Leibniz Supercomputing Centre
of the Bavarian Academy of Sciences and Humanities

The Quantum Fourier Transform

From Mathematical Formulation to Quantum Circuits

Mario Hernández Vera

18.02.2026, Department of Quantum Computing and Technologies,
Leibniz Supercomputing Centre (LRZ)

Introduction

Why the QFT in CUDA-Q?

- **The Quantum Fourier Transform is a core building block in several applications:** Shor's Algorithm, Phase Estimation (QPE), and HHL.

https://quantumalgorithmzoo.org

Introduction

Why the QFT in CUDA-Q?

- **The Quantum Fourier Transform is a core building block in several applications:** Shor's Algorithm, Phase Estimation (QPE), and HHL.
- **Bridging Theory and Code:** Moving from the discrete math of state vectors to optimized CUDA-Q kernels.

https://quantumalgorithmzoo.org

Introduction

Why the QFT in CUDA-Q?

- **The Quantum Fourier Transform is a core building block in several applications:** Shor’s Algorithm, Phase Estimation (QPE), and HHL.
- **Bridging Theory and Code:** Moving from the discrete math of state vectors to optimized CUDA-Q kernels.
- **Performance at Scale:** Why SDK choice matters—leveraging CPU/GPU backends for simulation and execution.

https://quantumalgorithmzoo.org

Introduction

Why the QFT in CUDA-Q?

- **The Quantum Fourier Transform is a core building block in several applications:** Shor's Algorithm, Phase Estimation (QPE), and HHL.
- **Bridging Theory and Code:** Moving from the discrete math of state vectors to optimized CUDA-Q kernels.
- **Performance at Scale:** Why SDK choice matters—leveraging CPU/GPU backends for simulation and execution.
- **Learning by Doing:** Mastering CUDA-Q Python API through modular circuit design.

https://quantumalgorithmzoo.org

Roadmap of this section

📚 Quantum Circuit, Basis and Binary Number System

Quantum Fourier Transform

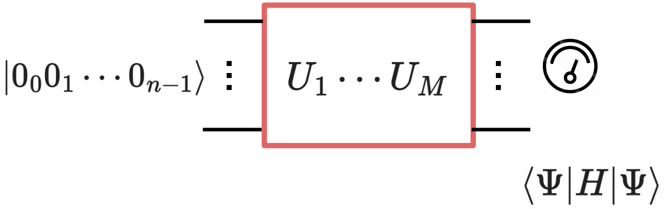
Quantum Phase Estimation

Implementations in CUDA-Q

Quantum Circuit Model

State Preparation and Measurements

Complexity of quantum algorithms



Simulation of circuits is difficult

$$|\Psi\rangle = \sum_{j_0 \cdots j_{n-1}} \underbrace{a_{j_0 \cdots j_{n-1}}}_{2^n} |j_0 j_1 \cdots j_{n-1}\rangle$$

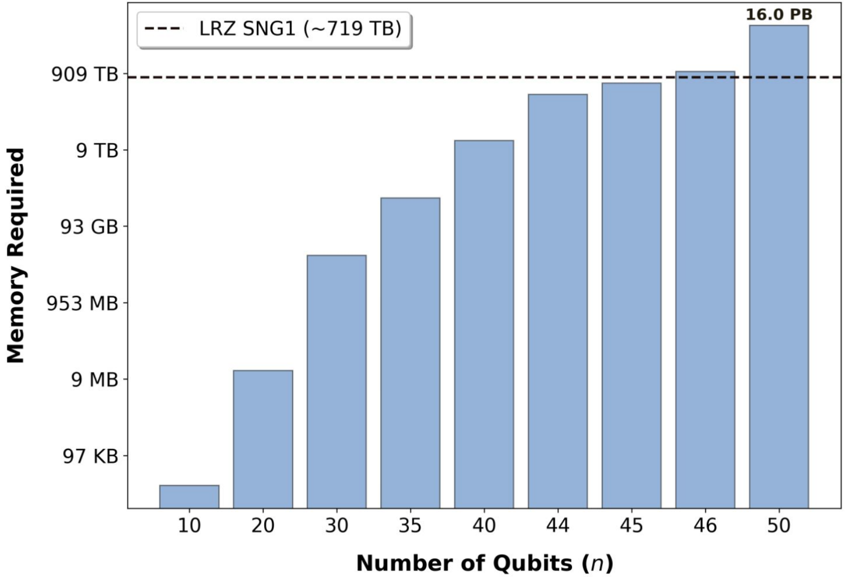


Fig. Memory required for state vector simulations of quantum circuits.

Computational Basis Representation

Basis and Binary Numbers

$$|\Psi\rangle = \sum_{j_0\cdots j_{n-1}} a_{j_0\cdots j_{n-1}} |j_0 j_1 \cdots j_{n-1}\rangle$$

Binary to decimal conversion:

$$j = j_0 2^{n-1} + j_1 2^{n-2} + \cdots j_{n-1} 2^0$$

$$|\Psi\rangle = \sum_{j=0}^{N-1} a_j |j\rangle$$

Example

Bell State

order-2 tensor

$$|\Psi\rangle = a_{00}|00\rangle + a_{11}|11\rangle$$

vector representation

$$|\Psi\rangle = a_0|0\rangle + a_3|3\rangle$$

Binary to decimal conversion:

$$1 \cdot 2^1 + 1 \cdot 2^0 = 3$$

$$0 \cdot 2^1 + 0 \cdot 2^0 = 0$$

Computational Basis Representation

Basis and Binary Numbers

$$|\Psi\rangle = \sum_{j_0 \cdots j_{n-1}} a_{j_0 \cdots j_{n-1}} |j_0 j_1 \cdots j_{n-1}\rangle$$

Elements of the computational basis can also represent *fractional values*

$$0.j_0 j_1 \cdots j_{n-1} = \frac{j_0}{2} + \frac{j_1}{4} + \cdots \frac{j_{n-1}}{2^n}$$

simplified notation of a fraction

Example

Transform the bits of a computational basis formed by 3 qubits to a fraction:

$$|010\rangle \longrightarrow 0.010 \equiv \frac{0}{2} + \frac{1}{4} + \frac{0}{8} = 0.25$$

😇 don't panic now if you see this

$$\exp(2\pi\, 0.j_0\, j_1\, j_2) \equiv \exp(2\pi\, \theta)$$

Mathematical Definition of the QFT

Action of QFT on Basis States

$$|j\rangle \longrightarrow \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{2\pi i jk/N} |k\rangle$$

💡 Because the QFT is linear, knowing how the basis transforms tells us how any general function transforms.

Mathematical Definition of the QFT

Action of QFT on Basis States

$$|j\rangle \longrightarrow \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{2\pi ijk/N} |k\rangle$$

💡 Because the QFT is linear, knowing how the basis transforms tells us how any general function transforms.

Transformation of the expansion coefficients

$$\begin{cases} [0, \cdots, 1, \cdots, 0] \rightarrow [y_0, y_1, \cdots, y_{N-1}] \\ y_k = \frac{1}{\sqrt{N}} e^{2\pi ijk/N} \end{cases}$$

💡 Coefficients transform similarly to the classical Discrete Fourier Transform

Mathematical Definition of the QFT

Action of QFT on Basis States

QFT on a basis results in tensor product of qubit states. Here binary fractions are formed from the bits associated to the transformed computational basis | j ⟩.

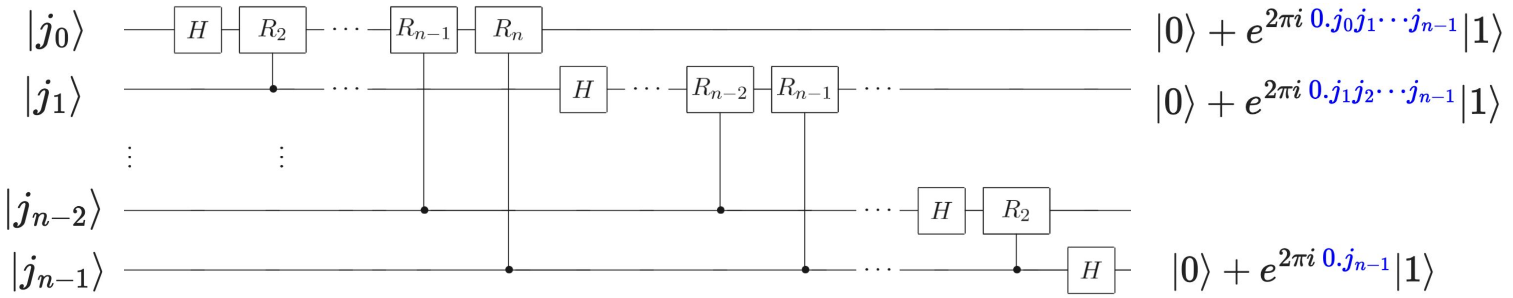
$$|j\rangle \longrightarrow \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{2\pi i jk/N} |k\rangle$$

$$\longrightarrow 1/\sqrt{N}\ \underbrace{\left(|0\rangle + e^{2\pi i\, 0.j_0 j_1 \cdots j_{n-1}}|1\rangle\right)}_{\text{qubit 0}} \underbrace{\left(|0\rangle + e^{2\pi i\, 0.j_1 j_2 \cdots j_{n-1}}|1\rangle\right)}_{\text{qubit 1}} \cdots \underbrace{\left(|0\rangle + e^{2\pi i\, 0.j_{n-1}}|1\rangle\right)}_{\text{qubit (n-1)}}$$

Nielsen & Chuang, *Quantum Comp. & Quantum Info.* (2010)

Mathematical Definition of the QFT

Circuit for the Quantum Fourier Transform



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

Hadamard gate
(denoted h in CUDA-Q)

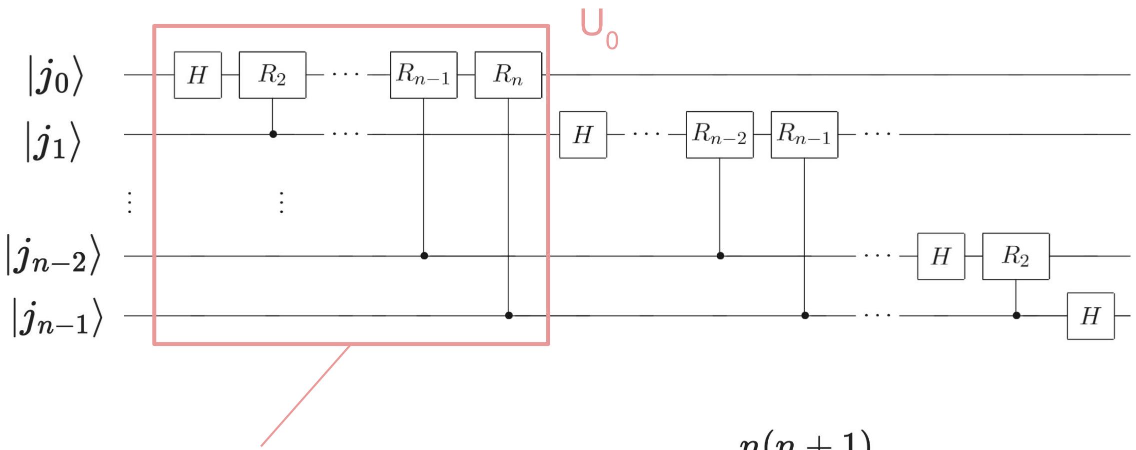
$$R_k \equiv \begin{bmatrix} 1 & 0 \\ 0 & e^{2\pi i/2^k} \end{bmatrix}$$

Single qubit phase shift
(denoted r1 in CUDA-Q)

Operations used in the circuit

Mathematical Definition of the QFT

Complexity Analysis of the Quantum Fourier Transform

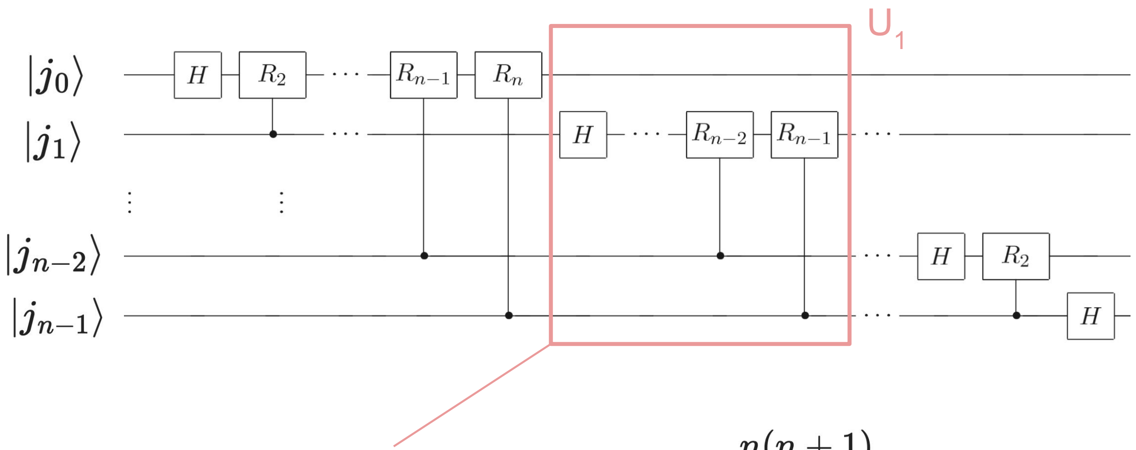


Gate count: $n + (n-1) + (n-2) + \cdots 1 = \dfrac{n(n+1)}{2}$

Complexity: $O(n^2)$

Mathematical Definition of the QFT

Complexity Analysis of the Quantum Fourier Transform

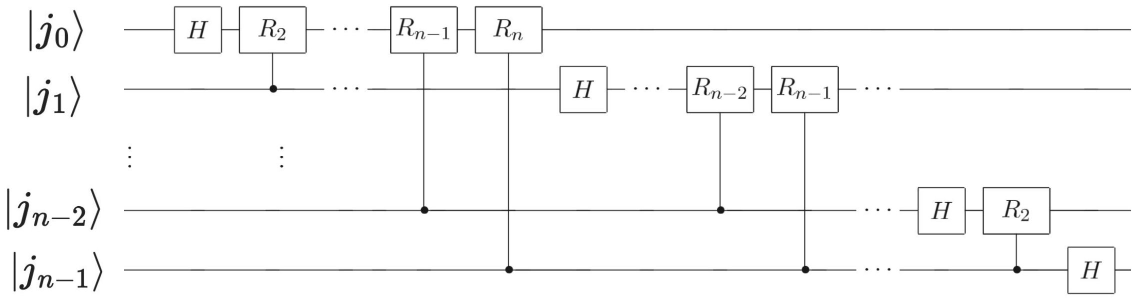


Gate count: $n + (n-1) + (n-2) + \cdots 1 = \frac{n(n+1)}{2}$

Complexity: $O(n^2)$

Mathematical Definition of the QFT

Complexity Analysis of the Quantum Fourier Transform



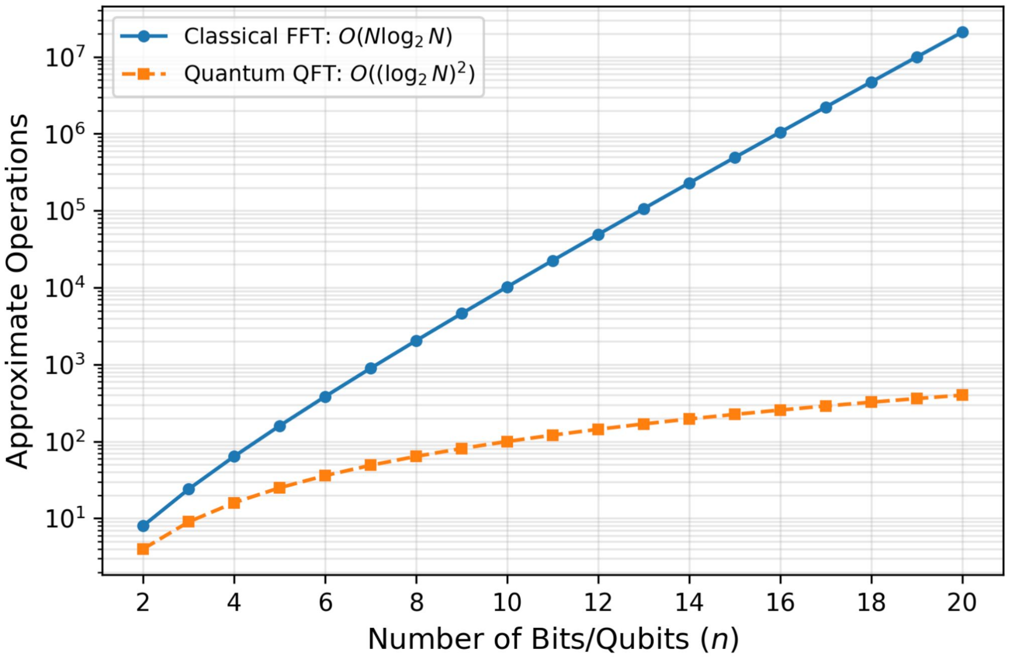
Gate count: $n + (n-1) + (n-2) + \cdots 1 = \frac{n(n+1)}{2}$

$U_{n-1} \cdots U_0$

Complexity: $O(n^2)$

Mathematical Definition of the QFT

Complexity Analysis of the Quantum Fourier Transform



💡 We have not included sample complexity. What if we use an algorithm that ideally does not requires too many samples?

Fig. Scaling Comparison: Classical FFT vs. Quantum QFT

Applications of QFT

The Quantum Phase Estimation

The Quantum Phase Estimation

Statement of the Problem

Given *U* and one of its eigenstates:

$$U|u\rangle = e^{2\pi i\phi}|u\rangle$$

goal: compute the phase

$$\phi = 0.j_0\,j_1\ \cdots\ j_{t-1}$$

$$|\phi\rangle \equiv |j\rangle = |j_0 j_i \cdots j_{n-1}\rangle$$

💡 the **basis** that provides the best n-bit approximation of the phase.

The Quantum Phase Estimation

Statement of the Problem

Given *U* and one of its eigenstates:

$$U|u\rangle = e^{2\pi i \phi}|u\rangle$$

goal: compute the phase

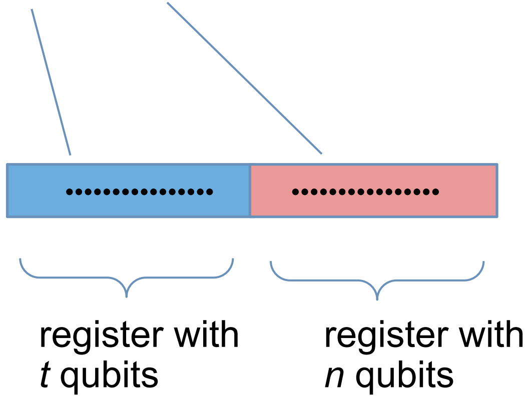
$$\phi = 0.j_0\, j_1\, \cdots\, j_{t-1}$$

$$|\phi\rangle \equiv |j\rangle = |j_0 j_i \cdots j_{n-1}\rangle$$

💡 the **basis** that provides the best n-bit approximation of the phase.

QPE works with 2 registers in the circuit

$$|0\rangle \otimes |u\rangle \longrightarrow |\phi\rangle \otimes |u\rangle$$

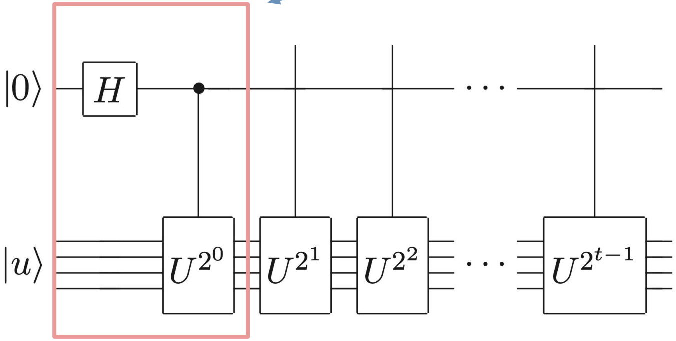


The Quantum Phase Estimation (QPE)

Circuit for the Quantum Phase Estimation

$$CU\left(H|0\rangle \otimes |u\rangle\right) = CU\left[\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes |u\rangle\right]$$

$$= \frac{1}{\sqrt{2}}\left(|0\rangle + e^{2\pi i\phi}|1\rangle\right) \otimes |u\rangle$$



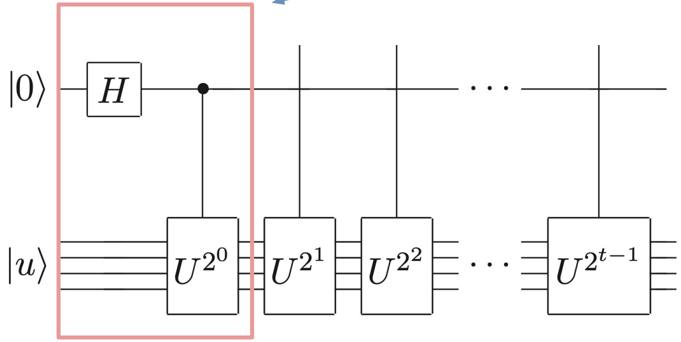
💡 this *phase kickback* operation effectively transfers the phase from the second register to the first register.

The Quantum Phase Estimation (QPE)

Circuit for the Quantum Phase Estimation

$$CU\left(H|0\rangle \otimes |u\rangle\right) = CU\left[\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes |u\rangle\right]$$

$$= \frac{1}{\sqrt{2}}\left(|0\rangle + e^{2\pi i\phi}|1\rangle\right) \otimes |u\rangle$$



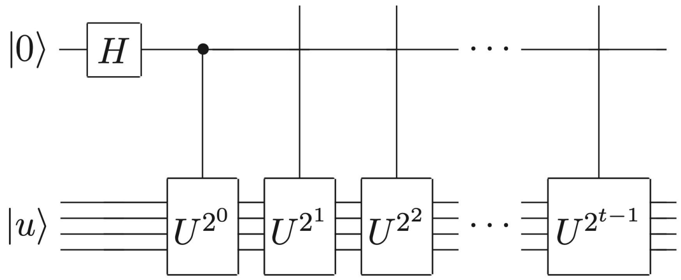
$$|0\rangle + e^{2\pi i\, 0.j_0 j_1 \cdots j_{t-1}}|1\rangle$$

💡 this *phase kickback* operation effectively transfers the phase from the second register to the first register.

$$|j\rangle \equiv |\phi\rangle = |j_0\, j_1\, \cdots\, j_{t-1}\rangle$$

The Quantum Phase Estimation (QPE)

Circuit for the Quantum Phase Estimation

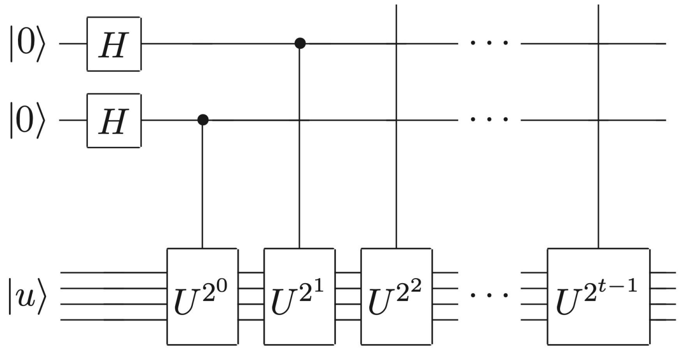


$$|0\rangle + e^{2\pi i\, 0.j_0 j_1 \cdots j_{t-1}}|1\rangle$$

$$|j\rangle \equiv |\phi\rangle = |j_0\, j_1\, \cdots\, j_{t-1}\rangle$$

The Quantum Phase Estimation (QPE)

Circuit for the Quantum Phase Estimation



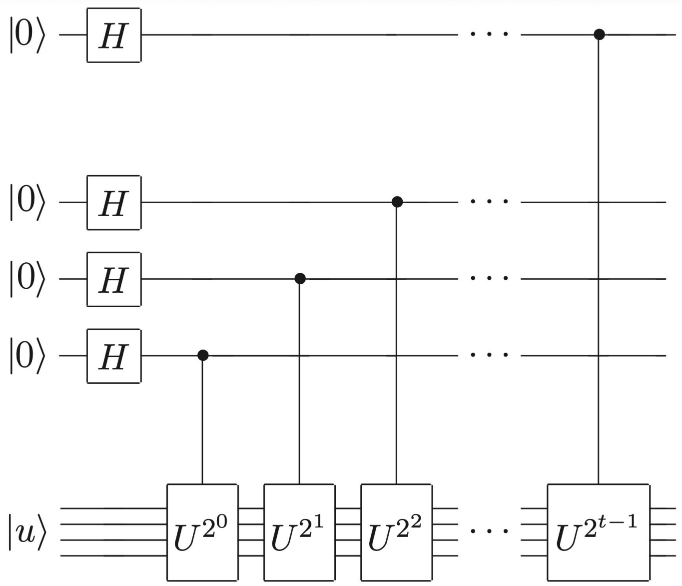
$$|0\rangle + e^{2\pi i \, 0.j_1 \cdots j_{t-1}}|1\rangle$$

$$|0\rangle + e^{2\pi i \, 0.j_0 j_1 \cdots j_{t-1}}|1\rangle$$

$$|j\rangle \equiv |\phi\rangle = |j_0 \, j_1 \, \cdots \, j_{t-1}\rangle$$

The Quantum Phase Estimation (QPE)

Circuit for the Quantum Phase Estimation



$$|0\rangle + e^{2\pi i\, j_{t-1}}|1\rangle$$

$$|0\rangle + e^{2\pi i\, 0.j_2\cdots j_{t-1}}|1\rangle$$

$$|0\rangle + e^{2\pi i\, 0.j_1\cdots j_{t-1}}|1\rangle$$

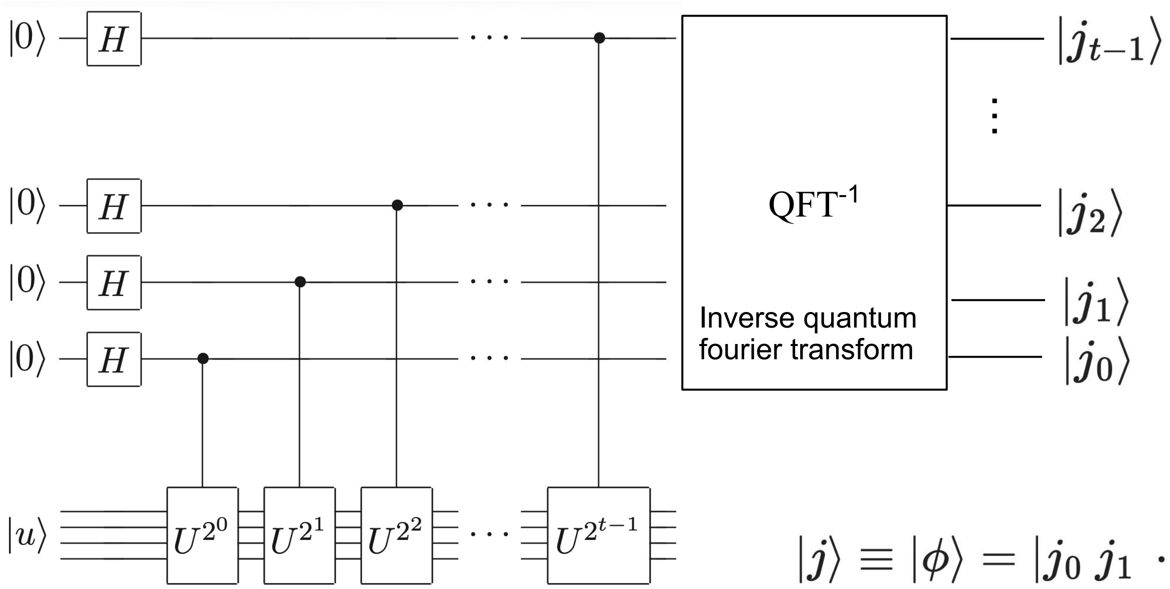
$$|0\rangle + e^{2\pi i\, 0.j_0 j_1\cdots j_{t-1}}|1\rangle$$

💡 This is the quantum fourier transform of *j* - the basis that describes the phase.

$$|j\rangle \equiv |\phi\rangle = |j_0\, j_1\, \cdots\, j_{t-1}\rangle$$

The Quantum Phase Estimation (QPE)

Circuit for the Quantum Phase Estimation



💡 Measurements should indicate the basis *j* as the most probably outcome.

$$|j\rangle \equiv |\phi\rangle = |j_0\, j_1\, \cdots\, j_{t-1}\rangle$$

The Quantum Phase Estimation (QPE)

The QPE in Action: Sampling Eigenvalues

$$e^{-iHt}|\Phi_n\rangle = e^{-i\epsilon_n t}|\Phi_n\rangle = e^{2\pi i\phi_n}|\Phi_n\rangle$$

Imagine that we want the ground state eigenvalue of a Hamiltonian (*H*).

The Quantum Phase Estimation (QPE)

The QPE in Action: Sampling Eigenvalues

$$e^{-iHt}|\Phi_n\rangle = e^{-i\epsilon_n t}|\Phi_n\rangle = e^{2\pi i\phi_n}|\Phi_n\rangle$$

Imagine that we want the ground state eigenvalue of a Hamiltonian (*H*).

$$|\tilde{\Psi}\rangle \approx \sum_m |\Phi_m\rangle$$

And that we have a method to construct a “good” Ansatz - overlap with the ground state.

The Quantum Phase Estimation (QPE)

The QPE in Action: Sampling Eigenvalues

$$e^{-iHt}|\Phi_n\rangle = e^{-i\epsilon_n t}|\Phi_n\rangle = e^{2\pi i\phi_n}|\Phi_n\rangle$$

$$|\tilde{\Psi}\rangle \approx \sum_m |\Phi_m\rangle$$

$$\mathbf{QFE} \cdot |0\rangle \otimes |\tilde{\Psi}\rangle = \sum_m a_m|\phi_m\rangle \otimes |\Phi_m\rangle$$

Imagine that we want the ground state eigenvalue of a Hamiltonian (*H*).

And that we have a method to construct a “good” Ansatz - overlap with the ground state.

💡 QPE will “sample” with some probability the ground states phase associated with the eigenvalue!

The Quantum Phase Estimation (QPE)

The QPE in Action: Sampling Eigenvalues

$$e^{-iHt}|\Phi_n\rangle = e^{-i\epsilon_n t}|\Phi_n\rangle = e^{2\pi i\phi_n}|\Phi_n\rangle$$

$$|\tilde{\Psi}\rangle \approx \sum_m |\Phi_m\rangle$$

$$\mathbf{QFE}\cdot|0\rangle\otimes|\tilde{\Psi}\rangle = \sum_m a_m|\phi_m\rangle\otimes|\Phi_m\rangle$$

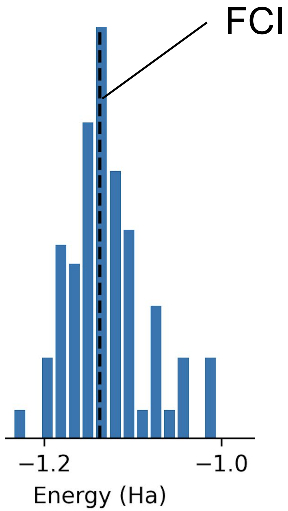


Fig. Sampling the ground state energy of the H2 molecule using STO-3G basis.

PRX QUANTUM 5, 040339 (2024)

Summary

- **Foundations:** Reviewing quantum circuits, basis states, and binary mapping.
- **The QFT:** Exploring the mechanics of the Quantum Fourier Transform.
- **Applications:** Leveraging the QFT for Quantum Phase Estimation (QPE).

👩🏻‍💻 Now let’s move from theory to implementation with CUDA-Q!