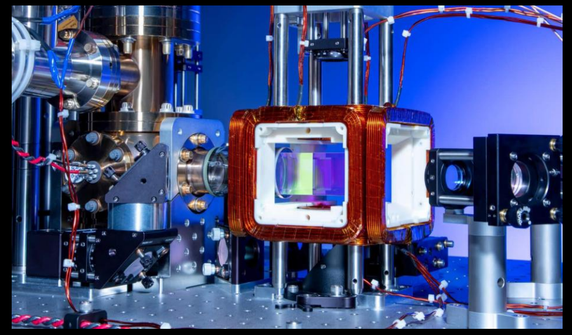
Defining the Accelerated Quantum Supercomputer

Yun Yuan Wang (Pika Wang), Sr. Solutions Architect | NVAITC, NVIDIA Taiwan

Year of Quantum Breakthroughs

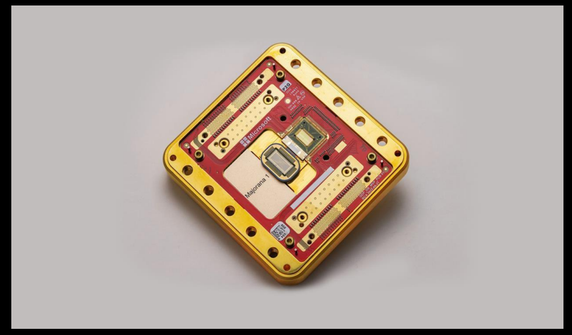
Quantum Error Correction

Google Quantum AI

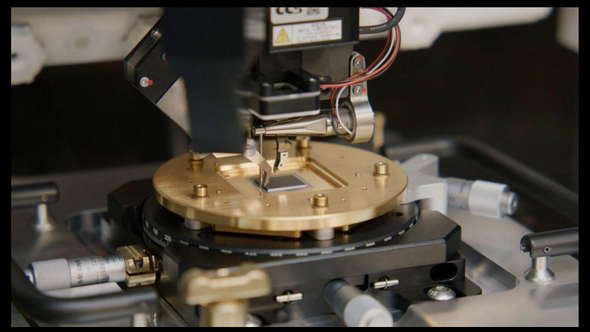


IQuEra

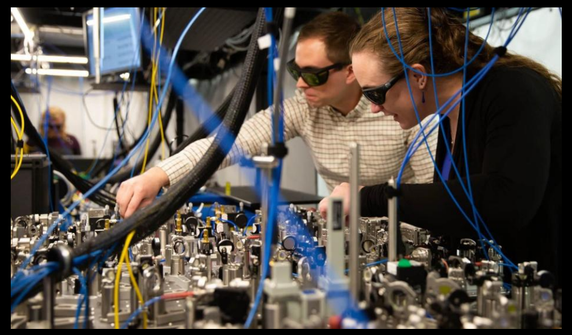
New Architectures



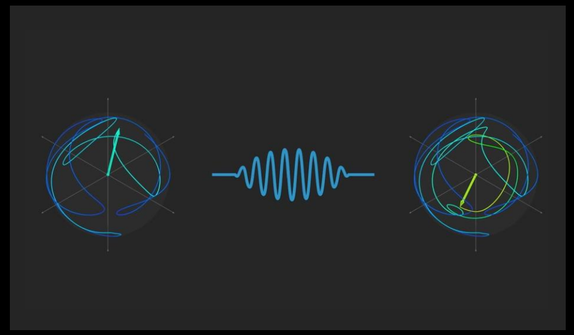
Microsoft



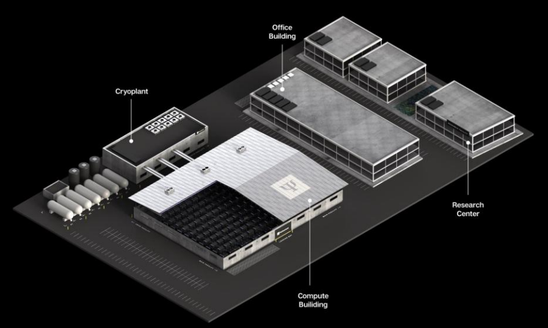
aws



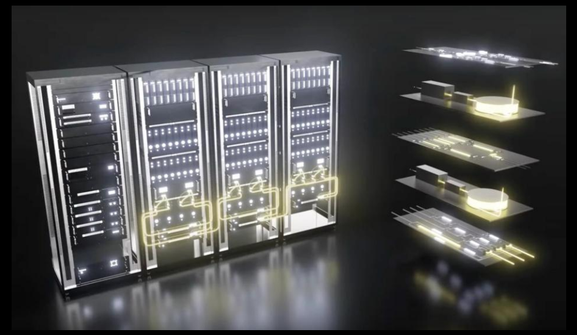
QUANTINUUM



Infleqtion



PsiQuantum



XANADU

NVIDIA

NVIDIA Quantum Product Map

CUDA-Q is the entry point into our products for most users

NVIDIA Quantum Cloud

CUDA-QX Libraries

CUDA-Q

Libraries | Programming Model | Tools | Infrastructure

Quantum

cuQuantum

Libraries for simulating applications and hardware

QPU

CUDA-Q supports range of QPU vendor backends

OQC · Quantinuum · ORCA Computing · Alice & Bob · IonQ · Quantum Brilliance · Pasqal · IQM · QuEra Computing Inc. · seeqc · rigetti · Infleqtion · Anyon Technologies

DGX-Q

Reference architecture for low-latency QPU/DGX integration

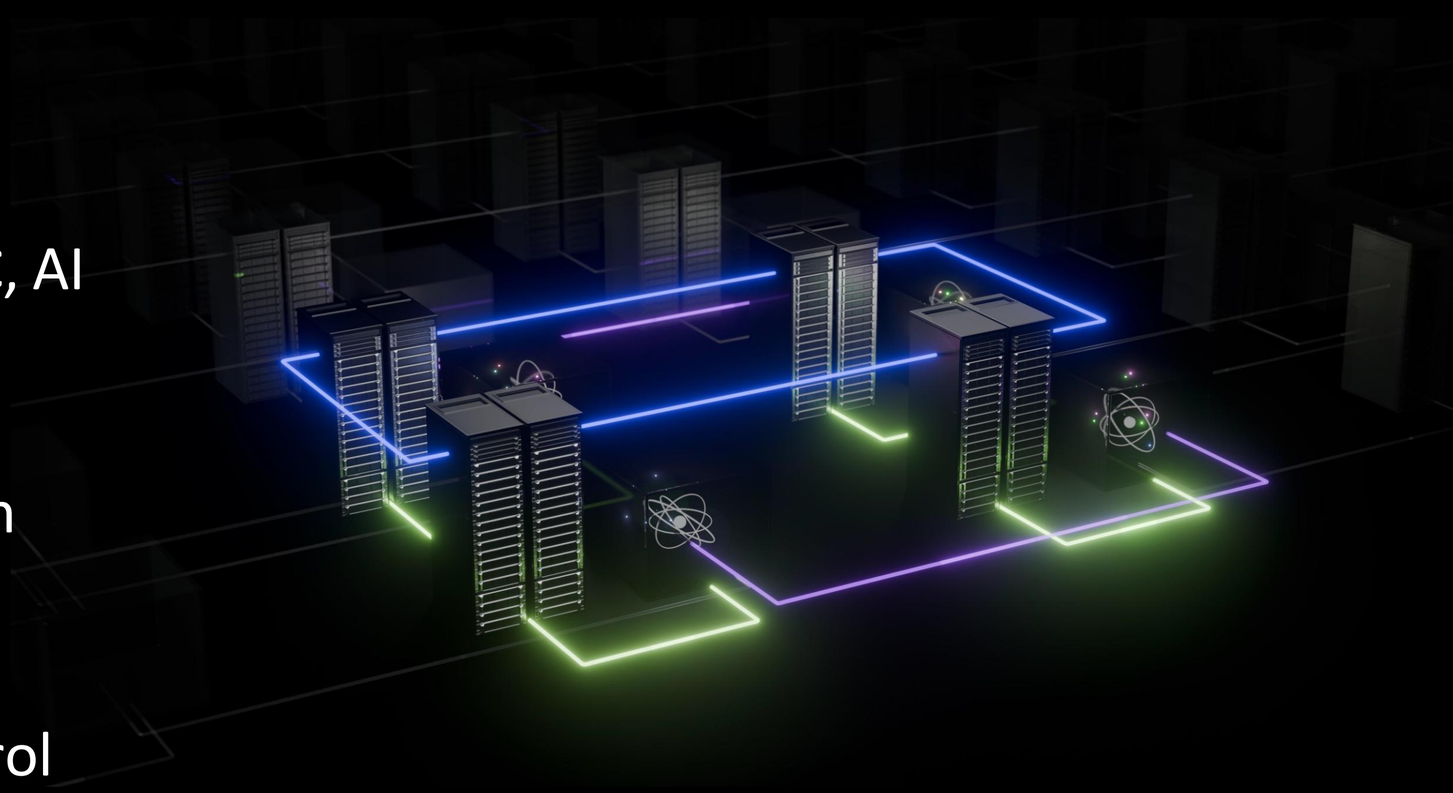
Classical

DGX-SuperPOD

NVIDIA

The Accelerated Quantum Supercomputer

- AI supercomputing architecture **connecting quantum hardware**
- **Hybrid applications** – combining HPC, AI and QC resources
- A **software platform** enabling domain scientists
- **Qubit-agnostic** development of control and error correction



The NVIDIA Quantum Platform

Bringing AI Supercomputing to Enable Useful Quantum Computing

CUDA-Q

Open-Source Software Platform for Accelerated Quantum Supercomputers

The NVIDIA Quantum Platform

Bringing AI Supercomputing to Enable Useful Quantum Computing

CUDA-Q

Open-Source Software Platform for Accelerated Quantum Supercomputers

Pasqal — New at GTC25

QUANTINUUM — Free Access to H1-1 Through CUDA-Q

Anyon Technologies
aws
Infleqtion
IONQ
IQM
ORCA Computing
OQC
Pasqal
QUANTINUUM
QUANTUM BRILLIANCE
QuEra Computing Inc.
rigetti

The NVIDIA Quantum Platform

Bringing AI Supercomputing to Enable Useful Quantum Computing

CUDA-Q
Open-Source Software Platform for Accelerated Quantum Supercomputers

cuQuantum
Accelerated Quantum Simulations

Cirq
PENNYLANE
QIBO
Qiskit
Quest
Qulacs
TKET™
Torch Quantum

The NVIDIA Quantum Platform

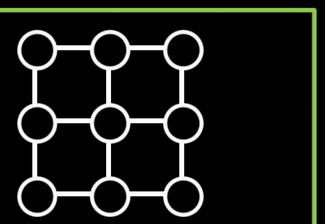
Bringing AI Supercomputing to Enable Useful Quantum Computing

CUDA-QX
Turnkey Quantum Research and Application Development

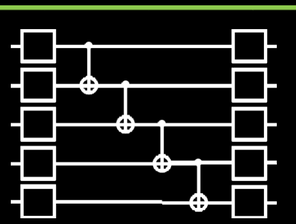
CUDA-Q
Open-Source Software Platform for Accelerated Quantum Supercomputers

cuQuantum
Accelerated Quantum Simulations

DGX Quantum
Reference Architecture for Low Latency Integration



CUDA-Q QEC



CUDA-Q Solvers

QUANTUM MACHINES

NVIDIA

CUDA-Q Python Bindings

Core C++ compiler platform exposed via builder API
JIT compile Quake representation at runtime

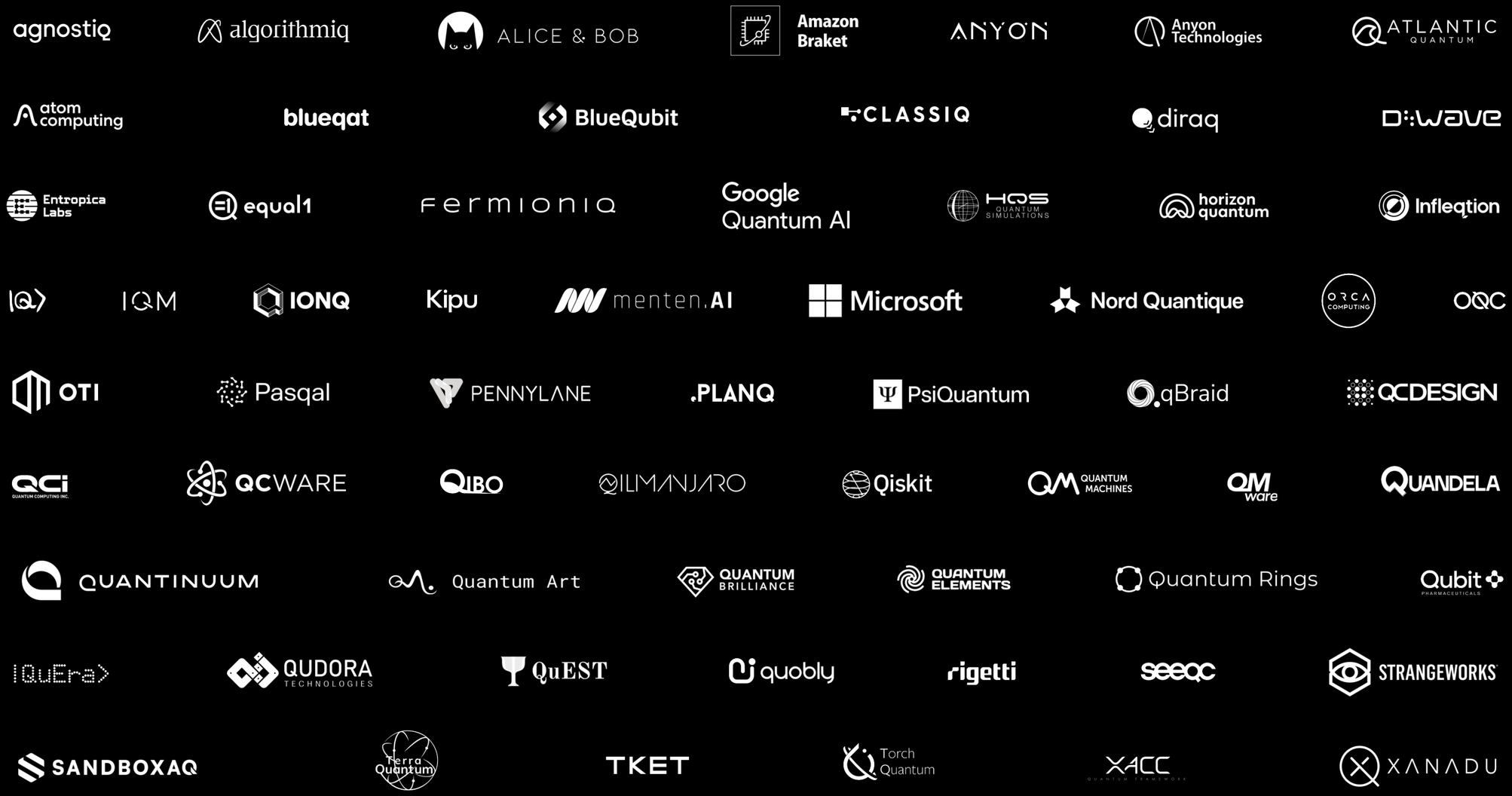
- Simple ways to create quantum kernels

```python
import cudaq

@cudaq.kernel
def kernel():
  # A single qubit initialized to the
  ground / zero state
  qubit = cudaq.qubit()

  # Apply the Pauli x gate to the qubit
  x(qubit)

  # Measurement operator
  mz(qubit)


# Sample the qubit for 1000 shots to gather
statistics
result = cudaq.sample(kernel,
shots_count=1000)
```

```python
import cudaq

@cudaq.kernel
def kernel(angles: List[float]):
  # Allocate a qubit that is initialized to
  the |0> state.
  qubit = cudaq.qubit()

  # Apply parameterized gates to the qubit
  rx(angles[0], qubit)
  ry(angles[1], qubit)

# Gate parameters which initialize the qubit
in the zero state
initial_parameters = [0, 0]

# Draw the circuit diagram for the kernel
print(cudaq.draw(kernel, initial_parameters))
```

The NVIDIA Quantum Ecosystem

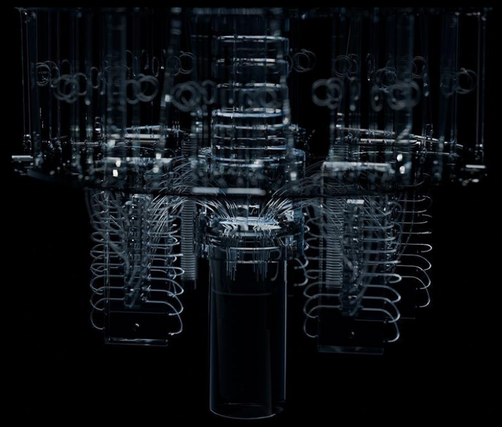
Accelerating the Quantum World

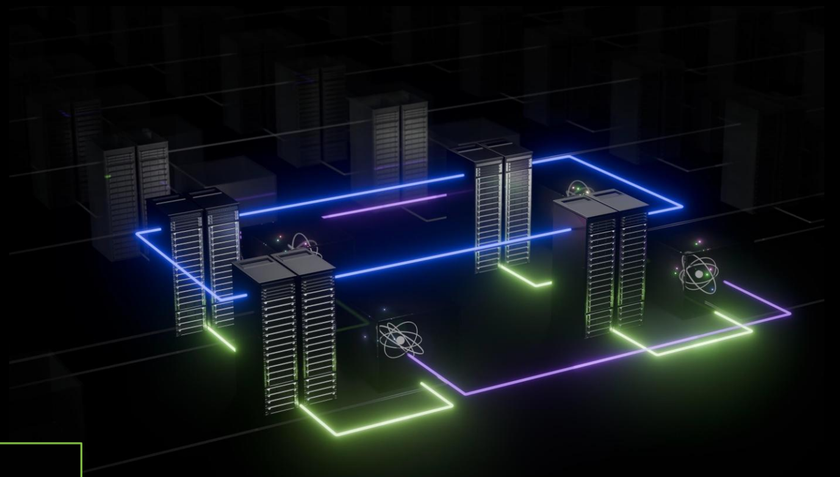


NVIDIA

Challenges Ahead

How to Turn Qubits into Accelerated Quantum Supercomputers





Qubits

Better QPUs

GPU

Better Algorithms

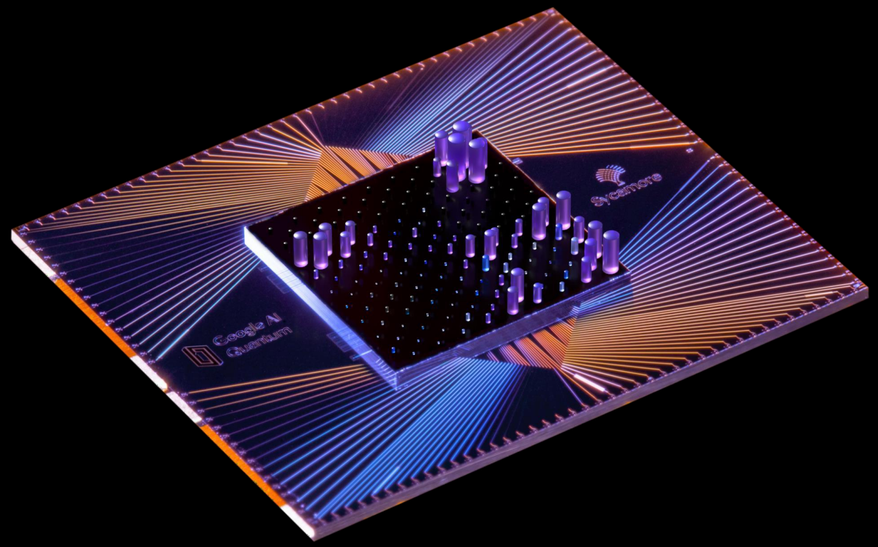
Scaling QEC

Accelerated Quantum Supercomputers

Developing Better QPUs with Simulation

NVIDIA

Google Quantum AI



Qubits

Better QPUs

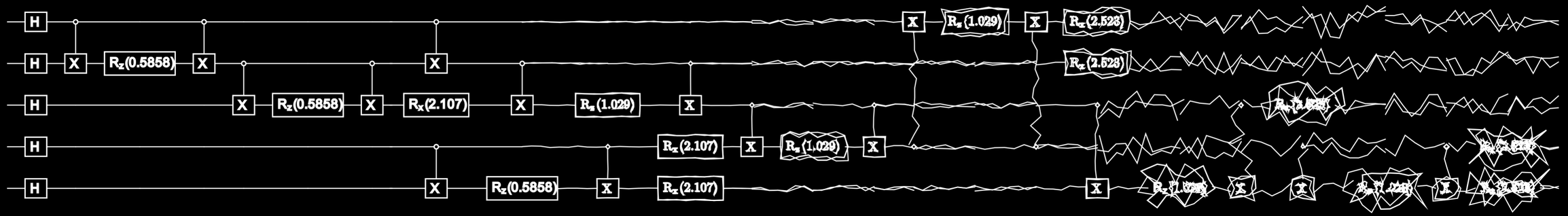
GPU

Better Algorithms

Scaling QEC

Accelerated Quantum Supercomputers

Noise Limits Today's Quantum Hardware

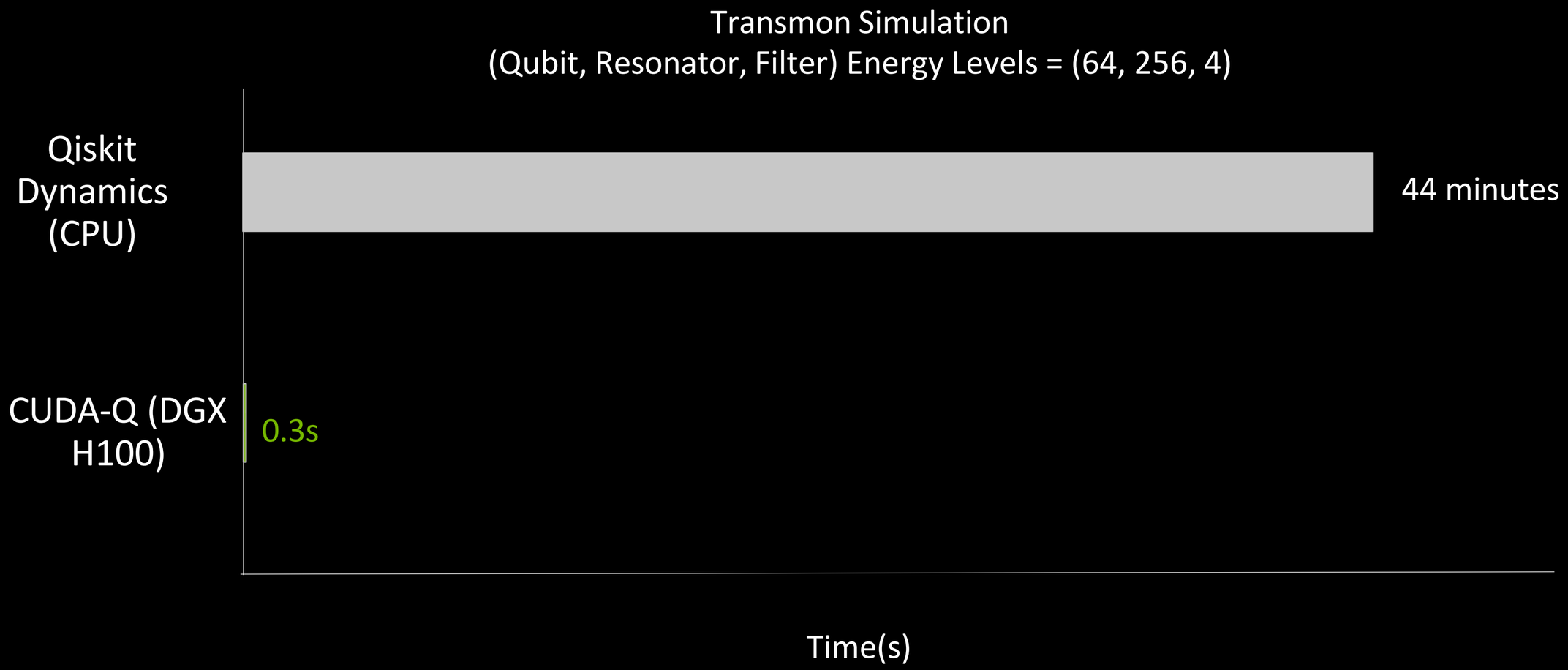


Current quantum hardware is limited to just hundreds of instructions by noise

Quantum
Processor

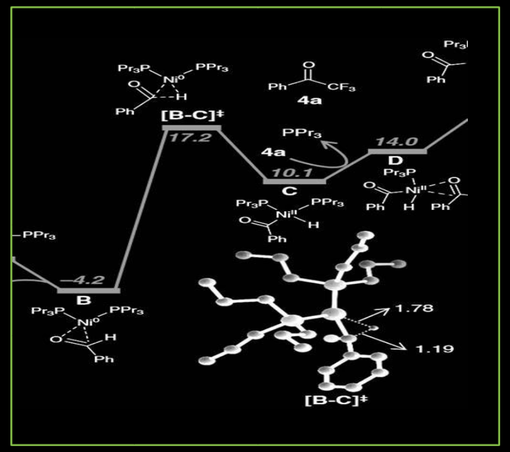
Dynamics in CUDA-Q

Enabling Better QPU Designs

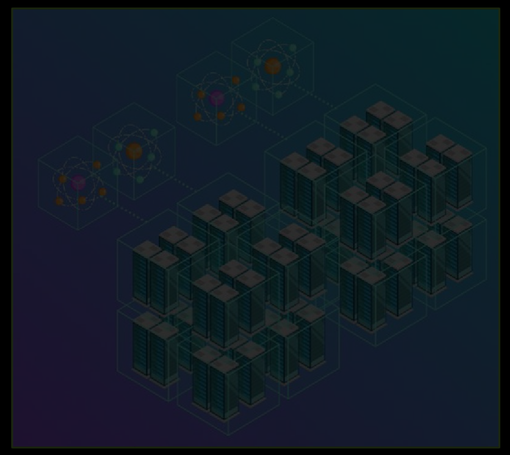


CPU: Intel Xeon 8480CL | GPU: H100
Energy levels: 64 Transmon x 128 Resonator x 4 Purcell Filter

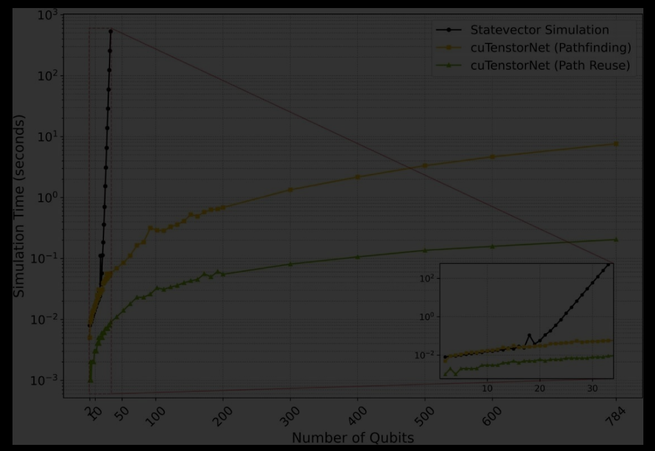
Better Algorithms with Large-Scale Simulations and AI



IonQ/AWS/AstraZeneca
AF-QMC simulation for electronic structure modeling



HPE Labs
Distributed QC with Adaptive Circuit Knitting



NCHC
784 qubit simulation to validate QML approaches

Qubits

Better QPUs

GPU

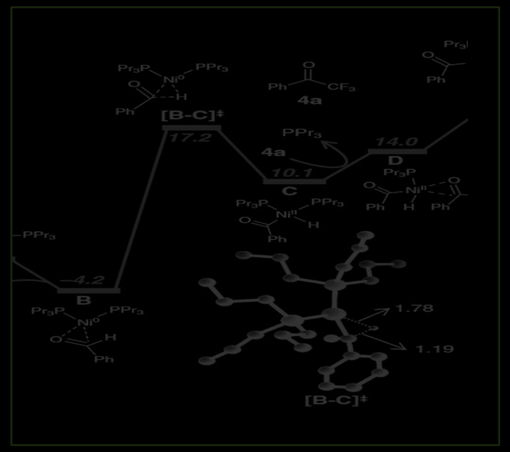
Better Algorithms

Scaling QEC

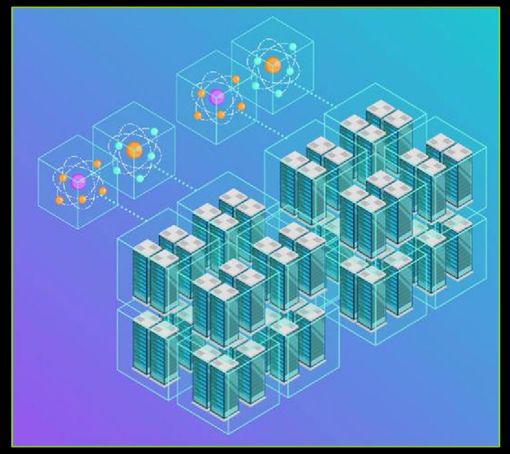
Accelerated Quantum Supercomputers

NVIDIA

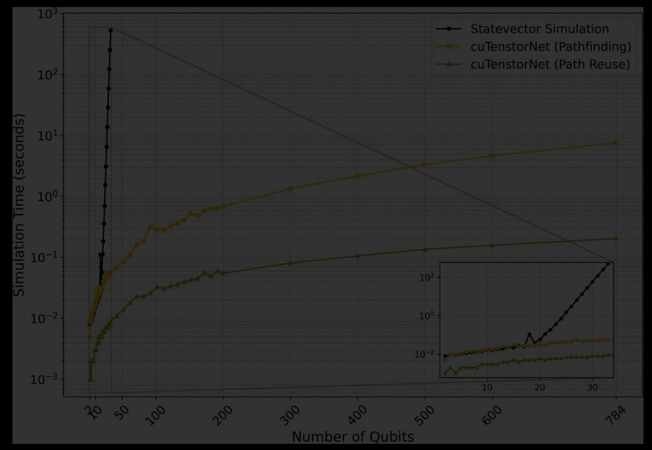
Better Algorithms with Large-Scale Simulations and AI



IonQ/AWS/AstraZeneca
AF-QMC simulation for electronic structure modeling



HPE Labs
Distributed QC with Adaptive Circuit Knitting



NCHC
784 qubit simulation to validate QML approaches

Qubits

Better QPUs

GPU

Better Algorithms

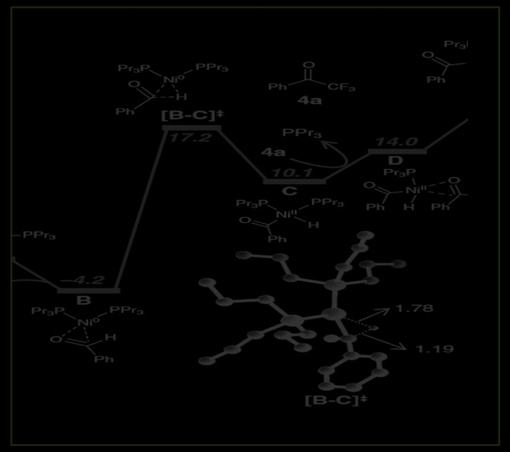
Scaling QEC

Accelerated Quantum Supercomputers

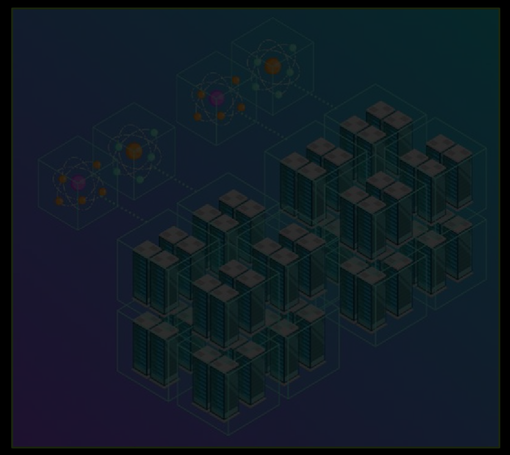
Mohseni *et. al*. How to Build a Quantum Supercomputer: Scaling Challenges and Opportunities
https://arxiv.org/html/2411.10406v1 (2024) GTC Talk DD73669

NVIDIA

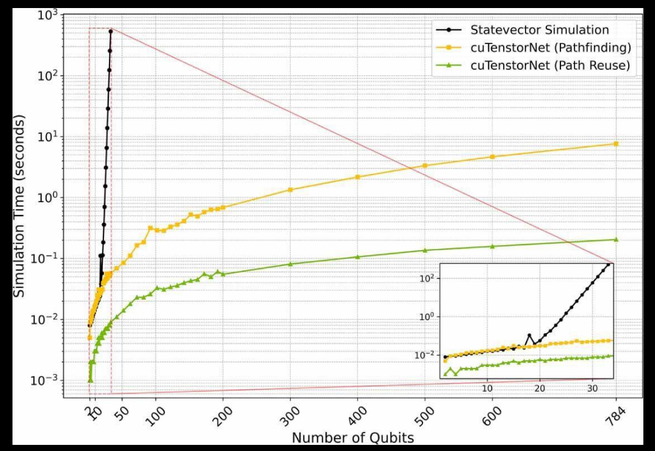
Better Algorithms with Large-Scale Simulations and AI



IonQ/AWS/AstraZeneca
AF-QMC simulation for electronic structure modeling



HPE Labs
Distributed QC with Adaptive Circuit Knitting



NCHC
784 qubit simulation to validate QML approaches

Qubits

Better QPUs

GPU

Better Algorithms

Scaling QEC

Accelerated Quantum Supercomputers

Chen *et. al*. cuTN-QSVM: cuTensorNet-accelerated Quantum Support Vector Machine with cuQuantum SDK. https://arxiv.org/html/2405.02630v2 (2024)

Solar Energy Prediction with NVIDIA CUDA-Q

Demonstration of hybrid neural networks for sustainable energy

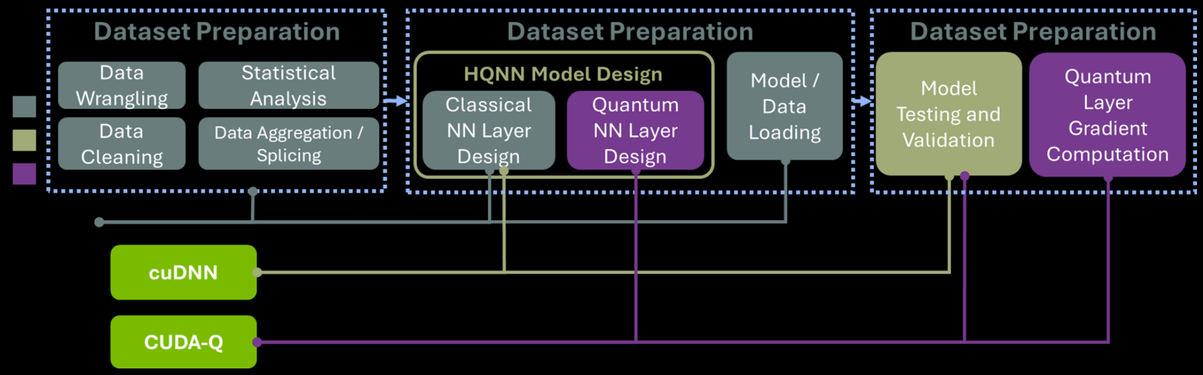
CUDA-Q + 

Challenge

- Variations in renewable energy generation calls for highly reliable forecasting
- Conventional machine learning approaches to solar energy prediction are computationally challenging

Solution

- Hybrid Quantum+Classical neural networks offer promise for improved solar energy forecasting
- Accelerated workflow with CUDA-Q allows exploration and assessment of hybrid forecasting techniques

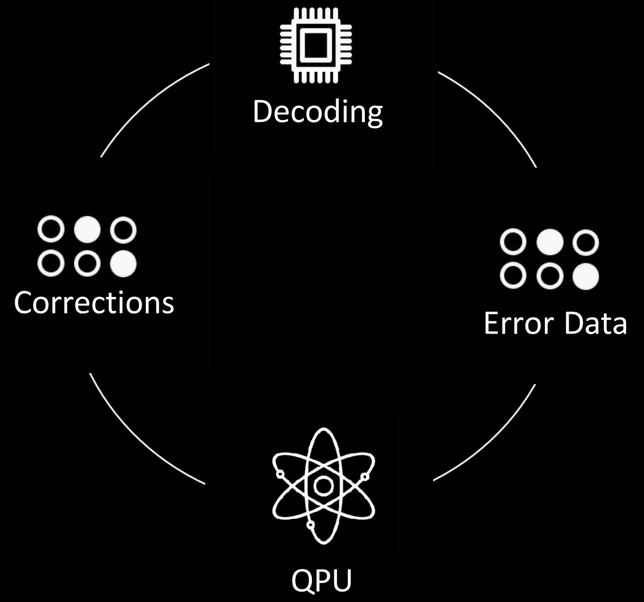


Faster training times

Increased model accuracy

Ying-Yi Hong, Dylan Josh Lopez, et al. "Accelerating Quantum Algorithms for Solar Energy Prediction with NVIDIA CUDA-Q and NVIDIA cuDNN" NVIDIA Technical Blog (2024).

Quantum Error Correction – The Challenge of Decoding



10^{-3}
State of the art error rates

$<10^{-10}$
Expected Error rates needed

> 1000
Logical qubits needed to factor a 2,048-bit number in > 100 hours

~ 1 million
Physical qubits needed with QEC

Qubits

Better QPUs

GPU

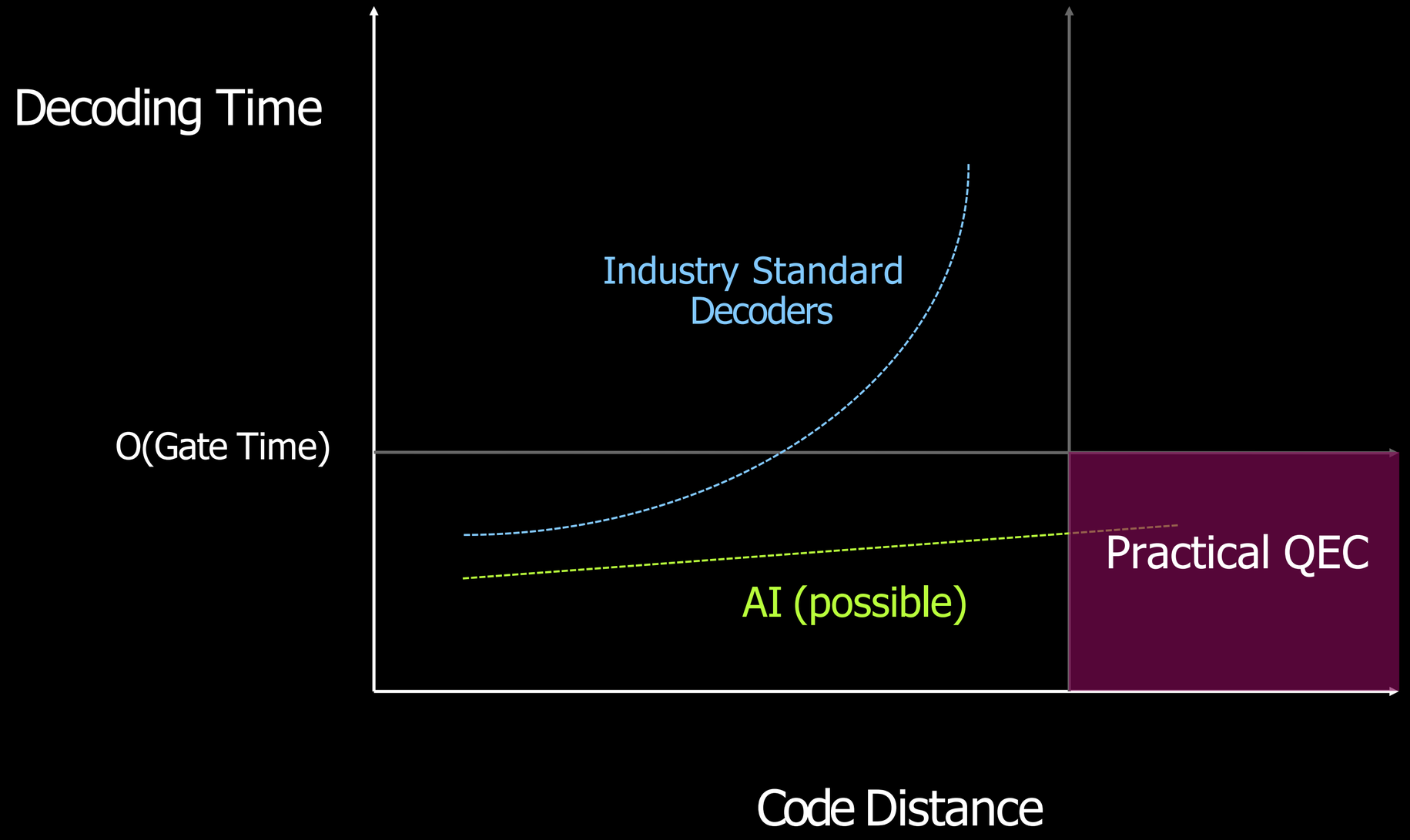
Better Algorithms

Scaling QEC

Accelerated Quantum Supercomputers

Craig Gidney. How to factor 2048 bit RSA integers with less than a million noisy qubits. https://www.arxiv.org/abs/2505.15917 (2025)

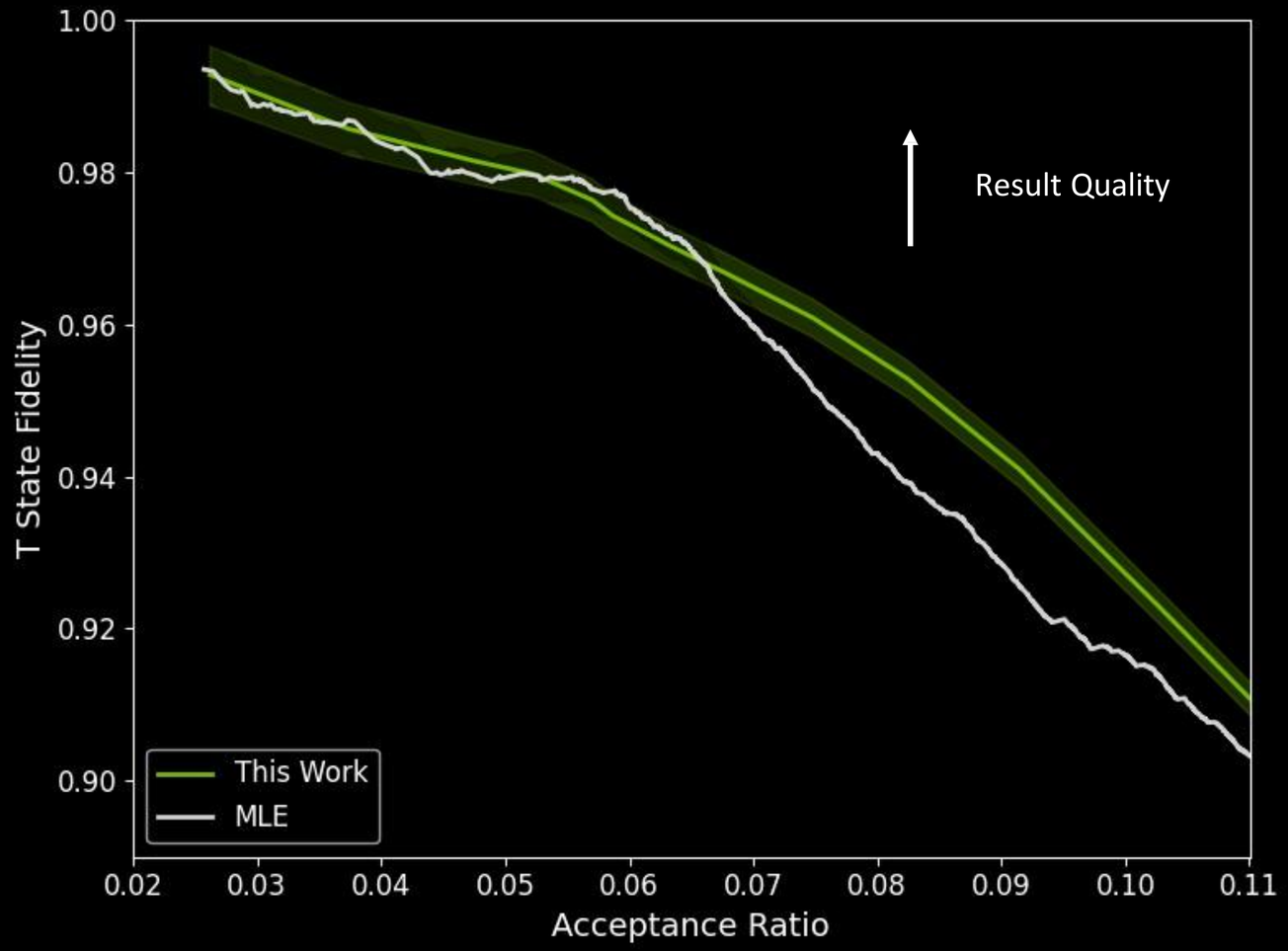
Quantum Error Correction – Why Quantum Computing Needs AI Supercomputing



Decoding QuEra's QPU with AI

Transformer-based decoder

- Trained on synthetic data with QuEra noise model
- Improved accuracy over MLE for small codes – d = 3
- Decoding time ~1ms, 50x improvement over SotA

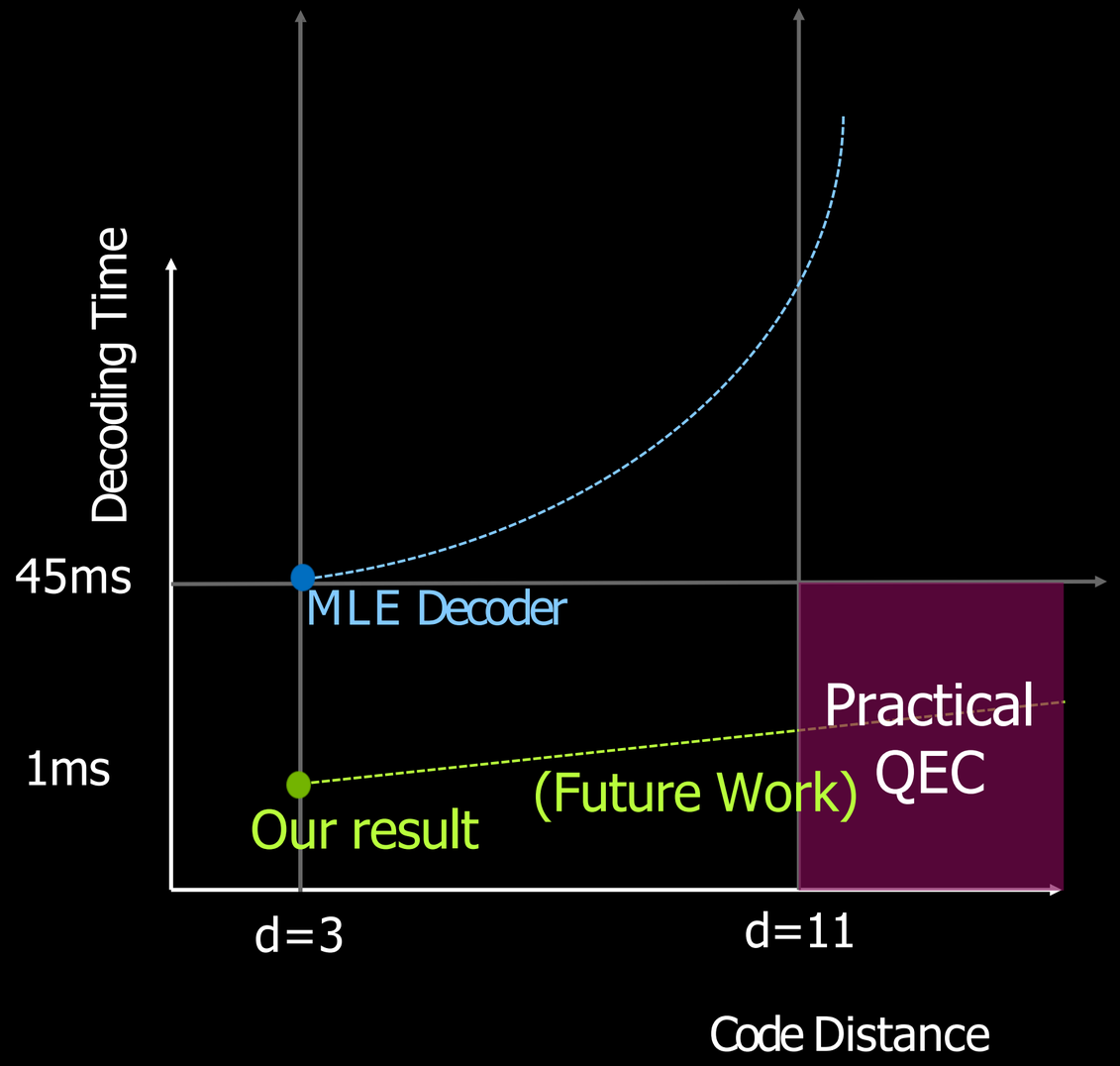


Decoding QuEra's QPU with AI

Transformer-based decoder

Future Work

- Larger codes – much more training data needed
- Fine-tune on QPU data for higher accuracy
- Deploy for real-time decoding



Let’s begin Hands-on