



# A Scalability Study of Quantum Algorithms for Dimensionality Reduction of Multidimensional Data

Kareem El-Araby<sup>1</sup>, Doru Thom Popovici<sup>2</sup>, Alvir Nobel<sup>1</sup>, Sunday Ogundele<sup>1</sup>, Katherine Klymko<sup>2</sup>, Daan Camps<sup>2</sup>, Anastasiia Butko<sup>2</sup>, and Esam El-Araby<sup>1</sup>

<sup>1</sup>The University of Kansas, <sup>2</sup>Lawrence Berkeley National Laboratory

{kareem\_el\_araby,islam.alvir,sunday,esam}@ku.edu  
{dtpopovici,kklymko,dcamps,abutko}@lbl.gov



Office of Science



## Abstract

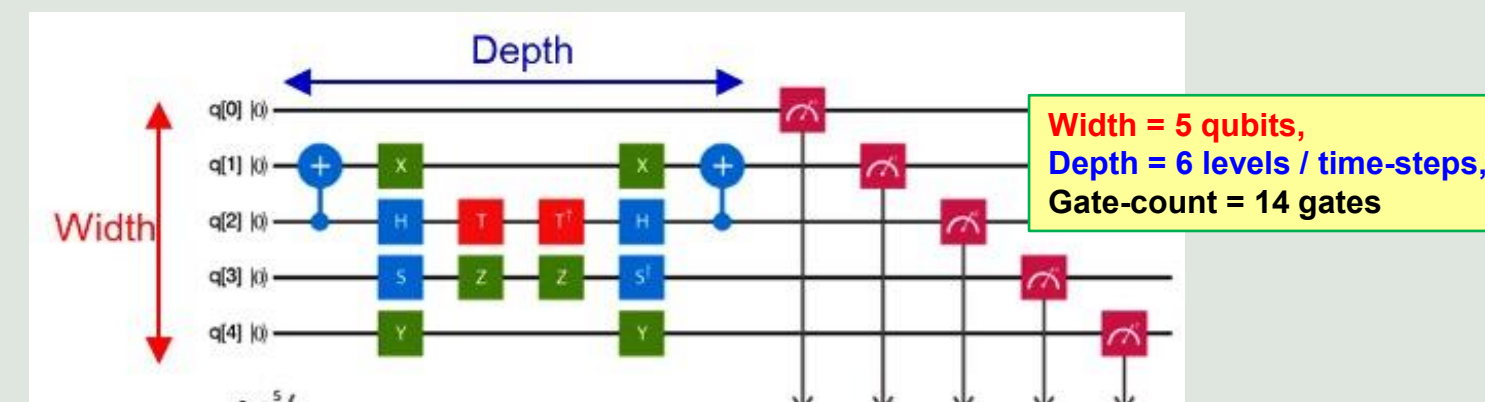
- Quantum computing promises **exponential speedups** for problems in cryptography, search, and linear algebra by leveraging quantum mechanical properties like **superposition** and **entanglement**.
- As quantum algorithms grow in complexity, **classical simulation** remains essential for evaluating **correctness**, **scalability**, and **resource demands**.
- This work focuses on studying the **scalability** of structured quantum algorithms such as the **Quantum-Haar-Transform (QHT)** usually used for reducing data dimensionality in signal/image processing and remote sensing hyperspectral imagery.
- We simulate QHT circuits on **High-Performance-Computing (HPC)** systems by constructing full unitary models that mirror the transform's **hierarchical decomposition**.
- Simulations track performance metrics such as **circuit width**, **circuit depth**, and **execution time**.
- Our results provide insight into the practical implementation of structured quantum circuits and serve as a reference for **validating algorithmic correctness** and guiding future quantum algorithm design.

## Background

### How do we apply computation on qubits?

- Use **quantum gates** (unitary matrices) to evolve the state vector
- Gates are applied in **quantum circuits** via matrix-vector multiplication
- Final **measurement** collapses superposition to classical output

Gate	Symbol	Matrix	Gate	Symbol	Matrix
Hadamard		$\frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$	Controlled Rotation x		$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\theta) & 0 & 0 \\ 0 & 0 & \cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
Pauli X		$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$	Controlled Rotation y		$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\theta) & 0 & 0 \\ 0 & 0 & \cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
Swap		$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$	Controlled Rotation z		$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\theta) & 0 & 0 \\ 0 & 0 & \cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
Controlled CZ		$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}$	Controlled phase shift		$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & e^{i\theta} \end{bmatrix}$
Controlled NOT		$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$			
Controlled Pauli Z		$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}$			



### What are the advantages of quantum computers?

- Enables **massive parallelism** during computation
- n qubits** encode  $2^n$  **data points** (amplitudes), enabling compact representation of high-dimensional data

### How to simulate quantum algorithms efficiently?

- Efficient simulation of quantum algorithms can be achieved using **highly parallel architectures** with large memory bandwidth, such as multi-GPU systems.

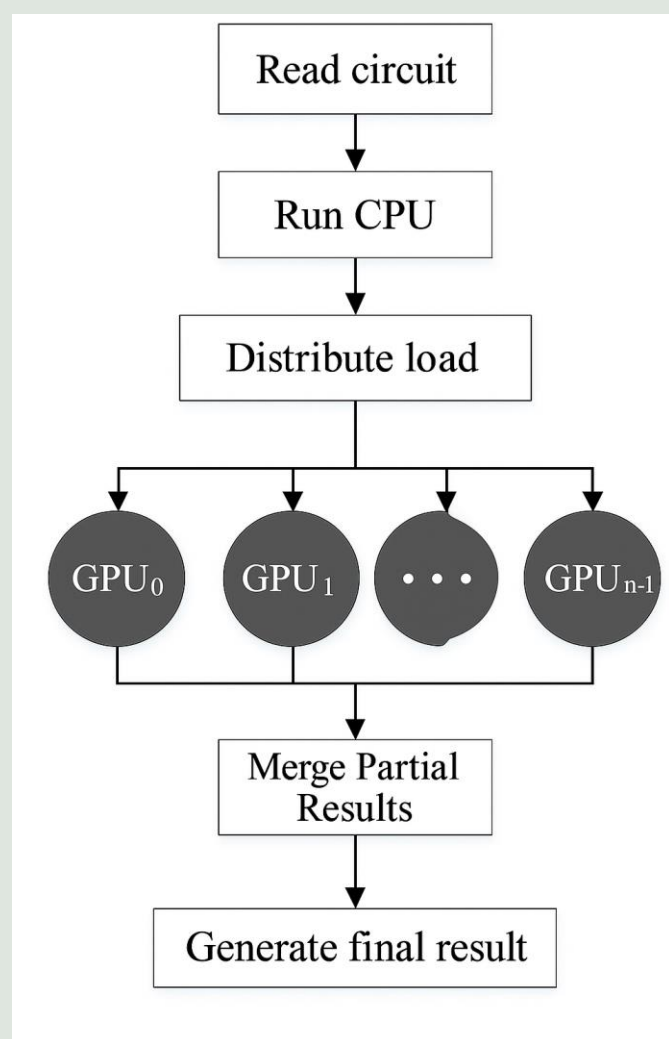


Figure 1: Quantum Gate Definitions (a) Example circuit showing depth and width (b) Distributed Quantum Simulation Model

## Related Work

### Qibo [1]

- Utilizing multi-threading CPUs, single GPUs, and multi-GPU accelerators

### Multi-Shot [2]

- Using batch execution and shot-branching to optimize multi-shot quantum computing simulations on GPUs

### QuEST [3]

- Hybrid GPU-accelerated simulator designed for universal quantum circuits that can handle pure and mixed states

### Reconfigurable Emulation [4]

- Reconfigurable emulation of quantum algorithms focusing on achieving high precision and high throughput

### Quantum Rings [5]

- High-performance simulation framework for complex quantum circuit simulation with GPU acceleration.

## Methodology

### Quantum Haar Transform (QHT)

- Parallel**  $d$ -dimensional wavelet transform
- Decomposes multidimensional data in **multiple decomposition levels**
- Input data separated into **low** and **high frequency** components/replicas
- Retains spatial and temporal locality of the input data **without significant data loss**

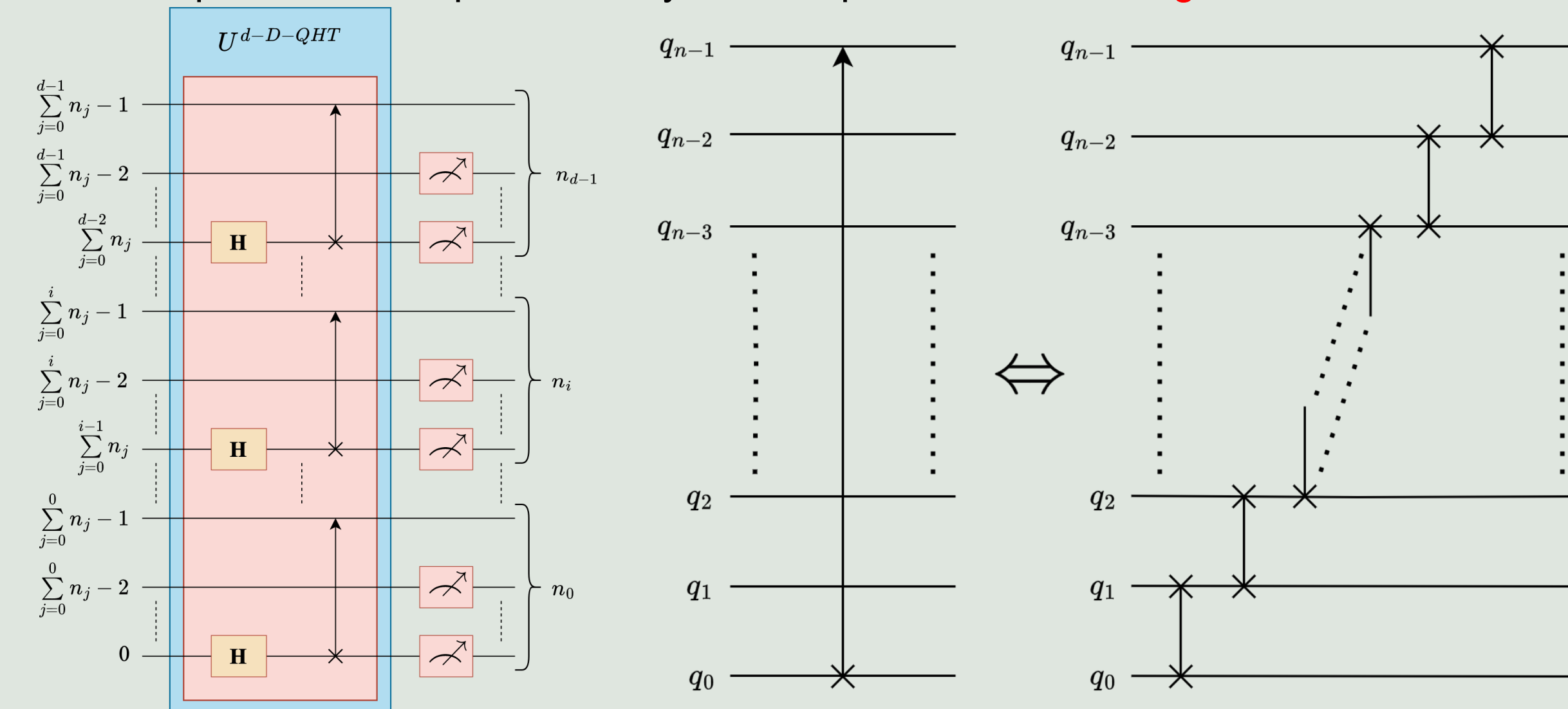


Figure 2: Diagram of the QHT circuit for  $d$ -dimensions (left) and a diagram of the Rotate-Right gate (RoR) using swap gates (right)

### Simulating Quantum Circuits

- Goal of this work is to find an **efficient** way to implement quantum algorithms on a computational device capable enough to test the algorithm's **scalability**
- Utilizing the computational power of the **Perlmutter** HPC system to test the algorithms
- Intermediate steps for **simulation** on Perlmutter were developed for the study

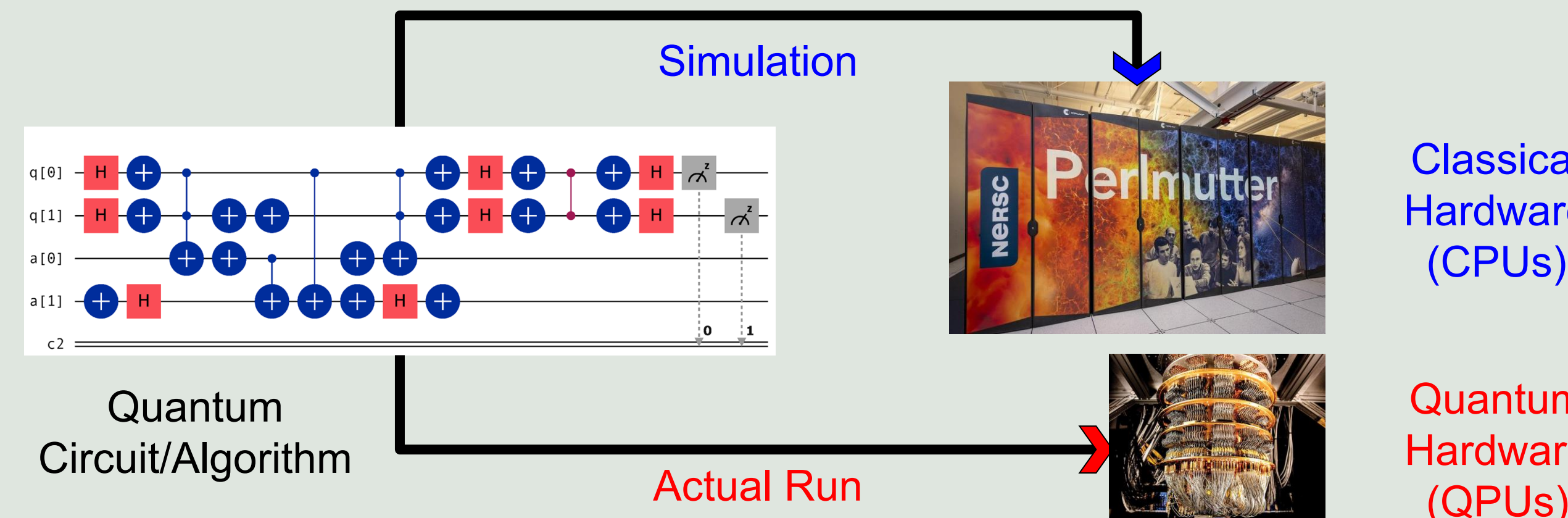


Figure 3: Flowchart representing different paths for implementing quantum circuits (simulation vs actual run)

### Implementation

- Quantum gates making up a circuit are nothing but **matrices**
- The matrices grow too massive for **general matrix multiplication** to be viable
- Exploiting their **sparsity** we can make simple loop structures for the circuits

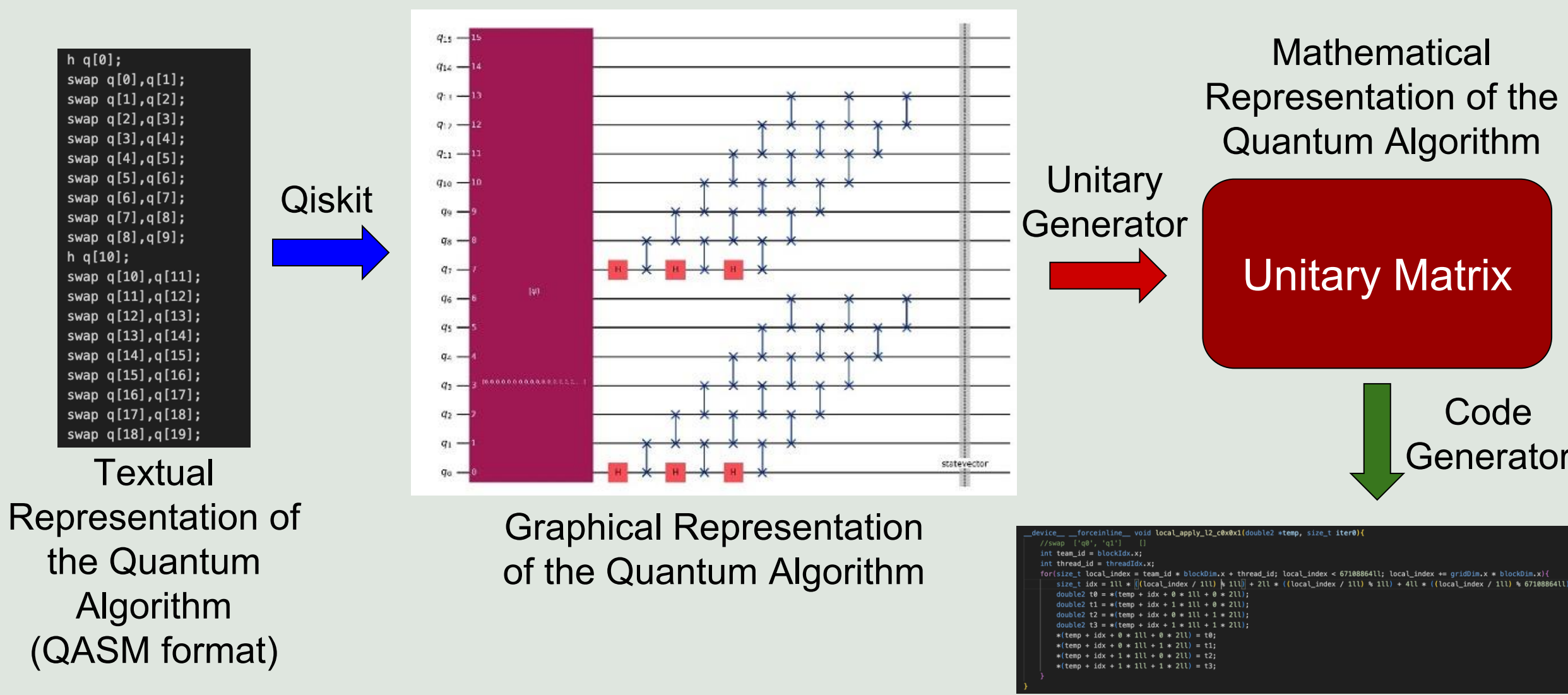
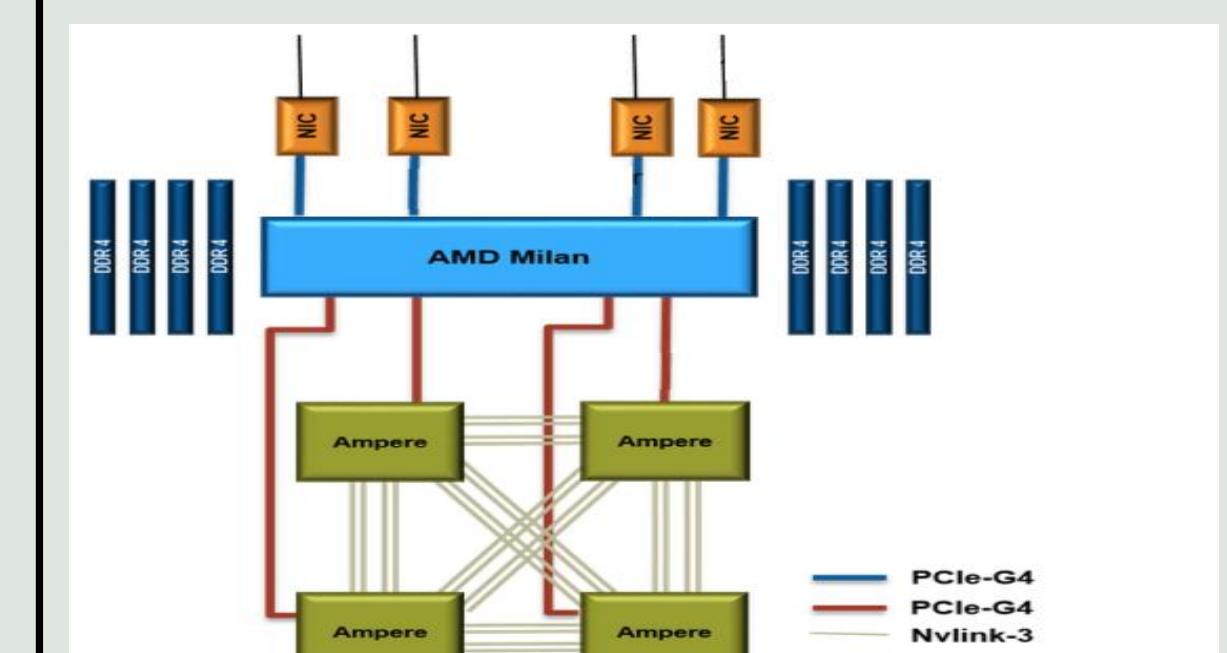


Figure 4: Different representations of the quantum algorithm

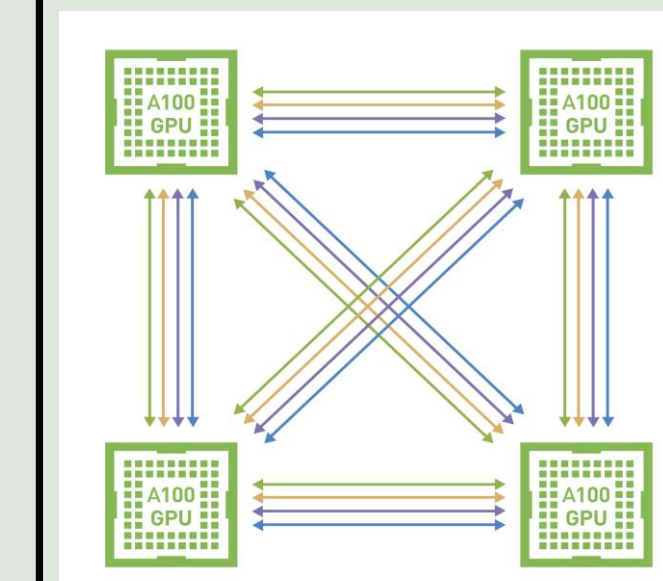
## Experimental Results

### Experimental Setup

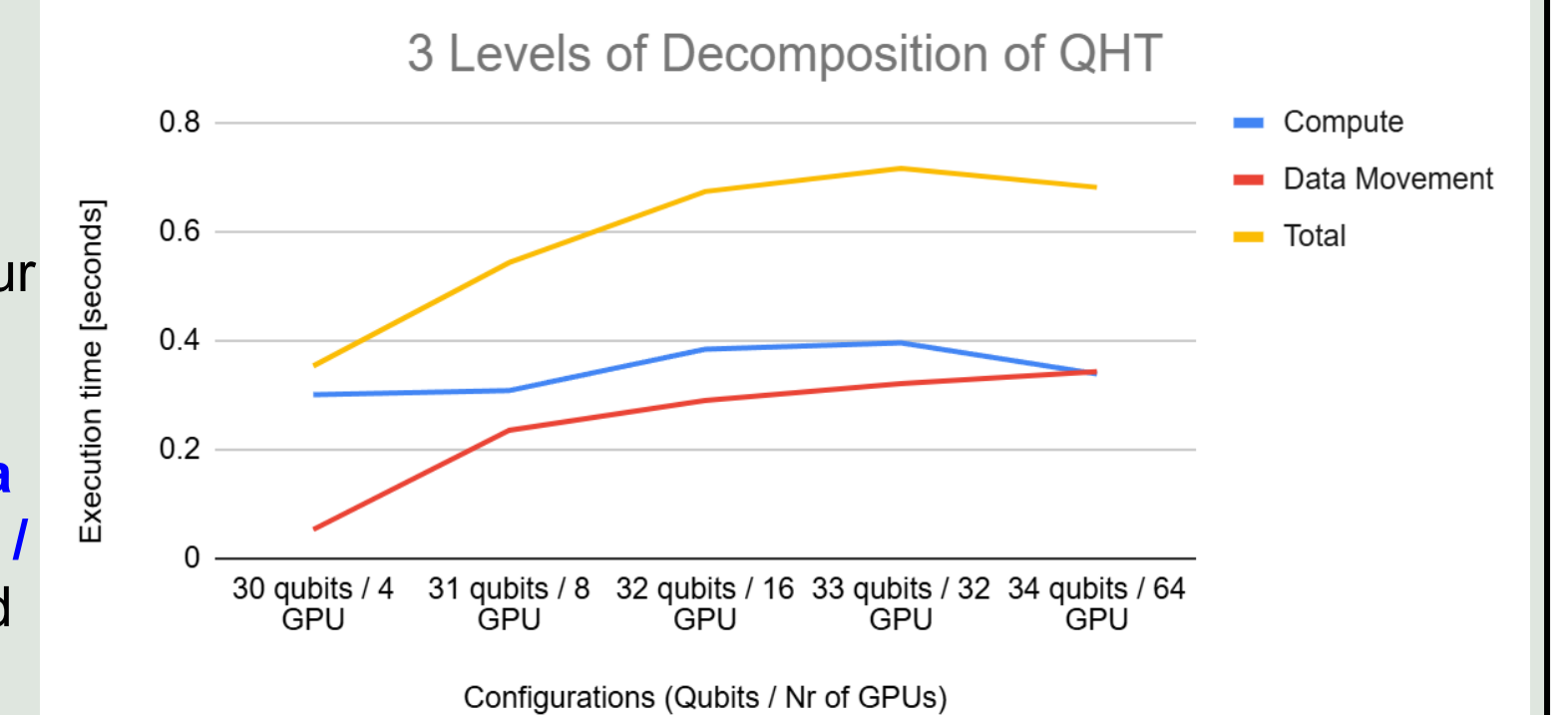
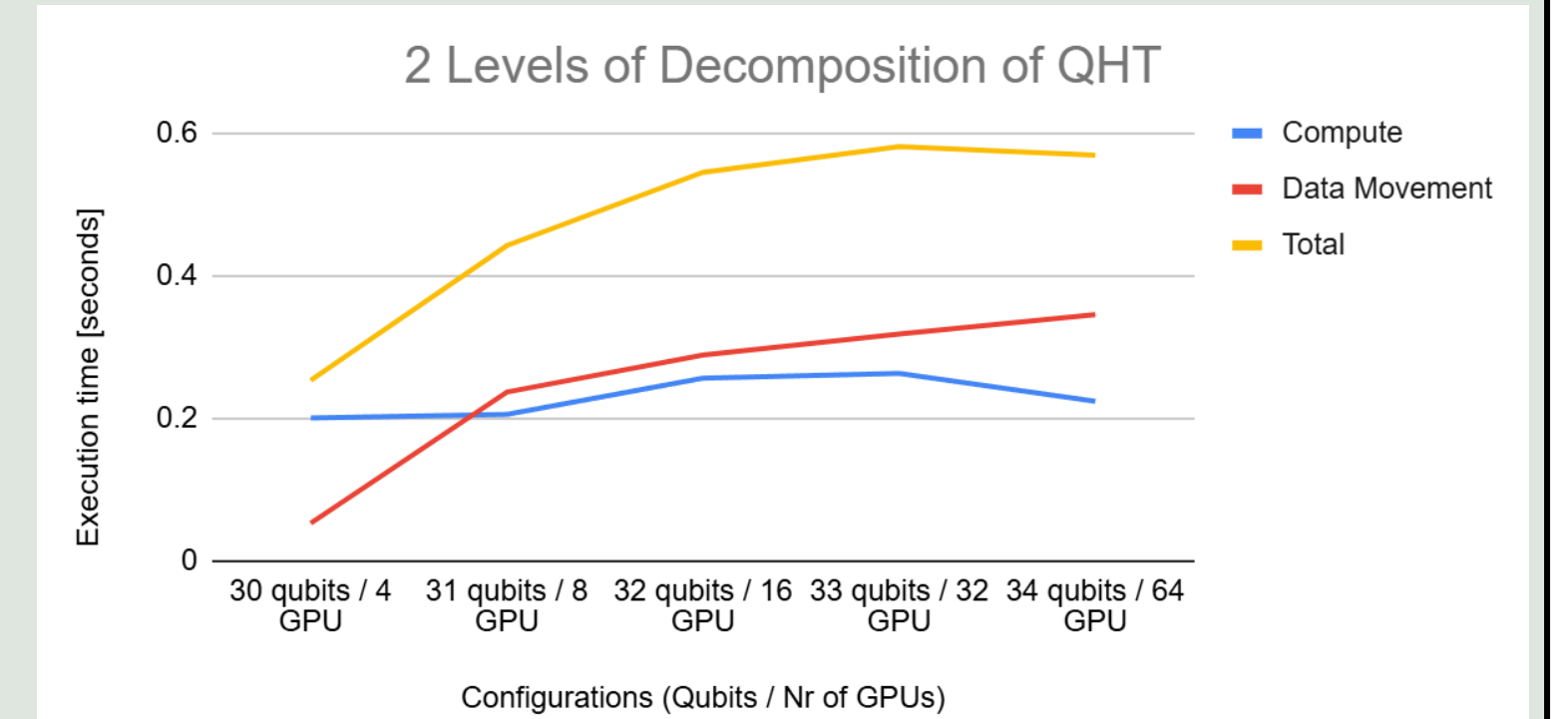
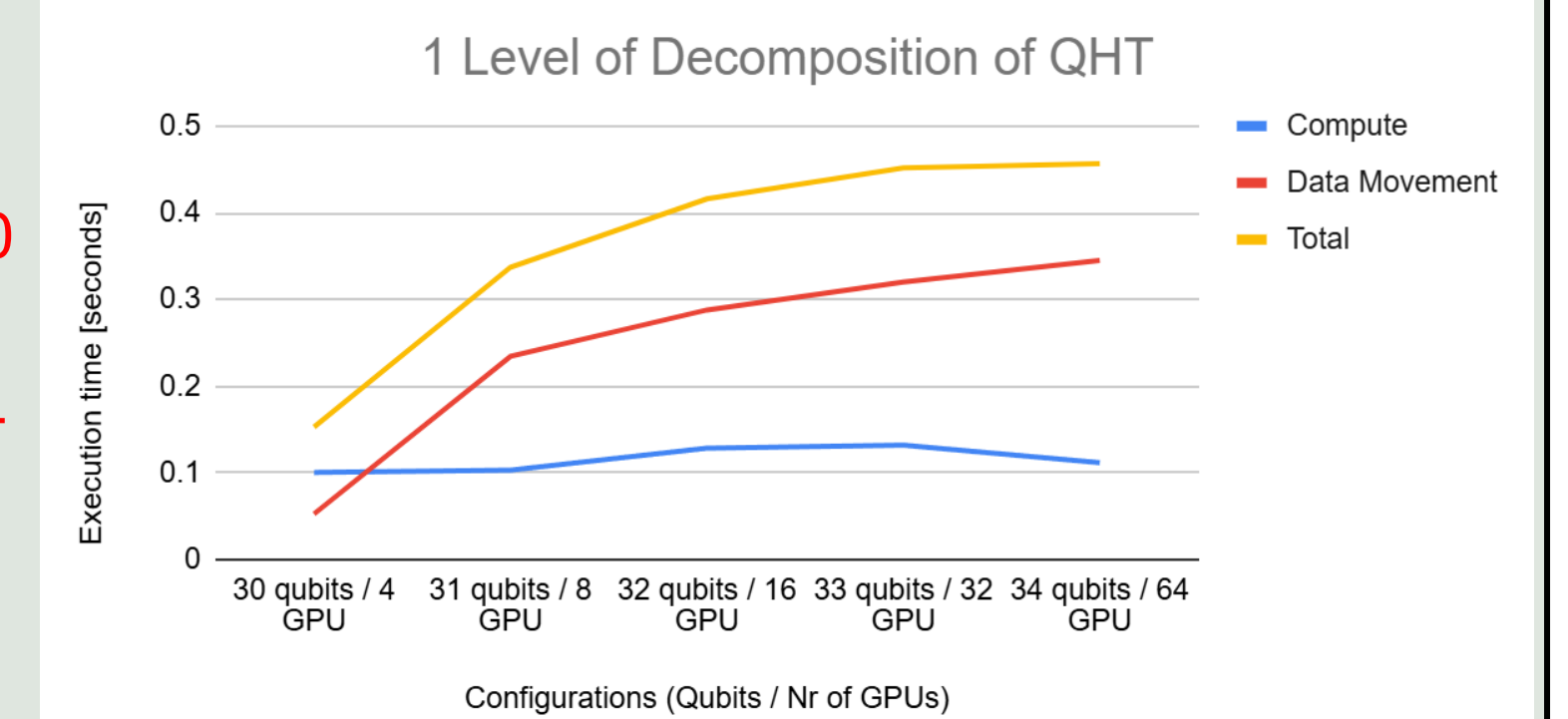
- Perlmutter** HPC system
- Each node consists of 4 **NVIDIA A100** GPUs
- Simulated QHT algorithm using **30-34 qubits** on **4-64 GPUs** on 3D data
- Weak scaling** conducted to see if the **execution time is constant** when we increase the number of GPUs and qubits



### Perlmutter system architecture



### Perlmutter node architecture



The experimental results show that our implementation is optimized with respect to the **data size (circuit width / No. of qubits)** and scales with a time complexity of  $\sim O(1)$

The experimental results also show that our implementation scales with a time complexity of  $\sim O(L)$  where "L" is number of **decomposition levels (circuit depth)** which could potentially be further optimized  $\rightarrow$  **(Future Work)**

Figure 5: Architecture of the Perlmutter's GPU nodes and how the data is transferred as well as the results of how the QHT algorithm performed

## Conclusions & Future Work

- In this work, we provide an analysis of the **scalability** and **performance** of the Quantum-Haar-Transform (QHT) when simulated using **Perlmutter** supercomputer.
- The simulation strategy employed enables testing of circuits well beyond the **size limitations** of current **quantum hardware**.
- These results not only validate the **scalability** of the algorithm but also serves as a reference for guiding future quantum algorithm design.
- Future work can benefit from **reducing communication latency** of the simulation engine and extending its support to other quantum algorithms with structured circuit representations.

## References

- [1] Efthymiou, S., Ramos-Calderer, S., Bravo-Prieto, C., Perez-Salinas, A., Garcia-Martín, D., Garcia-Saez, A., Latorre, J.I. and Carrazza, S., 2021. Qibo: a framework for quantum simulation with hardware acceleration. Quantum Science and Technology, 7(1), p.015018.
- [2] Horii, H. and Wood, C., 2023. Efficient techniques to gpu accelerations of multi-shot quantum computing simulations. arXiv preprint arXiv:2308.03399.
- [3] Jones, T., Brown, A., Bush, I. and Benjamin, S.C., 2019. QuEST and high performance simulation of quantum computers. Scientific reports, 9(1), p.10736.
- [4] Mahmud, N., El-Araby, E. and Caliga, D., 2019. Scaling reconfigurable emulation of quantum algorithms at high precision and high-throughput. Quantum Engineering, 1(2), p.e19.
- [5] Kasirajan, V., Battelle, T. and Wold, B., 2024. Empowering Large Scale Quantum Circuit Development: Effective Simulation of Sycamore Circuits. arXiv preprint arXiv:2411.12131.

## Acknowledgements

This work was supported in part by the U.S. Department of Energy, Office of Science, Office of Workforce Development for Teachers and Scientists (WDTS) under the Science Undergraduate Laboratory Internship (SULI) program. Special thanks to the Applied Mathematics and Computational Research Division (AMCR), the National Energy Research Scientific Computing Center (NERSC) and to the KU Advanced Reconfigurable Quantum (KUARQ) computing group.

