Quantum Scientific Machine Learning:
A path to enhanced SciML

Oleksandr Kyriienko

University of Exeter, UK

https://kyriienko.github.io/

UNIVERSITY OF EXETER

QuDOS group

- **We reside at Streatham Campus, University of Exeter (SW England)**
- UoE has over 25,000 students and Physics department with >50 members of faculty members
- **We work in quantum optics and quantum computing**

Special thanks to:

EPSRC
Engineering and Physical Sciences Research Council

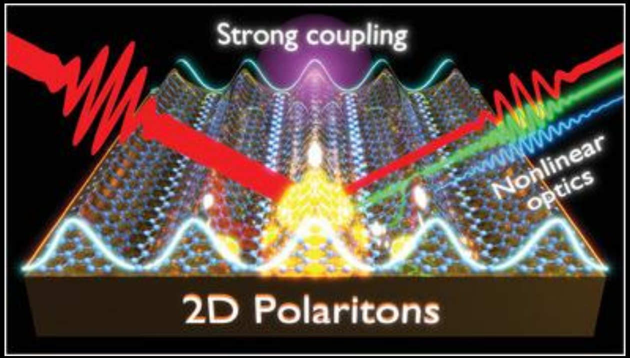
Innovate UK

NATO
OTAN

PASQAL

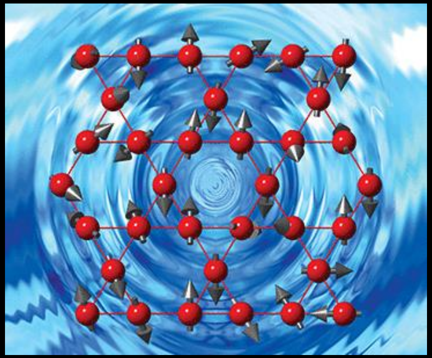
Quantum Optics, Dynamics, and Computing

My research is based on theoretical physics and targets **three streams** of quantum tech.



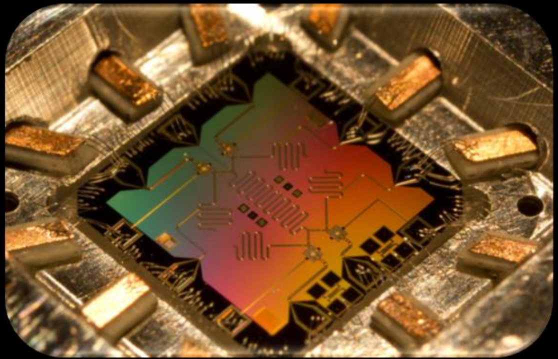
quantum optics

- Strong light-matter coupling in semiconducting materials
- 2D polaritons and nonlinear optics in flatland
- Close collaboration with experimental teams
- Establishing quantum polaritonics



quantum simulation and dynamics

- Developing fundamental understanding of quantum time crystals
- Proposing new approaches to strongly-correlated systems



quantum computing

- Exploring capabilities of modern quantum devices
- Developing efficient protocols for **near-term** quantum computers
- Finding new use-cases (chemistry, differential equations)
- Solving **open problems** in quantum machine learning

QuDOS group

Starting as a lecturer in the fall of 2019, I have shaped a team of researchers that has grown rapidly

Dr Oleksandr Kyriienko (PI)

Ms Chelsea Williams (PhD student w/ Pasqal)

Ms Annie Paine (PhD student w/ Qu&Co-Pasqal)

Mr Chiddy Umeano (PhD student; DTP)

Dr Kok Wee Song (Postdoc)

Mr Salvatore Chiavazzo (PhD student)

- One more postdoc in QC starting soon

- Two postdoctoral positions are available (recruiting):

  (1) QC for topological data analysis

  (2) quantum optics in Moire structures

We are a small but rapidly growing team, looking forward to establishing collaborations.

Quantum SciML

1. **Basic blocks of quantum computing**
    - qubits, superposition
    - interference
    - quantum mechanics rulebook

2. **Quantum gates and algorithms**
    - single-qubit gates
    - two-qubit gates
    - scaling
    - complexity
    - full-stack

3. **Quantum hardware requirement and state-of-the-art**
    - superconducting circuits
    - trapped ions
    - Rydberg atoms
    - photonics

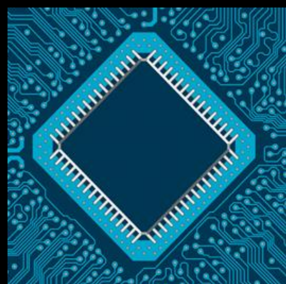
4. **Promising applications areas of quantum computing**
    - chemistry
    - materials
    - PDEs
    - optimisation
    - finances
    - machine learning

5. **Quantum Machine Learning and Scientific Computing**
    - linear system solvers
    - challenges an overheads
    - near-term solutions
    - differentiable circuits

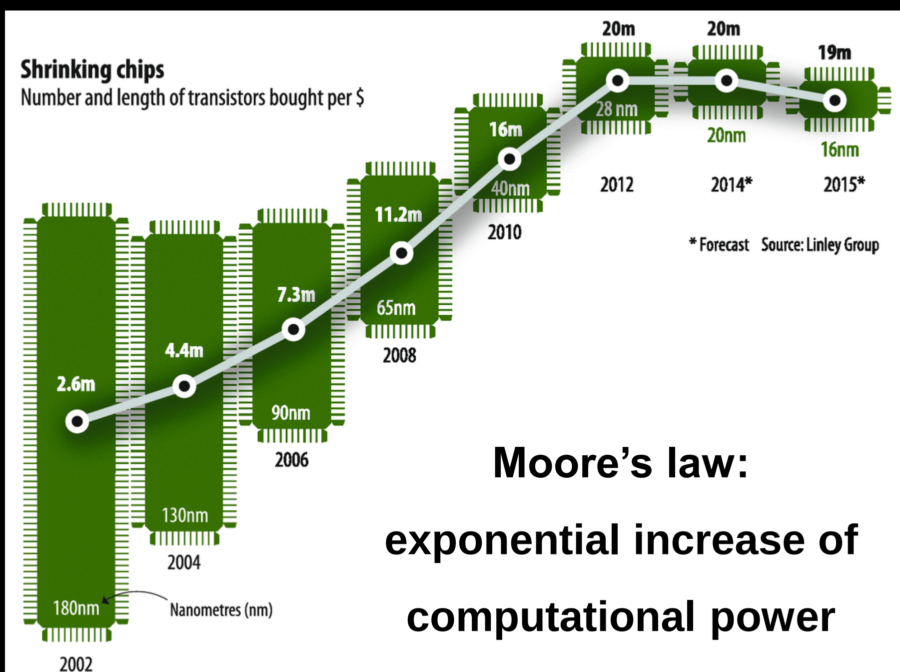
Classical computing

encode information in a binary form:
0 & 1 bits → bit strings → gates

0



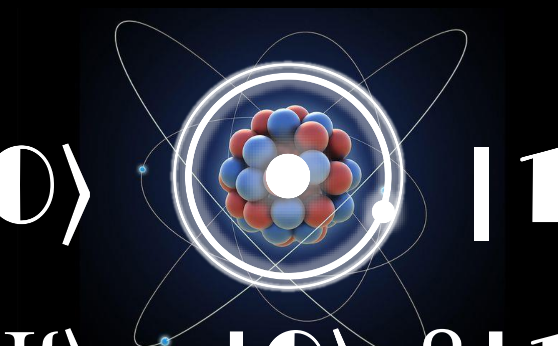
1



Moore's law:

exponential increase of computational power

Quantum computing



$|0\rangle$ $|1\rangle$

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

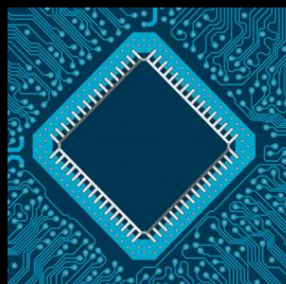
two-level atom = qubit

a qubit: quantum superposition

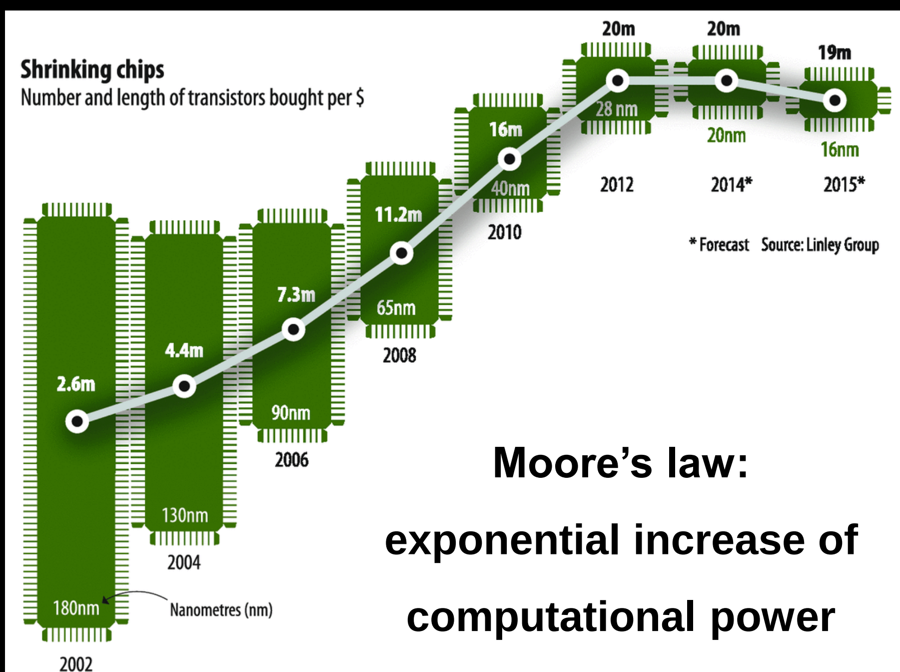
Classical computing

encode information in a binary form:
0 & 1 bits → bit strings → gates

0



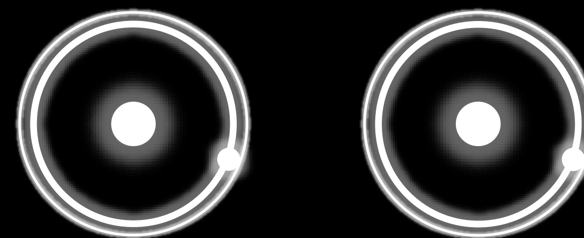
1



Moore's law:

exponential increase of computational power

Quantum computing



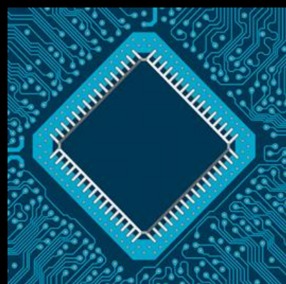
$$|\Psi\rangle = \alpha|00\rangle + \beta|10\rangle + \gamma|01\rangle + \delta|11\rangle$$

two qubits: quantum entanglement

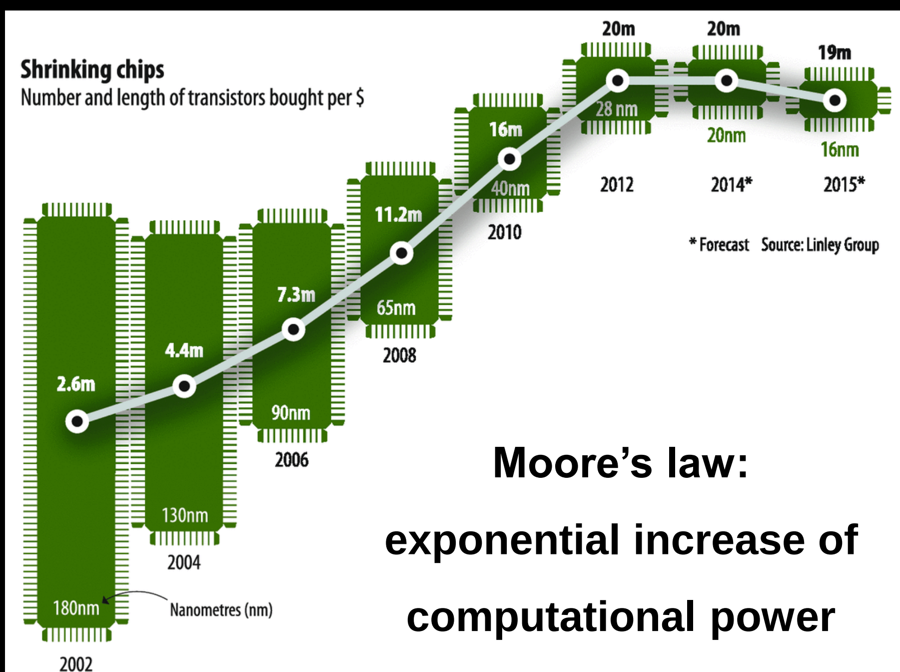
Classical computing

encode information in a binary form:
0 & 1 bits → bit strings → gates

0



1

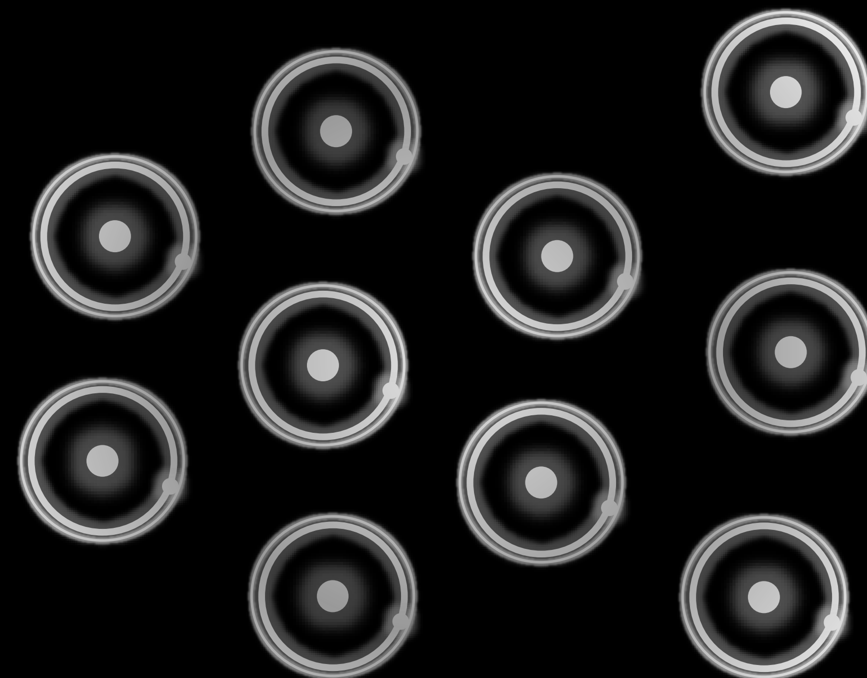


Moore's law:

exponential increase of

computational power

Quantum computing



N qubits = 2^N possible states

For N = 60 we have ~ 10^{18} states

and need 16 exabytes to store it!

Quantum mechanics

Quantum evolution: $e^{-it\hat{H}} \cdot |\psi\rangle$

propagator

quantum state

quantum computing = (restricted) matrix mechanics

Quantum mechanics

More formally, quantum mechanics relies on system's state being propagated in time following the **Schrödinger equation**:

$$i\hbar \frac{d}{dt}|\Psi(t)\rangle = \hat{H}\,|\Psi(t)\rangle$$

time-dependent state

Hamiltonian ($\hat{H} = \hat{H}^{\dagger}$)

Erwin Rudolf Josef Alexander Schrödinger

This differential equation is **linear** and can solved formally as

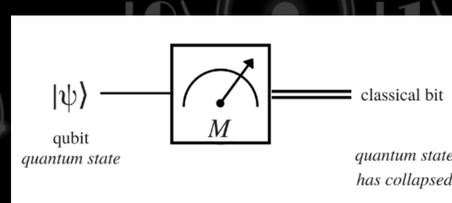
$$\frac{d}{dt}|\Psi(t)\rangle = -\frac{i}{\hbar}\hat{H}\,|\Psi(t)\rangle \quad \Rightarrow \quad |\Psi(t)\rangle = \underbrace{e^{-\frac{i}{\hbar}\hat{H}t}}_{\text{propagator}}|\Psi(0)\rangle$$

final time

initial time

Quantum operations rely on **coherent unitary evolution.** However, this shall be followed by **measurement** – non-unitary collapse to one of the states according to the **Born rule**:

$$\hat{\mathcal{M}} : |\Psi\rangle \mapsto \frac{\hat{\mathcal{M}}|\Psi\rangle}{\sqrt{\langle\Psi|\hat{\mathcal{M}}^{\dagger}\hat{\mathcal{M}}|\Psi\rangle}}$$



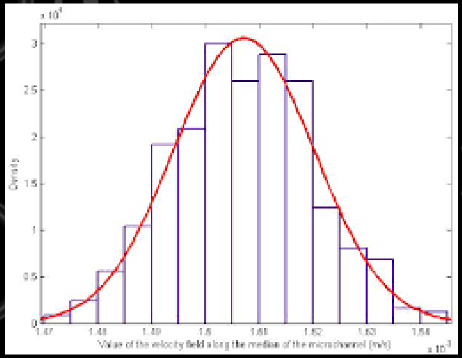
$$|\Psi\rangle \longrightarrow \lambda_k$$

$$p_k = |\langle\Psi|k\rangle|^2$$

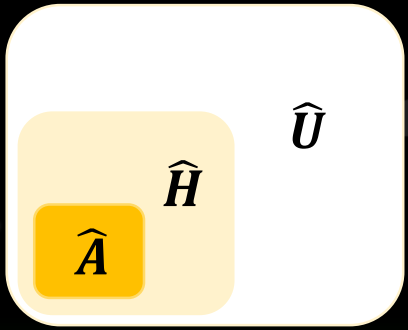
Quantum mechanics rulebook

There are several **important implications** coming from **unitarity of quantum mechanics, non-commutativity** of operators, and **destructive measurement process.**

- **Quantum states cannot be cloned** as a consequence of linearity of QM (**no-cloning theorem** = no FAN-OUT).



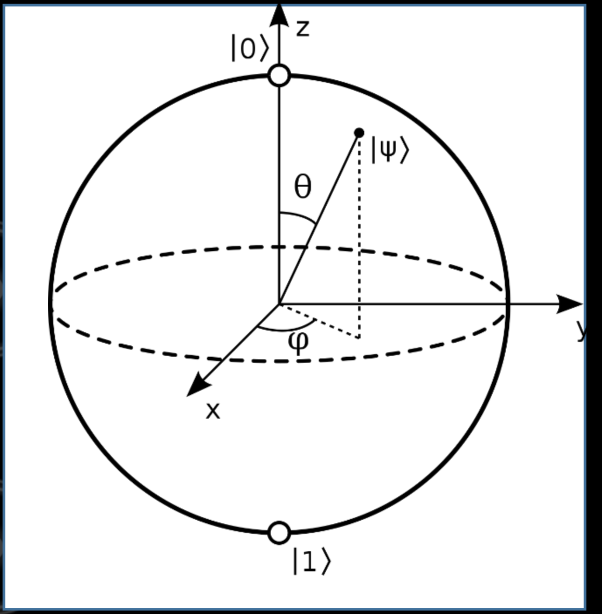
- Measurement gives only one classical bit (shot) of the state (collapsed wave function). Many shots are needed to read out observables (**probabilistic operation**).
- Quantum amplitudes cannot be accessed easily, unless a full tomography is performed.
- Coherent correction of quantum errors is difficult as we cannot use simple repetition, and require special **quantum error correction** techniques.
- Performing **non-unitary** operations requires **ancillary qubits**



The development of efficient algorithms relies on minimizing losses due to these restrictions.

Quantum gates

We can picture the state of qubit using the **Bloch sphere**, and **decompose** each qubit operation as a sum of **Pauli matrices** $\vec{\boldsymbol{\sigma}} = (\hat{X}, \hat{Y}, \hat{Z})$ and **identity matrix**: SU(2) rotation



evolution phase **Bloch sphere angles**

$$e^{i\frac{\theta}{2}(\vec{n}\cdot\vec{\sigma})} = \mathbb{1}\cos\left(\frac{\theta}{2}\right) + i(\vec{n}\cdot\vec{\sigma})\sin\left(\frac{\theta}{2}\right)$$

single qubit unitary operator and parametrized state

$$|\psi(\theta,\phi)\rangle = \cos\left(\theta/2\right)|0\rangle + e^{i\phi}\sin\left(\theta/2\right)|1\rangle$$

What are the **operations** we want to perform? Here we are motivated by **computational science** and classical gate set, as well as arbitrary rotations that map $(\theta, \varphi) \rightarrow |\psi_{\theta,\varphi}\rangle$:

Complete set:

$$\hat{X} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \hat{Y} = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad \hat{Z} = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \quad \hat{\mathbb{1}} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

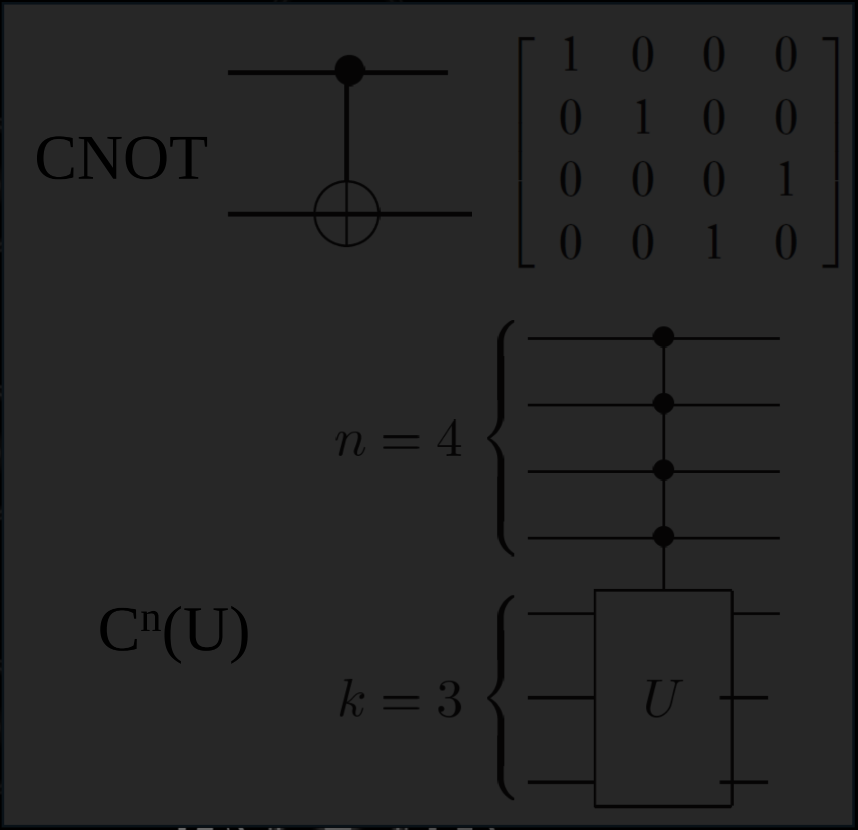
$$\hat{Y} = i\hat{X}\hat{Z}$$

Pauli matrices and identity ("do nothing")

Quantum gates

Quantum gates: evolution of certain **one**-qubit, two-qubit, three-qubit Hamiltonian for fixed time t. They **e**xploit algebra of **Pauli operators** and their tensor products

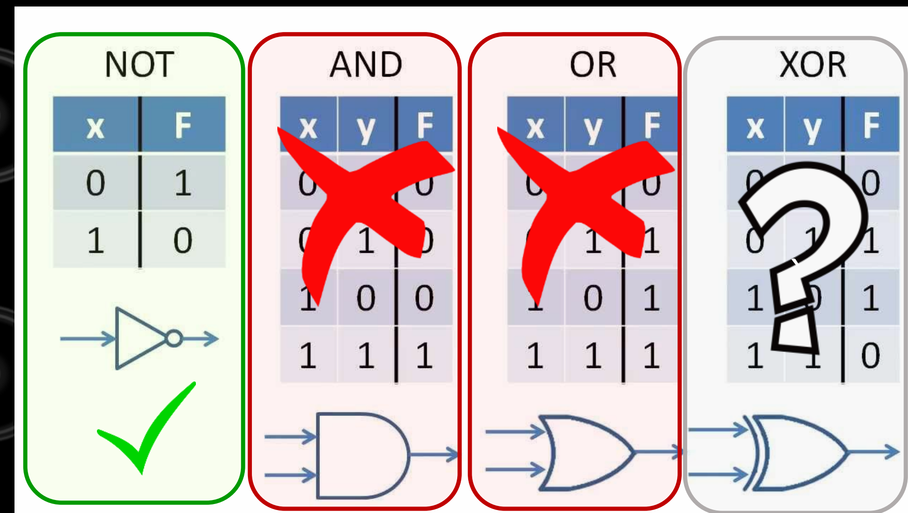
Hadamard	H	$\frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix}$
Pauli-X	X	$\begin{bmatrix}0 & 1\\ 1 & 0\end{bmatrix}$
Pauli-Y	Y	$\begin{bmatrix}0 & -i\\ i & 0\end{bmatrix}$
Pauli-Z	Z	$\begin{bmatrix}1 & 0\\ 0 & -1\end{bmatrix}$
Phase	S	$\begin{bmatrix}1 & 0\\ 0 & i\end{bmatrix}$
$\pi/8$	T	$\begin{bmatrix}1 & 0\\ 0 & e^{i\pi/4}\end{bmatrix}$



[textbook: Nielsen&Chuang “Quantum Computation and Quantum Information” and qiskit-textbook by IBM] https://qiskit.org/textbook/what-is-quantum.html

Quantum gates

For introducing 2-qubit gates, we can establish the analogy with classical gates



logical gates

state definitions (no superposition)

$$|0\rangle \stackrel{\text{def}}{=} \begin{pmatrix} 1 \\ 0 \end{pmatrix} \qquad |1\rangle \stackrel{\text{def}}{=} \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$\mathrm{NOT} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \Leftrightarrow \hat{X} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

logical negation can be performed on a quantum computer

Classical **FAN-IN operation** corresponds to tracking only one **target output**, and using another as a **control**. We can compose corresponding 2q operators from truth tables, considering **AND** gate as an example

x	y	F
0	0	0

$$|00\rangle \rightarrow |00\rangle$$

x	y	F
0	1	0

$$|01\rangle \rightarrow |00\rangle$$

x	y	F
1	0	0

$$|10\rangle \rightarrow |10\rangle$$

x	y	F
0	1	0

$$|11\rangle \rightarrow |11\rangle$$

$$\mathrm{AND} = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

non-unitary gate (non-reversible)

Quantum mechanics

We can write XOR operator observing that is corresponds to a **permutation matrix**:

x	y	F
0	0	0

$|00\rangle \rightarrow |00\rangle$

x	y	F
0	1	1

$|01\rangle \rightarrow |01\rangle$

x	y	F
1	0	1

$|10\rangle \rightarrow |11\rangle$

x	y	F
1	1	0

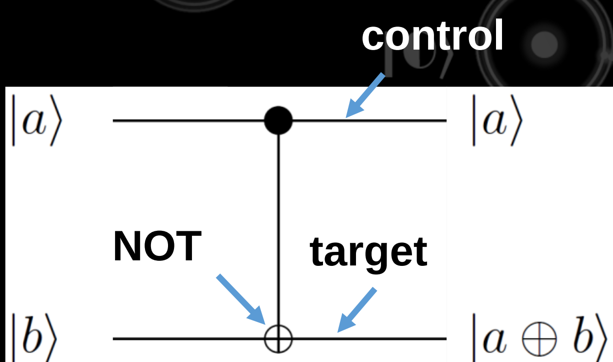
$|11\rangle \rightarrow |10\rangle$

$$\mathrm{XOR} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

unitary and **reversible**

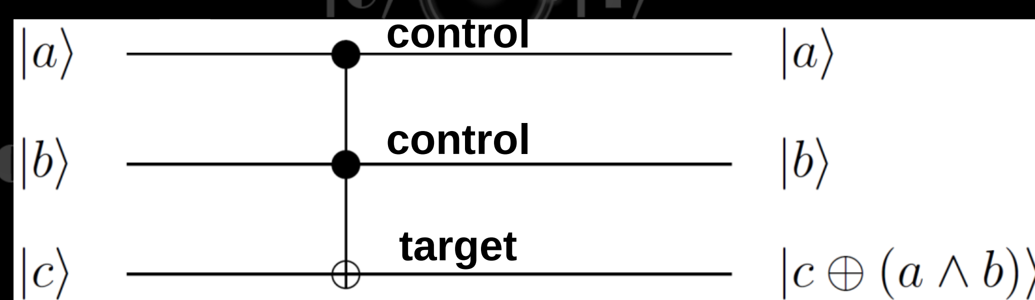
Implemented as a quantum gate, and it is called a **controlled-NOT** (or **CNOT**) operation, that applies **negation** to the **target qubit** depending on the state of the **control qubit**:

$$\mathrm{CNOT} = |00\rangle\langle 00| + |01\rangle\langle 01| + |10\rangle\langle 11| + |11\rangle\langle 10|$$



CNOT gate

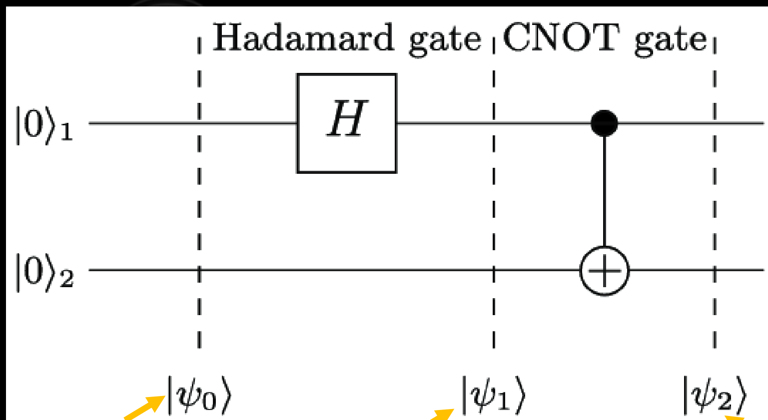
Generalize to **3-qubits**:



Toffoli gate (c-c-NOT) – reversible classical/quantum gate

Quantum mechanics

We can now compose our **first quantum programme**, which creates an **entangled** state.



$$|00\rangle$$

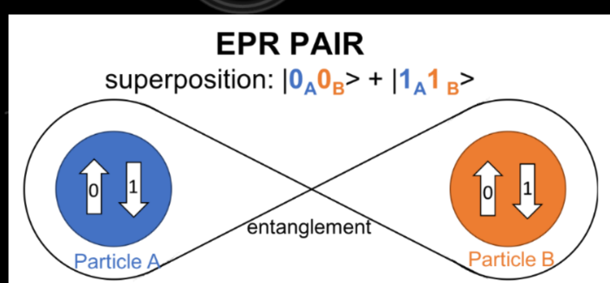
$$(\hat{\mathrm{H}}|0\rangle) \otimes |0\rangle = \frac{1}{\sqrt{2}}\Big(|00\rangle + |10\rangle\Big)$$

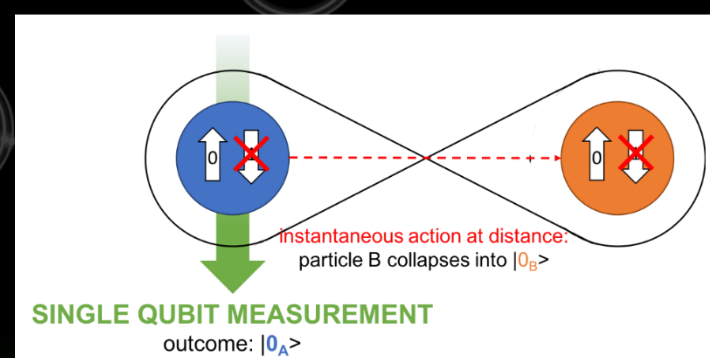
$$\mathrm{CNOT}\Big(\frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|10\rangle\Big) = \frac{1}{\sqrt{2}}\Big(|00\rangle + |11\rangle\Big) = |\Phi^+\rangle$$

Using a single qubit **Hadamard** gate and a **CNOT** gate we can prepare a maximally **entangled Bell state** from trivial zero state.

We can also reverse the circuit to measure if we are in the specific Bell state (**Bell measurement**).

Entanglement allows for "**action-at-distance**",* as outcomes of distant qubits are correlated.





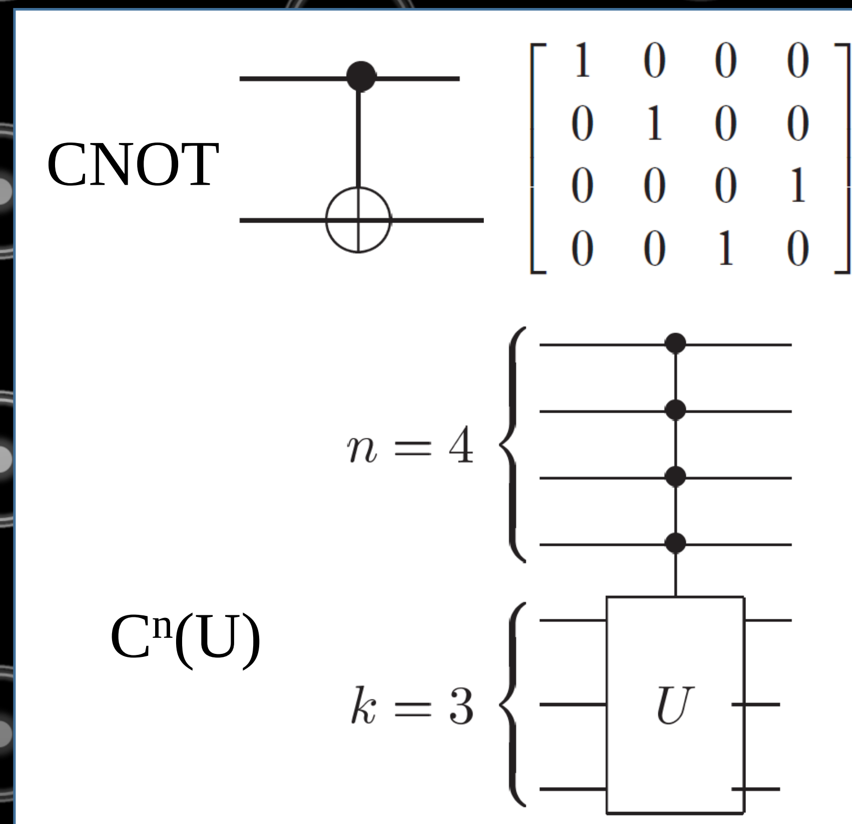
**Relativity is preserved by non-cloning.*

Quantum gates

Quantum gates: evolution of certain one, two, three-qubit Hamiltonian for fixed time **t**

Exploit algebra of **Pauli operators** and their tensor products

Hadamard	$-\boxed{H}-$	$\frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix}$
Pauli-X	$-\boxed{X}-$	$\begin{bmatrix}0 & 1\\ 1 & 0\end{bmatrix}$
Pauli-Y	$-\boxed{Y}-$	$\begin{bmatrix}0 & -i\\ i & 0\end{bmatrix}$
Pauli-Z	$-\boxed{Z}-$	$\begin{bmatrix}1 & 0\\ 0 & -1\end{bmatrix}$
Phase	$-\boxed{S}-$	$\begin{bmatrix}1 & 0\\ 0 & i\end{bmatrix}$
$\pi/8$	$-\boxed{T}-$	$\begin{bmatrix}1 & 0\\ 0 & e^{i\pi/4}\end{bmatrix}$



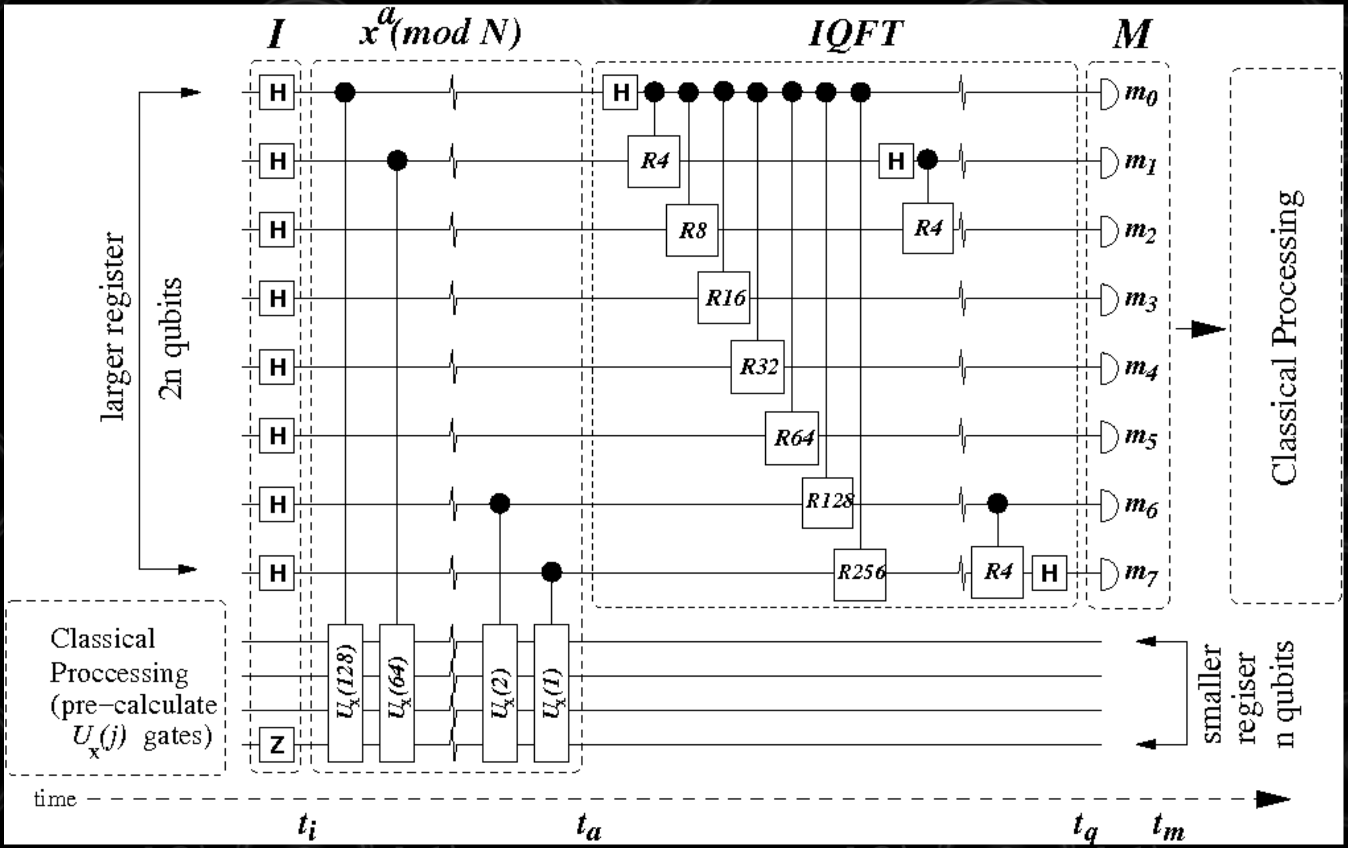
[textbook: Nielsen&Chuang “Quantum Computation and Quantum Information” and qiskit-textbook by IBM] https://qiskit.org/textbook/what-is-quantum.html

Quantum prime factoring

Shor's algorithm (1994): use quantum computer to speed up factoring by using greatest common divider and **quantum Fourier transform** as subroutines

factoring:

15 = 5 × 3



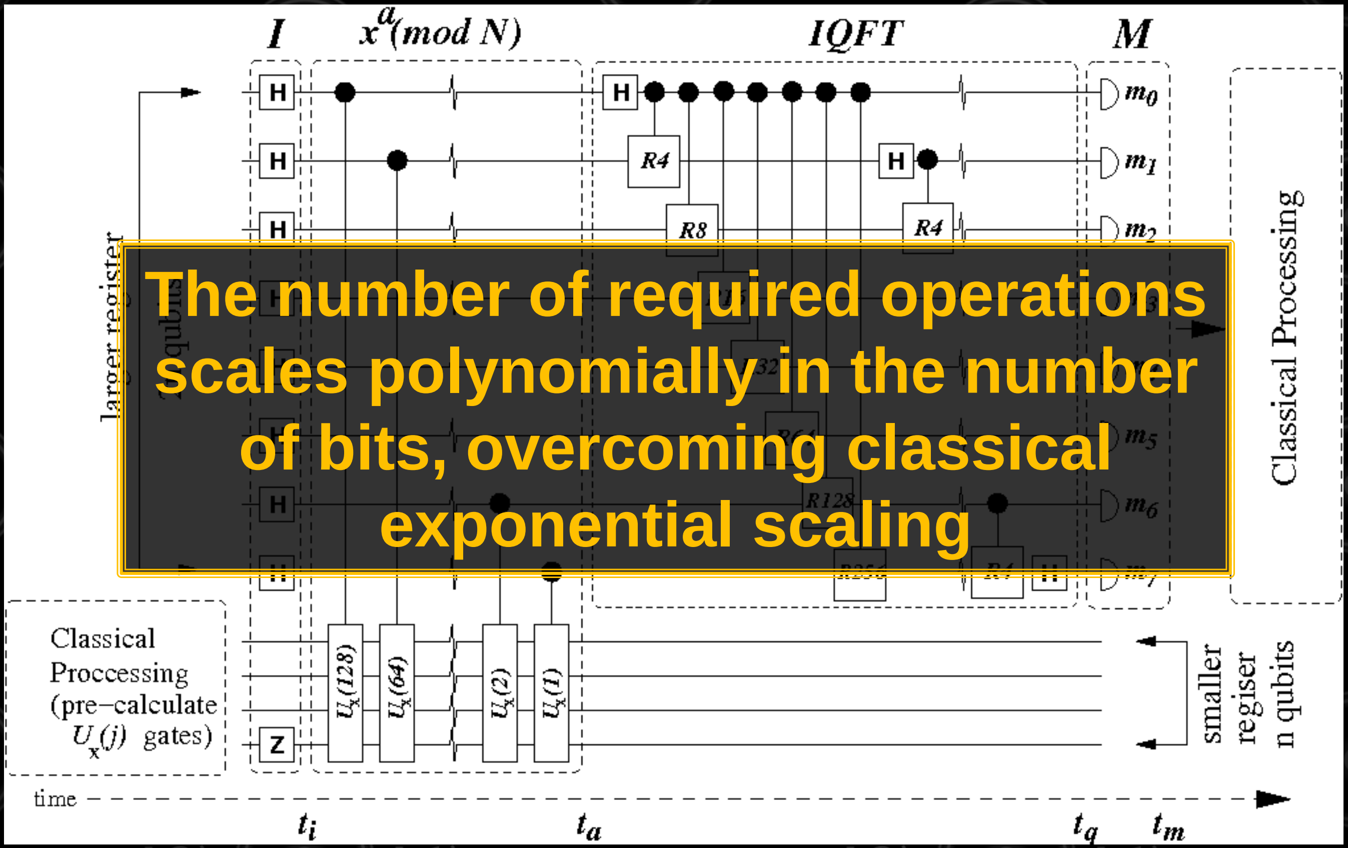
The power of QC comes from constructive and destructive **interference**.

Quantum prime factoring

Shor's algorithm (1994): use quantum computer to speed up factoring by using greatest common divider and **quantum Fourier transform** as subroutines

factoring:

15 = 5 × 3

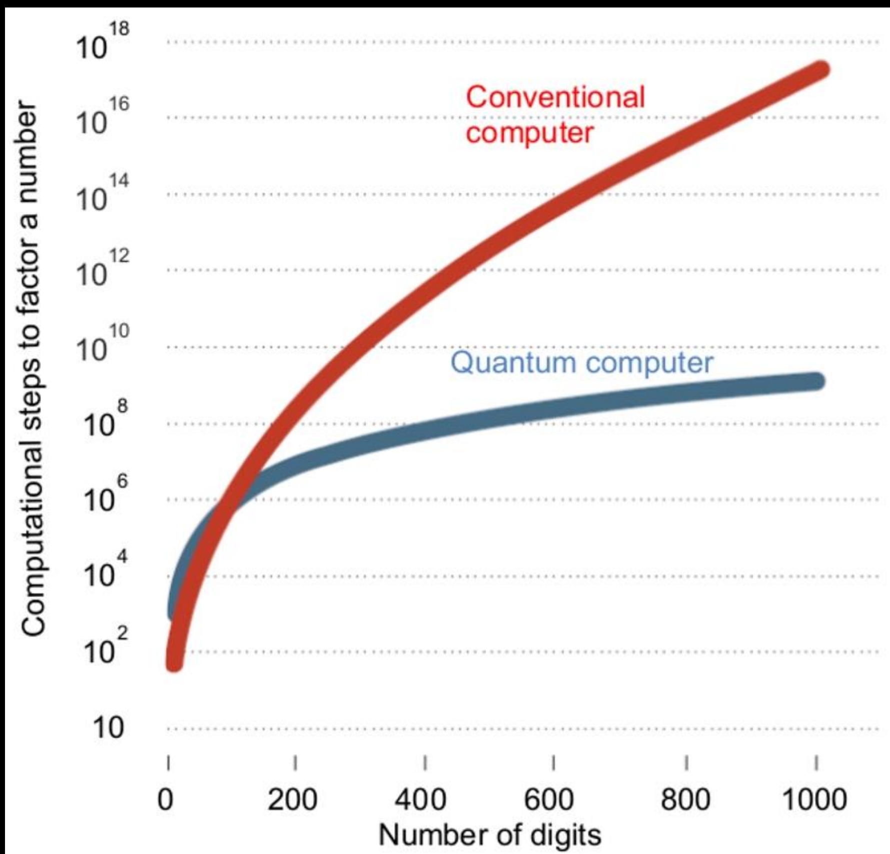


The number of required operations scales polynomially in the number of bits, overcoming classical exponential scaling

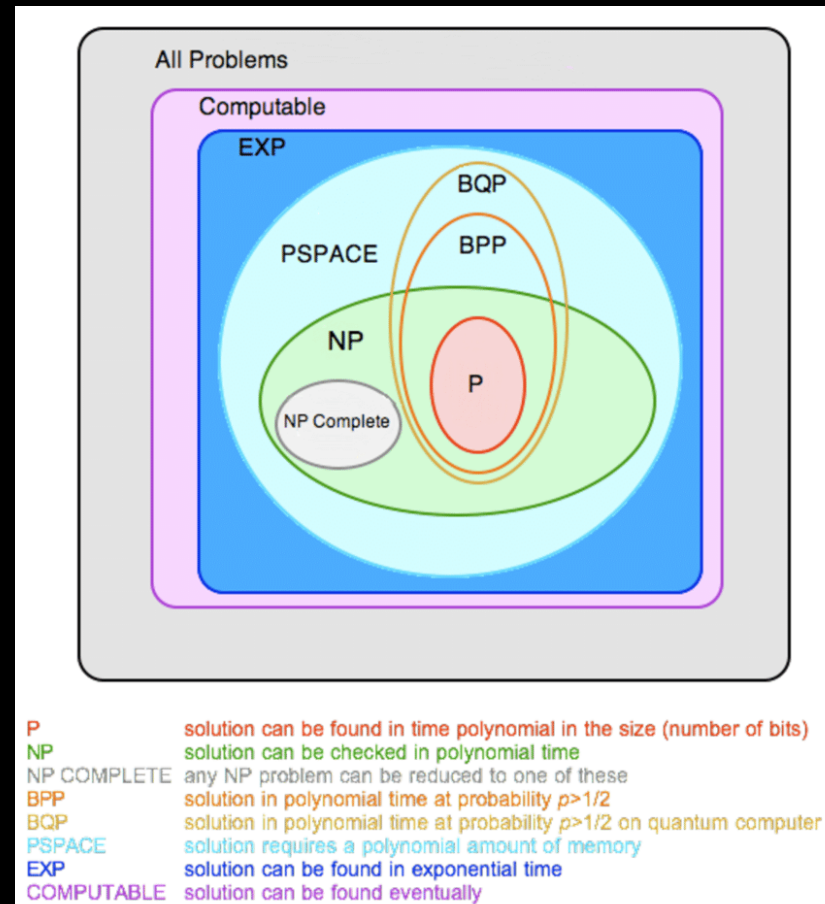
The power of QC comes from constructive and destructive **interference**.

Quantum speed-up

By using distinct quantum operation principles we can develop algorithms that may have a scaling that is qualitatively better than classical.



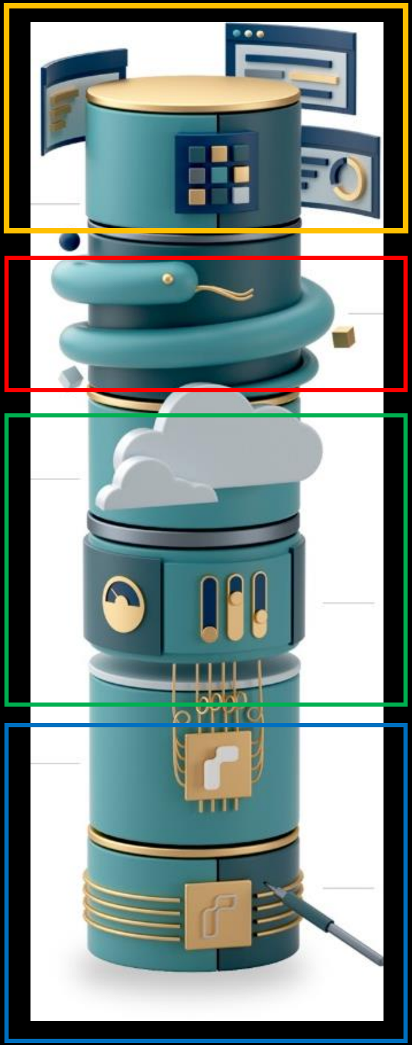
distinct **algorithmic scaling** (quantum vs classical complexity) and software-based advantage



complexity classes and **BQP** as a prime quantum contender

Full quantum stack

To perform quantum computing we need a full-stack that supports efficient algorithms.



algorithms
(quantum software)

OS and instruction languages (modules in Python/Julia/C#)

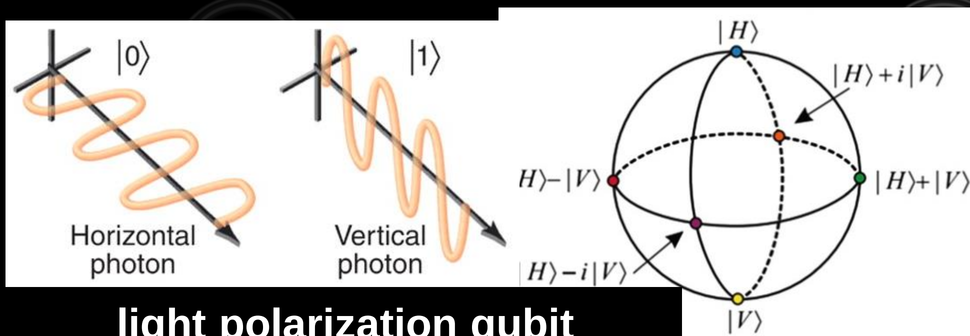
control and pulse engineering
(low/high level)

quantum hardware (QPU)

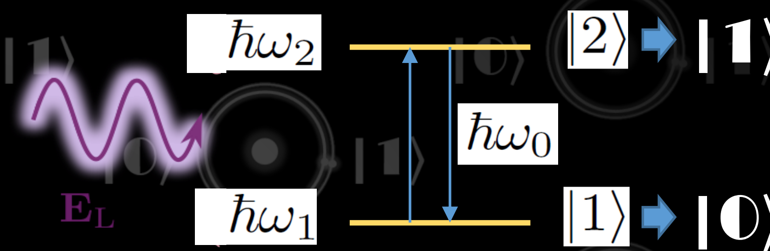
- ✓ **algorithmic scaling** is important
- ✓ **absolute running time** is important
- ✓ **prefactors** are important
- ✓ **hardware capabilities** are important
- ✓ **engineering and classical "bottlenecks"** are important
- ✓ **noise properties of devices** are important
- ✓ **correcting errors** is important

Quantum hardware

There are many different ways to create **a qubit**.



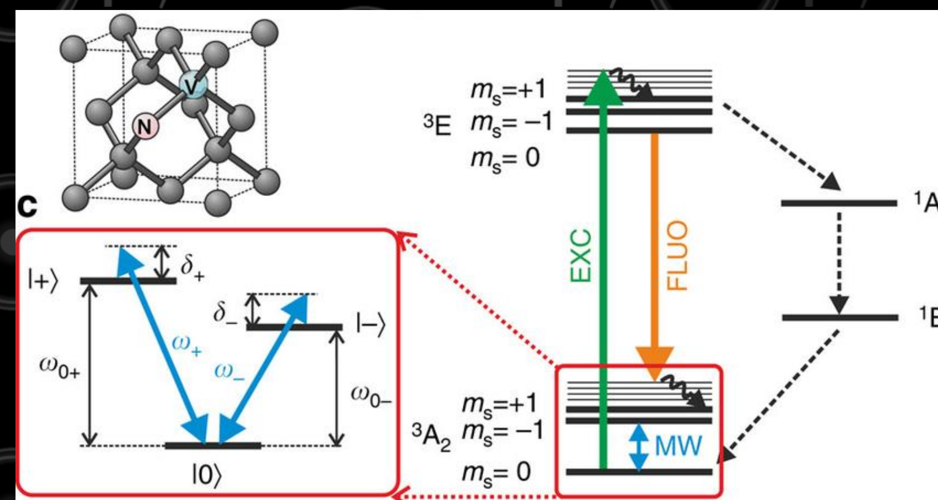
light polarization qubit



idealization: two-level system

Physically qubits can be represented by:

- **two atomic states**
- **spin-1/2** particles
- **nonlinear bosonic** modes
- two **polarizations** of **light**
- single **photon occupation** (0/1)
- photon location (dual-rail encoding)
- vacancy centres
- topological excitations
- and many others.

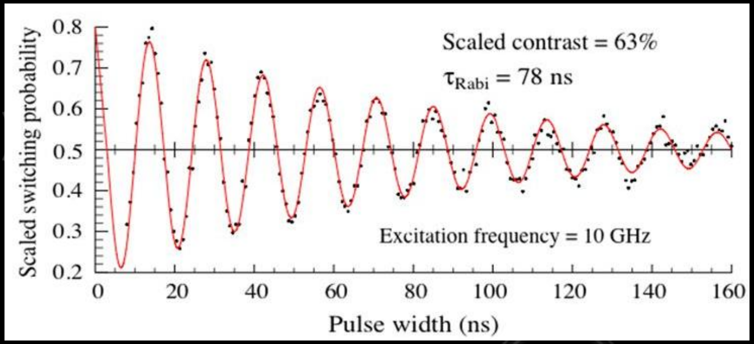


reality: defect in diamond – NV centre

Next step: we need to make sure that our platform is **fast, scalable, error-prone** and satisfies **certain criteria**.

Quantum hardware

Requirements: long coherence time, ability to perform **unitary operations** on single qubits (rotations), and have ***at least one*** **two-qubit entangling operation**.

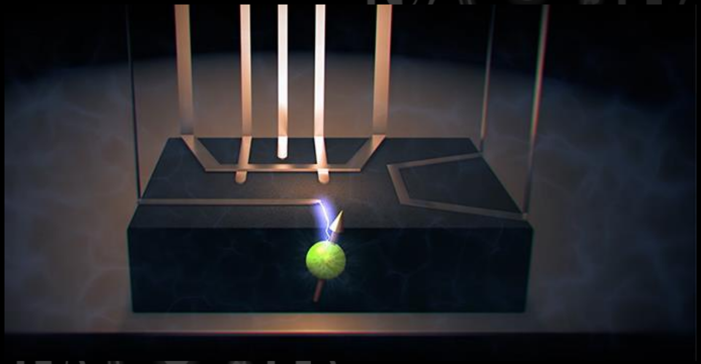


Rabi oscillations: decay

DiVincenzo criteria for QIP (2000):

1. A **scalable** physical system with a well characterized qubit
2. The ability to initialize qubits to a simple fiducial state
3. **Long** relative **decoherence times**
4. A "universal" set of **quantum gates**
5. A qubit-specific **measurement** capability

6-7. Ability to **transmit information** (quantum communication)



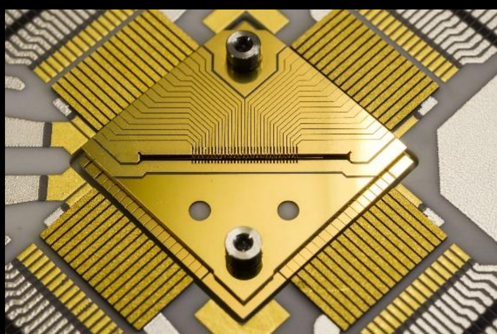
nuclear spin: protected from environment (up to ~1h), but also from other qubits(!)

- ✓ Nowadays we come to understanding that coherence time by itself is not important – it is the **ratio of interaction strength to decay** that defines **high gate fidelity**.
- ✓ **Absolute values** for **operation rates** are crucial – additional consideration when choosing and developing the quantum information processing platform.

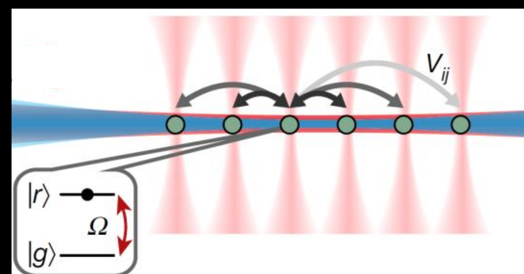
Quantum hardware

Physical systems with large coherence times and fast control are required:

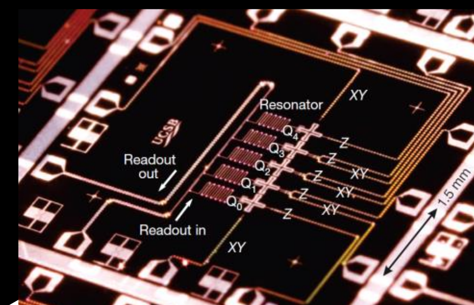
trapped ions



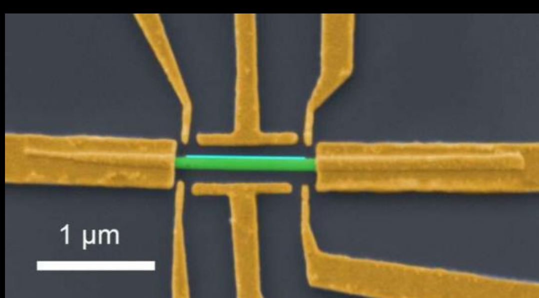
Rydberg atoms



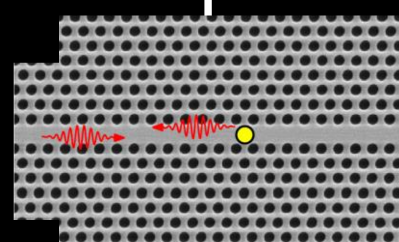
superconducting
circuits



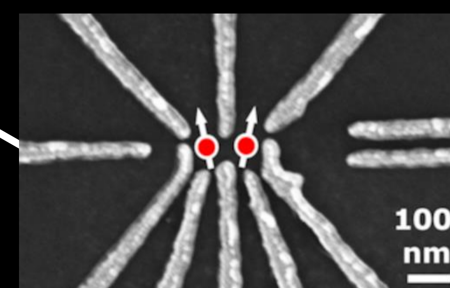
platforms



Majorana qubits[?]



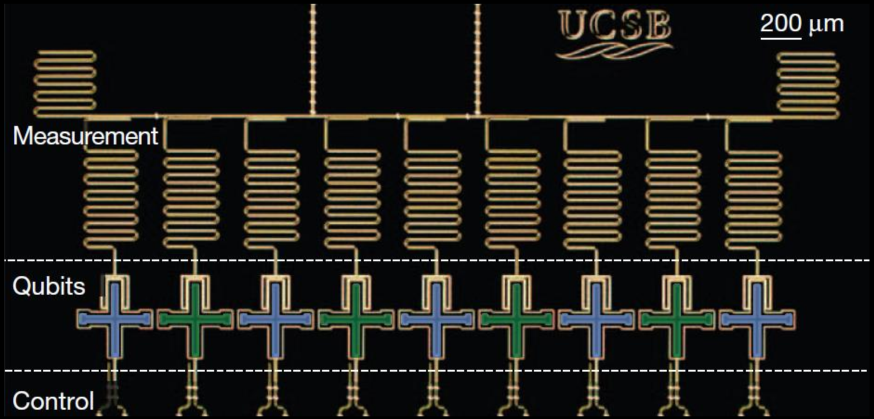
nonlinear photonics



spin-based QD
qubits

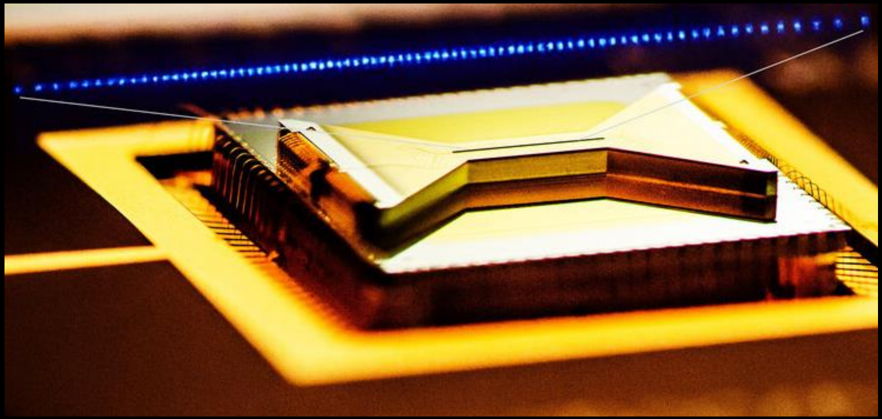
Quantum hardware

Superconducting circuits and trapped ions are currently most advanced platforms in terms of individual qubit control, gate fidelities, and scalability.



superconducting qubits as nonlinear LC resonators

- GHz operation frequencies, ~10-100ns gate times, MHz measurement rates
- relatively easy fabrication (lithography)
- low operation temperatures, need dilution fridge (30mK)
- susceptible to disorder, defect-based noise and cosmic rays

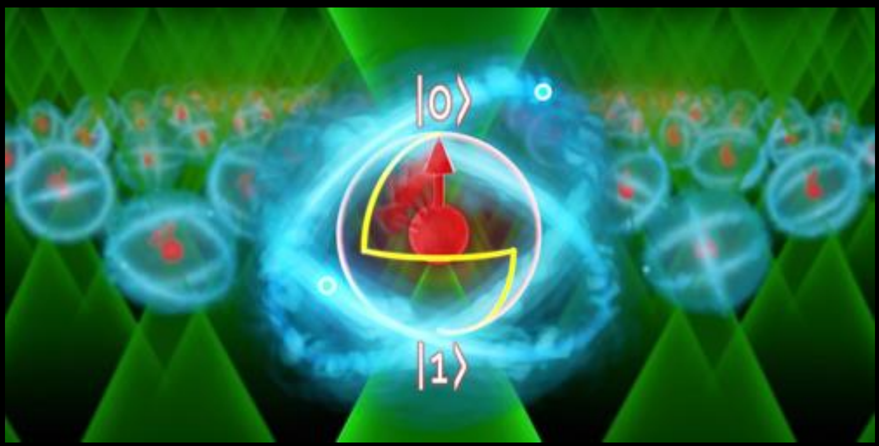


ions trapped between RF electrodes and coupled through shared trap potential

- MHz operation frequencies, ~1-10μs gate times, KHz measurement rates
- high connectivity inside a trap
- long qubit lifetime and slow absolute clock rates
- scalability is limited to ~50q for now, need photonic links to scale further

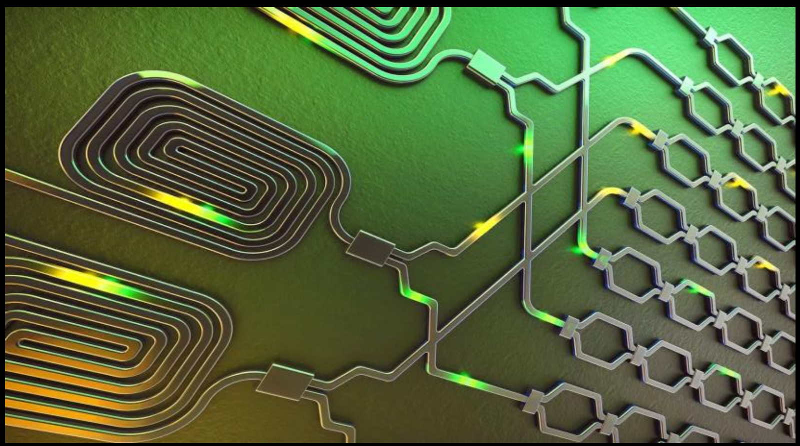
Quantum hardware

The race for the best quantum platform is far from being over, and various other technologies may advance for being the leader in the long term.



Rydberg atom arrays as reconfigurable qubit lattices

- microwave/optical addressing, 100μs-1ms coherence times, ~1μs gate times, 200ms for the full sequence
- reconfigurable (1D, 2D, 3D) and straightforwardly scalable to ~1000 qubits
- relatively high operation temperatures
- difficult to achieve individual addressability – suited to simulation



photonic qubits as quantum of light interfered and coupled through medium

- ultrafast operation at 1-10ps times, fast measurement
- long propagation length and pathways towards distributed operation
- limited nonlinearity and thus low fidelities
- scalability depends on possible realisations (low TRL for computing), and may be suitable for long-term QC

Quantum hardware: summary

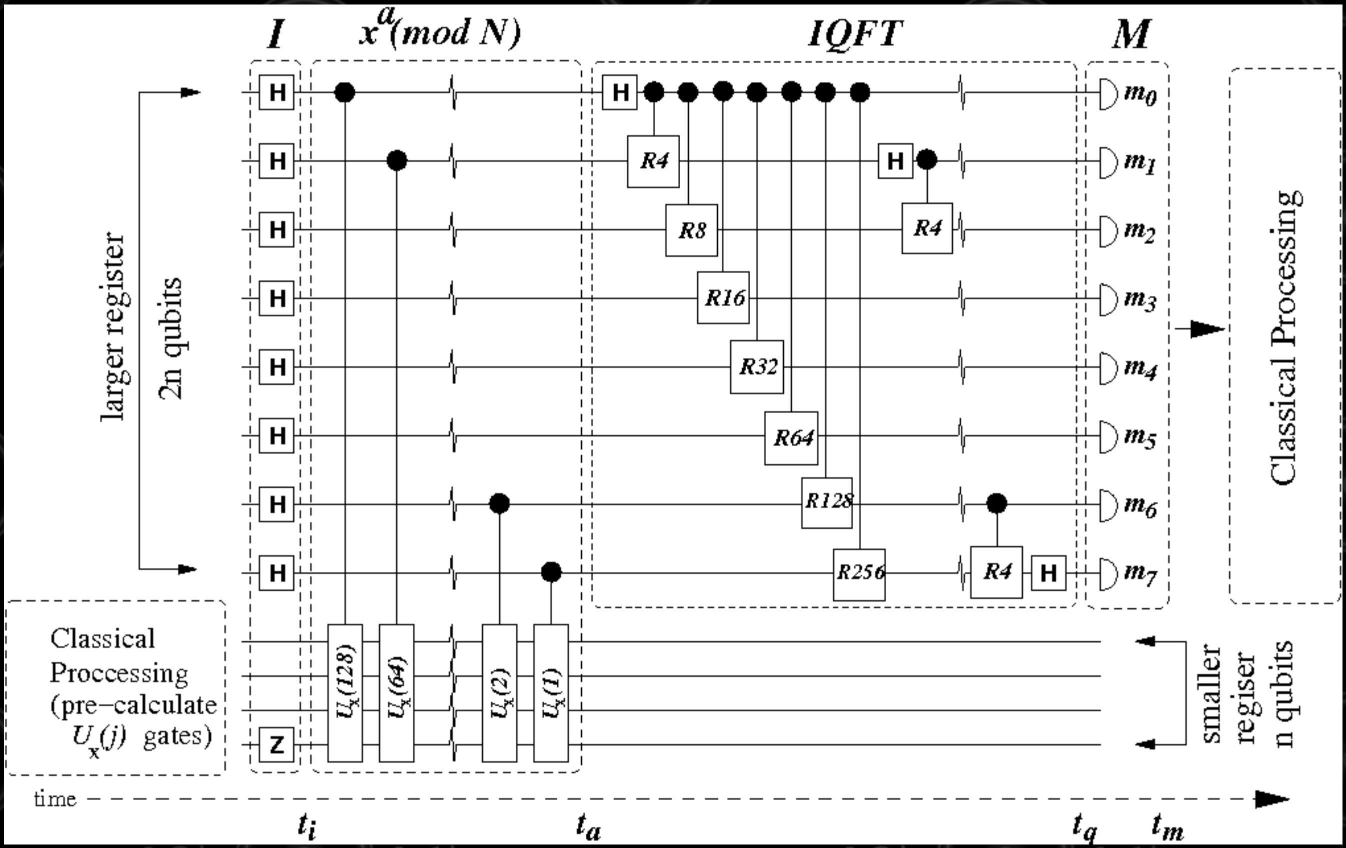
Qubit type or technology	Superconducting[2]	Trapped ion	Photonic	Silicon-based[3]	Topological[8]
Description of qubit encoding	Two-level system of a superconducting circuit	Electron spin direction of ionized atoms in vacuum	Occupation of a waveguide pair of single photons	Nuclear or electron spin or charge of doped P atoms in Si	Majorana particles in a nanowire
Physical qubits[4,5]	IBM: 20, Rigetti: 19, Alibaba: 11, Google: 9	Lab environment: AQT[6]: 20, IonQ: 14	6×3[9]	2	target: 1 in 2018
Qubit lifetime	~50–100 μs	~50 s	~150 μs	~1–10 s	target ~100 s
Gate fidelity[7]	~99.4%	~99.9%	~98%	~90%	target ~99.9999%
Gate operation time	~10–50 ns	~3-50 μs	~1 ns	~1–10 ns	–
Connectivity	Nearest neighbors	All-to-all	To be demonstrated	Nearest neighbor	–
Scalability	No major road-blocks near-term	Scaling beyond one trap (>50 qb)	Single photon sources and detection	Novel technology potentially high scalability	?
Maturity or technology readiness level	TRL[10] 5	TRL 4	TRL 3	TRL 3	TRL 1

Quantum prime factoring

Shor's algorithm (1994): use quantum computer to speed up factoring by using greatest common divider and quantum Fourier transform as subroutines

factoring:

15 = 5 × 3



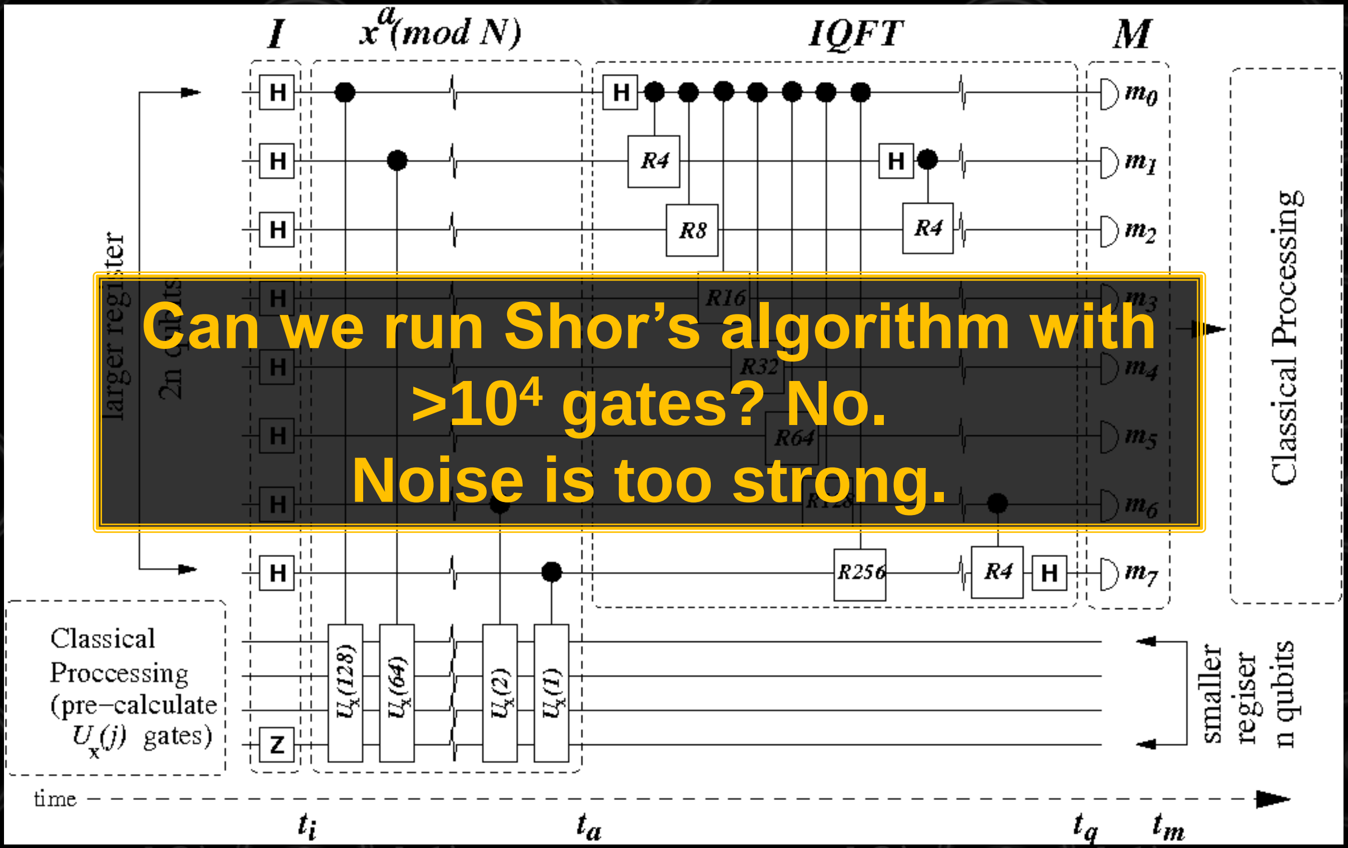
The power of QC comes from constructive and destructive **interference**.

Quantum prime factoring

Shor's algorithm (1994): use quantum computer to speed up factoring by using greatest common divider and quantum Fourier transform as subroutines

factoring:

15 = 5 × 3



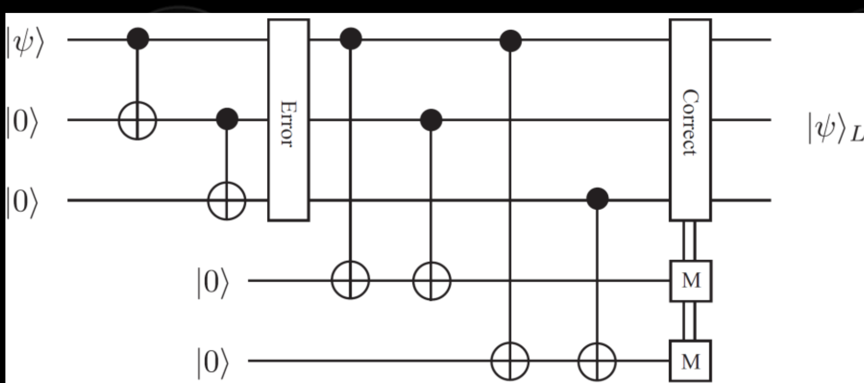
Can we run Shor's algorithm with >10^4 gates? No. Noise is too strong.

The power of QC comes from constructive and destructive **interference**.

Quantum error correction

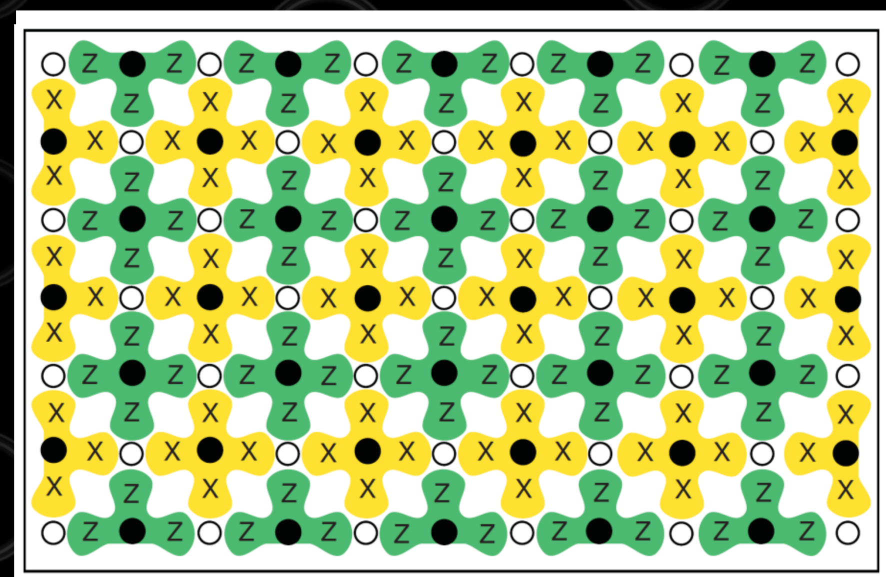
- **Quantum information** is **prone to errors** – if we (environment) have learnt something about the state, the quantum information is gone.
- So does also the **measurement**.
- Quantum information **cannot be cloned** (remember: no cloning).
- However, we can still measure the **parity** of the system.

$|0\rangle \rightarrow |0_L\rangle \equiv |000\rangle$ $\quad$ $|1\rangle \rightarrow |1_L\rangle \equiv |111\rangle$



simple repetition code

[rev: S. Devitt, W. Munro, and Kae Nemoto, Reports on Progress in Physics, (2013)]



toric/surface code by Kitaev (1998)

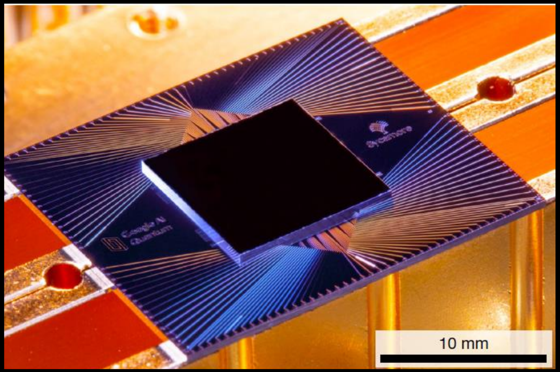
[rev: A. Fowler et al. PRA 86, 032324 (2012)]

Quantum computing industry

IBM

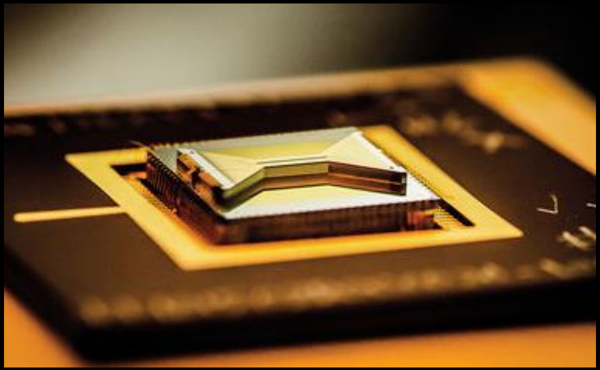
127 Q chip “Eagle” (2021)

Google



72 Q chip “Sycamore” (2022)

IONQ



79 Q trapped ions chain

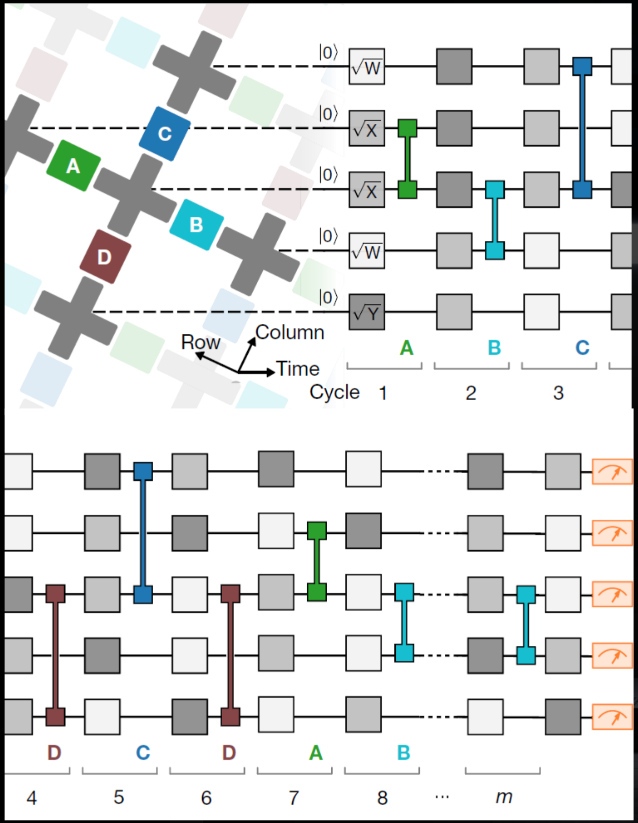
PASQAL

~200 Q arrays (2022)

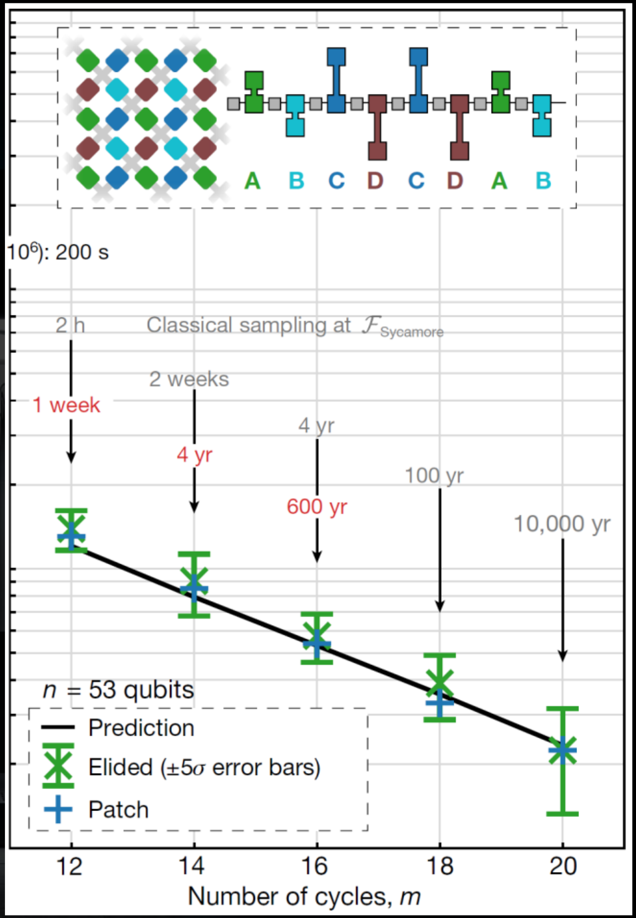
- The scale of quantum chips is getting larger every year, and state space is now intractable
- Without quantum error correction qubit operations are noisy, and we cannot perform more that thousands of operations (many algorithms require >10^6 gates).
- Now we are in the noisy intermediate scale quantum (NISQ) era of quantum computing, which requires new algorithms.

Quantum supremacy

Goal: find *a* problem which can be solved with a quantum computer, but impossible classically. Google AI: we *can* sample the bits from random circuit with N = 53 Q and depth of 20 in 200 seconds, compared to 10,000 years with classical supercomputers.*



random noisy circuit

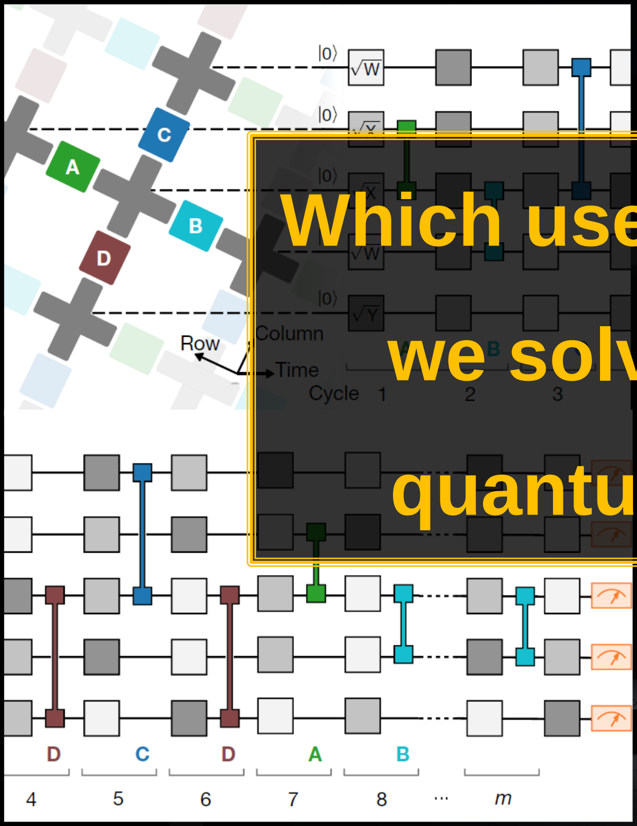


quantum supremacy regime

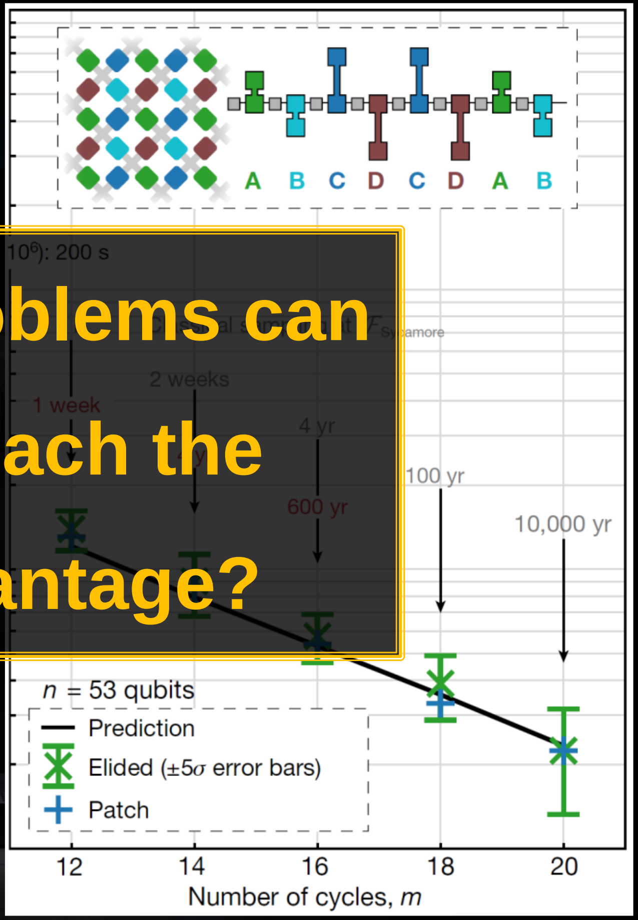
[F. Arute et al. (Google), Nature 574, 505 (2019)]

Quantum supremacy

Goal: find *a* problem which can be solved with a quantum computer, but impossible classically. Google AI: we *can* sample the bits from random circuit with N = 53 Q and depth of 20 in 200 seconds, compared to 10,000 years with classical supercomputers.*



random noisy circuit



quantum supremacy regime

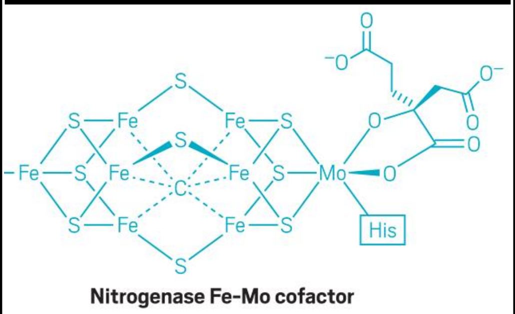
Which useful problems can we solve to reach the quantum advantage?

[F. Arute et al. (Google), Nature 574, 505 (2019)]

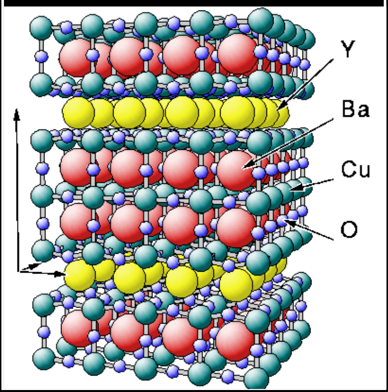
Quantum computing applications

Possible areas for reaching the advantage depend on state-of-the-art and the competition.

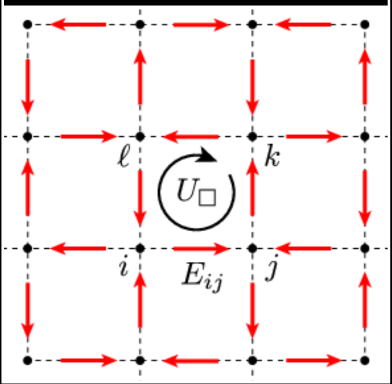
Quantum chemistry



Material science

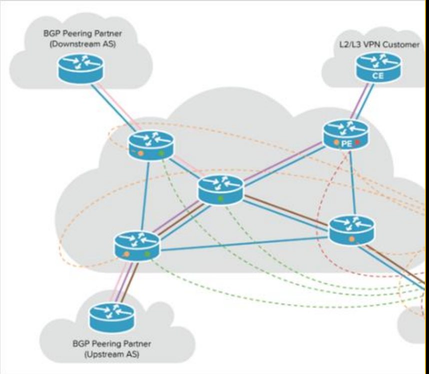


HEP

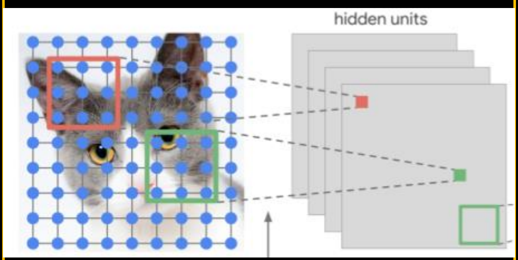


Quantum Finance

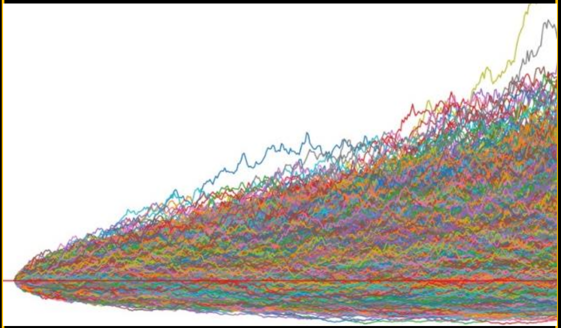
Optimization



Quantum Machine Learning

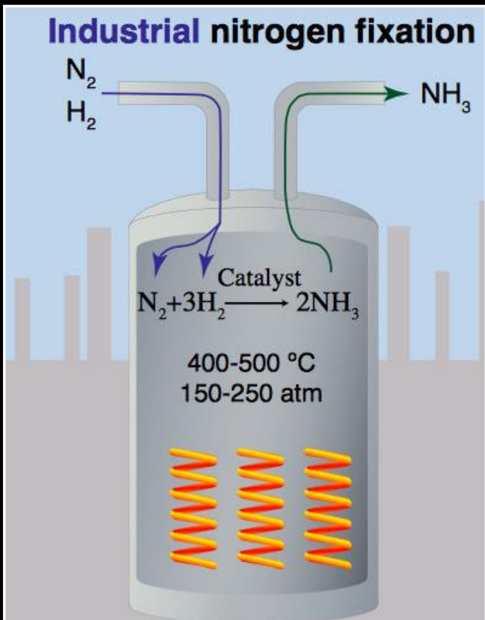


Quantum SciML



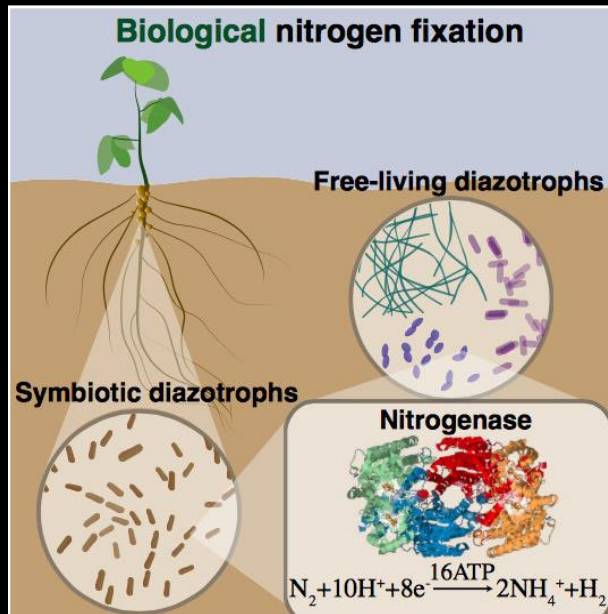
Applications: Quantum chemistry

fertilizer production



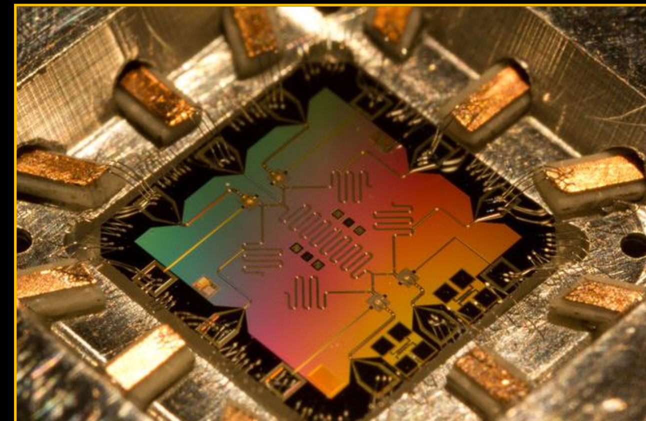
Haber-Bosch process (2-5% world's natural gas consumption)

nitrogen cycle



knowledge of energy structure for a large active complex is required

quantum simulator



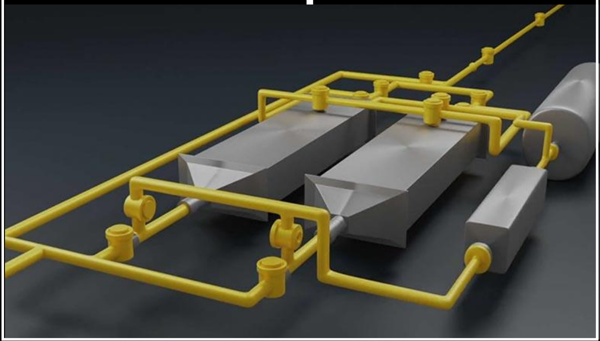
+ classical supercomputer

Large scale quantum simulation can help to cut fossil fuel emission and save ~£40 billion/year

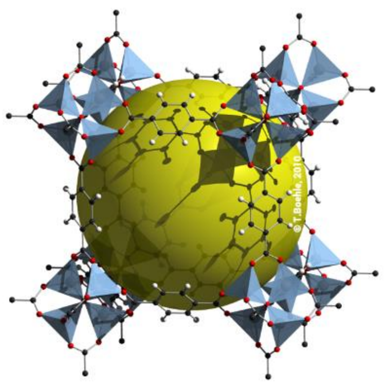
Applications: Quantum chemistry

Current goal: find applications where quantum computing challenges classical HPC and gives advantage for reasonably small system sizes

CO_2 capture by absorption



Metal–organic framework



Hamiltonian in second quantization

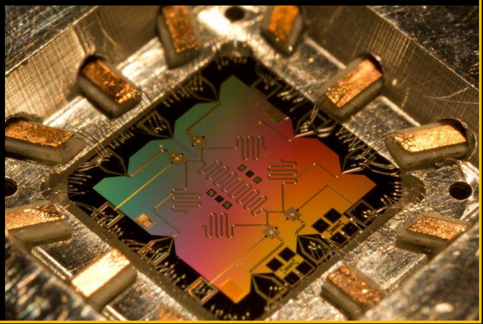
$$H = \sum_{pq} h_{pq} a_p^\dagger a_q + \frac{1}{2} \sum_{pqrs} h_{pqrs} a_p^\dagger a_q^\dagger a_r a_s$$

- performing efficient computation of low energy spectrum requires developing new quantum algorithms
- chemistry methods can capture dynamical correlation at sufficiently large scale, and static correlations are important

[arXiv:2009.12472 (2020)]

classical computer

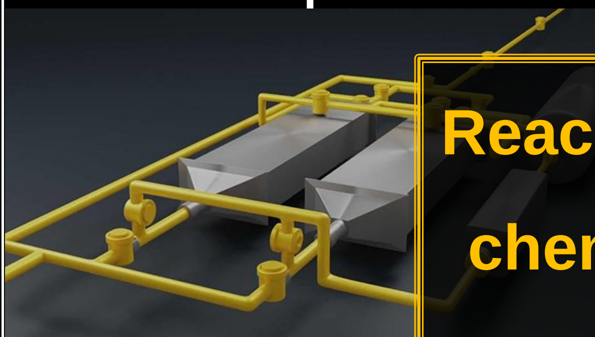
quantum computer



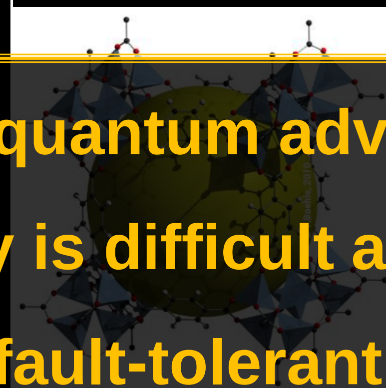
Applications: Quantum chemistry

Current goal: find applications where quantum computing challenges classical HPC and gives advantage for reasonably small system sizes

CO_2 capture by absorption



Metal–organic framework



Hamiltonian in second quantization

$$H = \sum_{pq} h_{pq} a_p^\dagger a_q + \frac{1}{2} \sum_{pqrs} h_{pqrs} a_p^\dagger a_q^\dagger a_r a_s$$

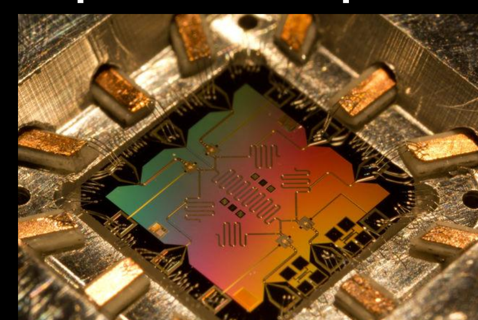
Reaching quantum advantage in chemistry is difficult and likely requires fault-tolerant devices

- performing efficient computation of low energy spectrum requires developing new quantum algorithms
- chemistry methods can capture dynamical correlation at sufficiently large scale, and static correlations are important

[arXiv:2009.12472 (2020)]

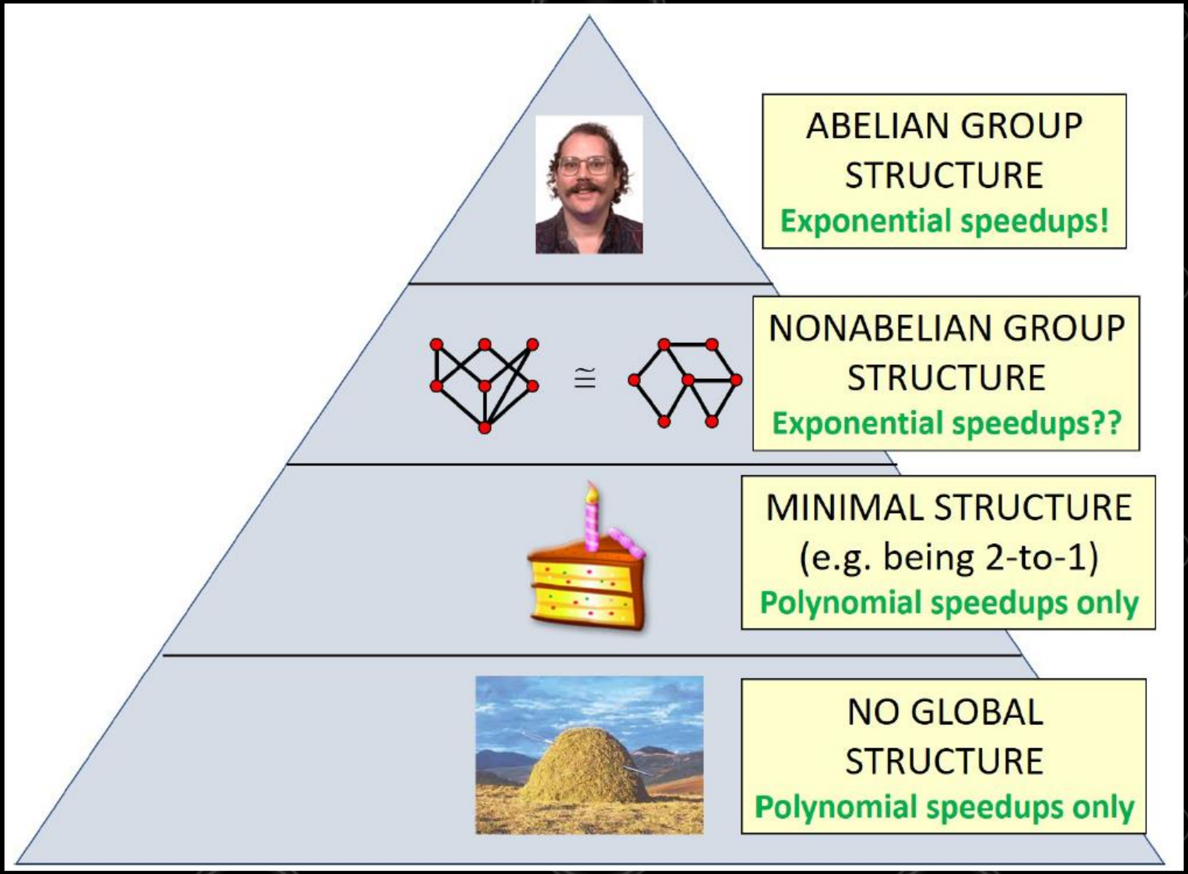
classical computer

quantum computer



Quantum algorithms hierarchy

Scott Aaronson's vision of speed up vs structure competition in quantum computing.

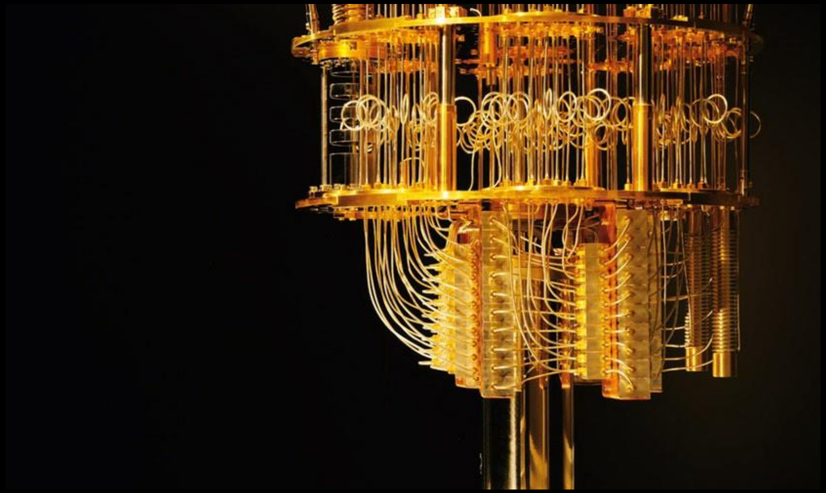


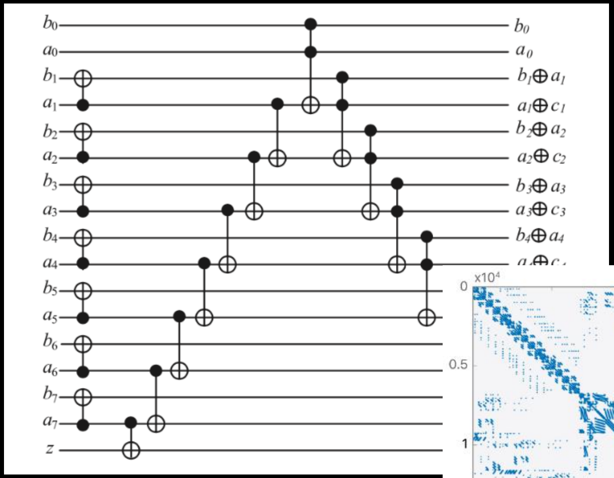
"How Much Structure Is Needed for Huge Quantum Speedups?"
arXiv:2209.06930 (Sep 2022)

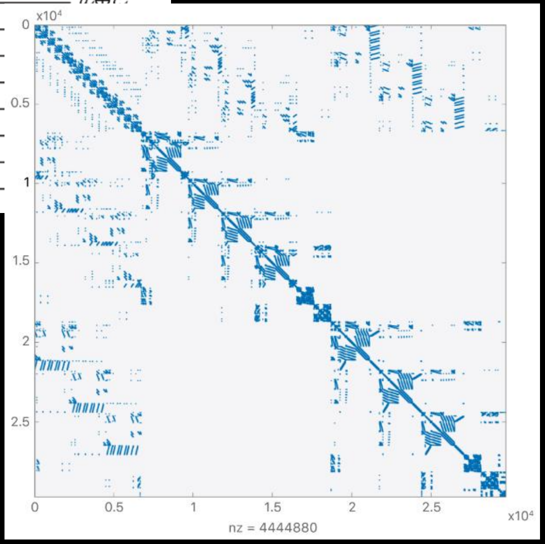
(Interim) Conclusions and questions

- ✓ Quantum computers offer a powerful and innovative paradigm for performing challenging calculations of certain type
- ✓ Quantum hardware improves every year but is still limited by noise and will depend on ability to correct (mitigate) errors
- ✓ Quantum software and algorithmic improvements can bring quantum advantage for tailored tasks
- ✓ Getting a significant speed-up for real industrial problems depends on scaling, absolute clock rates of QPUs, and classical state-of-the-art
- ✓ Can we achieve the quantum advantage in the near-term without error correction, or does the time shift to 10+Y future?
- ✓ Which areas are the potential prime beneficiaries of QC power?
- ✓ Can we use it for advancing scientific computing?

Quantum Machine Learning and SciML

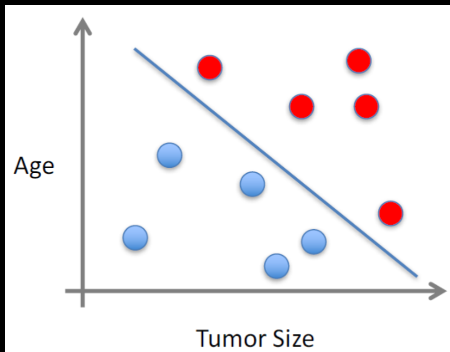


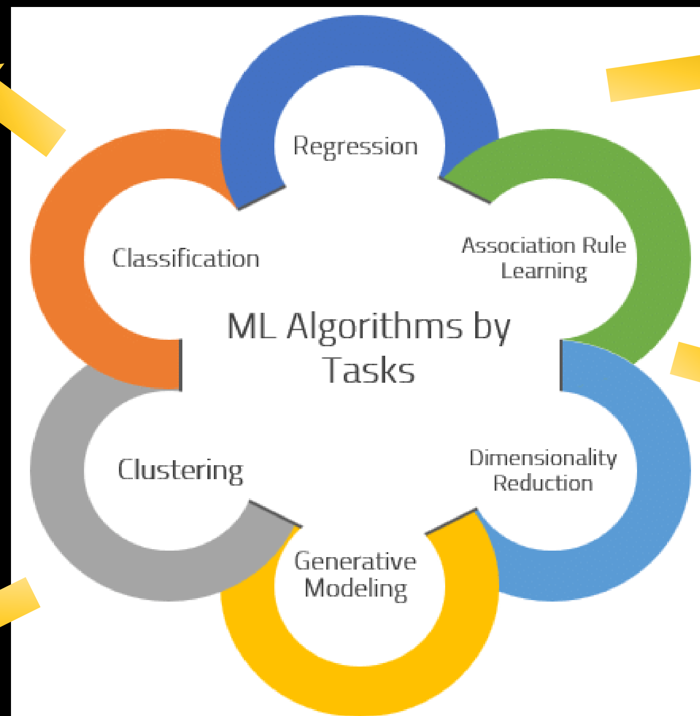


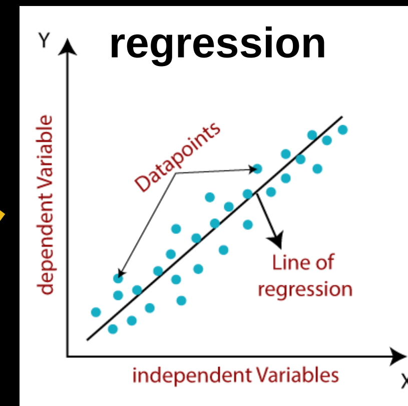


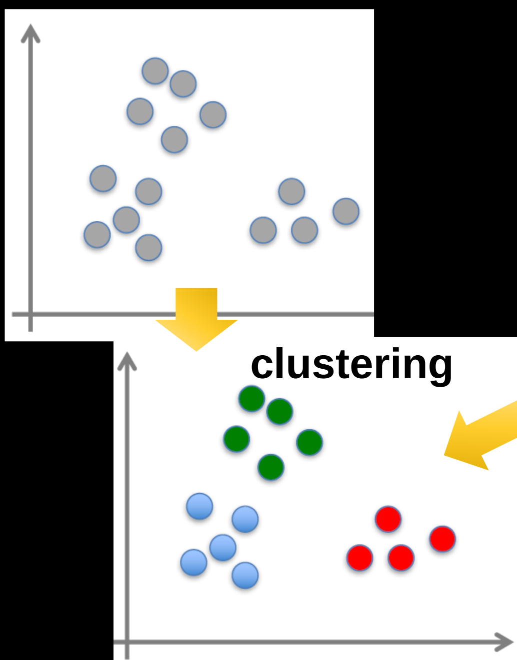
Machine learning

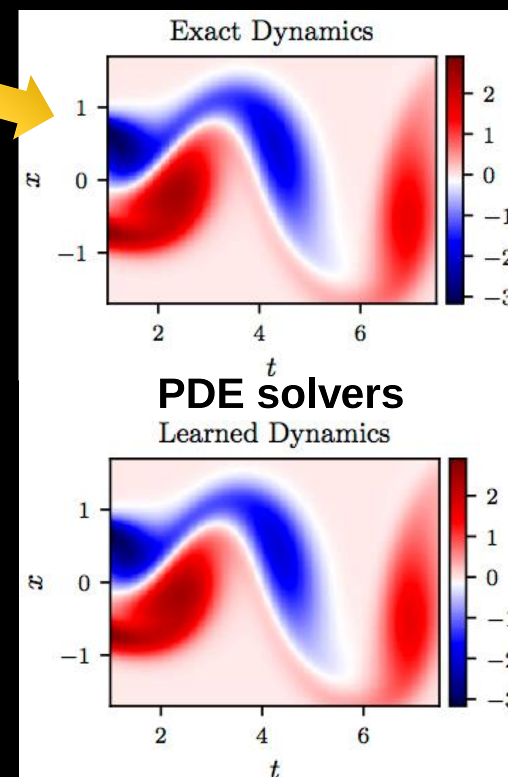
classification



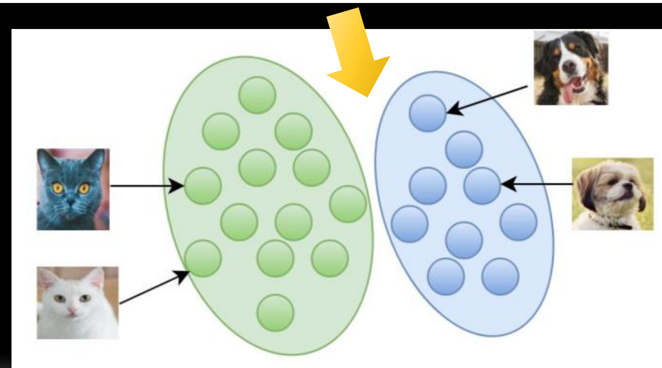






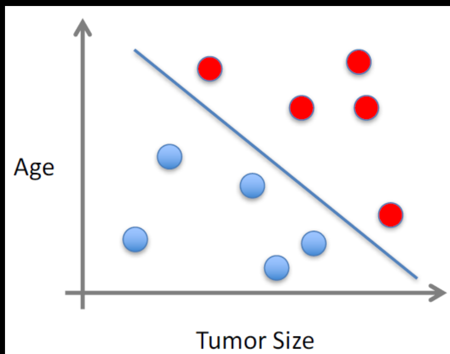


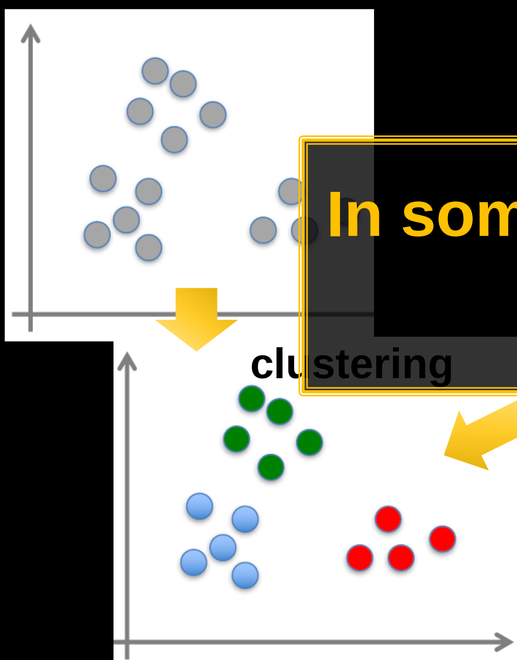
generative modelling



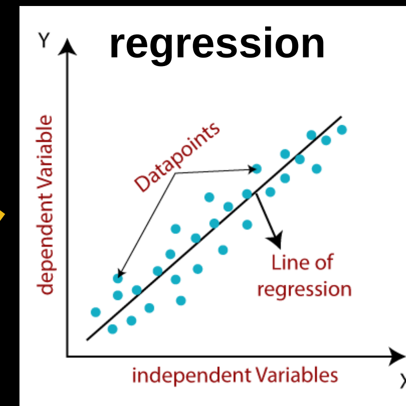
Machine learning

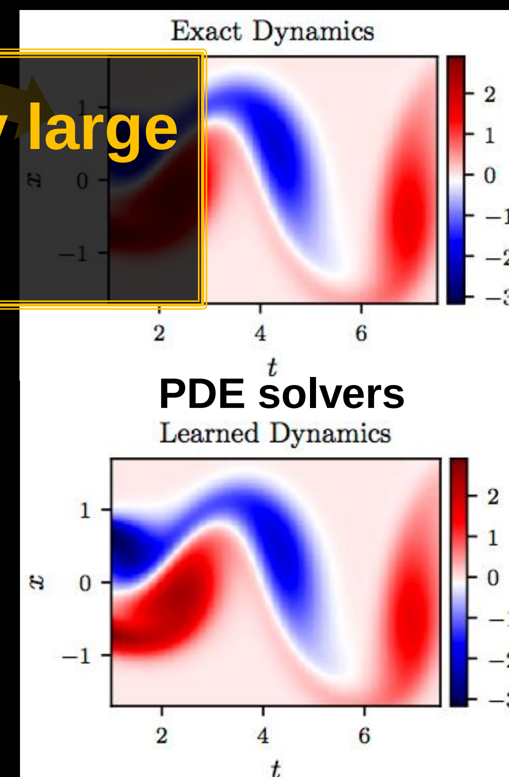
classification





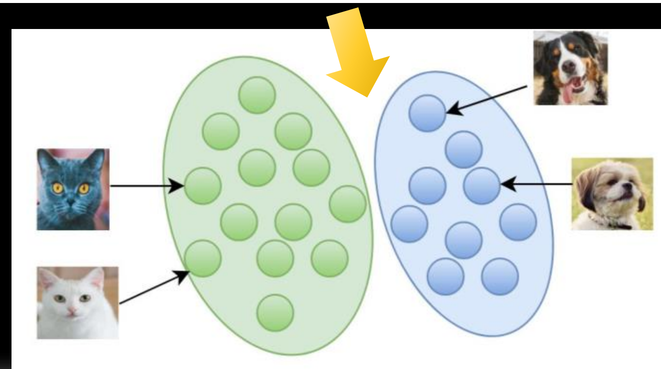
clustering





PDE solvers

generative modelling

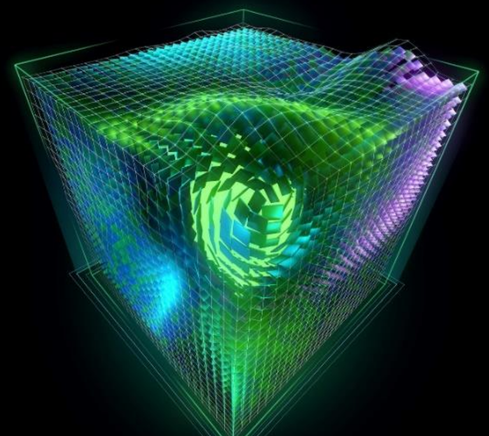


In some tasks we are limited by large problem dimensionality

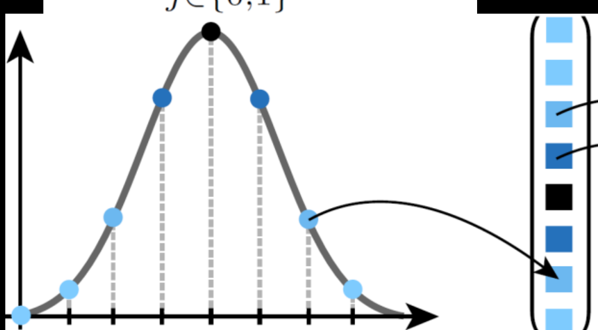
Scientific computing

Another area of science where quantum computing may help corresponds to systems of linear equations, and in particular discretised partial differential equations.

finite difference mesh



$$|\psi\rangle = \sum_{j\in\{0,1\}^n} g(x_j)|j\rangle$$



quantum amplitude encoding

- Start by specifying a systems of differential equations

$$\nabla \cdot u = 0 \qquad \rho\frac{du}{dt} = -\nabla p + \mu\nabla^2 u + \mathbf{F}$$

- Discretise derivatives and write it in a matrix form

$$\frac{\partial^2 u}{\partial x^2} \to \frac{1}{\Delta x^2}\begin{bmatrix} -2 & 1 & 0 & \cdots & 0 & 1 \\ 1 & -2 & 1 & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & & \\ \vdots & & & & & \vdots \\ & & & & & 0 \\ \vdots & \cdots & 0 & 1 & -2 & 1 \\ 1 & 0 & \cdots & 0 & 1 & -2 \end{bmatrix}\begin{pmatrix} u_0 \\ u_1 \\ \vdots \\ u_n \end{pmatrix}$$

- Use linearization techniques to make it look as a system of linear equations with initial state provided

$$\hat{A}\overrightarrow{x} = \overrightarrow{b}$$

Quantum embedding allows for exponentially increasing mesh with linear increase of resources (qubit number)

Linear systems of equations: HHL

The goal: encode a system of linear equations into Hilbert space and invert $2^N x 2^N$ matrix **A**. Classically can use conjugate gradients and solve in O(N s κ log(1/ε)) time [depends on the condition number κ and precision ε].

$$\hat{A}\vec{x} = \vec{b}$$

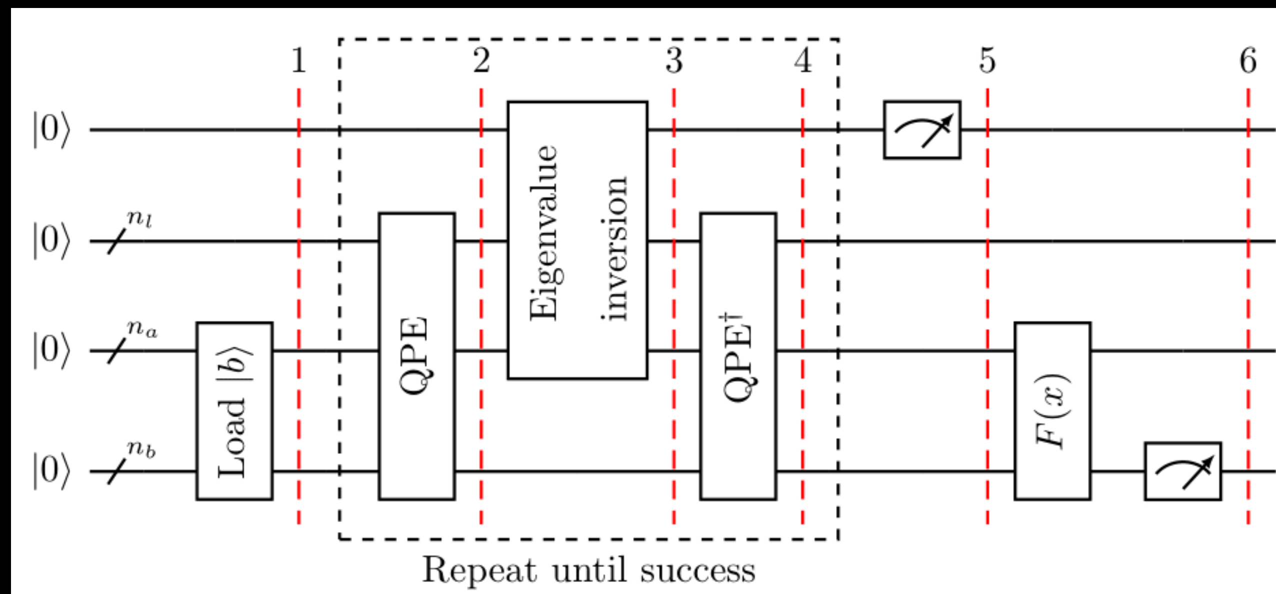
find $\vec{x}$

HHL: reformulate a problem as inversion of Hermitian operator **A**

$$|x\rangle = \hat{A}^{-1}|b\rangle$$

- prepare **input** (QRAM)
- **push** eigenvalues of A bitwise to ancillas
- **invert** eigs with the top ancilla (amp. ampl.)
- QPE-back, **measure**
- **read-out** result for some observable

$$\langle x|\hat{M}|x\rangle$$



standard HHL protocol (3 registers, "QRAM", phase estimation)

[A. Harrow, A. Hassidim, S. Lloyd, Phys. Rev. Lett. 15, 150502 (2009)]

Quantum solution runs in O(log(N) s^2 κ^2 /ε) time, and can be further improved to κ log(1/ε).

Using Euler's finite differencing one can solve linear differential equations in a similar way.

Linear systems of equations: LCU

The advantage: instead of oracle based approach, use the approximation theory and best Hamiltonian simulation with **linear combination of unitaries (LCU)** approach.

Prepare **quantum version** of the algorithm, where the matrix is **H**, and we need to **construct inverse Hamiltonian operator**.

Matrix inverse = a sum of exponents

$$H^{-1} = \frac{i}{\sqrt{2\pi}} \int_{0}^{+\infty} dy \int_{-\infty}^{+\infty} dz z \exp(-z^2/2) \exp(-iyzH)$$

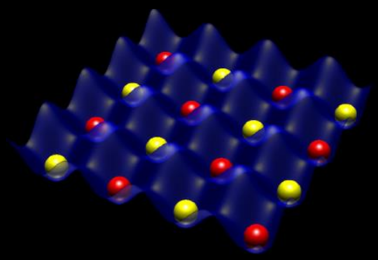
$$\rightarrow \frac{i}{\sqrt{2\pi}} \sum_{j_y=0}^{M_y-1} \Delta_y \sum_{j_z=-M_z}^{+M_z} \Delta_z (j_z \Delta_z) \exp[-j_z^2 \Delta_z^2/2] \exp[-i(j_y \Delta_y)(j_z \Delta_z)H]$$

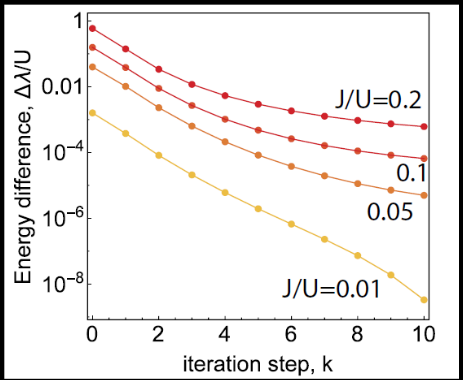
$$\hat{\mathcal{H}}^{-k} = \sum_{\ell=1}^{L_k} c_{k,\ell} \exp(-i\phi_{k,\ell}\hat{\mathcal{H}}) \equiv \hat{\mathcal{H}}_{\mathrm{a}}^{-k}$$

Ancilla-based implementation – way to go with fault-tolerant devices
[Childs-Kothari-Somma, SIAM J. Comp. (2017); arXiv:1511.02306]

Sequential estimation by applying each unitary separately (time-grid method)

$$\lambda_k^{(\mathrm{a})} = \frac{\sum_{\ell,\ell'} \langle \psi_{k,\ell'} | \hat{\mathcal{H}} | \psi_{k,\ell} \rangle}{\sum_{\ell,\ell'} \langle \psi_{k,\ell'} | \psi_{k,\ell} \rangle}$$





[OK, npj Quantum Information 6, 1 (2020)]

ground state of Bose-Hubbard model

Linear systems of equations: LCU

We can implement non-unitary LCU with amplitude amplification protocol:

Implement $M = \sum_i \alpha_i U_i$ using SELECT $V|0^m\rangle := \frac{1}{\sqrt{\alpha}} \sum_i \sqrt{\alpha_i}|i\rangle$

and controlled unitaries $U := \sum_i |i\rangle\langle i| \otimes U_i$ such that $W := V^\dagger U V$

implements

$$\boxed{W|0^m\rangle|\psi\rangle = \frac{1}{\alpha}|0^m\rangle M|\psi\rangle + |\Psi^\perp\rangle}$$

repeat $O((\alpha/\||M|\psi\rangle\||)^2)$ times until reach desired outcome,

or use amplitude amplification originally studied in [G.Brassard, P. Hoyer, M. Mosca, A. Tapp, arXiv:quant-ph/0005055 (2002)], also see [Childs-Kothari-Somma, arXiv:1511.02306]

Power/inverse power iteration: $K = O[\{\log(\varepsilon) \sin^{-2}(\theta_0)\}/\log(\lambda_2/\lambda_1)]$ iterations to reach error ε – log in error, depends on overlap – convergence similar to QPE.

QC for differential equations

However, the challenges of described scheme include: 1) the input problem; 2) the output problem; 3) deep circuits and ancilla overhead; 4) linearity; 5) dependence on the finite differencing. Overall computationally efficient, but very NISQ-unfriendly.

REVIEW

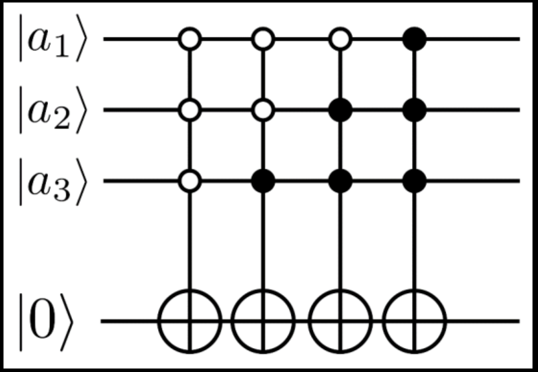
Quantum machine learning

Jacob Biamonte[1,2], Peter Wittek[3], Nicola Pancotti[4], Patrick Rebentrost[5], Nathan Wiebe[6] & Seth Lloyd[7]

The **input problem** corresponds to the challenge of preparing an arbitrary **input state** $|b\rangle$ using the **amplitude encoding**, as this may require qRAM with **exponentially many operations**.

amplitude encoding:

$$|b\rangle = \sum_{k=1}^{2^N} \beta_k |k\rangle$$



QRAM-type oracle

The **output problem** corresponds to the challenge of reading out information about the prepared function from the quantum state with exponentially many entries – this may require full state tomography.

$$|x\rangle = \hat{A}^{-1}|b\rangle$$

prepare $|x\rangle$

Finally, at the size where quantum overcomes classical linear solvers, the **gate counts** for relevant problems can be **gigantic** (depth of the order of 10^{25} that with 1ns operation would take approx. 3×10^8 years, even without error correction)

[arXiv:1505.06552]

QC for differential equations

However, the challenges of described scheme include: 1) the input problem; 2) the output problem; 3) deep circuits and ancilla overhead; 4) linearity; 5) dependence on the finite differencing. Overall computationally efficient, but very NISQ-unfriendly.


REVIEW

Quantum machine learning

Jacob Biamonte[1,2], Peter Wittek[3], Nicola Pancotti[4], Patrick Rebentrost[5], Nathan Wiebe[6] & Seth Lloyd[7]

The **input problem** corresponds to the challenge of preparing an arbitrary **input state** $|b\rangle$ using the amplitude encoding, and this may require qRAM with **exponentially many operations**.

amplitude encoding: $|b\rangle = \sum_{k=1}^{N} \beta_k |k\rangle$

We may need to rethink the way quantum machine learning is approached

QRAM-type oracle

The **output problem** corresponds to the challenge of reading out information about the prepared function from the quantum state with exponentially many entries – this may require full state tomography.

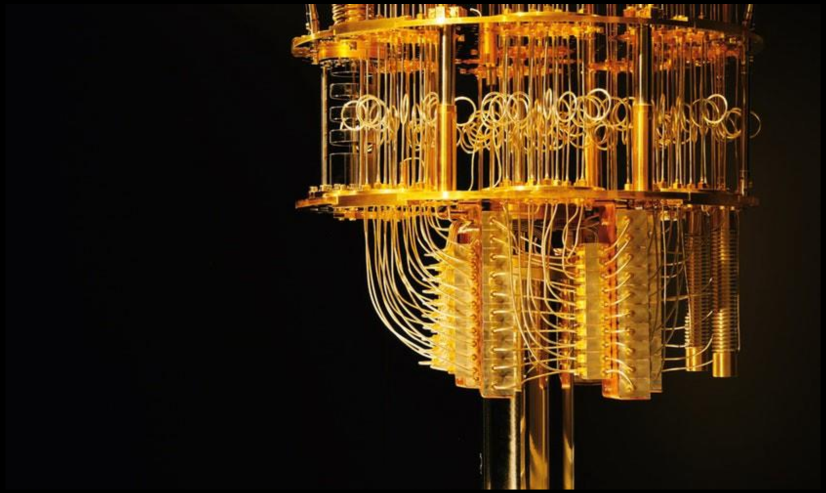
$$|x\rangle = \hat{A}^{-1}|b\rangle$$

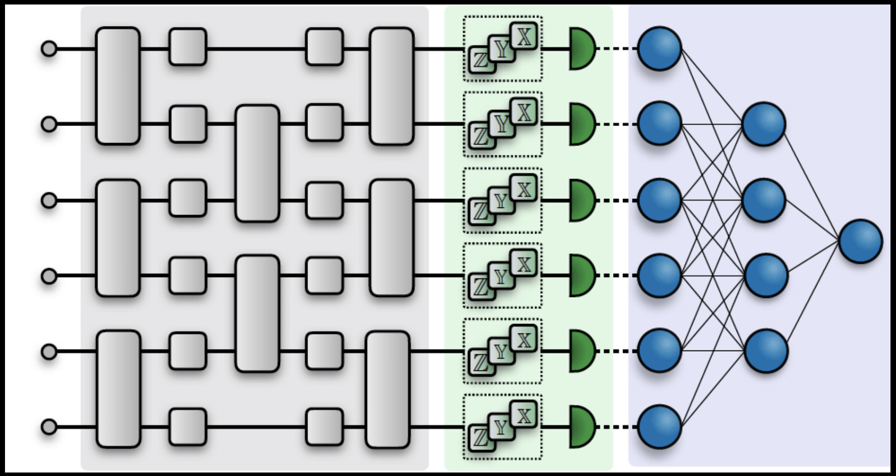
prepare $|x\rangle$

Finally, at the size where quantum overcomes classical linear solvers, the **gate counts** for relevant problems can be **gigantic** (depth of the order of 10^{25} that with 1ns operation would take approx. 3×10^8 years, even without error correction)

[arXiv:1505.06552]

Quantum Machine Learning





to be discussed in the second lecture…