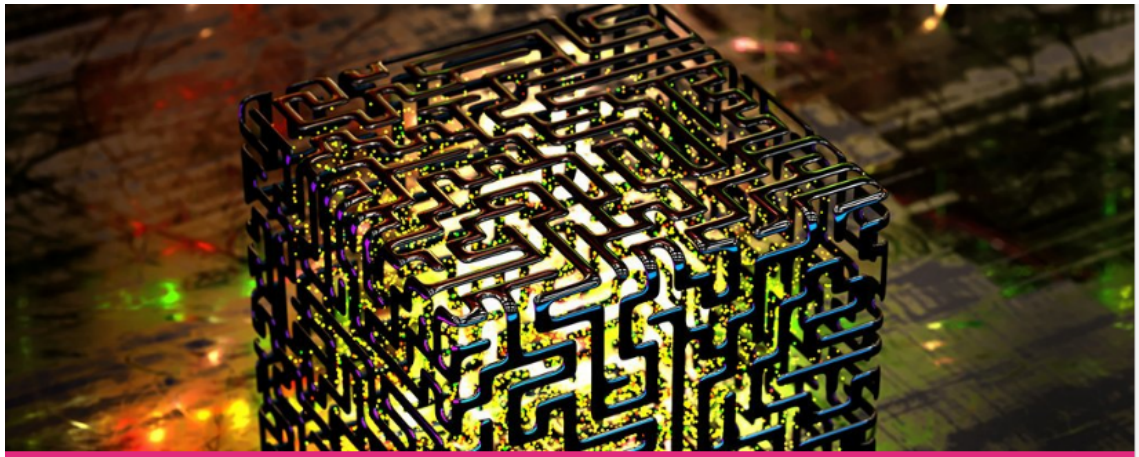
- Example: Measure a qubit in the state $\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$
- Potential outcomes:
  1. Classical bit 0 and successor state $|0\rangle$ with probability $\left(\frac{1}{\sqrt{2}}\right)^2 = \frac{1}{2}$, and
  2. classical bit 1 and successor state $|1\rangle$ with probability $\frac{1}{2}$

Quantum entanglement

- A system is entangled if the corresponding quantum state cannot be written as a tensor product of its components.
- Example: A 2 qubit system in state $\frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle$
- This is called Bell state or ERP pair
- Suppose 1 qubit is in Vienna and 1 is on the moon
- We measure the qubit in Vienna
- Surprise: If we measure 1, we have instantly a 1 on the moon
- Measurement affects both qubits at the same time regardless where they are ("spooky action at a distance" © A. Einstein)

Entanglement

A realization record from the University of Innsbruck



(NiPlot/iStock)

PHYSICS

Physicists Just Broke a Quantum Record, Taking Entanglement to a Spooky New Level

MICHELLE STARR 16 APR 2018

If we want quantum computers, we're going to need a complex system of quantum entangled particles - particles that are intrinsically linked so that whatever happens to one instantaneously affects another.

That's a whole lot easier said than done, of course - but a team of physicists has just breached an exciting milestone by creating a 20-bit quantum register.

Quantum bits, or qubits, are the basic building blocks of quantum computing, just like bits are the building blocks of traditional computing.

But what's challenging about them is that they rely on subatomic particles' spooky ability to exist in more than one state at the same time.

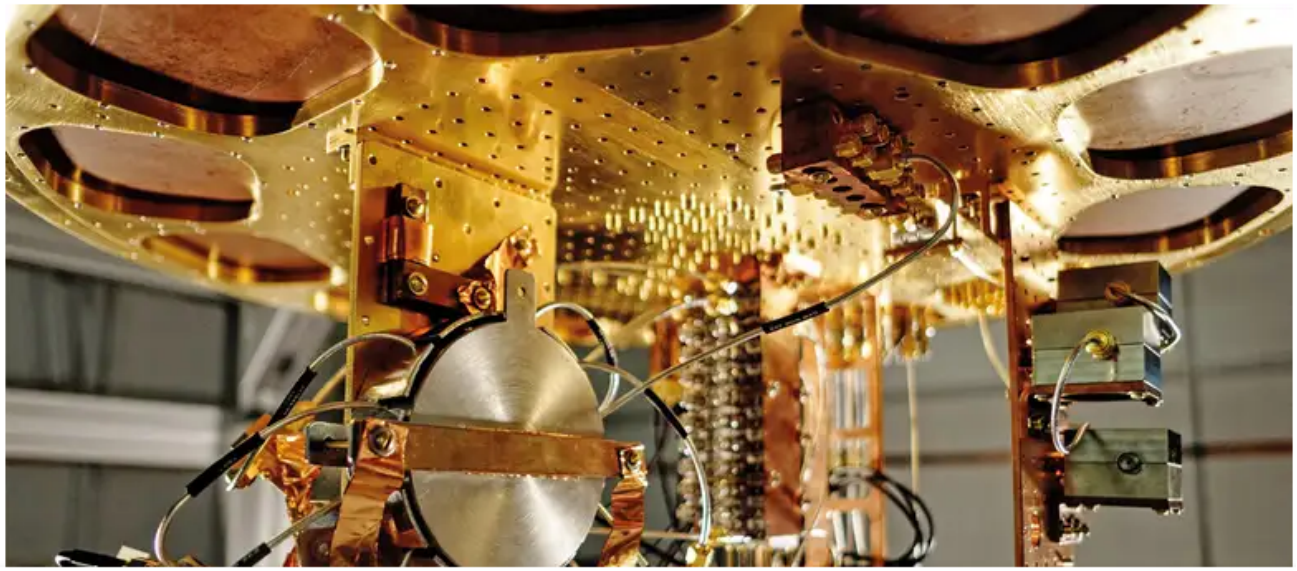
What can quantum computers do better (perhaps)?

NewScientist

Google has reached quantum supremacy – here's what it should do next

TECHNOLOGY | ANALYSIS 26 September 2019

By Chelsea Whyte



What can quantum computers do better (perhaps)?

NewScientist

~~Google has reached quantum supremacy – here's what it should do next~~

Solving the sampling problem of the Sycamore quantum supremacy circuits

Feng Pan,[1,2] Keyang Chen,[1,3] and Pan Zhang[1,*]

[1]*CAS Key Laboratory for Theoretical Physics, Institute of Theoretical Physics, Chinese Academy of Sciences, Beijing 100190, China*
[2]*School of Physical Sciences, University of Chinese Academy of Sciences, Beijing 100049, China*
[3]*Yuanpei College, Peking University, Beijing 100871, China.*

We study the problem of generating independent samples from the output distribution of Google's Sycamore quantum circuits with a target fidelity, which is believed to be beyond the reach of classical supercomputers and has been used to demonstrate quantum supremacy. We propose a new method to classically solve this problem by contracting the corresponding tensor network just once, and is massively more efficient than existing methods in obtaining a large number of *uncorrelated* samples with a target fidelity. For the Sycamore quantum supremacy circuit with 53 qubits and 20 cycles, we have generated one million *uncorrelated* bitstrings $\mathbf{s}$ which are sampled from a distribution $\widehat{P}(\mathbf{s}) = |\widehat{\psi}(\mathbf{s})|^2$, where the approximate state $\widehat{\psi}$ has fidelity $F \approx 0.0037$. The whole computation has cost about 15 hours on a computational cluster with 512 GPUs. The obtained one million samples, the contraction code and contraction order are made public. If our algorithm could be implemented with high efficiency on a modern supercomputer with ExaFLOPS performance, we estimate that ideally, the simulation would cost a few dozens of seconds, which is faster than Google's quantum hardware.

What can quantum computers do better (perhaps)?

We briefly discuss here two problems

P1 Integer factorization (and the discrete logarithm problem)

P2 Search in unstructured databases

Consequences of polynomial-time algorithms for problem P1

- Attack against RSA and elliptic curve primitives
- Attack against different key exchange protocols

Cryptographers look for "post-quantum cryptography"

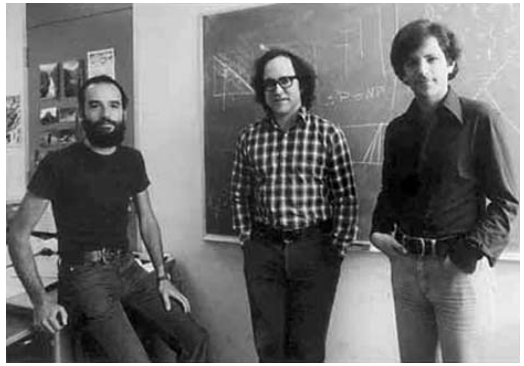
Shor's algorithm (1994)

Integer factorization (IF)

is the decomposition of a composite number into a product of smaller integers. If these integers are further restricted to prime numbers, the process is called prime factorization.

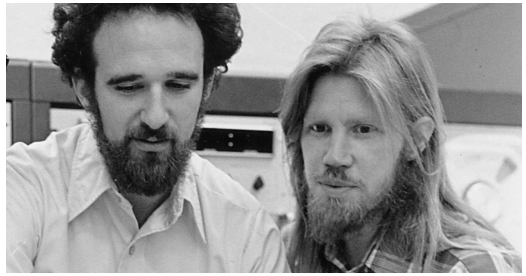
- No efficient classical algorithm known
- Best classical algorithm: General Number Field Sieve
- It has run-time $O(\exp[c(\log n)^{1/3}(\log\log n)^{2/3}])$ where c is a constant depending on the sieve type
- P. Shor (1994): Quantum algorithm for efficient IF
- It solves also the discrete logarithm (DLP) problem efficiently

The security of RSA



- Published by R. Rivest, A. Shamir and L. Adleman 1978
- Usage: Encryption and decryption, digital signature
- An equivalent system obtained by Ellis, Cocks, Williamson at GCHQ but kept secret til 1997 (Ellis' report is from 1973)
- Security of RSA not proven
- It is based on the computational difficulty to solve the factorization problem for integers
- Factorization problem not "provably infeasible", but ...
- ... no polynomial classical algorithm known at the moment
- Factoring challenge: RSA-704/768/240 (2012/09/19) (closed)

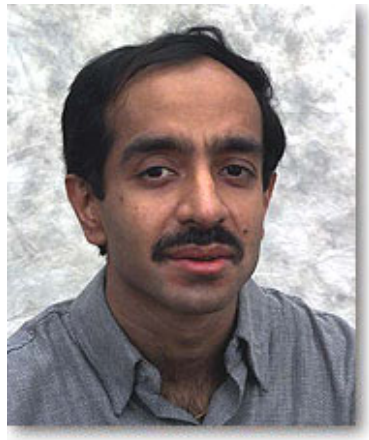
The security of the Diffie-Hellman key exchange protocol



- Used to securely exchange a private key
- Underlying security assumption: Solving DLP is hard
- If we can solve DLP fast, we can break the protocol
- No polynomial classical algorithm for DLP currently known
- But Shor's quantum algorithm breaks the protocol

Turing award for Whitfield Diffie, and Martin E. Hellman *"for inventing and promulgating both asymmetric public-key cryptography, including its application to digital signatures, and a practical cryptographic key-exchange method."* *ACM*

Grover's algorithm (1996)



- Quantum algorithm for efficient search in unsorted data
- Based on inverting a function
- Grover's algorithm uses $O(\sqrt{N})$ time (N: nbr of data items)
- Every classical algorithm runs in $O(N)$ time
- Grover's algorithm provides (provably) a quadratic speed-up
- Only algorithm with proven speed-up by quantum computers
- Algorithm does not solve NP-complete problems efficiently

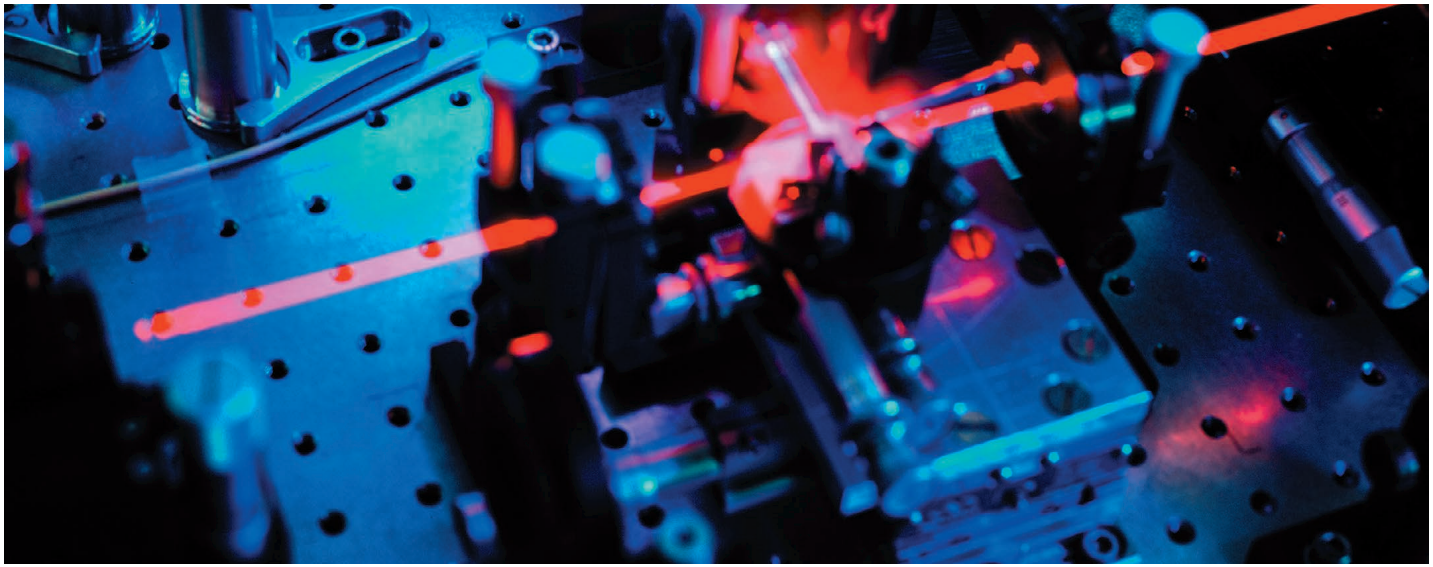
The security of blockchain algorithms (from Forbes online)

14,984 views | Sep 24, 2019, 09:50am

Google's Quantum Computing Breakthrough Brings Blockchain Resistance Into the Spotlight Again

Darryn Pollock Contributor ⓘ
Crypto & Blockchain

The security of blockchain algorithms (from Nature online)



Quantum cryptography equipment, which uses the principle of entanglement to encode data that only the sender and receiver can access.

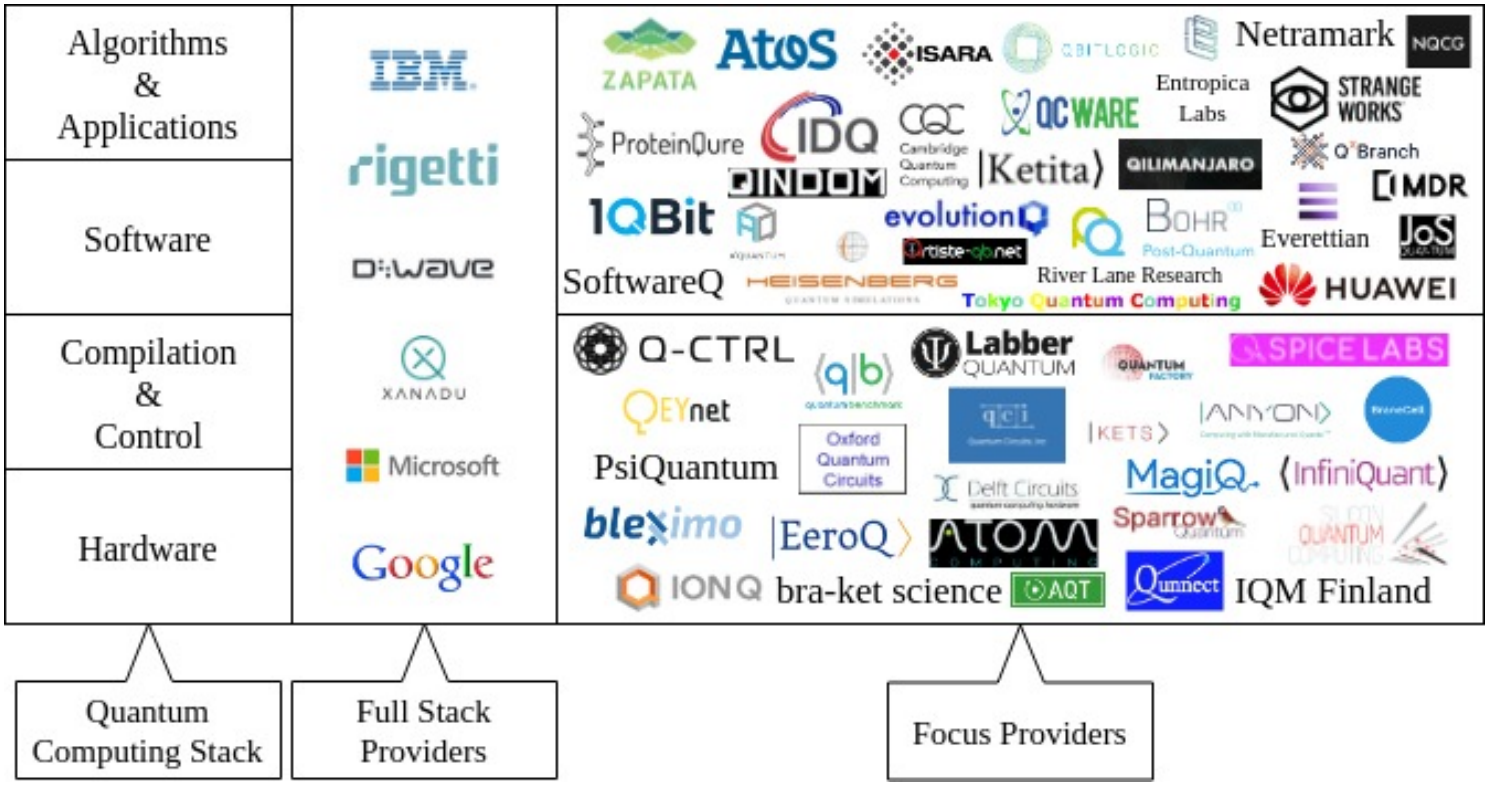
Quantum computers put blockchain security at risk

Bitcoin and other cryptocurrencies will founder unless they integrate quantum technologies, warn **Aleksey K. Fedorov**, **Evgeniy O. Kiktenko** and **Alexander I. Lvovsky**.

By 2025, up to 10% of global gross domestic product is likely to be stored on blockchains[1]. A blockchain is a digital tool that uses cryptography techniques to protect information from unauthorized changes. It lies at the root of the Bitcoin cryptocurrency[2]. Blockchain-related products are used everywhere from finance and manufacturing to health care, in a market worth more than US$150 billion.

When information is money, data security, transparency and accountability are crucial. A blockchain is a secure digital record, or ledger. It is maintained collectively by users around the globe, rather than by one central administration. Decisions such as whether to add an entry (or block) to the ledger are based on consensus — so personal trust ▶

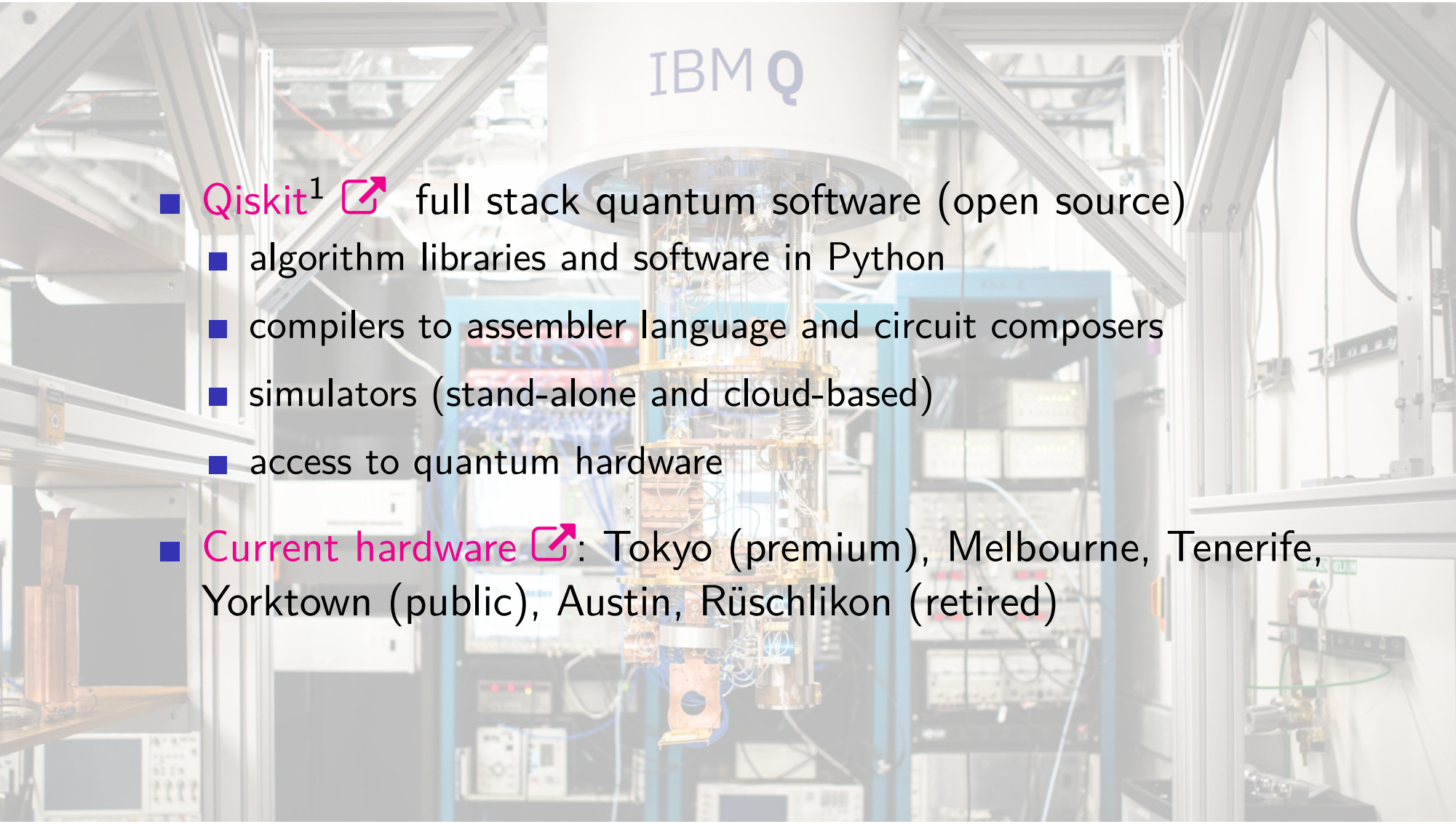
Current big (industrial) players



Example: ZAPATA: SW development for quantum computers

For a list of quantum processors, consult this Wikipedia entry

IBM's tool stack



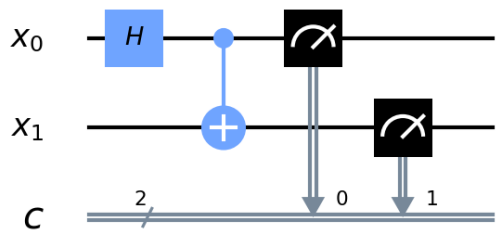
- Qiskit[1] full stack quantum software (open source)
  - algorithm libraries and software in Python
  - compilers to assembler language and circuit composers
  - simulators (stand-alone and cloud-based)
  - access to quantum hardware
- Current hardware: Tokyo (premium), Melbourne, Tenerife, Yorktown (public), Austin, Rüschlikon (retired)

[1] Quantum information science kit

Example program in IBM's Qiskit

```
from qiskit import *

qr = QuantumRegister(2, 'x')
cr = ClassicalRegister(2, 'c')
circ = QuantumCircuit(qr, cr)
circ.h(qr[0])
circ.cx(qr[0], qr[1])
circ.measure(qr, cr)
```




```
be = BasicAer.get_backend('qasm_simulator')
results = execute(circ, backend=be, shots=1024).result()
counts = results.get_counts()

print(counts)

{'11': 510, '00': 514}
```

Attention: Qubits (like x_0, x_1) are initialized with $|0\rangle$

Problems with current quantum computers

- Example circuit was run in the simulator
- Problems when circuits run on a quantum computer: Noise
- Two sources of noise:
  1. Gate infidelity: User-specified gate does not precisely correspond to the physically implemented gate
  2. Decoherence: Gradually over time, a quantum computer looses its "quantumness", e.g., due to interaction with the environment. Then it behaves more like a classical computer.
- Both effects limit the depth of quantum circuits in practice

Structure of the lecture

- Mathematical foundations
- The postulates of quantum mechanics
- Complexity classes for quantum computation
- Quantum circuits
- Quantum algorithms
  - Superdense coding and quantum teleportation
  - The algorithm of Deutsch
  - The algorithm of Deutsch and Jozsa
  - The algorithm of Bernstein and Vazirani
  - The algorithm of Grover
  - The algorithm of Simon
  - The algorithm of Shor

Information to the history of quantum computing can be found here