PSC

Carnegie Mellon University

University of Pittsburgh

Bridges-2 Webinar

A Hands-On Introduction to Quantum Computing with NVIDIA's CUDA Quantum

Pooja Rao

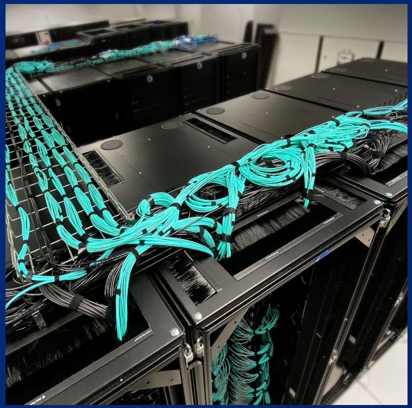
NVIDIA

April 19, 2024

© Pittsburgh Supercomputing Center, All Rights Reserved

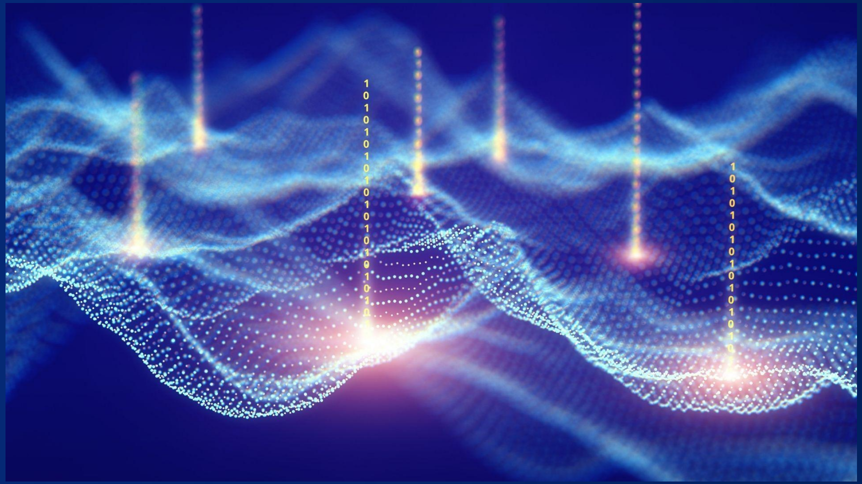
Pittsburgh Supercomputing Center

enabling discovery since 1986



The **Pittsburgh Supercomputing Center (PSC)** provides advanced research computing capability, education, and expertise to the national research community.

Since 1986, PSC has provided university, government, and industry researchers with access to some of the most powerful systems available for high-performance computing, enabling discovery across all fields of science.



OUR AREAS OF EXPERTISE

- high-performance and data-intensive computing
- data management technologies
- software architecture, implementation, and optimization
- enabling ground-breaking science, computer science, and engineering
- user support for all phases of research and education
- STEM outreach in data science, bioinformatics, and coding

Welcome!

BRIDGES-2

Hewlett Packard Enterprise is delivering *Bridges-2*

Bridges-2 Leadership Team

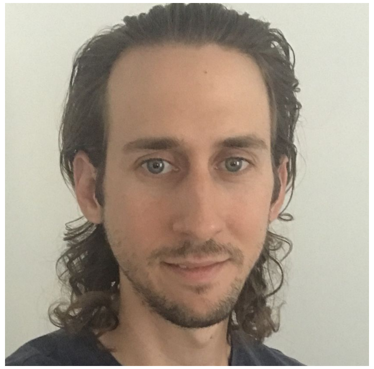
Sergiu Sanielevici
PI & Dir. Support for Sci. Apps.

Robin Scibek
Dir. Comms.
co-PI

Paola Buitrago
Dir. AI & Big Data
co-PI

Edward Hanna
Dir. Systems & Ops.
co-PI

Tom Maiden
User Services Mgr.
co-PI



Stephen Deems
Project Manager

Andrew Adams
Information Security Officer

© Pittsburgh Supercomputing Center, All Rights Reserved

Bridges-2 Webinars

- A forum for the Bridges-2 community to learn and share ideas and achievements.
- Topics and speakers of interest to work that is being done, or that may be done in future.
- Please suggest future speakers (including from your own team) and/or topics (including your own)!

Email: sergiu@psc.edu

➔ Questions and discussion after this event:

https://github.com/NVIDIA/cuda-quantum/discussions

© Pittsburgh Supercomputing Center, All Rights Reserved

Introducing today’s presenter: Dr. Pooja Rao

Dr. Pooja Rao is a Senior Quantum Application Engineer at NVIDIA, specializing in the intersection of quantum and high-performance computing. Her primary interests lie in quantum-assisted machine learning and leveraging quantum computing for fluid applications. During her doctoral and postdoctoral research, she focused on developing numerical methods and computational algorithms to simulate fluid flow across diverse scales and in various contexts. Her work ranged from turbulence modeling of fluid instabilities encountered in inertial confinement fusion to simulating fluid-particle transfer phenomena in multi-scale aerospace applications.

© Pittsburgh Supercomputing Center, All Rights Reserved

Q&A Logistics

- All of us except Pooja will be muted during his presentation.
- Please type your questions into the Zoom chat.
- We may be able to address some questions in the chat while Pooja is presenting.
- When Pooja finishes her presentation, she may wish to answer questions live during the final ~10 minutes of this webinar.
- For any remaining or follow-up questions, Pooja may engage after the webinar: *porao@nvidia.com*

© Pittsburgh Supercomputing Center, All Rights Reserved

A Hands-On Introduction to Quantum Computing with NVIDIA's CUDA Quantum

Pooja Rao, PhD, Senior Quantum Application Engineer, NVIDIA

Objectives

- Basic differences between quantum and classical information
- Building blocks of quantum circuits
- Introduction to hybrid algorithms
- Running simple circuits on Bridges-2

Prerequisites

- Basics linear algebra
- Familiarity with Python programming

Outline

Introduction to Quantum

- Governing Equations, Qubits and Representation
- Quantum Hardware
- Quantum Gates and Circuits
- Quantum Circuit Simulation
- Circuit primitives
- Variational Quantum Eigensolver

Introduction to CUDA-Q

- Overview
- Backends and Targets
- CUDA-Q in Action
- Multi-GPU Workflows

Quantum Computing Basics

Governing equations, Bloch sphere, single qubit gates, entangling gates

Governing Equation

- Quantum mechanics describes the phenomena at atomic and subatomic scales.
- Time-dependent Schrodinger equations

$$i\hbar\frac{\partial}{\partial t}|\psi(t)\rangle = H\,|\psi(t)\rangle$$

where H is the Hamiltonian and the state of the system is represented by the

$$|\psi(t)\rangle = e^{-iHt}\,|\psi(0)\rangle\,,\ where\quad U = e^{-iHt}.$$

Quantum evolution is unitary!

Unitary Operators

U is a unitary transformation if $UU^* = U^*U = I$,.i.e., inverse is its adjoint.

- Unitary operators are invertible.
- Unitary operators preserve inner product.
- The eigenvectors of a unitary matrix are orthogonal.

Quantum Computing Basics Operations

Superposition and Measurement

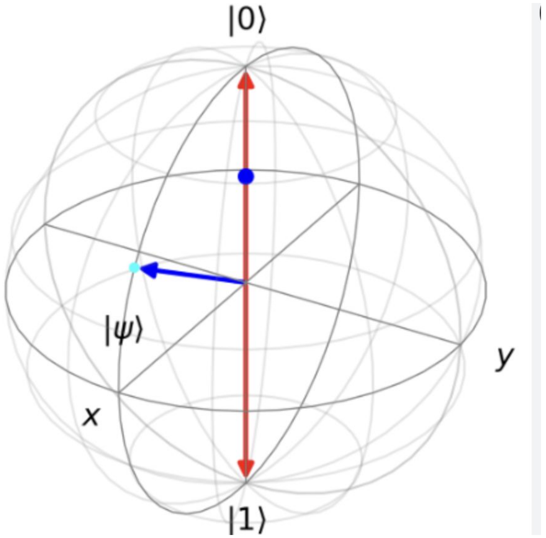
Bit

0

1

A bit is a binary electrical signal that can represent either the "0" state or the "1" state.

Qubit



A qubit can exist in "0" state, the "1" state, or any state that is a linear combination of 0 and 1.

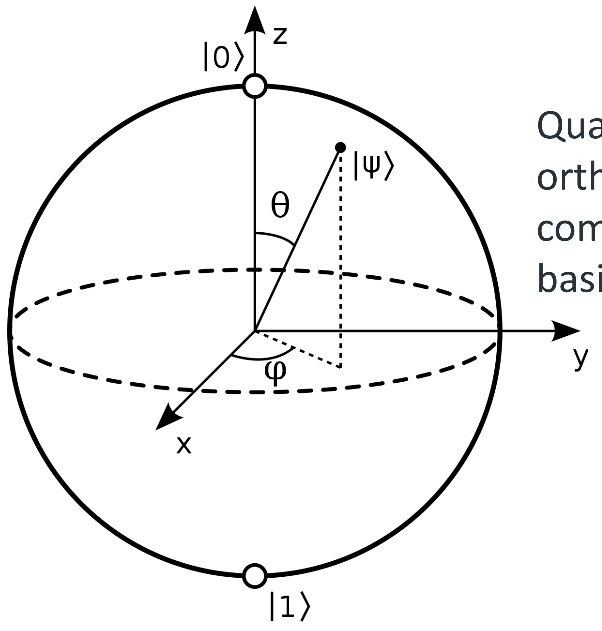
Represents the wavefunction in the Schrodinger equation.

The state of a qubit can be represented as a vector on the Bloch Sphere.

Quantum Computing Basics Operations

Superposition and Measurement

Qubit



Bloch Sphere

Quantum states |0> and |1> form orthogonal basis also called computational basis or canonical basis.

The coefficients a and b, known as the amplitudes, are complex numbers representing the qubit.

$$|\Psi\rangle = a|0\rangle + b|1\rangle = \begin{bmatrix} a \\ b \end{bmatrix}$$

Measurement: wavefunction collapse - measure only one state

$$P_0 = |a|^2$$

$$P_1 = |b|^2$$

More generally, the prob of observing state | i > :

$$P_i = |\langle\psi|i\rangle|^2$$

- Unitary transformations preserve norm, so evolved state on the sphere as well.
- Unitary transformations are reversible.

Multi-qubit Systems

Multi-qubit bases

- Single-qubit states live in a 2D Hilbert space spanned by basis vectors $|0\rangle$ and $|1\rangle$.

- Multi-qubit basis states are tensor product of single-qubit bases.

$$\begin{pmatrix} a \\ b \end{pmatrix} \otimes \begin{pmatrix} c \\ d \end{pmatrix} = \begin{pmatrix} a \begin{pmatrix} c \\ d \end{pmatrix} \\ b \begin{pmatrix} c \\ d \end{pmatrix} \end{pmatrix} = \begin{pmatrix} ac \\ ad \\ bc \\ bd \end{pmatrix}$$

Exponential Scaling in Hilbert Space

Hilbert space is a vector space with an inner product!

A qubit is represented as a complex vector of size 2 in the Hilbert space.

q0 → $c_0|0\rangle + c_1|1\rangle = \begin{bmatrix} c_0 \\ c_1 \end{bmatrix}$

A two-qubit system is a new space is which is the tensor product of two single qubit systems.

q1 q0 → $c_{00}|00\rangle + c_{01}|01\rangle + c_{10}|10\rangle + c_{11}|11\rangle = \begin{bmatrix} c_{00} \\ c_{01} \\ c_{10} \\ c_{11} \end{bmatrix}$

N qubits are represented by 2^N complex numbers.

q2 q1 q0 → $c_{000}|000\rangle + c_{001}|001\rangle + c_{010}|010\rangle + c_{011}|011\rangle + c_{100}|100\rangle + c_{101}|101\rangle + c_{110}|110\rangle + c_{111}|111\rangle = \begin{bmatrix} c_{000} \\ c_{001} \\ c_{010} \\ c_{011} \\ c_{100} \\ c_{101} \\ c_{110} \\ c_{111} \end{bmatrix}$

Quantum Phenomena

Superposition

- A quantum system can exist in multiple states or configurations simultaneously.
- Quantum computers can perform a large number of operations in parallel. (Quantum Parallelism)

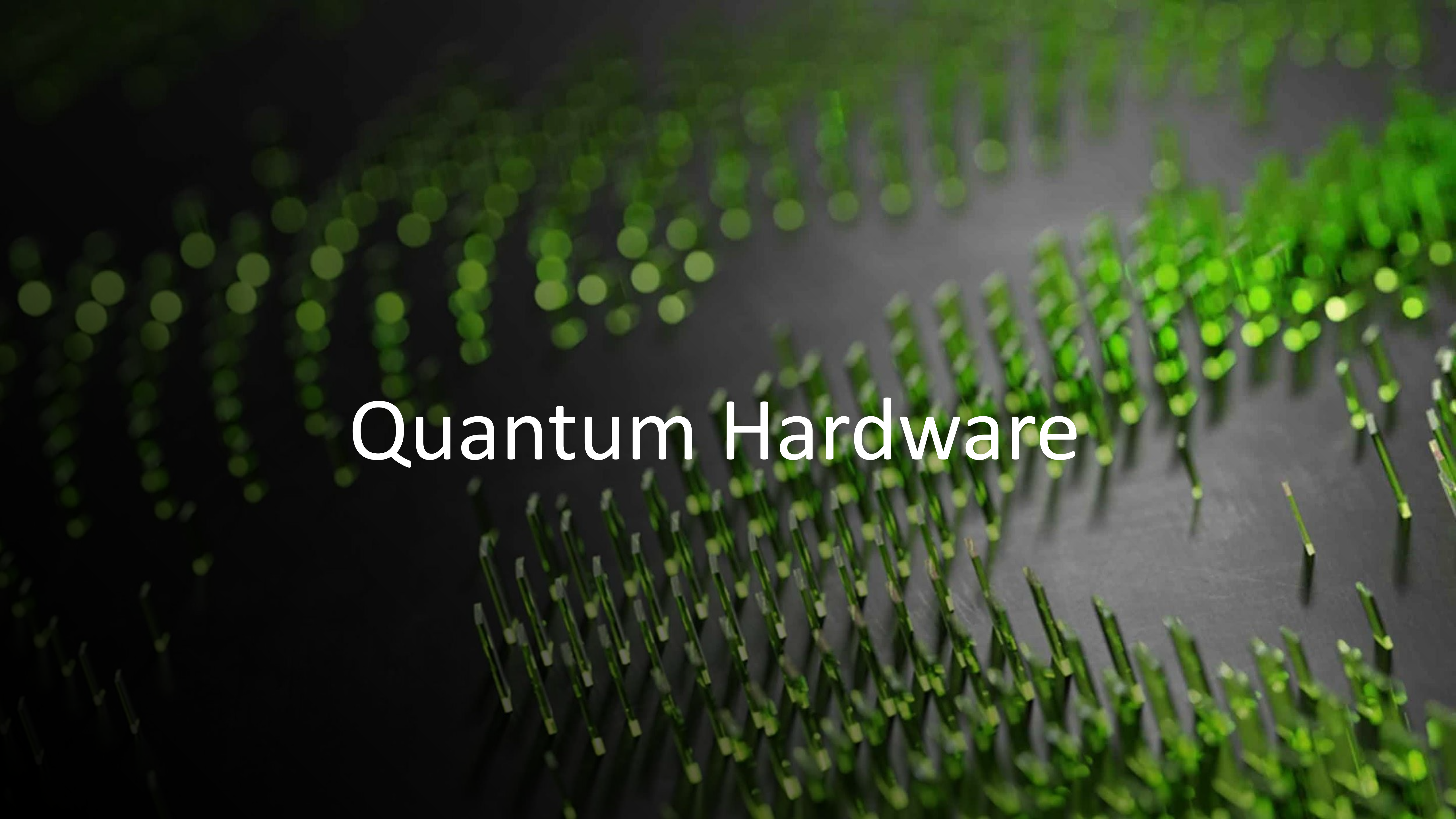
Entanglement

- Shared quantum state between two or more quantum systems.
- Wavefunction cannot be expressed as a product of individual wavefunctions for each system.
- The state of entangled objects cannot be described independently.
- A measurement on one member of an entangled pair will immediately determine measurements on its partner.

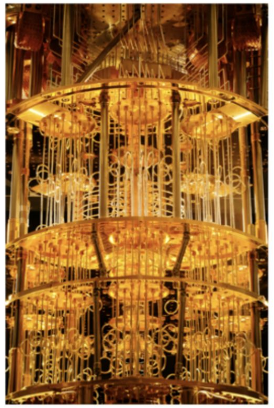
Interference

- When a particle is in a superposition of multiple states, these states can interfere with each other leading to constructive or destructive interference.
- By applying quantum gates that create superpositions of qubits, and by controlling the relative phases of the states, interference can be used to amplify certain outcomes and suppress others.

Quantum Hardware



Quantum Hardware



Source: nist.gov

Desired characteristics

Coherence

- Qubits need to be protected from the environment to preserve quantum information.

Fidelity

- Algorithms require qubit entanglement, shuffled around physical architectures, and controllable on demand.

Scalability

- To build useful applications and algorithms, the hardware needs to have a significant number of qubits and error-corrected operations.

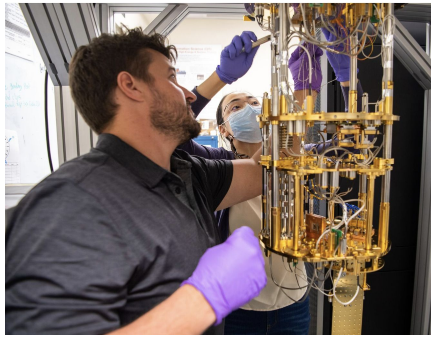
Some Common Qubit Modalities

Many ways to realize a qubit

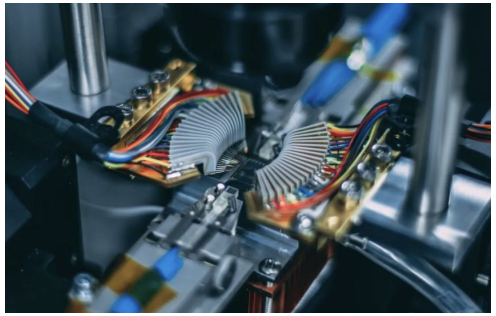
- **Superconducting qubits** - leverage Josephson junctions, enabling the transmission of supercurrent without resistance. They include variants like transmon qubits, flux qubits, and phase qubits.
- **Trapped ions qubits** - ions trapped by electric fields and manipulated with lasers. Long coherence times.
- **Photonic qubits** - based on the quantum properties of light, such as polarization and phase. They are manipulated using optical components such as beam splitters, phase shifters, and detectors.

Some other types

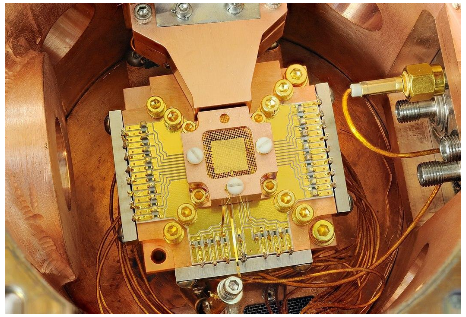
- Nuclear magnetic resonance
- Topological qubits
- Quantum dots



Superconducting (Source: PNNL)



Photonics (Source: Xanadu)



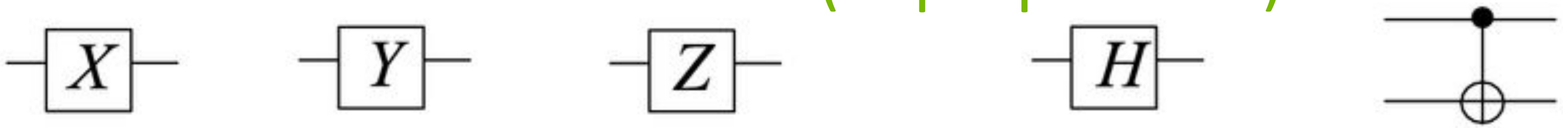
Trapped- ion (Source: wikipedia)

Quantum Circuits Basics

Important Gates

Hadamard (Superposition)

CNOT (Entangling)

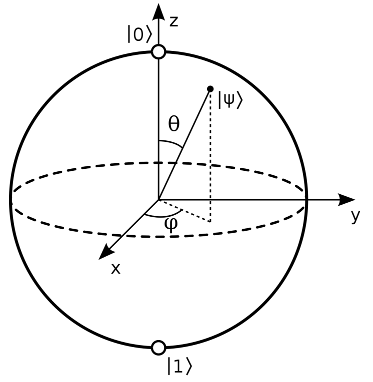


$$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \quad \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \quad \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \quad \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

X, Y, Z with Identity form the Pauli gates.

(a) **(b)** **(c)** **(d)** **(e)**

Source: Zhao, Qiang & Li, Qiong & Mao, Haokun & Wen, Xuan & Han, Qi & Li, Minghui. (2020). Fault-tolerant quantum error correction code preparation in UBQC. Quantum Information Processing. 19. 10.1007/s11128-020-02735-0.



$|0>$ and $|1>$ as the eigenvectors of the Pauli Z gate with eigenvalues 1 and -1.

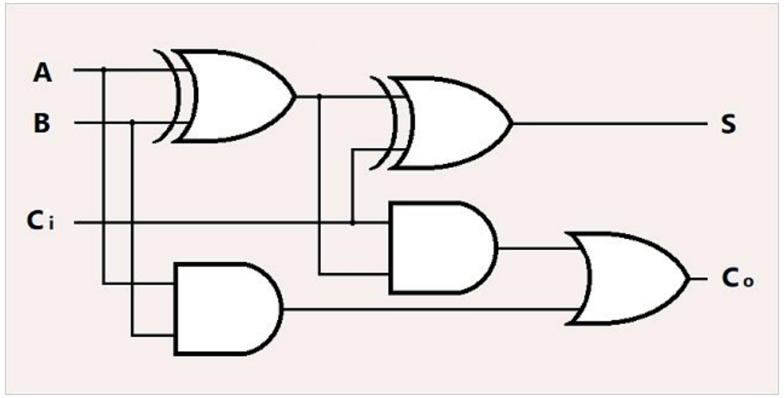
Quantum Gates

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

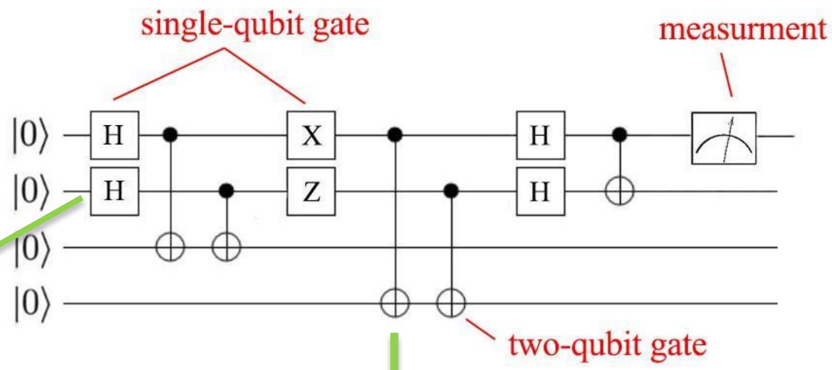
Common Quantum Logic Gates (Source: Wikipedia)

Quantum Circuits

Classical Circuit



Quantum Circuit

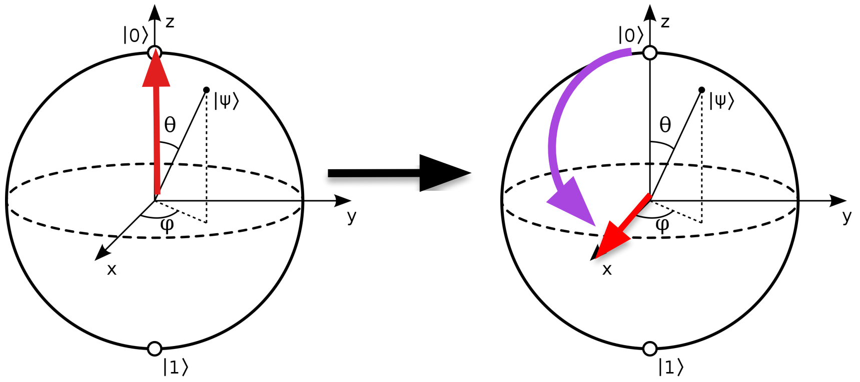


Superposition

Hadamard Gate:

Had|0⟩ = |0⟩+|1⟩

Had|1⟩ = |0⟩ - |1⟩

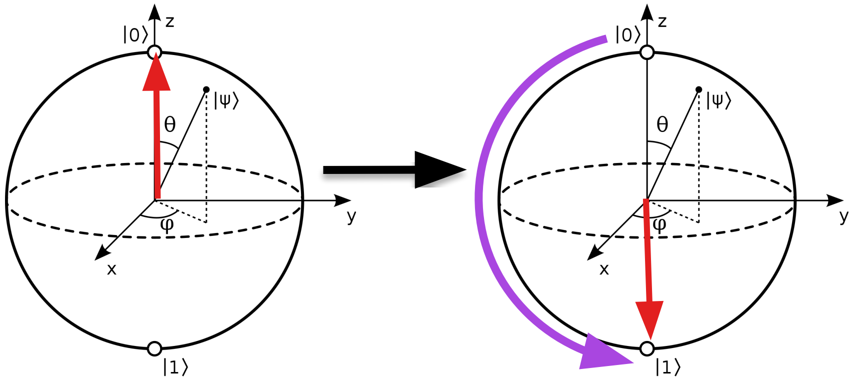


Entanglement

CNOT Gate:

CNOT|10⟩ = |11⟩

CNOT|11⟩ =|10⟩

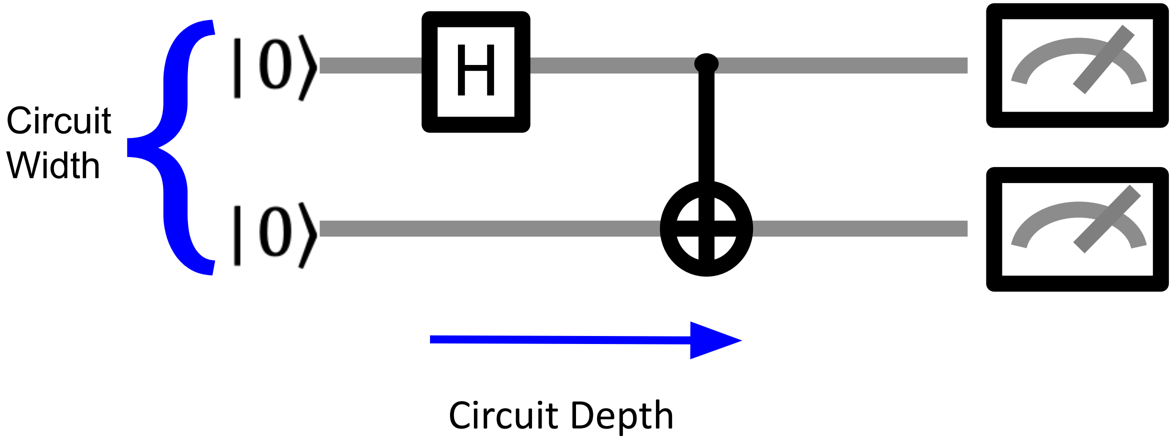


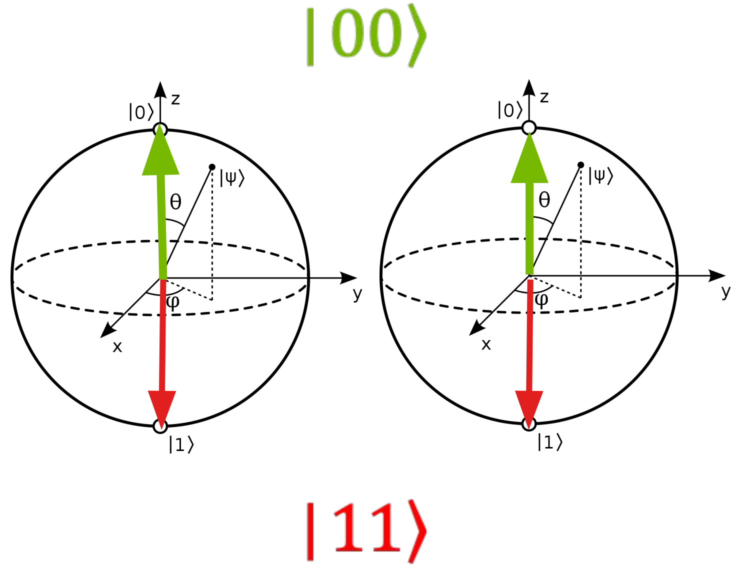
Quantum Entanglement

Bell State

Hadamard Gate:
$\text{Had}|0\rangle = |0\rangle + |1\rangle$
$\text{Had}|1\rangle = |0\rangle - |1\rangle$

CNOT Gate:
$\text{CNOT}|10\rangle = |11\rangle$
$\text{CNOT}|11\rangle = |10\rangle$





$$|\psi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle).$$

$$|00\rangle \Rightarrow |00\rangle + |11\rangle$$

An entangled state can't be factorized as a tensor product of local states.

Entanglement at Larger Scales

- Full Entangled : No qubits are independent of the others.

  E.g. Greenberg-Horne-Zeilinger (GHZ) state

$$|\psi\rangle = \frac{1}{2\sqrt{2}}(|000\rangle + |111\rangle) \neq |\psi_1\rangle \otimes |\psi_2\rangle$$

- Bipartite state : Some qubits are entangled, but not all.

$$\frac{1}{2}(|000\rangle + |100\rangle + |011\rangle + |111\rangle) = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$$

Circuit Primitives

Sampling

- Quantum mechanics is inherently probabilistic in nature.
- Running the quantum circuit repeatedly and measuring the qubits at the end of the computation.
- The result is a classical bitstring, where each bit represents the outcome of measuring a qubit.
- Provides the distribution of possible measurement outcomes.

Expectation Values Computation

- We are often interested in some physical quantity of the system, such as its energy.

- Physical quantities are related to an observable. For e.g. the energy is related to the Hamiltonian, which is a Hermitian matrix (real eigenvalues!).

- The expectation value of any operator H is given by

$$\langle\psi| \boldsymbol{H} |\psi\rangle$$

Conjugate transpose

Exercise!

Quantum Circuit Simulation

Quantum Circuit Simulator Backends

Numerically simulating quantum computation (on a classical computer) is referred to as quantum circuit simulation.

(Another meaning is the usage of real quantum devices to simulate other quantum systems. E.g. quantum chemistry simulations.)

Two common ways

i. Statevector simulator

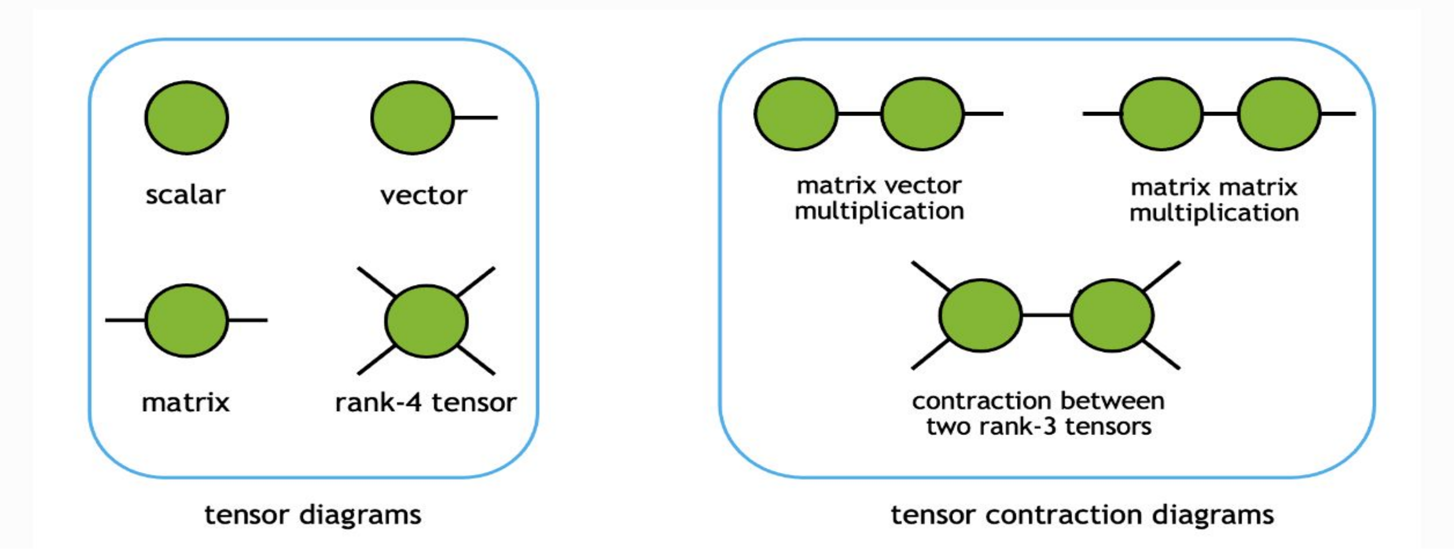
ii. Tensornetwork-based simulator

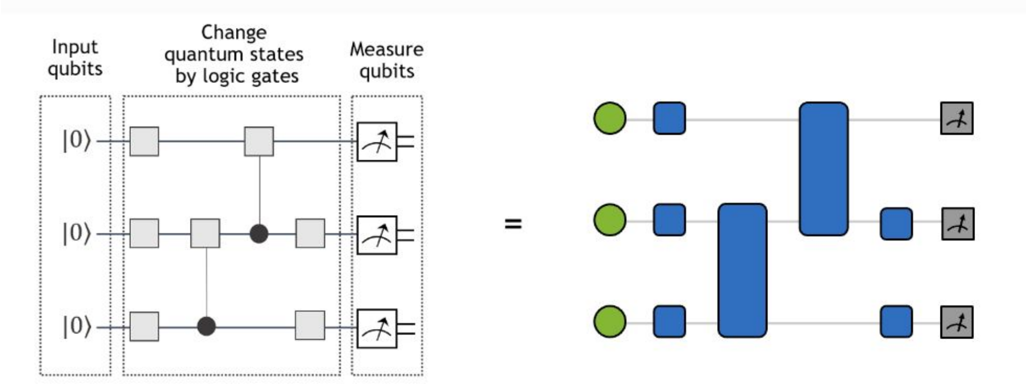
Statevector Simulator

- Matrix-vector view of quantum circuit operations and states
- Initial state is a vector and then each subsequent unitary application is a matrix-vector multiplication leading to a new vector.
- The problem size scales exponentially in the number of qubits
- Appropriate for deep circuits.

Tensor-network Based Simulator Backends

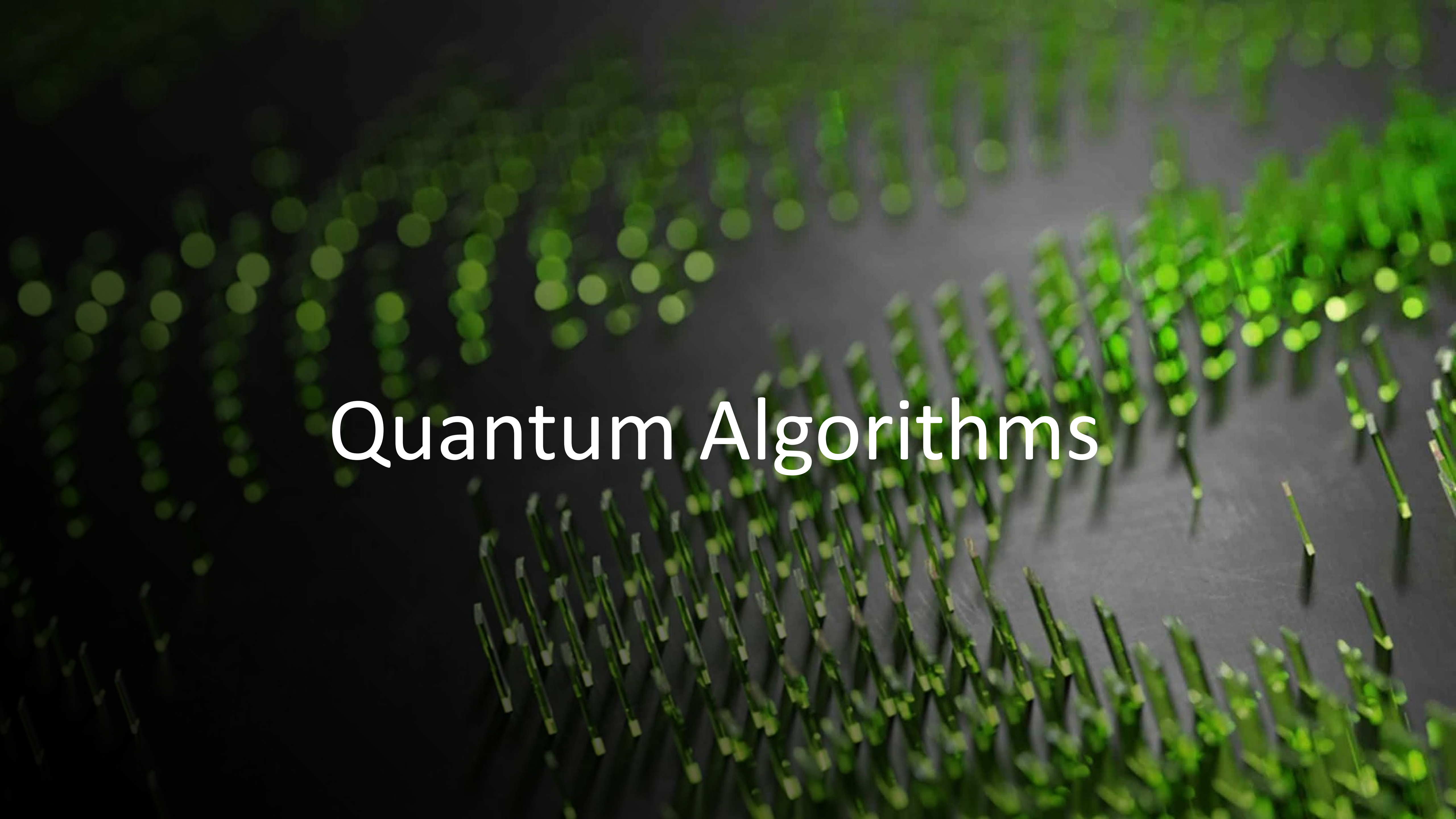
Image source:
cuQuantum webpage





- Convert the circuit to tensor network representation
- Find a cost-optimal tensor network contraction path
- Perform the actual tensor network contraction to produce the output tensor
- Higher qubit count, low circuit depth, low entanglement

Quantum Algorithms

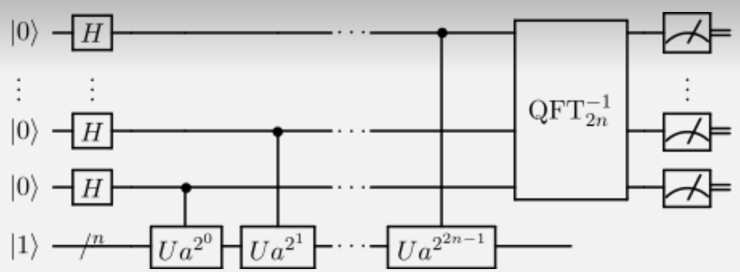


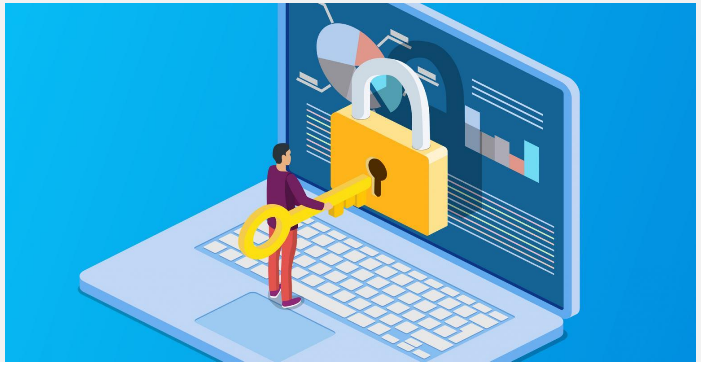
Fault Tolerant Quantum Applications

Rigorous proofs of advantage
Many “perfect” qubits required however

SHOR’S ALGORITHM

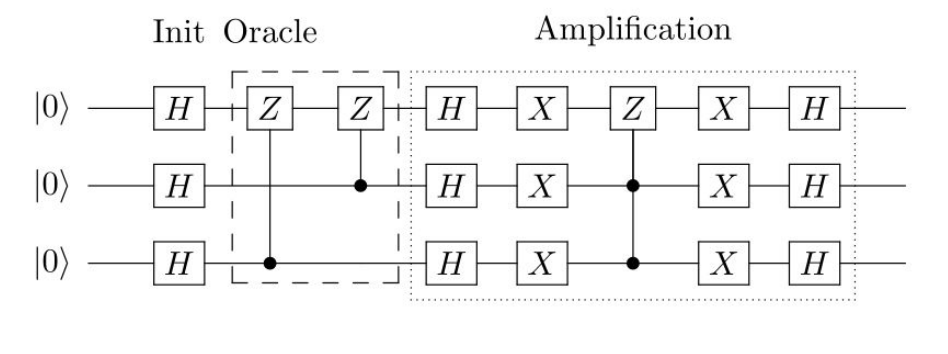
- Prime factorization of numbers - encryption
- Exponential speed-up

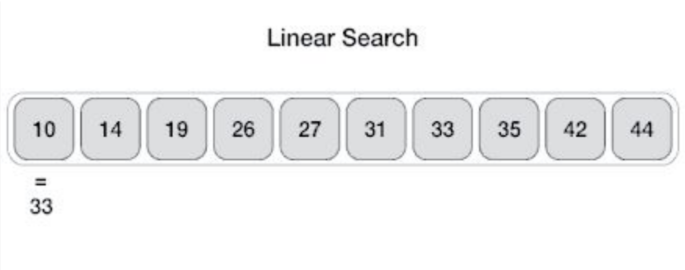




GROVER’S ALGORITHM

- Unstructured search
- Quadratic speed-up





Hardware Challenges : limited qubit count and connectivity, noise.

Hybrid Algorithms

Quantum Accelerated Hybrid Computing

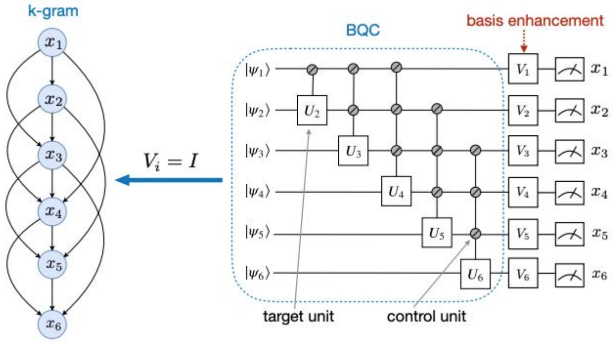
- Quantum computing is not going to be better for everything, just for some things.
- Most of the workflows in the future are going to be hybrid.
- Requires closely integrating classical computing with quantum processing.
- Examples: variational algorithms, AI enabled quantum algorithms, quantum error correction.

Near Term Quantum Computing Use-cases

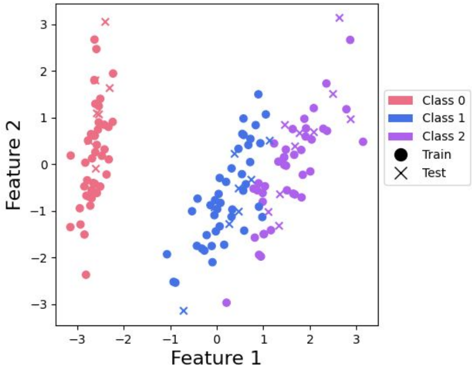
Applications with near term potential but quantum advantage is an open question

Quantum Machine Learning

Quantum Generative Models

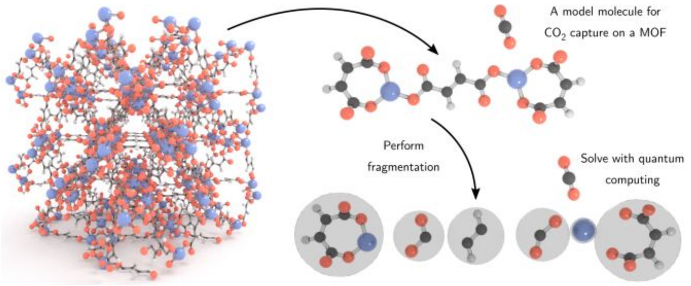


Quantum Support Vector Machine

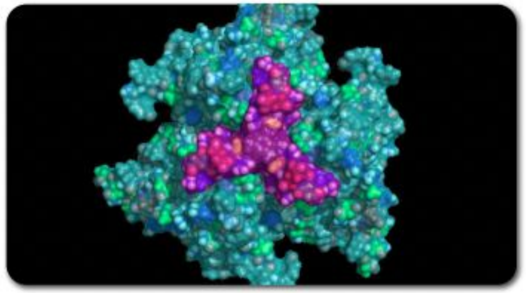


Quantum Chemistry

Variational Quantum Eigensolver for carbon capture



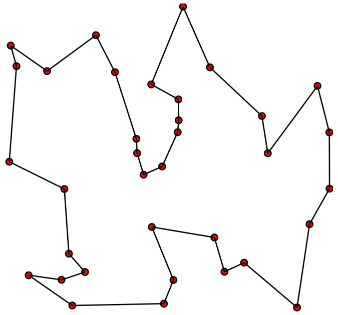
Protein folding



Combinatorial Optimization

QAOA for resource allocation

Logistics optimization



Gao, et al, Phys. Rev. X **12**, 021037
Pennylane.ai

Greene-Diniz, et al, arXiv:2203.15546,
Menten.ai

Image from ibm.com
Wikipedia.com

Variational Quantum Eigensolver (VQE)

- Chemistry problem: What is the ground state energy of a molecule?
- Solving is equivalent to finding the min eigenvalue of the Hamiltonian matrix H of size $2^n \times 2^n$.
- Classically, diagonalize the exponentially-sized matrix.
- Quantum phase estimation can solve in poly-time. Not suitable for current hardware.
- VQE was published in 2014 as a NISQ algorithm.

Variational Quantum Eigensolver (VQE)

Variational Method

$\langle\psi| H |\psi\rangle$ is an overestimate of the lowest eigenvalue.

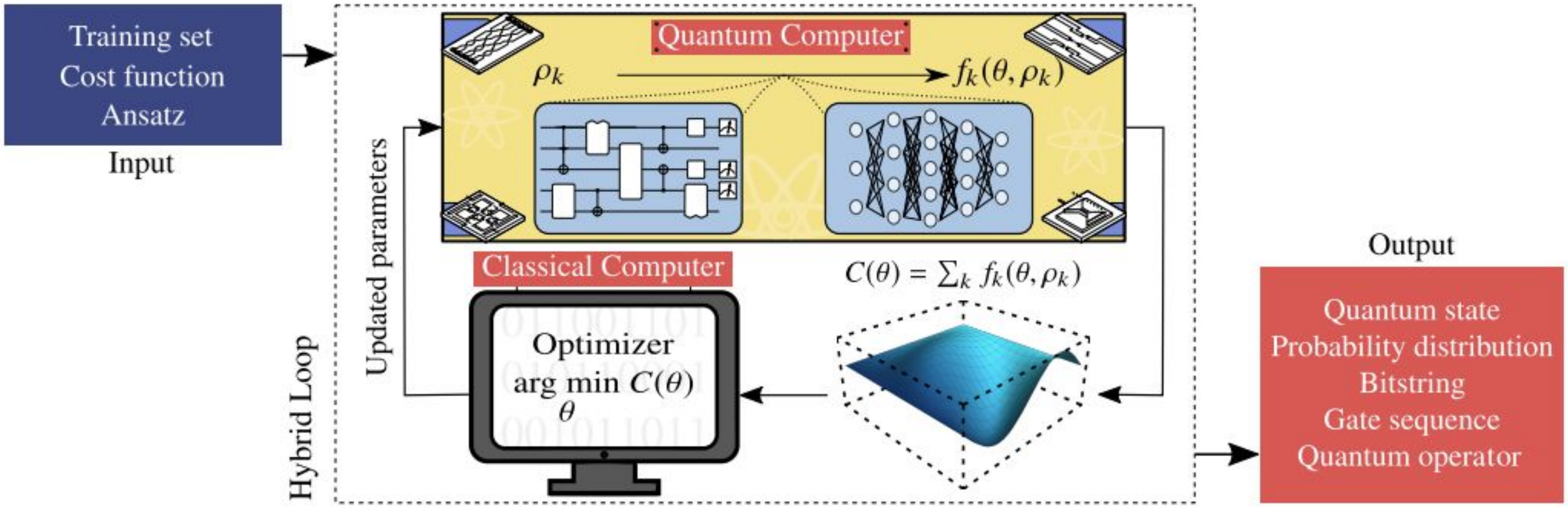
Algorithm

1. Initial state prep $|\psi(\vec{\theta})\rangle$

2. Measure $\langle\psi(\vec{\theta})| H |\psi(\vec{\theta})\rangle$

3. Optimize $\langle\psi(\vec{\theta})| H |\psi(\vec{\theta})\rangle$ over $\vec{\theta}$ using a classical optimizer.

Schematic of a Variational Algorithm

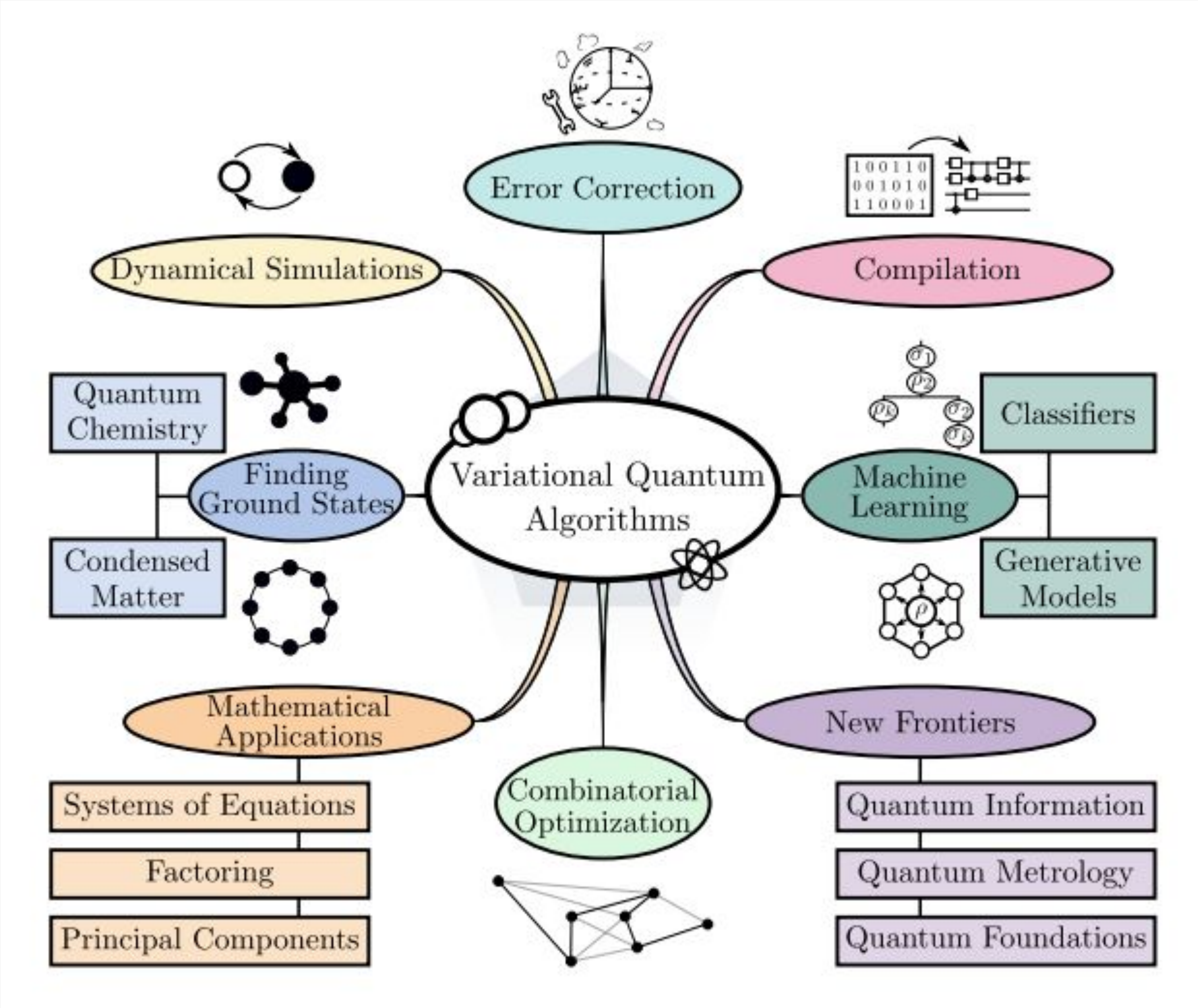


Source: Variational quantum algorithms, https://arxiv.org/abs/2012.09265

Variational Quantum Eigensolver (VQE)

- The task for quantum computers is to estimate the expectation value.
- The number of possible state vectors spans an exponentially large Hilbert space, it is important to restrict the family of candidate states (ansatz).
- The expectation value is minimized wrt parameters describing the trial wavefunction using a classical optimizer.
- Iteratively find the approximate lowest-energy state within a set of states described by the trial wavefunction.

Variational Algorithm Use-cases



Source: Variational quantum algorithms, https://arxiv.org/abs/2012.09265

Challenges

- The Hamiltonian is decomposed into tensor product of Paulis (Pauli Strings) that can be measured by NISQ devices and then, each of these quantities is estimated by measuring quantum circuits.
- The number of the Pauli strings scales as $O(N^4)$ for a Hamiltonian of electronic state in a molecule, where N is the number of qubits.
- Increased number of parameters could lead to optimization issues like barren plateau problem.

Code Repository for Hands-On

https://github.com/poojarao8/psc-workshop

CUDA Quantum : Quick Overview

NVIDIA, HPC, and Quantum Computing

Integrate quantum computers seamlessly with the modern scientific computing ecosystem

- HPC centers and many other groups worldwide are focused on the integration of quantum computers with classical supercomputers
- We expect quantum computers will accelerate some of today's most important computational problems and HPC workloads
  - Quantum chemistry, materials simulation, AI
- We also expect CPUs and GPUs to be able to enhance the performance of QPUs
  - Classical preprocessing (circuit optimization) and postprocessing (error correction)
  - Optimal control and QPU calibration
  - Hybrid workflows
- Want to enable researchers to seamlessly integrate CPUs, GPUs, and QPUs
  - Develop new hybrid applications and accelerate existing ones
  - Leverage classical GPU computing for control, calibration, error mitigation, and error correction

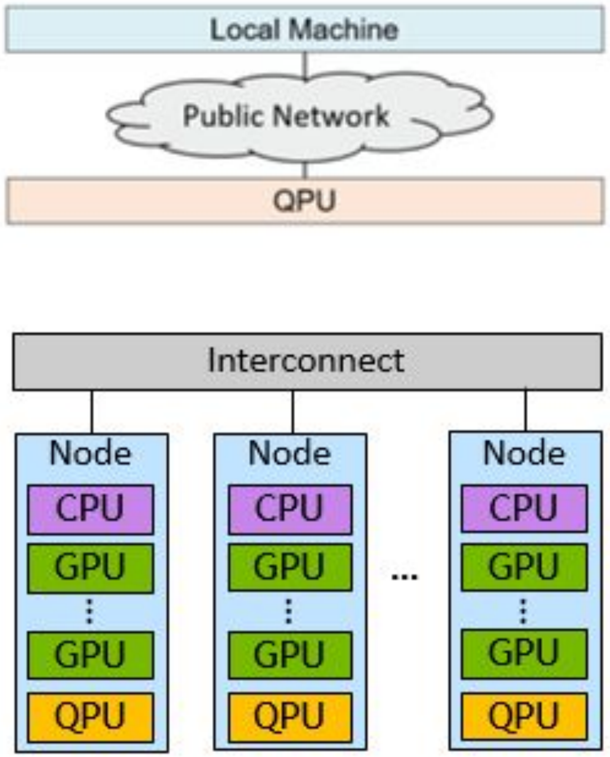
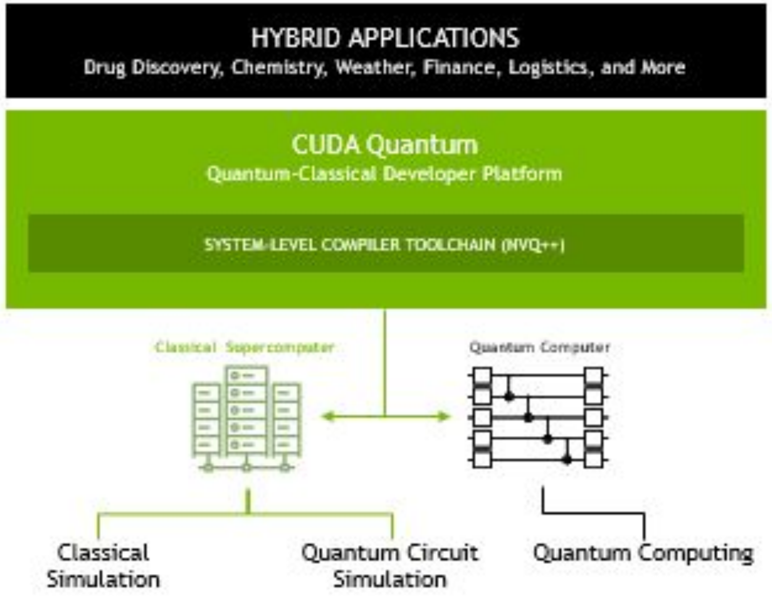


Figure adapted from:
Quantum Computers for High-Performance Computing.
Humble, McCaskey, Lyakh, Gowrishankar, Frisch, Monz.
IEEE Micro Sept 2021. 10.1109/MM.2021.3099140

CUDA Quantum: OSS Platform for Quantum Accelerated Supercomputing

Develop applications for integrated quantum-classical computing

CUDA QUANTUM PLATFORM



CUDA QUANTUM FEATURES

Single source Python and C++ programming models

High-performance compiler for hybrid GPU/CPU/QPU systems

QPU Agnostic - Works with any type of QPU, emulated or physical

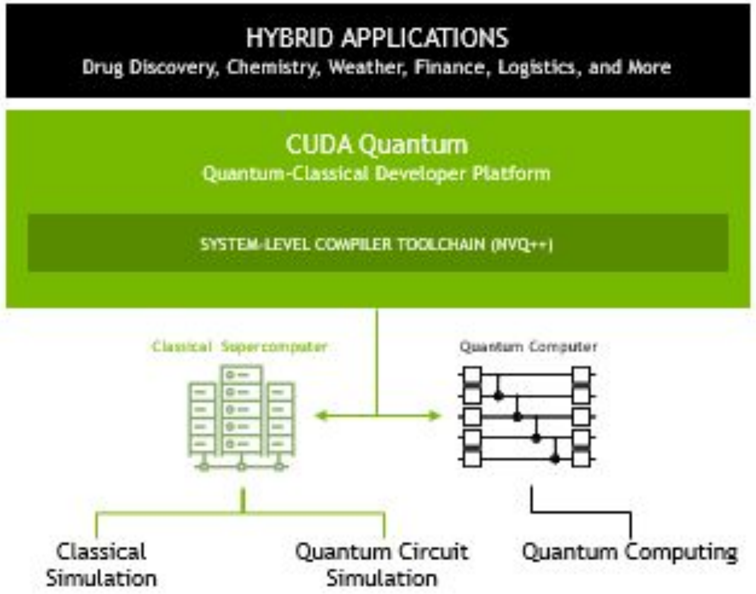
Interoperable with leading scientific computing and AI tools

github.com/nvidia/cuda-quantum | https://pypi.org/project/cuda-quantum/

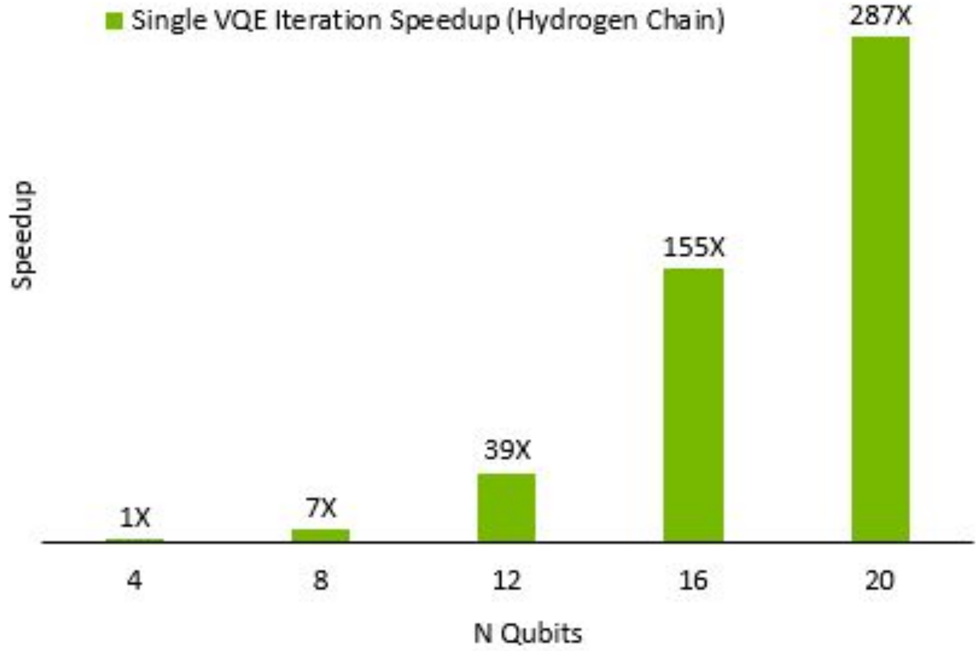
CUDA Quantum: OSS Platform for Quantum Accelerated Supercomputing

Develop applications for integrated quantum-classical computing

CUDA QUANTUM PLATFORM



CUDA QUANTUM FEATURES



github.com/nvidia/cuda-quantum | https://pypi.org/project/cuda-quantum/

Scale and Performance with Multiple GPUs

Single (large) QPU

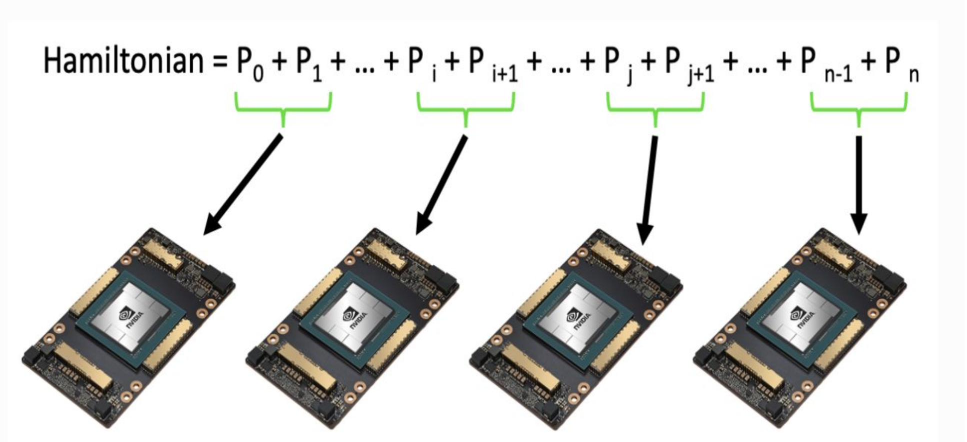
- Distribute the statevector over multiple gpus over multiple nodes.
- Mimics a single large QPU
- Appropriate for pushing the qubit count up.
- Exponential scaling of the vector space limits scaling.

Multi-QPUs

- Distribute the circuit execution over multiple-gpus over multiple-nodes.
- Each gpu acts as a virtual QPU.
- Concurrent and asynchronous execution leads to faster runtimes.
- E.g. expectation value of multi-term Hamiltonian.

Note: The remote-mqpu backend combines these two functionalities.

Multi-gpu workflows:
https://nvidia.github.io/cuda-quantum/0.6.0/examples/python/tutorials/multi_gpu_workflows.html



Simulators and QPU Backends

Simulators

- State vector
  - Uses cuQuantum cuStateVec
  - Limited by memory – 2^n to represent n qubits.
  - Targets: ‘nvidia’, ‘nvidia-mgpu’, ‘nvidia-mqpu’. ‘remote-mqpu’
- Tensor network
  - Uses cuQuantum cuTensorNet
  - Can simulate 1000s of qubits
  - Works well for sparse, low entangled problems
  - Can run on multiple GPUs
  - Target: ‘tensornet’
- Matrix Product State (MPS)
  - Approximate tensor network method
  - Target: ‘tensornet-mps’

QPUs

- Quantinuum
- IonQ
- IQM
- Oxford Quantum Circuits (OQC)

Simulation Backends:
https://nvidia.github.io/cuda-quantum/0.6.0/using/simulators.html

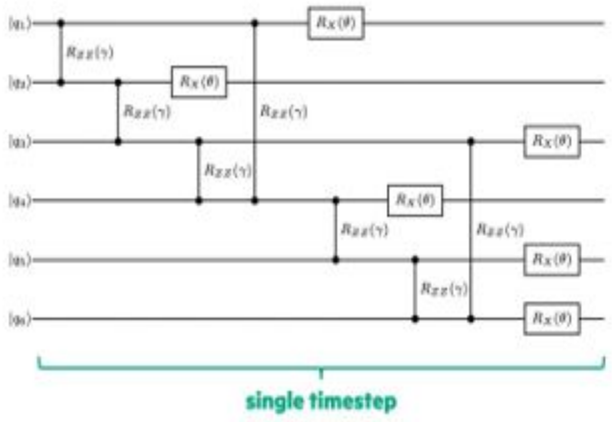
CUDA Quantum in Action

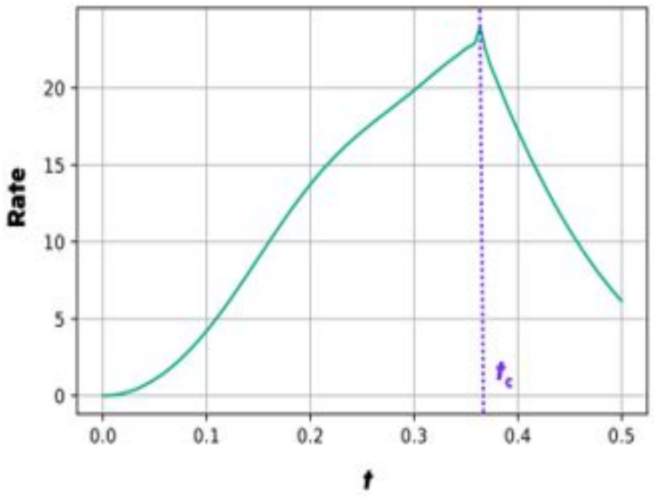
Speed-ups for time-evolution of the transverse field Ising model (TFIM)

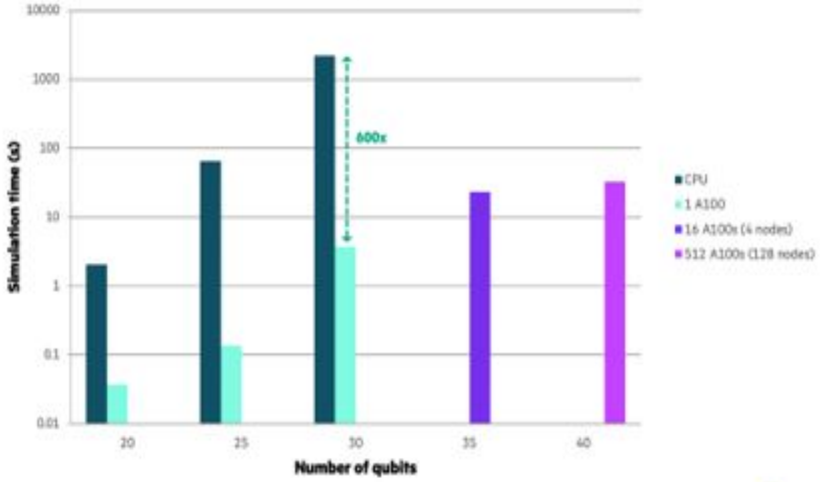
- Collaboration with Hewlett Packard Labs
- Study dynamical quantum phase transitions
  - Requires computation of overlap of initial state with time evolved state
- Leverage NVIDIA multi-node, multi-GPU simulation backend.
  - Distributed state-vector simulator
- 600x performance increase over multi-threaded CPU approaches

```
@cudaq.kernel()
def tfimEvolve(timeStep: float, params: list[float]):
    qubits = cudaq.qvector(40)
    ... Circuit, use input params ...

for time in range(finalTime):
    state = cudaq.get_state(tfimEvolve, time, params)
    overlaps.append(state.overlap(initialState))
```



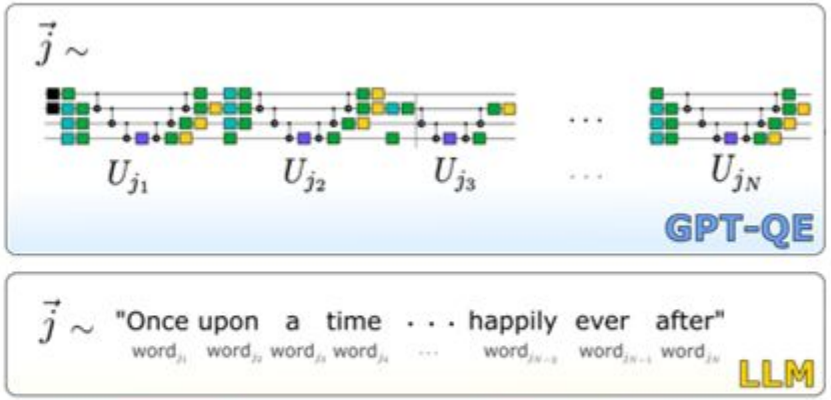




CUDA Quantum in Action

GPT-QE - University of Toronto and St. Jude Children's Research Hospital with CUDA Quantum

- Developed a novel Generative Pre-Trained Transformer-based (GPT) method for computing the ground-state energy of molecules of interest
- The first demonstration of a GPT-generated quantum circuit in the literature
- A powerful example of leveraging AI to accelerate quantum computing
- Executed using CUDA Quantum on A100 GPUs on Perlmutter
- Opens the door to a wide variety of novel Generative Quantum Algorithms (GQAs) for drug discovery, materials science, and environmental challenges



https://arxiv.org/pdf/2401.09253.pdf

UNIVERSITY OF TORONTO

St. Jude Children's Research Hospital
Finding cures. Saving children.

Thank you. Questions?