

- They should Ask questions when arise
- Ask about knowledge status of audience

Felix Gemeinhardt
Johannes Kepler University Linz
Institute for Business Informatics –
Software Engineering

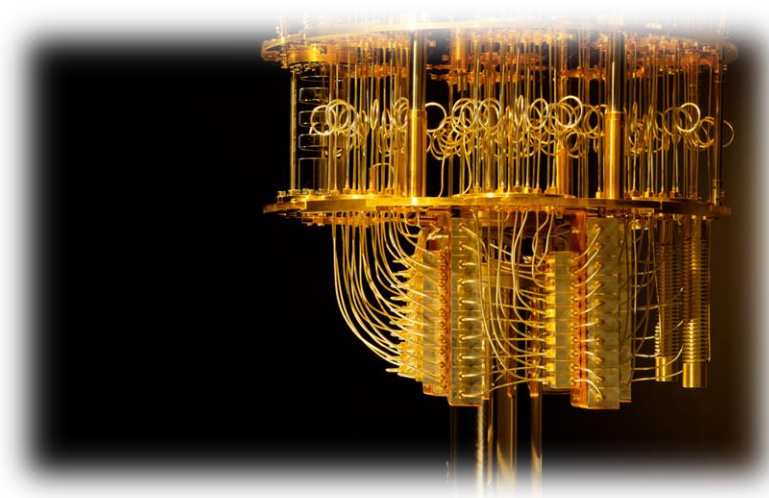
Quantum Computing

From Fundamentals to first Quantum Algorithms

- **This material, no matter whether in printed or electronic form, may be used for personal and non-commercial educational use only. Any reproduction of this material, no matter whether as a whole or in parts, no matter whether in printed or in electronic form, requires explicit prior acceptance of the authors.**

Goal: Overview and basic understanding of working principles

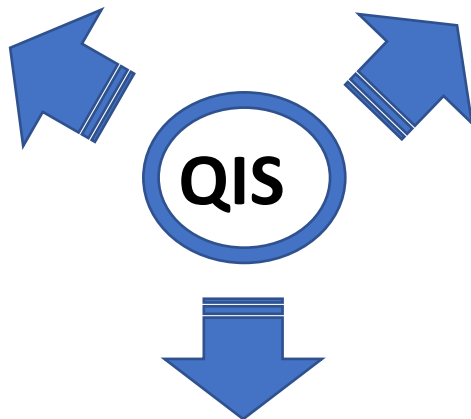
1. **Motivation** and Overview
2. Basic **Working Principles**
3. **Near-term** Applications
4. Simple Quantum **Algorithms**
5. **Challenges** and Limitations
6. Quantum **Software** Engineering



Overview and Motivation



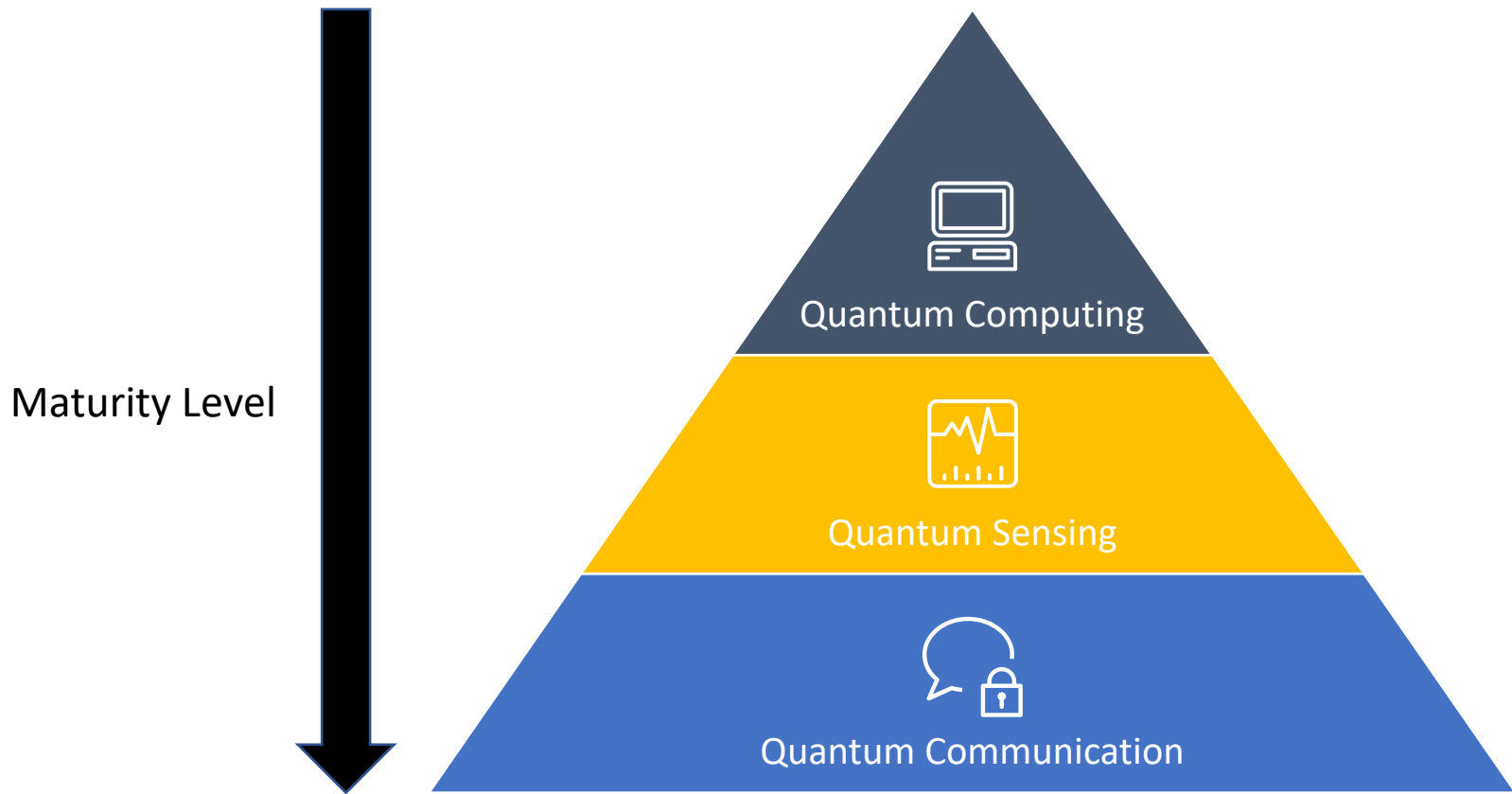
**Computer
Science**



**Information
Theory**

**Quantum
Mechanics**

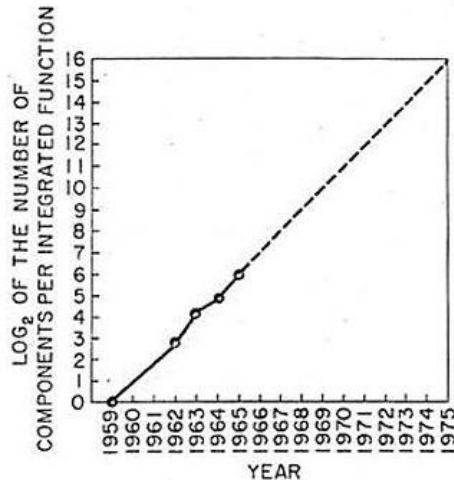
- Emerged in the 1920s
- Inventions like: Transistors, Lasers and GPS



Classical Computing: Limitations - Hardware

■ Limits of Moore's law

- Doubling of transistor counts on microchips every 12-24 months
- Physical limitations



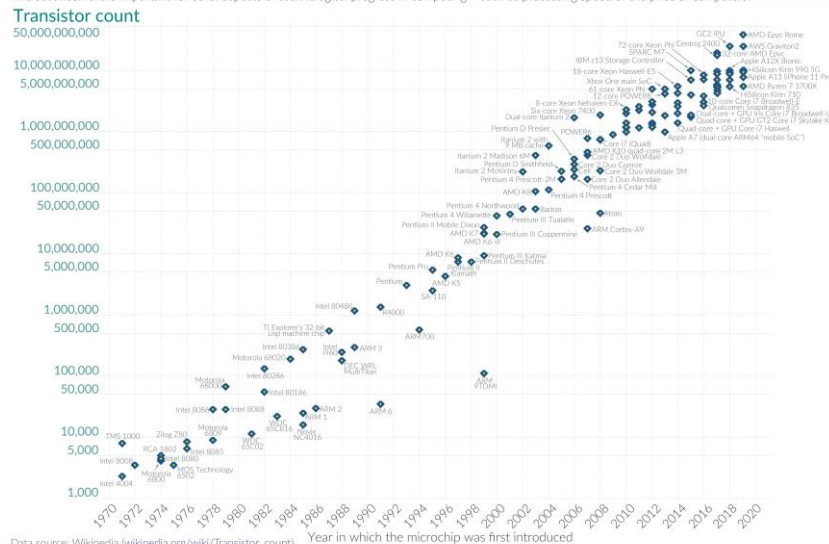
Source:

https://web.archive.org/web/20211221191600/https://www.intel.com/pressroom/kits/events/moores_law_40th/index.htm?iid=tech_mooreslaw+body_presskit

Moore's Law: The number of transistors on microchips doubles every two years

Moore's law describes the empirical regularity that the number of transistors on integrated circuits doubles approximately every two years. This advancement is important for other aspects of technological progress in computing – such as processing speed or the price of computers.

Our World
in Data

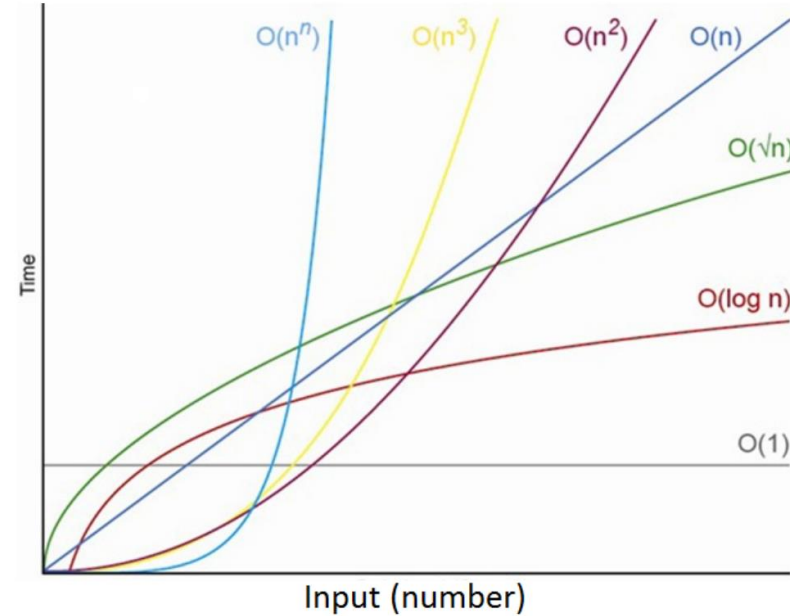


Data source: Wikipedia ([wikipedia.org/wiki/Transistor_count](https://en.wikipedia.org/wiki/Transistor_count))
OurWorldInData.org – Research and data to make progress against the world's largest problems.

Licensed under CC-BY by the authors Hannah Ritchie and Max Roser.

Source: <https://ourworldindata.org/technological-progress>

- **Many complex problems are intractable for classical computing,**
e.g.:
 - Exponentially growing search spaces
 - Simulation of quantum processes
- **Best case:**
 - From $O(n^n)$ to $O(n^1)$



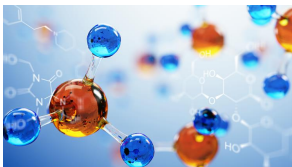
Source: Hidary (2019). Quantum Computing: An Applied Approach

Applications – from research to operations

Research applications



Batteries



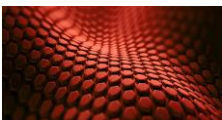
Drug discovery



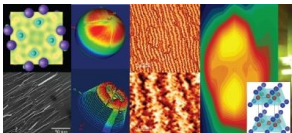
Semiconductors



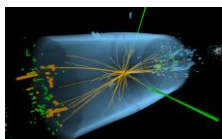
Fertilizer production



Materials design



Condensed matter physics



High-energy
particle physics



Machine Learning

Operations applications



Transportation



Finance



Energy utilities



Telecoms



Manufacturing




Marketing

- **Technical Challenges:**

- Error prone (coherence time)
- Sensitivity to environment and to each other (noise)
- Accuracy of Quantum Operations
- ...

- **Regimes**

- Noisy Intermediate Scale Quantum (NISQ-era)
- Fault-tolerant Quantum Computing

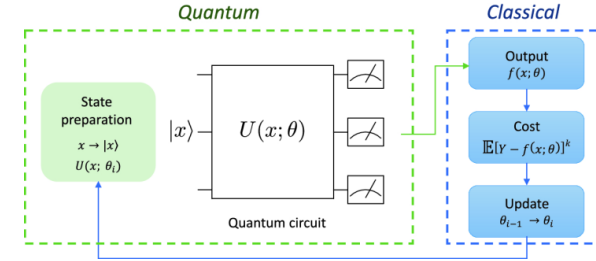


Preskill, J., 2018. Quantum computing in the NISQ era and beyond. *Quantum*, 2, p.79.

NISQ-Era Approaches to QC

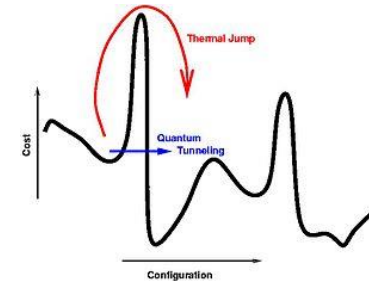
■ Variational Quantum Algorithms

- Similar to neural nets in ML
- VQE, QAOA
- Gate-based → sequential programming



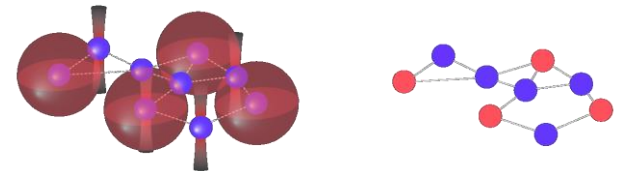
■ Quantum Annealing

- Encode optimization problem into energy of quantum system
- System “wants” to stay in minimum



■ Quantum Simulators

- Encode problem into energy of quantum system
- Different quantum phenomena

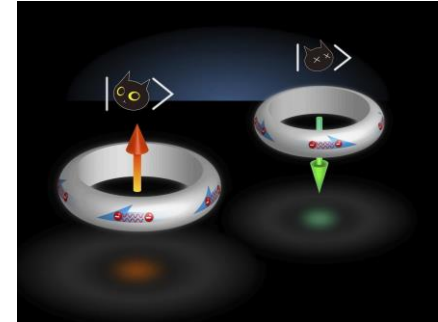


▪ **Photonics**

- Photons are information carrier
- Optical elements (mirrors, phase shifters) for manipulation

▪ **Superconductors**

- Google, IBM, Rigetti,...
- Electric current produces magnetic moment (spin)
- Temperatures: mK
- Microwave pulses for manipulation

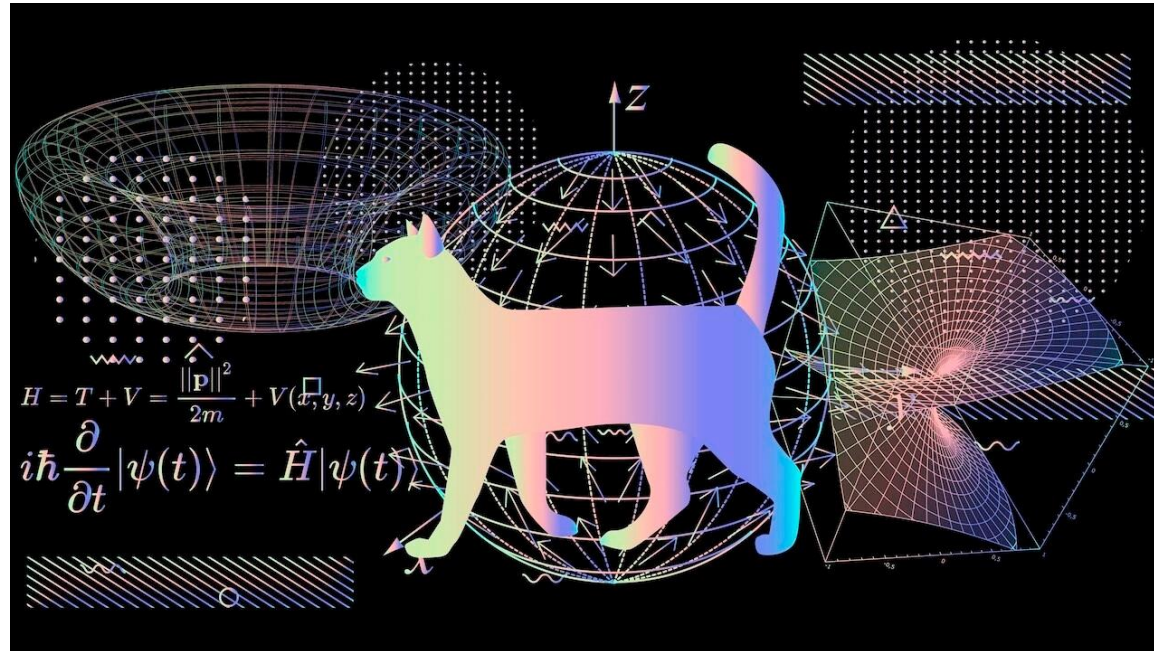


Source: Johnston et. al (2019).
Programming Quantum Computers

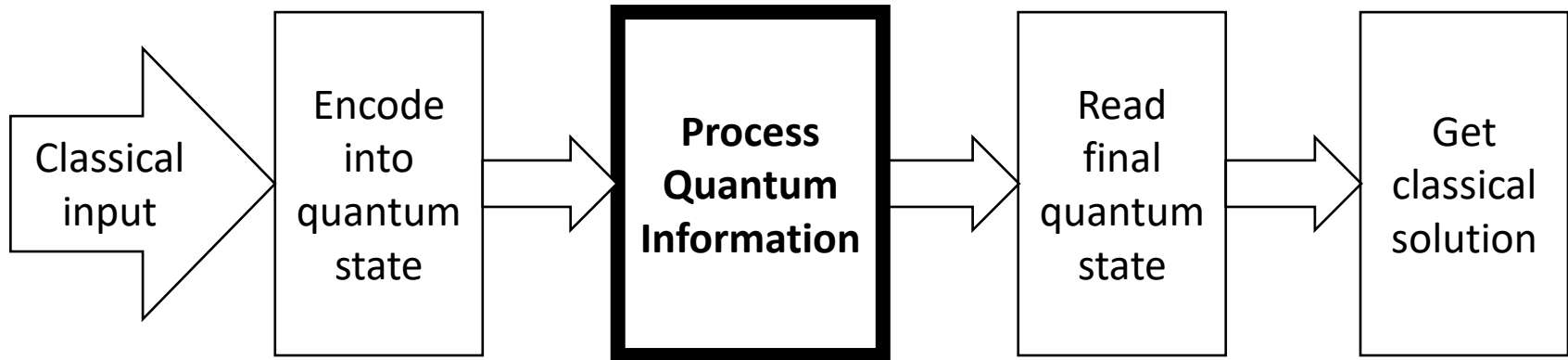
- **Trapped Ion**
 - Ions in electromagnetic field
 - Lasers for manipulation
- **And many more:**
 - Topological Quantum Computation
 - Silicon-based
 - ...

All these approaches seek to make the jump to the next regime. To do this, they try to better model a **Qubit**.

Basic Working Principles



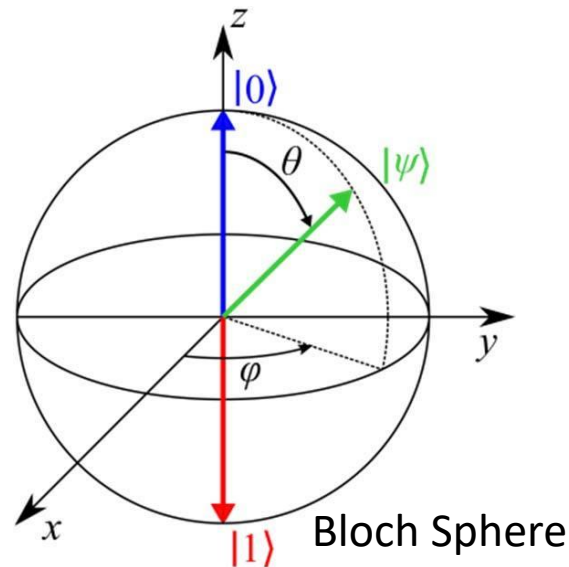
Quantum Information Processing – Pipeline



- A qubit is a **two-level** quantum mechanical system
- The **state** of the qubit at any given time can be represented by a **vector**

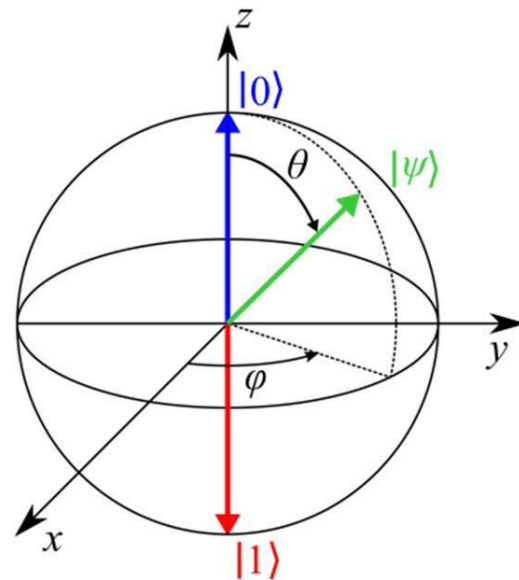
$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

- Similar to classical bit 0,1 $\rightarrow |0\rangle, |1\rangle$
- Can also be a mixture \rightarrow **superposition**

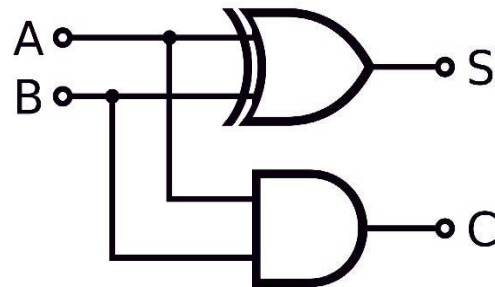


Phase

- Additional **degree of freedom** in quantum systems
- Often useful to **encode information** in the phase
- Can then be **transformed to amplitudes via QFT** → see later
→ intuitively: transformation from φ to θ

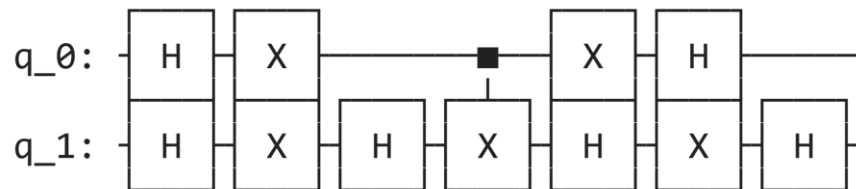


- **Classical Computing Circuit**

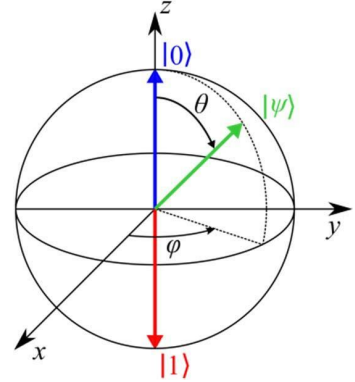


- **Quantum Computing Circuit**



- Construct and read these diagrams from left to right
- Input and output space are the same



- Quantum operations **are matrices**
- **Isolated quantum system**
 - Every quantum operation is reversible
 - Every quantum operation is unitary
 - describes rotation but no change in vector length
- **Reversibility**
 - $U^{-1}U|\Psi\rangle = U^\dagger U|\Psi\rangle = |\Psi\rangle$
 - U^\dagger is U transposed and complex conjugated

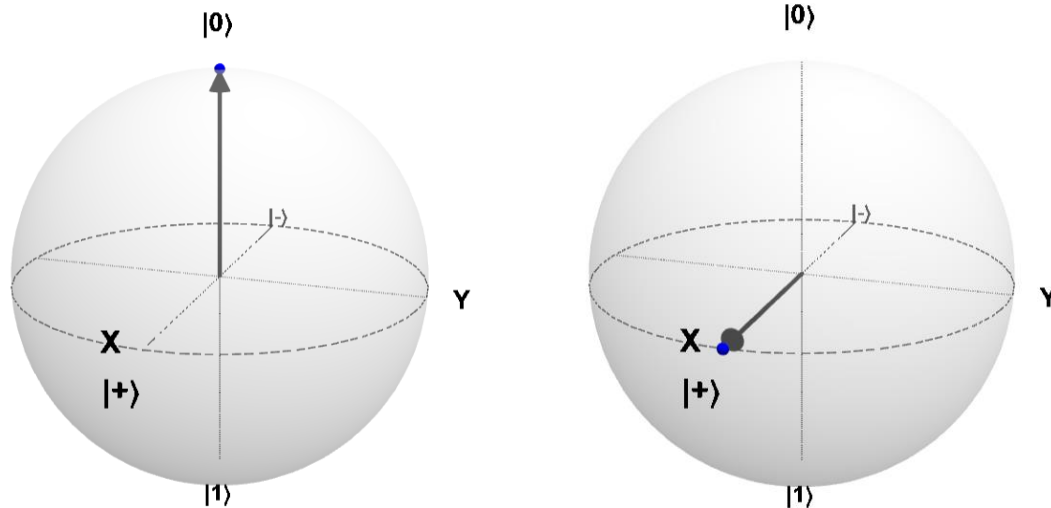


Basic Concepts – From Classical to Quantum Operations

Properties	Classical Operations	Quantum Operations
Reversibility	 Only NOT operation	
Universality	<ol style="list-style-type: none">1. Set{AND, NOT, OR, NAND, XOR, FANOUT}2. Set{NAND}	<ol style="list-style-type: none">1. Set {Toffoli, basis-changing unary operator with real coefficients (such as H)}2. Set{CNOT, T, Hg}3. Set{RX,RY,RZ,P,CNOT}

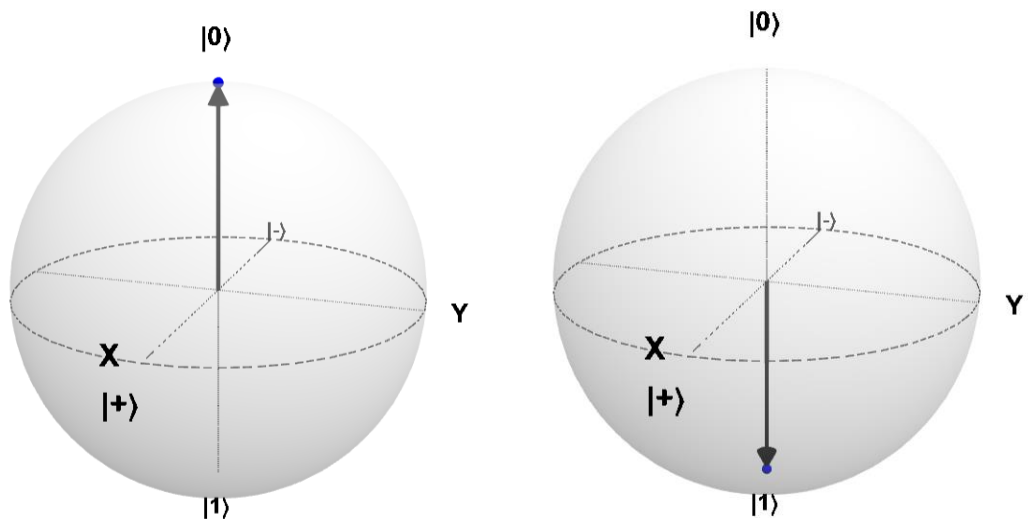
Quantum Operations – Hadamard

- Hadamard operator is *crucial* in quantum computing
- Takes a qubit into a superposition of two states
- Bloch Sphere:



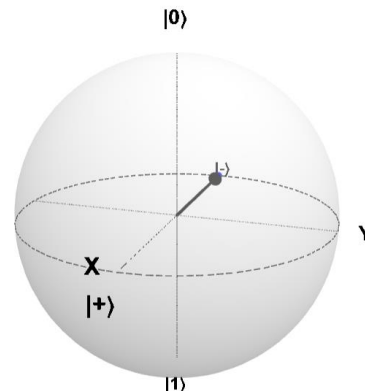
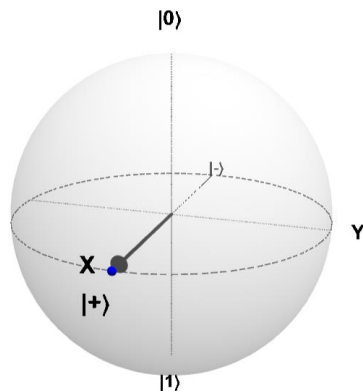
Quantum Operations – Pauli X

- Similar behavior like *Not* in classical computing
- Also known as *Not Gate*

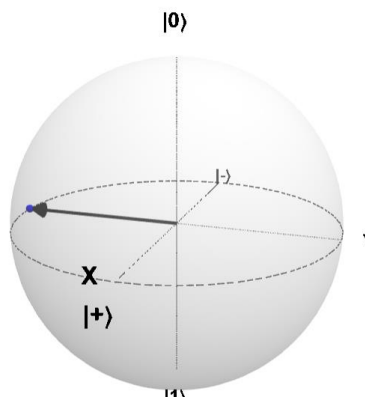
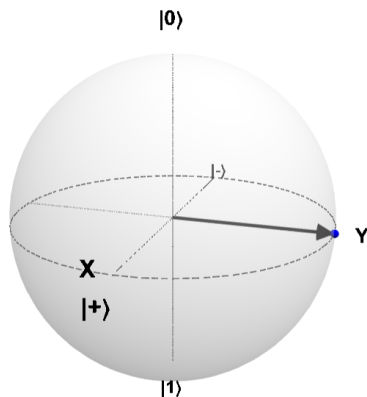


Quantum Operations – Pauli Y & Z

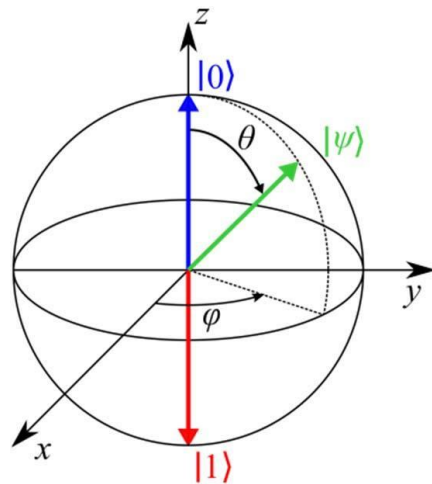
- Pauli Y



- Pauli Z

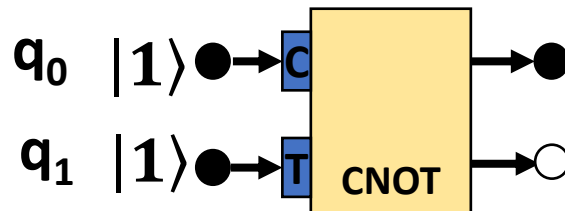
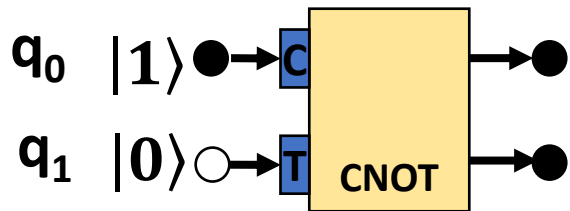


- **Bloch sphere rotations can be parametrized**
 - E.g., rotation of φ around z-axis
- **3 angles for any arbitrary rotation**
 - Euler's rotation theorem
- **Examples:**
 - RX, RY, RZ

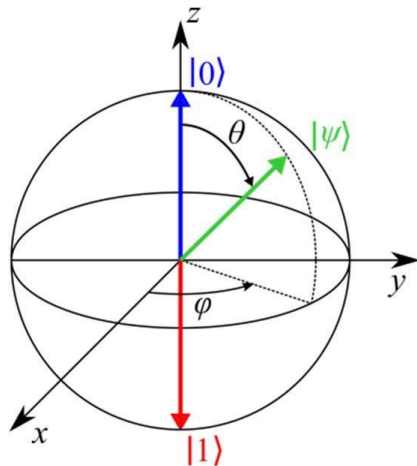


Quantum Operations – CNOT

- Controlled-NOT (CNOT)
- First Qubit is the *control qubit*
- Second Qubit is the *target* qubit
- Examples



- **Measurement destroys superposition**
 - Non-reversible quantum operation
 - What was state before measurement?
- Probability distribution → Quantum state
- No-cloning theorem
 - Repeated computation and measurement
- **Intermediate states** of the quantum system are **not accessible**



- Also known as: Dirac Notation

$$\langle \mathbf{0} | = (\mathbf{1} \ \mathbf{0}), \quad | \mathbf{0} \rangle = \begin{pmatrix} \mathbf{1} \\ \mathbf{0} \end{pmatrix} \quad \langle \mathbf{1} | = (\mathbf{0} \ \mathbf{1}), \quad | \mathbf{1} \rangle = \begin{pmatrix} \mathbf{0} \\ \mathbf{1} \end{pmatrix}$$

- Multi-qubit state representation

$$| \mathbf{00} \rangle = \begin{pmatrix} \mathbf{1} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix} \quad | \mathbf{01} \rangle = \begin{pmatrix} \mathbf{0} \\ \mathbf{1} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix} \quad | \mathbf{10} \rangle = \begin{pmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{1} \\ \mathbf{0} \end{pmatrix} \quad | \mathbf{11} \rangle = \begin{pmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{1} \end{pmatrix}$$

- Quantum State: Bra-ket Notation

Diagram illustrating the normalization of a quantum state $|\psi\rangle$.

The state is defined as:

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

Labels: "Amplitudes" points to α and β ; "Ket" points to $|0\rangle$ and $|1\rangle$.

The normalization condition is shown as:

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

A blue arrow labeled "Normalization" points from the state equation to the normalization equation.

The bra state is defined as:

$$\langle\psi| = \alpha^*\langle 0| + \beta^*\langle 1|$$

Label: "Bra" points to $\langle 0|$ and $\langle 1|$.

- Superposition $if \begin{cases} \alpha \neq 0 \\ \beta \neq 0 \end{cases}$

- Description of space for 2 (or multiple) qubits
- Notation \otimes
- 2-qubit-state example

Product state:

$$\begin{pmatrix} a_1 \\ b_1 \end{pmatrix} \otimes \begin{pmatrix} a_2 \\ b_2 \end{pmatrix} = \begin{pmatrix} a_1 * \begin{pmatrix} a_2 \\ b_2 \end{pmatrix} \\ b_1 * \begin{pmatrix} a_2 \\ b_2 \end{pmatrix} \end{pmatrix} = \begin{pmatrix} a_1 a_2 \\ a_1 b_2 \\ b_1 a_2 \\ b_1 b_2 \end{pmatrix} = \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix}$$

In general : $|\psi\rangle = \mathbf{a}|00\rangle + \mathbf{b}|01\rangle + \mathbf{c}|10\rangle + \mathbf{d}|11\rangle$

Condition for separability: $\frac{a}{b} = \frac{c}{d}$, otherwise: „**entangled**“

n qubits \rightarrow length of vector: 2^n

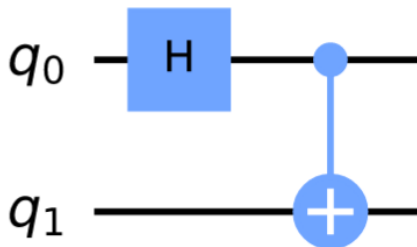
- **Correlation between states of qubits**

- One can gain information about a qubits state by knowing the states of the other qubits
- Non-entangled states can be simulated efficiently by classical computers → power of QC comes from entanglement

- **E.g.,: Bell States (completely entangled):**

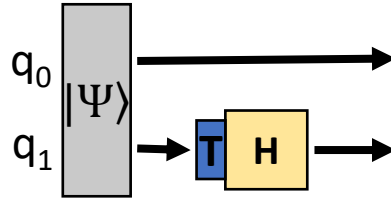
- $|\Psi_+\rangle = \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle$
- $|\Psi_-\rangle = \frac{1}{\sqrt{2}}|00\rangle - \frac{1}{\sqrt{2}}|11\rangle$
- $|\Phi_+\rangle = \frac{1}{\sqrt{2}}|01\rangle + \frac{1}{\sqrt{2}}|10\rangle$
- $|\Phi_-\rangle = \frac{1}{\sqrt{2}}|01\rangle - \frac{1}{\sqrt{2}}|10\rangle$

- Consider the following example:



- $\mathbf{H} |00\rangle = \frac{1}{\sqrt{2}} (|00\rangle + |01\rangle) = |0+\rangle$
- $\mathbf{CNOT} |0+\rangle = \frac{1}{\sqrt{2}} (|00\rangle + |11\rangle) \rightarrow \mathbf{Bell-state}$

▪ **Example:**



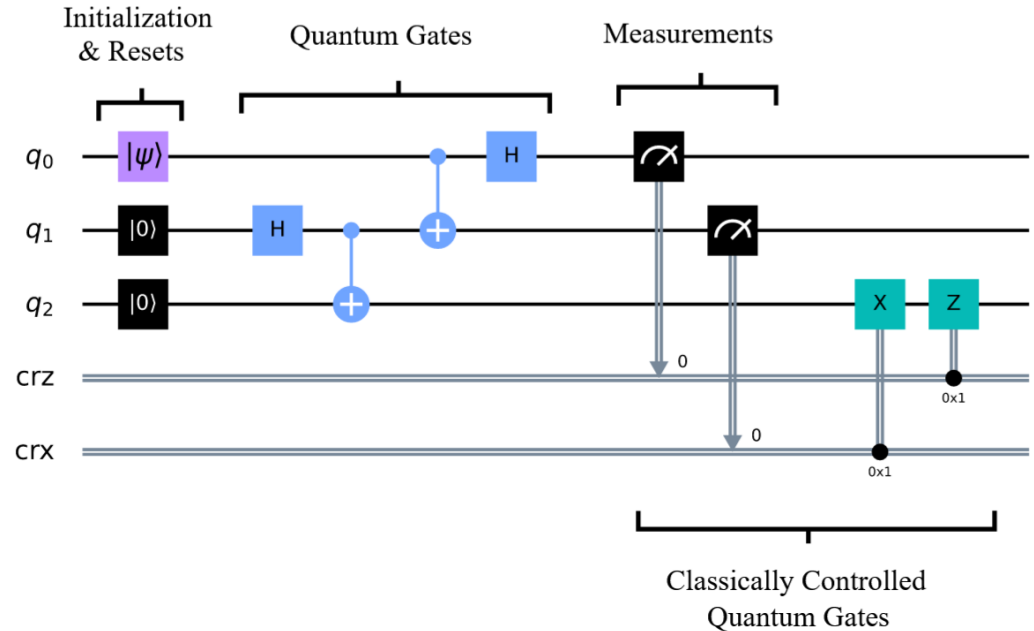
▪ **Why not just $H \begin{pmatrix} a_1 \\ b_1 \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{pmatrix} a_1 \\ b_1 \end{pmatrix}$? (\rightarrow Entanglement)**

▪ **Tensor product: $H \otimes I = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} & 1 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\ 1 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} & -1 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \end{bmatrix} =$**

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

▪ Qiskit definition:

„A **quantum circuit** is a computational routine consisting of coherent **quantum operations** on **quantum data**, such as qubits. It is an **ordered sequence** of quantum gates, measurements and resets, which may be conditioned on real-time classical computation.”

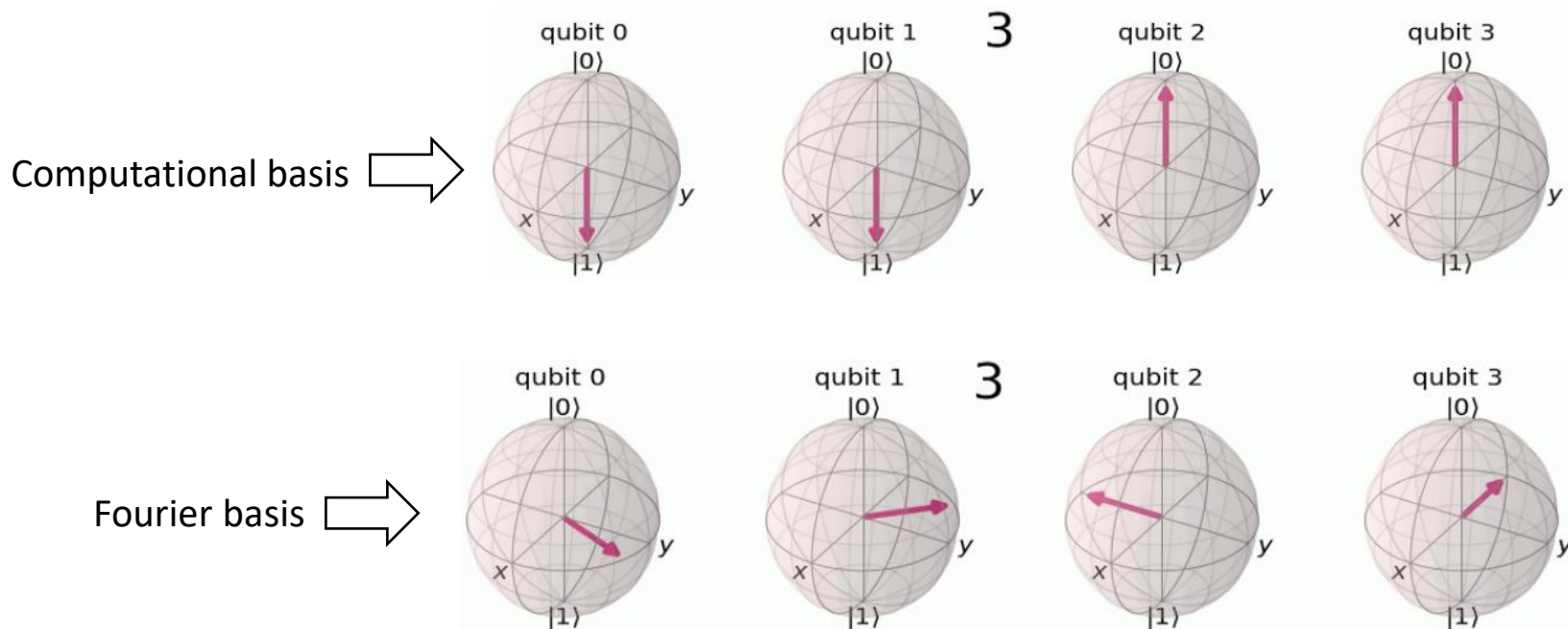


Algorithms & Application Areas



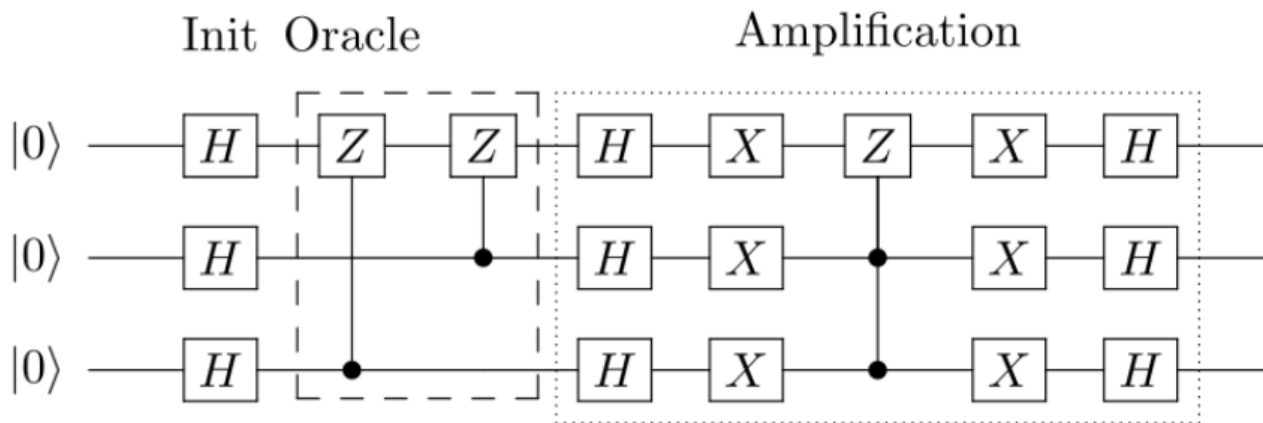
Quantum Fourier Transform

- **Quantum** implementation of **discrete Fourier transform**
- **Part** of many **quantum algorithms** (Shor,...)

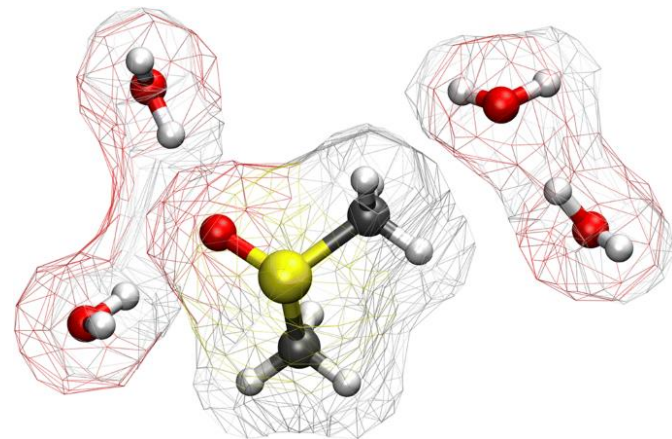


Grover-search algorithm

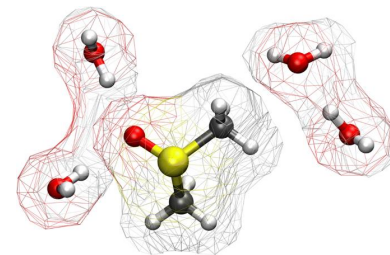
- **Database searches**, subroutine in other algorithms,...
- **Quadratic** speed-up

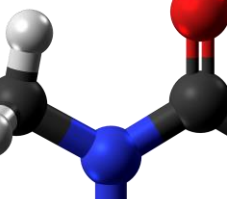


- **Closest to idea of Feynman 1981:**
 - Simulate quantum systems (molecules) with quantum systems (QC)
- **Scientific insights**
 - Quantum mechanical properties of molecular systems
 - Physiological processes (e.g., photosynthesis, DNA mutation)



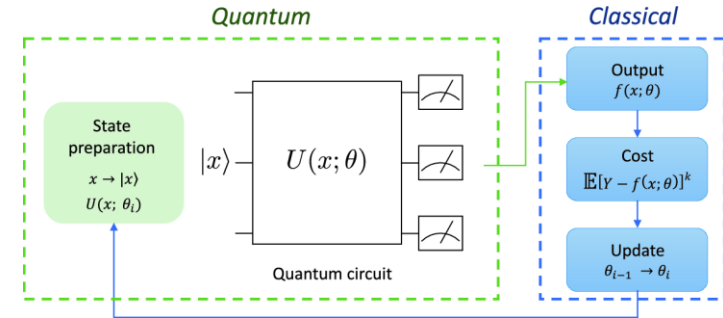
- **Simulation of molecular behaviour at quantum level:**
 - Drug design
 - Materials design
 - Development of new chemicals (e.g. catalyst in agriculture)
- **Classical approach:**
 - Calculations based on simplified model of molecule
 - Check a posteriori validity of the model



- 
- $C_8H_{10}N_4O_2$

Variational Quantum Eigensolver – VQE

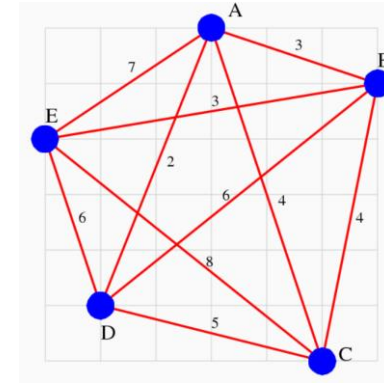
- Originally used for **quantum chemistry**
→ e.g., **ground state energy**
- Makes use of **parameterized gates (VQA)**
- **Procedure:**
 - Generate trial state with $U(\theta)$
 - Measure in computational basis
 - Calculate cost function: energy
 - Update parameters classically (e.g. gradient descent)



Application Areas – Quantum Optimization

- **Industrial relevance**

- Logistics,
- Manufacturing,
- ...



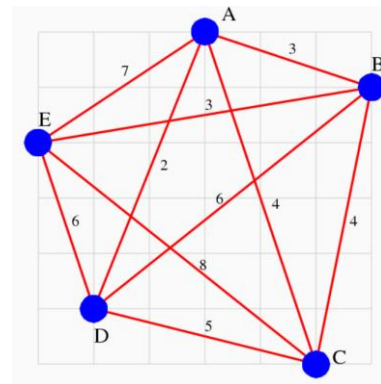
- **Optimizing operational metrics** (e.g., time, energy, fuel, cost)
- **Examples:** graph optimization, routing, scheduling
 - Usually exponentially growing search space
- **Classical computation**
 - Expensive algorithms (e.g., Brute force algorithms)
 - Use of approximative heuristics (e.g., Genetic Programming)

- **Travelling Salesman Problem**

- Visit all cities → shortest route?
- E.g., 20 cities: $20 \times 19 \times 18 \times \dots \times 2 \times 1 =$
2,430,000,000,000,000,000 combinations

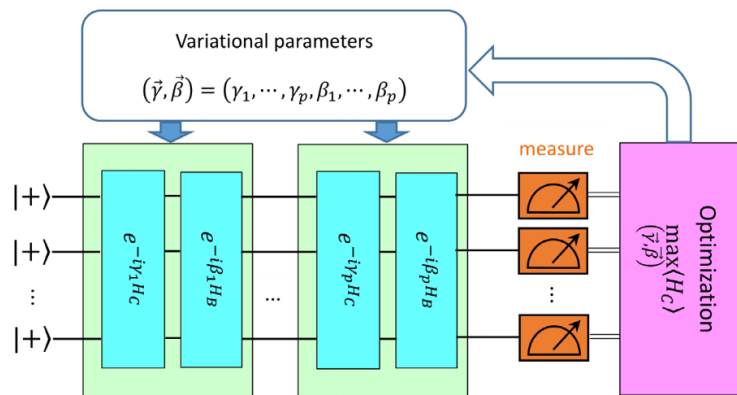
- **Quantum Computation**

- Iteratively increase probability of getting optimal result
- Usual form: Quadratic Unconstrained Binary Optimization (QUBO): $f(x) = x^T Q x$

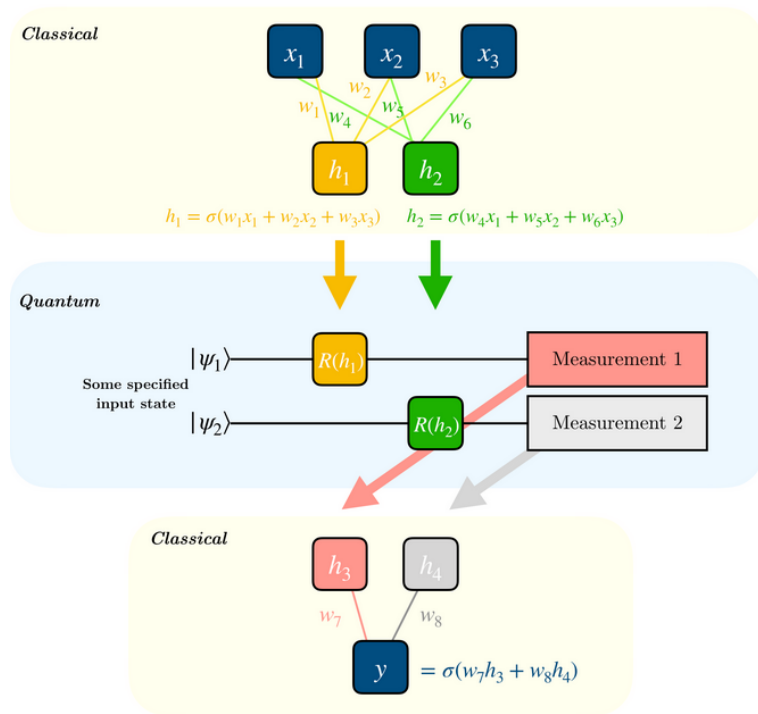


Quantum Approximate Optimization Algorithm – QAOA

- Algorithm for **combinatorial optimization** problems
- Very **similar to VQE** but with a defined ansatz
- **Procedure:**
 - Generate trial state with $U_C(\gamma), U_B(\beta)$
 - $U_C(\gamma)$: problem unitary
 - $U_B(\beta)$: mixing unitary
 - Measure in computational basis
 - Calculate cost function
 - Update parameters classically



- **Mostly quantum-enhanced ML**
 - Hybrid nature
 - Difficult subroutines outsourced to QC
 - E.g., Quantum GAN
- **Quantum Neural Networks**
 - Variational Quantum Algorithms
- **Quantum Topology Analysis**
- And many more...



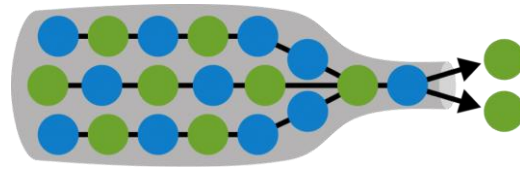
Quantum Algorithms – Requirements



Solve useful problem



**Speed-up or
other advantage**



Relatively small data



Correctness guarantees



Resources can be estimated

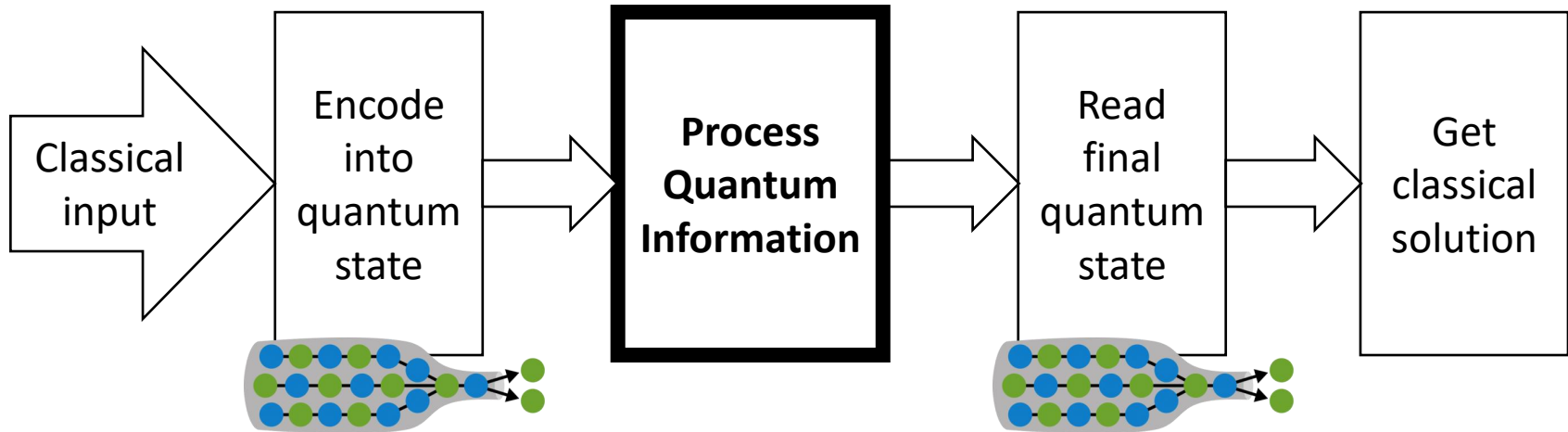


**→ goal today: find promising problem where hybrid
algorithm is better heuristic than purely classical approach**

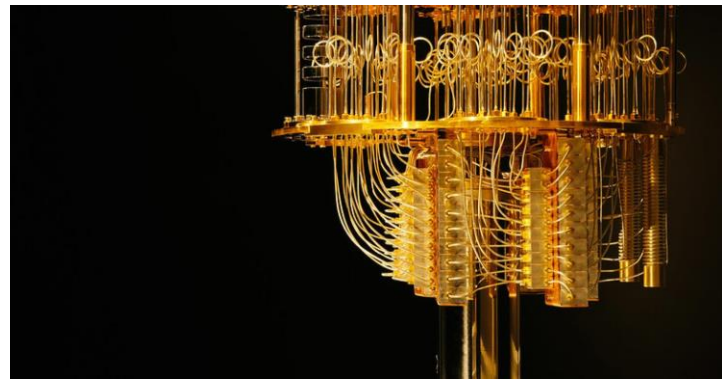
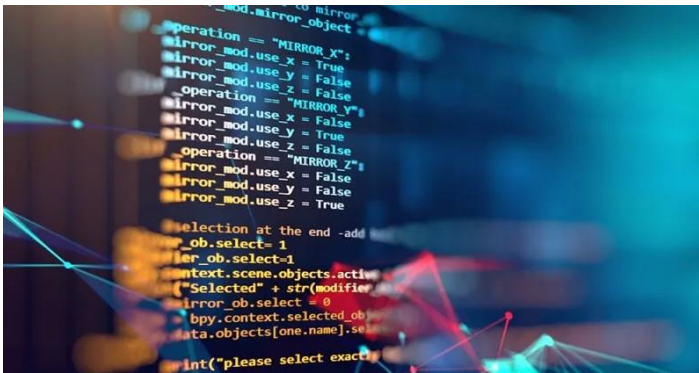
Challenges & Limitations



Quantum Information Processing – Bottlenecks







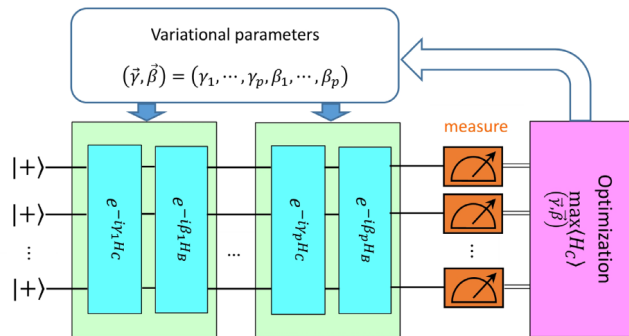
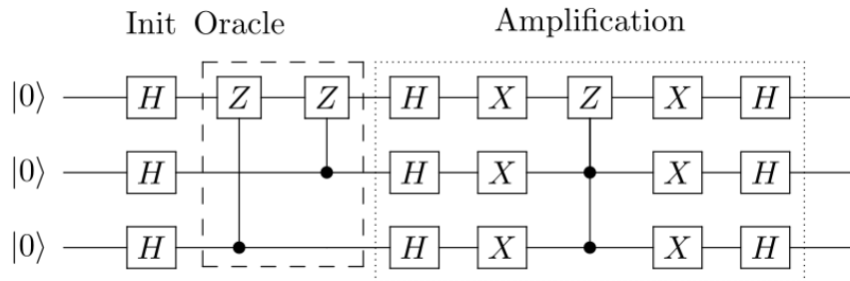
Algorithms & Software

- Dequantization
- Error correction
- Programming languages
- Compilers
- Interface to classical regime
- Standards & Protocols

Hardware

- Fidelity
- Error correction
- Scalability
- Interface to classical regime

Challenges and Limitations



Fundamental

- No copies
- No assessment of intermediate states
- Decoherence
- Analog machines: $F = f^n$
- ...

Variational Quantum Algorithms

- Exponentially small gradients
- Optimization of Parameters is NP-hard
- Requires a LOT of runs
- ...

Summary of Challenges



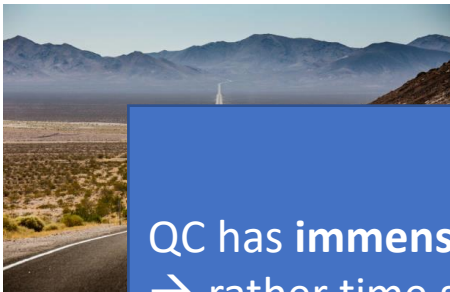
Fault-tolerant Quantum Computing:

- Provable improvement for some applications
- Requires a lot of research

NISQ-era:

- No provable improvement
- Maybe still better heuristic especially in combination with classical computing

-
- **Fidelity** has to improve drastically
 - QCs will **NEVER replace** classical ones!!!



BUT:

QC has **immense transformative potential**

→ rather time scale is questionable

→ topics still requires

- a lot of fundamental and applied research
- strong connection between levels of abstraction
 - Fundamental & applied
 - Hardware & algorithms/software

Fault-tolerant Quantum Computing

- Provable improvement in some applications
- Requires a lot of research

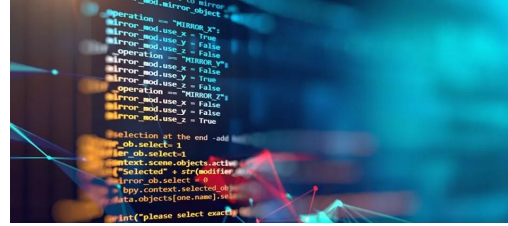
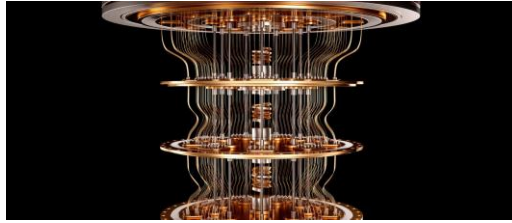
- **Fidelity** has to improve drastically
- QCs will **NEVER replace** classical ones!!!

Quantum Software Engineering



Quantum Software Engineering

Emerging Field:



Goal: apply lessons learned from classical software engineering

Problem:

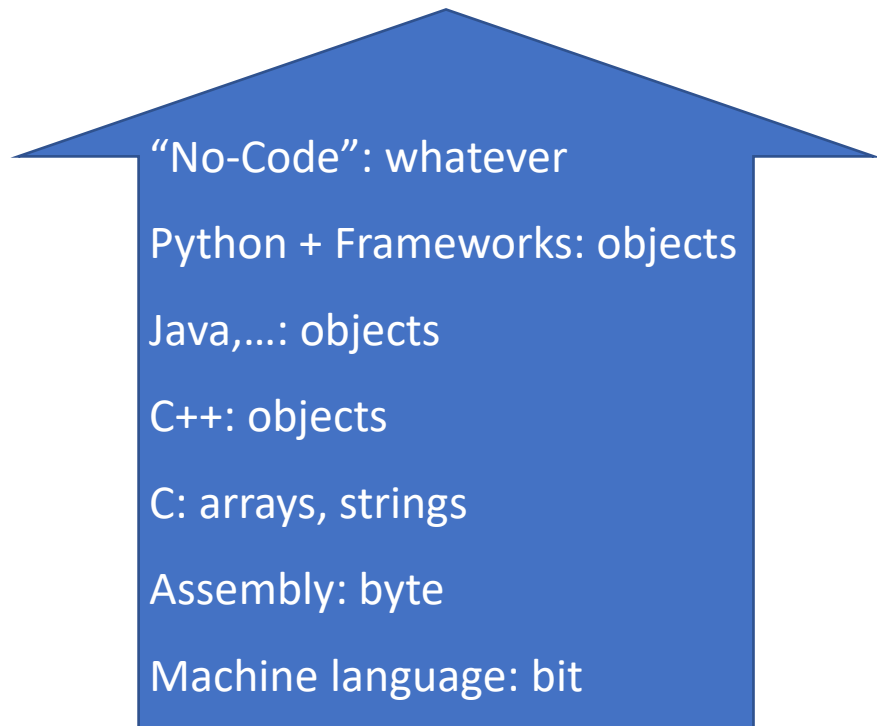
- Fundamentally different working principles
→ Requires to raise abstraction levels

Review-Article:

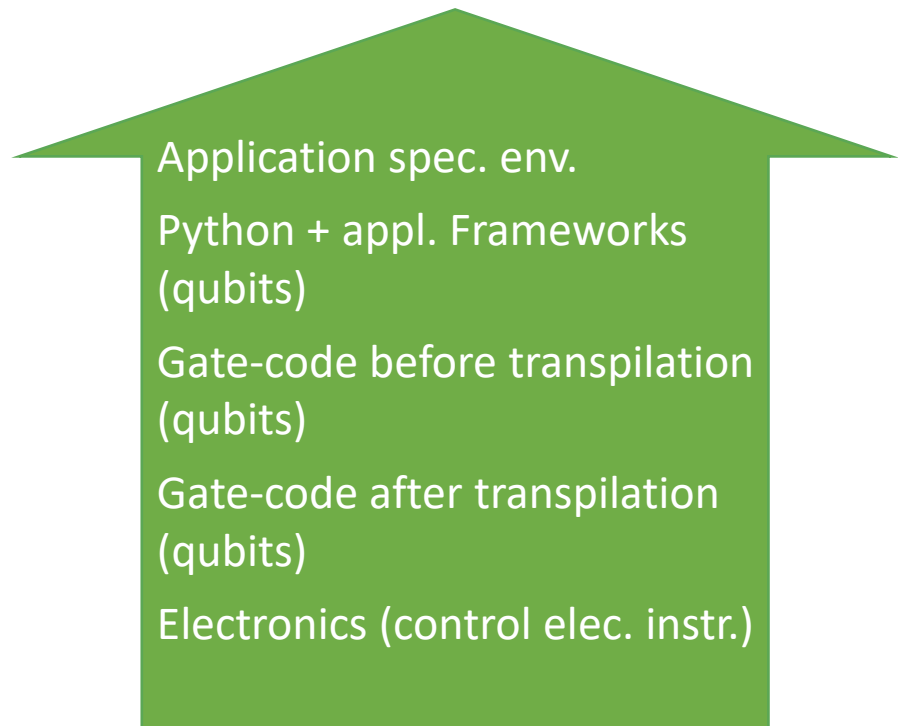
De Stefano, M., Pecorelli, F., Di Nucci, D., Palomba, F., & De Lucia, A. (2022). Software engineering for quantum programming: How far are we?. *Journal of Systems and Software*, 190, 111326.

Coding abstraction level

Classical



Quantum



→ same abstraction level with qubit gates

- Dedicated tools
 - from assembly languages to software development kits
- Vendor-specific or vendor-agnostic
 - E.g., IBM (Qiskit), Google (Cirq), D-Wave (Ocean), AWS (Braket), Microsoft (Q#)
- Mostly based on Python & open-source
- Simulator vs. real quantum computer as backend

Wrap-Up

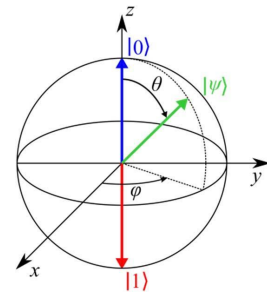
1. Motivation and Overview

- Classical computing faces severe scaling issues
- QC is applicable to a variety of computational problems
- There are diverse approaches to quantum computing



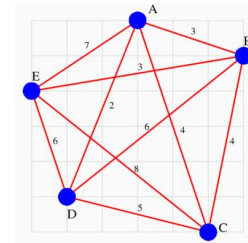
2. Basic Working Principles

- QC harnesses quantum mechanical phenomena
- Mathematically its linear algebra



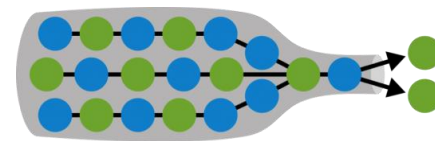
3. Near-term Applications are

- Quantum chemistry
- Quantum optimization
- Quantum machine learning



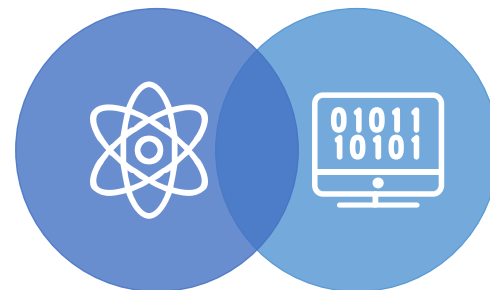
4. Challenges and Limitations

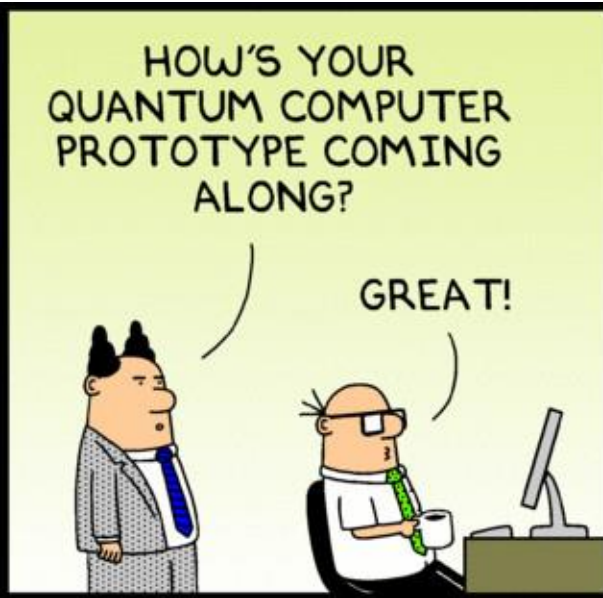
- Interesting challenges remain regarding quantum hardware, software, and their interaction
- Quantum computers will always be special purpose machines
- The potential is worth the effort



5. Quantum Software Engineering

- Which concepts from classical SE can be applied to QC?
- What are sound SE principles for engineering quantum software?
- What are quantum-specific challenges and how to consider them?





Dilbert.com DilbertCartoonist@gmail.com



4-17-12 ©2012 Scott Adams, Inc. /Dist. by Universal Uclick

