CUDA-Qが拓く量子古典ハイブリッド計算の未来

Ikko Hamamura | Quantum Algorithm Engineer, NVIDIA

第5回 スーパーコンピュータ「不老」 ユーザ会 | Oct 2, 2024

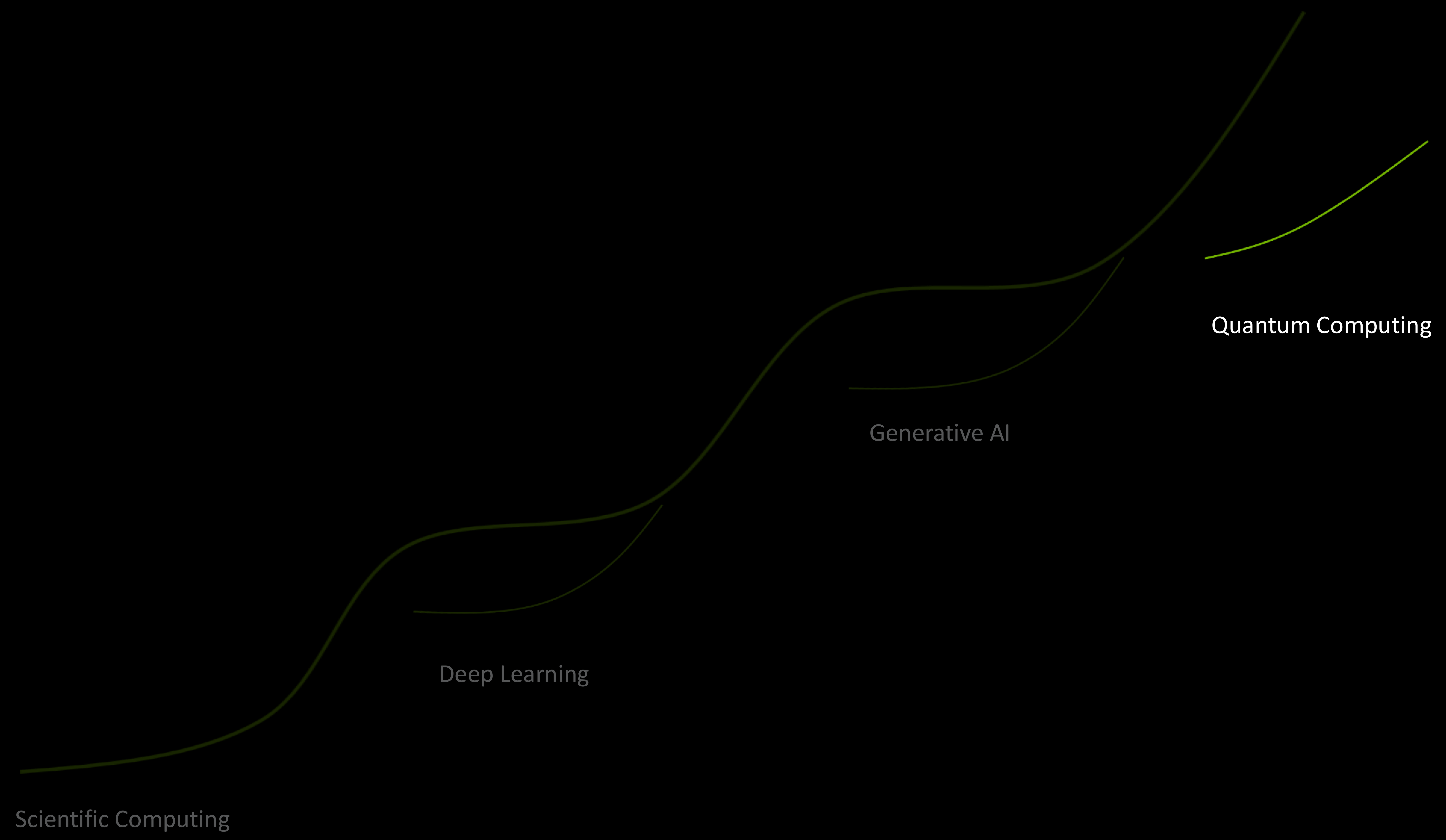
NVIDIA is not building
Qubits

NVIDIA is building all

Accelerated Quantum Supercomputers

NVIDIA's History of Enabling Computing Revolutions

NVIDIA's History of Enabling Computing Revolutions



NVIDIA Accelerates Quantum

Access to HPC for the whole quantum ecosystem

QPU vendors | App developers | Academics | ...

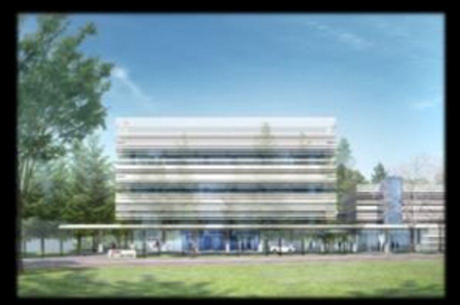
NVIDIA Quantum

Qubit agnostic

PASQAL
QUANTUM BRILLIANCE
|QuEra> Computing Inc.
XANADU
IONQ
rigetti
QUANTINUUM
ORCA Computing
ANYON
ATOM COMPUTING
OQC
IQM

Integrated with HPC

Gefion | NNF | Denmark



ABCI-Q | AIST | Japan

Julich SCC | Germany

Pawsey SCC | Australia

Broad partner network

>90%
Largest Startups Working with NVIDIA

>75%
QPUs Integrating NVIDIA software

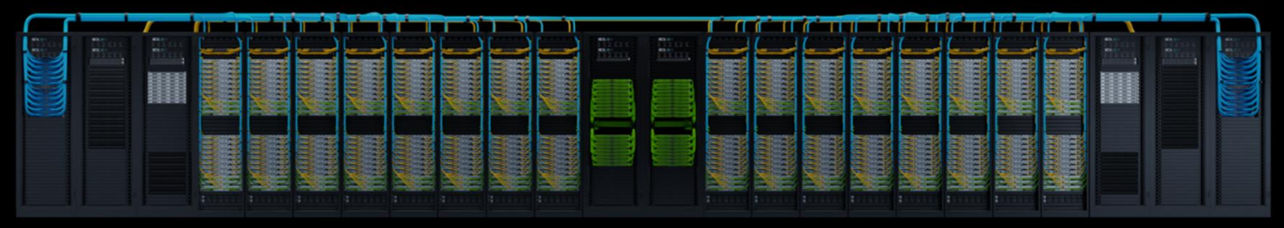
15/17
Leading Quantum Development Frameworks Accelerated

Quantum Accelerated Supercomputing

Supercomputers are the foundation of Quantum R&D

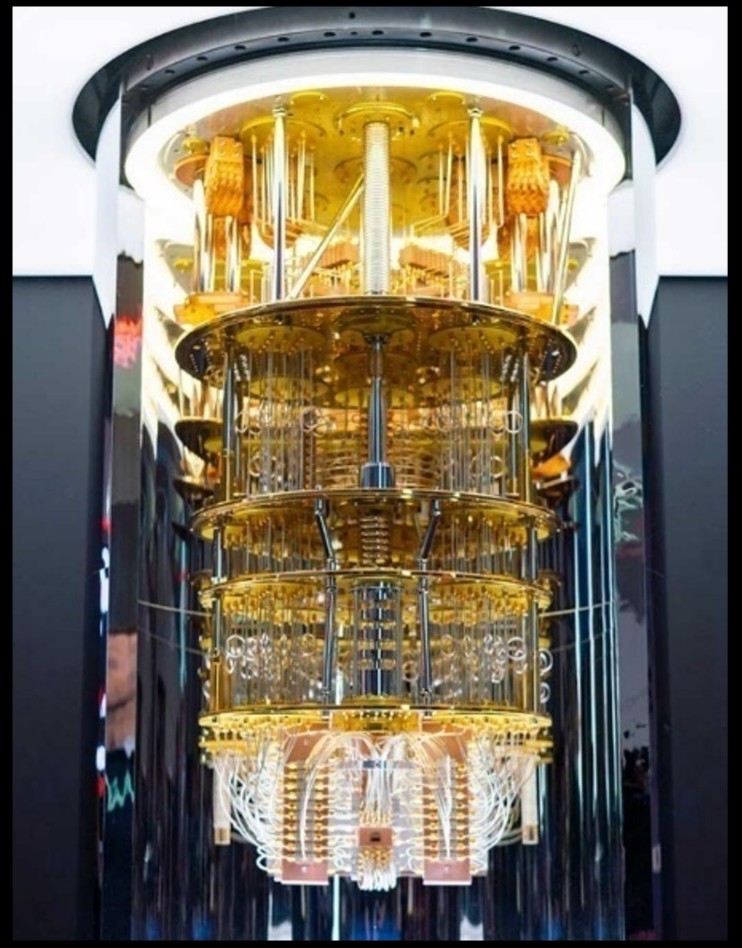
Simulation

- Quantum computers are small and error-prone -> simulation
- **Today:** Powerful simulators enable algorithm and application R&D – new approaches (e.g. tensor networks)
- **Future:** Digital twins of quantum computers for design and architecture optimization



HPC Quantum Integration

- Useful quantum computing will be hybrid
- **Today:** Enable domain scientists to start developing for QPUs, enable quantum researchers to use accelerated computing
- **Future:** quantum computers will integrate tightly with supercomputers as accelerators and be co-programmed



AI for Quantum

- Error correction, calibration, control, compilation are challenging computationally, real-time compute often needed
- Accelerated computing and AI can solve these problems
- **Today:** Enable AI research for all of the above
- **Future:** Hybrid Quantum+AI supercomputer with low-latency link

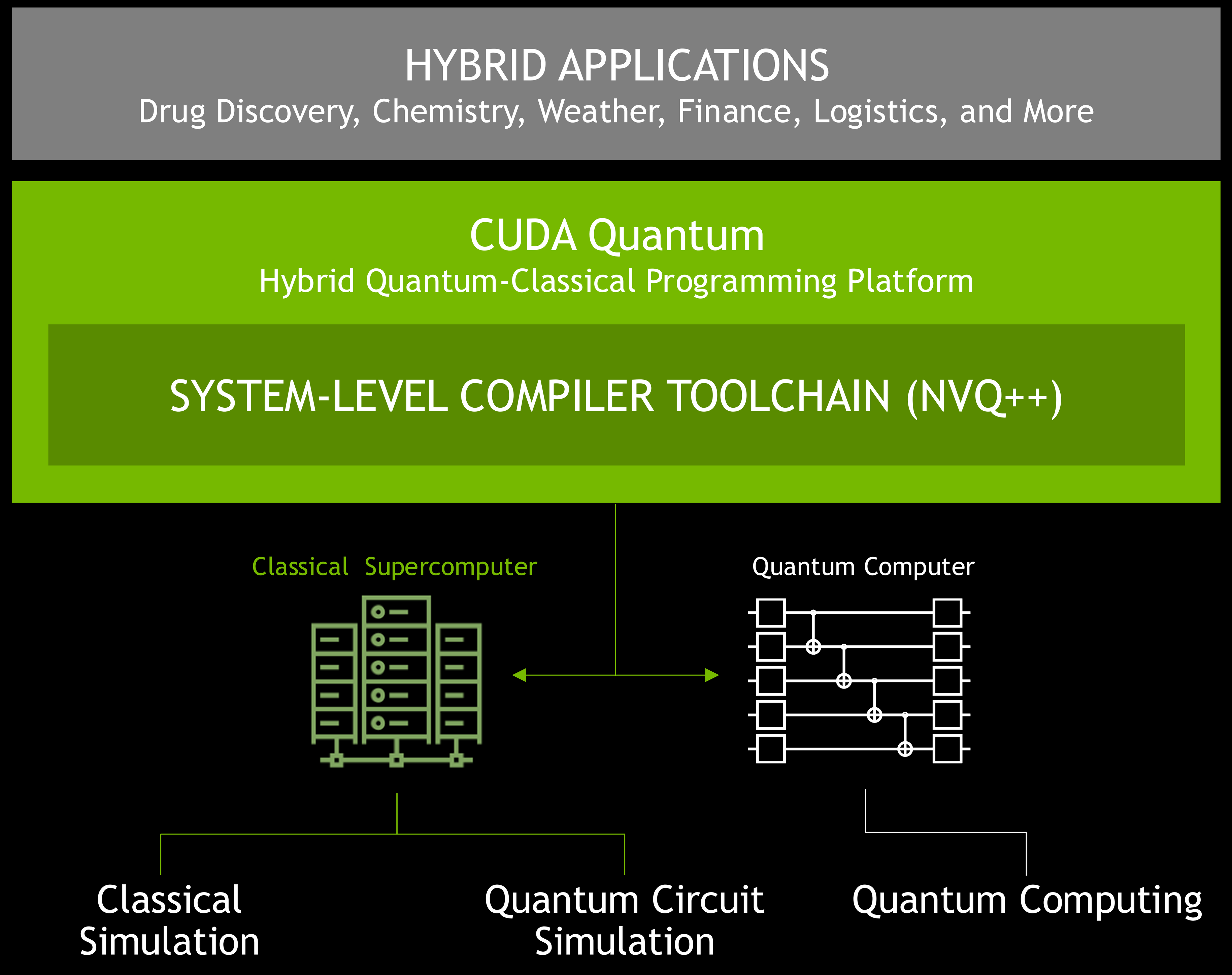
CUDA-Q

Platform for unified quantum-classical accelerated computing

- Programming model extending C++ and Python with quantum kernels
- Open programming model, open-source compiler
  - https://github.com/NVIDIA/cuda-quantum
- QPU Agnostic – Partnering broadly including superconducting, trapped ion, neutral atom, photonic, and NV center QPUs
- Interoperable with the modern scientific computing ecosystem
- Seamless transition from simulation to physical QPU

```
auto ansatz = [](std::vector<double> thetas) __qpu__ {
  cudaq::qreg<3> q;
  x(q[0]);
  ry(thetas[0], q[1]);
  ry(thetas[1], q[2]);
  x<cudaq::ctrl>(q[2], q[0]);
  x<cudaq::ctrl>(q[0], q[1]);
  ry(-thetas[0], q[1]);
  x<cudaq::ctrl>(q[0], q[1]);
  x<cudaq::ctrl>(q[1], q[0]);
};

cudaq::spin_op H = ...;
double energy = cudaq::observe(ansatz, H, {M_PI, M_PI_2});
```



CUDA-Q Now Available

The hybrid quantum computing platform

CUDA-Q 0.8 Now Available

C++ – download from GitHub (https://github.com/NVIDIA/cuda-quantum/releases)

Python – Install from PyPi

CUDA-Q Academic

Educational resources for CUDA-Q (https://github.com/NVIDIA/cuda-q-academic)

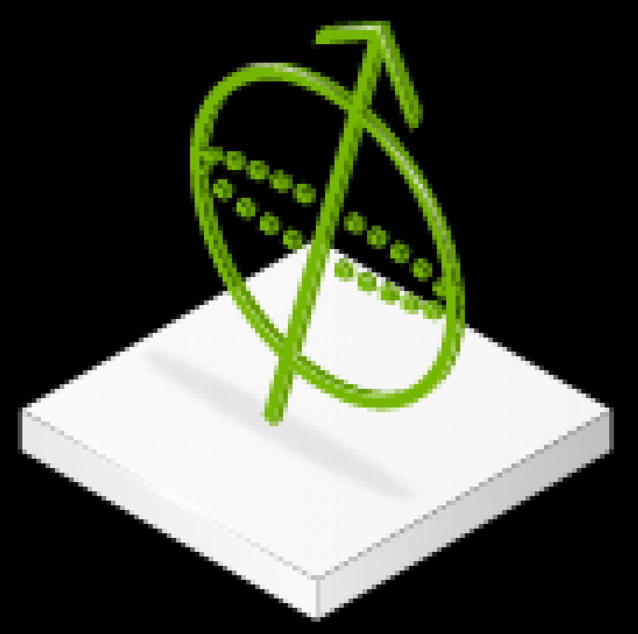
NVIDIA Quantum Cloud

GPUs and QPUs in the cloud.

Early Access: https://www.nvidia.com/en-us/solutions/quantum-computing/cloud/

cuStateVec

Part of NVIDIA cuQuantum SDK



- cuStateVec: a library to accelerate statevector-based quantum circuit simulation
  - Most computations are “in-place” to reduce memory usage
  - Provides low-level primitives to cover common use cases:
    1) **Apply gate matrix**
    2) Apply diagonal/general permutation matrix
    3) Apply exponential of Pauli matrix product
    4) Expectation using matrix as observable
    5) Expectation on Pauli basis
    6) Sampling
    7) Measurement on a Z-product basis
    8) Batched single qubit measurement
    9) State vector segment extraction/update
    10) **Qubit reordering on single/multiple device(s)**

C API

```
custatevecStatus_t custatevecApplyMatrix(
    custatevecHandle_t handle,
    void *sv,
    cudaDataType_t svDataType,
    const uint32_t nIndexBits,
    const void *matrix,
    cudaDataType_t matrixDataType,
    custatevecMatrixLayout_t layout,
    const int32_t adjoint,
    const int32_t *targets,
    const uint32_t nTargets,
    const int32_t *controls,
    const int32_t *controlBitValues,
    const uint32_t nControls,
    custatevecComputeType_t computeType,
    void *extraWorkspace,
    size_t extraWorkspaceSizeInBytes)
```

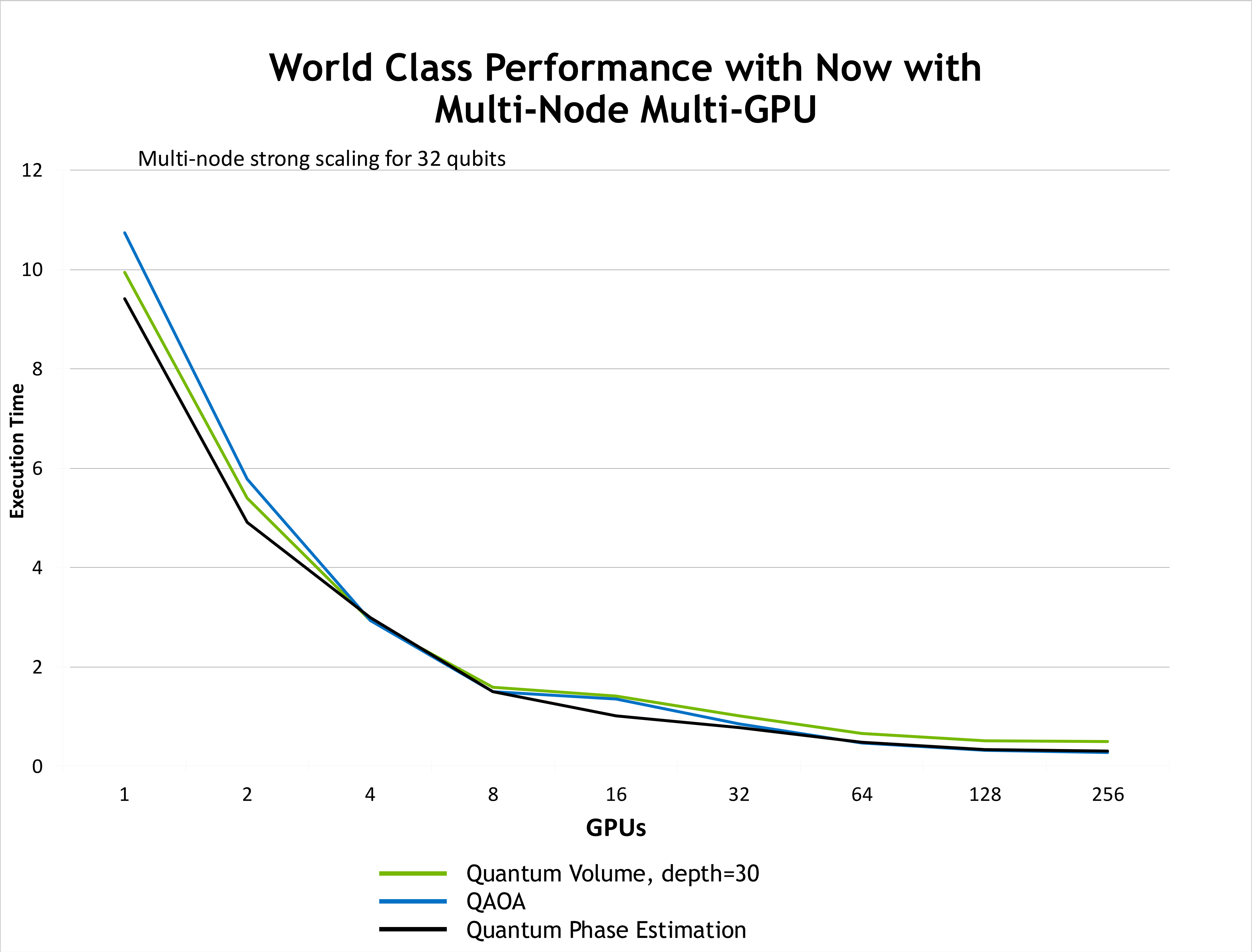
Python API

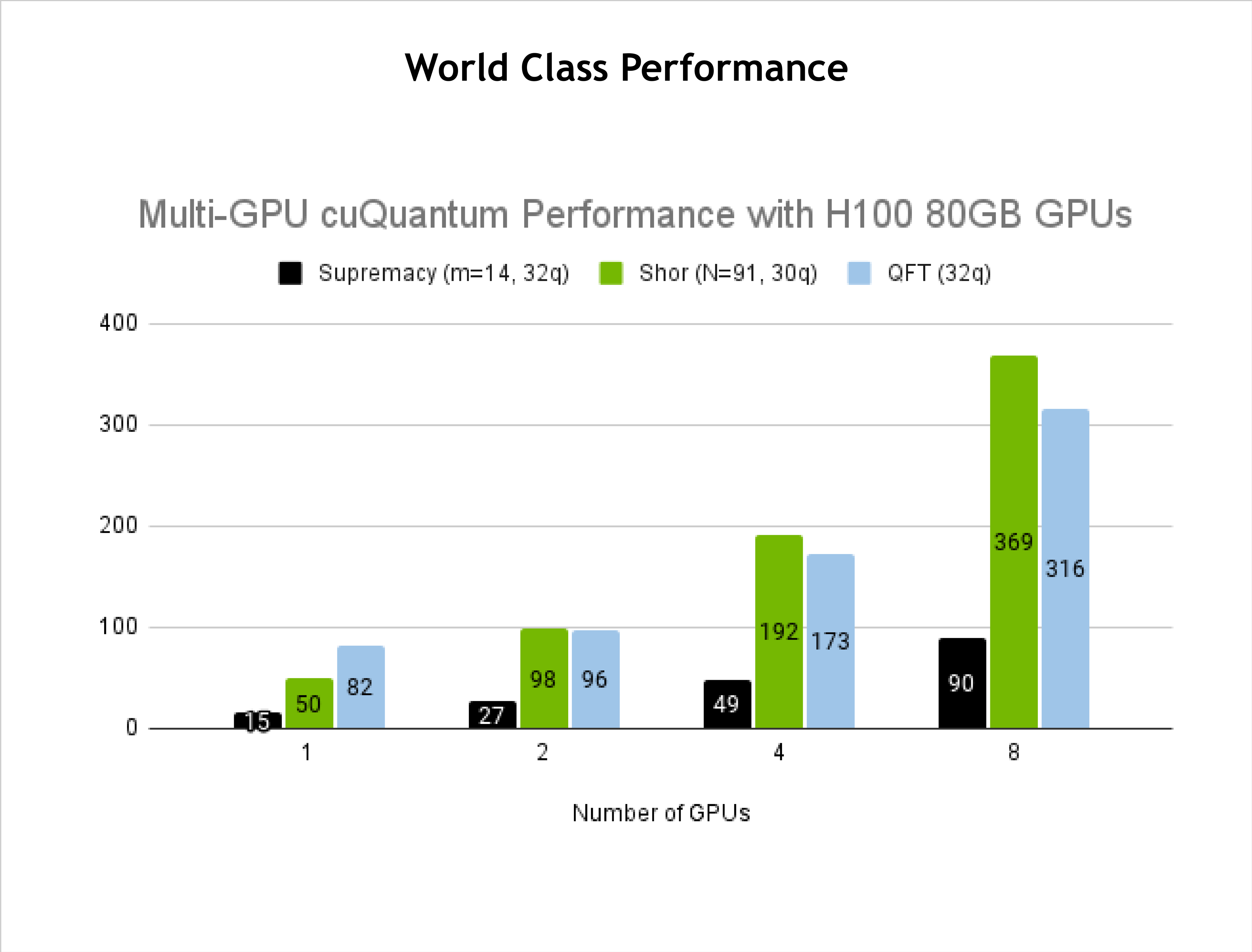
```
cuquantum.custatevec.apply_matrix(
    handle,
    sv,
    sv_data_type,
    n_index_bits,
    matrix,
    matrix_data_type,
    layout,
    adjoint,
    targets,
    n_targets,
    controls,
    control_bit_values,
    n_controls,
    compute_type,
    workspace,
    workspace_size)
```

- Easy integration & adoption for a wide variety of frameworks & programming languages
- Also available in the cuQuantum Appliance container (standalone & Cirq/Qsim backend)

cuStatevec

Research the Quantum Computer of Tomorrow on the most Powerful Computer Today



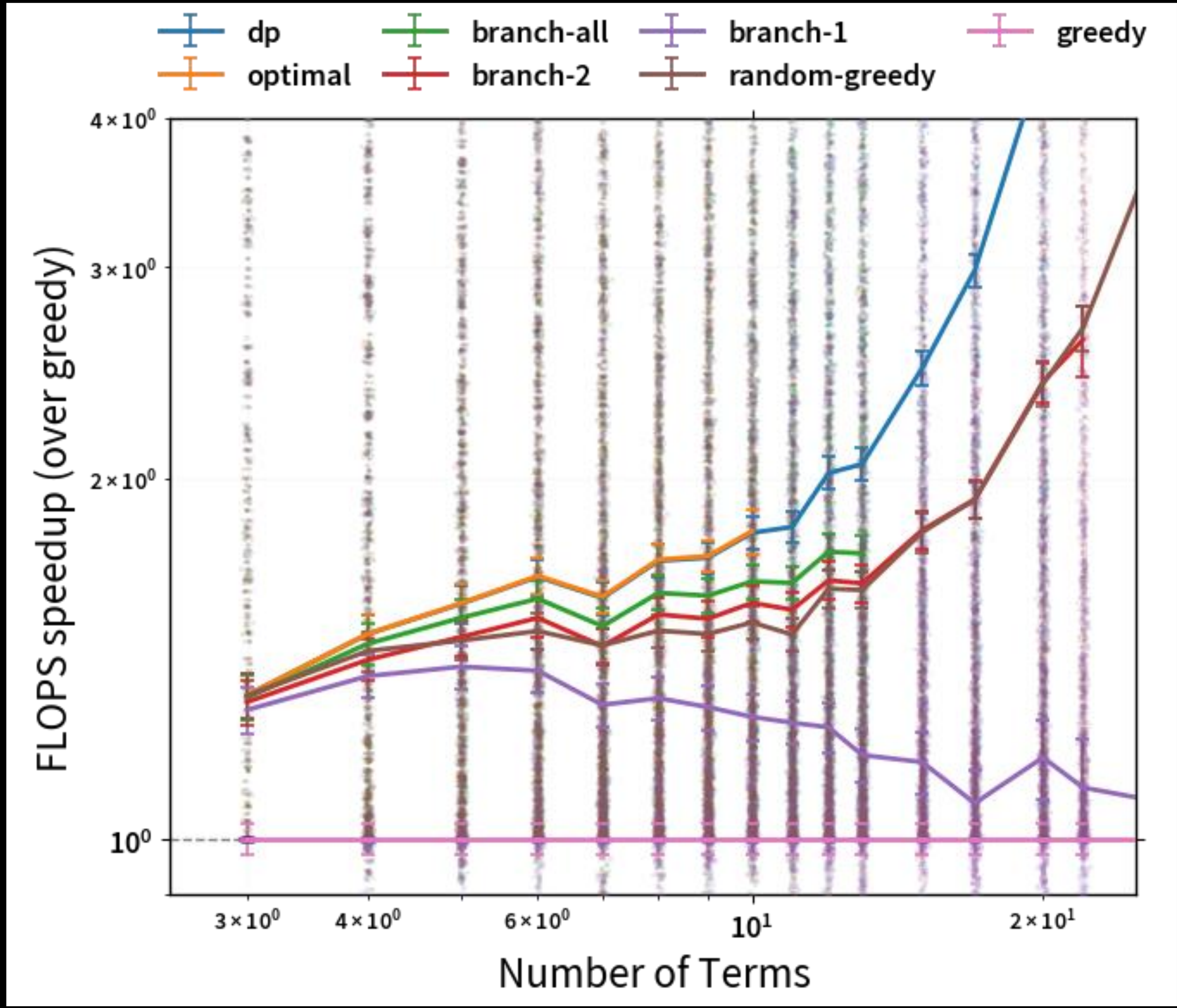


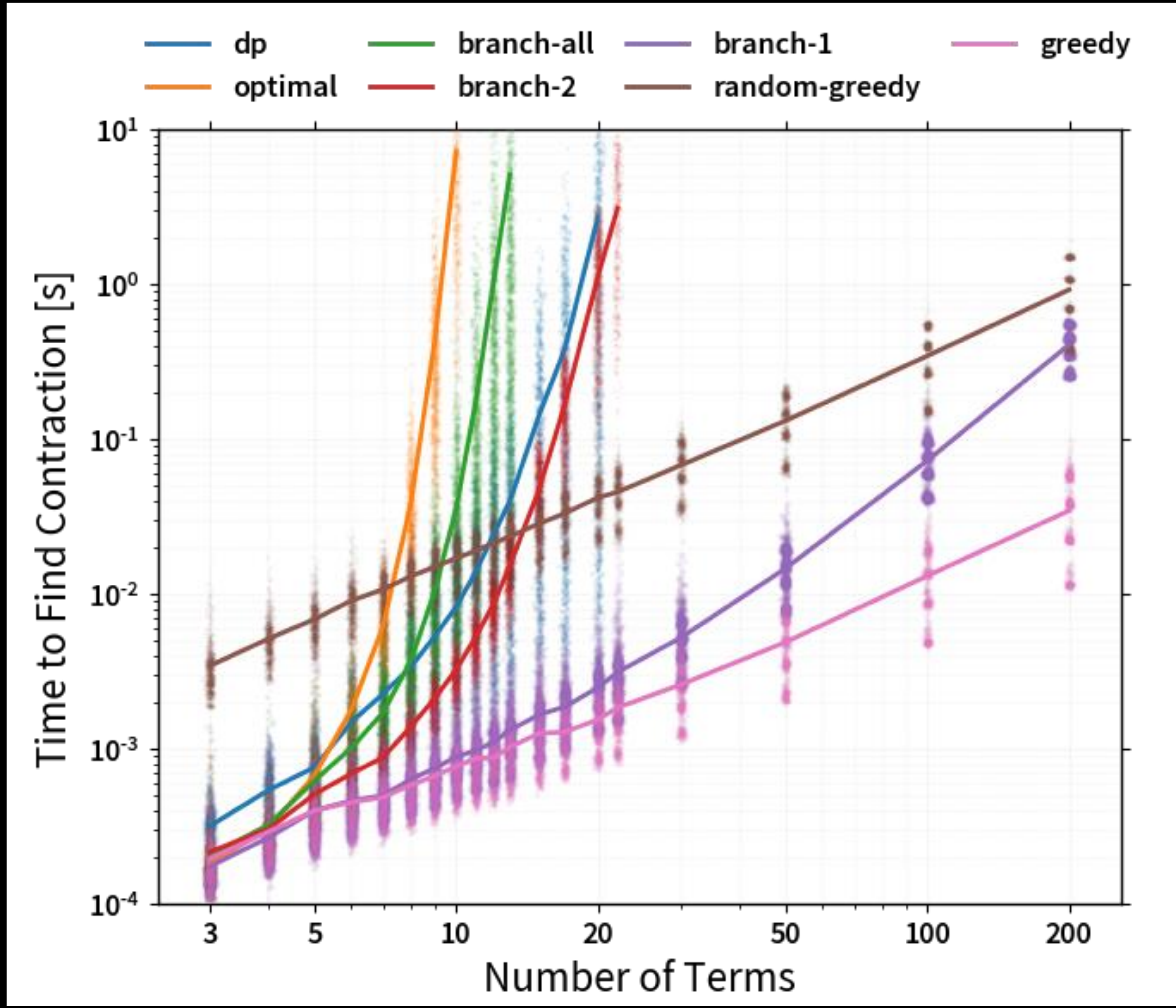
cuTensorNet

A Library to accelerate tensor network based quantum circuit simulation

Motivation: What is contraction path optimization and why is it needed ?

- Reordering the contraction is important to minimize flops as well as memory requirements.
- Finding the optimal path is NP hard problem, which is why they only can be used for small number of tensors (terms)





cuTensorNet

Research the Quantum Computer of Tomorrow on the most Powerful Computer Today

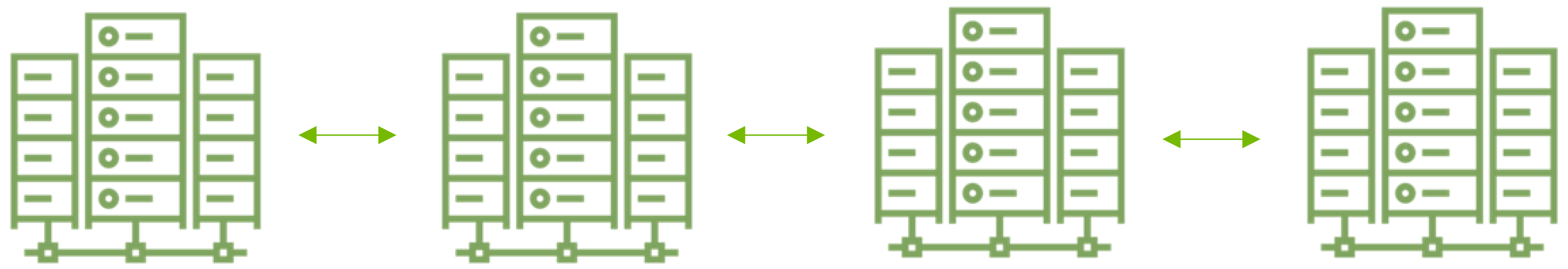
cuQuantum

Quantum Algorithms
Drug Discovery, Chemistry, Weather, Finance, Logistics, and More

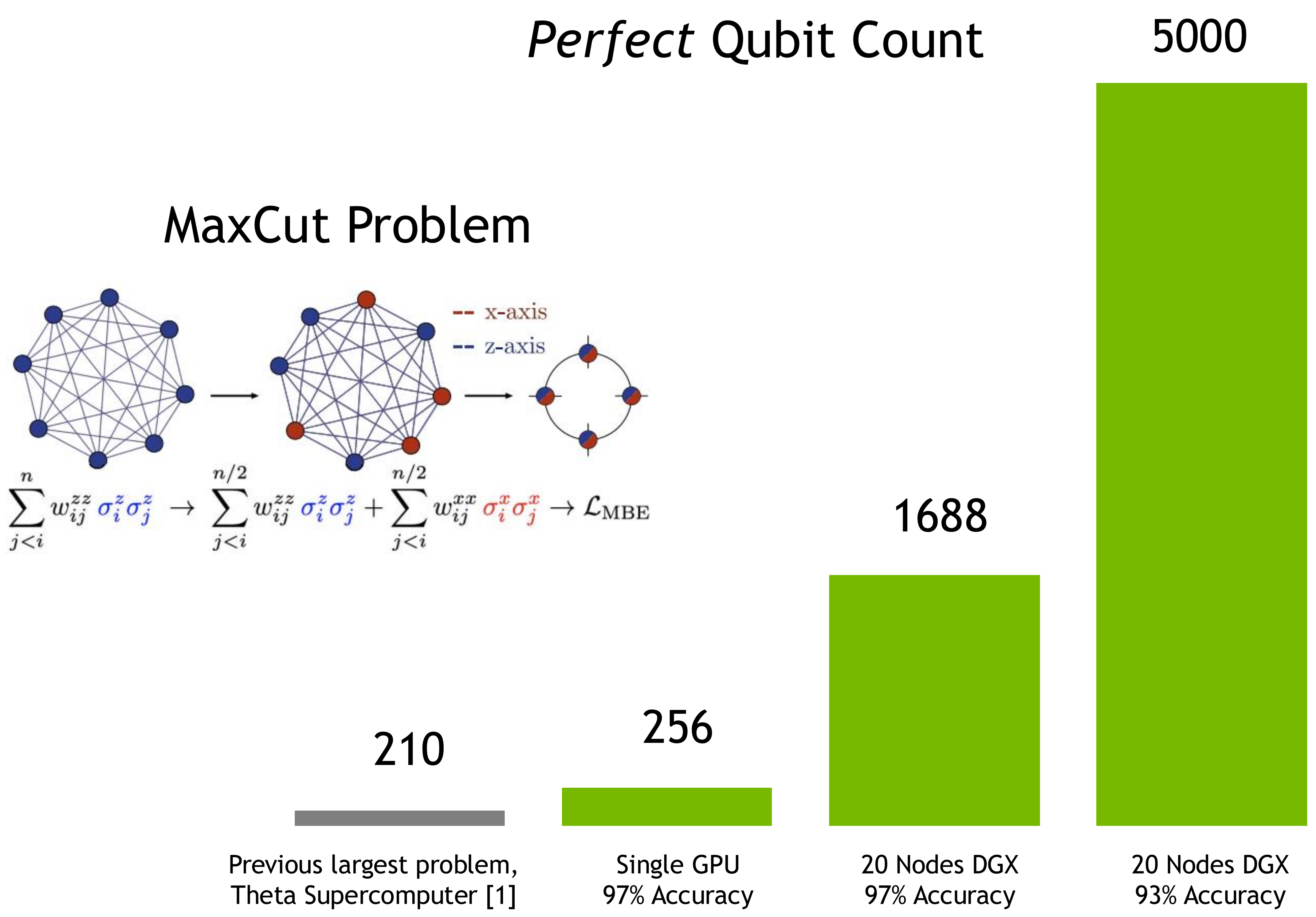
Quantum Simulation Frameworks
Cirq, Qiskit, Pennylane, TensorFlow Quantum, QIBO, …

cuQuantum
cuStateVec, cuTensorNet

GPU Supercomputing



Supercomputing Simulation at FTQC Scale



[1] Danylo Lykov et al, Tensor Network Quantum Simulator With Step-Dependent Parallelization, 2020
https://arxiv.org/pdf/2012.02430.pdf
[2] Taylor Patti et al, Variational Quantum Optimization with Multibasis Encodings, 2022
https://arxiv.org/abs/2106.13304

Quantum Accelerated Supercomputing

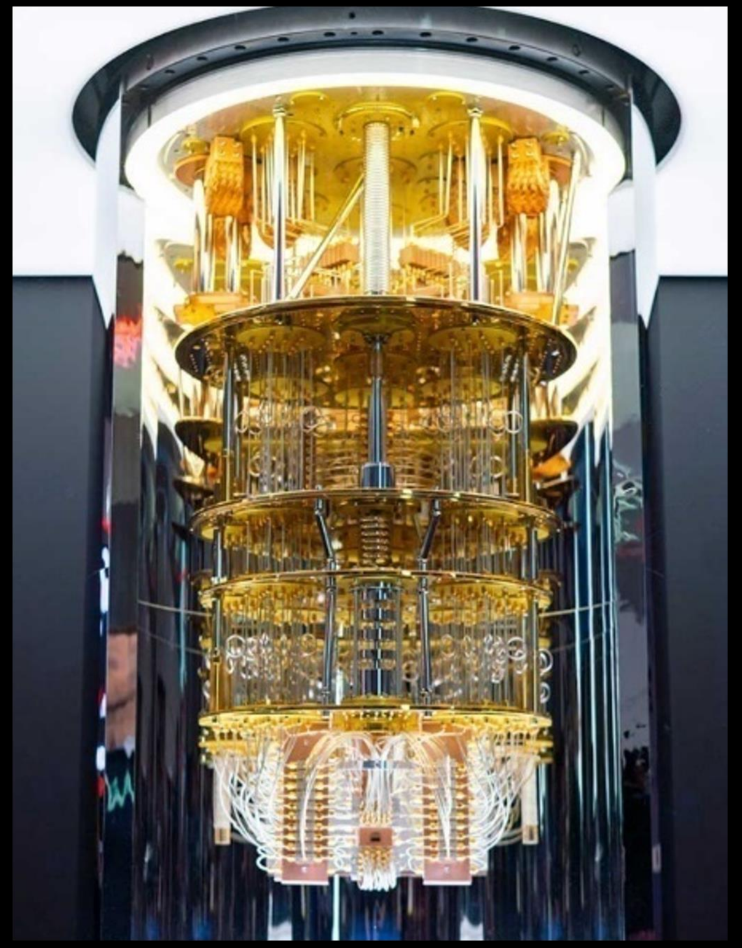
Supercomputers are the foundation of Quantum R&D

Simulation

- Quantum computers are small and error-prone -> simulation
- **Today:** Powerful simulators enable algorithm and application R&D – new approaches (e.g. tensor networks)
- **Future:** Digital twins of quantum computers for design and architecture optimization

HPC Quantum Integration

- Useful quantum computing will be hybrid
- **Today:** Enable domain scientists to start developing for QPUs, enable quantum researchers to use accelerated computing
- **Future:** quantum computers will integrate tightly with supercomputers as accelerators and be co-programmed



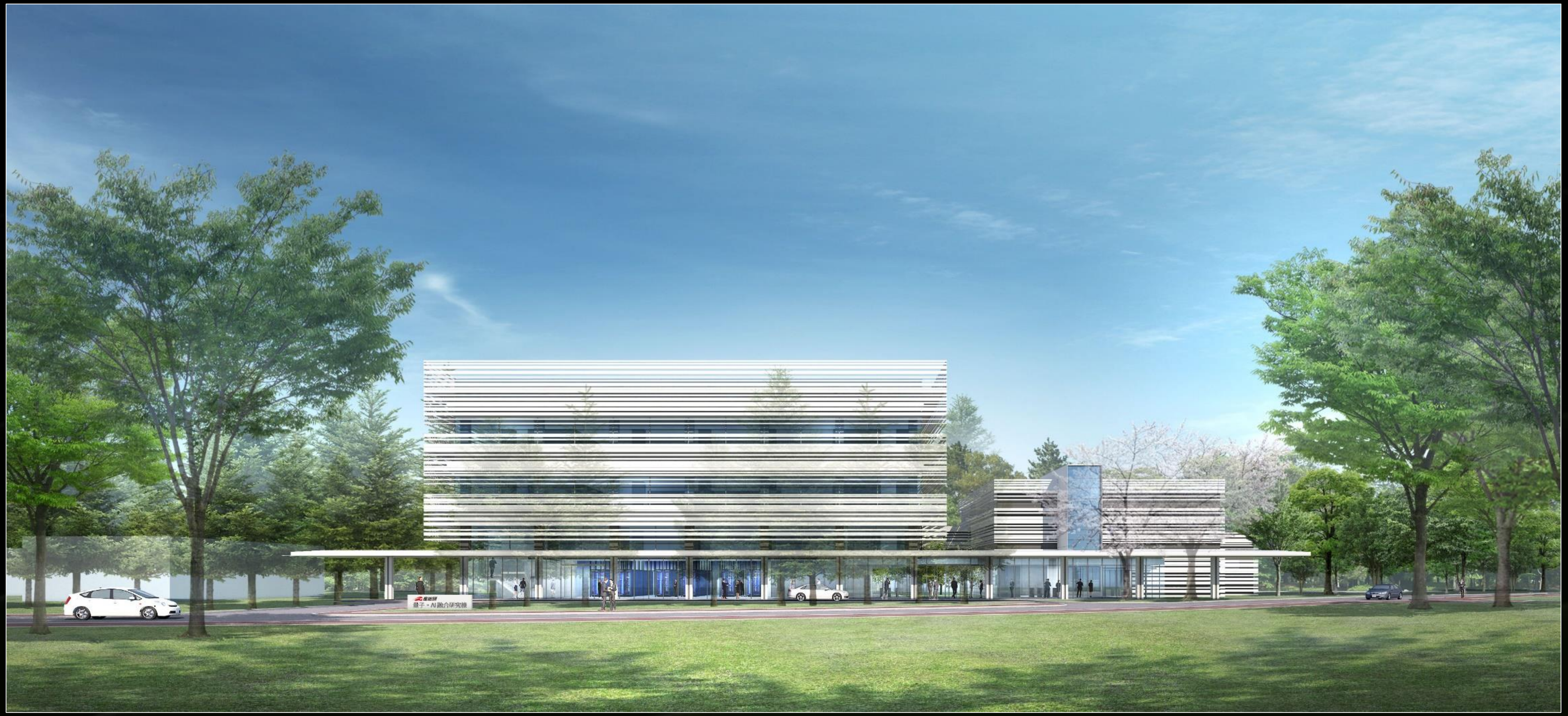
AI for Quantum

- Error correction, calibration, control, compilation are challenging computationally, real-time compute often needed
- Accelerated computing and AI can solve these problems
- **Today:** Enable AI research for all of the above
- **Future:** Hybrid Quantum+AI supercomputer with low-latency link

ABCI-Q

Japanese National Supercomputer for Quantum Research

- 2000+ H100 GPUs in over 500 nodes, connected by Infiniband and powered by CUDA-Q
- Built by Fujitsu, at the G-QuAT/AIST ABCI Supercomputing Center in Tsukuba
- A platform for the advancement of quantum simulation, the integration of quantum-classical systems, and the development of new algorithms inspired by quantum technology



"ABCI-Q will let Japanese researchers explore quantum computing technology to test and accelerate the development of its practical applications. The NVIDIA CUDA-Q platform and NVIDIA H100 will help these scientists pursue the next frontiers of quantum computing research."

- Masahiro Horibe, deputy director of G-QuAT/AIST

Quantum Accelerated Supercomputing

Supercomputers are the foundation of Quantum R&D

Simulation

- Quantum computers are small and error-prone -> simulation
- **Today:** Powerful simulators enable algorithm and application R&D – new approaches (e.g. tensor networks)
- **Future:** Digital twins of quantum computers for design and architecture optimization

HPC Quantum Integration

- Useful quantum computing will be hybrid
- **Today:** Enable domain scientists to start developing for QPUs, enable quantum researchers to use accelerated computing
- **Future:** quantum computers will integrate tightly with supercomputers as accelerators and be co-programmed

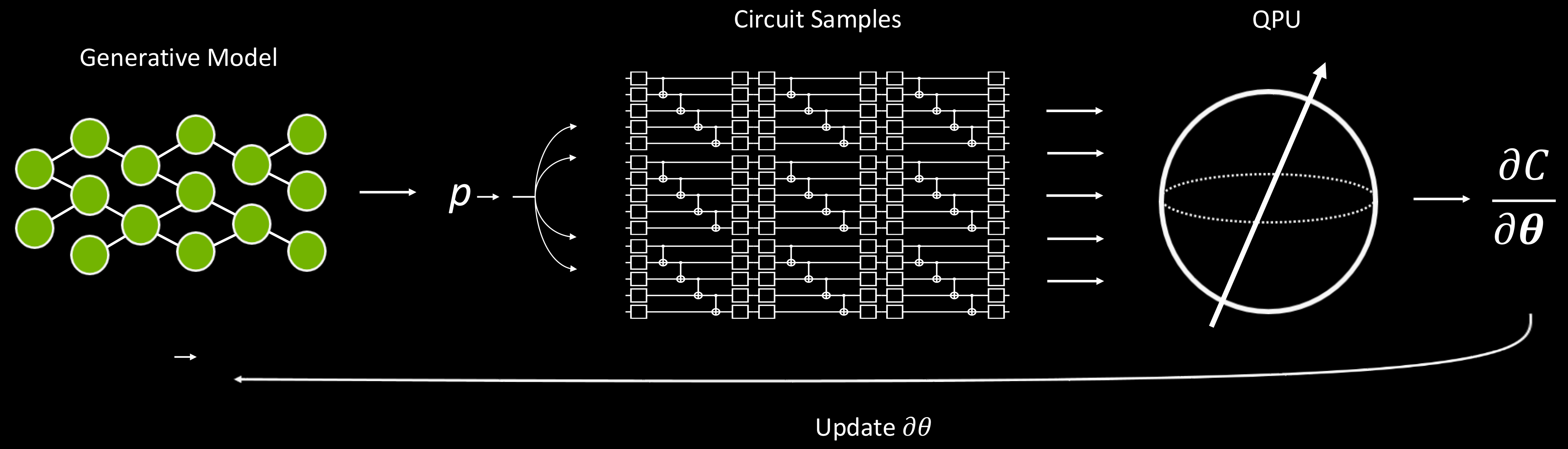
AI for Quantum

- Error correction, calibration, control, compilation are challenging computationally, real-time compute often needed
- Accelerated computing and AI can solve these problems
- **Today:** Enable AI research for all of the above
- **Future:** Hybrid Quantum+AI supercomputer with low-latency link

Generative AI + Quantum Algorithms

University of Toronto, St Jude's, and NVIDIA partner to invent GPT-QE

- Generative Pre-Trained Transformer-based (GPT) method for computing the ground state energies
- First GPT-generated quantum circuit
- Run via CUDA-Q on NERSC Perlmutter



UNIVERSITY OF TORONTO

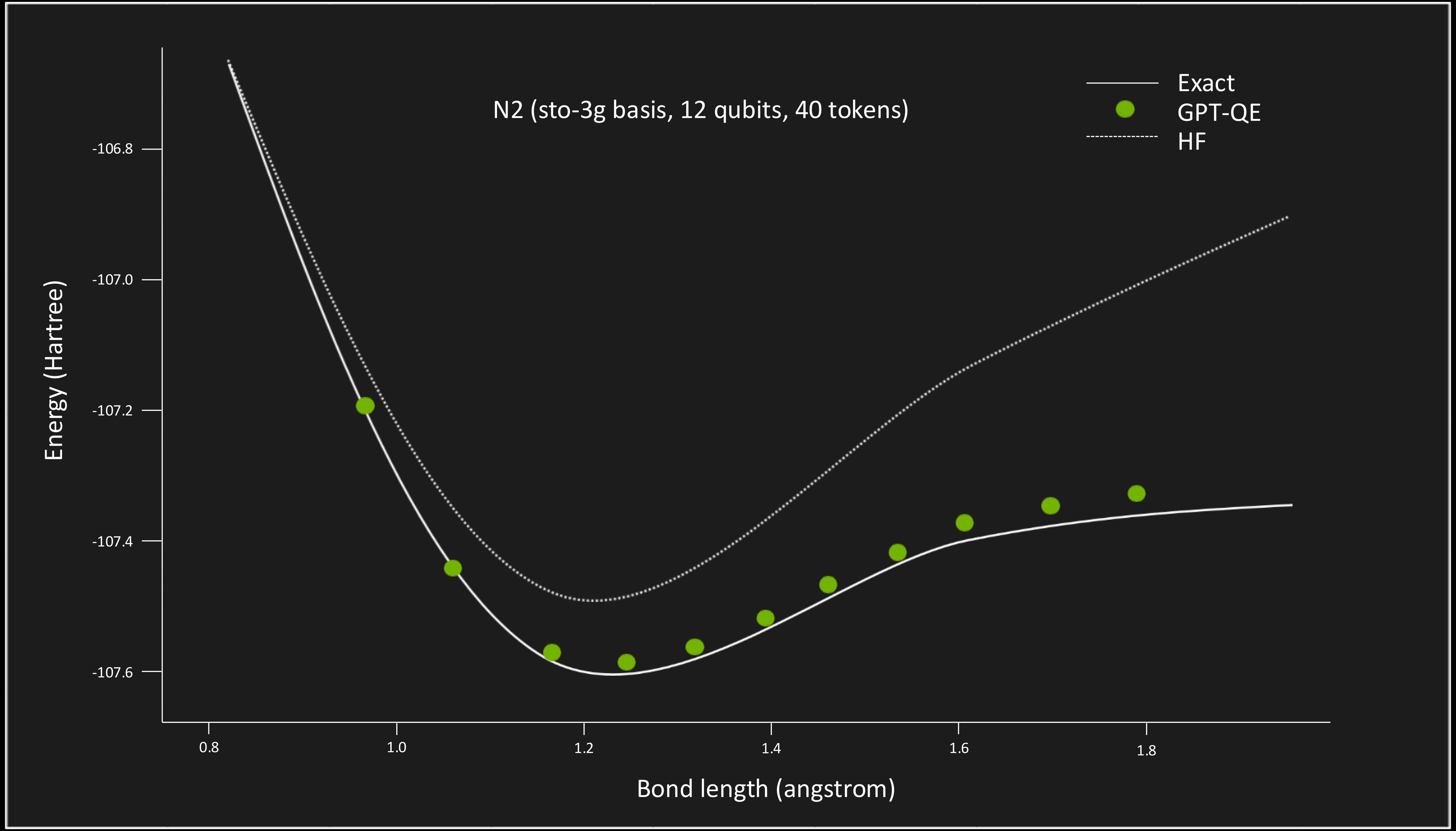
NVIDIA

St. Jude Children's Research Hospital

Generative AI + Quantum Algorithms

University of Toronto, St Jude's, and NVIDIA partner to invent GPT-QE

- Generative Pre-Trained Transformer-based (GPT) method for computing the ground state energies
- First GPT-generated quantum circuit
- Run via CUDA-Q on NERSC Perlmutter



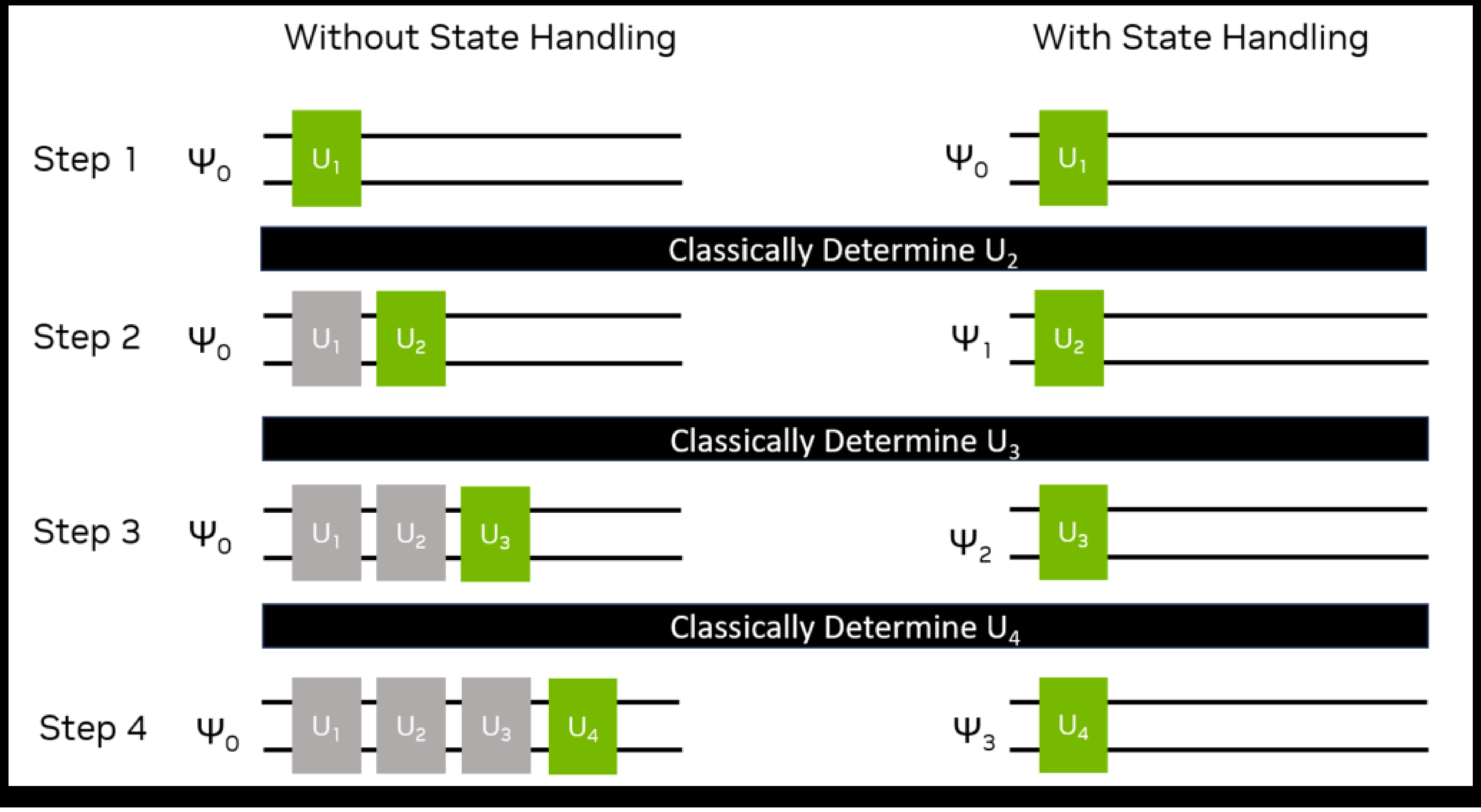
UNIVERSITY OF TORONTO

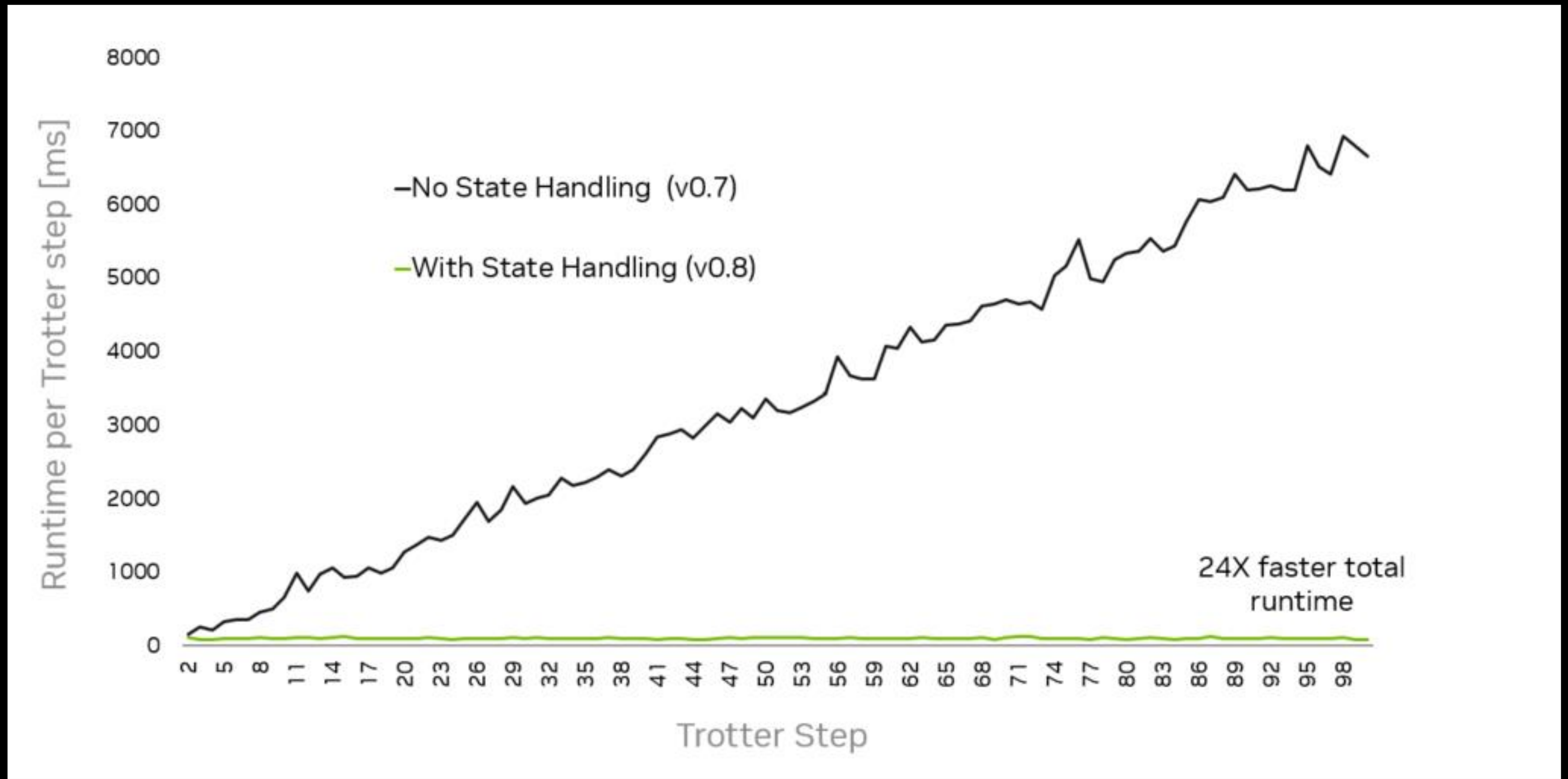
NVIDIA

St. Jude Children's Research Hospital

CUDA-Q 0.8 Update

State handling, Pauli words, Custom unitary operations, Visualization tools, NVIDIA Grace Hopper integration





FEATURES

- State handling
- Pauli words
- Custom unitary operations
- Visualization tools (latex output of circuit drawer)
- NVIDIA Grace Hopper integration

https://developer.nvidia.com/blog/performant-quantum-programming-even-easier-with-nvidia-cuda-q-v0-8/

DGX Quantum

System for Integration of Quantum with GPU supercomputing

- Tightly integrates Quantum with GPU Supercomputing
- Qubit Agnostic – Supports different qubit modalities
- Reduces GPU-QPU latency by 1-2 orders of magnitude
- Enables GPU Acceleration of Quantum Error Correction, Calibration, and Hybrid Algorithms
- Scalable for more GPU compute and larger QPUs

NVIDIA®

QUANTUM MACHINES

DGX Quantum

System for Integration of Quantum with GPU supercomputing

Typical Latencies

Classical-Quantum Latencies

Remote QPU, Web API — 1-10 seconds

Local QPU, Ethernet — 10 microseconds

Typical Error Correction Budget* — 10 microseconds

DGX Quantum PCIe — 400 nanoseconds

*Includes decoding time

CUDA-Q Hands-on workshop

「CUDA-Qハンズオン講習会」（現地参加のみ）を開催します

日時、開催場所、定員

- 2024年 10月 18日（金） 13:00 - 17:00
  - 現地（名古屋大学東山キャンパス　情報基盤センター2F　演習室）
  - 現地 30名

  - 終了時間は前後する場合があります

講師

- エヌビディア合同会社　講師（濱村、丹、古家）

開催趣旨

「CUDA-Q」はNVIDIAが開発している量子古典ハイブリッド計算のためのオープンソースプラットフォーム です。

量子コンピュータ向けのアルゴリズム研究やアプリケーション開発には、量子回路シミュレーションが大きな役割を果たしています。

CUDA-Qは複数GPU、複数ノードを用いた量子回路シミュレーションをサポートしているので、「不老」が搭載するNVIDIA V100の豊富なGPU・メモリ資源を活用することができます。

CUDA-Qは実機を含めたバックエンドを選択し量子計算を実行でき、古典（従来型）高性能計算と組み合わせたハイブリッド計算を実行できます。

本講習会では、量子計算プラットフォームに興味のある大学・研究所・企業の研究者及び学生を対象に、ハンズオンを通して「不老」におけるCUDA-Qの利用方法を理解することを目的とします。